



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Escuela Técnica Superior de Ingeniería del Diseño

Automatización y control de un proceso de recogida de
huevos

Trabajo Fin de Grado

Grado en Ingeniería Electrónica Industrial y Automática

AUTOR/A: Casanova López, Ana María

Tutor/a: Saiz Jimenez, Juan Ángel

CURSO ACADÉMICO: 2021/2022

Agradecimientos

Quiero dar las gracias a mis padres, por haberme formado como persona y así llegar a quien soy hoy. También a mi novio Carlos, que me ha apoyado siempre, con cada decisión que he tomado. A mis amigas, porque, aunque no nos hayamos visto mucho siempre han estado ahí. Y por supuesto a mis compañeros de clase; han sido todo un apoyo tanto a nivel académico como a nivel personal y sin ellos las clases hubieran sido mucho más aburridas.

Dar las gracias también a mi tutor, que me ha enseñado muchas cosas sobre la vida, gracias, Juan Ángel. También me gustaría mencionar a aquellos profesores se han implicado en la gran responsabilidad que es la enseñanza y he tenido el placer de tratar con ellos: gracias, Rocío, gracias, Roser, gracias, Marce, gracias, Salva.

Resumen

Este proyecto consiste en la automatización y control de un proceso de recogida de huevos en una granja de gallinas. El objetivo del sistema implementado es controlar un proceso con alto grado de seguridad, para disminuir al máximo posible que se eche a perder el producto. Cualquier imprevisto precisará de la intervención de un operario, es por ello por lo que también se ha implementado un sistema de supervisión y control.

Para realizar todo lo mencionado anteriormente, se ha realizado la programación de un PLC y un sistema HMI, que, conectados ofrecen la posibilidad tanto de informar sobre el estado del proceso como de hacer modificaciones.

Resum

Aquest projecte consisteix en l'automatització i control d'un procés de recollida d'ous en una granja de gallines. L'objectiu del sistema implementat és controlar un procés amb alt grau de seguretat, per a disminuir al màxim possible que es faça malbé el producte. Qualsevol imprevist precisarà de la intervenció d'un operari, és per aiplataformaò que també s'ha implementat un sistema de supervisió i control.

Per a realitzar tot l'esmentat anteriorment, s'ha realitzat la programació d'un PLC i un sistema HMI, que, connectats ofereiplataformaen la possibilitat tant d'informar sobre l'estat del procés com de fer modificacions.

Abstract

This project consists of the automation and control of an egg collection process in a hen farm. The objective of the implemented system is to control a process with a high degree of safety, in order to reduce as much as possible, the spoilage of the product. Any unforeseen event will require the intervention of an operator, which is why a supervision and control system has also been implemented.

In order to carry out all the above, a PLC and an HMI system have been programmed, which, when connected, offer the possibility of both reporting on the status of the process and making modifications.

Índice general

I	Memoria	1
II	Planos	147
III	Pliego de condiciones	155
IV	Presupuesto	169
	Bibliografía	177

Documento I

Memoria

Índice

Índice	3
Índice de figuras	7
Índice de tablas	9
Lista de código fuente	11
1 Objeto	13
2 Antecedentes	15
2.1 Legislación	17
2.2 Información técnica	17
2.3 Proyectos similares	18
3 Estudio de necesidades	19
3.1 Lógica	19
3.2 Visualización	20
3.3 Simulación	20
3.4 Escalabilidad	20
4 Estudio de alternativas	21
4.1 Breve descripción de alternativas	21
4.1.1 Automatización	21
4.1.2 Visualización	23
4.1.3 Simulación	24
4.2 Criterio de selección	25
4.3 Justificación de la solución adoptada	25
4.3.1 Lógica	25
4.3.2 Visualización	25
4.3.3 Simulación	25
5 Descripción detallada de la solución adoptada	27
5.1 Simulación	27
5.2 Programación	30
5.2.1 GRAFCET	31
5.2.1.1 GRAFCET superior	33
5.2.1.2 F2	34
5.2.1.3 F1	36

5.2.1.4	F3	36
5.2.1.5	F4	38
5.2.2	Implementación	38
5.2.2.1	Ecuaciones de Activación	40
5.2.2.2	Ecuaciones de salida	42
5.3	HMI	44
5.3.1	Pantalla principal	44
5.3.2	Pantalla vista simbólica	46
5.3.3	Pantalla Fallo Guardamotores y Fallo EDM	46
5.3.4	Pantalla Alarmas Activas y Alarmas Históricas	47
Apéndices		48
A Tabla variables Factory I/O		51
B Programación PLC		55
B.1	Main [OB1]	56
B.2	Simulación [FC1]	57
B.3	Asignación Entradas [FC4]	61
B.4	Ecuaciones de activación [FC6]	63
B.4.1	Error crítico	63
B.4.2	Condiciones iniciales	63
B.4.3	Fin de ciclo reset	64
B.4.4	Comprobación desorden reset	64
B.4.5	Último huevo	64
B.4.6	Petición de huevos nueva, disponible y nueva	64
B.4.7	Petición nueva reset	65
B.4.8	Creación de arrays	65
B.4.9	OK	69
B.4.10	Control Partes cintas	71
B.4.11	Inicializaciones cuando A1	73
B.4.12	OK Aguas Abajo	73
B.4.13	Check todo parado / todo en marcha	73
B.4.14	F5 continuar	74
B.4.15	Etapas actual	74
B.4.16	Estados (marcha/paro/error) HMI	75
B.5	Temporizadores [FC8]	79
B.6	Fallo_EDM [FC3]	80
B.7	Actualización de estados [FC5]	88
B.7.1	GRAFCET superior	88
B.7.2	F2 [FB5]	90
B.7.3	F1 [FB6]	92
B.7.4	F3 [FB7]	92
B.7.5	F4 [FB8]	97
B.8	Ecuaciones de salida [FC7]	98
B.9	Asignación de alarmas [FC2]	110
B.10	Configuración de dispositivos	112
B.11	Dispositivos y redes	112

B.12 Variables PLC	113
B.12.1 Tabla de variables estándar	113
B.12.2 DI	113
B.12.3 DQ	116
C Programación HMI	121
C.1 Administración de imágenes: Plantillas	121
C.2 Imagen raíz	123
C.3 E_ConfMarcha y E_Guardamotores	126
C.4 Vista simbólica	127
C.5 ALARMAS_Activas y ALARMAS_Historicas	128
C.6 Variables HMI	129
C.7 Lista de textos y gráficos	133
C.8 Avisos HMI	136
D Manual de uso	139
D.1 Inicio	140
D.2 Producción	141
D.3 Emergencia	145

Índice de figuras

1	Distribución de cintas en la planta	15
2	Sinóptico detallado funcionamiento mesa en conjunto con las otras cintas . .	16
3	Esquema eléctrico control de la potencia	16
4	Señal de entrada	18
5	Cintas transportadoras en diferentes industrias	18
6	Alternativas lógica	21
7	FPGA [12]	22
8	Alternativas lógica	22
9	Diferentes elementos de visualización	23
10	Diferentes software de simulación de fábricas con proyectos de cintas trans- portadoras	24
11	Vista general planta	27
12	Actuadores "repetidos"	28
13	Distribución partes	28
14	Distribución mesas	29
15	Fotocélula	29
16	Cuadro simulación	30
17	Selección estados, imagen sin editar [24]	31
18	GRAF CET superior	33
19	Macro etapa F2	35
20	Macro etapa F1	36
21	Macro etapa F3	37
22	Macro etapa F4	38
23	Barras comunes a todas las pantallas	44
24	Pantalla principal	44
25	Pantalla principal blanco	45
26	Pantalla principal gris	45
27	Pantalla vista simbólica	46
28	Pantalla fallo guardamotores	47
29	Pantalla fallo EDM	47
30	Pantalla alarmas activas	48
31	Pantalla alarmas históricas	48
32	Bloques de programa	55
33	Distribución tarjetas	112
34	Comunicación PROFINET	112
35	Plantilla_1	121
36	Características dinámicas FB_Left_Rectangular	122
37	Barras comunes a todas las pantallas	122

38	Imagen raíz con rectángulos superiores visibles	123
39	HMI con interruptores ON/OFF	124
40	Generación fallo	124
41	Pantalla principal gris	125
42	Animación apariencia	125
43	Animación apariencia mesas	125
44	Pantalla fallo guardamotors	126
45	Pantalla fallo EDM	126
46	Pantalla vista simbólica	127
47	Lista gráficos CN1	127
48	Configuración visor de avisos	128
49	Pantallas alarmas	128
50	Software listo	139
51	Petición de huevos lista para "Arrancar"	140
52	Preparación	141
53	Marcha	141
54	Ejemplo comprobación manual	142
55	Finalizando	143
56	Comprobación desorden	144
57	Generación de emergencia mediante fallo guardamotor	145
58	Visualización emergencia fallo guardamotor	146

Índice de tablas

1	Tabla variables Factory I/O	51
2	Tabla variables estándar	113
3	Tabla entradas digitales	113
4	Tabla de salidas digitales	116
5	Tabla variables HMI	129
6	Lista de textos HMI	133
7	Lista de textos HMI	133
8	Lista de Avisos HMI	136

Lista de código fuente

1	Ecuaciones de salida cinta general A parte 1	43
2	Main	56
3	Simulación	60
4	Asignación de entradas	62
5	Segmento 1: Error crítico	63
6	Segmento 2: Condiciones Iniciales	63
7	Segmento 3: FinCiclo reset	64
8	Segmento 4: Comprobación desorden reset	64
9	Segmento 5: Último huevo	64
10	Segmento 6: Petición de huevos nueva, disponible y nueva	65
11	Segmento 7: Petición nueva reset	65
12	Segmento 8: Creación arrays	68
13	Segmento 9: CG_OK, DM_OK, CM_OK, CN_OK	70
14	Segmento 10: Control Partes cintas	72
15	Segmento 11: Inicializaciones cuando A1	73
16	Segmento 12: OK Aguas Abajo	73
17	Segmento 13: Check todo parado / todo en marcha	74
18	Segmento 14: F5 continuar	74
19	Segmento 15: Etapa actual	74
20	Segmento 17: Estados (marcha/paro/error) HMI	78
21	Temporizadores	79
22	EDM parte 1	80
23	EDM parte 2	81
24	EDM parte 3	82
25	EDM parte 4	83
26	EDM parte 5	84
27	EDM parte 6	85
28	EDM parte 7	86
29	Segmento 6: Fallo EDM	87
30	Actualización de estados pt1	88
31	Actualización de estados pt2	89
32	Actualización de estados pt3	90
33	Actualización de estados F2 pt1	90
34	Actualización de estados F2 pt2	91
35	Actualización de estados F2 pt3	92
36	Actualización de estados F1	92
37	Actualización de estados F3 pt1	93
38	Actualización de estados F3 pt2	94
39	Actualización de estados F3 pt3	95
40	Actualización de estados F3 pt4	96

41	Actualización de estados F3 pt5	97
42	Actualización de estados F4	97
43	Segmento 2: Cinta A	98
44	Segmento 3: Cinta B	99
45	Segmento 5: Mesa 1	100
46	Segmento 6: Mesa 2	101
47	Segmento 7: Mesa 3	102
48	Segmento 8: Mesa 4	104
49	Segmento 9: Mesa 5	105
50	Segmento 10: Mesa 6	106
51	Segmento 12: Nave 1	107
52	Segmento 13: Nave 2	107
53	Segmento 14: Nave 3	107
54	Segmento 15: Nave 4	107
55	Segmento 16: Nave 5	108
56	Segmento 17: Nave 6	108
57	Segmento 18: Nave 7	108
58	Segmento 19: Nave 8	108
59	Segmento 20: Nave 9	109
60	Segmento 21: Nave 10	109
61	Segmento 22: Nave 11	109
62	Segmento 23: Nave 12	109
63	Asignación de alarmas	111

Capítulo 1

Objeto

Actualmente, el uso de cintas transportadoras está muy extendido en todos los sectores económicos. Podemos encontrarlas en las minas para transportar el material que extraen; en las fábricas, para mover el producto en el proceso de fabricación; e incluso en el supermercado, para acercar los artículos a la caja registradora.

Su automatización puede ser sencilla, como en el caso del supermercado, donde, normalmente, un sensor fotoeléctrico hace que la cinta se mueva hasta detectar un objeto. También puede llegar a ser muy compleja, como en el sector del automóvil, donde la coordinación de las diferentes cintas hace que cada componente esté en el lugar y momento que debe.

Un fallo puede provocar pérdidas económicas pequeñas si se retrasan los componentes o algo más importantes si causa la rotura de algún material de alto valor económico. Además, si el fallo se produce cerca de seres humanos, podría suponer un riesgo para sus vidas.

Otro sector en el que se utilizan muchas cintas es el de la producción avícola, sobre todo cuando esta actividad se realiza de forma intensiva. Este proyecto pretende diseñar la automatización de una serie de cintas transportadoras encargadas de la recogida y el transporte de huevos en una macro granja de gallinas. Se basa en un proyecto en el que participé mientras realizaba prácticas en empresa.

El objetivo principal es controlar los diferentes elementos de la planta para recoger el producto de las naves y transportarlo a la clasificadora. El segundo objetivo es supervisar la planta mediante una interfaz hombre-máquina. Al tratarse de un espacio muy amplio, esto es crucial para que un operario pueda conocer el estado del proceso sin tener que desplazarse por el recinto.

Este proyecto va a consistir en la simulación de la planta, pero consiguiendo la máxima similitud con el funcionamiento real. Para conseguir esto, el [Estudio de necesidades](#) es bastante concreto con las limitaciones existentes. Para poder realizar la puesta en marcha de la planta faltaría especificar el cableado eléctrico, así como el cumplimiento del REBT.

A lo largo de este grado, se han estudiado diferentes asignaturas que hacen posible el desarrollo de este proyecto, como son Automatización Industrial e Informática Industrial I. En estas asignaturas se han obtenido las competencias necesarias para la programación en lenguajes como GRAFCET y Ladder y para la realización de interfaces de usuario. Además, también se ha realizado un exhaustivo estudio sobre programación en TIA Portal y simulación de industrias.

Capítulo 2

Antecedentes

Este proyecto está basado en un trabajo que realicé mientras hacía prácticas de empresa en Automatiza S.L., que consistió en realizar el esquema eléctrico de la recogida de huevos en una macro granja de gallinas ubicada en Angola. He decidido basarme en este proyecto pero trasladar la ubicación a una población de Castilla-La Mancha llamada Campillo de Altobuey.

En esta macro granja lo que se pretende es automatizar la recogida de huevos. Esta macro granja está compuesta por 12 naves donde se alojan las gallinas, una cinta particular para cada nave (CN) que lleva los huevos a una mesa distribuidora (M); a esta mesa llegan huevos de dos naves distintas que se encuentran enfrentadas, y distribuye los huevos a la cinta general (CG) A o a la B, ambas están divididas en 4 partes.

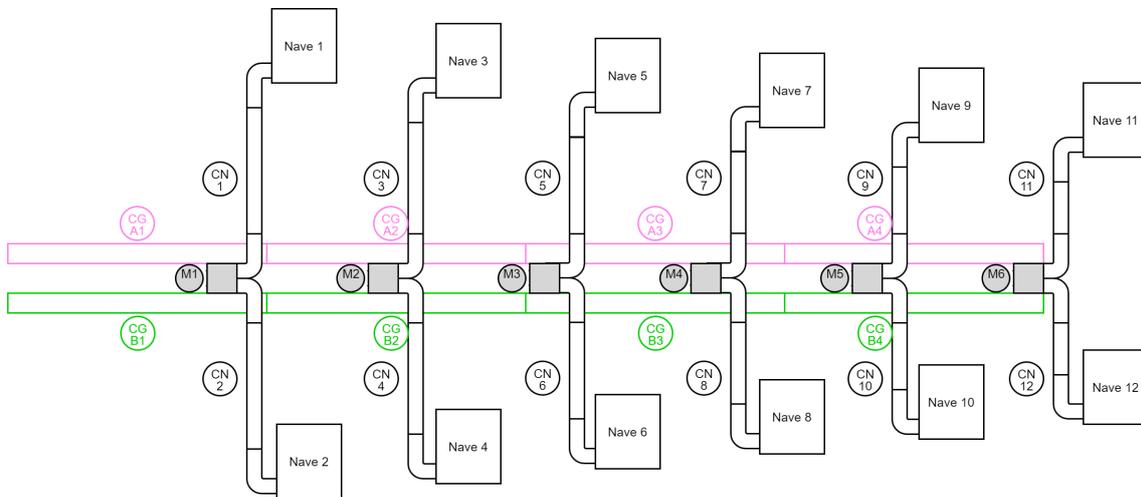


Figura 1: Distribución de cintas en la planta

Las cintas sólo funcionan en un sentido y se activan mediante un relé, las mesas, en cambio, constan de dos partes: la cinta, que mueve los huevos y el clasificador, que levanta los huevos y los mueve en un sentido u otro, hacia la cinta general A o a la B.

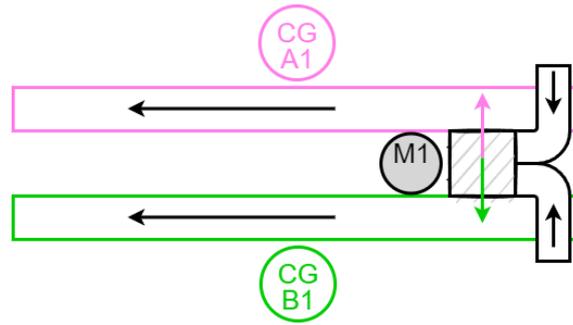


Figura 2: Sinóptico detallado funcionamiento mesa en conjunto con las otras cintas

Todas las salidas activan relés que a su vez activan contactores (si se cumplen las condiciones de seguridad) que son los que dan potencia a la maquinaria, además se utiliza un contacto normalmente abierto (n.a.) para confirmar el buen funcionamiento del contactor y tener un sistema más seguro. En el caso de los motores de las cintas, los guardamotores tienen un contacto auxiliar que se utiliza para saber si se ha disparado o no.

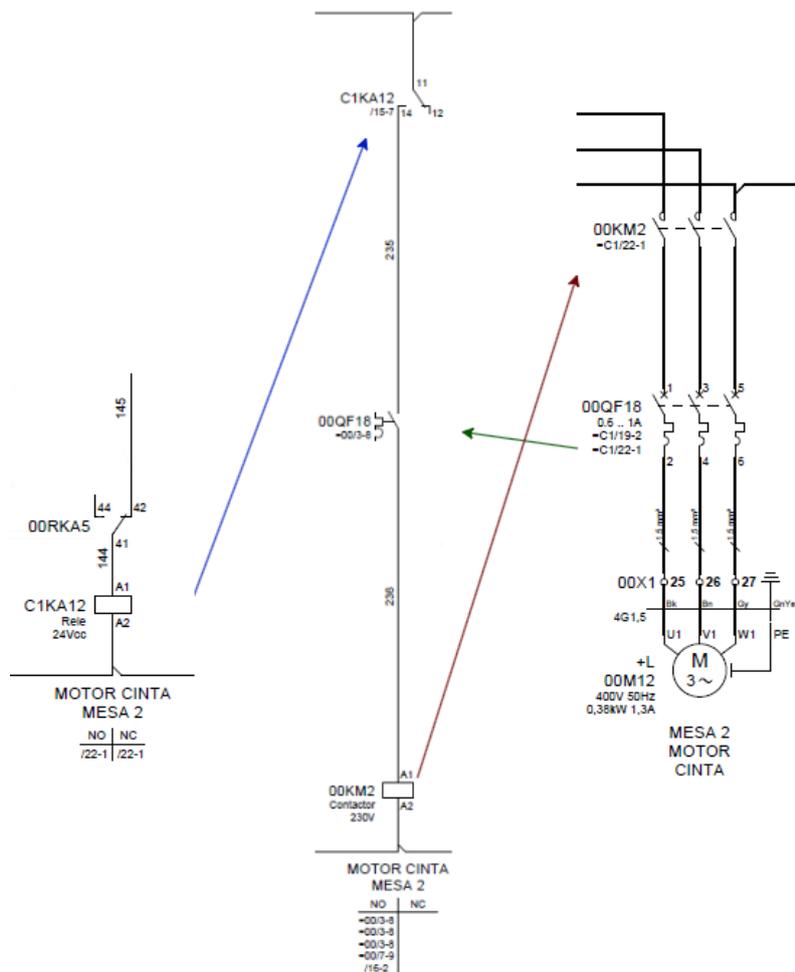


Figura 3: Esquema eléctrico control de la potencia

Respecto a los sensores, hay una fotocélula al final de cada cinta, su función es detectar el paso de los huevos, para así poder conocer cuánto tiempo ha pasado desde el último. Las mesas, además de contar con el guardamotor, también tienen un sensor que indica si la mesa está haciendo que el objeto avance (hacia la cinta general B), retroceda (hacia la cinta general A), o esté quieto.

También hay diversas setas de emergencia enseriadas a lo largo de la planta, que activarán el protocolo de seguridad pertinente.

Con todo esto, se establecen las diferentes entradas y salidas del sistema a controlar.

2.1. Legislación

La normativa que seguir para la programación de autómatas es la siguiente:

- IEC 61131-3:2013 Autómatas programables. Parte 3: Lenguajes de programación, donde se definen la sintaxis y la semántica de los lenguajes de programación para controladores programables.
- ISO 13850:2015 Seguridad de las máquinas. Función de parada de emergencia. Principios para el diseño., donde establece los requisitos para el correcto funcionamiento de la función de parada de emergencia de una máquina.
- IEC 60848:2013 Lenguaje de especificación GRAFCET para diagramas funcionales en secuencia, donde se especifican los símbolos y reglas para la representación del lenguaje.
- UNE-EN 61508 Seguridad funcional de los sistemas eléctricos / electrónicos / electrónicos programables relacionados con la seguridad.
- UNE-EN 62439 Redes de comunicación industrial. Redes de automatización de alta disponibilidad.

2.2. Información técnica

A falta del sistema de automatización, el resto de los elementos se encuentran cableados, por lo que se tiene la siguiente información:

- Hay posibilidad de alimentar el sistema con red trifásica de 400 V, monofásica de 230 V, ambas de corriente alterna, o monofásica de 24 V, con un consumo máximo de 120W.
- Todas las señales de entrada son contactos normalmente abiertos o cerrados alimentados a 24 V.

- Las señales de salida tienen que ser capaces de activar los relés correspondientes, en este caso se trata del relé 46.52.9.024.0020 de la marca FINDER [1], que trabajan con una tensión de entrada de 24V.

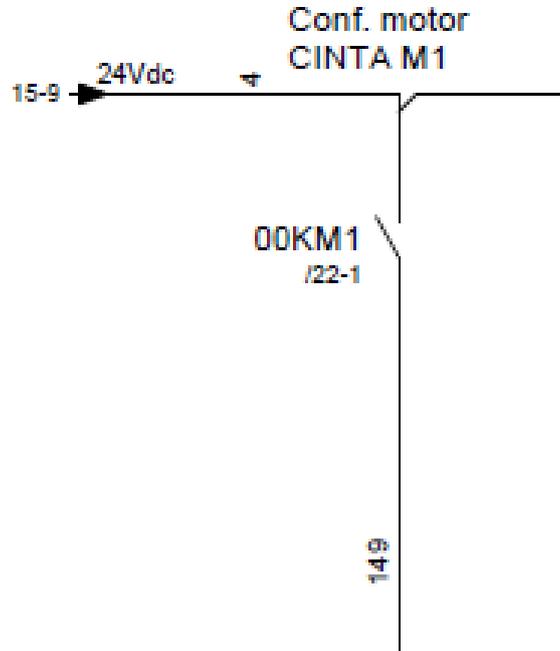


Figura 4: Señal de entrada

2.3. Proyectos similares

Como se ha explicado en el Capítulo 1 Objeto, el uso de cintas transportadoras está muy extendido en muchos procesos de fabricación o producción, se pueden encontrar ejemplos en sectores como el agrario o minero.



(a) Sector agrario [2]



(b) Industria cementera [3]

Figura 5: Cintas transportadoras en diferentes industrias

Capítulo 3

Estudio de necesidades

3.1. Lógica

Las entradas y salidas del controlador deben tener las siguientes características, previamente especificadas en la [Sección 2.2 Información técnica](#):

- Posibilidad de alimentación trifásica de 400 V, monofásica de 230 V, ambas de corriente alterna o 24 V de continua, consumo máximo de 120 W.
- 40 salidas digitales: 16 cintas de recolección, 8 de transporte y 6 de distribución. Voltaje de control de todas las salidas digitales 24 V.
- 89 entradas digitales: una seta de emergencia, un botón de arranque y otro de rearme; una fotocélula por cinta de transporte; una entrada de guardamotor y de confirmación de marcha por cinta; además cada mesa cuenta con 3 señales cada una que indican la dirección. Voltaje de 24 V.

Todos los elementos deberán seguir una activación secuencial, evitando que huevos que se encuentren en cintas en movimiento choquen con otros en cintas en marcha. La desactivación se llevará a cabo siguiendo también una secuencia, aunque no tiene por qué ser la inversa de la primera.

La desactivación de la señal de las setas, así como la activación de algún guardamotor supondrá el paro inmediato de todos los elementos. Si hay un fallo en la señal de confirmación de algún elemento que se prolongue durante un tiempo superior a treinta segundos, también generará el mismo protocolo de emergencia.

Tras una parada de emergencia, las cintas se deben vaciar, para poder tener un control sobre esto en la programación se comprobarán las fotocélulas. Si estas detectan algún obstáculo, significará que aún hay huevos en las cintas y que la planta no puede volver a su estado inicial.

3.2. Visualización

Los operarios tendrán a su disposición una interfaz en la que poder observar el estado de la planta. Además, en ella deberán poder configurar las cintas de recolección.

Debido a que este proyecto no abarca la puesta en marcha, en la visualización se deberán de poder generar los mismos fallos o errores que tendría una planta real.

3.3. Simulación

Uno de los requerimientos del cliente es poder ver, mediante una simulación 3D el funcionamiento de la planta. De esta forma, valorará si proceder o no a la puesta en marcha real.

3.4. Escalabilidad

El control tiene que ser fácilmente expansible a más entradas o salidas o un cambio por completo de la lógica de forma fácil y sin un incremento significativo del coste.

Capítulo 4

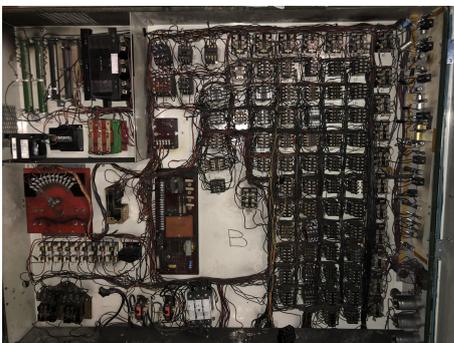
Estudio de alternativas

Este capítulo se va a centrar en la exposición y elección de diferentes componentes que conforman el proyecto, teniendo en cuenta las necesidades del cliente.

4.1. Breve descripción de alternativas

4.1.1. Automatización

- **Lógica cableada:** este método ofrece un bajo coste inicial en lo relacionado al material ya que se utilizan relés, como por ejemplo el 46.52.9.024.0020 de FINDER con un coste de 7,64 €[4], aunque es imposible saber la cantidad total de relés de antemano sin realizar la parte de control del diseño eléctrico. Por otro lado, en lo que refiere al diseño eléctrico puede ser tan complejo como se desee que sea la lógica, pero ante cualquier cambio en la instalación o en la lógica, supone un cambio significativo en el cableado y por tanto el precio y tiempo invertido aumentan significativamente.
- **Controlador lógico programable (PLC):** esta solución es ampliamente utilizada por la industria por diversos motivos: lenguajes de programación y protocolos de comunicación altamente estandarizados, relativo bajo coste (CPU de Siemens con un módulo para 32 entradas y otro módulo para 32 salidas tiene un coste total de 1467 €[5][6][7]) y su manejo de las entradas y salidas es eficiente [8]. Por otro lado, las entradas digitales pueden ser de 24 V o 230 V.



(a) Armario con lógica cableada [9]



(b) PLC en un cuadro eléctrico [10]

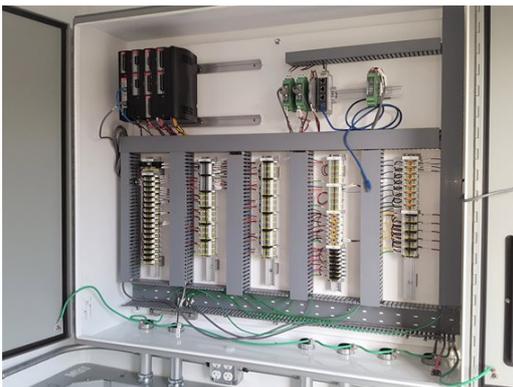
Figura 6: Alternativas lógica

- **Matriz de puertas lógicas programable en campo (FPGA):** estos dispositivos utilizan hardware en vez de software para el control de la lógica y la activación de las salidas. De esta forma no se utiliza capacidad de procesamiento para estas tareas simples, pero que pueden ocupar mucha capacidad, dejando así espacio para cálculos más complejos como es el de trayectorias. Su precio suele ser de unos 1574 €[11]. Las entradas necesitan un voltaje mínimo de 3.2 V y máximo de 3.465 V [12].



Figura 7: FPGA [12]

- **Unidad Terminal Remota (RTU):** en esta sección encontramos un aparato que dispone de muchas características integradas y que sólo hay que configurar. Las entradas y salidas de éste funcionan a 24 V [13]. El precio inicial ronda los 5000 €[14].
- **Controlador de automatización programable (PAC):** estos controladores tienen una gran capacidad de procesamiento, tienen integradas pantallas que se pueden utilizar como HMI y tanto las entradas como las salidas pueden operar a un voltaje de 24 V. El precio del módulo de expansión es de 1160 €[15], no se ha podido encontrar información sobre el precio del módulo principal.



(a) RTU en un armario eléctrico [16]



(b) PAC [15]

Figura 8: Alternativas lógica

Como se puede observar, existe una gran cantidad de opciones a elegir, con una variedad bastante amplia en cuanto a precio y características. Su utilidad o no depende de la aplicación donde se vaya a implementar.

4.1.2. Visualización

Respecto la visualización hay dos grandes alternativas: mediante señalización en el cuadro eléctrico, como luces o zumbadores; o pantalla HMI. En realidad, estos elementos se pueden utilizar conjuntamente para realizar un control y visualización completa, pero hay que elegir el sistema más adecuado para esta aplicación.

- **Señalización en el cuadro:** con elementos como sinópticos y bombillas o LEDs es posible transmitir de forma clara el estado de los diferentes componentes de la planta. Mediante selectores físicos es posible configurar las cintas de recolección. Esta opción es de bajo coste, pero al igual que la lógica cableada, es complicado de actualizar en caso de que la planta se expanda.
- **Pantalla HMI:** este elemento generalmente se monta sobre el propio armario eléctrico, mediante algún protocolo de comunicación recibe datos de un PLC y muestra en la pantalla tanto elementos estáticos como dinámicos. Su programación se puede modificar en cualquier momento tras la puesta en marcha y puede ser tan intuitiva como así lo desee el equipo de programación. Respecto al precio, es de 649 €[17].



(a) Sinóptico y señalización [18]



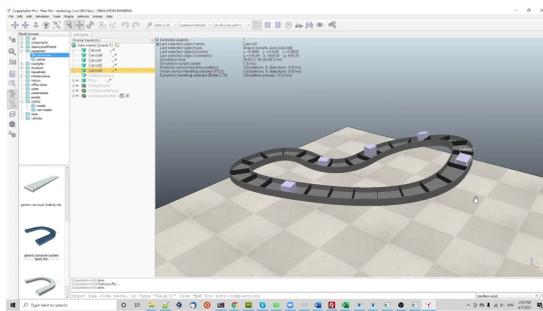
(b) HMI en un armario eléctrico [19]

Figura 9: Diferentes elementos de visualización

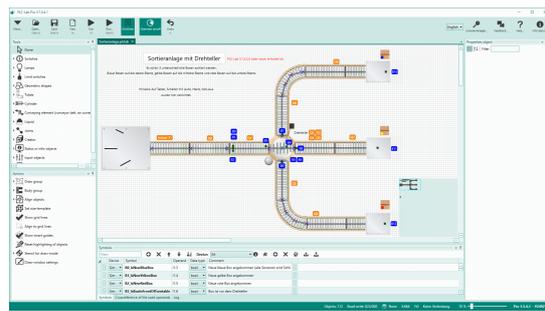
En este caso, no existe tanta variedad de opciones. Ambas están ampliamente integradas en diferentes industrias, y el uso de una alternativa u otra suele depender principalmente de la existencia o no de un PLC y del presupuesto.

4.1.3. Simulación

- **CoppeliaSim:** es una plataforma de simulación de robots. Está basado en una arquitectura de control distribuida, pudiendo controlar cada elemento de forma individual utilizando scripts en diferentes lenguajes. Su simulación es 3D y ofrece diferentes librerías con elementos prediseñados. La comunicación con aplicaciones externas se puede realizar mediante ZeroMQ, WebSocket, puerto serie o mediante plugins.
- **PLCLab:** es un software de simulación 2D, con una serie de librerías con objetos típicamente utilizados en la industria. Ofrece la posibilidad de crear nuevos elementos y ofrecerles características físicas, como masa o magnetismo de forma rápida y fácil. Puede interactuar de forma directa con diferentes PLCs y softwares de programación de estos.
- **Factory I/O:** es un programa de simulación 3D para aprender sobre automatización. Su interfaz es de fácil uso, y tiene una librería lo suficientemente extensa como para poder realizar una simulación de este proyecto. Ofrece la posibilidad de comunicarse con PLCs o con sus plataformas de programación.



(a) CoppeliaSim [20]



(b) PLCLab [21]



(c) Factory I/O [22]

Figura 10: Diferentes software de simulación de fábricas con proyectos de cintas transportadoras

En este aspecto las opciones presentadas también son escasas, pero en este caso la simulación de la planta no está integrada en la mayoría de las industrias, lo que explica la escasez de alternativas.

4.2. Criterio de selección

El principal requisito es que cumpla con los requerimientos del cliente, en caso de que esto se cumpla, su integración tiene que ser asequible, poniendo un presupuesto máximo en el apartado de la lógica de 5000 €. Finalmente, todos los elementos elegidos tienen que ser capaces que integrarse entre ellos de una forma rápida.

4.3. Justificación de la solución adoptada

4.3.1. Lógica

Se ha descartado la lógica cableada, por su difícil escalabilidad, ya que un pequeño cambio en la lógica puede suponer mucho tiempo en cambiar el cableado. Por no cumplir los requisitos técnicos se ha eliminado la alternativa de la FPGA, ya que el voltaje requerido es mucho mayor que el ofrecido por la FPGA.

Teniendo en cuenta los requisitos del cliente, quedan tres opciones en la lógica: el PLC, la RTU, y el PAC. Teniendo en cuenta el precio máximo marcado para materiales, la RTU queda descartada y ante la imposibilidad de poder acceder a la información sobre el precio del PAC, también queda descartado.

Por lo que finalmente, se ha optado por un PLC. En concreto se ha elegido el modelo SIMATIC S7-1511-1 PN de la marca Siemens [5].

4.3.2. Visualización

Aunque en relación tiempo/precio hubiera sido más asequible optar por la señalización mediante LEDs, debido a que toda la instalación se encuentra ya cableada a excepción de la lógica, se ha elegido utilizar una pantalla HMI. Esto se debe a que teniendo en cuenta la posible escalabilidad de la planta es mucho más rápido reprogramar la pantalla que volver a cablear los LEDs.

4.3.3. Simulación

En lo referente a la simulación se ha optado por el software Factory I/O, ya que es el único con comunicación directa con PLCs y ofrece simulación 3D.

Capítulo 5

Descripción detallada de la solución adoptada

5.1. Simulación

Como se ha expuesto en la [Subsección 4.3.3](#), se ha empleado el software de simulación Factory I/O, se ha recreado, en la medida de lo posible la disposición de las cintas y las mesas como estarían dispuestas en la vida real. Aunque el resultado es bastante aceptable, no ha sido posible incluir huevos como los elementos a transportar, en cambio los "Emitters" emitirán unos objetos ligeros de color azul. Como no ha sido posible recrear los guardamotores ni los contactores, no va a haber elemento equivalente en la simulación. Más adelante, en la [Sección 5.2](#) se explica cómo se ha atajado este problema.



Figura 11: Vista general planta

A continuación, se va a exponer como está dividida la planta, explicando los componentes de cada sección y su función. Cabe tener en cuenta que componentes como los alineadores son pasivos y no se van a explicar en esta sección.

Como referencia para la distribución se va a tomar la [Figura 1](#). Debido a la falta de

personalización del software de simulación, se han tenido que añadir más cintas de las previamente pensadas. El nombre de los elementos extra termina con un ".1", ".2", ".3" o "PT2" para poder diferenciarlos más fácilmente.



Figura 12: Actuadores "repetidos"

La planta está dividida en 4 partes, según las cintas generales (CG). La cinta general A (CGA) se encuentra más cercana a las naves con número impar, mientras que la cinta general B (CGB) se encuentra más cercana a las pares.

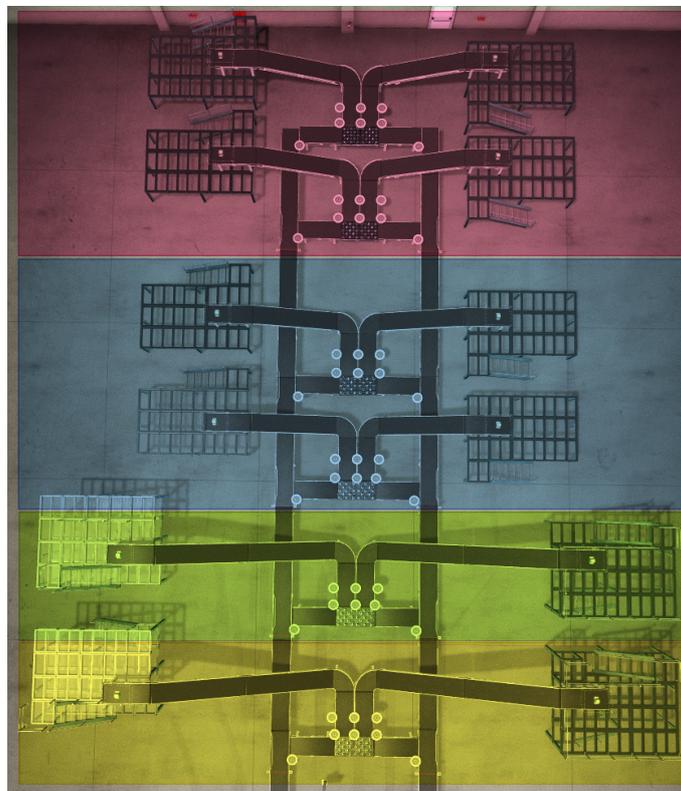


Figura 13: Distribución partes

La primera parte, señalada en la **Figura 13** de color amarillo, comprende a las naves 1 y 2 y a la mesa 1. En el color verde están las naves 3 y 4 y la mesa 2, que conforman la segunda parte. La tercera, de color azul, incluye las naves 5, 6, 7 y 8 y a las mesas 3 y 4. Por último de color rosa, se encuentran las naves 9, 10, 11 y 12 que juntamente con las mesas 5 y 6 forman la cuarta parte.

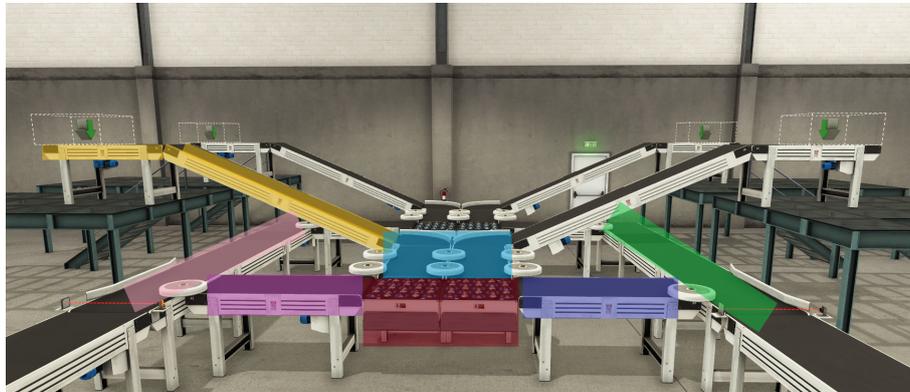


Figura 14: Distribución mesas

En la **Figura 14** se encuentran coloreados los diferentes elementos que conforman el funcionamiento en conjunto de las naves con las mesas y estas con las cintas generales. Como se puede observar, por las cintas de las naves (amarillo), bajará el producto; una vez lleguen a las cintas de las mesas (azul claro), son desplazados a las mesas (rojo), y dependiendo de la orden de a qué cinta tienen que ir los huevos de esas mesas, serán desplazados a "Retroseso" (violeta) y por tanto a la CGA (rosa) o a "Avance" (azul) y por tanto a la CGB (verde). Por lo tanto, las mesas están compuestas por dirección mesa (DM) y cinta mesa (CM).

Al final de cada parte se encuentra una fotocélula, que se utiliza para saber si hay o no huevos en la cinta, y en el caso del sensor de la primera parte, también se utiliza para calcular si ha pasado o no el último huevo de la tanda. Para su utilización es necesario el sensor y un reflector, como se muestra en la imagen.

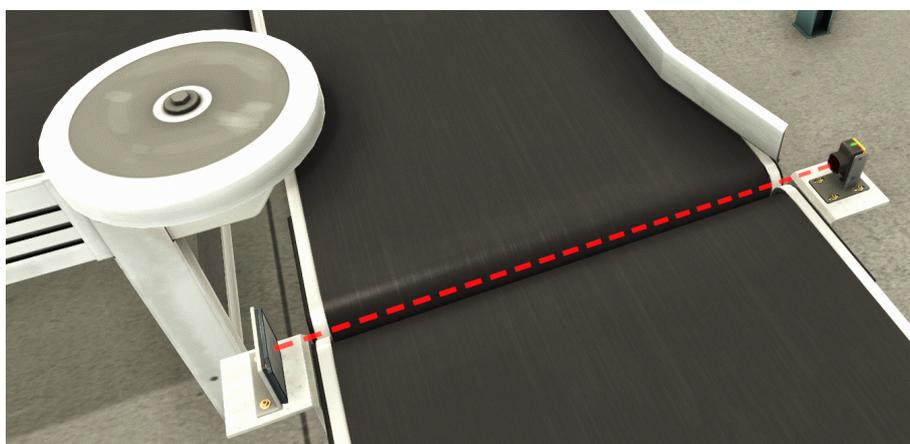


Figura 15: Fotocélula

Además, en la simulación se ha incorporado una columna con un cuadro eléctrico, donde se encuentran el botón de parada de emergencia, arranque y de rearme. Estos elementos también se encontrarían en el cuadro eléctrico real, además de elementos de seguridad como guardamotores, etc.



Figura 16: Cuadro simulación

En [Apéndice A](#) se encuentra una tabla con las entradas y salidas de elementos en el software al PLC. En el [Parte III](#) se explica detalladamente cómo conectar el software de simulación de la planta con la simulación del PLC y la HMI.

5.2. Programación

La programación se ha realizado en el software TIA PORTAL, de Siemens, en concreto en la versión V16.

El programa del autómatas se ha creado en el lenguaje GRAFCET, según la norma **IEC 60848**, siguiendo la guía GEMMA. Aunque su implementación final se ha llevado a cabo combinando Ladder (también llamado KOP, del alemán *Kontaktplan*) y SCL, que corresponde al lenguaje de programación "ST" Structured Text [23], estando ambos lenguajes incluidos en la norma **IEC 61131-3**. En este apartado sólo se van a exponer las ecuaciones de activación, tanto de los estados como de las salidas, su implementación final se puede encontrar en el [Apéndice B](#).

5.2.1. GRAFCET

Siguiendo los pasos que marca la guía, primero hay que elegir qué rectángulos-estado necesitamos, y por tanto cuáles necesitamos quitar.

Esta planta necesita maniobras de preparación y cierre, ya que es necesario ir encendiendo/apagando las cintas siguiendo una secuencia, para evitar que huevos que se encuentren en una cinta en movimiento colisionen contra otros en una cinta en marcha. Además, un procedimiento ante paradas de emergencia con su posterior preparación para la marcha es necesario, ya que, por una parte, ante una emergencia hay que parar toda la maquinaria, para evitar cualquier daño de la maquinaria o humano; y, por otra parte, si ha habido alguna parada de emergencia será imprescindible comprobar que no quedan huevos en las cintas de cuando ocurrió la parada.

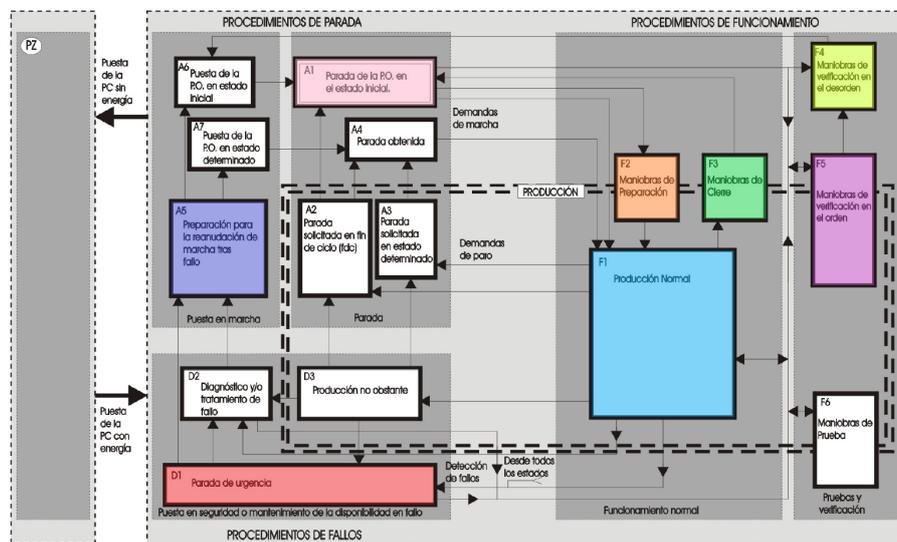


Figura 17: Selección estados, imagen sin editar [24]

Tras esto, se definen los eventos de transición entre los rectángulos que quedan. Y por último se diseñan los modelos de cada estado.

- Cualquier error crítico, que definiremos más adelante, llevará al estado de Parada de emergencia D1.
- Para poder volver al estado inicial tras una parada de emergencia, es necesario rearmar y que las cintas estén en las condiciones iniciales. Para esto se utiliza un estado adicional, A6 Puesta de la P.O. en estado inicial.
- Respecto al funcionamiento normal, como se ha comentado anteriormente, es necesario realizar un procedimiento de preparación. En este, las cintas generales, las mesas y las cintas de las mesas se pondrán en marcha por orden. Para poder pasar a la producción normal, será necesario que todo esté correctamente aguas abajo de las cintas de las naves.

- Una vez está en marcha la producción normal, las cintas de las naves se activan. Este estado se mantiene activo mientras no haya pasado el último huevo, mientras el operario no pulse el botón de Fin de Ciclo, o se cambie la petición de huevos. Si la petición de huevos es válida (se pide de una nave como mínimo y de una cinta general como mínimo), se vuelve a las maniobras de preparación, en caso de que la petición sea nula, se vuelve al estado inicial.
- Si mientras la planta se encuentra en el estado de producción normal el operario pulsa el botón de ciclo o ha pasado el último huevo, se realizan las maniobras de cierre. Estas consisten en parar todas las cintas de las naves, y tras esto parar el resto de las cintas progresivamente. Una vez está todo parado, se vuelve al estado inicial.
- Además, también se han añadido maniobras de verificación, tanto en orden como en desorden. La primera se puede activar en cualquiera de los estados de funcionamiento normal, F1, F2 o F3; y con ayuda de un botón para continuar se seguirá la secuencia de activación/desactivación del GRAFCET inferior hasta llegar a A1. En cambio, para la maniobra de verificación en desorden, sólo se puede activar estando en A1 y vuelve a este mismo estado inicial una vez ha activado y desactivado todas las cintas (sólo se activará la dirección de las mesas en sentido avanzar).

Con esta información es posible realizar el dibujo del GRAFCET superior.

5.2.1.1. GRAFCET superior

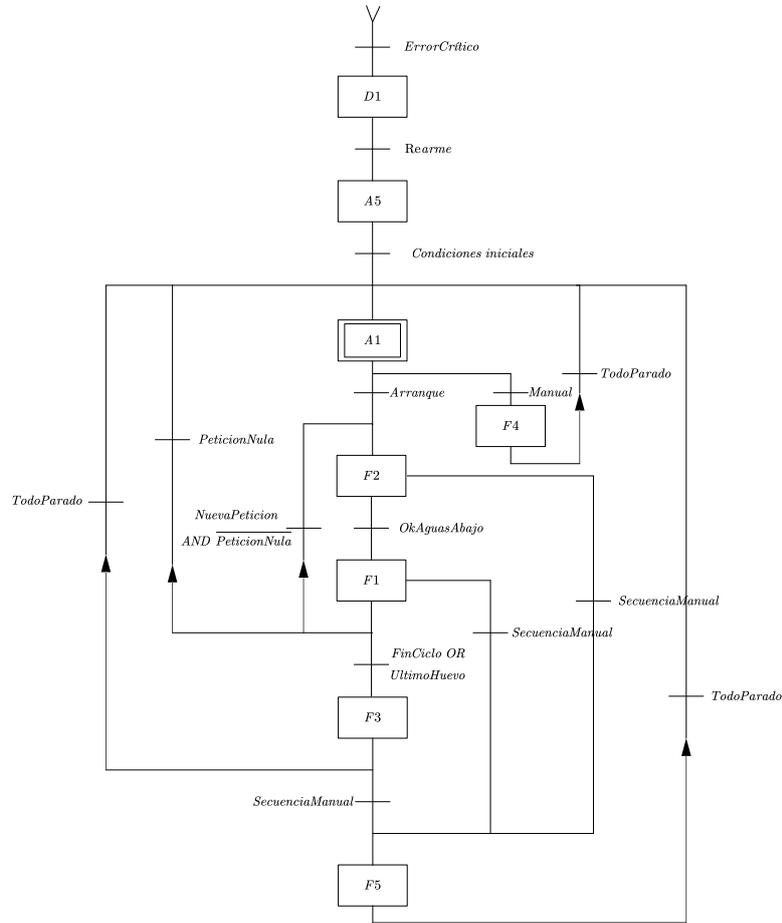


Figura 18: GRAFCET superior

La programación referente a la activación y desactivación de etapas se ha realizado mediante LADDER, tal y como se explicó en clase de Automatización Industrial, primero, definiendo la fórmula de cada etapa:

$$E_n = E_{n-1} \cdot Tr_n + E_n \cdot \overline{E_{n+1}} \quad (5.1)$$

Aplicando esta fórmula a el GRAFCET superior obtenemos las siguientes ecuaciones:

$$D1 = ErrorCrítico + D1 \cdot \overline{A6}$$

$$A6 = D1 \cdot Rearme + A6 \cdot \overline{A1}$$

$$A1 = FirstScan + F1 \cdot PeticiónNula + F4 \cdot TodoParado + F3 \cdot TodoParado + F5 \cdot TodoParado + A6 \cdot CondicionesIniciales + A1 \cdot \overline{F2} \cdot \overline{F4} \cdot \overline{F5} \cdot \overline{D1} \cdot \overline{A5}$$

$$F4 = A1 \cdot Manual + F4 \cdot \overline{A1} \cdot \overline{D1}$$

$$F2 = A1 \cdot Arranque + F1 \cdot PeticiónNueva \cdot \overline{PeticiónNula} + F2 \cdot \overline{F1} \cdot \overline{F5} \cdot \overline{D1}$$

$$F1 = F2 \cdot OKAguasAbajo + F1 \cdot \overline{A1} \cdot \overline{F3} \cdot \overline{F2} \cdot \overline{F5} \cdot \overline{D1}$$

$$F3 = F1 \cdot (FinCiclo + UltimoHuevo) + F3 \cdot \overline{A1} \cdot \overline{F5} \cdot \overline{D1}$$

$$F5 = SecuenciaManual \cdot (F2 + F1 + F3) + F5 \cdot \overline{A1} \cdot \overline{F2} \cdot \overline{F1} \cdot \overline{F3} \cdot \overline{D1}$$

Los únicos estados que cuentan con un GRAFCET inferior son F2, F1, F3 y F4.

5.2.1.2. F2

En la primera etapa de F2, F20 se desactivan todas las cintas generales, y siempre se pasa a la siguiente, F21. En esta se activan las cintas generales de la parte 1, siempre y cuando tenga que estar activa esa parte.

A partir de esta etapa y hasta la F25, todas tendrán una rama que, si se ha confirmado la activación de las cintas correspondientes y hay más partes que se tengan que activar, se pasará a la siguiente etapa; en caso contrario, es decir, si no hay más partes que activar, se pasa directamente a F25, siempre y cuando se confirme la activación de las cintas correspondientes.

En F25 se activa la dirección de la mesa que toque y se desactiva la contraria. Para pasar a la siguiente etapa, la dirección de las mesas tiene que ser la correcta; en esta, F26, se activan las cintas de las mesas que se tienen que activar. Por último, una vez se confirma que las cintas de las mesas se han activado correctamente, se pasa a la etapa F27.

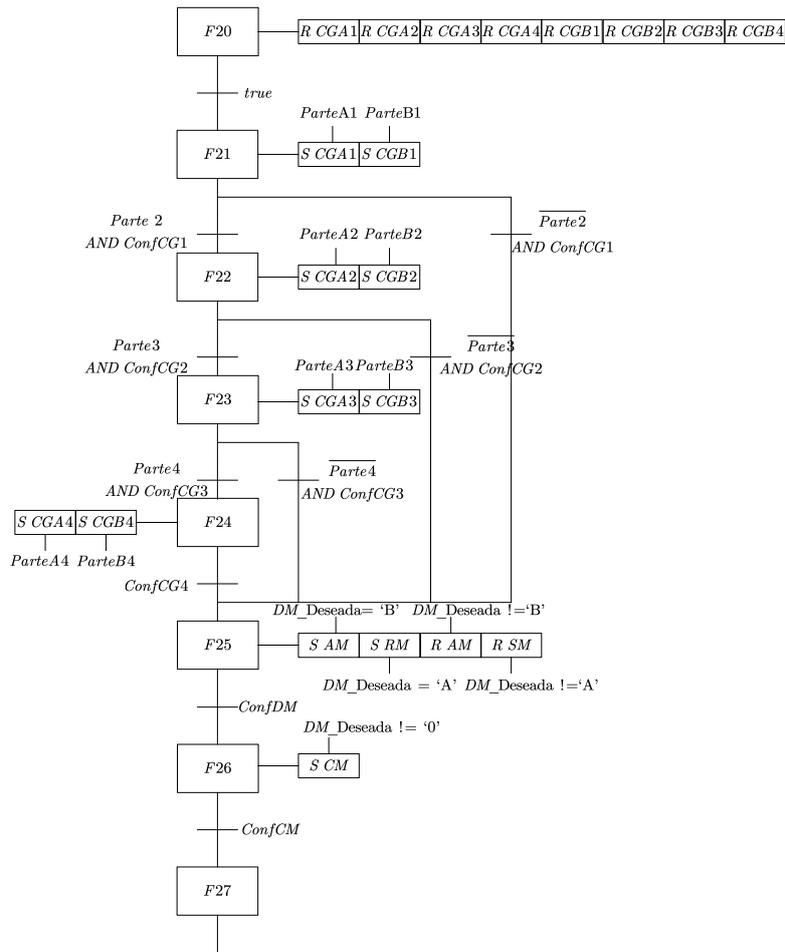


Figura 19: Macro etapa F2

Las ecuaciones de activación de cada estado quedarían así:

$$\begin{aligned}
 F20 &= A1 \cdot \text{Arranque} + F1 \cdot \text{PeticiónNueva} \cdot \overline{\text{PeticiónNula}} + F20 \cdot \overline{F21} \cdot \overline{D1} \\
 F21 &= F20 \cdot \text{true} + F21 \cdot \overline{F22} \cdot \overline{D1} \\
 F22 &= F21 \cdot (\text{Parte2} \cdot \text{ConfCG1}) + F22 \cdot \overline{F23} \cdot \overline{D1} \\
 F23 &= F22 \cdot (\text{Parte3} \cdot \text{ConfCG2}) + F23 \cdot \overline{F24} \cdot \overline{D1} \\
 F24 &= F23 \cdot (\text{Parte4} \cdot \text{ConfCG3}) + F24 \cdot \overline{F25} \cdot \overline{D1} \\
 F25 &= F24 \cdot \text{ConfCG4} + F23 \cdot (\overline{\text{Parte4}} \cdot \text{ConfCG3}) + F22 \cdot (\overline{\text{Parte3}} \cdot \text{ConfCG2}) \\
 &\quad + F21 \cdot (\overline{\text{Parte2}} \cdot \text{ConfCG1}) + F25 \cdot \overline{F26} \cdot \overline{D1} \\
 F26 &= F25 \cdot \text{ConfDM} + F26 \cdot \overline{F27} \cdot \overline{D1} \\
 F27 &= F26 \cdot \text{ConfCM} + F27 \cdot \overline{F1_F10} \cdot \overline{D1}
 \end{aligned}$$

5.2.1.3. F1

Respecto a la etapa F1, Producción normal, es bastante simple. La primera etapa se activa cuando la última de etapa de F2 está activa y todo está bien aguas abajo, y se pasa a la segunda, F11 cuando se ha confirmado que todas las naves que tienen que estar activas lo están.

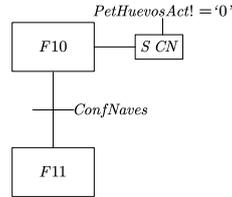


Figura 20: Macro etapa F1

Las ecuaciones de activación de cada estado quedarían así:

$$F10 = F2 \cdot OKAguasAbajo + F10 \cdot \overline{F11} \cdot \overline{D1}$$

$$F11 = F10 \cdot ConfNaves + F11 \cdot \overline{F3} \cdot \overline{F30} \cdot \overline{D1}$$

5.2.1.4. F3

La macro etapa de desactivación, F3, es más compleja ya que se quiere ir comprobando etapa a etapa la desactivación de las salidas, para poder encontrar rápidamente el problema ante un fallo de programación. El orden que se ha seguido para la desactivación es empezando por la parte 4, por las cintas de las naves 11 y 12, seguido de la cinta y dirección de la mesa 6, y continua así, comprobando la desactivación hasta llegar a la etapa F52. Cabe comentar que cuando se añade una "M" al principio, por ejemplo, cuando se indica $\overline{MCN12}$, se refiere a la entrada de revisión marcha de la cinta de la nave 12, y así con el resto de los textos que la incluyen.

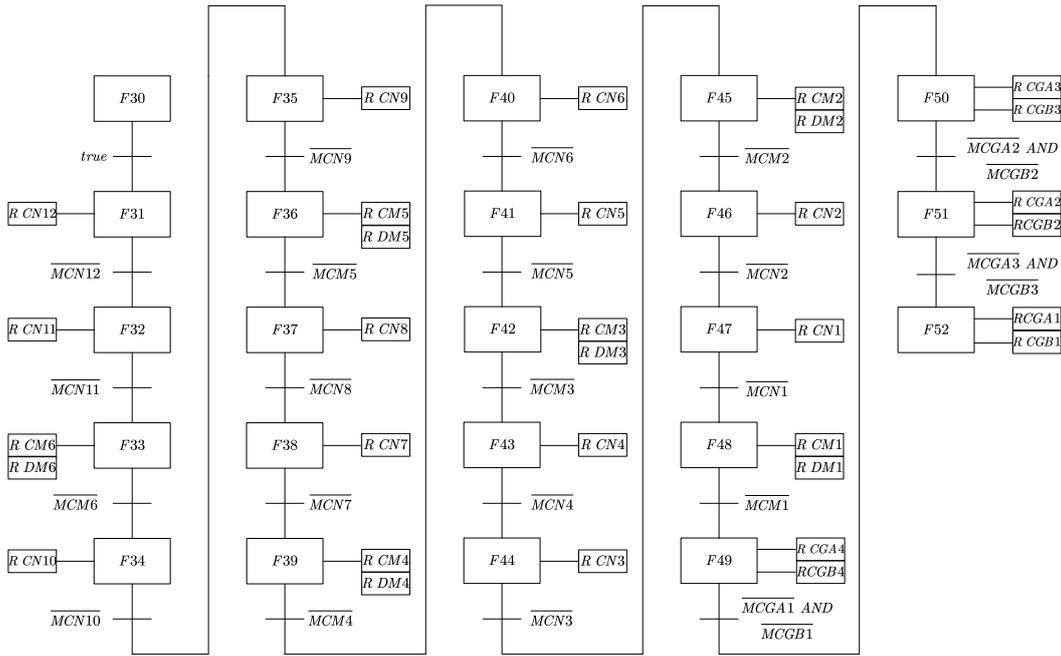


Figura 21: Macro etapa F3

Las ecuaciones de activación de cada estado quedarían así:

$$F30 = F1 \cdot (FinCiclo + UltimoHuevo) + F30 \cdot \overline{F31} \cdot \overline{D1}$$

$$F31 = F30 \cdot true + F31 \cdot \overline{F32} \cdot \overline{D1}$$

$$F32 = F31 \cdot \overline{MCN12} + F32 \cdot \overline{F33} \cdot \overline{D1}$$

$$F33 = F32 \cdot \overline{MCN11} + F33 \cdot \overline{F34} \cdot \overline{D1}$$

$$F34 = F33 \cdot \overline{MCM6} + F34 \cdot \overline{F35} \cdot \overline{D1}$$

$$F35 = F34 \cdot \overline{MCN10} + F35 \cdot \overline{F36} \cdot \overline{D1}$$

$$F36 = F35 \cdot \overline{MCN9} + F36 \cdot \overline{F37} \cdot \overline{D1}$$

$$F37 = F36 \cdot \overline{MCM5} + F37 \cdot \overline{F37} \cdot \overline{D1}$$

$$F38 = F37 \cdot \overline{MCN8} + F38 \cdot \overline{F33} \cdot \overline{D1}$$

$$F39 = F37 \cdot \overline{MCN7} + F39 \cdot \overline{F40} \cdot \overline{D1}$$

$$F40 = F39 \cdot \overline{MCM4} + F40 \cdot \overline{F41} \cdot \overline{D1}$$

$$F41 = F40 \cdot \overline{MCN6} + F41 \cdot \overline{F42} \cdot \overline{D1}$$

$$F42 = F41 \cdot \overline{MCN5} + F42 \cdot \overline{F43} \cdot \overline{D1}$$

$$F43 = F42 \cdot \overline{MCM3} + F43 \cdot \overline{F44} \cdot \overline{D1}$$

$$F44 = F43 \cdot \overline{MCN4} + F44 \cdot \overline{F45} \cdot \overline{D1}$$

$$F45 = F44 \cdot \overline{MCN3} + F45 \cdot \overline{F46} \cdot \overline{D1}$$

$$F46 = F45 \cdot \overline{MCM2} + F46 \cdot \overline{F47} \cdot \overline{D1}$$

$$F47 = F46 \cdot \overline{MCN2} + F47 \cdot \overline{F48} \cdot \overline{D1}$$

$$\begin{aligned}
 F48 &= F47 \cdot \overline{MCN1} + F48 \cdot \overline{F49} \cdot \overline{D1} \\
 F49 &= F48 \cdot \overline{MCM1} + F49 \cdot \overline{F50} \cdot \overline{D1} \\
 F50 &= F49 \cdot \overline{CGA4} \cdot \overline{CGB4} + F50 \cdot \overline{F51} \cdot \overline{D1} \\
 F51 &= F50 \cdot \overline{CGA3} \cdot \overline{CGB3} + F51 \cdot \overline{F52} \cdot \overline{D1} \\
 F52 &= F51 \cdot \overline{CGA2} \cdot \overline{CGB2} + F52 \cdot \overline{A1} \cdot \overline{D1}
 \end{aligned}$$

5.2.1.5. F4

Por último, la macro etapa F4, tiene dos etapas, F40, que activa todas las cintas y la dirección de "Avanzar", y F41, que una vez ha pasado un minuto desde que se activó F40 y se ha detectado que todas las salidas que corresponde están en marcha, se activa y desactiva todas las salidas que habían sido activadas.

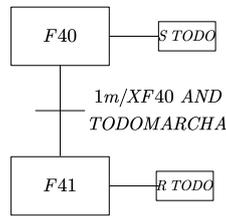


Figura 22: Macro etapa F4

Las ecuaciones de activación de cada estado quedarían así:

$$\begin{aligned}
 F40 &= A1 \cdot Manual + F40 \cdot \overline{F41} \cdot \overline{D1} \\
 F41 &= F40 \cdot TodoEnMarcha \cdot TON(IN := F40, PT := t\#1s) + F41 \cdot \overline{F3} \cdot \overline{F30} \cdot \overline{D1}
 \end{aligned}$$

5.2.2. Implementación

El GRAFCET superior se ha implementado en una función (FC) llamada Actualización de Estados. Mientras que los macro estados F1, F2, F3 y F4 se han implementado en bloques de función (FB), ya que es necesario acceder a sus variables internas (estados de cada macro estado). En ambos casos se han implementado las ecuaciones previamente presentadas en lenguaje Ladder.

Todas las funciones, ya sean FB o FC son llamadas desde el OB1 Main, en el que se ejecutan instrucciones o se llama a otros bloques de forma cíclica. Se ha intentado seguir el orden establecido en la literatura [25] para la llamada de las diferentes funciones, pero debido a la existencia de elementos extra, no se ha podido seguir esta estructura.

Además, se han implementado las funciones (FC):

- **Asignación Alarmas:** donde se vinculan los bits de cada palabra de aviso de HMI con una variable.
- **Asignación Entradas:** donde cada entrada del bloque de datos (DB) se asigna su correspondiente entrada física %I. Durante la simulación no se llama a esta función, sino a Simulación.
- **Ecuaciones de Activación:** aquí, mediante expresiones lógicas, de comparación, aritméticas y estructuras de control, se asignan valores a diferentes variables auxiliares. En la [Subsubsección 5.2.2.1](#) se explica con detalle las diferentes variables que se utilizan.
- **Ecuaciones de Salida:** teniendo en cuenta el diseño de los GRAFCET, se asignan las salidas físicas. En la [Subsubsección 5.2.2.2](#) se puede encontrar más información.
- **Fallo_EDM:** se comprueba si la activación de la salida corresponde a la confirmación de marcha correspondiente, en caso negativo, se activa un temporizador TON con una duración marcada por "*Variables GEMMA*". *tiempoFalloEDM*. Cuando este temporizador acaba de contar, activa el bit de fallo correspondiente.
- **MHJ-PLC-Lab-Function-S71500:** esta función se utiliza para comunicar PLC-Sim con Factory I/O, lo proporciona Factory I/O.
- **Simulación:** se realiza una asignación de entradas correspondiente a lo que se puede obtener del Factory I/O y el HMI, además se incluye la asignación de los word **PART TO EMIT** que se utiliza para la simulación en Factory I/O.
- **Temporizadores:** juntamente con **Fallo_EDM** son las funciones que contienen temporizadores, en este caso los que se encuentran en esta función temporizan cuando es la última vez que ha pasado un hueco en cada cinta general, si esta está activa.

Respecto a los bloques de datos (DB), se genera uno automáticamente al compilar los FB. Además, se han incluido cuatro más para ayudar a organizar la programación.

- **AlarmasHMI:** se encuentran las palabras (word) W0, W1, W2 y W3 que se utilizan en el HMI para mostrar avisos cuando algún bit de estos word se activen.
- **ED:** se establecen aquí las variables de entrada. Cuando está activa la simulación, se asignan los valores de estas variables mediante el bloque **Simulación**, como se ha explicado anteriormente; si no está activa la simulación, se asigna el valor de las entradas físicas. A lo largo de la programación del autómatas, siempre se llama a este bloque de datos en vez de a las entradas físicas.
- **Variables GEMMA:** se encuentran los bits del GRAFCET superior, así como todas las variables auxiliares necesarias para el desarrollo tanto de este GRAFCET como de los inferiores.
- **Variables_HMI:** se encuentran todas las variables relacionadas con el HMI, a excepción de los avisos.

5.2.2.1. Ecuaciones de Activación

Como se ha podido observar en los GRAFCET, hay muchas transiciones que son variables las cuales no habíamos definido aún, como son *ErrorCritico*, *CondicionesIniciales*, etc. Se va a exponer qué representan cada una de estas variables, pero la programación de cada una se puede encontrar en el [Apéndice B, página 23-1](#)

- ***ErrorCritico***: se activa cuando alguna señal crítica (fallo del guardamotor, error de EDM (del inglés, *External Device Monitoring*), paro de seguridad, etc.) lo hace, por lo que activa D1 y se desactivan el resto de las etapas, incluidas las del GRAFCET inferior.
- ***CondicionesIniciales***: se activa cuando ninguna fotocélula detecta ningún objeto entre el sensor y el reflector.
- ***FinCiclo***: es una señal que se activa mediante el HMI, para indicar que todas las cintas tienen que parar progresivamente hasta que todas estén paradas para volver al estado inicial de la planta.
- ***Manual***: se utiliza para iniciar la comprobación desordenada (F4), se activa mediante HMI, sólo se permite su activación cuando la planta está en el estado A1.
- ***UltimoHuevo***: si la planta está en marcha y hace más de dos minutos que las fotocélulas no detectan un huevo, la planta entra en modo "Fin de ciclo" y para las cintas progresivamente para volver al estado inicial de la planta.
- ***Peticion_nueva_disponible_HMI***: se trata de una variable que muestra o no un rectángulo blanco sobre la pantalla principal del HMI, para evitar que se cambie la petición de naves y cintas cuando la planta se encuentra en modo de preparación, alguna de las comprobaciones, en emergencia o a la espera de ser puesta la planta en estado inicial.
- ***PeticionNueva***: cada ciclo, se comprueba si la petición de huevos (que se realiza mediante HMI) es igual a la que ya había, en caso de cambiar, este bit se activa. Se desactiva una vez el estado de la planta llega a F2.
- ***OKAguasAbajo***: antes de activar F1 es necesario comprobar de nuevo si todas las cintas y mesas que tienen que estar activas lo están, esto se realiza para evitar que los huevos en movimiento lleguen a una cinta parada y colisionen o se precipiten.
- ***PetHuevosAnt***: array de 12 caracteres que almacena la petición anterior de huevos de cada nave (un carácter por nave). Se actualiza cuando se llega al final de la etapa de preparación con los valores de la petición actual.
- ***PetHuevosAct***: siendo del mismo tipo que ***PetHuevosAnt***, actualiza cada ciclo la petición de las naves, actualizándose cada ciclo según la combinación entre los botones "A", "B" y ***#PetHuevosNave***.
- ***NaveMaxAct***: a partir de la petición de huevos actual, se calcula de cada cinta

general cuál es la nave "más alta" de la que se piden huevos. Este entero se utiliza para calcular hasta qué parte de cada cinta general tiene que estar activa.

- ***PeticionNula***: este bit se actualiza cada ciclo para saber si se ha introducido una petición nueva sin naves o sin cintas antes de llegar a fin de ciclo o último huevo. Se utiliza para pasar del funcionamiento normal a reposo.
- ***ConfDM***: Array de caracteres que según el sensor de mesa que se haya activado marca la dirección, un carácter por mesa. Si sólo se activa el sensor FC0, la dirección de la mesa está parada; si sólo se activa el sensor FCA, la dirección está girando hacia A y si sólo se activa el sensor FCB, la dirección está girando hacia B. Si se activa más de un sensor a la vez o no se activa ninguno, marcará dirección 'X'.
- ***ConfCG***: array de bits de cada cinta general, un bit por parte se activa cada uno cuando recibe la correspondiente confirmación de marcha de las entradas.
- ***ConfCM***: array de bits de cada cinta de mesa, un bit por mesa se activa cada uno cuando recibe la correspondiente confirmación de marcha de las entradas.
- ***ConfCN***: array de bits de cada cinta de nave, un bit por nave se activa cada uno cuando recibe la correspondiente confirmación de marcha de las entradas.
- ***PeticionHuevosNave***: array de bits que almacena la combinación ON/OFF de cada nave en la HMI.
- ***CG_OK***: este array de bits compara si debiera estar activa cada cinta general con la confirmación de la marcha, activándose si todo está correcto.
- ***#DMOKInd***: este entero almacena cuántas direcciones de cintas están correctas, comparando su valor deseado con el de la confirmación de marcha.
- ***DM_OK***: si el entero ***#DMOKInd*** es igual al número de mesas, este bit se activa.
- ***#CMOKInd***: este entero almacena cuantas cintas de mesas están actuando correctamente, ya sea estando activas o desactivadas.
- ***CMOK***: si el entero ***#CMOKInd*** es igual al número de mesas, este bit se activa.
- ***#CNOKInd***: este entero almacena cuantas cintas de naves están actuando correctamente, ya sea estando activas o desactivadas.
- ***CNOK***: si el entero ***#CNOKInd*** es igual al número de naves, este bit se activa.
- ***ParteAct***: según el número de ***NaveMaxAct*** se calcula por cada cinta general, qué partes tienen que estar activas. En caso de que se quieran huevos de la *Parte X*, se tienen que activar todas las cintas generales desde *Parte 1* hasta *Parte X*.
- ***TodoParado***: utiliza las confirmaciones de marcha de todos los actuadores para activarse si todos se encuentran desactivados.

- **TodoEnMarcha**: utiliza las confirmaciones de marcha de todos los actuadores para activarse si todos se encuentran activos.
- **PART TO EMIT**: este word se utiliza específicamente para indicar al software Factory I/O que tiene que emitir los objetos azules.
- **cont**: este bit se activa cuando detecta un flanco del botón del HMI y él mismo no está activo. Se utiliza para pausar la continuación de los estados cuando se está en modo comprobación ordenada.
- **EstadoAct_HMI**: es un entero que según el estado del GRAFCET superior activo, coge un valor de entero.
- **Estado_CG_HMI**: array de enteros que representa el modo en el que se encuentran cada una de las cuatro partes de las cintas generales, un array por CG. Si tiene algún fallo, ya sea por guardamotor o EDM, el estado es 2, en caso contrario, tiene valor de un 1 o un 0 según si se encuentra activo o no, respectivamente.
- **Estado_M_HMI**: array de enteros con un funcionamiento parecido al de **Estado_CG_HMI**, pero puede tener un valor más, para indicar la dirección. Por lo que la asignación quedaría de la siguiente forma: 3 para error, 1 avanzar, 2 retroceder, 0 reposo.
- **Estado_CN_HMI**: array de enteros con funcionamiento idéntico a **Estado_CG_HMI**, pero muestra el estado de las cintas de las naves.
- **Simulación_activa**: bit que inicia siempre con valor *TRUE*, activa el FC de simulación y comunicación con Factory I/O y desactiva la asignación de entradas.

5.2.2.2. Ecuaciones de salida

Teniendo en cuenta los GRAFCET anteriormente expuestos, se han definido las diferentes condiciones de activación y desactivación de las cintas. Como se ha podido observar en los diferentes GRAFCET inferiores, todas las salidas tienen activación (S) y desactivación (R). Es por ello por lo que se ha decidido programar mediante SCL, con sentencias condicionales.

Como ejemplo, estas son las ecuaciones de activación y desactivación de CGA1 en forma de pseudocódigo:

$$\begin{aligned}CGA1_S &= F2_F21 \cdot Parte1_A + F4_F40 \\CGA1_R &= D1 + F2_F20 + F3_F52 + A1\end{aligned}$$

Escrito en código de SCL tiene la siguiente forma:

```
1      IF ("F2_DB".F21 AND "Variables GEMMA".ParteAct_A[1]) OR "F4_DB".F40 THEN
2          "CGA1" := 1;
3          "CGA1 PT2" := 1;
4      ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F52 OR "Variables GEMMA".A1
5      ↪ OR "F4_DB".F41 THEN
6          "CGA1" := 0;
7          "CGA1 PT2" := 0;
      END_IF;
```

Código 1: Ecuaciones de salida cinta general A parte 1

Ya que hay una gran cantidad de código y las salidas se han especificado con anterioridad, éste va a ser el único fragmento de código en este documento, dejando el resto expuesto en el [Apéndice B](#).

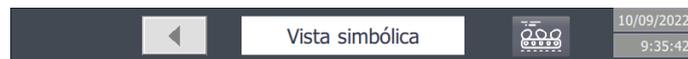
Es necesario comentar, que la disposición de salidas físicas del PLC en [Apéndice B](#) no corresponde a la planta ideada, si no a las salidas necesarias para la correcta programación de la simulación. Es por ello por lo que se encuentran salidas duplicadas, todas ellas llevan en el nombre .1, .2, .3 o PT2. Así como las salidas tipo WORD necesarias para la emisión de los objetos azules, que tampoco estaría en la implementación real.

5.3. HMI

La interfaz hombre máquina consta de seis pantallas diferentes. Respecto la navegación, es bastante intuitiva.

Todas las pantallas cuentan con la misma barra inferior, donde podemos encontrar botones y elementos con diferentes funciones, de izquierda a derecha: botón con forma de casa, activa la pantalla principal; botón con un triángulo y una exclamación, activa la pantalla de avisos activos; en el centro se encuentra un visor de avisos, donde se pueden ver todos los avisos activos desde todas las pantallas; por último, se encuentra el botón de salida, que cierra el runtime de la HMI.

En la barra superior se encuentra el nombre de la pantalla actual, un botón con un dibujo de unos huevos sobre una cinta transportadora, que abre la vista simbólica y, por último, la fecha y la hora. En todas las pantallas menos en la principal, a la izquierda del nombre de la pantalla hay un botón con forma de flecha que lleva a la pantalla anterior.



(a) Barra superior pantalla vista simbólica



(b) Barra inferior

Figura 23: Barras comunes a todas las pantallas

5.3.1. Pantalla principal

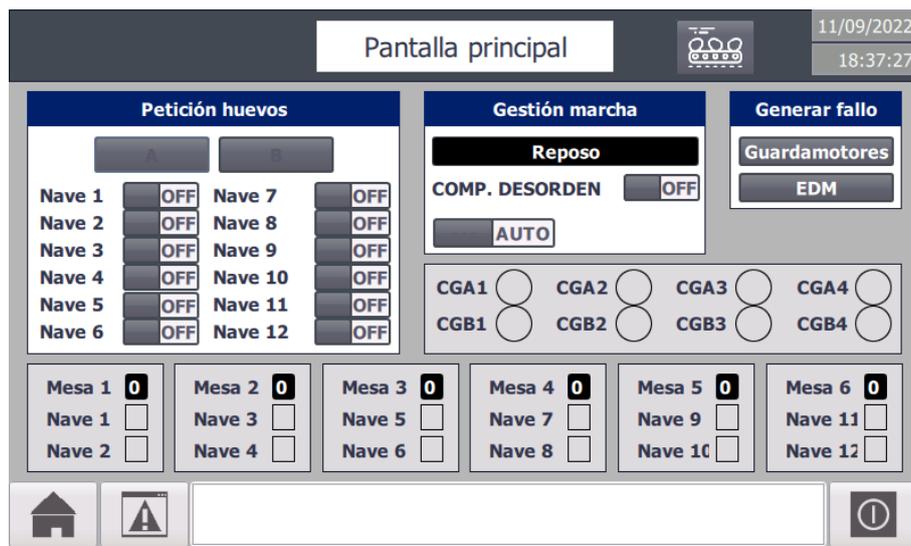


Figura 24: Pantalla principal

La pantalla principal se divide en dos partes: la blanca, donde el operario puede pulsar botones para cambiar el estado de la planta o ir a otras pantallas; y la gris, donde el HMI muestra el estado de los diferentes elementos de forma muy directa.



Figura 25: Pantalla principal blanco

En la parte blanca se encuentra la petición de huevos, para realizar una petición correctamente tiene que haber al menos una cinta general y una nave, la letra del botón de cada cinta general tiene color gris cuando está desactivado y blanco cuando está activo; en cambio, las naves se activan mediante botones de ON/OFF situados a su derecha, para cambiar la petición de la nave hay que pulsar la parte gris de dicho botón, cabe tener en cuenta que el cambio de petición sólo se puede realizar cuando la planta está o en reposo o en F1. En esta parte también se encuentra la gestión de marcha, que cuenta con un recuadro que indica la etapa en la que se encuentra la planta: Reposo para A1, Preparación para F2, Marcha para F1, Finalizando para F3, Comp. orden para F5, Comp. desorden para F4, EMERGENCIA para D1 y Espera C.I. para A6; además, cuando la planta se encuentra en reposo se pueden activar el bit "Manual" con el interruptor a la izquierda del texto "COMP. DESORDEN" y se puede activar el bit "SecuenciaManual" con el interruptor que indica "MAN/AUTO", cuando se active, a su derecha aparecerá un botón para continuar con la secuencia ordenada. Este espacio quedará cubierto por un recuadro blanco cuando la planta esté en un estado diferente de Reposo. A la izquierda de la pantalla también se encuentran dos botones, llamados "Guardamotores" y "EDM", que activan sendas pantallas para generar fallos en la planta, si el bit "Simulacion_activa" se encuentra desactivado, estos botones no serán visibles.

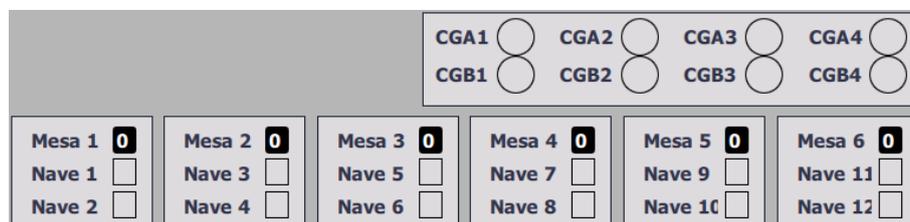


Figura 26: Pantalla principal gris

En la parte gris, por un lado, se encuentran las cintas generales, y por otro las mesas con cada pareja de naves. Tanto los cuadrados al lado de las naves como los círculos al lado de las cintas generales cuentan con el mismo código de colores: gris si se encuentran en reposo o apagados, verde si están activos y rojo parpadeante con amarillo si hay algún fallo de ese

elemento, ya sea de guardamotor o EDM. Las mesas cuentan con un recuadro negro con la letra '0' cuando están inactivas, el recuadro cambia a color verde con el avance y muestra la letra 'A', mientras que si retrocede muestra la letra 'R' sobre un fondo fucsia; si la mesa tuviera algún fallo en la cinta o en la dirección la letra mostrada sería la 'E' con un color parpadeante en amarillo y rojo alternado con el fondo de los mismos colores.

5.3.2. Pantalla vista simbólica

La pantalla de vista simbólica sólo sirve para mostrar de forma más gráfica el estado de los diferentes elementos de la planta en su conjunto. Mostrándose con fondo blanco cuando están parados, en verde cuando están en marcha y en rojo si hay algún problema, ya sea con el guardamotor o con el EDM. La imagen completa corresponde a la Figura 1, aunque con los correspondientes cambios de color de los elementos ya mencionados, las mesas además cuando se encuentran en marcha tienen líneas oblicuas que cambian de color según la dirección de la mesa y también cuenta con una flecha que indica lo mismo.

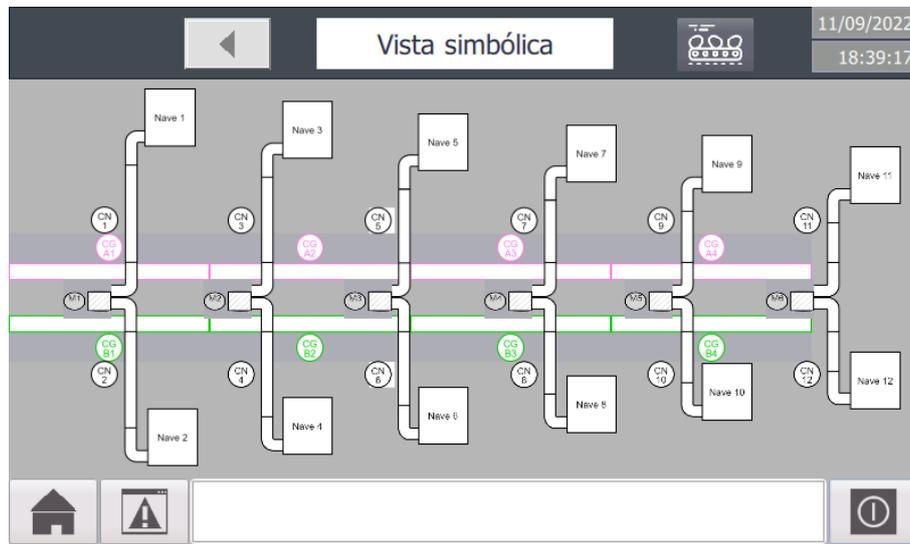


Figura 27: Pantalla vista simbólica

5.3.3. Pantalla Fallo Guardamotores y Fallo EDM

Se han agrupado estas dos pantallas ya que la distribución es bastante parecida y el funcionamiento es idéntico. Cada pantalla está dividida en cuatro partes, con cada cinta y mesa por parte, a su derecha hay uno o dos botones que inicialmente están en blanco, con la palabra "OK". Si se pulsa algún botón, se genera el correspondiente fallo, de guardamotor o EDM según en la pantalla que nos encontremos; el botón cambia a ser de color verde y la palabra indica "NOK". Para quitar el fallo, será necesario volver a pulsar sobre el mismo botón por el cual se ha generado.

Parte 1		Parte 2	
Cinta	Estado guardamotor	Cinta	Estado guardamotor
CGA1	OK	CGA2	OK
CGB1	OK	CGB2	OK
Mesa 1	OK	Mesa 2	OK
Nave 1	OK	Nave 3	OK
Nave 2	OK	Nave 4	OK

Parte 3		Parte 4	
Cinta	Estado guardamotor	Cinta	Estado guardamotor
CGA3	OK	CGA4	OK
CGB3	OK	CGB4	OK
Mesa 3	OK	Mesa 5	OK
Nave 5	OK	Nave 9	OK
Nave 6	OK	Nave 10	OK
Mesa 4	OK	Mesa 6	OK
Nave 7	OK	Nave 11	OK
Nave 8	OK	Nave 12	OK

Figura 28: Pantalla fallo guardamotors

Cabe indicar que los fallos de guardamotor son inmediatos, mientras que los de EDM tardan un tiempo en aparecer establecido por la variable "*Variables GEM-MA*". *tiempoFalloEDM*.

Parte 1		Parte 2	
Cinta	Estado conf. marcha	Cinta	Estado conf. marcha
CGA1	OK	CGA2	OK
CGB1	OK	CGB2	OK
Mesa 1	OK	Mesa 2	OK
Nave 1	OK	Nave 3	OK
Nave 2	OK	Nave 4	OK

Parte 3		Parte 4	
Cinta	Estado conf. marcha	Cinta	Estado conf. marcha
CGA3	OK	CGA4	OK
CGB3	OK	CGB4	OK
Mesa 3	OK	Mesa 5	OK
Nave 5	OK	Nave 9	OK
Nave 6	OK	Nave 10	OK
Mesa 4	OK	Mesa 6	OK
Nave 7	OK	Nave 11	OK
Nave 8	OK	Nave 12	OK

Figura 29: Pantalla fallo EDM

5.3.4. Pantalla Alarmas Activas y Alarmas Históricas

En la pantalla de alarmas activas se encuentra un visor de avisos más grande y con más información que el de la barra inferior. Además, también hay un botón que abre la pantalla de alarmas históricas.

En la pantalla de alarmas históricas hay un visor con todas las alarmas que ha habido

desde que el runtime del HMI se activó si no se han borrado con el botón en la parte inferior de la pantalla, con texto "Borrar histórico".



Figura 30: Pantalla alarmas activas



Figura 31: Pantalla alarmas históricas

Apéndices

Apéndice A

Tabla variables Factory I/O

Tabla 1: Tabla variables Factory I/O

name	Tag	type	addr
Simulacion_TFG	Arranque	Bool	%I0.0
Simulacion_TFG	Rearme	Bool	%I0.1
Simulacion_TFG	FOTOCELULA A1	Bool	%I0.2
Simulacion_TFG	FOTOCELULA A2	Bool	%I0.3
Simulacion_TFG	FOTOCELULA A3	Bool	%I0.4
Simulacion_TFG	FOTOCELULA A4	Bool	%I0.5
Simulacion_TFG	FOTOCELULA B1	Bool	%I0.6
Simulacion_TFG	FOTOCELULA B2	Bool	%I0.7
Simulacion_TFG	FOTOCELULA B3	Bool	%I1.0
Simulacion_TFG	FOTOCELULA B4	Bool	%I1.1
Simulacion_TFG	PARO SEGURIDAD	Bool	%I1.2
Simulacion_TFG	CGA1 PT1	Bool	%Q0.0
Simulacion_TFG	CGA2 PT1	Bool	%Q0.1
Simulacion_TFG	CGA3 PT1	Bool	%Q0.2
Simulacion_TFG	CGA4 PT1	Bool	%Q0.3
Simulacion_TFG	CGB1 PT1	Bool	%Q0.4
Simulacion_TFG	CGB2 PT1	Bool	%Q0.5
Simulacion_TFG	CGB3 PT1	Bool	%Q0.6
Simulacion_TFG	CGB4 PT1	Bool	%Q0.7
Simulacion_TFG	Cinta M1	Bool	%Q1.0
Simulacion_TFG	CINTA M2	Bool	%Q1.1
Simulacion_TFG	CINTA M3	Bool	%Q1.2
Simulacion_TFG	CINTA M4	Bool	%Q1.3
Simulacion_TFG	CINTA M5	Bool	%Q1.4
Simulacion_TFG	CINTA M6	Bool	%Q1.5

Apéndice A Tabla variables Factory I/O

Simulacion_TFG	CN1.1	Bool	%Q1.6
Simulacion_TFG	CN2.1	Bool	%Q1.7
Simulacion_TFG	CN3.1	Bool	%Q2.0
Simulacion_TFG	CN4.1	Bool	%Q2.1
Simulacion_TFG	CN5.1	Bool	%Q2.2
Simulacion_TFG	CN6.1	Bool	%Q2.3
Simulacion_TFG	CN7.1	Bool	%Q2.4
Simulacion_TFG	CN8.1	Bool	%Q2.5
Simulacion_TFG	CN9.1	Bool	%Q2.6
Simulacion_TFG	CN10.1	Bool	%Q2.7
Simulacion_TFG	CN11.1	Bool	%Q3.0
Simulacion_TFG	CN12.1	Bool	%Q3.1
Simulacion_TFG	M1 AVANZAR	Bool	%Q3.2
Simulacion_TFG	M1 RETROCEDER	Bool	%Q3.3
Simulacion_TFG	M2 AVANZAR	Bool	%Q3.4
Simulacion_TFG	M2 RETROCEDER	Bool	%Q3.5
Simulacion_TFG	M3 AVANZAR	Bool	%Q3.6
Simulacion_TFG	M3 RETROCEDER	Bool	%Q3.7
Simulacion_TFG	M4 AVANZAR	Bool	%Q4.0
Simulacion_TFG	M4 RETROCEDER	Bool	%Q4.1
Simulacion_TFG	M5 AVANZAR	Bool	%Q4.2
Simulacion_TFG	M5 RETROCEDER	Bool	%Q4.3
Simulacion_TFG	M6 AVANZAR	Bool	%Q4.4
Simulacion_TFG	M6 RETROCEDER	Bool	%Q4.5
Simulacion_TFG	CGA1 PT2	Bool	%Q4.6
Simulacion_TFG	CGA4 PT2	Bool	%Q4.7
Simulacion_TFG	CGB1 PT2	Bool	%Q5.0
Simulacion_TFG	CGB4 PT2	Bool	%Q5.1
Simulacion_TFG	Cinta M1.1	Bool	%Q5.2
Simulacion_TFG	Cinta M1.2	Bool	%Q5.3
Simulacion_TFG	Cinta M1.3	Bool	%Q5.4
Simulacion_TFG	CINTA M2.1	Bool	%Q5.5
Simulacion_TFG	CINTA M2.2	Bool	%Q5.6
Simulacion_TFG	CINTA M2.3	Bool	%Q5.7
Simulacion_TFG	CINTA M3.1	Bool	%Q6.0
Simulacion_TFG	CINTA M3.2	Bool	%Q6.1
Simulacion_TFG	CINTA M3.3	Bool	%Q6.2
Simulacion_TFG	CINTA M4.1	Bool	%Q6.3

Simulacion_TFG	CINTA M4.2	Bool	%Q6.4
Simulacion_TFG	CINTA M4.3	Bool	%Q6.5
Simulacion_TFG	CINTA M5.1	Bool	%Q6.6
Simulacion_TFG	CINTA M5.2	Bool	%Q6.7
Simulacion_TFG	CINTA M5.3	Bool	%Q7.0
Simulacion_TFG	CINTA M6.1	Bool	%Q7.1
Simulacion_TFG	CINTA M6.2	Bool	%Q7.2
Simulacion_TFG	CINTA M6.3	Bool	%Q7.3
Simulacion_TFG	CN1.2	Bool	%Q7.4
Simulacion_TFG	CN1.4	Bool	%Q7.5
Simulacion_TFG	CN2.2	Bool	%Q7.6
Simulacion_TFG	CN2.4	Bool	%Q7.7
Simulacion_TFG	CN3.2	Bool	%Q8.0
Simulacion_TFG	CN3.4	Bool	%Q8.1
Simulacion_TFG	CN4.2	Bool	%Q8.2
Simulacion_TFG	CN4.4	Bool	%Q8.3
Simulacion_TFG	CN5.2	Bool	%Q8.4
Simulacion_TFG	CN6.2	Bool	%Q8.5
Simulacion_TFG	CN7.2	Bool	%Q8.6
Simulacion_TFG	CN8.2	Bool	%Q8.7
Simulacion_TFG	CN9.2	Bool	%Q9.0
Simulacion_TFG	CN10.2	Bool	%Q9.1
Simulacion_TFG	CN11.2	Bool	%Q9.2
Simulacion_TFG	CN12.2	Bool	%Q9.3
Simulacion_TFG	M1 AVANZAR.1	Bool	%Q9.4
Simulacion_TFG	M1.1 AVANZAR	Bool	%Q9.5
Simulacion_TFG	M2 AVANZAR.1	Bool	%Q9.6
Simulacion_TFG	M2.1 AVANZAR	Bool	%Q9.7
Simulacion_TFG	M3 AVANZAR.1	Bool	%Q10.0
Simulacion_TFG	M3.1 AVANZAR	Bool	%Q10.1
Simulacion_TFG	M4 AVANZAR.1	Bool	%Q10.2
Simulacion_TFG	M4.1 AVANZAR	Bool	%Q10.3
Simulacion_TFG	M5 AVANZAR.1	Bool	%Q10.4
Simulacion_TFG	M5.1 AVANZAR	Bool	%Q10.5
Simulacion_TFG	M6 AVANZAR.1	Bool	%Q10.6
Simulacion_TFG	M6.1 AVANZAR	Bool	%Q10.7
Simulacion_TFG	M1 RETROCEDER.1	Bool	%Q11.0
Simulacion_TFG	M1.1 RETROCEDER	Bool	%Q11.1

Apéndice A Tabla variables Factory I/O

Simulacion_TFG	M2 RETROCEDER.1	Bool	%Q11.2
Simulacion_TFG	M2.1 RETROCEDER	Bool	%Q11.3
Simulacion_TFG	M3 RETROCEDER.1	Bool	%Q11.4
Simulacion_TFG	M3.1 RETROCEDER	Bool	%Q11.5
Simulacion_TFG	M4 RETROCEDER.1	Bool	%Q11.6
Simulacion_TFG	M4.1 RETROCEDER	Bool	%Q11.7
Simulacion_TFG	M5 RETROCEDER.1	Bool	%Q12.0
Simulacion_TFG	M5.1 RETROCEDER	Bool	%Q12.1
Simulacion_TFG	M6 RETROCEDER.1	Bool	%Q12.2
Simulacion_TFG	M6.1 RETROCEDER	Bool	%Q12.3
Simulacion_TFG	NAVE 1	Bool	%Q12.4
Simulacion_TFG	NAVE 2	Bool	%Q12.5
Simulacion_TFG	NAVE 3	Bool	%Q12.6
Simulacion_TFG	NAVE 4	Bool	%Q12.7
Simulacion_TFG	NAVE 5	Bool	%Q13.0
Simulacion_TFG	NAVE 6	Bool	%Q13.1
Simulacion_TFG	NAVE 7	Bool	%Q13.2
Simulacion_TFG	NAVE 8	Bool	%Q13.3
Simulacion_TFG	NAVE 9	Bool	%Q13.4
Simulacion_TFG	NAVE 10	Bool	%Q13.5
Simulacion_TFG	NAVE 11	Bool	%Q13.6
Simulacion_TFG	NAVE 12	Bool	%Q13.7
Simulacion_TFG	PART TO EMIT 1	Int	%QW30
Simulacion_TFG	PART TO EMIT 2	Int	%QW32
Simulacion_TFG	PART TO EMIT 3	Int	%QW34
Simulacion_TFG	PART TO EMIT 4	Int	%QW36
Simulacion_TFG	PART TO EMIT 5	Int	%QW38
Simulacion_TFG	PART TO EMIT 6	Int	%QW40
Simulacion_TFG	PART TO EMIT 7	Int	%QW42
Simulacion_TFG	PART TO EMIT 8	Int	%QW44
Simulacion_TFG	PART TO EMIT 9	Int	%QW46
Simulacion_TFG	PART TO EMIT 10	Int	%QW48
Simulacion_TFG	PART TO EMIT 11	Int	%QW50
Simulacion_TFG	PART TO EMIT 12	Int	%QW52

Apéndice B

Programación PLC

Como se ha explicado en [Capítulo 5](#), la programación se ha realizado utilizando diferentes FC y FB, así como DB para organizar las variables.

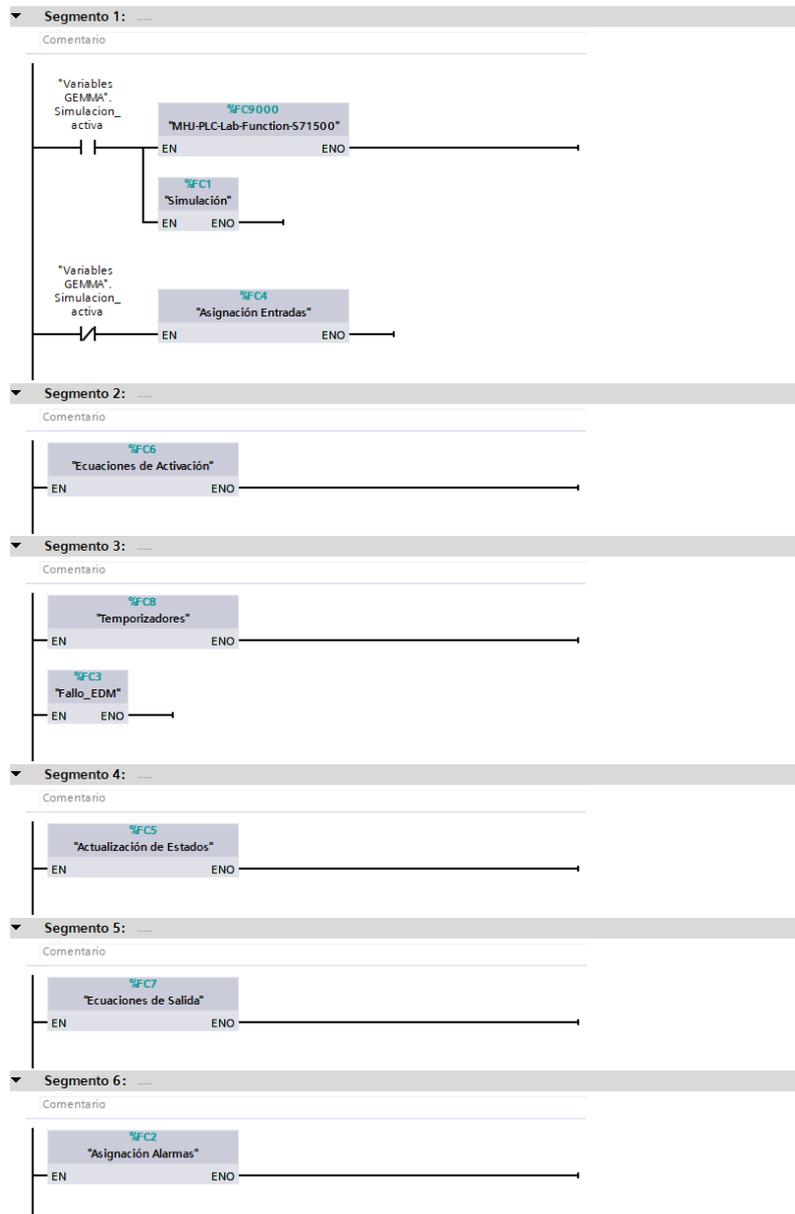


Figura 32: Bloques de programa

Todos ellos son llamados por el bloque de organización OB1 Main. Se va a explicar la programación de todas las funciones y bloques que se han programado de cero, es decir, todos menos el MHJ-PLC-Lab-Function-S71500 [FC9000], ya que como se ha explicado anteriormente, lo proporciona Factory I/O.

B.1. Main [OB1]

Primero, según el valor de la variable "Variables GEMMA".Simulación_activa, se llama a los bloques de simulación (MHJ-PLC-Lab-Function-S71500 y Simulación) o a "Asignación de entradas". En los siguientes segmentos se siguen los pasos dictados por la literatura [25]: inicialmente las ecuaciones de activación, donde se actualizan los valores de las variables auxiliares; seguido de las funciones con temporizadores (Temporizadores y Fallo_EDM), continuado por la actualización de los estados (tanto del GRAFCET superior como del inferior), seguido por la asignación de salidas físicas. Por último, como extra necesario se encuentra la Asignación de Alarmas.



Código 2: Main

B.2. Simulación [FC1]

En esta función se asignan las variables del bloque de datos "ED" para que tengan los valores deseados, que en este caso son valores o calculados o de entradas de Factory I/O. Además, también está la asignación de salidas físicas de los word de los "Emitters". Toda esta función está programada en SCL.

```

1
2 REGION Entradas Factory I/O
3   "ED"."Arranque" := "Arranque";
4   "ED"."Rearme"  := "Rearme";
5   "ED"."Fotocélula A1" := "Fotocélula A1";
6   "ED"."Fotocélula A2" := "Fotocélula A2";
7   "ED"."Fotocélula A3" := "Fotocélula A3";
8   "ED"."Fotocélula A4" := "Fotocélula A4";
9   "ED"."Fotocélula B1" := "Fotocélula B1";
10  "ED"."Fotocélula B2" := "Fotocélula B2";
11  "ED"."Fotocélula B3" := "Fotocélula B3";
12  "ED"."Fotocélula B4" := "Fotocélula B4";
13  "ED"."PARO SEGURIDAD" := NOT "PARO SEGURIDAD";
14 END_REGION
15
16
17 REGION Entradas inicialización primer scan
18 IF "FirstScan" THEN
19   "ED"."Mesa 1 FC0" := 1;
20   "ED"."Mesa 2 FC0" := 1;
21   "ED"."Mesa 3 FC0" := 1;
22   "ED"."Mesa 4 FC0" := 1;
23   "ED"."Mesa 5 FC0" := 1;
24   "ED"."Mesa 6 FC0" := 1;
25 END_IF;
26 END_REGION
27
28
29 REGION Entradas confirmación de marcha con fallo EDM
30 "ED"."Marcha cinta A pt1" := "CGA1" XOR "Variables_HMI".Fallo_EDM_CGA_HMI[1];
31 "ED"."Marcha cinta A pt2" := "CGA2" XOR "Variables_HMI".Fallo_EDM_CGA_HMI[2];
32 "ED"."Marcha cinta A pt3" := "CGA3" XOR "Variables_HMI".Fallo_EDM_CGA_HMI[3];
33 "ED"."Marcha cinta A pt4" := "CGA4" XOR "Variables_HMI".Fallo_EDM_CGA_HMI[4];
34
35 "ED"."Marcha cinta B pt1" := "CGB1" XOR "Variables_HMI".Fallo_EDM_CGB_HMI[1];
36 "ED"."Marcha cinta B pt2" := "CGB2" XOR "Variables_HMI".Fallo_EDM_CGB_HMI[2];
37 "ED"."Marcha cinta B pt3" := "CGB3" XOR "Variables_HMI".Fallo_EDM_CGB_HMI[3];
38 "ED"."Marcha cinta B pt4" := "CGB4" XOR "Variables_HMI".Fallo_EDM_CGB_HMI[4];
39
40
41 "ED"."Marcha nave 1" := "NAVE 1" XOR "Variables_HMI".Fallo_EDM_CN_HMI[1];
42 "ED"."Marcha nave 2" := "NAVE 2" XOR "Variables_HMI".Fallo_EDM_CN_HMI[2];
43 "ED"."Marcha nave 3" := "NAVE 3" XOR "Variables_HMI".Fallo_EDM_CN_HMI[3];
44 "ED"."Marcha nave 4" := "NAVE 4" XOR "Variables_HMI".Fallo_EDM_CN_HMI[4];
45 "ED"."Marcha nave 5" := "NAVE 5" XOR "Variables_HMI".Fallo_EDM_CN_HMI[5];
46 "ED"."Marcha nave 6" := "NAVE 6" XOR "Variables_HMI".Fallo_EDM_CN_HMI[6];
47 "ED"."Marcha nave 7" := "NAVE 7" XOR "Variables_HMI".Fallo_EDM_CN_HMI[7];
48 "ED"."Marcha nave 8" := "NAVE 8" XOR "Variables_HMI".Fallo_EDM_CN_HMI[8];
49 "ED"."Marcha nave 9" := "NAVE 9" XOR "Variables_HMI".Fallo_EDM_CN_HMI[9];

```

```

50 "ED"."Marcha nave 10" := "NAVE 10" XOR "Variables_HMI".Fallo_EDM_CN_HMI[10];
51 "ED"."Marcha nave 11" := "NAVE 11" XOR "Variables_HMI".Fallo_EDM_CN_HMI[11];
52 "ED"."Marcha nave 12" := "NAVE 12" XOR "Variables_HMI".Fallo_EDM_CN_HMI[12];
53
54
55 "ED"."Marcha cinta mesa 1" := "CINTA M1" XOR "Variables_HMI".Fallo_EDM_CM_HMI[1];
56 "ED"."Marcha cinta mesa 2" := "CINTA M2" XOR "Variables_HMI".Fallo_EDM_CM_HMI[2];
57 "ED"."Marcha cinta mesa 3" := "CINTA M3" XOR "Variables_HMI".Fallo_EDM_CM_HMI[3];
58 "ED"."Marcha cinta mesa 4" := "CINTA M4" XOR "Variables_HMI".Fallo_EDM_CM_HMI[4];
59 "ED"."Marcha cinta mesa 5" := "CINTA M5" XOR "Variables_HMI".Fallo_EDM_CM_HMI[5];
60 "ED"."Marcha cinta mesa 6" := "CINTA M6" XOR "Variables_HMI".Fallo_EDM_CM_HMI[6];
61
62 IF ("M1 AVANZAR" AND NOT "M1 RETROCEDER") XOR "Variables_HMI".Fallo_EDM_DM_HMI[1]
↳ THEN
63     "ED"."Mesa 1 FC0" := 0;
64     "ED"."Mesa 1 FCB" := 1;
65     "ED"."Mesa 1 FCA" := 0;
66     ELSIF (NOT "M1 AVANZAR" AND "M1 RETROCEDER") XOR
↳ "Variables_HMI".Fallo_EDM_DM_HMI[1] THEN
67         "ED"."Mesa 1 FC0" := 0;
68         "ED"."Mesa 1 FCB" := 0;
69         "ED"."Mesa 1 FCA" := 1;
70     ELSIF (NOT "M1 AVANZAR" AND NOT "M1 RETROCEDER") XOR
↳ "Variables_HMI".Fallo_EDM_DM_HMI[1] THEN
71         "ED"."Mesa 1 FC0" := 1;
72         "ED"."Mesa 1 FCB" := 0;
73         "ED"."Mesa 1 FCA" := 0;
74     ELSE
75         "ED"."Mesa 1 FC0" := 0;
76         "ED"."Mesa 1 FCB" := 0;
77         "ED"."Mesa 1 FCA" := 0;
78     END_IF;
79
80
81 IF ("M2 AVANZAR" AND NOT "M2 RETROCEDER") XOR "Variables_HMI".Fallo_EDM_DM_HMI[2]
↳ THEN
82     "ED"."Mesa 2 FC0" := 0;
83     "ED"."Mesa 2 FCB" := 1;
84     "ED"."Mesa 2 FCA" := 0;
85     ELSIF (NOT "M2 AVANZAR" AND "M2 RETROCEDER") XOR
↳ "Variables_HMI".Fallo_EDM_DM_HMI[2] THEN
86         "ED"."Mesa 2 FC0" := 0;
87         "ED"."Mesa 2 FCB" := 0;
88         "ED"."Mesa 2 FCA" := 1;
89     ELSIF (NOT "M2 AVANZAR" AND NOT "M2 RETROCEDER") XOR
↳ "Variables_HMI".Fallo_EDM_DM_HMI[2] THEN
90         "ED"."Mesa 2 FC0" := 1;
91         "ED"."Mesa 2 FCB" := 0;
92         "ED"."Mesa 2 FCA" := 0;
93     ELSE
94         "ED"."Mesa 2 FC0" := 0;
95         "ED"."Mesa 2 FCB" := 0;
96         "ED"."Mesa 2 FCA" := 0;
97     END_IF;
98
99
100 IF ("M3 AVANZAR" AND NOT "M3 RETROCEDER") XOR "Variables_HMI".Fallo_EDM_DM_HMI[3]
↳ THEN

```

```

101     "ED"."Mesa 3 FC0" := 0;
102     "ED"."Mesa 3 FCB" := 1;
103     "ED"."Mesa 3 FCA" := 0;
104     ELSIF (NOT "M3 AVANZAR" AND "M3 RETROCEDER") XOR
↪     "Variables_HMI".Fallo_EDM_DM_HMI[3] THEN
105         "ED"."Mesa 3 FC0" := 0;
106         "ED"."Mesa 3 FCB" := 0;
107         "ED"."Mesa 3 FCA" := 1;
108     ELSIF (NOT "M3 AVANZAR" AND NOT "M3 RETROCEDER") XOR
↪     "Variables_HMI".Fallo_EDM_DM_HMI[3] THEN
109         "ED"."Mesa 3 FC0" := 1;
110         "ED"."Mesa 3 FCB" := 0;
111         "ED"."Mesa 3 FCA" := 0;
112     ELSE
113         "ED"."Mesa 3 FC0" := 0;
114         "ED"."Mesa 3 FCB" := 0;
115         "ED"."Mesa 3 FCA" := 0;
116     END_IF;
117
118
119     IF ("M4 AVANZAR" AND NOT "M4 RETROCEDER") XOR "Variables_HMI".Fallo_EDM_DM_HMI[4]
↪     THEN
120         "ED"."Mesa 4 FC0" := 0;
121         "ED"."Mesa 4 FCB" := 1;
122         "ED"."Mesa 4 FCA" := 0;
123     ELSIF (NOT "M4 AVANZAR" AND "M4 RETROCEDER") XOR
↪     "Variables_HMI".Fallo_EDM_DM_HMI[4] THEN
124         "ED"."Mesa 4 FC0" := 0;
125         "ED"."Mesa 4 FCB" := 0;
126         "ED"."Mesa 4 FCA" := 1;
127     ELSIF (NOT "M4 AVANZAR" AND NOT "M4 RETROCEDER") XOR
↪     "Variables_HMI".Fallo_EDM_DM_HMI[4] THEN
128         "ED"."Mesa 4 FC0" := 1;
129         "ED"."Mesa 4 FCB" := 0;
130         "ED"."Mesa 4 FCA" := 0;
131     ELSE
132         "ED"."Mesa 4 FC0" := 0;
133         "ED"."Mesa 4 FCB" := 0;
134         "ED"."Mesa 4 FCA" := 0;
135     END_IF;
136
137
138     IF ("M5 AVANZAR" AND NOT "M5 RETROCEDER") XOR "Variables_HMI".Fallo_EDM_DM_HMI[5]
↪     THEN
139         "ED"."Mesa 5 FC0" := 0;
140         "ED"."Mesa 5 FCB" := 1;
141         "ED"."Mesa 5 FCA" := 0;
142     ELSIF (NOT "M5 AVANZAR" AND "M5 RETROCEDER") XOR
↪     "Variables_HMI".Fallo_EDM_DM_HMI[5] THEN
143         "ED"."Mesa 5 FC0" := 0;
144         "ED"."Mesa 5 FCB" := 0;
145         "ED"."Mesa 5 FCA" := 1;
146     ELSIF (NOT "M5 AVANZAR" AND NOT "M5 RETROCEDER") XOR
↪     "Variables_HMI".Fallo_EDM_DM_HMI[5] THEN
147         "ED"."Mesa 5 FC0" := 1;
148         "ED"."Mesa 5 FCB" := 0;
149         "ED"."Mesa 5 FCA" := 0;
150     ELSE

```

```
151     "ED"."Mesa 5 FC0" := 0;
152     "ED"."Mesa 5 FCB" := 0;
153     "ED"."Mesa 5 FCA" := 0;
154 END_IF;
155
156
157 IF ("M6 AVANZAR" AND NOT "M6 RETROCEDER") XOR "Variables_HMI".Fallo_EDM_DM_HMI[6]
↪ THEN
158     "ED"."Mesa 6 FC0" := 0;
159     "ED"."Mesa 6 FCB" := 1;
160     "ED"."Mesa 6 FCA" := 0;
161 ELSIF (NOT "M6 AVANZAR" AND "M6 RETROCEDER") XOR
↪ "Variables_HMI".Fallo_EDM_DM_HMI[6] THEN
162     "ED"."Mesa 6 FC0" := 0;
163     "ED"."Mesa 6 FCB" := 0;
164     "ED"."Mesa 6 FCA" := 1;
165 ELSIF (NOT "M6 AVANZAR" AND NOT "M6 RETROCEDER") XOR
↪ "Variables_HMI".Fallo_EDM_DM_HMI[6] THEN
166     "ED"."Mesa 6 FC0" := 1;
167     "ED"."Mesa 6 FCB" := 0;
168     "ED"."Mesa 6 FCA" := 0;
169 ELSE
170     "ED"."Mesa 6 FC0" := 0;
171     "ED"."Mesa 6 FCB" := 0;
172     "ED"."Mesa 6 FCA" := 0;
173 END_IF;
174 END_REGION
175
176
177 REGION Salidas exclusivas Factory I/O
178     "PART TO EMIT 1" := 16;
179     "PART TO EMIT 2" := 16;
180     "PART TO EMIT 3" := 16;
181     "PART TO EMIT 4" := 16;
182     "PART TO EMIT 5" := 16;
183     "PART TO EMIT 6" := 16;
184     "PART TO EMIT 7" := 16;
185     "PART TO EMIT 8" := 16;
186     "PART TO EMIT 9" := 16;
187     "PART TO EMIT 10" := 16;
188     "PART TO EMIT 11" := 16;
189     "PART TO EMIT 12" := 16;
190 END_REGION
191
```

Código 3: Simulación

El resto de las variables dentro del bloque de datos "ED" está asignado en la propia HMI, ya que la modificación de estos valores está restringida. Sólo es posible si "Variables GEMMA".Simulacion_activa está activo; de esta forma nos aseguramos de que no haya conflictos de variables que se sobrescriben.

B.3. Asignación Entradas [FC4]

Al contrario que Sección B.2, en esta función se utilizan todas las entradas físicas para definir las variables del bloque de datos ED.

```

1      "ED"."Arranque" := "Arranque";
2
3      "ED"."Rearme" := "Rearme";
4
5      "ED"."Fotocélula A1" := "Fotocélula A1";
6      "ED"."Fotocélula A2" := "Fotocélula A2";
7      "ED"."Fotocélula A3" := "Fotocélula A3";
8      "ED"."Fotocélula A4" := "Fotocélula A4";
9      "ED"."Fotocélula B1" := "Fotocélula B1";
10     "ED"."Fotocélula B2" := "Fotocélula B2";
11     "ED"."Fotocélula B3" := "Fotocélula B3";
12     "ED"."Fotocélula B4" := "Fotocélula B4";
13     "ED"."PARO SEGURIDAD" := NOT "PARO SEGURIDAD";
14     "ED"."Marcha cinta A pt1" := "Marcha cinta A pt1";
15     "ED"."Marcha cinta A pt2" := "Marcha cinta A pt2";
16     "ED"."Marcha cinta B pt1" := "Marcha cinta B pt1";
17     "ED"."Marcha cinta B pt2" := "Marcha cinta B pt2";
18     "ED"."Marcha cinta A pt3" := "Marcha cinta A pt3";
19     "ED"."Marcha cinta A pt4" := "Marcha cinta A pt4";
20     "ED"."Marcha cinta B pt3" := "Marcha cinta B pt3";
21     "ED"."Marcha cinta B pt4" := "Marcha cinta B pt4";
22     "ED"."Mesa 1 FCA" := "Mesa 1 FCA";
23     "ED"."Mesa 1 FC0" := "Mesa 1 FC0";
24     "ED"."Mesa 1 FCB" := "Mesa 1 FCB";
25     "ED"."Mesa 2 FCA" := "Mesa 2 FCA";
26     "ED"."Mesa 2 FC0" := "Mesa 2 FC0";
27     "ED"."Mesa 2 FCB" := "Mesa 2 FCB";
28     "ED"."Mesa 3 FCA" := "Mesa 3 FCA";
29     "ED"."Mesa 3 FC0" := "Mesa 3 FC0";
30     "ED"."Mesa 3 FCB" := "Mesa 3 FCB";
31     "ED"."Mesa 4 FCA" := "Mesa 4 FCA";
32     "ED"."Mesa 4 FC0" := "Mesa 4 FC0";
33     "ED"."Mesa 4 FCB" := "Mesa 4 FCB";
34     "ED"."Mesa 5 FCA" := "Mesa 5 FCA";
35     "ED"."Mesa 5 FC0" := "Mesa 5 FC0";
36     "ED"."Mesa 5 FCB" := "Mesa 5 FCB";
37     "ED"."Mesa 6 FCA" := "Mesa 6 FCA";
38     "ED"."Mesa 6 FC0" := "Mesa 6 FC0";
39     "ED"."Mesa 6 FCB" := "Mesa 6 FCB";
40     "ED"."Marcha cinta mesa 1" := "Marcha cinta mesa 1";
41     "ED"."Marcha cinta mesa 2" := "Marcha cinta mesa 2";
42     "ED"."Marcha cinta mesa 3" := "Marcha cinta mesa 3";
43     "ED"."Marcha cinta mesa 4" := "Marcha cinta mesa 4";
44     "ED"."Marcha cinta mesa 5" := "Marcha cinta mesa 5";
45     "ED"."Marcha cinta mesa 6" := "Marcha cinta mesa 6";
46     "ED"."Marcha nave 1" := "Marcha nave 1";
47     "ED"."Marcha nave 2" := "Marcha nave 2";
48     "ED"."Marcha nave 3" := "Marcha nave 3";
49     "ED"."Marcha nave 4" := "Marcha nave 4";
50     "ED"."Marcha nave 5" := "Marcha nave 5";
51     "ED"."Marcha nave 6" := "Marcha nave 6";
52     "ED"."Marcha nave 7" := "Marcha nave 7";
53     "ED"."Marcha nave 8" := "Marcha nave 8";
54     "ED"."Marcha nave 9" := "Marcha nave 9";

```

```
54 "ED"."Marcha nave 10" := "Marcha nave 10";
55 "ED"."Marcha nave 11" := "Marcha nave 11";
56 "ED"."Marcha nave 12" := "Marcha nave 12";
57 "ED"."Guardamotores cinta A pt1" := "Guardamotores cinta A pt1";
58 "ED"."Guardamotores cinta A pt2" := "Guardamotores cinta A pt2";
59 "ED"."Guardamotores cinta B pt1" := "Guardamotores cinta B pt1";
60 "ED"."Guardamotores cinta B pt2" := "Guardamotores cinta B pt2";
61 "ED"."Guardamotores cinta A pt3" := "Guardamotores cinta A pt3";
62 "ED"."Guardamotores cinta A pt4" := "Guardamotores cinta A pt4";
63 "ED"."Guardamotores cinta B pt3" := "Guardamotores cinta B pt3";
64 "ED"."Guardamotores cinta B pt4" := "Guardamotores cinta B pt4";
65 "ED"."Guardamotores mesa 1" := "Guardamotores mesa 1";
66 "ED"."Guardamotores mesa 2" := "Guardamotores mesa 2";
67 "ED"."Guardamotores mesa 3" := "Guardamotores mesa 3";
68 "ED"."Guardamotores mesa 4" := "Guardamotores mesa 4";
69 "ED"."Guardamotores mesa 5" := "Guardamotores mesa 5";
70 "ED"."Guardamotores mesa 6" := "Guardamotores mesa 6";
71 "ED"."Guardamotores nave 1" := "Guardamotores nave 1";
72 "ED"."Guardamotores nave 2" := "Guardamotores nave 2";
73 "ED"."Guardamotores nave 3" := "Guardamotores nave 3";
74 "ED"."Guardamotores nave 4" := "Guardamotores nave 4";
75 "ED"."Guardamotores nave 5" := "Guardamotores nave 5";
76 "ED"."Guardamotores nave 6" := "Guardamotores nave 6";
77 "ED"."Guardamotores nave 7" := "Guardamotores nave 7";
78 "ED"."Guardamotores nave 8" := "Guardamotores nave 8";
79 "ED"."Guardamotores nave 9" := "Guardamotores nave 9";
80 "ED"."Guardamotores nave 10" := "Guardamotores nave 10";
81 "ED"."Guardamotores nave 11" := "Guardamotores nave 11";
82 "ED"."Guardamotores nave 12" := "Guardamotores nave 12";
```

Código 4: Asignación de entradas

B.4. Ecuaciones de activación [FC6]

En esta función se definen los valores de las variables auxiliares, es decir, las que no son ni entradas, ni salidas, ni etapas. Aunque en la función ponga que el lenguaje es *KOP*, casi todos los segmentos están programados mediante SCL.

B.4.1. Error crítico

Error crítico se tiene que actualizar cada ciclo, tendrá un valor de "1" siempre que alguna de las variables críticas lo sea. Por variables críticas se entiende que son todas las señales de guardamotores, la variable "Variables GEMMA".Fallo_EDM y la entrada de paro de seguridad.

```

1 "Variables GEMMA".ErrorCritico := "ED"."PARO SEGURIDAD" OR "ED"."Guardamotores cinta A
  ↳ pt1" OR "ED"."Guardamotores cinta A pt2" OR "ED"."Guardamotores cinta A pt3" OR
  ↳ "ED"."Guardamotores cinta A pt4" OR "ED"."Guardamotores cinta B pt1" OR
  ↳ "ED"."Guardamotores cinta B pt2" OR "ED"."Guardamotores cinta B pt3" OR
  ↳ "ED"."Guardamotores cinta B pt4" OR
2 "ED"."Guardamotores mesa 1" OR "ED"."Guardamotores mesa 2" OR "ED"."Guardamotores mesa
  ↳ 3" OR "ED"."Guardamotores mesa 4" OR "ED"."Guardamotores mesa 5" OR
  ↳ "ED"."Guardamotores mesa 6" OR
3 "ED"."Guardamotores nave 1" OR "ED"."Guardamotores nave 2" OR "ED"."Guardamotores nave
  ↳ 3" OR "ED"."Guardamotores nave 4" OR "ED"."Guardamotores nave 5" OR
  ↳ "ED"."Guardamotores nave 6" OR "ED"."Guardamotores nave 7" OR "ED"."Guardamotores
  ↳ nave 8" OR "ED"."Guardamotores nave 9" OR "ED"."Guardamotores nave 10" OR
  ↳ "ED"."Guardamotores nave 11" OR "ED"."Guardamotores nave 12"
4 OR "Variables GEMMA".Fallo_EDM;

```

Código 5: Segmento 1: Error crítico

B.4.2. Condiciones iniciales

Para saber si la planta está en condiciones iniciales, se observa la entrada de las fotocélulas. Si todas las fotocélulas están mandando una señal *HIGH*, quiere decir que no están detectando huevos y que por tanto la planta está en condiciones iniciales.

```

1 "Variables GEMMA".CondicionesIniciales := ("ED"."Fotocélula A1" AND
  ↳ "ED"."Fotocélula A2" AND "ED"."Fotocélula A3" AND "ED"."Fotocélula A4" AND
  ↳ "ED"."Fotocélula B1" AND "ED"."Fotocélula B2" AND "ED"."Fotocélula B3" AND
  ↳ "ED"."Fotocélula B4");

```

Código 6: Segmento 2: Condiciones Iniciales

B.4.3. Fin de ciclo reset

Una vez se ha pulsado el botón de "Fin de ciclo" en el HMI, la variable "FinCiclo" se activa. Es necesario que una vez se entre en el modo "Finalizando" esta variable vuelva a desactivarse.

```
1 IF "F3_DB".F30 THEN
2   "Variables GEMMA".FinCiclo := 0;
3 END_IF;
```

Código 7: Segmento 3: FinCiclo reset

B.4.4. Comprobación desorden reset

Una vez se ha pulsado el botón de "Comprobación desorden" en el HMI, la variable "Manual" se activa. Es necesario que una vez se entre en el estado F4 esta variable vuelva a desactivarse.

```
1 IF "F4_DB".F41 THEN
2   "Variables GEMMA".Manual := 0;
3 END_IF;
```

Código 8: Segmento 4: Comprobación desorden reset

B.4.5. Último huevo

Se sabe que ha pasado el último huevo cuando alguno de los temporizadores de "Variables GEMMA".UltimoHuevoPt1 o alguna otra de sus partes se activa.

```
1 "Variables GEMMA".UltimoHuevo := "Variables GEMMA".UltimoHuevoPt1 OR "Variables
↳ GEMMA".UltimoHuevoPt2 OR "Variables GEMMA".UltimoHuevoPt3 OR "Variables
↳ GEMMA".UltimoHuevoPt4;
```

Código 9: Segmento 5: Último huevo

B.4.6. Petición de huevos nueva, disponible y nueva

En este segmento se actualizan las variables Peticion_nueva_disponible_HMI, Peticion-Nueva. La primera está activa siempre que la planta esté en reposo o en marcha; la segunda, siempre que se cumpla lo anterior y además la "PetHuevosAct" tiene que ser diferente a "PetHuevosAnt" y no tiene que ser una PeticionNula. Si todo esto se cumple, además se reinician las variables #DMOKInd y OKAguasAbajo. Si no se cumple, PeticionNueva será 0.

En este segmento también se actualiza el valor de "PetHuevosAnt", que tiene que cambiar cuando se active el estado F2.F27. La variable PeticionNula será "1" si la petición de huevos de las naves en la cinta A y en la cinta B son 0; en caso contrario, PeticionNula será "0".

```

1      (* Si la parte actual es diferente a la anterior el estado es 1, si son iguales el
   ↪ estado es 0*)
2      IF "Variables GEMMA".F1 OR "Variables GEMMA".A1 THEN
3          "Variables GEMMA".Peticion_nueva_disponible_HMI := 1;
4          FOR #i := 1 TO 12 DO
5              IF ("Variables GEMMA"."PetHuevosAct"[#i] <> "Variables
   ↪ GEMMA"."PetHuevosAnt"[#i]) AND NOT "Variables GEMMA".PeticionNula THEN
6                  "Variables GEMMA".PeticionNueva := 1;
7                  #DMOKInd := 0;
8                  "Variables GEMMA".OKAguasAbajo := 0;
9                  EXIT;
10             ELSE
11                 "Variables GEMMA".PeticionNueva := 0;
12             END_IF;
13         END_FOR;
14     ELSE
15         "Variables GEMMA".Peticion_nueva_disponible_HMI := 0;
16     END_IF;
17
18     (*Actualiza #PetHuevosAnt*)
19     IF "F2_DB".F27 THEN
20         FOR #i := 1 TO 12 DO
21             "Variables GEMMA".PetHuevosAnt[#i] := "Variables GEMMA".PetHuevosAct[#i];
22         END_FOR;
23     END_IF;
24
25     IF ("Variables GEMMA".NaveMaxAct_A = 0) AND ("Variables GEMMA".NaveMaxAct_B = 0)
   ↪ THEN
26         "Variables GEMMA".PeticionNula := 1;
27     ELSE
28         "Variables GEMMA".PeticionNula := 0;
29     END_IF;

```

Código 10: Segmento 6: Petición de huevos nueva, disponible y nueva

B.4.7. Petición nueva reset

Tal y como indica el nombre, se realiza el reset de la variable "PeticionNueva", que tiene que realizarse cuando se inicia el modo de preparación, o lo que es lo mismo, en F2.F20.

```

1      IF "F2_DB".F20 THEN
2          "Variables GEMMA".PeticionNueva := 0;
3      END_IF;

```

Código 11: Segmento 7: Peticion nueva reset

B.4.8. Creación de arrays

Para poder agilizar la programación, ha sido necesario crear diferentes arrays. Muchos de ellos están relacionados con las entradas de confirmación de marcha. Otros arrays obtienen los valores del HMI de la petición de huevos y lo transforma en la variable PeticionHuevosNave, y tras juntarlo con la variable de HMI "Petición huevos cinta A", se obtiene el array "PetHuevosAct".

```
1 IF "ED"."Mesa 1 FC0" AND NOT "ED"."Mesa 1 FCA" AND NOT "ED"."Mesa 1 FCB" THEN
2     "Variables GEMMA".ConfDM[1] := '0';
3 ELSIF NOT "ED"."Mesa 1 FC0" AND "ED"."Mesa 1 FCA" AND NOT "ED"."Mesa 1 FCB" THEN
4     "Variables GEMMA".ConfDM[1] := 'A';
5 ELSIF NOT "ED"."Mesa 1 FC0" AND NOT "ED"."Mesa 1 FCA" AND "ED"."Mesa 1 FCB" THEN
6     "Variables GEMMA".ConfDM[1] := 'B';
7 ELSE
8     "Variables GEMMA".ConfDM[1] := 'X';
9 END_IF;
10
11 IF "ED"."Mesa 2 FC0" AND NOT "ED"."Mesa 2 FCA" AND NOT "ED"."Mesa 2 FCB" THEN
12     "Variables GEMMA".ConfDM[2] := '0';
13 ELSIF NOT "ED"."Mesa 2 FC0" AND "ED"."Mesa 2 FCA" AND NOT "ED"."Mesa 2 FCB" THEN
14     "Variables GEMMA".ConfDM[2] := 'A';
15 ELSIF NOT "ED"."Mesa 2 FC0" AND NOT "ED"."Mesa 2 FCA" AND "ED"."Mesa 2 FCB" THEN
16     "Variables GEMMA".ConfDM[2] := 'B';
17 ELSE
18     "Variables GEMMA".ConfDM[2] := 'X';
19 END_IF;
20
21 IF "ED"."Mesa 3 FC0" AND NOT "ED"."Mesa 3 FCA" AND NOT "ED"."Mesa 3 FCB" THEN
22     "Variables GEMMA".ConfDM[3] := '0';
23 ELSIF NOT "ED"."Mesa 3 FC0" AND "ED"."Mesa 3 FCA" AND NOT "ED"."Mesa 3 FCB" THEN
24     "Variables GEMMA".ConfDM[3] := 'A';
25 ELSIF NOT "ED"."Mesa 3 FC0" AND NOT "ED"."Mesa 3 FCA" AND "ED"."Mesa 3 FCB" THEN
26     "Variables GEMMA".ConfDM[3] := 'B';
27 ELSE
28     "Variables GEMMA".ConfDM[3] := 'X';
29 END_IF;
30
31 IF "ED"."Mesa 4 FC0" AND NOT "ED"."Mesa 4 FCA" AND NOT "ED"."Mesa 4 FCB" THEN
32     "Variables GEMMA".ConfDM[4] := '0';
33 ELSIF NOT "ED"."Mesa 4 FC0" AND "ED"."Mesa 4 FCA" AND NOT "ED"."Mesa 4 FCB" THEN
34     "Variables GEMMA".ConfDM[4] := 'A';
35 ELSIF NOT "ED"."Mesa 4 FC0" AND NOT "ED"."Mesa 4 FCA" AND "ED"."Mesa 4 FCB" THEN
36     "Variables GEMMA".ConfDM[4] := 'B';
37 ELSE
38     "Variables GEMMA".ConfDM[4] := 'X';
39 END_IF;
40
41 IF "ED"."Mesa 5 FC0" AND NOT "ED"."Mesa 5 FCA" AND NOT "ED"."Mesa 5 FCB" THEN
42     "Variables GEMMA".ConfDM[5] := '0';
43 ELSIF NOT "ED"."Mesa 5 FC0" AND "ED"."Mesa 5 FCA" AND NOT "ED"."Mesa 5 FCB" THEN
44     "Variables GEMMA".ConfDM[5] := 'A';
45 ELSIF NOT "ED"."Mesa 5 FC0" AND NOT "ED"."Mesa 5 FCA" AND "ED"."Mesa 5 FCB" THEN
46     "Variables GEMMA".ConfDM[5] := 'B';
47 ELSE
48     "Variables GEMMA".ConfDM[5] := 'X';
49 END_IF;
50
51 IF "ED"."Mesa 6 FC0" AND NOT "ED"."Mesa 6 FCA" AND NOT "ED"."Mesa 6 FCB" THEN
52     "Variables GEMMA".ConfDM[6] := '0';
53 ELSIF NOT "ED"."Mesa 6 FC0" AND "ED"."Mesa 6 FCA" AND NOT "ED"."Mesa 6 FCB" THEN
54     "Variables GEMMA".ConfDM[6] := 'A';
55 ELSIF NOT "ED"."Mesa 6 FC0" AND NOT "ED"."Mesa 6 FCA" AND "ED"."Mesa 6 FCB" THEN
56     "Variables GEMMA".ConfDM[6] := 'B';
57 ELSE
58     "Variables GEMMA".ConfDM[6] := 'X';
```

```

59 END_IF;
60
61 (*Array Confirmación de partes de cinta*)
62 "Variables GEMMA"."ConfCGA"[1] := "ED"."Marcha cinta A pt1";
63 "Variables GEMMA"."ConfCGA"[2] := "ED"."Marcha cinta A pt2";
64 "Variables GEMMA"."ConfCGA"[3] := "ED"."Marcha cinta A pt3";
65 "Variables GEMMA"."ConfCGA"[4] := "ED"."Marcha cinta A pt4";
66
67 "Variables GEMMA"."ConfCGB"[1] := "ED"."Marcha cinta B pt1";
68 "Variables GEMMA"."ConfCGB"[2] := "ED"."Marcha cinta B pt2";
69 "Variables GEMMA"."ConfCGB"[3] := "ED"."Marcha cinta B pt3";
70 "Variables GEMMA"."ConfCGB"[4] := "ED"."Marcha cinta B pt4";
71
72 (*Array Confirmación cinta de las mesas*)
73 "Variables GEMMA"."ConfCM"[1] := "ED"."Marcha cinta mesa 1";
74 "Variables GEMMA"."ConfCM"[2] := "ED"."Marcha cinta mesa 2";
75 "Variables GEMMA"."ConfCM"[3] := "ED"."Marcha cinta mesa 3";
76 "Variables GEMMA"."ConfCM"[4] := "ED"."Marcha cinta mesa 4";
77 "Variables GEMMA"."ConfCM"[5] := "ED"."Marcha cinta mesa 5";
78 "Variables GEMMA"."ConfCM"[6] := "ED"."Marcha cinta mesa 6";
79
80 (*Array Confirmación cinta de las naves*)
81 "Variables GEMMA".ConfCN[1] := "ED"."Marcha nave 1";
82 "Variables GEMMA".ConfCN[2] := "ED"."Marcha nave 2";
83 "Variables GEMMA".ConfCN[3] := "ED"."Marcha nave 3";
84 "Variables GEMMA".ConfCN[4] := "ED"."Marcha nave 4";
85 "Variables GEMMA".ConfCN[5] := "ED"."Marcha nave 5";
86 "Variables GEMMA".ConfCN[6] := "ED"."Marcha nave 6";
87 "Variables GEMMA".ConfCN[7] := "ED"."Marcha nave 7";
88 "Variables GEMMA".ConfCN[8] := "ED"."Marcha nave 8";
89 "Variables GEMMA".ConfCN[9] := "ED"."Marcha nave 9";
90 "Variables GEMMA".ConfCN[10] := "ED"."Marcha nave 10";
91 "Variables GEMMA".ConfCN[11] := "ED"."Marcha nave 11";
92 "Variables GEMMA".ConfCN[12] := "ED"."Marcha nave 12";
93
94
95
96 (*Array "Variables GEMMA"."PetHuevosNave para saber de forma dinámica en qué nave se
97 ↪ solicitan huevos*)
98 IF "Variables GEMMA".A1 OR "Variables GEMMA".F1 THEN // solo actualizamos en A1 o F1
99 "Variables GEMMA".PeticionHuevosNave[1] := "Variables_HMI"."Petición huevos nave
100 ↪ 1";
101 "Variables GEMMA".PeticionHuevosNave[2] := "Variables_HMI"."Petición huevos nave
102 ↪ 2";
103 "Variables GEMMA".PeticionHuevosNave[3] := "Variables_HMI"."Petición huevos nave
104 ↪ 3";
105 "Variables GEMMA".PeticionHuevosNave[4] := "Variables_HMI"."Petición huevos nave
106 ↪ 4";
107 "Variables GEMMA".PeticionHuevosNave[5] := "Variables_HMI"."Petición huevos nave
108 ↪ 5";
109 "Variables GEMMA".PeticionHuevosNave[6] := "Variables_HMI"."Petición huevos nave
110 ↪ 6";
111 "Variables GEMMA".PeticionHuevosNave[7] := "Variables_HMI"."Petición huevos nave
112 ↪ 7";
113 "Variables GEMMA".PeticionHuevosNave[8] := "Variables_HMI"."Petición huevos nave
114 ↪ 8";
115 "Variables GEMMA".PeticionHuevosNave[9] := "Variables_HMI"."Petición huevos nave
116 ↪ 9";

```

```

107     "Variables GEMMA".PeticionHuevosNave[10] := "Variables_HMI"."Petición huevos nave
↪ 10";
108     "Variables GEMMA".PeticionHuevosNave[11] := "Variables_HMI"."Petición huevos nave
↪ 11";
109     "Variables GEMMA".PeticionHuevosNave[12] := "Variables_HMI"."Petición huevos nave
↪ 12";
110 END_IF;
111
112 (*Array "Variables GEMMA"."PetHuevosAct para saber en qué posición se desea que esté el
↪ actuador de cada mesa*)
113 FOR #i := 1 TO 12 DO
114
115     IF "Variables GEMMA".PeticionHuevosNave[#i] THEN
116
117         IF "Variables_HMI"."Petición huevos cinta A" AND NOT "Variables_HMI"."Petición
↪ huevos cinta B" THEN
118             (*Sólo huevos por la cinta A*)
119             "Variables GEMMA"."PetHuevosAct"[#i] := 'A';
120         ELSIF "Variables_HMI"."Petición huevos cinta B" AND NOT
↪ "Variables_HMI"."Petición huevos cinta A" THEN
121             (*Sólo huevos por la cinta B*)
122             "Variables GEMMA"."PetHuevosAct"[#i] := 'B';
123         ELSIF "Variables_HMI"."Petición huevos cinta A" AND "Variables_HMI"."Petición
↪ huevos cinta B" THEN
124
125             IF (#i <= 6) THEN
126                 "Variables GEMMA"."PetHuevosAct"[#i] := 'A';
127             ELSE
128                 "Variables GEMMA"."PetHuevosAct"[#i] := 'B';
129             END_IF;
130
131         ELSE
132             (*No se activa ninguna cinta*)
133             "Variables GEMMA"."PetHuevosAct"[#i] := '0';
134         END_IF;
135     ELSE
136         (*No se activa ninguna cinta*)
137         "Variables GEMMA"."PetHuevosAct"[#i] := '0';
138     END_IF;
139
140 END_FOR;
141

```

Código 12: Segmento 8: Creación arrays

B.4.9. OK

En este segmento se comparan los valores deseados de cada elemento con su respectiva entrada de confirmación, si es todo correcto "OK" es 1.

```

1 REGION OK_CG
2   FOR #i := 1 TO 4 DO
3     "Variables GEMMA".CG_OK[#i] := ((NOT "Variables GEMMA".ParteAct_A[#i] AND NOT
    ↪ "Variables GEMMA".ConfCGA[#i]) AND (NOT "Variables GEMMA".ParteAct_B[#i]
    ↪ AND NOT "Variables GEMMA".ConfCGB[#i])) OR ((NOT "Variables
    ↪ GEMMA".ParteAct_A[#i] AND NOT "Variables GEMMA".ConfCGA[#i]) AND
    ↪ ("Variables GEMMA".ParteAct_B[#i] AND "Variables GEMMA".ConfCGB[#i])) OR
    ↪ (("Variables GEMMA".ParteAct_A[#i] AND "Variables GEMMA".ConfCGA[#i]) AND
    ↪ (NOT "Variables GEMMA".ParteAct_B[#i] AND NOT "Variables
    ↪ GEMMA".ConfCGB[#i])) OR (("Variables GEMMA".ParteAct_A[#i] AND "Variables
    ↪ GEMMA".ConfCGA[#i]) AND ("Variables GEMMA".ParteAct_B[#i] AND "Variables
    ↪ GEMMA".ConfCGB[#i]));
4   END_FOR;
5 END_REGION
6
7
8 REGION Dirección mesas OK
9   IF "Variables GEMMA".F2 OR "Variables GEMMA".F5 THEN
10    #DMOKInd := 0;
11    FOR #i := 1 TO 6 DO
12      IF "Variables GEMMA".PetHuevosAct[2 * #i - 1] = "Variables
    ↪ GEMMA".PetHuevosAct[2 * #i] THEN
13        "Variables GEMMA".DM_Deseada[#i] := "Variables GEMMA".PetHuevosAct[2 *
    ↪ #i];
14        IF "Variables GEMMA".PetHuevosAct[2 * #i - 1] = "Variables
    ↪ GEMMA".ConfDM[#i] THEN
15          #DMOKInd := #DMOKInd + 1;
16        END_IF;
17
18      ELSE
19        IF "Variables GEMMA".PetHuevosAct[2 * #i - 1] = '0' THEN
20          "Variables GEMMA".DM_Deseada[#i] := "Variables
    ↪ GEMMA".PetHuevosAct[2 * #i];
21          IF "Variables GEMMA".PetHuevosAct[2 * #i] = "Variables
    ↪ GEMMA".ConfDM[#i] THEN
22            #DMOKInd := #DMOKInd + 1;
23          END_IF;
24
25          ELSIF "Variables GEMMA".PetHuevosAct[2 * #i] = '0' THEN
26            "Variables GEMMA".DM_Deseada[#i] := "Variables
    ↪ GEMMA".PetHuevosAct[2 * #i - 1];
27            IF "Variables GEMMA".PetHuevosAct[2 * #i - 1] = "Variables
    ↪ GEMMA".ConfDM[#i] THEN
28              #DMOKInd := #DMOKInd + 1;
29            END_IF;
30
31          END_IF;
32
33        END_IF;
34
35      END_FOR;
36    END_IF;
37

```

```
38     "Variables GEMMA".DM_OK := (#DMOKInd = 6);
39
40
41 END_REGION
42
43
44 REGION CINTAS MESAS OK
45     IF "Variables GEMMA".F2 OR "Variables GEMMA".F5 THEN
46         #CMOKInd := 0;
47         FOR #i := 1 TO 6 DO
48             IF ("Variables GEMMA".DM_Deseada[#i] <> '0') THEN
49                 IF "Variables GEMMA".ConfCM[#i] THEN
50                     #CMOKInd := #CMOKInd + 1;
51                 END_IF;
52             ELSE
53                 IF NOT "Variables GEMMA".ConfCM[#i] THEN
54                     #CMOKInd := #CMOKInd + 1;
55                 END_IF;
56             END_IF;
57         END_FOR;
58     END_IF;
59
60     "Variables GEMMA".CM_OK := (#CMOKInd = 6);
61
62 END_REGION
63
64 REGION CN_OK
65
66     IF "Variables GEMMA".F1 OR "Variables GEMMA".F5 THEN
67         #CNOKInd := 0;
68         FOR #i := 1 TO 12 DO
69             IF "Variables GEMMA".PeticiónHuevosNave[#i] THEN // Statement section FOR
70                 IF "Variables GEMMA".ConfCN[#i] THEN
71                     #CNOKInd := #CNOKInd + 1;
72                 END_IF;
73             ELSE
74                 IF NOT "Variables GEMMA".ConfCN[#i] THEN
75                     #CNOKInd := #CNOKInd + 1;
76                 END_IF;
77             END_IF;
78         END_FOR;
79
80     "Variables GEMMA".CN_OK := #CNOKInd = 12;
81 END_IF;
82
83 END_REGION
84
85
86
87
```

Código 13: Segmento 9: CG_OK, DM_OK, CM_OK, CN_OK

B.4.10. Control Partes cintas

En este segmento, utilizando la información de "PetHuevosAct", se calcula la nave "más alta" de la que se piden huevos. Después, sabiendo la distribución de las naves, es posible saber qué parte de cada cinta general es necesario que se active para llevar los huevos.

```

1  (*Buscamos la nave "más alta" de la que se piden huevos*)
2  "Variables GEMMA".NaveMaxAct_A := 0;
3  "Variables GEMMA".NaveMaxAct_B := 0;
4      FOR #i := 1 TO 12 DO
5
6          IF "Variables GEMMA".PetHuevosAct[#i] = 'A' THEN
7              "Variables GEMMA".NaveMaxAct_A := #i;
8          END_IF;
9          IF "Variables GEMMA".PetHuevosAct[#i] = 'B' THEN
10             "Variables GEMMA".NaveMaxAct_B := #i;
11         END_IF;
12
13     END_FOR;
14
15     (*Se busca saber qué partes tienen que estar activas ahora, por bloque de cintas*)
16     (*Cinta A*)
17     IF "Variables GEMMA".NaveMaxAct_A > 0 THEN
18         (*Si hay alguna petición de nave, activamos parte 1*)
19         "Variables GEMMA".ParteAct_A[1] := 1;
20
21     IF "Variables GEMMA".NaveMaxAct_A > 2 THEN
22         (*La parte 2 se activa cuando hay petición de nave mayor que la 2*)
23         "Variables GEMMA".ParteAct_A[2] := 1;
24
25     IF "Variables GEMMA".NaveMaxAct_A > 4 THEN
26         (*Si hay alguna petición superior a la nave 4, se activa la parte 3*)
27         "Variables GEMMA".ParteAct_A[3] := 1;
28
29         IF "Variables GEMMA".NaveMaxAct_A > 8 THEN
30             (*La parte 4 se activa cuando hay petición de nave mayor que la 8*)
31             "Variables GEMMA".ParteAct_A[4] := 1;
32         ELSE
33             "Variables GEMMA".ParteAct_A[4] := 0;
34
35         END_IF;
36
37     ELSE
38         "Variables GEMMA".ParteAct_A[4] := 0;
39         "Variables GEMMA".ParteAct_A[3] := 0;
40
41     END_IF;
42
43     ELSE
44         "Variables GEMMA".ParteAct_A[4] := 0;
45         "Variables GEMMA".ParteAct_A[3] := 0;
46         "Variables GEMMA".ParteAct_A[2] := 0;
47
48     END_IF;
49
50     ELSE
51         "Variables GEMMA".ParteAct_A[4] := 0;
52         "Variables GEMMA".ParteAct_A[3] := 0;

```

```
53     "Variables GEMMA".ParteAct_A[2] := 0;
54     "Variables GEMMA".ParteAct_A[1] := 0;
55     END_IF;
56
57
58     (*Cinta B*)
59
60     IF "Variables GEMMA".NaveMaxAct_B > 0 THEN
61         (*Si hay alguna petición de nave, activamos parte 1*)
62         "Variables GEMMA".ParteAct_B[1] := 1;
63
64         IF "Variables GEMMA".NaveMaxAct_B > 2 THEN
65             (*La parte 2 se activa cuando hay petición de nave mayor que la 2*)
66             "Variables GEMMA".ParteAct_B[2] := 1;
67
68             IF "Variables GEMMA".NaveMaxAct_B > 4 THEN
69                 (*Si hay alguna petición superior a la nave 4, se activa la parte 3*)
70                 "Variables GEMMA".ParteAct_B[3] := 1;
71
72                 IF "Variables GEMMA".NaveMaxAct_B > 8 THEN
73                     (*La parte 4 se activa cuando hay petición de nave mayor que la 8*)
74                     "Variables GEMMA".ParteAct_B[4] := 1;
75                 ELSE
76                     "Variables GEMMA".ParteAct_B[4] := 0;
77
78                 END_IF;
79
80             ELSE
81                 "Variables GEMMA".ParteAct_B[4] := 0;
82                 "Variables GEMMA".ParteAct_B[3] := 0;
83
84             END_IF;
85
86         ELSE
87             "Variables GEMMA".ParteAct_B[4] := 0;
88             "Variables GEMMA".ParteAct_B[3] := 0;
89             "Variables GEMMA".ParteAct_B[2] := 0;
90         END_IF;
91
92     ELSE
93         "Variables GEMMA".ParteAct_B[4] := 0;
94         "Variables GEMMA".ParteAct_B[3] := 0;
95         "Variables GEMMA".ParteAct_B[2] := 0;
96         "Variables GEMMA".ParteAct_B[1] := 0;
97
98     END_IF;
```

Código 14: Segmento 10: Control Partes cintas

B.4.11. Inicializaciones cuando A1

Como su nombre indica aquí se agrupan las inicializaciones que se tienen que realizar cuando el estado de la planta es A1.

```

1 IF "Variables GEMMA".A1 THEN
2   "Variables GEMMA".NaveMaxAct_A := 0;
3   "Variables GEMMA".NaveMaxAct_B := 0;
4 END_IF;
```

Código 15: Segmento 11: Inicializaciones cuando A1

B.4.12. OK Aguas Abajo

Cuando la planta está en los estados F1, F2 o F5, es necesario comprobar que aguas abajo de las cintas de las naves todo lo que debería estar activado lo está.

```

1 IF "Variables GEMMA".F2 OR "Variables GEMMA".F1 OR "Variables GEMMA".F5 THEN
2   "Variables GEMMA".OKAguasAbajo := (#AguasAbajoOkInd = #ParteMax) AND "Variables
   ↪ GEMMA".DM_OK AND ("Variables GEMMA".F1 OR "F2_DB".F27);
3   #AguasAbajoOkInd := 0;
4   FOR #i := 1 TO 4 DO
5     IF "Variables GEMMA".ParteAct_A[#i] OR "Variables GEMMA".ParteAct_B[#i] THEN
6       #ParteMax := #i;
7       IF ("Variables GEMMA".ParteAct_A[#i] AND "Variables GEMMA".ConfCGA[#i]) OR
   ↪ ("Variables GEMMA".ParteAct_B[#i] AND "Variables GEMMA".ConfCGB[#i])
   ↪ THEN
8         #AguasAbajoOkInd := #AguasAbajoOkInd + 1;
9         END_IF;
10      END_IF;
11    END_FOR;
12  END_IF;
```

Código 16: Segmento 12: OK Aguas Abajo

B.4.13. Check todo parado / todo en marcha

Para pasar entre los estados de la macro etapa de F4 es necesario saber cuándo está todo activo y todo parado. Todo parado también se utiliza para volver a A1 por F3 y F5.

```

1 "Variables GEMMA".TodoParado := NOT ("ED"."Marcha cinta A pt1" OR "ED"."Marcha cinta A
   ↪ pt2" OR "ED"."Marcha cinta A pt3" OR "ED"."Marcha cinta A pt4" OR "ED"."Marcha
   ↪ cinta B pt1" OR "ED"."Marcha cinta B pt2" OR "ED"."Marcha cinta B pt3" OR
   ↪ "ED"."Marcha cinta B pt4"
2 OR "ED"."Marcha cinta mesa 1" OR "ED"."Marcha cinta mesa 2" OR "ED"."Marcha cinta mesa
   ↪ 3" OR "ED"."Marcha cinta mesa 4" OR "ED"."Marcha cinta mesa 5" OR "ED"."Marcha
   ↪ cinta mesa 6"
3 OR "ED"."Mesa 1 FCA" OR "ED"."Mesa 1 FCB" OR "ED"."Mesa 2 FCA" OR "ED"."Mesa 2 FCB" OR
   ↪ "ED"."Mesa 3 FCA" OR "ED"."Mesa 3 FCB" OR "ED"."Mesa 4 FCA" OR "ED"."Mesa 4 FCB" OR
   ↪ "ED"."Mesa 5 FCA" OR "ED"."Mesa 5 FCB" OR "ED"."Mesa 6 FCA" OR "ED"."Mesa 6 FCB"
4 OR "ED"."Marcha nave 1" OR "ED"."Marcha nave 2" OR "ED"."Marcha nave 3" OR "ED"."Marcha
   ↪ nave 4" OR "ED"."Marcha nave 5" OR "ED"."Marcha nave 6" OR "ED"."Marcha nave 7" OR
   ↪ "ED"."Marcha nave 8" OR "ED"."Marcha nave 9" OR "ED"."Marcha nave 10" OR
   ↪ "ED"."Marcha nave 11" OR "ED"."Marcha nave 12");
```

```

5 "Variables GEMMA".TodoEnMarcha := ("ED"."Marcha cinta A pt1" AND "ED"."Marcha cinta A
  ↳ pt2" AND "ED"."Marcha cinta A pt3" AND "ED"."Marcha cinta A pt4" AND "ED"."Marcha
  ↳ cinta B pt1" AND "ED"."Marcha cinta B pt2" AND "ED"."Marcha cinta B pt3" AND
  ↳ "ED"."Marcha cinta B pt4"
6 AND "ED"."Marcha cinta mesa 1" AND "ED"."Marcha cinta mesa 2" AND "ED"."Marcha cinta
  ↳ mesa 3" AND "ED"."Marcha cinta mesa 4" AND "ED"."Marcha cinta mesa 5" AND
  ↳ "ED"."Marcha cinta mesa 6"
7 AND "ED"."Mesa 1 FCB" AND "ED"."Mesa 2 FCB" AND "ED"."Mesa 3 FCB" AND "ED"."Mesa 4 FCB"
  ↳ AND "ED"."Mesa 5 FCB" AND "ED"."Mesa 6 FCB"
8 AND "ED"."Marcha nave 1" AND "ED"."Marcha nave 2" AND "ED"."Marcha nave 3" AND
  ↳ "ED"."Marcha nave 4" AND "ED"."Marcha nave 5" AND "ED"."Marcha nave 6" AND
  ↳ "ED"."Marcha nave 7" AND "ED"."Marcha nave 8" AND "ED"."Marcha nave 9" AND
  ↳ "ED"."Marcha nave 10" AND "ED"."Marcha nave 11" AND "ED"."Marcha nave 12");

```

Código 17: Segmento 13: Check todo parado / todo en marcha

B.4.14. F5 continuar

En la pantalla se encuentra el botón "CONTINUAR" que sólo está activo cuando la planta está en F5. Cuando se pulsa, se necesita activar una variable que sólo actúe durante un ciclo. Esto se obtiene detectando el flanco de subida de pulsar el botón (con la variable que se activa al pulsarlo), y desactivando la variable con su activación.



Código 18: Segmento 14: F5 continuar

B.4.15. Etapa actual

Para poder mostrar en el HMI la etapa actual con una lista de textos, es necesario pasar los estados del GRAFCET superior a una variable de tipo entero.

```

1 IF "Variables GEMMA".A1 THEN "Variables_HMI".EstadoAct_HMI := 0;
2 ELSIF "Variables GEMMA".F2 THEN "Variables_HMI".EstadoAct_HMI := 1;
3 ELSIF "Variables GEMMA".F1 THEN "Variables_HMI".EstadoAct_HMI := 2;
4 ELSIF "Variables GEMMA".F3 THEN "Variables_HMI".EstadoAct_HMI := 3;
5 ELSIF "Variables GEMMA".F5 THEN "Variables_HMI".EstadoAct_HMI := 4;
6 ELSIF "Variables GEMMA".F4 THEN "Variables_HMI".EstadoAct_HMI := 5;
7 ELSIF "Variables GEMMA".D1 THEN "Variables_HMI".EstadoAct_HMI := 6;
8 ELSIF "Variables GEMMA".A6 THEN "Variables_HMI".EstadoAct_HMI := 7;
9 ELSE "Variables_HMI".EstadoAct_HMI := 8;
10 END_IF;

```

Código 19: Segmento 15: Etapa actual

B.4.16. Estados (marcha/paro/error) HMI

Al igual que en el segmento anterior, es necesario "traducir" a una variable de tipo entero los diferentes estados de los elementos, para poder utilizar las animaciones del HMI.

```

1  REGION Estado_CGA
2  IF "ED"."Guardamotores cinta A pt1" OR "Variables GEMMA".Fallo_EDM_CGA[1] THEN
3      "Variables_HMI".Estado_CGA_HMI[1] := 2;
4  ELSE
5      "Variables_HMI".Estado_CGA_HMI[1] := BOOL_TO_INT("CGA1");
6  END_IF;
7
8  IF "ED"."Guardamotores cinta A pt2" OR "Variables GEMMA".Fallo_EDM_CGA[2] THEN
9      "Variables_HMI".Estado_CGA_HMI[2] := 2;
10 ELSE
11     "Variables_HMI".Estado_CGA_HMI[2] := BOOL_TO_INT("CGA2");
12 END_IF;
13
14 IF "ED"."Guardamotores cinta A pt3" OR "Variables GEMMA".Fallo_EDM_CGA[3] THEN
15     "Variables_HMI".Estado_CGA_HMI[3] := 2;
16 ELSE
17     "Variables_HMI".Estado_CGA_HMI[3] := BOOL_TO_INT("CGA3");
18 END_IF;
19
20 IF "ED"."Guardamotores cinta A pt4" OR "Variables GEMMA".Fallo_EDM_CGA[4] THEN
21     "Variables_HMI".Estado_CGA_HMI[4] := 2;
22 ELSE
23     "Variables_HMI".Estado_CGA_HMI[4] := BOOL_TO_INT("CGA4");
24 END_IF;
25 END_REGION
26
27 REGION Estado_CGB
28     IF "ED"."Guardamotores cinta B pt1" OR "Variables GEMMA".Fallo_EDM_CGB[1] THEN
29         "Variables_HMI".Estado_CGB_HMI[1] := 2;
30 ELSE
31     "Variables_HMI".Estado_CGB_HMI[1] := BOOL_TO_INT("CGB1");
32 END_IF;
33
34 IF "ED"."Guardamotores cinta B pt2" OR "Variables GEMMA".Fallo_EDM_CGB[2] THEN
35     "Variables_HMI".Estado_CGB_HMI[2] := 2;
36 ELSE
37     "Variables_HMI".Estado_CGB_HMI[2] := BOOL_TO_INT("CGB2");
38 END_IF;
39
40 IF "ED"."Guardamotores cinta B pt3" OR "Variables GEMMA".Fallo_EDM_CGB[3] THEN
41     "Variables_HMI".Estado_CGB_HMI[3] := 2;
42 ELSE
43     "Variables_HMI".Estado_CGB_HMI[3] := BOOL_TO_INT("CGB3");
44 END_IF;
45
46 IF "ED"."Guardamotores cinta B pt4" OR "Variables GEMMA".Fallo_EDM_CGB[4] THEN
47     "Variables_HMI".Estado_CGB_HMI[4] := 2;
48 ELSE
49     "Variables_HMI".Estado_CGB_HMI[4] := BOOL_TO_INT("CGB4");
50 END_IF;
51 END_REGION
52
53 REGION DireccionMesa

```

```

54 IF "ED"."Guardamotores mesa 1" OR "Variables GEMMA".Fallo_EDM_DM[1] OR "Variables
↪ GEMMA".Fallo_EDM_CM[1] THEN
55     "Variables_HMI".Estado_M_HMI[1] := 3;
56 ELSE
57     IF "M1 AVANZAR" AND NOT "M1 RETROCEDER" THEN
58         "Variables_HMI".Estado_M_HMI[1] := 1;
59     ELSIF NOT "M1 AVANZAR" AND "M1 RETROCEDER" THEN
60         "Variables_HMI".Estado_M_HMI[1] := 2;
61     ELSIF "M1 AVANZAR" AND "M1 RETROCEDER" THEN
62         "Variables_HMI".Estado_M_HMI[1] := 3;
63     ELSE
64         "Variables_HMI".Estado_M_HMI[1] := 0;
65     END_IF;
66 END_IF;
67
68 IF "ED"."Guardamotores mesa 2" OR "Variables GEMMA".Fallo_EDM_DM[2] OR "Variables
↪ GEMMA".Fallo_EDM_CM[2] THEN
69     "Variables_HMI".Estado_M_HMI[2] := 3;
70 ELSE
71
72     IF "M2 AVANZAR" AND NOT "M2 RETROCEDER" THEN
73         "Variables_HMI".Estado_M_HMI[2] := 1;
74     ELSIF NOT "M2 AVANZAR" AND "M2 RETROCEDER" THEN
75         "Variables_HMI".Estado_M_HMI[2] := 2;
76     ELSIF "M2 AVANZAR" AND "M2 RETROCEDER" THEN
77         "Variables_HMI".Estado_M_HMI[2] := 3;
78     ELSE
79         "Variables_HMI".Estado_M_HMI[2] := 0;
80     END_IF;
81 END_IF;
82
83 IF "ED"."Guardamotores mesa 3" OR "Variables GEMMA".Fallo_EDM_DM[3] OR "Variables
↪ GEMMA".Fallo_EDM_CM[3] THEN
84     "Variables_HMI".Estado_M_HMI[3] := 3;
85 ELSE
86
87     IF "M3 AVANZAR" AND NOT "M3 RETROCEDER" THEN
88         "Variables_HMI".Estado_M_HMI[3] := 1;
89     ELSIF NOT "M3 AVANZAR" AND "M3 RETROCEDER" THEN
90         "Variables_HMI".Estado_M_HMI[3] := 2;
91     ELSIF "M3 AVANZAR" AND "M3 RETROCEDER" THEN
92         "Variables_HMI".Estado_M_HMI[3] := 3;
93     ELSE
94         "Variables_HMI".Estado_M_HMI[3] := 0;
95     END_IF;
96 END_IF;
97
98 IF "ED"."Guardamotores mesa 4" OR "Variables GEMMA".Fallo_EDM_DM[4] OR "Variables
↪ GEMMA".Fallo_EDM_CM[4] THEN
99     "Variables_HMI".Estado_M_HMI[4] := 3;
100 ELSE
101
102     IF "M4 AVANZAR" AND NOT "M4 RETROCEDER" THEN
103         "Variables_HMI".Estado_M_HMI[4] := 1;
104     ELSIF NOT "M4 AVANZAR" AND "M4 RETROCEDER" THEN
105         "Variables_HMI".Estado_M_HMI[4] := 2;
106     ELSIF "M4 AVANZAR" AND "M4 RETROCEDER" THEN
107         "Variables_HMI".Estado_M_HMI[4] := 3;

```

```

108     ELSE
109         "Variables_HMI".Estado_M_HMI[4] := 0;
110     END_IF;
111 END_IF;
112
113 IF "ED"."Guardamotores mesa 5" OR "Variables GEMMA".Fallo_EDM_DM[5] OR "Variables
↪ GEMMA".Fallo_EDM_CM[5] THEN
114     "Variables_HMI".Estado_M_HMI[5] := 3;
115 ELSE
116     IF "M5 AVANZAR" AND NOT "M5 RETROCEDER" THEN
117         "Variables_HMI".Estado_M_HMI[5] := 1;
118     ELSIF NOT "M5 AVANZAR" AND "M5 RETROCEDER" THEN
119         "Variables_HMI".Estado_M_HMI[5] := 2;
120     ELSIF "M5 AVANZAR" AND "M5 RETROCEDER" THEN
121         "Variables_HMI".Estado_M_HMI[5] := 3;
122     ELSE
123         "Variables_HMI".Estado_M_HMI[5] := 0;
124     END_IF;
125 END_IF;
126
127 IF "ED"."Guardamotores mesa 6" OR "Variables GEMMA".Fallo_EDM_DM[6] OR "Variables
↪ GEMMA".Fallo_EDM_CM[6] THEN
128     "Variables_HMI".Estado_M_HMI[6] := 3;
129 ELSE
130     IF "M6 AVANZAR" AND NOT "M6 RETROCEDER" THEN
131         "Variables_HMI".Estado_M_HMI[6] := 1;
132     ELSIF NOT "M6 AVANZAR" AND "M6 RETROCEDER" THEN
133         "Variables_HMI".Estado_M_HMI[6] := 2;
134     ELSIF "M6 AVANZAR" AND "M6 RETROCEDER" THEN
135         "Variables_HMI".Estado_M_HMI[6] := 3;
136     ELSE
137         "Variables_HMI".Estado_M_HMI[6] := 0;
138     END_IF;
139 END_IF;
140 END_REGION
141
142 REGION Estado_naves
143     IF "ED"."Guardamotores nave 1" OR "Variables GEMMA".Fallo_EDM_CN[1] THEN
144         "Variables_HMI".Estado_CN_HMI[1] := 2;
145     ELSE
146         "Variables_HMI".Estado_CN_HMI[1] := BOOL_TO_INT("CN1");
147     END_IF;
148
149     IF "ED"."Guardamotores nave 2" THEN
150         "Variables_HMI".Estado_CN_HMI[2] := 2;
151     ELSE
152         "Variables_HMI".Estado_CN_HMI[2] := BOOL_TO_INT("CN2");
153     END_IF;
154
155     IF "ED"."Guardamotores nave 3" THEN
156         "Variables_HMI".Estado_CN_HMI[3] := 2;
157     ELSE
158         "Variables_HMI".Estado_CN_HMI[3] := BOOL_TO_INT("CN3");
159     END_IF;
160
161     IF "ED"."Guardamotores nave 4" THEN
162         "Variables_HMI".Estado_CN_HMI[4] := 2;
163     ELSE

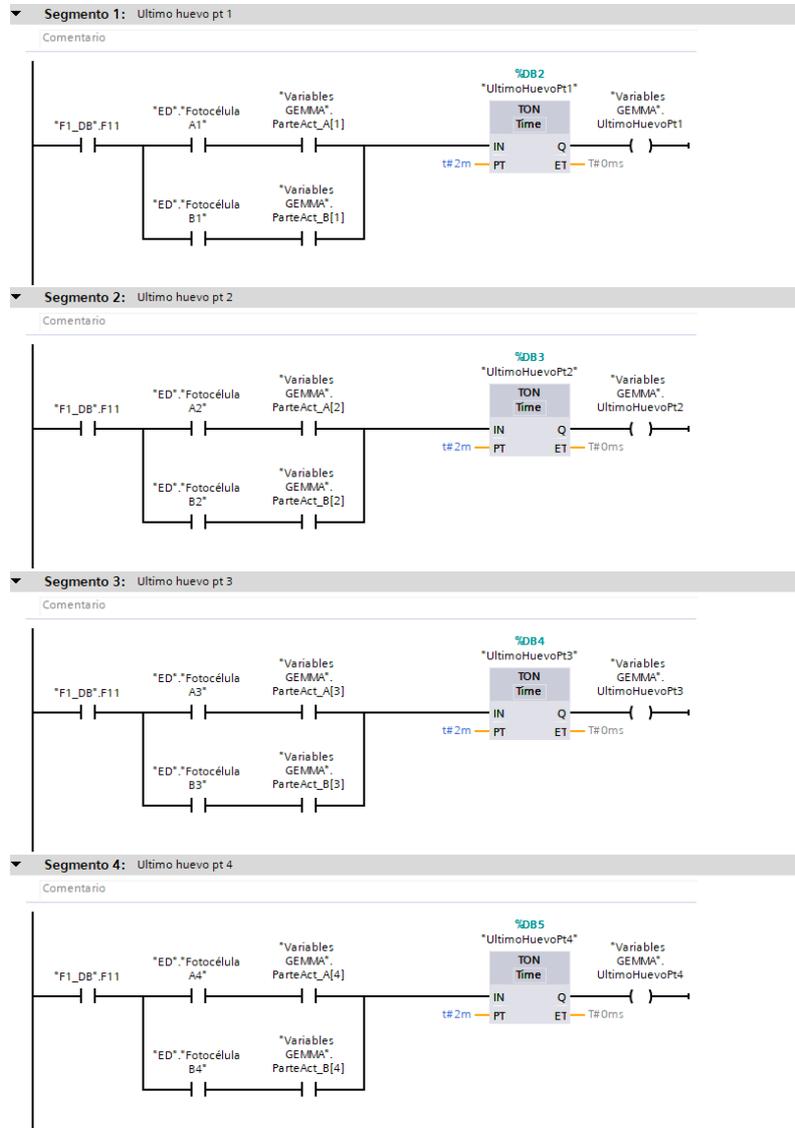
```

```
164     "Variables_HMI".Estado_CN_HMI[4] := BOOL_TO_INT("CN4");
165     END_IF;
166
167     IF "ED"."Guardamotores nave 5" THEN
168         "Variables_HMI".Estado_CN_HMI[5] := 2;
169     ELSE
170         "Variables_HMI".Estado_CN_HMI[5] := BOOL_TO_INT("CN5");
171     END_IF;
172
173     IF "ED"."Guardamotores nave 6" THEN
174         "Variables_HMI".Estado_CN_HMI[6] := 2;
175     ELSE
176         "Variables_HMI".Estado_CN_HMI[6] := BOOL_TO_INT("CN6");
177     END_IF;
178
179     IF "ED"."Guardamotores nave 7" THEN
180         "Variables_HMI".Estado_CN_HMI[7] := 2;
181     ELSE
182         "Variables_HMI".Estado_CN_HMI[7] := BOOL_TO_INT("CN7");
183     END_IF;
184
185     IF "ED"."Guardamotores nave 8" THEN
186         "Variables_HMI".Estado_CN_HMI[8] := 2;
187     ELSE
188         "Variables_HMI".Estado_CN_HMI[8] := BOOL_TO_INT("CN8");
189     END_IF;
190
191     IF "ED"."Guardamotores nave 9" THEN
192         "Variables_HMI".Estado_CN_HMI[9] := 2;
193     ELSE
194         "Variables_HMI".Estado_CN_HMI[9] := BOOL_TO_INT("CN9");
195     END_IF;
196
197     IF "ED"."Guardamotores nave 10" THEN
198         "Variables_HMI".Estado_CN_HMI[10] := 2;
199     ELSE
200         "Variables_HMI".Estado_CN_HMI[10] := BOOL_TO_INT("CN10");
201     END_IF;
202
203     IF "ED"."Guardamotores nave 11" THEN
204         "Variables_HMI".Estado_CN_HMI[11] := 2;
205     ELSE
206         "Variables_HMI".Estado_CN_HMI[11] := BOOL_TO_INT("CN11");
207     END_IF;
208
209     IF "ED"."Guardamotores nave 12" THEN
210         "Variables_HMI".Estado_CN_HMI[12] := 2;
211     ELSE
212         "Variables_HMI".Estado_CN_HMI[12] := BOOL_TO_INT("CN12");
213     END_IF;
214     END_REGION
```

Código 20: Segmento 17: Estados (marcha/paro/error) HMI

B.5. Temporizadores [FC8]

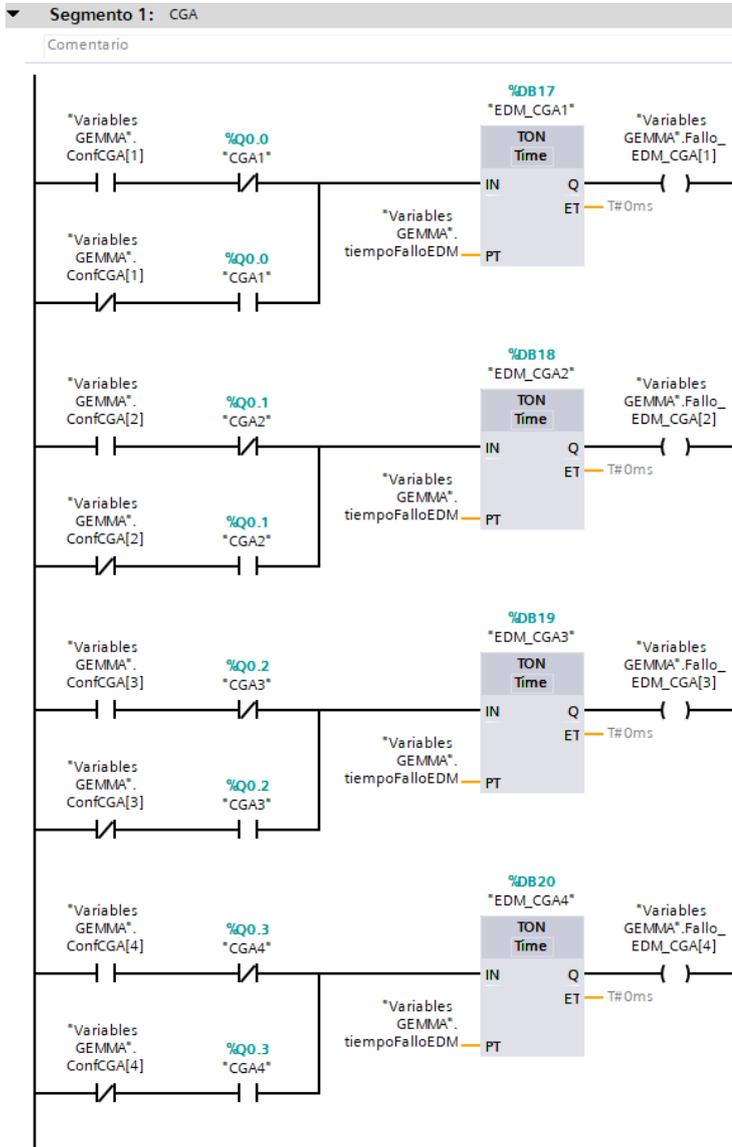
En esta sección se activan los temporizadores que marcan cuando ha pasado el último huevo. Hay un temporizador por parte.



Código 21: Temporizadores

B.6. Fallo_EDM [FC3]

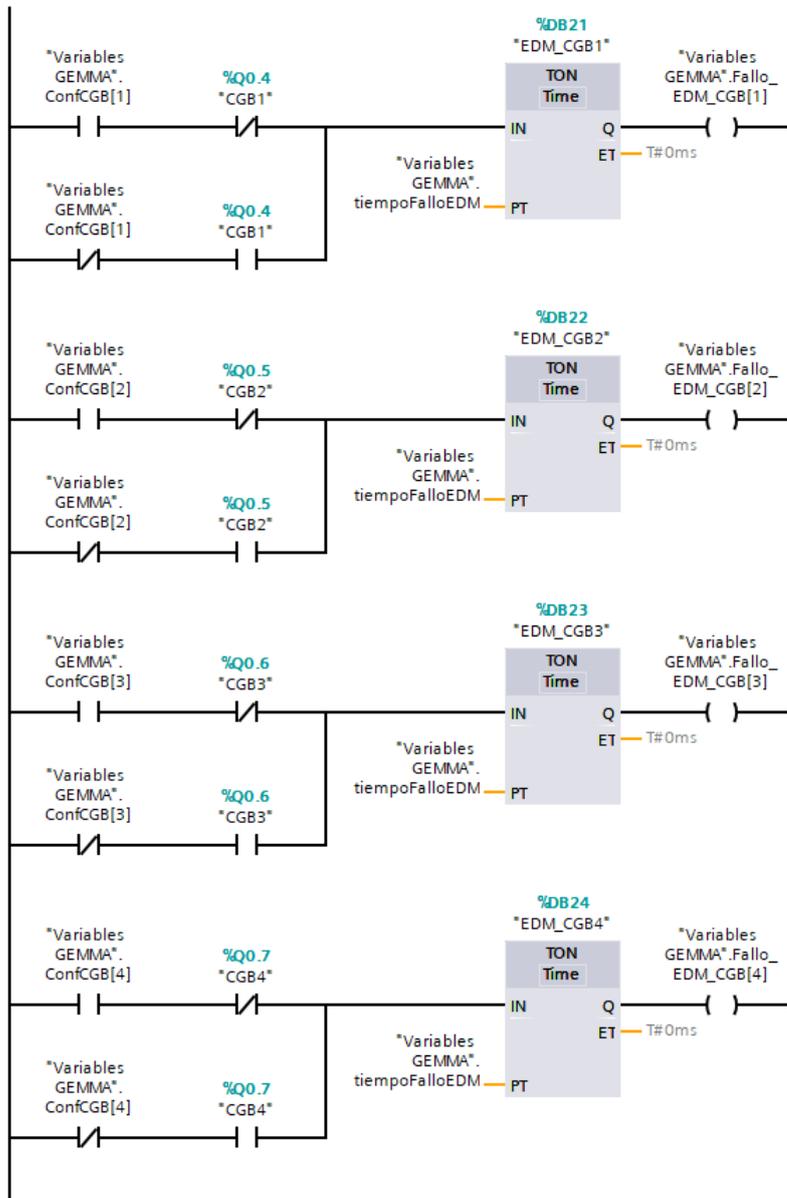
El fallo de EDM ocurre cuando no coincide el valor de salida física con el de la entrada de confirmación. Hay un temporizador con retardo a la activación por cada elemento con EDM.



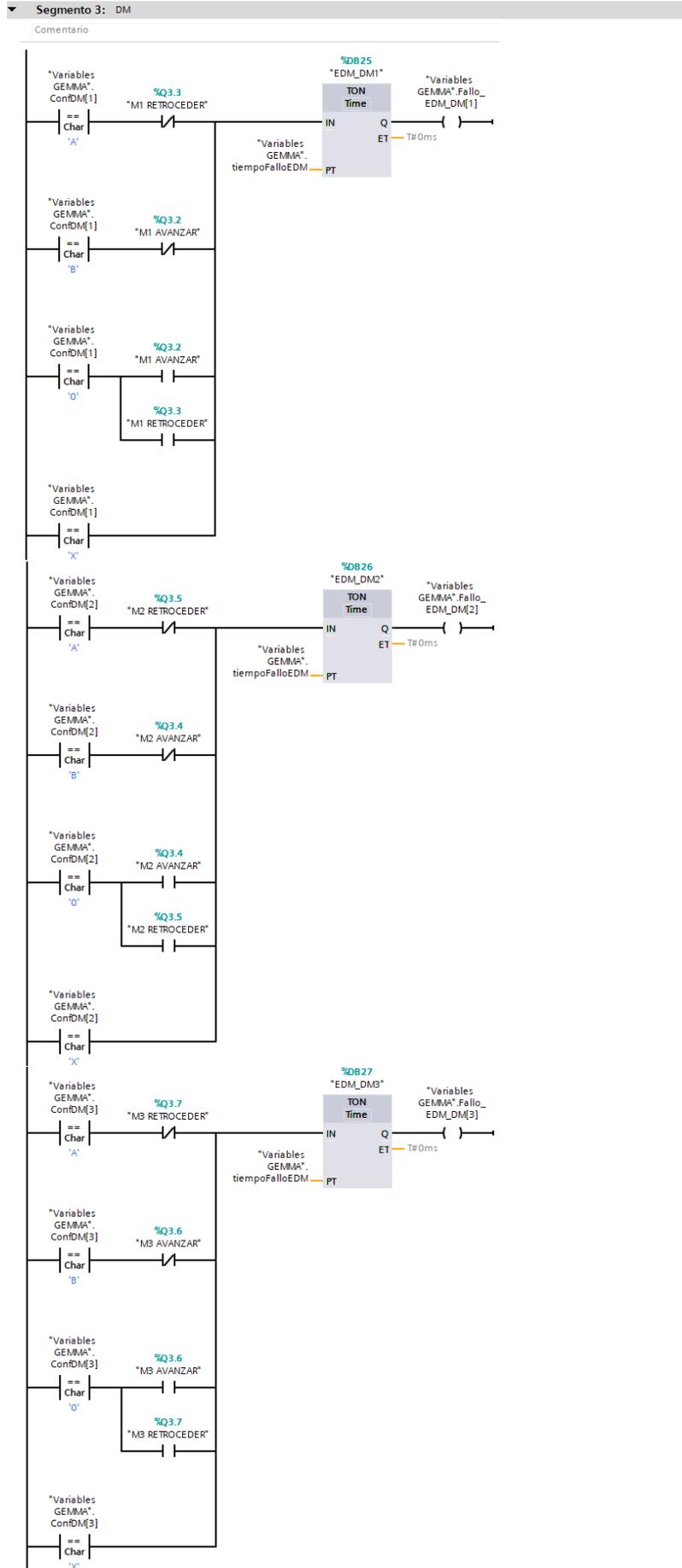
Código 22: EDM parte 1

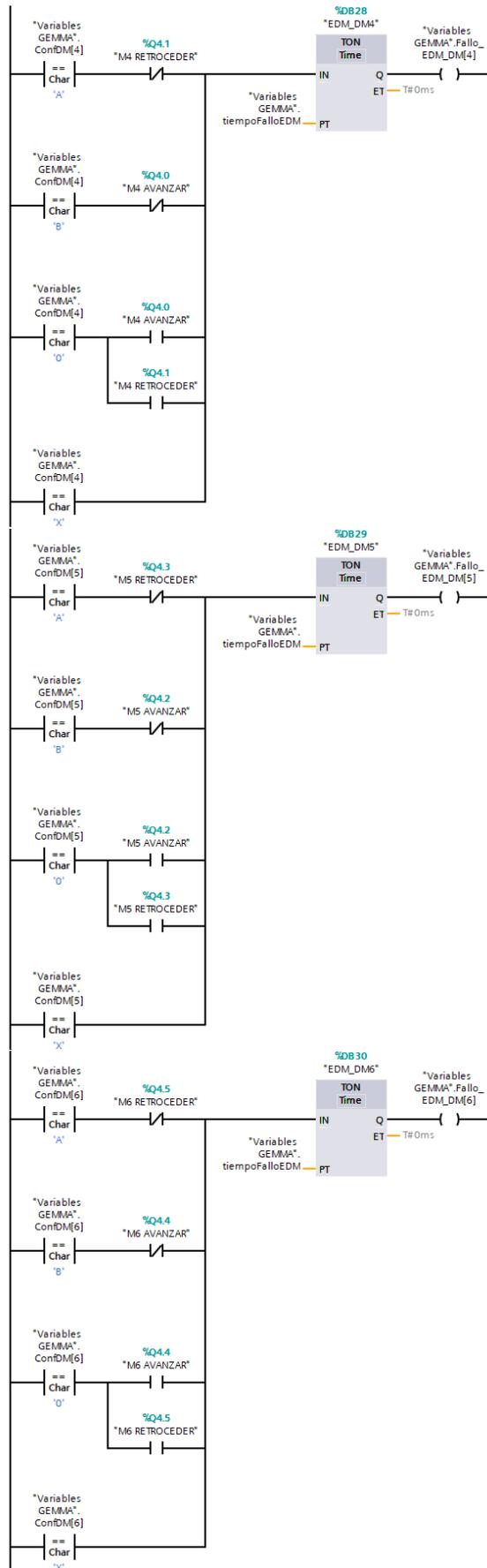
Segmento 2: CGB

Comentario

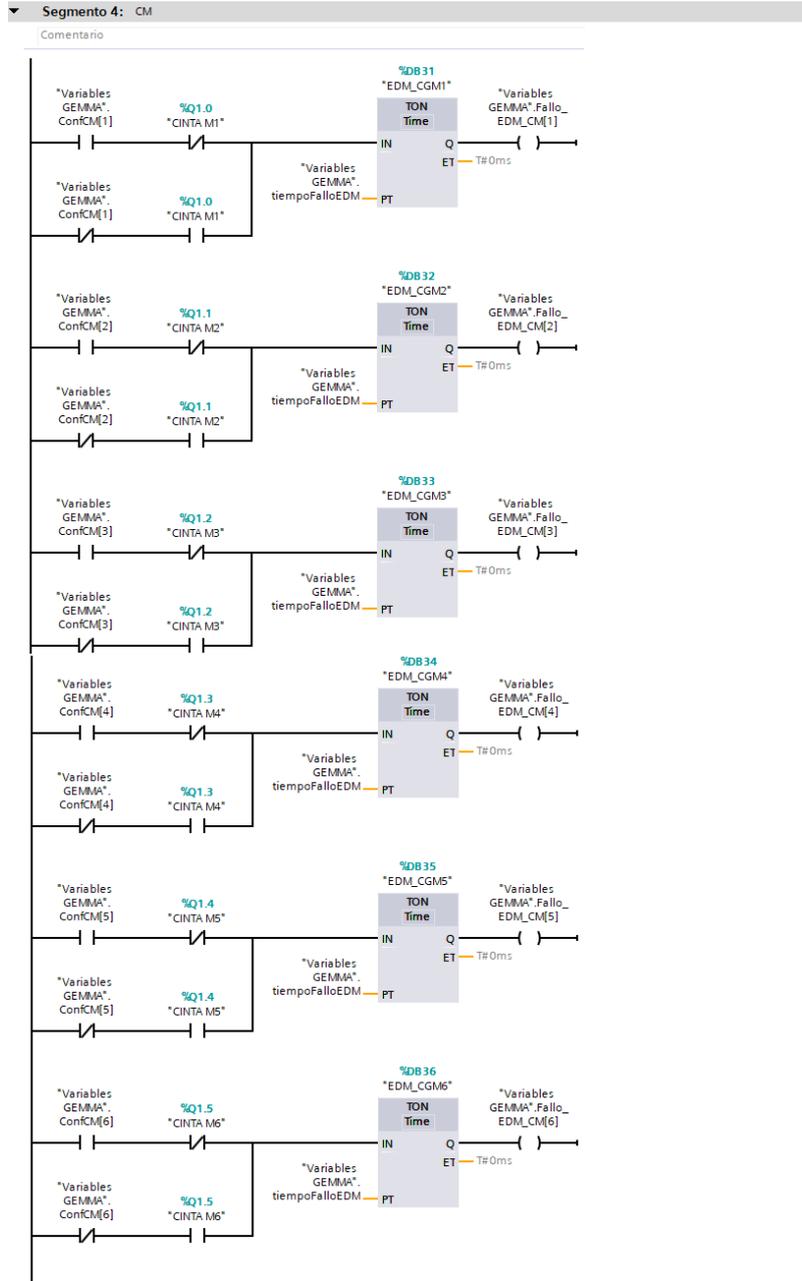


Código 23: EDM parte 2

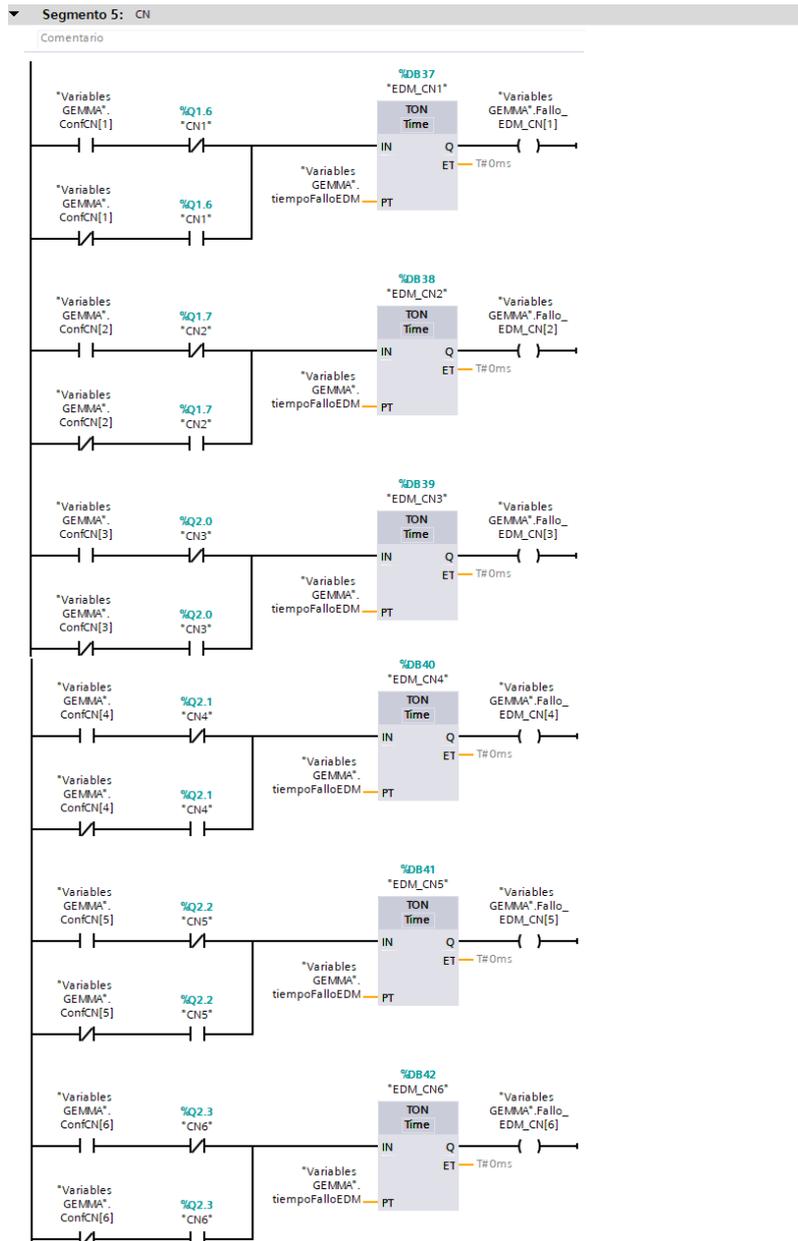




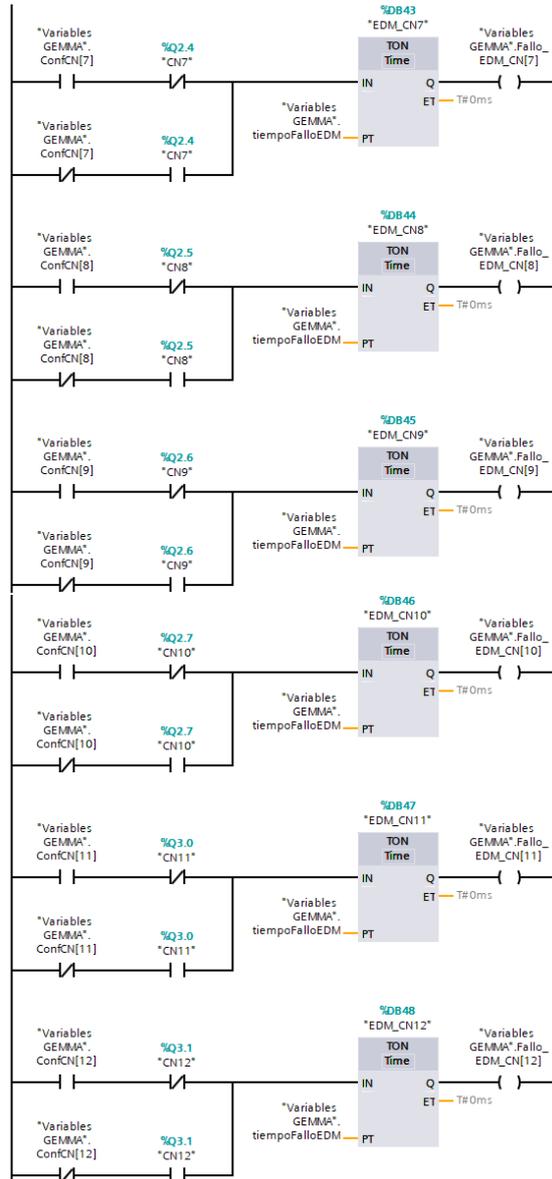
Código 25: EDM parte 4



Código 26: EDM parte 5



Código 27: EDM parte 6



Código 28: EDM parte 7

Finalmente, la variable "Fallo_EDM" se actualiza según la siguiente ecuación:

```

1      "Variables GEMMA".Fallo_EDM := "Variables GEMMA".Fallo_EDM_CGA[1] OR "Variables
      ↪ GEMMA".Fallo_EDM_CGA[2] OR "Variables GEMMA".Fallo_EDM_CGA[3] OR "Variables
      ↪ GEMMA".Fallo_EDM_CGA[4]
2 OR "Variables GEMMA".Fallo_EDM_CGB[1] OR "Variables GEMMA".Fallo_EDM_CGB[2] OR
      ↪ "Variables GEMMA".Fallo_EDM_CGB[3] OR "Variables GEMMA".Fallo_EDM_CGB[4]
3 OR "Variables GEMMA".Fallo_EDM_DM[1] OR "Variables GEMMA".Fallo_EDM_DM[2] OR "Variables
      ↪ GEMMA".Fallo_EDM_DM[3] OR "Variables GEMMA".Fallo_EDM_DM[4] OR "Variables
      ↪ GEMMA".Fallo_EDM_DM[5] OR "Variables GEMMA".Fallo_EDM_DM[6]
4 OR "Variables GEMMA".Fallo_EDM_DM[1] OR "Variables GEMMA".Fallo_EDM_CM[2] OR "Variables
      ↪ GEMMA".Fallo_EDM_CM[3] OR "Variables GEMMA".Fallo_EDM_CM[4] OR "Variables
      ↪ GEMMA".Fallo_EDM_CM[5] OR "Variables GEMMA".Fallo_EDM_CM[6]
5 OR "Variables GEMMA".Fallo_EDM_CN[1] OR "Variables GEMMA".Fallo_EDM_CN[2] OR "Variables
      ↪ GEMMA".Fallo_EDM_CN[3] OR "Variables GEMMA".Fallo_EDM_CN[4] OR "Variables
      ↪ GEMMA".Fallo_EDM_CN[5] OR "Variables GEMMA".Fallo_EDM_CN[6] OR "Variables
      ↪ GEMMA".Fallo_EDM_CN[7] OR "Variables GEMMA".Fallo_EDM_CN[8] OR "Variables
      ↪ GEMMA".Fallo_EDM_CN[9] OR "Variables GEMMA".Fallo_EDM_CN[10] OR "Variables
      ↪ GEMMA".Fallo_EDM_CN[11] OR "Variables GEMMA".Fallo_EDM_CN[12];

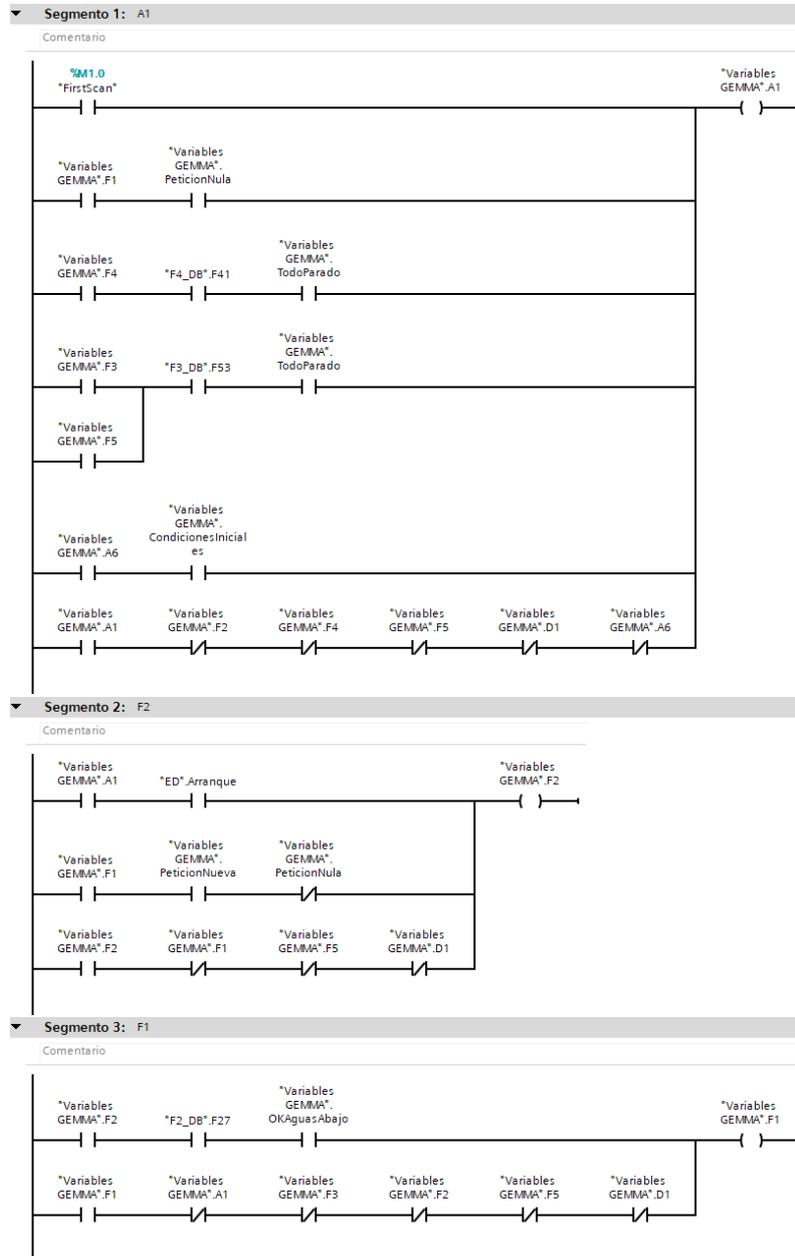
```

Código 29: Segmento 6: Fallo EDM

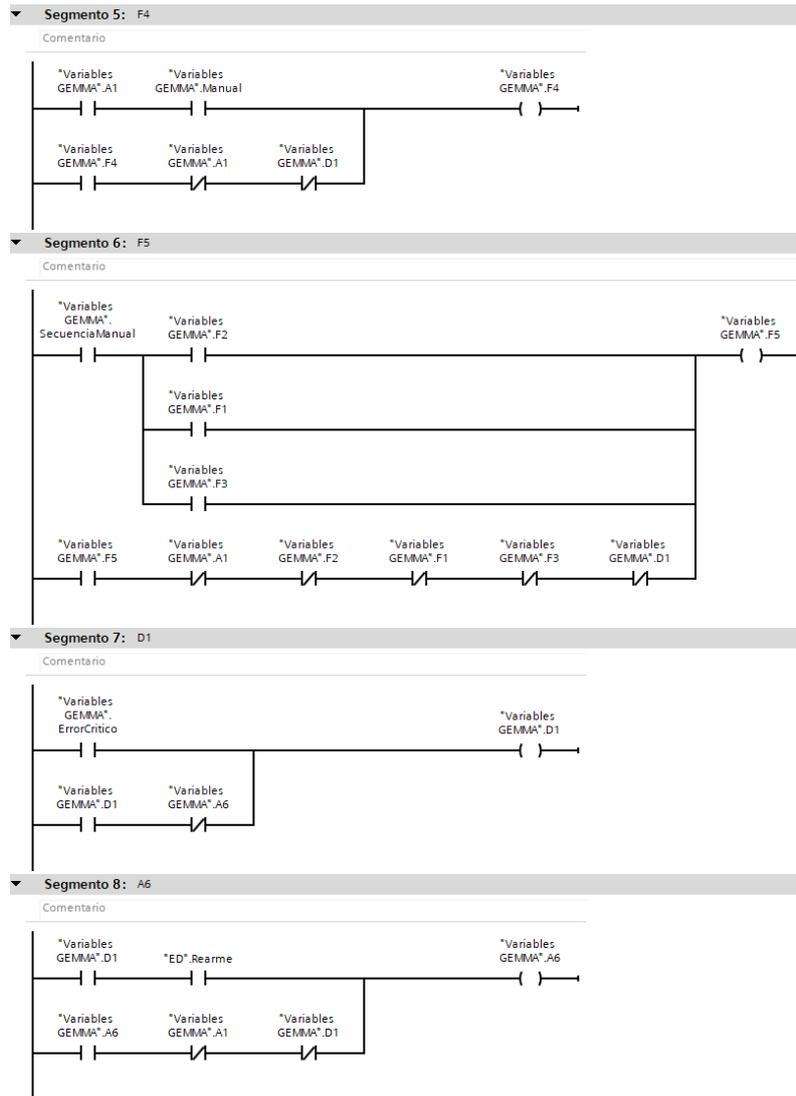
B.7. Actualización de estados [FC5]

B.7.1. GRAFCET superior

En la Subsubsección 5.2.1.1 se desarrollaron las ecuaciones de activación de cada etapa, con el siguiente código se implementan en el autómata:

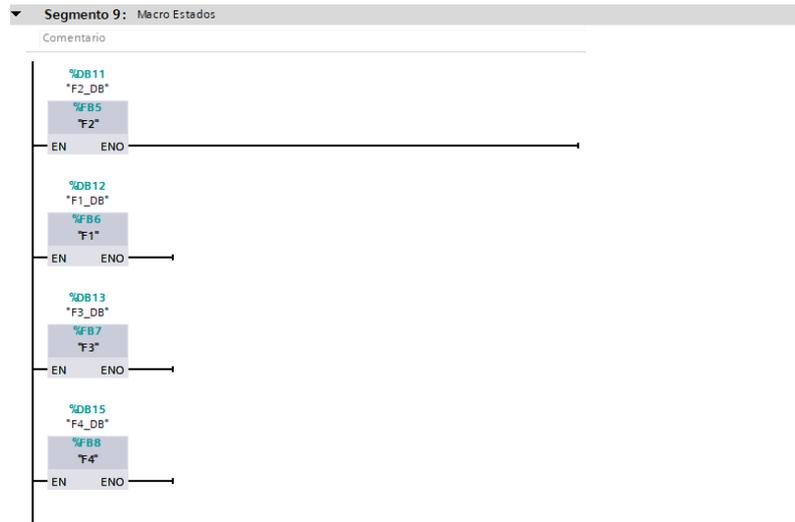


Código 30: Actualización de estados pt1



Código 31: Actualización de estados pt2

Finalmente, se llama a cada bloque de funciones de cada macro etapa:

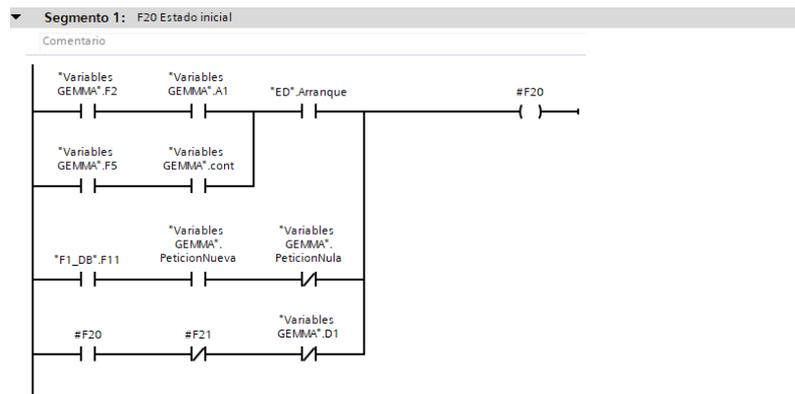


Código 32: Actualización de estados pt3

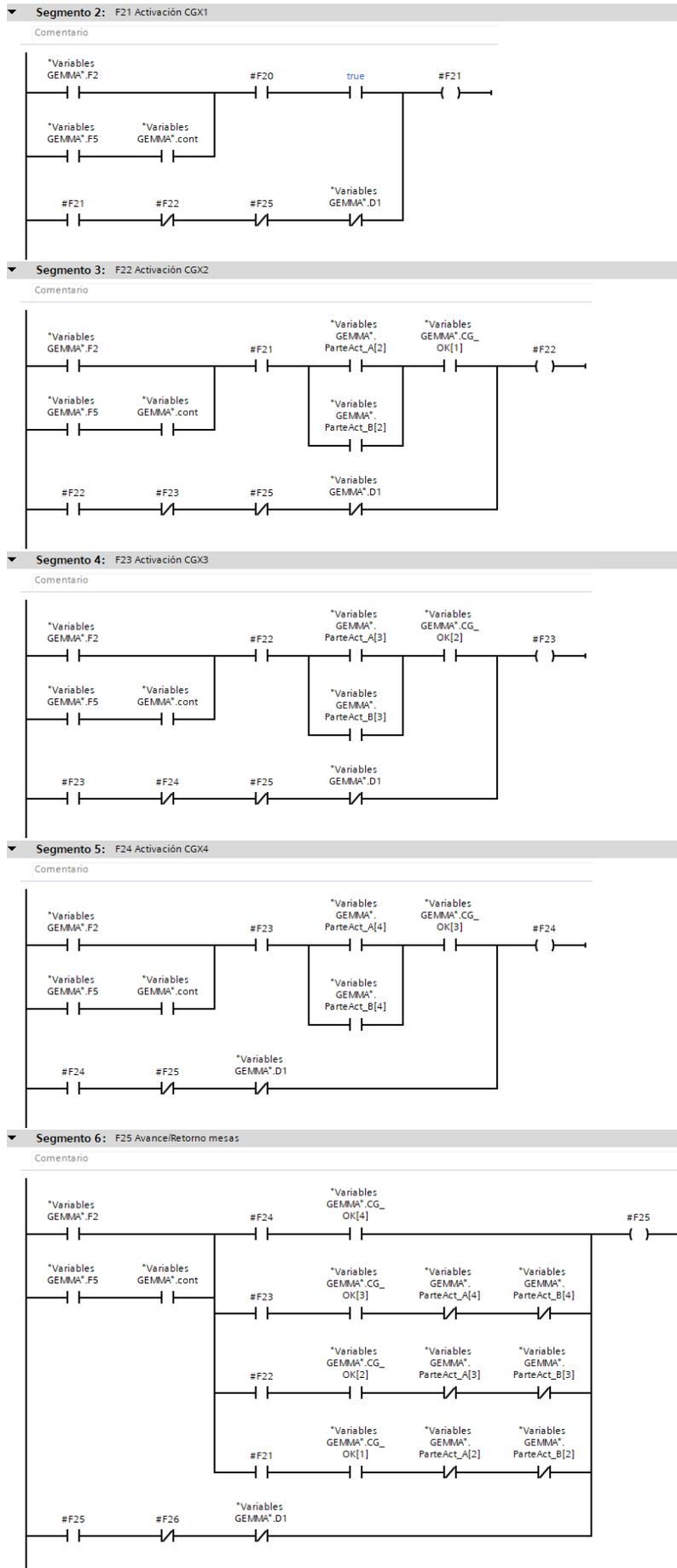
Cabe tener en cuenta que, aunque no lo marquen los estándares de programación GRAFCET, se han añadido las últimas etapas internas de los macro estados (F1, F2, F3 y F4) para la activación y desactivación de las etapas correspondientes del GRAFCET superior. Esto, también se utilizará para activar la primera etapa de cada macro estado y desactivar la última etapa de cada macro estado. Además, cómo la etapa F5 es virtual, se ha añadido la condición de activación en todas las etapas que esté activa su macro estado o F5 y la variable "cont".

B.7.2. F2 [FB5]

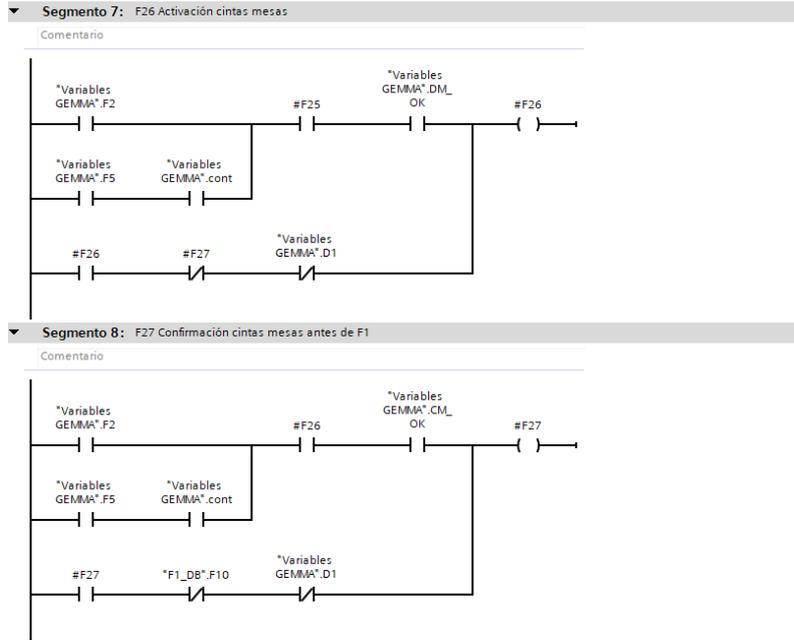
En el bloque de función F2, se genera también un bloque de datos, en él se encuentra cada estado de este macro estado. Como se ha realizado anteriormente, se implementan en Ladder las ecuaciones presentadas anteriormente.



Código 33: Actualización de estados F2 pt1



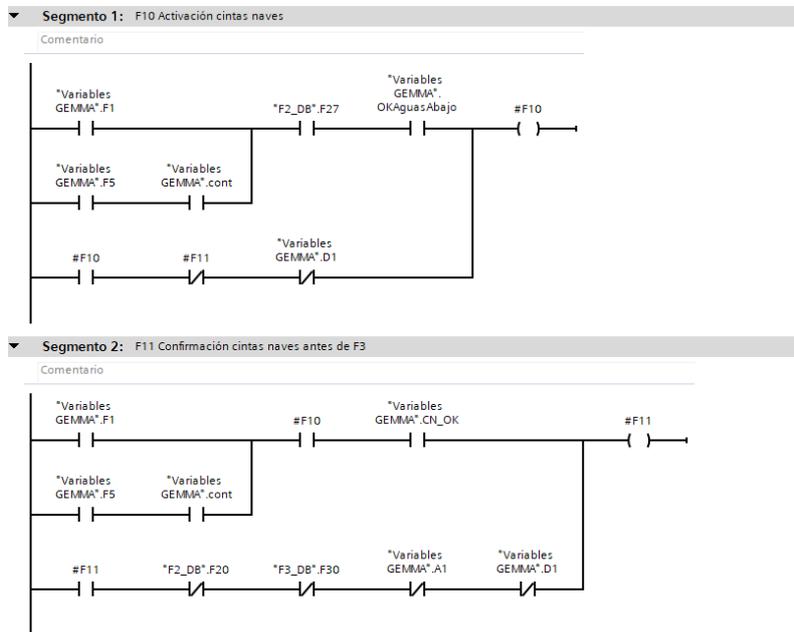
Código 34: Actualización de estados F2 pt2



Código 35: Actualización de estados F2 pt3

B.7.3. F1 [FB6]

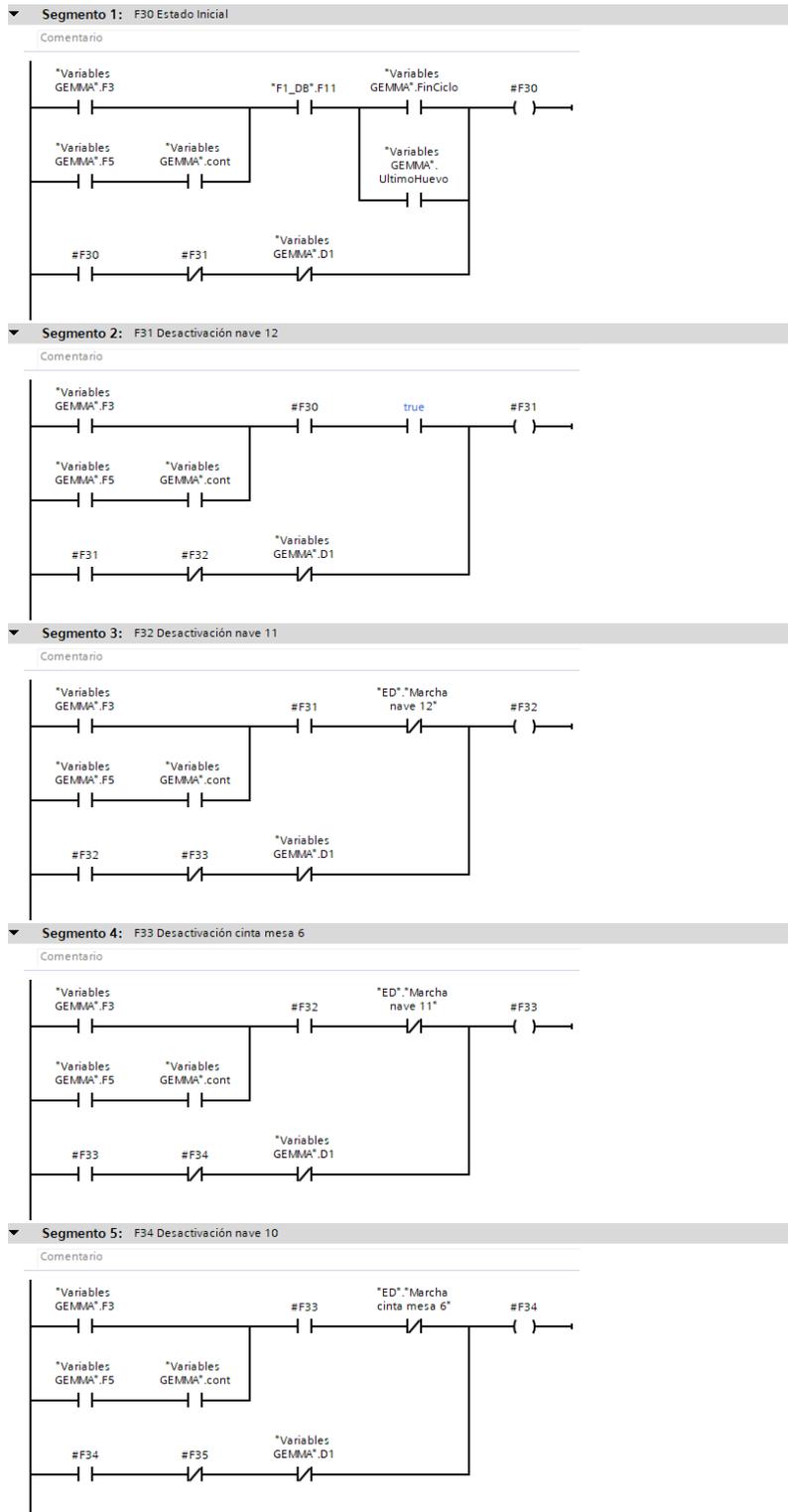
Como ya sabemos, la macro etapa de F1 es muy simple, ya que solo consta de dos etapas.



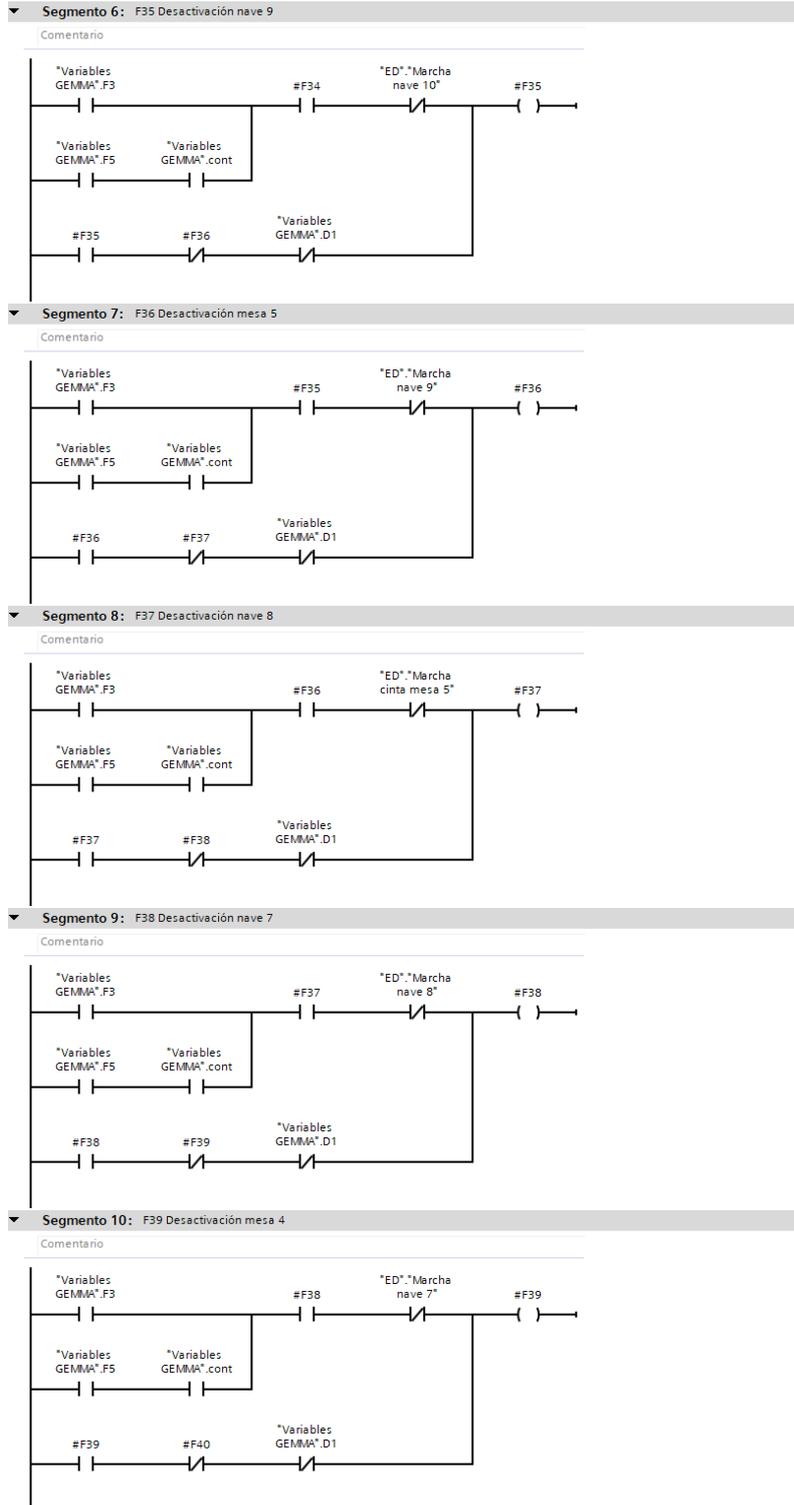
Código 36: Actualización de estados F1

B.7.4. F3 [FB7]

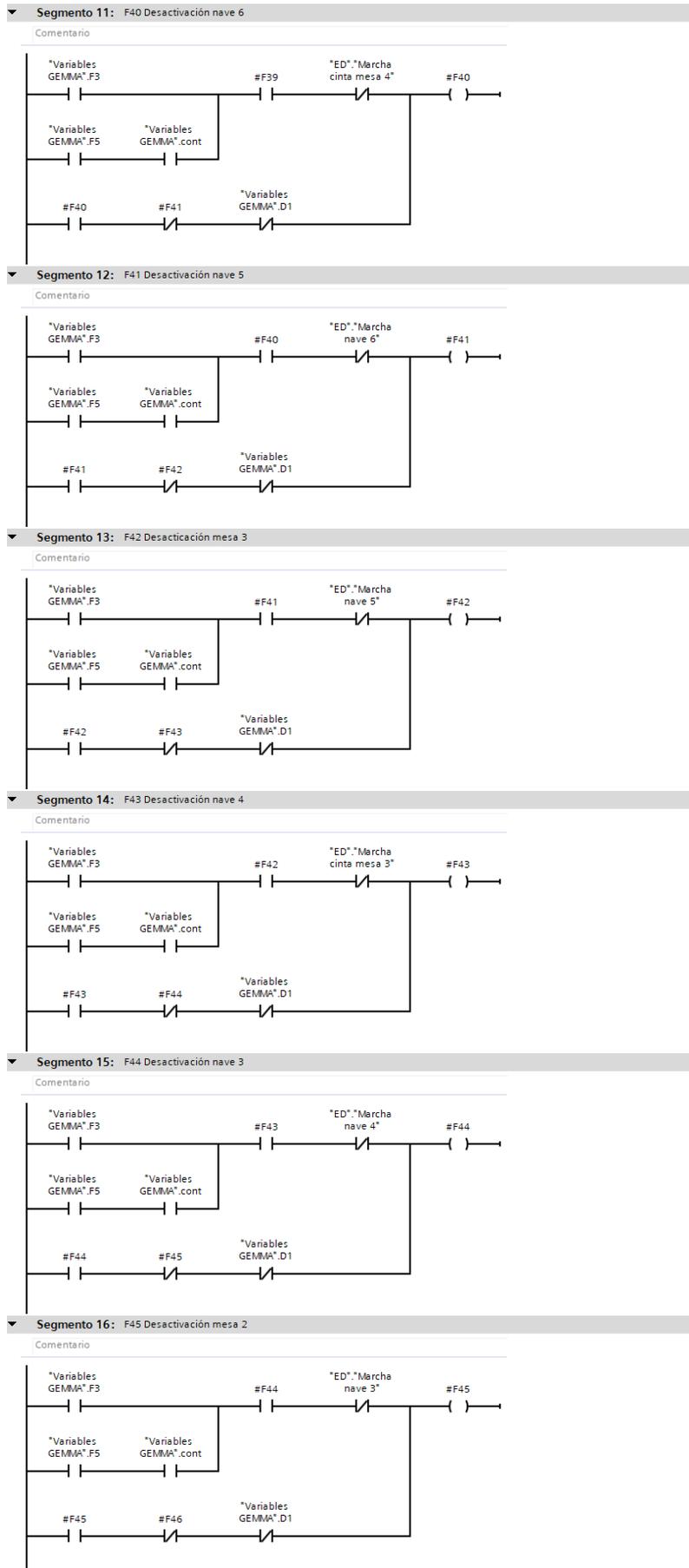
Se han aplicado las ecuaciones anteriormente expuestas sobre la macro etapa F3 y este es el resultado:



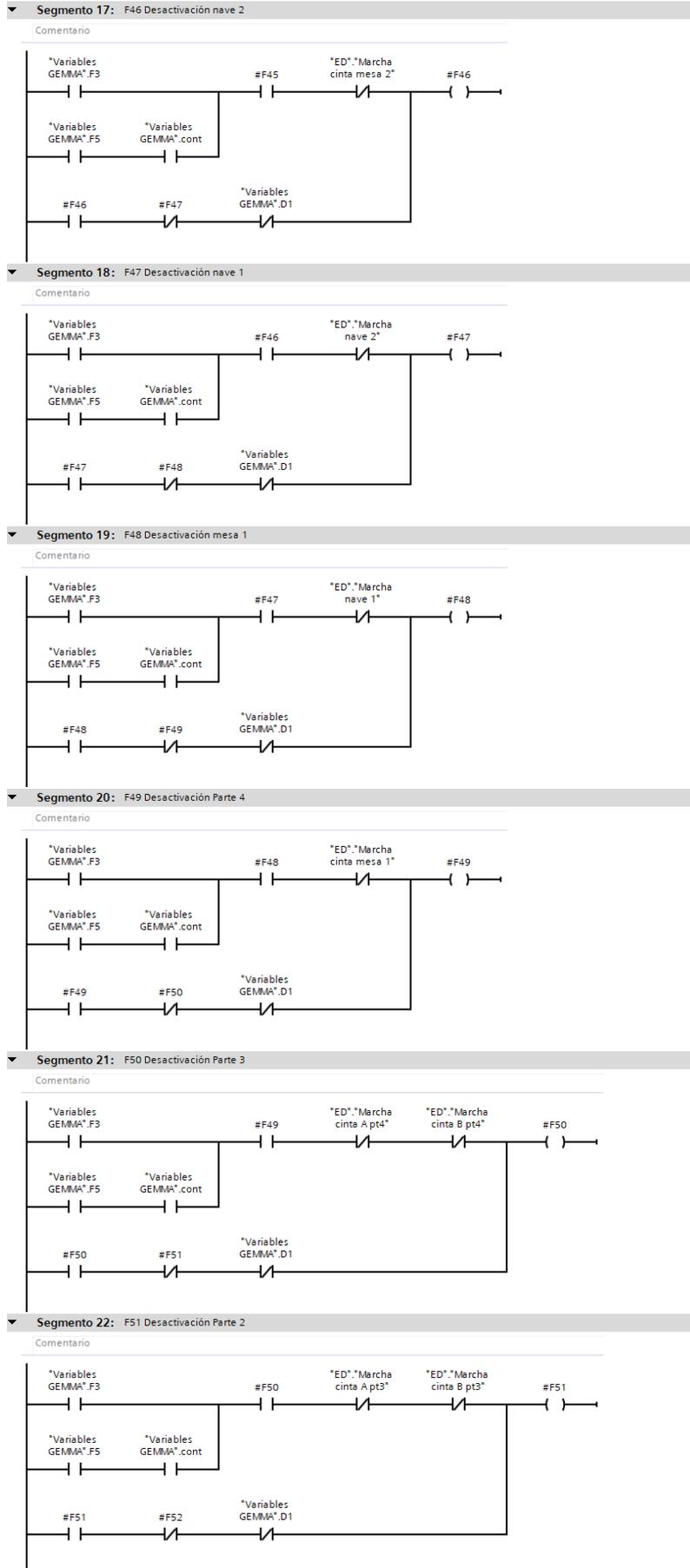
Código 37: Actualización de estados F3 pt1



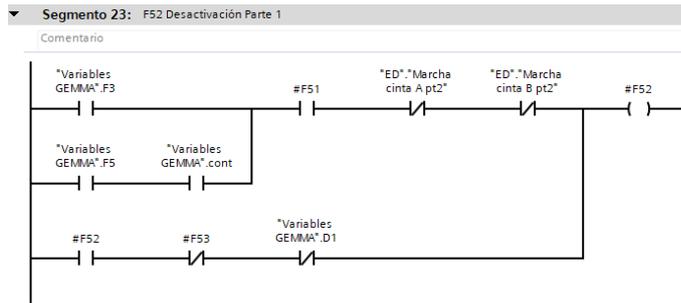
Código 38: Actualización de estados F3 pt2



Código 39: Actualización de estados F3 pt3



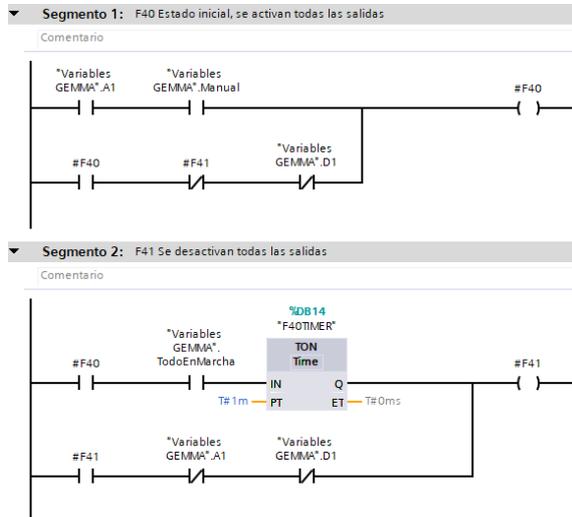
Código 40: Actualización de estados F3 pt4



Código 41: Actualización de estados F3 pt5

B.7.5. F4 [FB8]

Como con los anteriores macro estados, se han implementado las ecuaciones anteriormente expuestas.



Código 42: Actualización de estados F4

B.8. Ecuaciones de salida [FC7]

Como su nombre indica, en esta función se asignan los valores de salida según las ecuaciones.

```

1     IF ("F2_DB".F21 AND "Variables GEMMA".ParteAct_A[1]) OR "F4_DB".F40 THEN
2         "CGA1" := 1;
3         "CGA1 PT2" := 1;
4     ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F52 OR "Variables GEMMA".A1
↪     OR "F4_DB".F41 THEN
5         "CGA1" := 0;
6         "CGA1 PT2" := 0;
7     END_IF;
8
9     IF ("F2_DB".F22 AND "Variables GEMMA".ParteAct_A[2]) OR "F4_DB".F40 THEN
10        "CGA2" := 1;
11    ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F51 OR "Variables GEMMA".A1 OR
↪    "F4_DB".F41 THEN
12        "CGA2" := 0;
13    END_IF;
14
15    IF ("F2_DB".F23 AND "Variables GEMMA".ParteAct_A[3]) OR "F4_DB".F40 THEN
16        "CGA3" := 1;
17    ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F50 OR "Variables GEMMA".A1 OR
↪    "F4_DB".F41 THEN
18        "CGA3" := 0;
19    END_IF;
20
21    IF ("F2_DB".F24 AND "Variables GEMMA".ParteAct_A[4]) OR "F4_DB".F40 THEN
22        "CGA4" := 1;
23        "CGA4 PT2" := 1;
24    ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F49 OR "Variables GEMMA".A1 OR
↪    "F4_DB".F41 THEN
25        "CGA4" := 0;
26        "CGA4 PT2" := 0;
27    END_IF;

```

Código 43: Segmento 2: Cinta A

```

1     IF ("F2_DB".F21 AND "Variables GEMMA".ParteAct_B[1]) OR "F4_DB".F40 THEN
2         "CGB1" := 1;
3         "CGB1 PT2" := 1;
4     ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F52 OR "Variables GEMMA".A1 OR
↪    "F4_DB".F41 THEN
5         "CGB1" := 0;
6         "CGB1 PT2" := 0;
7     END_IF;
8
9     IF ("F2_DB".F22 AND "Variables GEMMA".ParteAct_B[2]) OR "F4_DB".F40 THEN
10        "CGB2" := 1;
11    ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F51 OR "Variables GEMMA".A1 OR
↪    "F4_DB".F41 THEN
12        "CGB2" := 0;
13    END_IF;
14
15    IF ("F2_DB".F23 AND "Variables GEMMA".ParteAct_B[3]) OR "F4_DB".F40 THEN

```

```

16     "CGB3" := 1;
17 ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F50 OR "Variables GEMMA".A1 OR
↪ "F4_DB".F41 THEN
18     "CGB3" := 0;
19 END_IF;
20
21 IF ("F2_DB".F24 AND "Variables GEMMA".ParteAct_B[4]) OR "F4_DB".F40 THEN
22     "CGB4" := 1;
23     "CGB4 PT2" := 1;
24 ELSIF "Variables GEMMA".D1 OR "F2_DB".F20 OR "F3_DB".F49 OR "Variables GEMMA".A1 OR
↪ "F4_DB".F41 THEN
25     "CGB4" := 0;
26     "CGB4 PT2" := 0;
27 END_IF;

```

Código 44: Segmento 3: Cinta B

```

1     IF "F2_DB".F25 THEN
2     IF "Variables GEMMA".DM_Deseada[1] = 'B' THEN
3         "M1 AVANZAR" := 1;
4         "M1.1 AVANZAR" := 1;
5         "M1 AVANZAR.1" := 1;
6         "M1 RETROCEDER" := 0;
7         "M1 RETROCEDER.1" := 0;
8         "M1.1 RETROCEDER" := 0;
9
10        ELSIF "Variables GEMMA".DM_Deseada[1] = 'A' THEN
11            "M1 AVANZAR" := 0;
12            "M1.1 AVANZAR" := 0;
13            "M1 AVANZAR.1" := 0;
14            "M1 RETROCEDER" := 1;
15            "M1.1 RETROCEDER" := 1;
16            "M1 RETROCEDER.1" := 1;
17        ELSE
18            "M1 AVANZAR" := 0;
19            "M1.1 AVANZAR" := 0;
20            "M1 AVANZAR.1" := 0;
21            "M1 RETROCEDER" := 0;
22            "M1.1 RETROCEDER" := 0;
23            "M1 RETROCEDER.1" := 0;
24        END_IF;
25    END_IF;
26
27    IF "F2_DB".F26 THEN
28        IF "Variables GEMMA".DM_Deseada[1] <> '0' THEN
29            "CINTA M1" := 1;
30            "CINTA M1.1" := 1;
31            "CINTA M1.2" := 1;
32            "CINTA M1.3" := 1;
33        ELSE
34            "CINTA M1" := 0;
35            "CINTA M1.1" := 0;
36            "CINTA M1.2" := 0;
37            "CINTA M1.3" := 0;
38        END_IF;
39    END_IF;
40

```

```

41 IF "F4_DB".F40 THEN
42     "CINTA M1" := 1;
43     "CINTA M1.1" := 1;
44     "CINTA M1.2" := 1;
45     "CINTA M1.3" := 1;
46     "M1 AVANZAR" := 1;
47     "M1.1 AVANZAR" := 1;
48     "M1 AVANZAR.1" := 1;
49     "M1 RETROCEDER" := 0;
50     "M1 RETROCEDER.1" := 0;
51     "M1.1 RETROCEDER" := 0;
52 END_IF;
53
54 IF "Variables GEMMA".D1 OR "F2_DB".F20 OR "Variables GEMMA".A1 OR "F3_DB".F48 OR
↪ "F4_DB".F41 THEN
55     "M1 AVANZAR" := 0;
56     "M1.1 AVANZAR" := 0;
57     "M1 AVANZAR.1" := 0;
58     "M1 RETROCEDER" := 0;
59     "M1.1 RETROCEDER" := 0;
60     "M1 RETROCEDER.1" := 0;
61     "CINTA M1" := 0;
62     "CINTA M1.1" := 0;
63     "CINTA M1.2" := 0;
64     "CINTA M1.3" := 0;
65 END_IF;

```

Código 45: Segmento 5: Mesa 1

```

1 IF "F2_DB".F25 THEN
2 IF "Variables GEMMA".DM_Deseada[2] = 'B' THEN
3     "M2 AVANZAR" := 1;
4     "M2.1 AVANZAR" := 1;
5     "M2 AVANZAR.1" := 1;
6     "M2 RETROCEDER" := 0;
7     "M2.1 RETROCEDER" := 0;
8     "M2 RETROCEDER.1" := 0;
9 ELSEIF "Variables GEMMA".DM_Deseada[2] = 'A' THEN
10    "M2 AVANZAR" := 0;
11    "M2.1 AVANZAR" := 0;
12    "M2 AVANZAR.1" := 0;
13    "M2 RETROCEDER" := 1;
14    "M2.1 RETROCEDER" := 1;
15    "M2 RETROCEDER.1" := 1;
16 ELSE
17    "M2 AVANZAR" := 0;
18    "M2.1 AVANZAR" := 0;
19    "M2 AVANZAR.1" := 0;
20    "M2 RETROCEDER" := 0;
21    "M2.1 RETROCEDER" := 0;
22    "M2 RETROCEDER.1" := 0;
23    "CINTA M2" := 0;
24 END_IF;
25 END_IF;
26
27 IF "F2_DB".F26 THEN
28     IF "Variables GEMMA".DM_Deseada[2] <> '0' THEN

```

```

29     "CINTA M2" := 1;
30     "CINTA M2.1" := 1;
31     "CINTA M2.2" := 1;
32     "CINTA M2.3" := 1;
33     ELSE
34         "CINTA M2" := 0;
35         "CINTA M2.1" := 0;
36         "CINTA M2.2" := 0;
37         "CINTA M2.3" := 0;
38     END_IF;
39 END_IF;
40
41 IF "F4_DB".F40 THEN
42     "CINTA M2" := 1;
43     "CINTA M2.1" := 1;
44     "CINTA M2.2" := 1;
45     "CINTA M2.3" := 1;
46     "M2 AVANZAR" := 1;
47     "M2.1 AVANZAR" := 1;
48     "M2 AVANZAR.1" := 1;
49     "M2 RETROCEDER" := 0;
50     "M2.1 RETROCEDER" := 0;
51     "M2 RETROCEDER.1" := 0;
52 END_IF;
53
54 IF "Variables GEMMA".D1 OR "F2_DB".F20 OR "Variables GEMMA".A1 OR "F3_DB".F45 OR
↵ "F4_DB".F41 THEN
55     "M2 AVANZAR" := 0;
56     "M2.1 AVANZAR" := 0;
57     "M2 AVANZAR.1" := 0;
58     "M2 RETROCEDER" := 0;
59     "M2.1 RETROCEDER" := 0;
60     "M2 RETROCEDER.1" := 0;
61     "CINTA M2" := 0;
62     "CINTA M2.1" := 0;
63     "CINTA M2.2" := 0;
64     "CINTA M2.3" := 0;
65
66 END_IF;

```

Código 46: Segmento 6: Mesa 2

```

1     IF "F2_DB".F25 THEN
2     IF "Variables GEMMA".DM_Deseada[3] = 'B' THEN
3         "M3 AVANZAR" := 1;
4         "M3.1 AVANZAR" := 1;
5         "M3 AVANZAR.1" := 1;
6         "M3 RETROCEDER" := 0;
7         "M3.1 RETROCEDER" := 0;
8         "M3 RETROCEDER.1" := 0;
9     ELSIF "Variables GEMMA".DM_Deseada[3] = 'A' THEN
10        "M3 AVANZAR" := 0;
11        "M3.1 AVANZAR" := 0;
12        "M3 AVANZAR.1" := 0;
13        "M3 RETROCEDER" := 1;
14        "M3.1 RETROCEDER" := 1;
15        "M3 RETROCEDER.1" := 1;

```

```

16     ELSE
17         "M3 AVANZAR" := 0;
18         "M3.1 AVANZAR" := 0;
19         "M3 AVANZAR.1" := 0;
20         "M3 RETROCEDER" := 0;
21         "M3.1 RETROCEDER" := 0;
22         "M3 RETROCEDER.1" := 0;
23         "CINTA M3" := 0;
24         "CINTA M3.1" := 0;
25         "CINTA M3.2" := 0;
26         "CINTA M3.3" := 0;
27     END_IF;
28 END_IF;
29
30 IF "F2_DB".F26 THEN
31     IF "Variables GEMMA".DM_Deseada[3] <> '0' THEN
32         "CINTA M3" := 1;
33         "CINTA M3.1" := 1;
34         "CINTA M3.2" := 1;
35         "CINTA M3.3" := 1;
36     ELSE
37         "CINTA M3" := 0;
38         "CINTA M3.1" := 0;
39         "CINTA M3.2" := 0;
40         "CINTA M3.3" := 0;
41     END_IF;
42 END_IF;
43
44 IF "F4_DB".F40 THEN
45     "CINTA M3" := 1;
46     "CINTA M3.1" := 1;
47     "CINTA M3.2" := 1;
48     "CINTA M3.3" := 1;
49     "M3 AVANZAR" := 1;
50     "M3.1 AVANZAR" := 1;
51     "M3 AVANZAR.1" := 1;
52     "M3 RETROCEDER" := 0;
53     "M3.1 RETROCEDER" := 0;
54     "M3 RETROCEDER.1" := 0;
55 END_IF;
56
57 IF "Variables GEMMA".D1 OR "F2_DB".F20 OR "Variables GEMMA".A1 OR "F3_DB".F42 OR
↪ "F4_DB".F41 THEN
58     "M3 AVANZAR" := 0;
59     "M3.1 AVANZAR" := 0;
60     "M3 AVANZAR.1" := 0;
61     "M3 RETROCEDER" := 0;
62     "M3.1 RETROCEDER" := 0;
63     "M3 RETROCEDER.1" := 0;
64     "CINTA M3" := 0;
65     "CINTA M3.1" := 0;
66     "CINTA M3.2" := 0;
67     "CINTA M3.3" := 0;
68 END_IF;
69

```

Código 47: Segmento 7: Mesa 3

```

1  IF "F2_DB".F25 THEN
2  IF "Variables GEMMA".DM_Deseada[4] = 'B' THEN
3      "M4 AVANZAR" := 1;
4      "M4.1 AVANZAR" := 1;
5      "M4 AVANZAR.1" := 1;
6      "M4 RETROCEDER" := 0;
7      "M4.1 RETROCEDER" := 0;
8      "M4 RETROCEDER.1" := 0;
9  ELSIF "Variables GEMMA".DM_Deseada[4] = 'A' THEN
10     "M4 AVANZAR" := 0;
11     "M4.1 AVANZAR" := 0;
12     "M4 AVANZAR.1" := 0;
13     "M4 RETROCEDER" := 1;
14     "M4.1 RETROCEDER" := 1;
15     "M4 RETROCEDER.1" := 1;
16  ELSE
17     "M4 AVANZAR" := 0;
18     "M4.1 AVANZAR" := 0;
19     "M4 AVANZAR.1" := 0;
20     "M4 RETROCEDER" := 0;
21     "M4.1 RETROCEDER" := 0;
22     "M4 RETROCEDER.1" := 0;
23     "CINTA M4" := 0;
24     "CINTA M4.1" := 0;
25     "CINTA M4.2" := 0;
26     "CINTA M4.3" := 0;
27  END_IF;
28 END_IF;
29
30 IF "F2_DB".F26 THEN
31     IF "Variables GEMMA".DM_Deseada[4] <> '0' THEN
32         "CINTA M4" := 1;
33         "CINTA M4.1" := 1;
34         "CINTA M4.2" := 1;
35         "CINTA M4.3" := 1;
36     ELSE
37         "CINTA M4" := 0;
38         "CINTA M4.1" := 0;
39         "CINTA M4.2" := 0;
40         "CINTA M4.3" := 0;
41     END_IF;
42 END_IF;
43
44 IF "F4_DB".F40 THEN
45     "CINTA M4" := 1;
46     "CINTA M4.1" := 1;
47     "CINTA M4.2" := 1;
48     "CINTA M4.3" := 1;
49     "M4 AVANZAR" := 1;
50     "M4.1 AVANZAR" := 1;
51     "M4 AVANZAR.1" := 1;
52     "M4 RETROCEDER" := 0;
53     "M4.1 RETROCEDER" := 0;
54     "M4 RETROCEDER.1" := 0;
55 END_IF;
56
57 IF "Variables GEMMA".D1 OR "F2_DB".F20 OR "Variables GEMMA".A1 OR "F3_DB".F39 OR
↵ "F4_DB".F41 THEN

```

```
58 "M4 AVANZAR" := 0;  
59 "M4.1 AVANZAR" := 0;  
60 "M4 AVANZAR.1" := 0;  
61 "M4 RETROCEDER" := 0;  
62 "M4.1 RETROCEDER" := 0;  
63 "M4 RETROCEDER.1" := 0;  
64 "CINTA M4" := 0;  
65 "CINTA M4.1" := 0;  
66 "CINTA M4.2" := 0;  
67 "CINTA M4.3" := 0;  
68 END_IF;  
69
```

Código 48: Segmento 8: Mesa 4

```
1 IF "F2_DB".F25 THEN  
2 IF "Variables GEMMA".DM_Deseada[5] = 'B' THEN  
3 "M5 AVANZAR" := 1;  
4 "M5.1 AVANZAR" := 1;  
5 "M5 AVANZAR.1" := 1;  
6 "M5 RETROCEDER" := 0;  
7 "M5.1 RETROCEDER" := 0;  
8 "M5 RETROCEDER.1" := 0;  
9 ELSIF "Variables GEMMA".DM_Deseada[5] = 'A' THEN  
10 "M5 AVANZAR" := 0;  
11 "M5.1 AVANZAR" := 0;  
12 "M5 AVANZAR.1" := 0;  
13 "M5 RETROCEDER" := 1;  
14 "M5.1 RETROCEDER" := 1;  
15 "M5 RETROCEDER.1" := 1;  
16 ELSE  
17 "M5 AVANZAR" := 0;  
18 "M5.1 AVANZAR" := 0;  
19 "M5 AVANZAR.1" := 0;  
20 "M5 RETROCEDER" := 0;  
21 "M5.1 RETROCEDER" := 0;  
22 "M5 RETROCEDER.1" := 0;  
23 "CINTA M5" := 0;  
24 "CINTA M5.1" := 0;  
25 "CINTA M5.2" := 0;  
26 "CINTA M5.3" := 0;  
27 END_IF;  
28 END_IF;  
29  
30 IF "F2_DB".F26 THEN  
31 IF "Variables GEMMA".DM_Deseada[5] <> '0' THEN  
32 "CINTA M5" := 1;  
33 "CINTA M5.1" := 1;  
34 "CINTA M5.2" := 1;  
35 "CINTA M5.3" := 1;  
36 ELSE  
37 "CINTA M5" := 0;  
38 "CINTA M5.1" := 0;  
39 "CINTA M5.2" := 0;  
40 "CINTA M5.3" := 0;  
41 END_IF;  
42 END_IF;
```

```

43
44 IF "F4_DB".F40 THEN
45     "CINTA M5" := 1;
46     "CINTA M5.1" := 1;
47     "CINTA M5.2" := 1;
48     "CINTA M5.3" := 1;
49     "M5 AVANZAR" := 1;
50     "M5.1 AVANZAR" := 1;
51     "M5 AVANZAR.1" := 1;
52     "M5 RETROCEDER" := 0;
53     "M5.1 RETROCEDER" := 0;
54     "M5 RETROCEDER.1" := 0;
55 END_IF;
56
57 IF "Variables GEMMA".D1 OR "F2_DB".F20 OR "Variables GEMMA".A1 OR "F3_DB".F36 OR
↪ "F4_DB".F41 THEN
58     "M5 AVANZAR" := 0;
59     "M5.1 AVANZAR" := 0;
60     "M5 AVANZAR.1" := 0;
61     "M5 RETROCEDER" := 0;
62     "M5.1 RETROCEDER" := 0;
63     "M5 RETROCEDER.1" := 0;
64     "CINTA M5" := 0;
65     "CINTA M5.1" := 0;
66     "CINTA M5.2" := 0;
67     "CINTA M5.3" := 0;
68 END_IF;
69

```

Código 49: Segmento 9: Mesa 5

```

1 IF "F2_DB".F25 THEN
2 IF "Variables GEMMA".DM_Deseada[6] = 'B' THEN
3     "M6 AVANZAR" := 1;
4     "M6.1 AVANZAR" := 1;
5     "M6 AVANZAR.1" := 1;
6     "M6 RETROCEDER" := 0;
7     "M6.1 RETROCEDER" := 0;
8     "M6 RETROCEDER.1" := 0;
9 ELSEIF "Variables GEMMA".DM_Deseada[6] = 'A' THEN
10    "M6 AVANZAR" := 0;
11    "M6.1 AVANZAR" := 0;
12    "M6 AVANZAR.1" := 0;
13    "M6 RETROCEDER" := 1;
14    "M6.1 RETROCEDER" := 1;
15    "M6 RETROCEDER.1" := 1;
16 ELSE
17    "M6 AVANZAR" := 0;
18    "M6.1 AVANZAR" := 0;
19    "M6 AVANZAR.1" := 0;
20    "M6 RETROCEDER" := 0;
21    "M6.1 RETROCEDER" := 0;
22    "M6 RETROCEDER.1" := 0;
23    "CINTA M6" := 0;
24    "CINTA M6.1" := 0;
25    "CINTA M6.2" := 0;
26    "CINTA M6.3" := 0;

```

```

27     END_IF;
28 END_IF;
29
30 IF "F2_DB".F26 THEN
31     IF "Variables GEMMA".DM_Deseada[6] <> '0' THEN
32         "CINTA M6" := 1;
33         "CINTA M6.1" := 1;
34         "CINTA M6.2" := 1;
35         "CINTA M6.3" := 1;
36     ELSE
37         "CINTA M6" := 0;
38         "CINTA M6.1" := 0;
39         "CINTA M6.2" := 0;
40         "CINTA M6.3" := 0;
41     END_IF;
42 END_IF;
43
44 IF "F4_DB".F40 THEN
45     "CINTA M6" := 1;
46     "CINTA M6.1" := 1;
47     "CINTA M6.2" := 1;
48     "CINTA M6.3" := 1;
49     "M6 AVANZAR" := 1;
50     "M6.1 AVANZAR" := 1;
51     "M6 AVANZAR.1" := 1;
52     "M6 RETROCEDER" := 0;
53     "M6.1 RETROCEDER" := 0;
54     "M6 RETROCEDER.1" := 0;
55 END_IF;
56
57 IF "Variables GEMMA".D1 OR "F2_DB".F20 OR "Variables GEMMA".A1 OR "F3_DB".F33 OR
↪ "F4_DB".F41 THEN
58     "M6 AVANZAR" := 0;
59     "M6.1 AVANZAR" := 0;
60     "M6 AVANZAR.1" := 0;
61     "M6 RETROCEDER" := 0;
62     "M6.1 RETROCEDER" := 0;
63     "M6 RETROCEDER.1" := 0;
64     "CINTA M6" := 0;
65     "CINTA M6.1" := 0;
66     "CINTA M6.2" := 0;
67     "CINTA M6.3" := 0;
68 END_IF;

```

Código 50: Segmento 10: Mesa 6

```

1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[1] <> '0') OR "F4_DB".F40 THEN
2         "CN1" := 1;
3         "CN1.2" := 1;
4         "CN1.4" := 1;
5         "NAVE 1" := 1;
6     ELSIF "Variables GEMMA".D1 OR "F3_DB".F47 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
↪ "Variables GEMMA".F2 THEN
7         "CN1" := 0;
8         "CN1.2" := 0;
9         "CN1.4" := 0;

```

```

10     "NAVE 1" := 0;
11 END_IF;

```

Código 51: Segmento 12: Nave 1

```

1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[2] <> '0') OR "F4_DB".F40 THEN
2     "CN2" := 1;
3     "CN2.2" := 1;
4     "CN2.4" := 1;
5     "NAVE 2" := 1;
6     ELSIF "Variables GEMMA".D1 OR "F3_DB".F46 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
↪ "Variables GEMMA".F2 THEN
7     "CN2" := 0;
8     "CN2.2" := 0;
9     "CN2.4" := 0;
10    "NAVE 2" := 0;
11 END_IF;

```

Código 52: Segmento 13: Nave 2

```

1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[3] <> '0') OR "F4_DB".F40 THEN
2     "CN3" := 1;
3     "CN3.2" := 1;
4     "CN3.4" := 1;
5     "NAVE 3" := 1;
6     ELSIF "Variables GEMMA".D1 OR "F3_DB".F44 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
↪ "Variables GEMMA".F2 THEN
7     "CN3" := 0;
8     "CN3.2" := 0;
9     "CN3.4" := 0;
10    "NAVE 3" := 0;
11 END_IF;

```

Código 53: Segmento 14: Nave 3

```

1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[4] <> '0') OR "F4_DB".F40 THEN
2     "CN4" := 1;
3     "CN4.2" := 1;
4     "CN4.4" := 1;
5     "NAVE 4" := 1;
6     ELSIF "Variables GEMMA".D1 OR "F3_DB".F43 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
↪ "Variables GEMMA".F2 THEN
7     "CN4" := 0;
8     "CN4.2" := 0;
9     "CN4.4" := 0;
10    "NAVE 4" := 0;
11 END_IF;

```

Código 54: Segmento 15: Nave 4

```
1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[5] <> '0') OR "F4_DB".F40 THEN
2     "CN5" := 1;
3     "CN5.2" := 1;
4     "NAVE 5" := 1;
5     ELSIF "Variables GEMMA".D1 OR "F3_DB".F41 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
6     ↪ "Variables GEMMA".F2 THEN
7     "CN5" := 0;
8     "CN5.2" := 0;
9     "NAVE 5" := 0;
10    END_IF;
```

Código 55: Segmento 16: Nave 5

```
1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[6] <> '0') OR "F4_DB".F40 THEN
2     "CN6" := 1;
3     "CN6.2" := 1;
4     "NAVE 6" := 1;
5     ELSIF "Variables GEMMA".D1 OR "F3_DB".F40 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
6     ↪ "Variables GEMMA".F2 THEN
7     "CN6" := 0;
8     "CN6.2" := 0;
9     "NAVE 6" := 0;
10    END_IF;
```

Código 56: Segmento 17: Nave 6

```
1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[7] <> '0') OR "F4_DB".F40 THEN
2     "CN7" := 1;
3     "CN7.2" := 1;
4     "NAVE 7" := 1;
5     ELSIF "Variables GEMMA".D1 OR "F3_DB".F38 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
6     ↪ "Variables GEMMA".F2 THEN
7     "CN7" := 0;
8     "CN7.2" := 0;
9     "NAVE 7" := 0;
10    END_IF;
```

Código 57: Segmento 18: Nave 7

```
1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[8] <> '0') OR "F4_DB".F40 THEN
2     "CN8" := 1;
3     "CN8.2" := 1;
4     "NAVE 8" := 1;
5     ELSIF "Variables GEMMA".D1 OR "F3_DB".F37 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
6     ↪ "Variables GEMMA".F2 THEN
7     "CN8" := 0;
8     "CN8.2" := 0;
9     "NAVE 8" := 0;
10    END_IF;
```

Código 58: Segmento 19: Nave 8

```

1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[9] <> '0') OR "F4_DB".F40 THEN
2     "CN9" := 1;
3     "CN9.2" := 1;
4     "NAVE 9" := 1;
5     ELSIF "Variables GEMMA".D1 OR "F3_DB".F35 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
6     ↪ "Variables GEMMA".F2 THEN
7     "CN9" := 0;
8     "CN9.2" := 0;
9     "NAVE 9" := 0;
10    END_IF;

```

Código 59: Segmento 20: Nave 9

```

1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[10] <> '0') OR "F4_DB".F40 THEN
2     "CN10" := 1;
3     "CN10.2" := 1;
4     "NAVE 10" := 1;
5     ELSIF "Variables GEMMA".D1 OR "F3_DB".F34 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
6     ↪ "Variables GEMMA".F2 THEN
7     "CN10" := 0;
8     "CN10.2" := 0;
9     "NAVE 10" := 0;
10    END_IF;

```

Código 60: Segmento 21: Nave 10

```

1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[11] <> '0') OR "F4_DB".F40 THEN
2     "CN11" := 1;
3     "CN11.2" := 1;
4     "NAVE 11" := 1;
5     ELSIF "Variables GEMMA".D1 OR "F3_DB".F32 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
6     ↪ "Variables GEMMA".F2 THEN
7     "CN11" := 0;
8     "CN11.2" := 0;
9     "NAVE 11" := 0;
10    END_IF;

```

Código 61: Segmento 22: Nave 11

```

1     IF ("F1_DB".F10 AND "Variables GEMMA".PetHuevosAct[12] <> '0') OR "F4_DB".F40 THEN
2     "CN12" := 1;
3     "CN12.2" := 1;
4     "NAVE 12" := 1;
5     ELSIF "Variables GEMMA".D1 OR "F3_DB".F31 OR "Variables GEMMA".A1 OR "F4_DB".F41 OR
6     ↪ "Variables GEMMA".F2 THEN
7     "CN12" := 0;
8     "CN12.2" := 0;
9     "NAVE 12" := 0;
10    END_IF;

```

Código 62: Segmento 23: Nave 12

B.9. Asignación de alarmas [FC2]

En esta función se asigna cada bit de los WORD que comunican con el HMI a la variable correspondiente. En el Apéndice C se explica para qué se utilizan estas variables.

```

1      "AlarmasHMI".W0.%X0 := "ED"."Guardamotores cinta A pt1";
2      "AlarmasHMI".W0.%X1 := "ED"."Guardamotores cinta A pt2";
3      "AlarmasHMI".W0.%X2 := "ED"."Guardamotores cinta A pt3";
4      "AlarmasHMI".W0.%X3 := "ED"."Guardamotores cinta A pt4";
5      "AlarmasHMI".W0.%X4 := "ED"."Guardamotores cinta B pt1";
6      "AlarmasHMI".W0.%X5 := "ED"."Guardamotores cinta B pt2";
7      "AlarmasHMI".W0.%X6 := "ED"."Guardamotores cinta B pt3";
8      "AlarmasHMI".W0.%X7 := "ED"."Guardamotores cinta B pt4";
9      "AlarmasHMI".W0.%X8 := "ED"."Guardamotores mesa 1";
10     "AlarmasHMI".W0.%X9 := "ED"."Guardamotores mesa 2";
11     "AlarmasHMI".W0.%X10 := "ED"."Guardamotores mesa 3";
12     "AlarmasHMI".W0.%X11 := "ED"."Guardamotores mesa 4";
13     "AlarmasHMI".W0.%X12 := "ED"."Guardamotores mesa 5";
14     "AlarmasHMI".W0.%X13 := "ED"."Guardamotores mesa 6";
15     "AlarmasHMI".W0.%X14 := "ED"."Guardamotores nave 1";
16     "AlarmasHMI".W0.%X15 := "ED"."Guardamotores nave 2";
17
18
19     "AlarmasHMI".W1.%X0 := "ED"."Guardamotores nave 3";
20     "AlarmasHMI".W1.%X1 := "ED"."Guardamotores nave 4";
21     "AlarmasHMI".W1.%X2 := "ED"."Guardamotores nave 5";
22     "AlarmasHMI".W1.%X3 := "ED"."Guardamotores nave 6";
23     "AlarmasHMI".W1.%X4 := "ED"."Guardamotores nave 7";
24     "AlarmasHMI".W1.%X5 := "ED"."Guardamotores nave 8";
25     "AlarmasHMI".W1.%X6 := "ED"."Guardamotores nave 9";
26     "AlarmasHMI".W1.%X7 := "ED"."Guardamotores nave 10";
27     "AlarmasHMI".W1.%X8 := "ED"."Guardamotores nave 11";
28     "AlarmasHMI".W1.%X9 := "ED"."Guardamotores nave 12";
29     "AlarmasHMI".W1.%X10 := "ED"."PARO SEGURIDAD";
30     "AlarmasHMI".W1.%X11 := "Variables GEMMA".Fallo_EDM_CGA[1];
31     "AlarmasHMI".W1.%X12 := "Variables GEMMA".Fallo_EDM_CGA[2];
32     "AlarmasHMI".W1.%X13 := "Variables GEMMA".Fallo_EDM_CGA[3];
33     "AlarmasHMI".W1.%X14 := "Variables GEMMA".Fallo_EDM_CGA[4];
34     "AlarmasHMI".W1.%X15 := "Variables GEMMA".Fallo_EDM_CGB[1];
35
36
37     "AlarmasHMI".W2.%X0 := "Variables GEMMA".Fallo_EDM_CGB[2];
38     "AlarmasHMI".W2.%X1 := "Variables GEMMA".Fallo_EDM_CGB[3];
39     "AlarmasHMI".W2.%X2 := "Variables GEMMA".Fallo_EDM_CGB[4];
40     "AlarmasHMI".W2.%X3 := "Variables GEMMA".Fallo_EDM_DM[1];
41     "AlarmasHMI".W2.%X4 := "Variables GEMMA".Fallo_EDM_DM[2];
42     "AlarmasHMI".W2.%X5 := "Variables GEMMA".Fallo_EDM_DM[3];
43     "AlarmasHMI".W2.%X6 := "Variables GEMMA".Fallo_EDM_DM[4];
44     "AlarmasHMI".W2.%X7 := "Variables GEMMA".Fallo_EDM_DM[5];
45     "AlarmasHMI".W2.%X8 := "Variables GEMMA".Fallo_EDM_DM[6];
46     "AlarmasHMI".W2.%X9 := "Variables GEMMA".Fallo_EDM_CM[1];
47     "AlarmasHMI".W2.%X10 := "Variables GEMMA".Fallo_EDM_CM[2];
48     "AlarmasHMI".W2.%X11 := "Variables GEMMA".Fallo_EDM_CM[3];
49     "AlarmasHMI".W2.%X12 := "Variables GEMMA".Fallo_EDM_CM[4];
50     "AlarmasHMI".W2.%X13 := "Variables GEMMA".Fallo_EDM_CM[5];
51     "AlarmasHMI".W2.%X14 := "Variables GEMMA".Fallo_EDM_CM[6];
52     "AlarmasHMI".W2.%X15 := "Variables GEMMA".Fallo_EDM_CN[1];
53

```

```
54 "AlarmasHMI".W3.%X0 := "Variables GEMMA".Fallo_EDM_CN[2];
55 "AlarmasHMI".W3.%X1 := "Variables GEMMA".Fallo_EDM_CN[3];
56 "AlarmasHMI".W3.%X2 := "Variables GEMMA".Fallo_EDM_CN[4];
57 "AlarmasHMI".W3.%X3 := "Variables GEMMA".Fallo_EDM_CN[5];
58 "AlarmasHMI".W3.%X4 := "Variables GEMMA".Fallo_EDM_CN[6];
59 "AlarmasHMI".W3.%X5 := "Variables GEMMA".Fallo_EDM_CN[7];
60 "AlarmasHMI".W3.%X6 := "Variables GEMMA".Fallo_EDM_CN[8];
61 "AlarmasHMI".W3.%X7 := "Variables GEMMA".Fallo_EDM_CN[9];
62 "AlarmasHMI".W3.%X8 := "Variables GEMMA".Fallo_EDM_CN[10];
63 "AlarmasHMI".W3.%X9 := "Variables GEMMA".Fallo_EDM_CN[11];
64 "AlarmasHMI".W3.%X10 := "Variables GEMMA".Fallo_EDM_CN[12];
65 "AlarmasHMI".W3.%X11 := FALSE;
66 "AlarmasHMI".W3.%X12 := FALSE;
67 "AlarmasHMI".W3.%X13 := FALSE;
68 "AlarmasHMI".W3.%X14 := FALSE;
69 "AlarmasHMI".W3.%X15 := FALSE;
```

Código 63: Asignación de alarmas

B.10. Configuración de dispositivos

En esta sección de la programación del PLC se establecen las diferentes tarjetas de entradas y salidas, así como sus direcciones.

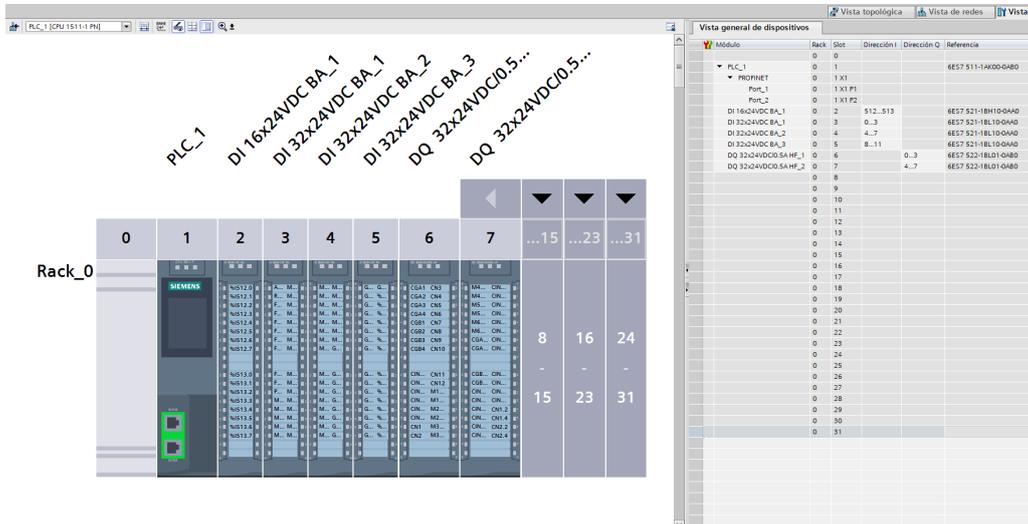


Figura 33: Distribución tarjetas

Se han necesitado un total de tres tarjetas de 32 bits de entradas digitales y dos tarjetas de 32 bits de salidas digitales. Se ha añadido una tarjeta adicional de 16 bits de entradas digitales, la que se encuentra en el Slot 1, que se utiliza para comunicar con Factory I/O.

B.11. Dispositivos y redes

El protocolo de comunicación seleccionado para conectar el HMI con el PLC es PROFINET. Ya que sólo van a comunicar estos dos elementos entre sí, la conexión es directa, mediante cable Ethernet. Con el software TIA Portal es muy sencillo configurar este protocolo de comunicación, ya que basta con seleccionar el puerto Ethernet de uno de los dispositivos y arrastrarlo al puerto Ethernet del segundo. El software comprueba que las direcciones IP sean compatibles, y ya está creada la conexión.

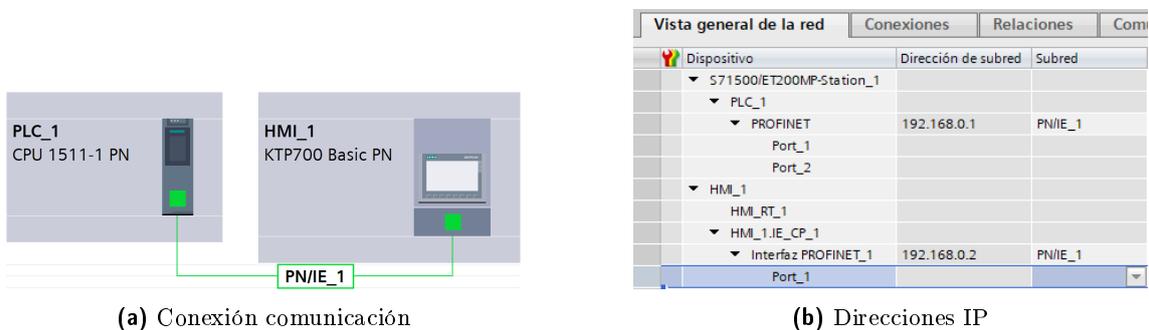


Figura 34: Comunicación PROFINET

B.12. Variables PLC

B.12.1. Tabla de variables estándar

Tabla 2: Tabla variables estándar

Name	Path	Data Type	Logical Address
System_Byte	Tabla de variables estándar	Byte	%MB1
FirstScan	Tabla de variables estándar	Bool	%M1.0
DiagStatusUpdate	Tabla de variables estándar	Bool	%M1.1
AlwaysTRUE	Tabla de variables estándar	Bool	%M1.2
AlwaysFALSE	Tabla de variables estándar	Bool	%M1.3
Clock_Byte	Tabla de variables estándar	Byte	%MB0
Clock_10Hz	Tabla de variables estándar	Bool	%M0.0
Clock_5Hz	Tabla de variables estándar	Bool	%M0.1
Clock_2.5Hz	Tabla de variables estándar	Bool	%M0.2
Clock_2Hz	Tabla de variables estándar	Bool	%M0.3
Clock_1.25Hz	Tabla de variables estándar	Bool	%M0.4
Clock_1Hz	Tabla de variables estándar	Bool	%M0.5
Clock_0.625Hz	Tabla de variables estándar	Bool	%M0.6
Clock_0.5Hz	Tabla de variables estándar	Bool	%M0.7
detecFlanco1	Tabla de variables estándar	Bool	%M3.1

B.12.2. DI

Tabla 3: Tabla entradas digitales

Name	Path	Data Type	Logical Address
Arranque	DI	Bool	%I0.0
Rearme	DI	Bool	%I0.1
Fotocélula A1	DI	Bool	%I0.2
Fotocélula A2	DI	Bool	%I0.3
Fotocélula A3	DI	Bool	%I0.4
Fotocélula A4	DI	Bool	%I0.5
Fotocélula B1	DI	Bool	%I0.6
Fotocélula B2	DI	Bool	%I0.7
Fotocélula B3	DI	Bool	%I1.0
Fotocélula B4	DI	Bool	%I1.1
PARO SEGURIDAD	DI	Bool	%I1.2
Marcha cinta A pt1	DI	Bool	%I1.3

Marcha cinta A pt2	DI	Bool	%I1.4
Marcha cinta B pt1	DI	Bool	%I1.5
Marcha cinta B pt2	DI	Bool	%I1.6
Marcha cinta A pt3	DI	Bool	%I1.7
Marcha cinta A pt4	DI	Bool	%I2.0
Marcha cinta B pt3	DI	Bool	%I2.1
Marcha cinta B pt4	DI	Bool	%I2.2
Mesa 1 FCA	DI	Bool	%I2.3
Mesa 1 FC0	DI	Bool	%I2.4
Mesa 1 FCB	DI	Bool	%I2.5
Mesa 2 FCA	DI	Bool	%I2.6
Mesa 2 FC0	DI	Bool	%I2.7
Mesa 2 FCB	DI	Bool	%I3.0
Mesa 3 FCA	DI	Bool	%I3.1
Mesa 3 FC0	DI	Bool	%I3.2
Mesa 3 FCB	DI	Bool	%I3.3
Mesa 4 FCA	DI	Bool	%I3.4
Mesa 4 FC0	DI	Bool	%I3.5
Mesa 4 FCB	DI	Bool	%I3.6
Mesa 5 FCA	DI	Bool	%I3.7
Mesa 5 FC0	DI	Bool	%I4.0
Mesa 5 FCB	DI	Bool	%I4.1
Mesa 6 FCA	DI	Bool	%I4.2
Mesa 6 FC0	DI	Bool	%I4.3
Mesa 6 FCB	DI	Bool	%I4.4
Marcha cinta mesa 1	DI	Bool	%I4.5
Marcha cinta mesa 2	DI	Bool	%I4.6
Marcha cinta mesa 3	DI	Bool	%I4.7
Marcha cinta mesa 4	DI	Bool	%I5.0
Marcha cinta mesa 5	DI	Bool	%I5.1
Marcha cinta mesa 6	DI	Bool	%I5.2
Marcha nave 1	DI	Bool	%I5.3
Marcha nave 2	DI	Bool	%I5.4
Marcha nave 3	DI	Bool	%I5.5
Marcha nave 4	DI	Bool	%I5.6
Marcha nave 5	DI	Bool	%I5.7
Marcha nave 6	DI	Bool	%I6.0
Marcha nave 7	DI	Bool	%I6.1

Marcha nave 8	DI	Bool	%I6.2
Marcha nave 9	DI	Bool	%I6.3
Marcha nave 10	DI	Bool	%I6.4
Marcha nave 11	DI	Bool	%I6.5
Marcha nave 12	DI	Bool	%I6.6
Guardamotores cinta A pt1	DI	Bool	%I6.7
Guardamotores cinta A pt2	DI	Bool	%I7.0
Guardamotores cinta B pt1	DI	Bool	%I7.1
Guardamotores cinta B pt2	DI	Bool	%I7.2
Guardamotores cinta A pt3	DI	Bool	%I7.3
Guardamotores cinta A pt4	DI	Bool	%I7.4
Guardamotores cinta B pt3	DI	Bool	%I7.5
Guardamotores cinta B pt4	DI	Bool	%I7.6
Guardamotores mesa 1	DI	Bool	%I7.7
Guardamotores mesa 2	DI	Bool	%I8.0
Guardamotores mesa 3	DI	Bool	%I8.1
Guardamotores mesa 4	DI	Bool	%I8.2
Guardamotores mesa 5	DI	Bool	%I8.3
Guardamotores mesa 6	DI	Bool	%I8.4
Guardamotores nave 1	DI	Bool	%I8.5
Guardamotores nave 2	DI	Bool	%I8.6
Guardamotores nave 3	DI	Bool	%I8.7
Guardamotores nave 4	DI	Bool	%I9.0
Guardamotores nave 5	DI	Bool	%I9.1
Guardamotores nave 6	DI	Bool	%I9.2
Guardamotores nave 7	DI	Bool	%I9.3
Guardamotores nave 8	DI	Bool	%I9.4
Guardamotores nave 9	DI	Bool	%I9.5
Guardamotores nave 10	DI	Bool	%I9.6
Guardamotores nave 11	DI	Bool	%I9.7
Guardamotores nave 12	DI	Bool	%I10.0

B.12.3. DQ

Tabla 4: Tabla de salidas digitales

Name	Path	Data Type	Logical Address
CGA1	DQ	Bool	%Q0.0
CGA2	DQ	Bool	%Q0.1
CGA3	DQ	Bool	%Q0.2
CGA4	DQ	Bool	%Q0.3
CGB1	DQ	Bool	%Q0.4
CGB2	DQ	Bool	%Q0.5
CGB3	DQ	Bool	%Q0.6
CGB4	DQ	Bool	%Q0.7
CINTA M1	DQ	Bool	%Q1.0
CINTA M2	DQ	Bool	%Q1.1
CINTA M3	DQ	Bool	%Q1.2
CINTA M4	DQ	Bool	%Q1.3
CINTA M5	DQ	Bool	%Q1.4
CINTA M6	DQ	Bool	%Q1.5
CN1	DQ	Bool	%Q1.6
CN2	DQ	Bool	%Q1.7
CN3	DQ	Bool	%Q2.0
CN4	DQ	Bool	%Q2.1
CN5	DQ	Bool	%Q2.2
CN6	DQ	Bool	%Q2.3
CN7	DQ	Bool	%Q2.4
CN8	DQ	Bool	%Q2.5
CN9	DQ	Bool	%Q2.6
CN10	DQ	Bool	%Q2.7
CN11	DQ	Bool	%Q3.0
CN12	DQ	Bool	%Q3.1
M1 AVANZAR	DQ	Bool	%Q3.2
M1 RETROCEDER	DQ	Bool	%Q3.3
M2 AVANZAR	DQ	Bool	%Q3.4
M2 RETROCEDER	DQ	Bool	%Q3.5
M3 AVANZAR	DQ	Bool	%Q3.6
M3 RETROCEDER	DQ	Bool	%Q3.7
M4 AVANZAR	DQ	Bool	%Q4.0

M4 RETROCEDER	DQ	Bool	%Q4.1
M5 AVANZAR	DQ	Bool	%Q4.2
M5 RETROCEDER	DQ	Bool	%Q4.3
M6 AVANZAR	DQ	Bool	%Q4.4
M6 RETROCEDER	DQ	Bool	%Q4.5
CGA1 PT2	DQ	Bool	%Q4.6
CGA4 PT2	DQ	Bool	%Q4.7
CGB1 PT2	DQ	Bool	%Q5.0
CGB4 PT2	DQ	Bool	%Q5.1
CINTA M1.1	DQ	Bool	%Q5.2
CINTA M1.2	DQ	Bool	%Q5.3
CINTA M1.3	DQ	Bool	%Q5.4
CINTA M2.1	DQ	Bool	%Q5.5
CINTA M2.2	DQ	Bool	%Q5.6
CINTA M2.3	DQ	Bool	%Q5.7
CINTA M3.1	DQ	Bool	%Q6.0
CINTA M3.2	DQ	Bool	%Q6.1
CINTA M3.3	DQ	Bool	%Q6.2
CINTA M4.1	DQ	Bool	%Q6.3
CINTA M4.2	DQ	Bool	%Q6.4
CINTA M4.3	DQ	Bool	%Q6.5
CINTA M5.1	DQ	Bool	%Q6.6
CINTA M5.2	DQ	Bool	%Q6.7
CINTA M5.3	DQ	Bool	%Q7.0
CINTA M6.1	DQ	Bool	%Q7.1
CINTA M6.2	DQ	Bool	%Q7.2
CINTA M6.3	DQ	Bool	%Q7.3
CN1.2	DQ	Bool	%Q7.4
CN1.4	DQ	Bool	%Q7.5
CN2.2	DQ	Bool	%Q7.6
CN2.4	DQ	Bool	%Q7.7
CN3.2	DQ	Bool	%Q8.0
CN3.4	DQ	Bool	%Q8.1
CN4.2	DQ	Bool	%Q8.2
CN4.4	DQ	Bool	%Q8.3
CN5.2	DQ	Bool	%Q8.4
CN6.2	DQ	Bool	%Q8.5
CN7.2	DQ	Bool	%Q8.6

CN8.2	DQ	Bool	%Q8.7
CN9.2	DQ	Bool	%Q9.0
CN10.2	DQ	Bool	%Q9.1
CN11.2	DQ	Bool	%Q9.2
CN12.2	DQ	Bool	%Q9.3
M1 AVANZAR.1	DQ	Bool	%Q9.4
M1.1 AVANZAR	DQ	Bool	%Q9.5
M2 AVANZAR.1	DQ	Bool	%Q9.6
M2.1 AVANZAR	DQ	Bool	%Q9.7
M3 AVANZAR.1	DQ	Bool	%Q10.0
M3.1 AVANZAR	DQ	Bool	%Q10.1
M4 AVANZAR.1	DQ	Bool	%Q10.2
M4.1 AVANZAR	DQ	Bool	%Q10.3
M5 AVANZAR.1	DQ	Bool	%Q10.4
M5.1 AVANZAR	DQ	Bool	%Q10.5
M6 AVANZAR.1	DQ	Bool	%Q10.6
M6.1 AVANZAR	DQ	Bool	%Q10.7
M1 RETROCEDER.1	DQ	Bool	%Q11.0
M1.1 RETROCEDER	DQ	Bool	%Q11.1
M2 RETROCEDER.1	DQ	Bool	%Q11.2
M2.1 RETROCEDER	DQ	Bool	%Q11.3
M3 RETROCEDER.1	DQ	Bool	%Q11.4
M3.1 RETROCEDER	DQ	Bool	%Q11.5
M4 RETROCEDER.1	DQ	Bool	%Q11.6
M4.1 RETROCEDER	DQ	Bool	%Q11.7
M5 RETROCEDER.1	DQ	Bool	%Q12.0
M5.1 RETROCEDER	DQ	Bool	%Q12.1
M6 RETROCEDER.1	DQ	Bool	%Q12.2
M6.1 RETROCEDER	DQ	Bool	%Q12.3
NAVE 1	DQ	Bool	%Q12.4
NAVE 2	DQ	Bool	%Q12.5
NAVE 3	DQ	Bool	%Q12.6
NAVE 4	DQ	Bool	%Q12.7
NAVE 5	DQ	Bool	%Q13.0
NAVE 6	DQ	Bool	%Q13.1
NAVE 7	DQ	Bool	%Q13.2
NAVE 8	DQ	Bool	%Q13.3
NAVE 9	DQ	Bool	%Q13.4

NAVE 10	DQ	Bool	%Q13.5
NAVE 11	DQ	Bool	%Q13.6
NAVE 12	DQ	Bool	%Q13.7
PART TO EMIT 1	DQ	Word	%QW30
PART TO EMIT 2	DQ	Word	%QW32
PART TO EMIT 3	DQ	Word	%QW34
PART TO EMIT 4	DQ	Word	%QW36
PART TO EMIT 5	DQ	Word	%QW38
PART TO EMIT 6	DQ	Word	%QW40
PART TO EMIT 7	DQ	Word	%QW42
PART TO EMIT 8	DQ	Word	%QW44
PART TO EMIT 9	DQ	Word	%QW46
PART TO EMIT 10	DQ	Word	%QW48
PART TO EMIT 11	DQ	Word	%QW50
PART TO EMIT 12	DQ	Word	%QW52

Apéndice C

Programación HMI

En este apartado se va a explicar cómo se ha realizado la programación del HMI, sólo se van a explicar los elementos dinámicos, es decir, los que tienen animaciones o generan eventos.

C.1. Administración de imágenes: Plantillas

Para la programación del HMI, se ha creado una plantilla, llamada Plantilla_1, como base para todas las pantallas. En ella se definen las barras inferior y superior, tal y como se muestra en la Figura 35.



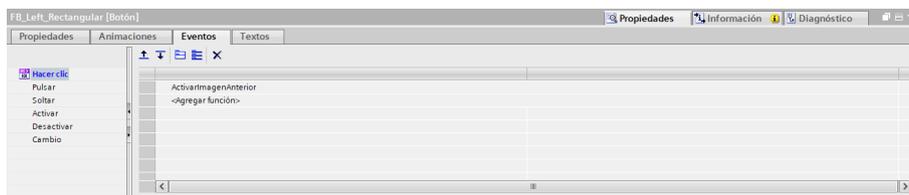
Figura 35: Plantilla_1

En la barra superior se encuentran dos botones:

- El que tiene una flecha hacia la izquierda, es un botón cogido de la librería global que se encuentra de forma predeterminada en TIA Portal, por lo que los gráficos "si pulsado" y "si no pulsado" se añaden automáticamente al botón. Como animación, su visibilidad depende de la variable interna del HMI "Número_imagen_variable", la cual cambia cuando se abre una imagen, adoptando el valor que cada pantalla le escribe. Esta variable, cuando su valor es "1" significa que se está en la pantalla principal, y por tanto no hay imagen anterior. Como evento, al hacer clic activa la función "ActivarImagenAnterior".



(a) Animación visibilidad

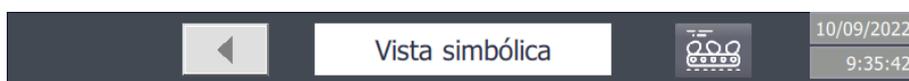


(b) Evento activación imagen

Figura 36: Características dinámicas FB_Left_Rectangular

- El Botón_4, es el que tiene el dibujo de unos huevos sobre una cinta transportadora (imagen: Flaticon). En este caso se ha decidido no poner una imagen para "si no pulsado". Como el botón anterior, también tiene un evento al hacer clic, pero en este caso la función es "ActivarImagen", al cual se le pasa el nombre de la imagen deseada que es "Vista simbólica".

El recuadro blanco de la barra superior es un campo de E/S simbólico. Está configurado para mostrar un texto diferente según la variable "Número_imagen_variable", se puede encontrar más información sobre el texto mostrado en la [Tabla 6](#).



(a) Barra superior pantalla vista simbólica



(b) Barra inferior

Figura 37: Barras comunes a todas las pantallas

En la barra inferior se encuentran tres botones predefinidos por TIA Portal, los incluye automáticamente al crear el HMI.

- El primero activa la imagen "Imagen raíz", independientemente de la imagen en la que te encuentres.
- El segundo activa la imagen "ALARMAS_Activas".
- El tercero cierra el runtime del HMI.

El fondo de la plantilla es del color RGB(181; 182; 181).

C.2. Imagen raíz

Esta imagen se puede decir que es la más compleja de todas, ya que han hecho falta muchas animaciones y ha sido necesario administrar las diferentes capas.

Por un lado, se encuentran los rectángulos superiores, es decir, los que cuando son visibles tapan los botones que tienen debajo. Hay dos, los cuales son texto, que dan información sobre cuando se podrán volver a utilizar los botones, en fondo blanco.

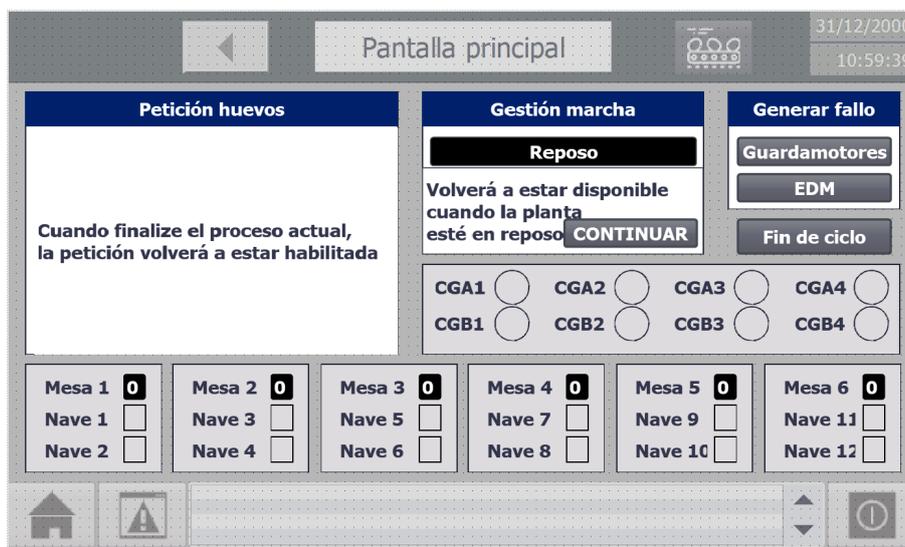


Figura 38: Imagen raíz con rectángulos superiores visibles

El rectángulo sobre "Petición huevos" tiene una animación de visibilidad, estará visible siempre que "Variables GEMMA.pet_available" sea "0". Esta variable viene del PLC, para poder enlazarla lo que se ha hecho es, una vez configurada la comunicación mediante PROFINET, del bloque de datos correspondiente se ha arrastrado hasta la casilla de variable. Se ha hecho así con todas las variables necesarias. Como se puede observar, reutilizar variables entre diferentes componentes es muy sencillo.

El rectángulo ubicado en "Gestión marcha" tiene configurada la visibilidad siempre que "Variables GEMMA.A1" sea "0".

Debajo de estos rectángulos se encuentran tanto interruptores como botones. Los botones tienen un color de fondo uniforme y siempre muestran el mismo texto; en cambio, los interruptores pueden cambiar tanto su texto como sus colores, se diferencian fácilmente por su forma con el texto "ON/OFF", sólo uno de los dos es visible y se cambia al otro pulsando sobre la parte gris. El que pone "MAN/AUTO" también es un interruptor, pero tiene texto diferente.

Mientras que a los botones hay que asignarles Eventos al Hacer clic, Pulsar o Soltar, los interruptores sólo necesitan que se les asigne una variable booleana. Con esta variable, se especifica qué texto se desea que aparezca cuando la variable es "0" y cuando es "1". De esta forma, si la variable cambia de valor en el PLC, será visible el cambio en el HMI, mientras que, si pulsamos al rectángulo gris del interruptor, se está cambiando el valor de la variable en el HMI.

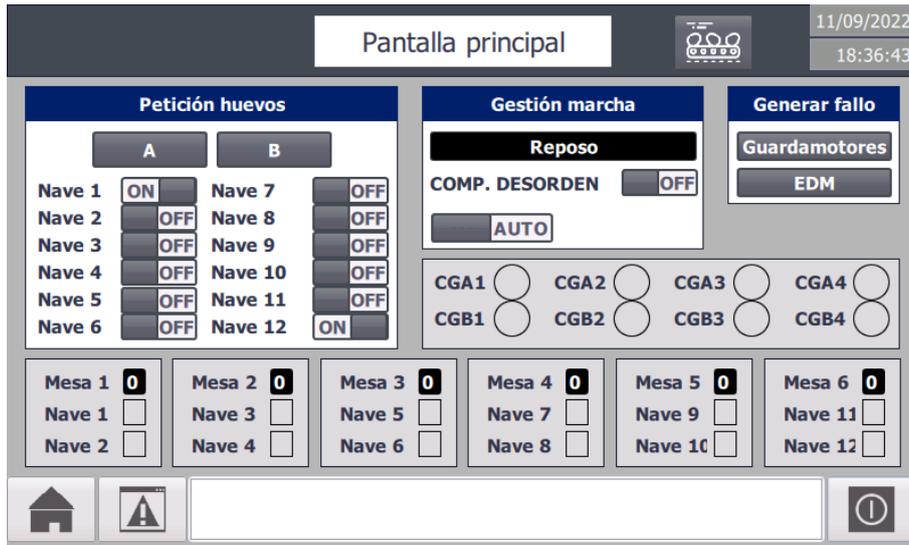


Figura 39: HMI con interruptores ON/OFF

"A" y "B" también son interruptores, pero en este caso se les ha asignado una animación de apariencia, en la cual tienen que cambiar el color del texto dependiendo de la variable "ED.Petición huevos cinta X", siendo X "A" o "B", variable la cual estos mismos interruptores modifican.

Los interruptores de las naves están ligados a las variables "ED.Petición huevos nave Y", siendo Y el número de nave.

El interruptor precedido por COMP. DESORDEN está vinculado a la variable "Variables GEMMA.Manual". Mientras que el interruptor con el texto "MAN/AUTO" está enlazado a la variable "Variables GEMMA.SecuenciaManual". Como se puede observar en la Figura 38, en la gestión de marcha también hay un botón con el texto CONTINUAR, el cual se ha programado una visualización para que sólo sea visible cuando "Variables GEMMA.F5" sea "1". Este botón también tiene dos eventos, uno al Pulsar y otro al Soltar el botón, el primero activa el bit de la variable "Variables_HMI.ContinuarPantalla_HMI", mientras que el segundo lo desactiva.

En el apartado de generación de fallo se hayan dos botones que activan sendas pantallas.

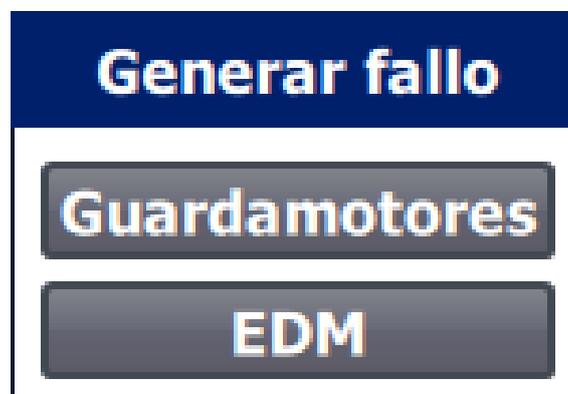


Figura 40: Generación fallo

En la Figura 38 se observa un botón con el texto "Fin ciclo", el cual sólo está visible cuando

"Variables GEMMA.F1" sea "1" y tiene programado un evento del tipo "Hacer clic" que activa el bit de la variable "Variables GEMMA.FinCiclo".



Figura 41: Pantalla principal gris

Los rectángulos grises al lado de las naves o los círculos grises al lado de las cintas generales tienen programada la misma animación de apariencia según la variable "Variables_HMI.Estado_X_[Y]", siendo X "CGA", "CGB" o "CN" y siendo Y el número. Esta variable siempre es un entero, y según su valor, se aplican las características de la siguiente figura:

Apariencia

Variable

Nombre: `Variables_HMI.Estado_CN_HMI[1]`

Dirección:

Rango	Color de fondo	Color Borde	Parpadeo
0	222; 219; ...	0; 0; 0	No
1	0; 255; 0	0; 0; 0	No
2	255; 0; 0	255; 255; 0	Sí

Figura 42: Animación apariencia

Los rectángulos negros, en cambio, son campos de E/S simbólica, que muestran una letra distinta según la variable "Variables_HMI.Estado_M_HMI[X]", siendo X el número de mesa. Esta variable también se utiliza para dinamizar la apariencia, su configuración es parecida a la de la Figura 42, pero esta variable está programada para que indique también la dirección de la mesa, que se representa con colores diferentes.

Apariencia

Variable

Nombre: `Variables_HMI.Estado_M_HMI[1]`

Dirección:

Rango	Color de fondo	Color de primer pl...	Parpadeo
0	0; 0; 0	255; 255; ...	No
1	0; 255; 0	255; 255; 255	No
2	255; 0; 255	255; 255; 255	No
3	255; 0; 0	255; 255; 0	Sí

Figura 43: Animación apariencia mesas

C.3. E_ConfMarcha y E_Guardamotores

Se ha decidido agrupar estas dos pantallas ya que la distribución es la misma, los botones son los mismos, sólo cambian las variables que hay seis botones más. Estas pantallas están divididas por las partes de las cintas generales, en cada parte encontramos dos columnas, en la primera encontramos texto, donde se indica la cinta; y a su derecha se encuentra uno o dos botones. En el caso de la confirmación de marcha de las mesas, hay dos botones, ya que en el primero se cambia la confirmación de marcha de la dirección de la mesa y en el segundo, de la cinta de la mesa.



Figura 44: Pantalla fallo guardamotores

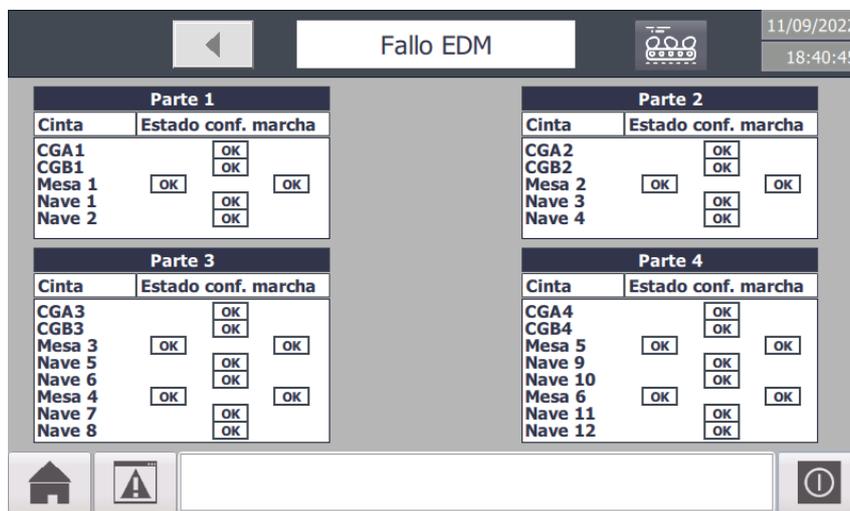


Figura 45: Pantalla fallo EDM

Estos botones tienen una animación de apariencia según la variable "Variable_HMI.Fallo_EDM_X_HMI[Y]", siendo X "CGA", "CGB", "CN", "DM", "CM"; en el caso de la pantalla de fallo EDM y según la variable "ED.Guardamotores X Y" en el caso de fallo guardamotor. El fondo cambia entre color blanco cuando la variable es 0 y verde, cuando la variable es 1. Además, estos mismos botones son los que activan y desactivan estos bits con un evento la "Hacer clic" Invertirbit.

C.4. Vista simbólica

Esta pantalla está formada por veintiséis Campos de E/S Gráfico, cada uno representa un elemento, cinta general, cinta nave o mesa. Estos campos cambian de imagen según "Variables_HMI.Estado_X_HMI[Y]", siendo X "CN", "CGA", "CGB" o "M" e Y el número, cada variable tiene su lista de gráficos propia.

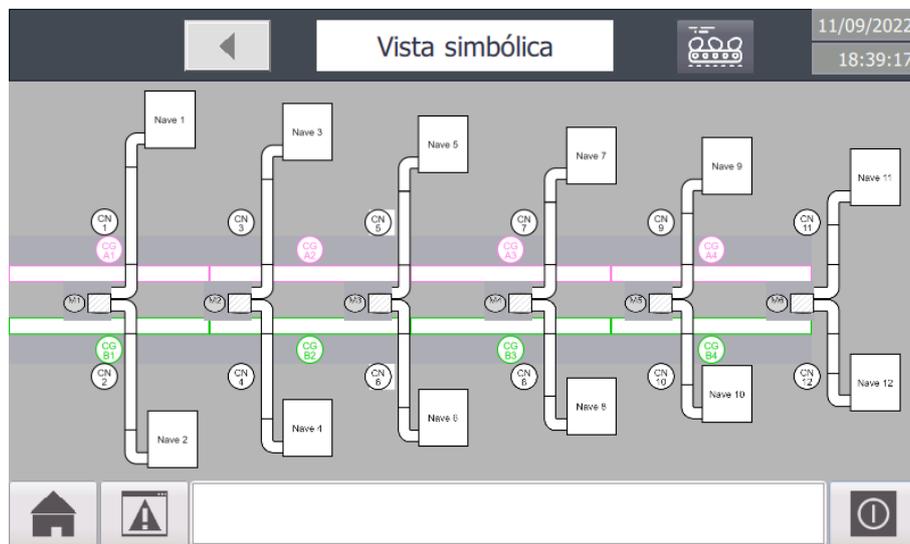


Figura 46: Pantalla vista simbólica

Cuando todos los motores están parados, los gráficos que aparecen son los "_reposito.drawio", el gráfico "_marcha.drawio" es el mismo, pero con fondo verde y el gráfico "_error1.drawio" es el mismo, pero con fondo rojo. En [Tabla 7](#) se puede encontrar más información.

Entradas de la lista de gráficos				
...	Predet..	Valor ▲	Nombre del gráfico	Gráfico
	<input type="radio"/>	0	CN1_reposo.drawio	
	<input type="radio"/>	1	CN1_marcha.drawio	
	<input type="radio"/>	2	CN1_error1.drawio	
		<Agregar>		

Figura 47: Lista gráficos CN1

C.5. ALARMAS_Activas y ALARMAS_Historicas

Ambas pantallas contienen un visor de avisos. En el caso de ALARMAS_Activas este visor muestra los estados de aviso actuales, mientras que ALARMAS_Historicas muestra el búfer de avisos.



Figura 48: Configuración visor de avisos

La pantalla ALARMAS_Activas cuenta, además con un botón "Alarmas Históricas" para llevarte a dicha pantalla; y ALARMAS_Historicas cuenta con un botón "Borrar Histórico" que activa el evento BorrarBúferDeAvisos con número de categoría "Todos los avisos".



(a) Pantalla alarmas activas



(b) Pantalla alarmas históricas

Figura 49: Pantallas alarmas

C.6. Variables HMI

Tabla 5: Tabla variables HMI

Name	DataType	Access Method
Número_imagen_variable	UInt	
ED.Petición huevos nave 1	Bool	Symbolic access
ED.Petición huevos nave 2	Bool	Symbolic access
ED.Petición huevos nave 3	Bool	Symbolic access
ED.Petición huevos nave 4	Bool	Symbolic access
ED.Petición huevos nave 5	Bool	Symbolic access
ED.Petición huevos nave 6	Bool	Symbolic access
Variables_HMI.Petición huevos nave 9	Bool	Symbolic access
ED.Petición huevos nave 10	Bool	Symbolic access
ED.Petición huevos nave 11	Bool	Symbolic access
ED.Petición huevos nave 12	Bool	Symbolic access
ED.Petición huevos nave 7	Bool	Symbolic access
ED.Petición huevos nave 8	Bool	Symbolic access
ED.Petición huevos cinta A	Bool	Symbolic access
ED.Petición huevos cinta B	Bool	Symbolic access
Variables GEMMA.F2	Bool	Symbolic access
Variables GEMMA.PeticionNueva	Bool	Symbolic access
Variables GEMMA.SecuenciaManual	Bool	Symbolic access
Variables GEMMA.cont	Bool	Symbolic access
Variables_HMI.ContinuarPantalla_HMI	Bool	Symbolic access
Variables GEMMA.F5	Bool	Symbolic access
Variables GEMMA.A1	Bool	Symbolic access
Variables_HMI.EstadoAct_HMI	Int	Symbolic access
Variables GEMMA.Manual	Bool	Symbolic access
Variables GEMMA.PosMesas[1]	Char	Symbolic access
CN1	Bool	Symbolic access
CN2	Bool	Symbolic access
CN3	Bool	Symbolic access
CN4	Bool	Symbolic access
CN5	Bool	Symbolic access
CN6	Bool	Symbolic access
CN7	Bool	Symbolic access
CN8	Bool	Symbolic access

CN9	Bool	Symbolic access
CN10	Bool	Symbolic access
CN11	Bool	Symbolic access
CN12	Bool	Symbolic access
Variables_HMI.Estado_M_HMI[1]	Int	Symbolic access
Variables_HMI.Estado_M_HMI[2]	Int	Symbolic access
Variables_HMI.Estado_M_HMI[3]	Int	Symbolic access
Variables_HMI.Estado_M_HMI[4]	Int	Symbolic access
Variables_HMI.Estado_M_HMI[5]	Int	Symbolic access
Variables_HMI.Estado_M_HMI[6]	Int	Symbolic access
CGA1	Bool	Symbolic access
CGA2	Bool	Symbolic access
CGA3	Bool	Symbolic access
CGA4	Bool	Symbolic access
CGB1	Bool	Symbolic access
CGB2	Bool	Symbolic access
CGB3	Bool	Symbolic access
CGB4	Bool	Symbolic access
ED.Marcha cinta A pt3	Bool	Symbolic access
ED.Guardamotores cinta A pt1	Bool	Symbolic access
ED.Guardamotores cinta B pt1	Bool	Symbolic access
ED.Guardamotores mesa 1	Bool	Symbolic access
ED.Guardamotores nave 1	Bool	Symbolic access
ED.Guardamotores nave 2	Bool	Symbolic access
ED.Guardamotores cinta A pt2	Bool	Symbolic access
ED.Guardamotores cinta B pt2	Bool	Symbolic access
ED.Guardamotores mesa 2	Bool	Symbolic access
ED.Guardamotores nave 3	Bool	Symbolic access
ED.Guardamotores nave 4	Bool	Symbolic access
ED.Guardamotores cinta A pt3	Bool	Symbolic access
ED.Guardamotores cinta B pt3	Bool	Symbolic access
ED.Guardamotores cinta A pt4	Bool	Symbolic access
ED.Guardamotores cinta B pt4	Bool	Symbolic access
ED.Guardamotores mesa 3	Bool	Symbolic access
ED.Guardamotores mesa 4	Bool	Symbolic access
ED.Guardamotores mesa 5	Bool	Symbolic access
ED.Guardamotores mesa 6	Bool	Symbolic access
ED.Guardamotores nave 5	Bool	Symbolic access

ED.Guardamotores nave 6	Bool	Symbolic access
ED.Guardamotores nave 7	Bool	Symbolic access
ED.Guardamotores nave 8	Bool	Symbolic access
ED.Guardamotores nave 9	Bool	Symbolic access
ED.Guardamotores nave 10	Bool	Symbolic access
ED.Guardamotores nave 11	Bool	Symbolic access
ED.Guardamotores nave 12	Bool	Symbolic access
AlarmasHMI.W0	Word	Symbolic access
AlarmasHMI.W1	Word	Symbolic access
AlarmasHMI.W2	Word	Symbolic access
Variables_HMI.Estado_CN_HMI[1]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[2]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[3]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[4]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[5]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[6]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[7]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[8]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[9]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[10]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[11]	Int	Symbolic access
Variables_HMI.Estado_CN_HMI[12]	Int	Symbolic access
Variables_HMI.Estado_CGA_HMI[1]	Int	Symbolic access
Variables_HMI.Estado_CGA_HMI[2]	Int	Symbolic access
Variables_HMI.Estado_CGA_HMI[3]	Int	Symbolic access
Variables_HMI.Estado_CGA_HMI[4]	Int	Symbolic access
Variables_HMI.Estado_CGB_HMI[1]	Int	Symbolic access
Variables_HMI.Estado_CGB_HMI[2]	Int	Symbolic access
Variables_HMI.Estado_CGB_HMI[3]	Int	Symbolic access
Variables_HMI.Estado_CGB_HMI[4]	Int	Symbolic access
Variables GEMMA.pet_available	Bool	Symbolic access
AlarmasHMI.W3	Word	Symbolic access
AlarmasHMI.W4	Word	Symbolic access
Variables GEMMA.FinCiclo	Bool	Symbolic access
Variables GEMMA.Fallo_EDM_CGA[1]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CGA_HMI[1]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CGA_HMI[2]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CGA_HMI[3]	Bool	Symbolic access

Variables_HMI.Fallo_EDM_CGA_HMI[4]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CGB_HMI[1]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CGB_HMI[2]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CGB_HMI[3]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CGB_HMI[4]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_DM_HMI[1]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_DM_HMI[2]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_DM_HMI[3]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_DM_HMI[4]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_DM_HMI[5]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_DM_HMI[6]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CM_HMI[1]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CM_HMI[2]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CM_HMI[3]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CM_HMI[4]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CM_HMI[5]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CM_HMI[6]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[1]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[2]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[3]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[4]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[5]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[6]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[7]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[8]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[9]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[10]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[11]	Bool	Symbolic access
Variables_HMI.Fallo_EDM_CN_HMI[12]	Bool	Symbolic access
Variables GEMMA.F1	Bool	Symbolic access
F1_DB.F11	Bool	Symbolic access
Variables GEMMA.Simulacion_activa	Bool	Symbolic access

C.7. Lista de textos y gráficos

Tabla 6: Lista de textos HMI

Parent	Value	Text [es-ES]
TextList_ScreenNames	1	Pantalla principal
TextList_ScreenNames	2	Fallo guardamotores
TextList_ScreenNames	3	Vista simbólica
TextList_ScreenNames	4	Fallo EDM
TextList_ScreenNames	5	Alarmas Activas
TextList_ScreenNames	6	Alarmas Historicas
EstadoPlanta	0	Reposo
EstadoPlanta	1	Preparación
EstadoPlanta	2	Marcha
EstadoPlanta	3	Finalizando
EstadoPlanta	4	Comp. orden
EstadoPlanta	5	Comp. desorden
EstadoPlanta	6	EMERGENCIA
EstadoPlanta	7	Espera C.I.
EstadoPlanta	8	Error
EstadoPlanta	9	error
EstadoPlanta	10	Error
MovimientoMesas	0	0
MovimientoMesas	1	A
MovimientoMesas	2	R
MovimientoMesas	3	E
E_Guardamotores	0	OK
E_Guardamotores	1	NOK

Tabla 7: Lista de textos HMI

Parent	Value	Image
CGA1	0	CGA1_reposo.drawio
CGA1	1	CGA1_marcha.drawio
CGA1	2	CGA1_error1.drawio
CGA2	0	CGA2_reposo.drawio
CGA2	1	CGA2_marcha.drawio
CGA2	2	CGA2_error1.drawio

CGA3	0	CGA3_reposo.drawio
CGA3	1	CGA3_marcha.drawio
CGA3	2	CGA3_error1.drawio
CGA4	0	CGA4_reposo.drawio
CGA4	1	CGA4_marcha.drawio
CGA4	2	CGA4_error1.drawio
CGB1	0	CGB1_reposo.drawio
CGB1	1	CGB1_marcha.drawio
CGB1	2	CGB1_error1.drawio
CGB2	0	CGB2_reposo.drawio
CGB2	1	CGB2_marcha.drawio
CGB2	2	CGB2_error1.drawio
CGB3	0	CGB3_reposo.drawio
CGB3	1	CGB3_marcha.drawio
CGB3	2	CGB3_error1.drawio
CGB4	0	CGB4_reposo.drawio
CGB4	1	CGB4_marcha.drawio
CGB4	2	CGB4_error1.drawio
CN1	0	CN1_reposo.drawio
CN1	1	CN1_marcha.drawio
CN1	2	CN1_error1.drawio
CN2	0	CN2_reposo.drawio
CN2	1	CN2_marcha.drawio
CN2	2	CN2_error1.drawio
CN3	0	CN3_reposo.drawio
CN3	1	CN3_marcha.drawio
CN3	2	CN3_error1.drawio
CN4	0	CN4_reposo.drawio
CN4	1	CN4_marcha.drawio
CN4	2	CN4_error1.drawio
CN5	0	CN5_reposo.drawio
CN5	1	CN5_marcha.drawio
CN5	2	CN5_error1.drawio
CN6	0	CN6_reposo.drawio
CN6	1	CN6_marcha.drawio
CN6	2	CN6_error1.drawio
CN7	0	CN7_reposo.drawio
CN7	1	CN7_marcha.drawio

CN7	2	CN7_error1.drawio
CN8	0	CN8_reposo.drawio
CN8	1	CN8_marcha.drawio
CN8	2	CN8_error1.drawio
CN9	0	CN9_reposo.drawio
CN9	1	CN9_marcha.drawio
CN9	2	CN9_error1.drawio
CN10	0	CN10_reposo.drawio
CN10	1	CN10_marcha.drawio
CN10	2	CN10_error1.drawio
CN11	0	CN11_reposo.drawio
CN11	1	CN11_marcha.drawio
CN11	2	CN11_error1.drawio
CN12	0	CN12_reposo.drawio
CN12	1	CN12_marcha.drawio
CN12	2	CN12_error1.drawio

C.8. Avisos HMI

Tabla 8: Lista de Avisos HMI

ID	Alarm text [es-ES]	Trigger tag	Trigger bit
1	Guardamotors cinta A pt1	"AlarmasHMI.W0"	0
2	Guardamotors cinta A pt2	"AlarmasHMI.W0"	1
3	Guardamotors cinta A pt3	"AlarmasHMI.W0"	2
4	Guardamotors cinta A pt4	"AlarmasHMI.W0"	3
5	Guardamotors cinta B pt1	"AlarmasHMI.W0"	4
6	Guardamotors cinta B pt2	"AlarmasHMI.W0"	5
7	Guardamotors cinta B pt3	"AlarmasHMI.W0"	6
8	Guardamotors cinta B pt4	"AlarmasHMI.W0"	7
9	Guardamotors mesa 1	"AlarmasHMI.W0"	8
10	Guardamotors mesa 2	"AlarmasHMI.W0"	9
11	Guardamotors mesa 3	"AlarmasHMI.W0"	10
12	Guardamotors mesa 4	"AlarmasHMI.W0"	11
13	Guardamotors mesa 5	"AlarmasHMI.W0"	12
14	Guardamotors mesa 6	"AlarmasHMI.W0"	13
15	Guardamotors nave 1	"AlarmasHMI.W0"	14
16	Guardamotors nave 2	"AlarmasHMI.W0"	15
17	Guardamotors nave 3	"AlarmasHMI.W1"	0
18	Guardamotors nave 4	"AlarmasHMI.W1"	1
19	Guardamotors nave 5	"AlarmasHMI.W1"	2
20	Guardamotors nave 6	"AlarmasHMI.W1"	3
21	Guardamotors nave 7	"AlarmasHMI.W1"	4
22	Guardamotors nave 8	"AlarmasHMI.W1"	5
23	Guardamotors nave 9	"AlarmasHMI.W1"	6
24	Guardamotors nave 10	"AlarmasHMI.W1"	7
25	Guardamotors nave 11	"AlarmasHMI.W1"	8
26	Guardamotors nave 12	"AlarmasHMI.W1"	9
27	PARO SEGURIDAD	"AlarmasHMI.W1"	10
28	Fallo EDM CGA1	"AlarmasHMI.W1"	11
29	Fallo EDM CGA2	"AlarmasHMI.W1"	12
30	Fallo EDM CGA3	"AlarmasHMI.W1"	13
31	Fallo EDM CGA4	"AlarmasHMI.W1"	14
32	Fallo EDM CGB1	"AlarmasHMI.W1"	15
33	Fallo EDM CGB2	"AlarmasHMI.W2"	0

34	Fallo EDM CGB3	"AlarmasHMI.W2"	1
35	Fallo EDM CGB4	"AlarmasHMI.W2"	2
36	Fallo EDM dirección mesa 1	"AlarmasHMI.W2"	3
37	Fallo EDM dirección mesa 2	"AlarmasHMI.W2"	4
38	Fallo EDM dirección mesa 3	"AlarmasHMI.W2"	5
39	Fallo EDM dirección mesa 4	"AlarmasHMI.W2"	6
40	Fallo EDM dirección mesa 5	"AlarmasHMI.W2"	7
41	Fallo EDM dirección mesa 6	"AlarmasHMI.W2"	8
42	Fallo EDM cinta mesa 1	"AlarmasHMI.W2"	9
43	Fallo EDM cinta mesa 2	"AlarmasHMI.W2"	10
44	Fallo EDM cinta mesa 3	"AlarmasHMI.W2"	11
45	Fallo EDM cinta mesa 4	"AlarmasHMI.W2"	12
46	Fallo EDM cinta mesa 5	"AlarmasHMI.W2"	13
47	Fallo EDM cinta mesa 6	"AlarmasHMI.W2"	14
48	Fallo EDM CN1	"AlarmasHMI.W2"	15
49	Fallo EDM CN2	"AlarmasHMI.W3"	0
50	Fallo EDM CN3	"AlarmasHMI.W3"	1
51	Fallo EDM CN4	"AlarmasHMI.W3"	2
52	Fallo EDM CN5	"AlarmasHMI.W3"	3
53	Fallo EDM CN6	"AlarmasHMI.W3"	4
54	Fallo EDM CN7	"AlarmasHMI.W3"	5
55	Fallo EDM CN8	"AlarmasHMI.W3"	6
56	Fallo EDM CN9	"AlarmasHMI.W3"	7
57	Fallo EDM CN10	"AlarmasHMI.W3"	8
58	Fallo EDM CN11	"AlarmasHMI.W3"	9
59	Fallo EDM CN12	"AlarmasHMI.W3"	10
60	RESERVA	"AlarmasHMI.W3"	11
61	RESERVA	"AlarmasHMI.W3"	12
62	RESERVA	"AlarmasHMI.W3"	13
63	RESERVA	"AlarmasHMI.W3"	14
64	RESERVA	"AlarmasHMI.W3"	15
65	RESERVA	"AlarmasHMI.W4"	0
66	RESERVA	"AlarmasHMI.W4"	1
67	RESERVA	"AlarmasHMI.W4"	2
68	RESERVA	"AlarmasHMI.W4"	3
69	RESERVA	"AlarmasHMI.W4"	4
70	RESERVA	"AlarmasHMI.W4"	5
71	RESERVA	"AlarmasHMI.W4"	6

72	RESERVA	" AlarmasHMI.W4"	7
73	RESERVA	" AlarmasHMI.W4"	8
74	RESERVA	" AlarmasHMI.W4"	9
75	RESERVA	" AlarmasHMI.W4"	10
76	RESERVA	" AlarmasHMI.W4"	11
77	RESERVA	" AlarmasHMI.W4"	12
78	RESERVA	" AlarmasHMI.W4"	13
79	RESERVA	" AlarmasHMI.W4"	14
80	RESERVA	" AlarmasHMI.W4"	15

Apéndice D

Manual de uso

Este documento se va a centrar en la descripción del funcionamiento en conjunto de la programación del PLC, con el HMI, así como con Factory I/O.

Respecto a los pasos a seguir para ejecutar correctamente los diferentes programas y dejar todo listo para empezar a simular está dispuesto en la [Sección 7.4 del Pliego de condiciones](#).

Una vez está todo conectado correctamente en marcha, tanto PLCsim como Factory I/O como el runtime del HMI, se puede iniciar a controlar la planta.

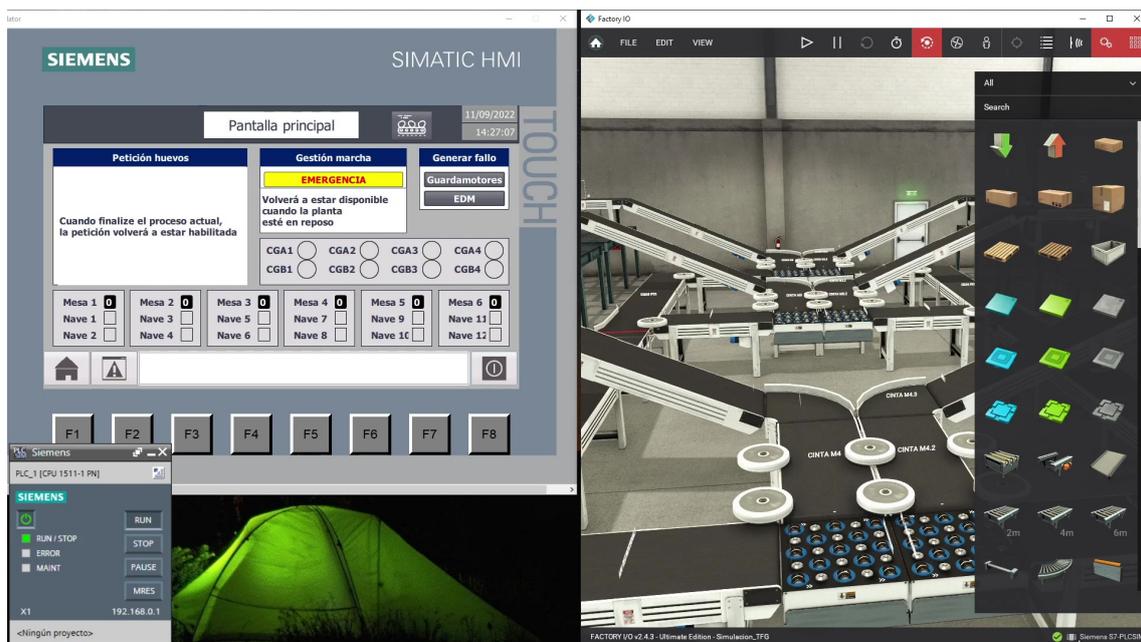


Figura 50: Software listo

Se va a explicar cómo es el inicio de la simulación, así como los diferentes modos de producción de la planta. Por último, se expondrán los diferentes tipos de emergencia, y cómo gestionarlas.

D.1. Inicio

Para poder iniciar la petición de huevos de las naves, primero es necesario comprobar que el estado de la planta es "Reposo". Esto se puede comprobar en el apartado "Gestión marcha" en el HMI.

En caso de mostrar el estado "EMERGENCIA" primero hay que ver de qué tipo de emergencia se trata, en la barra inferior aparecerá un mensaje con el tipo de emergencia que haya. En caso de que no aparezca ningún mensaje, significa que la emergencia ha sido resuelta y que el sistema está esperando que se rearme. Para rearmar, en Factory I/O hay que pulsar el botón amarillo del cuadro eléctrico, que pone "Rearme". Si hay un fallo de EDM o de guardamotor, hay que ir a la pantalla correspondiente y dejar en "OK" todos los selectores. Es normal que al principio de una simulación el estado de la planta sea "EMERGENCIA" ya que al iniciar del PLCSim antes que Factory I/O no se recibe la señal de la seta.

Una vez la planta está en reposo, en el apartado "Petición huevos" aparecerán los diferentes botones que permiten realizar la petición. Estos únicamente aparecen cuando la planta se encuentra en "Reposo" o "Marcha". Antes de iniciar el funcionamiento normal hay que elegir entre "AUTO" o "MAN", siendo el primero el activo por defecto, va siguiendo todos los pasos automáticamente hasta llegar a "Marcha", mientras que en "MAN" es necesario ir pulsando el botón "CONTINUAR" para seguir el proceso.



Figura 51: Petición de huevos lista para "Arrancar"

Una vez se ha seleccionado el funcionamiento deseado; las naves de las que se desea extraer huevos y la cinta general por la cual se desea que llegue a la clasificadora, hay que pulsar el botón "Arranque" en Factory I/O. En ese momento empezará la producción, siguiendo los pasos que se detallan en el siguiente apartado.

D.2. Producción

Si en la "Gestión marcha" se había seleccionado el modo "AUTO", la planta empezará con el modo "Preparación", y continuará con la marcha, indicándolo así en el estado de la planta.

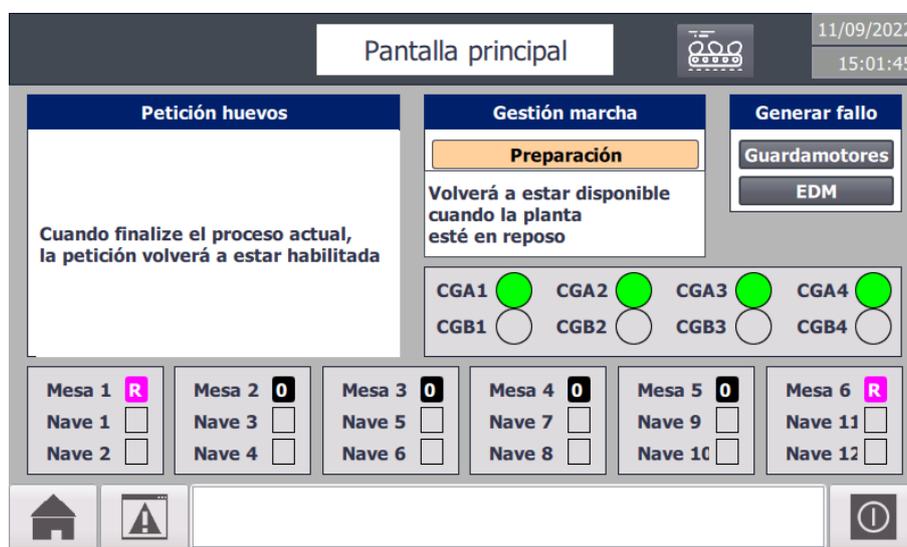


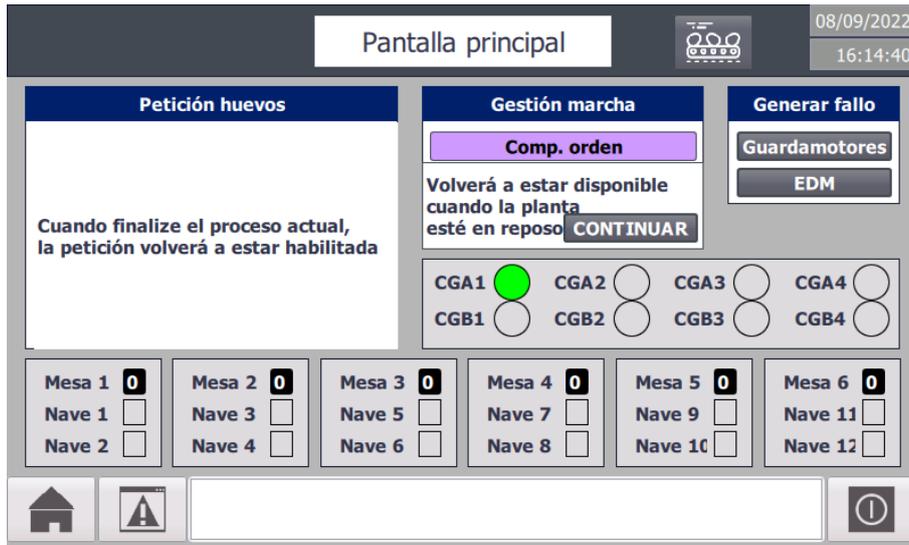
Figura 52: Preparación

Una vez en modo "Marcha", se puede cambiar la petición de huevos. Cada vez que se cambie una nave o una cinta, la planta seguirá los mismos pasos de preparación y luego marcha; a menos que la petición sea nula, es decir que no se seleccione ninguna nave o ninguna cinta general, en cuyo caso la planta entraría en modo "Reposo".

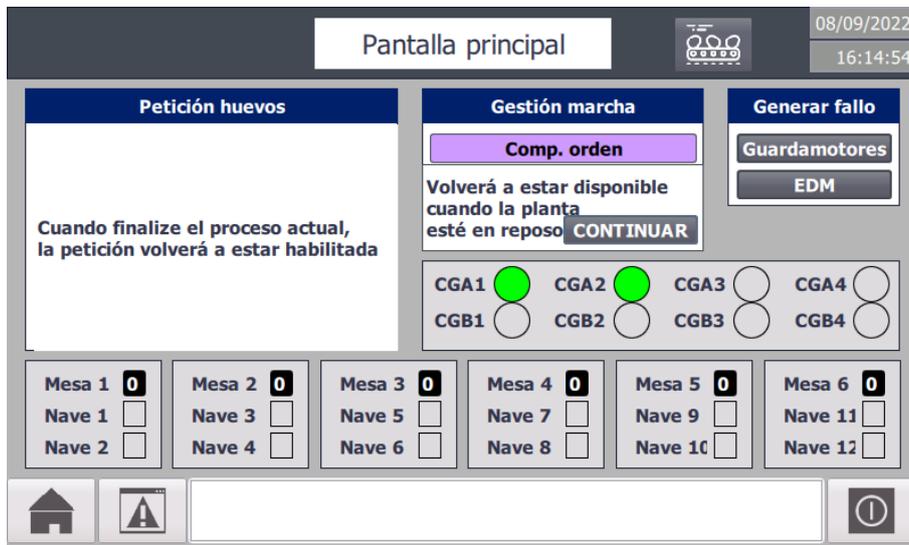


Figura 53: Marcha

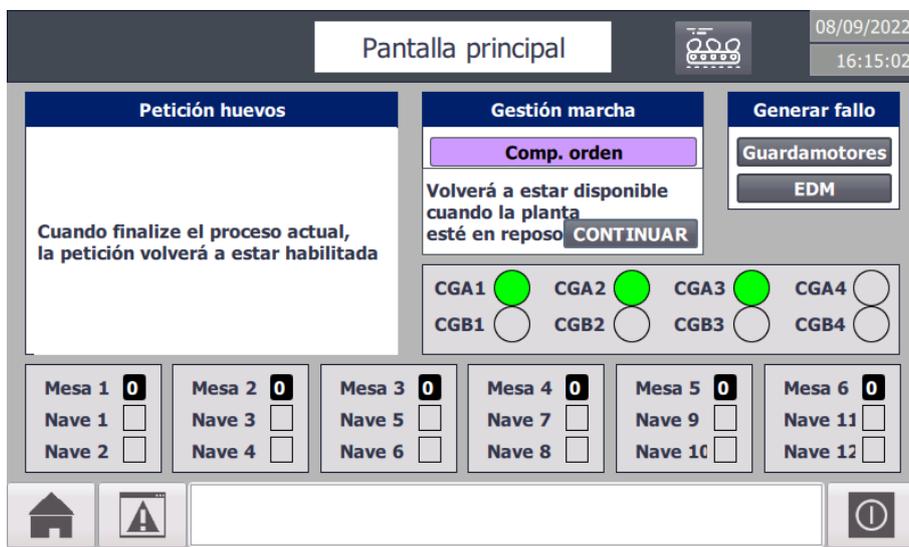
Si en la "Gestión marcha" se había seleccionado el modo "MAN", tras pulsar el botón de "Arranque", no se podrá modificar la petición de huevos hasta que la planta no pase por "Preparación", "Marcha" y "Finalizando". Para ello se tendrá que pulsar el botón "CONTINUAR" en el HMI.



(a) Comprobación manual 1



(b) Comprobación manual 2



(c) Comprobación manual 3

Figura 54: Ejemplo comprobación manual

Una vez la planta está en modo "Marcha" o en la comprobación en orden se haya llegado al equivalente de "Marcha", si se pulsa el botón del HMI "Fin de ciclo" o hace más de dos minutos que no pasan huevos por alguna de las cintas generales que están activas, la planta entrará en modo "Finalizando", y una vez se hayan parado todas las cintas, volverá al modo reposo. Si está seleccionado el modo "MAN", habrá que pulsar el botón de "CONTINUAR" para ir parando las cintas.

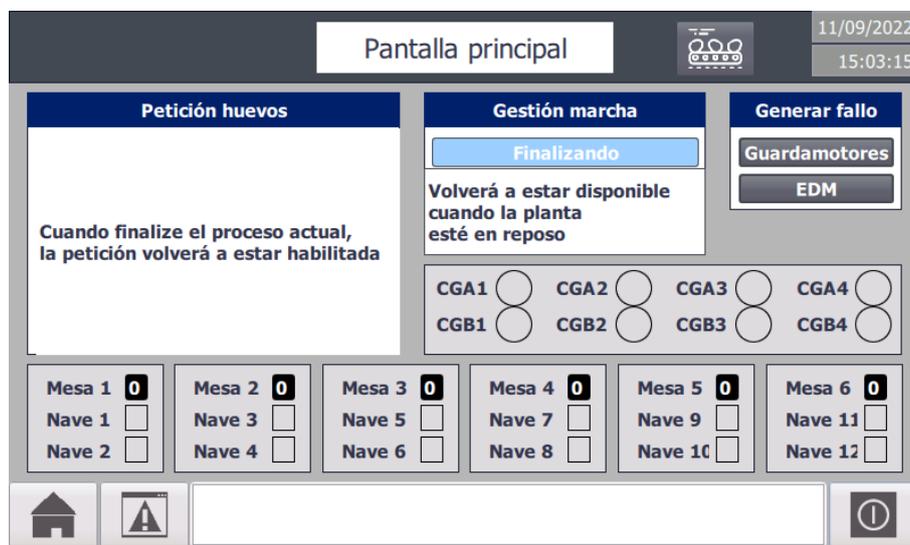


Figura 55: Finalizando

Una vez se paren todas las cintas, ya sea por petición nula o porque el modo "Finalizando" ya ha acabado su proceso, la planta volverá al modo "Reposo".

Si en "Reposo" se activa la opción "COMP. DESORDEN" del HMI, la planta activará todas las cintas de las naves, todas las cintas generales, todas las cintas de mesa y la dirección "Avanzar" de las mismas durante un minuto, tiempo durante el cual se producirá.



Figura 56: Comprobación desorden

Hay que tener en cuenta que la simulación de Factory I/O está diseñada para que, si las cintas de las naves están activas, que saque los objetos azules. Si se quiere comprobar que la planta entra en modo "Finalizando" porque hace más de dos minutos que ha pasado el último huevo, hay que volver al modo edición de Factory I/O por dos minutos. Al no cerrar el programa, no se activará el bit de emergencia de PARO SEGURIDAD, ya que se sigue enviando la señal de la seta, pero como en este modo las fotocélulas no detectan nada, el temporizador se pondrá en marcha.

D.3. Emergencia

El modo "EMERGENCIA" se activará siempre que se pulse la seta de emergencia en Factory I/O, o se genere un fallo de guardamotores o pase más del tiempo programado (predeterminado: 30 segundos) en fallo EDM de algún elemento, independientemente del estado en el cual se encuentre previamente la planta.

The screenshot shows the 'Fallo guardamotores' screen with a date of 11/09/2022 and time 18:40:07. It is divided into four parts (Parte 1 to Parte 4), each with a table of components and their status.

Parte 1	
Cinta	Estado guardamotor
CGA1	OK
CGB1	OK
Mesa 1	OK
Nave 1	OK
Nave 2	OK

Parte 2	
Cinta	Estado guardamotor
CGA2	OK
CGB2	OK
Mesa 2	OK
Nave 3	OK
Nave 4	OK

Parte 3	
Cinta	Estado guardamotor
CGA3	OK
CGB3	OK
Mesa 3	OK
Nave 5	OK
Nave 6	OK
Mesa 4	OK
Nave 7	OK
Nave 8	OK

Parte 4	
Cinta	Estado guardamotor
CGA4	OK
CGB4	OK
Mesa 5	OK
Nave 9	OK
Nave 10	OK
Mesa 6	OK
Nave 11	OK
Nave 12	OK

(a) Planta sin emergencias

The screenshot shows the 'Fallo guardamotores' screen with a date of 11/09/2022 and time 18:42:01. The status of 'Parte 1' is now 'NO OK', while all other components remain 'OK'. A message bar at the bottom indicates the fault: 'E 11/09/2022 18:41:56 Guardamotores cinta A pt1'.

Parte 1	
Cinta	Estado guardamotor
CGA1	NO OK
CGB1	OK
Mesa 1	OK
Nave 1	OK
Nave 2	OK

Parte 2	
Cinta	Estado guardamotor
CGA2	OK
CGB2	OK
Mesa 2	OK
Nave 3	OK
Nave 4	OK

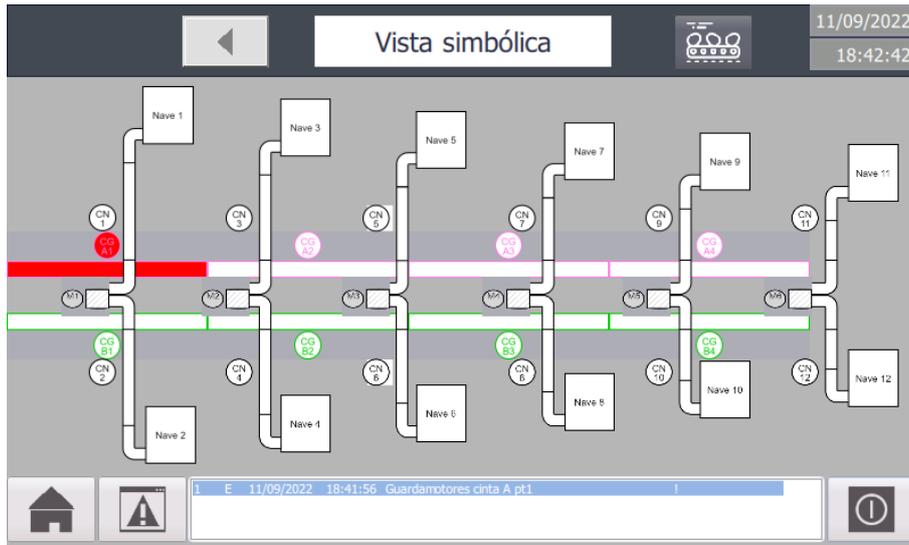
Parte 3	
Cinta	Estado guardamotor
CGA3	OK
CGB3	OK
Mesa 3	OK
Nave 5	OK
Nave 6	OK
Mesa 4	OK
Nave 7	OK
Nave 8	OK

Parte 4	
Cinta	Estado guardamotor
CGA4	OK
CGB4	OK
Mesa 5	OK
Nave 9	OK
Nave 10	OK
Mesa 6	OK
Nave 11	OK
Nave 12	OK

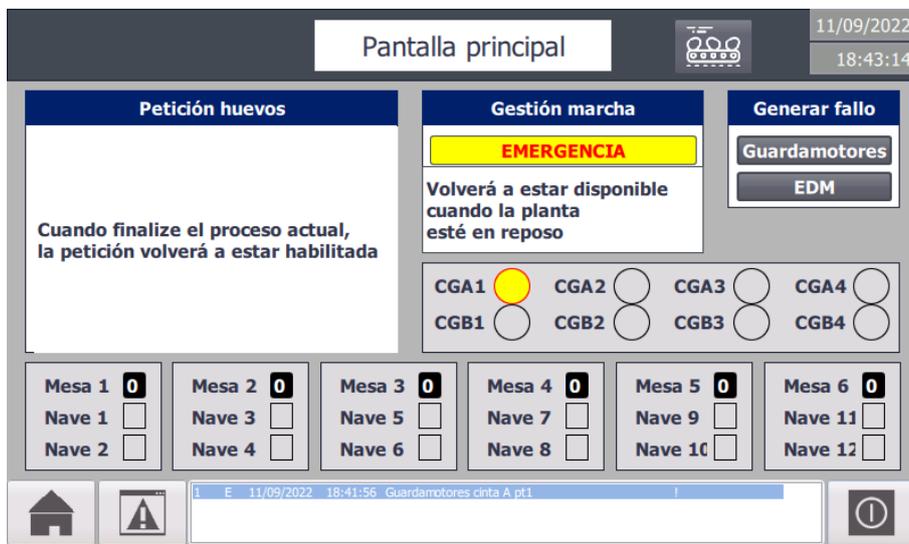
(b) Se genera emergencia fallo guardamotor

Figura 57: Generación de emergencia mediante fallo guardamotor

Ante este acontecimiento, en el HMI se mostrará en la ventana de avisos de la barra inferior un mensaje que indica la naturaleza del error. Desde la vista simbólica, si se trata de un fallo de algún actuador, mostrará de color rojo fijo el elemento causante del fallo. En la pantalla principal se mostrará también, pero parpadeante entre rojo y amarillo. En la Subsección 5.3.3 se explica como generar los fallos de guardamotor y EDM.



(a) Fallo guardamotor en vista simbólica



(b) Planta en emergencia

Figura 58: Visualización emergencia fallo guardamotor

Una vez esté resuelta la emergencia, ya no aparecerá en el visor de avisos. Para volver a "Reposo" hay que pulsar el botón "Rearme" de Factory I/O. Si aún hay algún aviso activo de fallo, pulsar "Rearme" no tendrá ningún efecto.

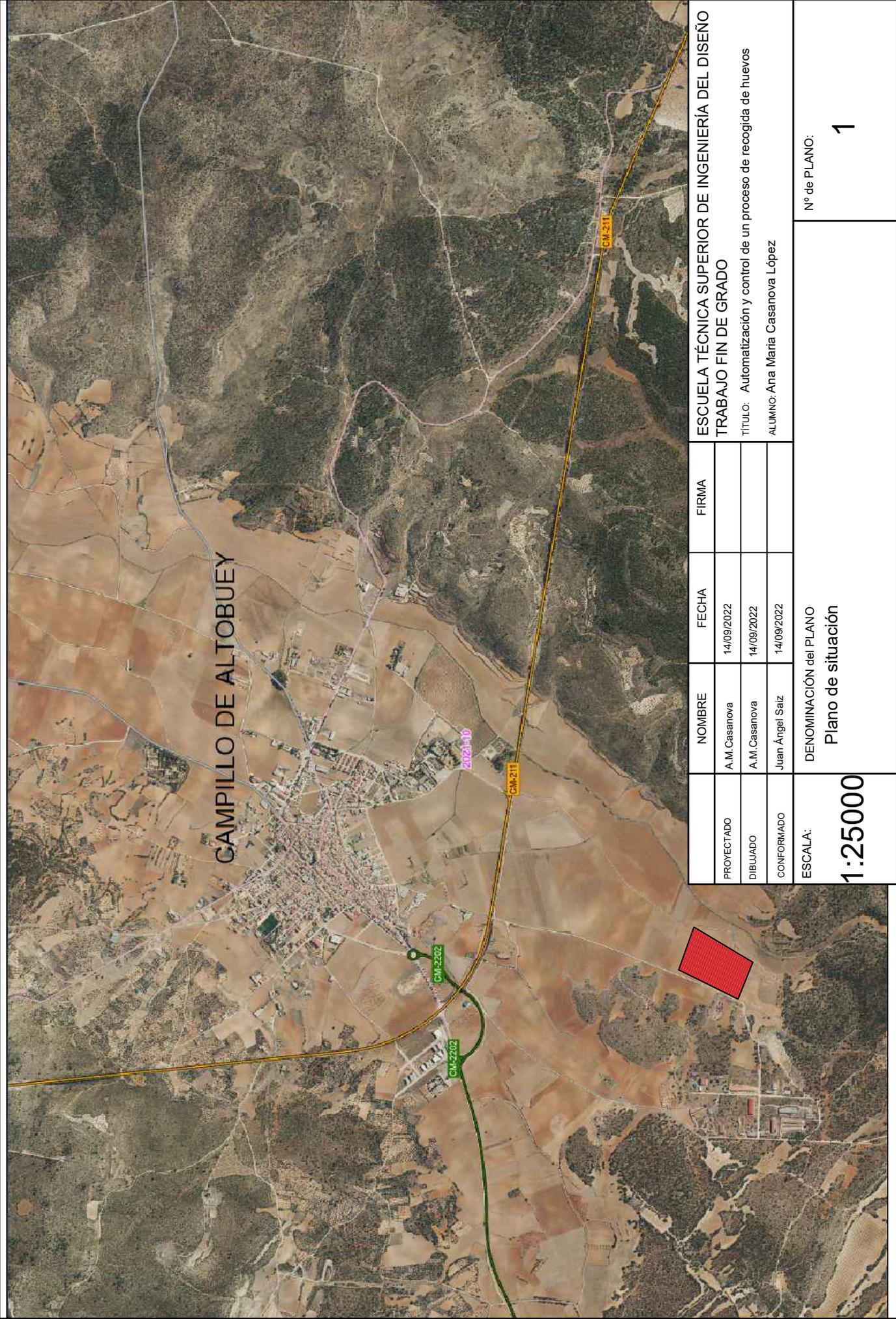
Para visualizar las alarmas, tanto las activas como las históricas, consultar la [Subsección 5.3.4 Pantalla Alarmas Activas y Alarmas Históricas](#).

Documento II

Planos

Índice

Índice	149
6.1 Plano de situación	151
6.2 Plano de ubicación	153



CAMPILLO DE ALTOBUEY

	NOMBRE	FECHA	FIRMA
PROYECTADO	A.M. Casanova	14/09/2022	
DIBUJADO	A.M. Casanova	14/09/2022	
CONFORMADO	Juan Ángel Saiz	14/09/2022	

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DEL DISEÑO
 TRABAJO FIN DE GRADO

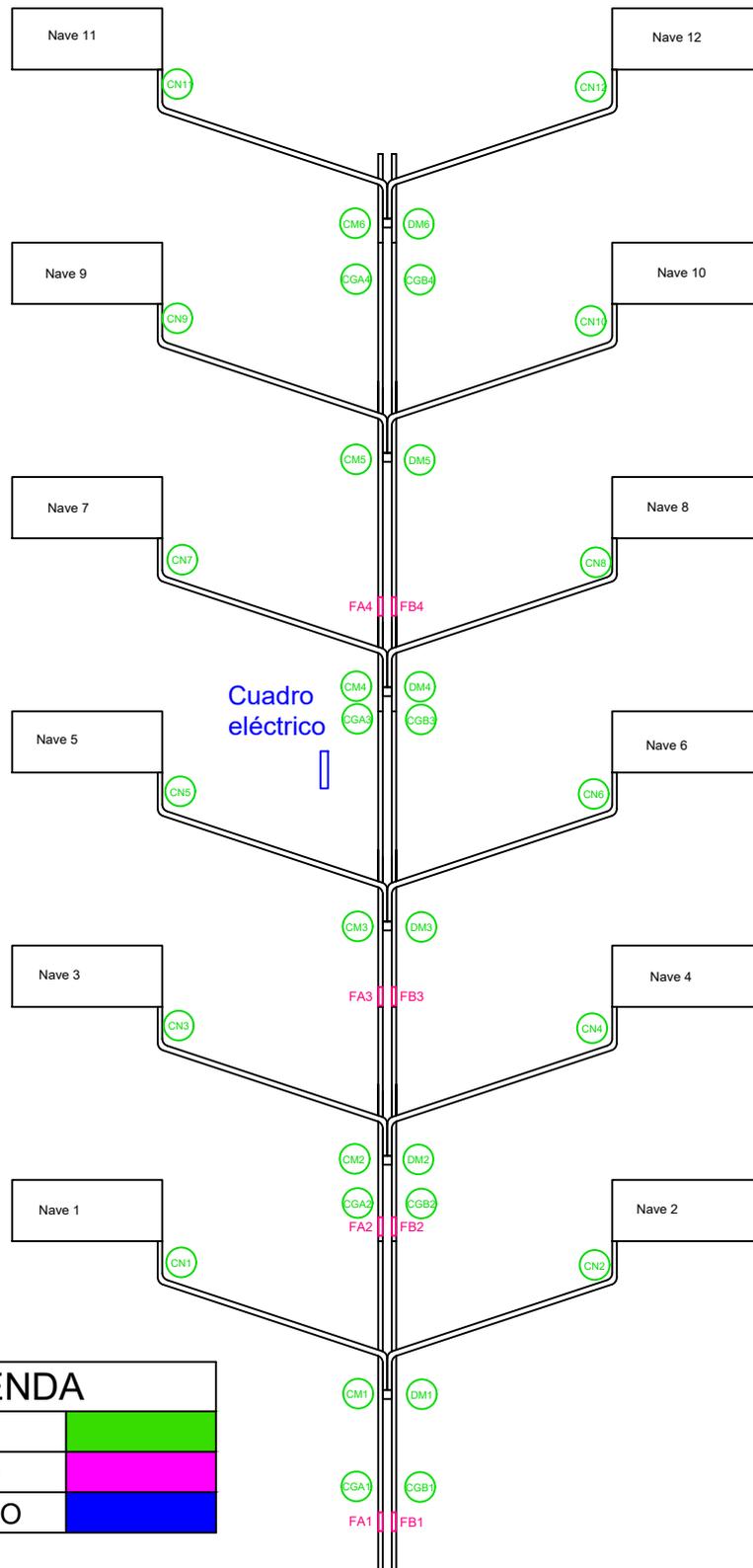
TÍTULO: Automatización y control de un proceso de recogida de huevos
 ALUMNO: Ana María Casanova López

DENOMINACIÓN del PLANO
 Plano de situación

Nº de PLANO:

1

ESCALA:
 1:25000



LEYENDA	
MOTORES	
SENSORES	
C. ELÉCTRICO	

	NOMBRE	FECHA	FIRMA	ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA DEL DISEÑO TRABAJO FIN DE GRADO
PROYECTADO	A.M.Casanova	14/09/2022		TÍTULO: Automatización y control de un proceso de recogida de huevos
DIBUJADO	A.M.Casanova	14/09/2022		ALUMNO: Ana Maria Casanova López
CONFORMADO	Juan Ángel Saiz	14/09/2022		
ESCALA: 1:200	DENOMINACIÓN del PLANO Plano de ubicación			Nº de PLANO: 2

Documento III

Pliego de condiciones

Índice

Índice	157
Índice de figuras	159
7.1 Objeto	160
7.2 Normativa	160
7.3 Condiciones de los equipos de simulación y control	160
7.3.1 Requisitos mínimos para Factory I/O v2.4.3	161
7.3.2 Requisitos mínimos para TIA Portal V16	161
7.4 Condiciones de ejecución del programa	162
7.5 Pruebas de conexión	167

Índice de figuras

1	Iniciar simulación TIA Portal pt I	162
2	Iniciar simulación TIA Portal pt II	163
3	Iniciar simulación TIA Portal pt III	164
4	Simulación HMI	165
5	Inicio Factory I/O	166
6	Drivers	167
7	Ejemplo error de conexión	168

7.1. Objeto

El objeto de este apartado es agrupar las características técnicas necesarias para desarrollar el proyecto previamente descrito.

El objetivo del proyecto es controlar y simular los diferentes elementos que conforman el proceso de recogida de huevos en una macro granja de gallinas.

7.2. Normativa

La normativa vigente en España que concierne al proyecto de programación y simulación es la siguiente:

- IEC 61131-3:2013 Autómatas programables. Parte 3: Lenguajes de programación, donde se definen la sintaxis y la semántica de los lenguajes de programación para controladores programables.
- ISO 13850:2015 Seguridad de las máquinas. Función de parada de emergencia. Principios para el diseño., donde establece los requisitos para el correcto funcionamiento de la función de parada de emergencia de una máquina.
- IEC 60848:2013 Lenguaje de especificación GRAFCET para diagramas funcionales en secuencia, donde se especifican los símbolos y reglas para la representación del lenguaje.
- UNE-EN 61508 Seguridad funcional de los sistemas eléctricos / electrónicos / electrónicos programables relacionados con la seguridad.
- UNE-EN 62439 Redes de comunicación industrial. Redes de automatización de alta disponibilidad.

7.3. Condiciones de los equipos de simulación y control

A continuación, se van a enunciar los diferentes requisitos mínimos que tiene que cumplir el ordenador donde se vaya a realizar la simulación del proyecto. Hay que tener en cuenta que el equipo tiene que ser capaz de utilizar ambos programas a la vez, por lo que las características de espacio y memoria finales es la suma de todos los requisitos, mientras que el requisito mínimo de CPU o sistema operativo será el mayor de ellos.

Hay que tener en cuenta que la programación se ha realizado para unos equipos concretos (CPU Siemens SIMATIC S7-1500 CPU 1511-1 PN, módulo de entradas digitales Siemens SIMATIC S7-1500 DI 32x24VDC BA, módulo de salidas digitales Siemens SIMATIC S7-1500 DQ 32x24VDC/0.5A HF y pantalla Siemens SIMATIC HMI KTP700 Basic color

PN) en caso de querer cambiar alguno de estos componentes será necesario comprobar la compatibilidad.

7.3.1. Requisitos mínimos para Factory I/O v2.4.3

- Sistema operativo: Windows 7 SP1+ o superior
- CPU: Soporte para conjunto de instrucciones SSE2
- GPU: NVIDIA desde 2006 (GeForce 8), AMD desde 2006 (Radeon HD 2000), Intel desde 2012 (HD 4000 / IvyBridge)
- RAM: 1 GB
- Disco duro: 700 MB libres

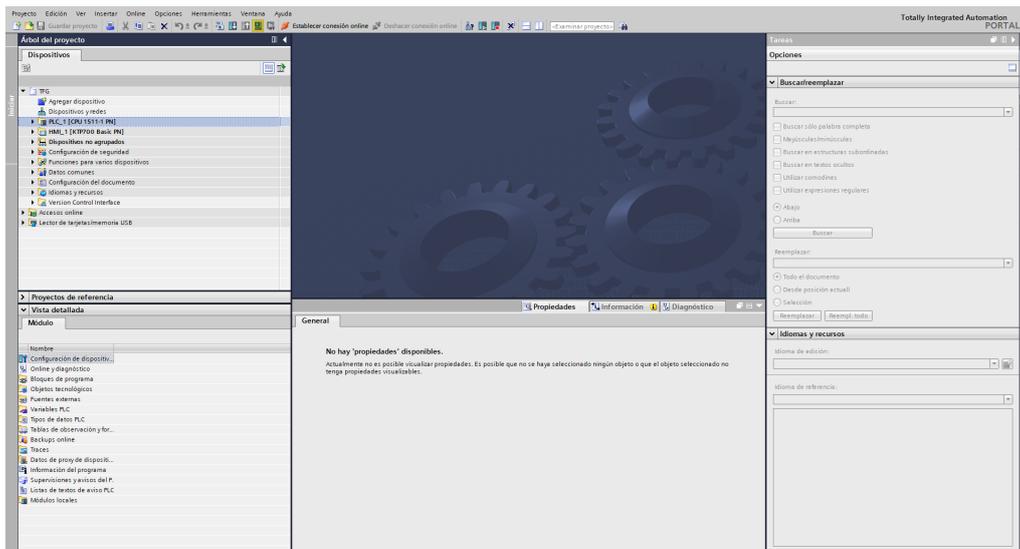
7.3.2. Requisitos mínimos para TIA Portal V16

- Sistema operativo: Windows 7 SP1+ o superior
- CPU: Intel®Core™i3-6100U, 2.30 GHz
- RAM: 8 GB
- Disco duro: S-ATA con 20 GB libres

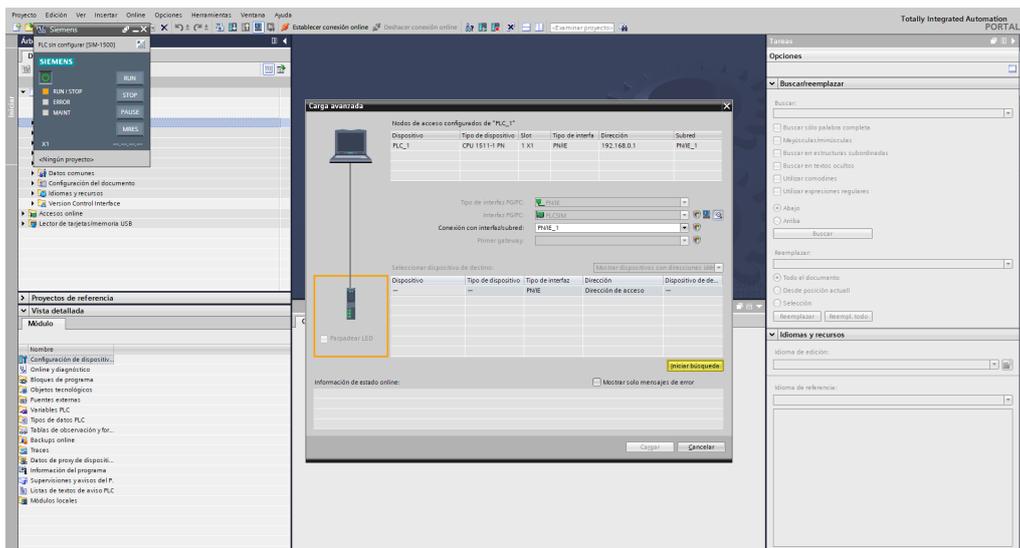
7.4. Condiciones de ejecución del programa

Para poder ejecutar el proyecto correctamente es necesario tener instalados los programas descritos anteriormente, siendo posible su ejecución si se tienen versiones superiores a las indicadas.

Primero, hay que ejecutar el software de programación TIA Portal, y abrir el archivo del proyecto. Una vez abierto, hay que seleccionar PLC_1 y darle el botón de simular. Se abrirá un diálogo donde hay que poner los valores indicados en la [Figura 1b](#) y darle el botón "Iniciar búsqueda".



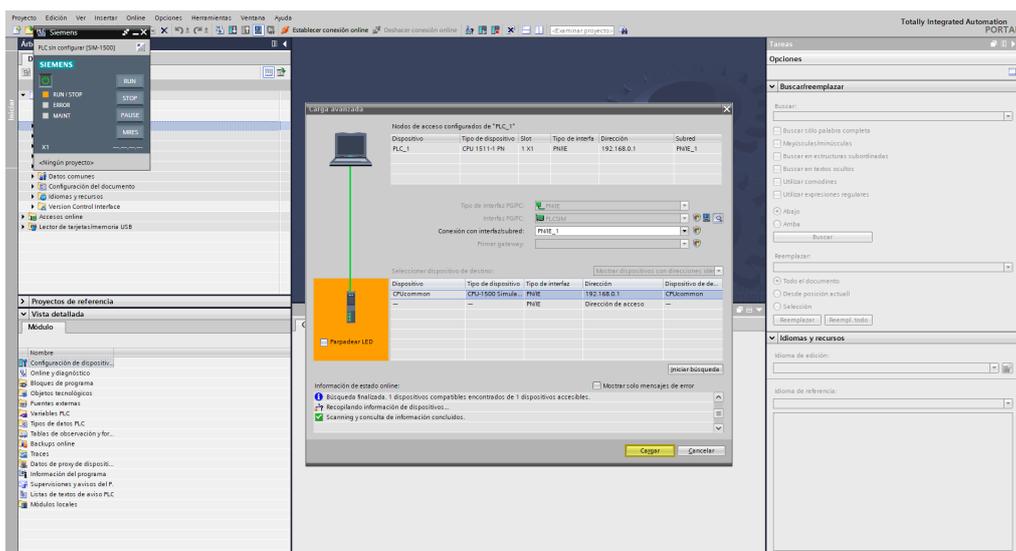
(a) Simular



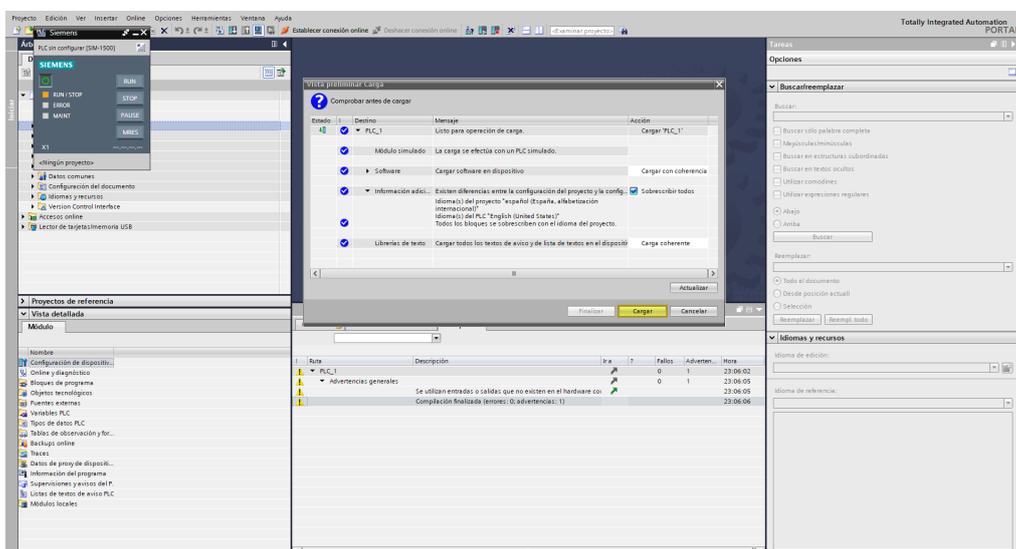
(b) Iniciar búsqueda

Figura 1: Iniciar simulación TIA Portal pt I

Una vez encuentre el PLCsim, mostrará la **Figura 2a** y para continuar habrá que darle a "Cargar". El software compilará el programa y dará que hay diferencias entre el idioma del PLC y del proyecto, se selecciona "Sobrescribir todos" y se pulsa de nuevo el botón de cargar. Si se ha cargado correctamente en el diálogo aparecerá el resultado de la operación de carga, y para realizar la simulación habrá que desplegar el diálogo y seleccionar "Arrancar módulo", una vez hecho esto se le puede dar a finalizar y la simulación del PLC ya habrá comenzado. Para comprobar que todo se ha cargado correctamente, el LED de RUN/STOP en el PLCsim estará en verde y el LED de ERROR estará desactivado, tal y como muestra la **Figura 3b**.

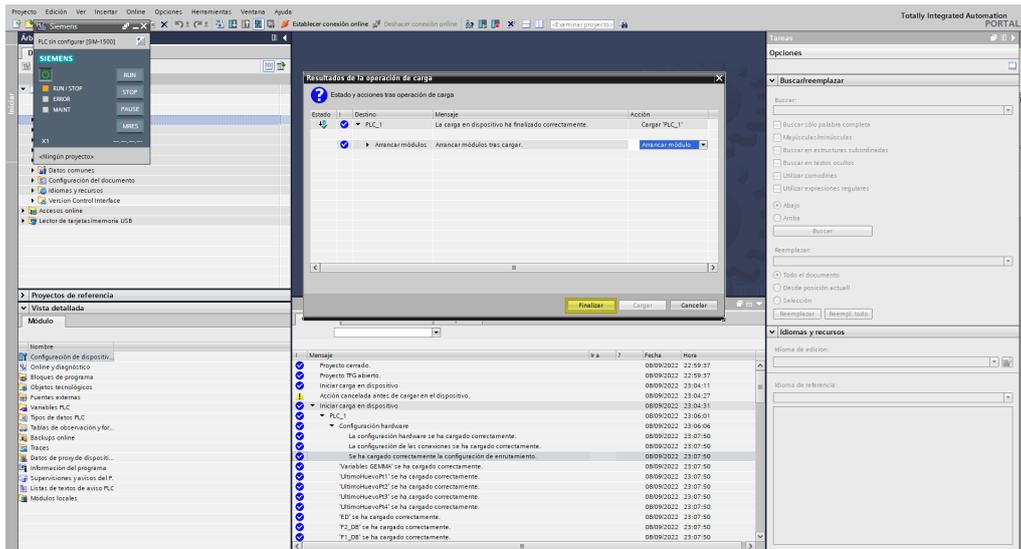


(a) Cargar

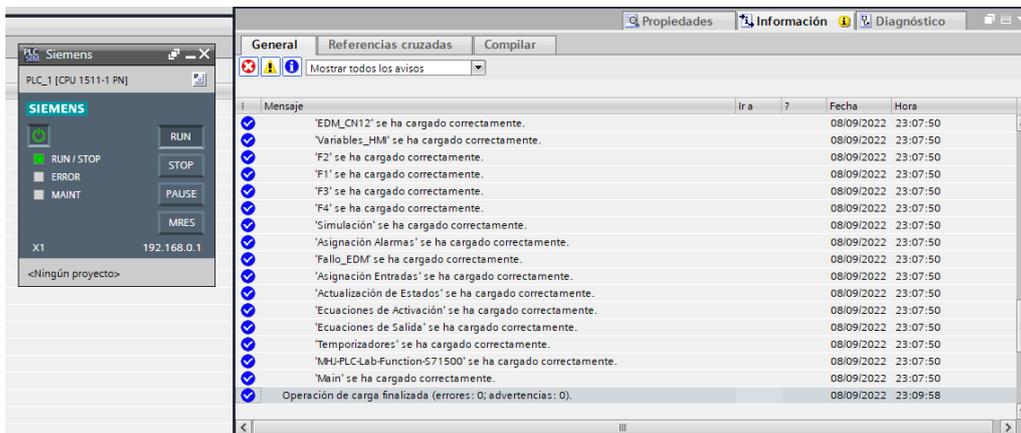


(b) Sobrescribir todos

Figura 2: Iniciar simulación TIA Portal pt II



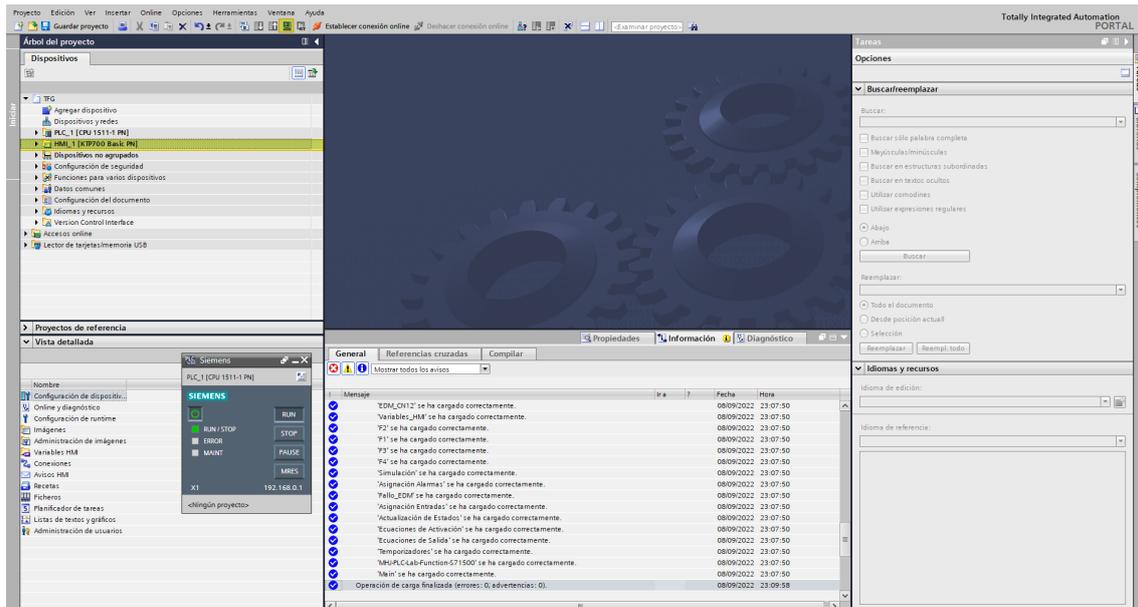
(a) Finalizar



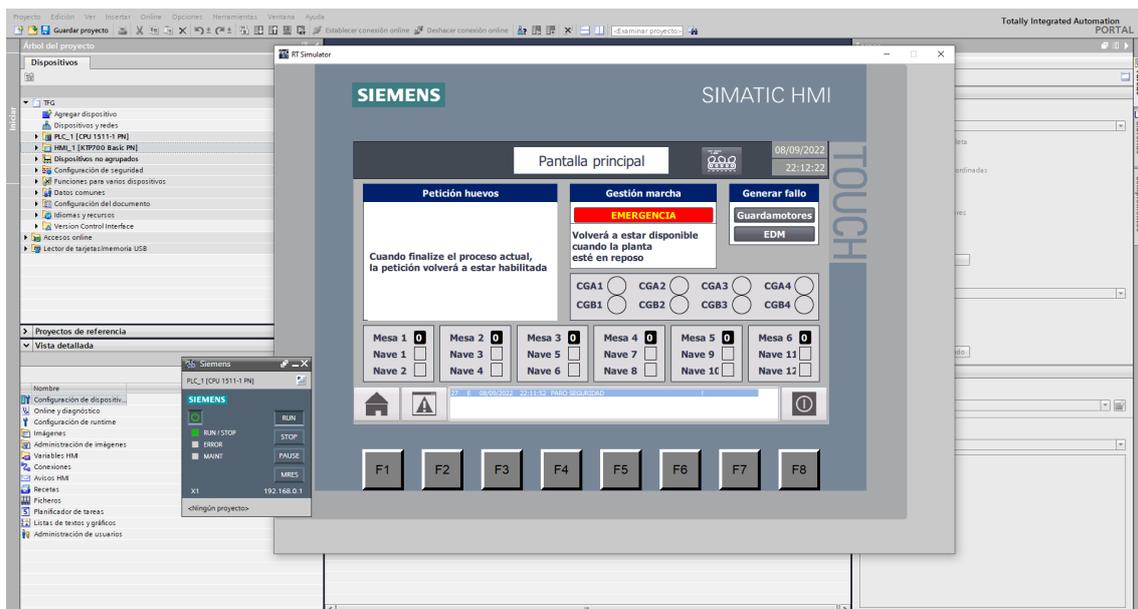
(b) Carga OK

Figura 3: Iniciar simulación TIA Portal pt III

Antes de continuar con la simulación, ya que está abierto el TIA Portal, se va a activar el runtime del HMI. Para esto, en el mismo proyecto hay que seleccionar el HMI y pulsar el botón de simular, descrito anteriormente. Una vez compilado el programa del HMI se abre su runtime.



(a) Seleccionar y simular



(b) Runtime activo

Figura 4: Simulación HMI

Una vez hecho esto, sólo queda iniciar el Factory I/O. Para ello hay que abrir el proyecto, y si se han seguido los pasos anteriores correctamente, el controlador deberá conectarse automáticamente. Para saber si es así hay que comprobar que en la esquina inferior derecha del software, al lado del símbolo de controlador tiene que haber un tic con fondo verde, tal y como se muestra en la siguiente figura:

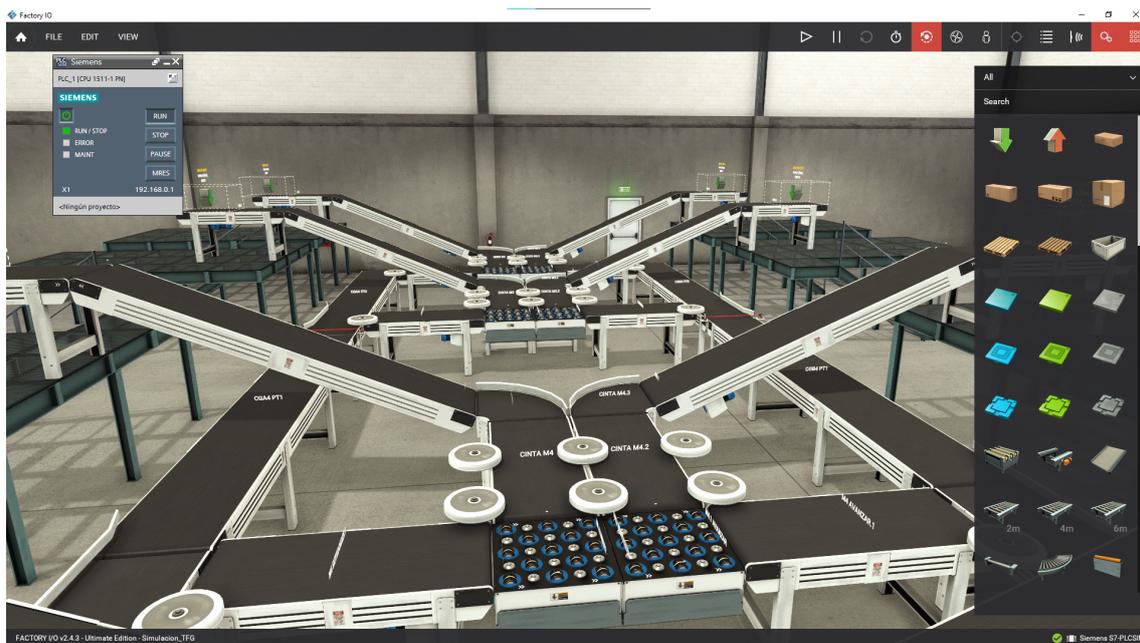


Figura 5: Inicio Factory I/O

Para empezar la simulación sólo hay que darle al play de la barra superior. Y ya está listo para iniciar la producción.

7.5. Pruebas de conexión

Una vez cumplidas todas las condiciones de ejecución, se puede comprobar la correcta transmisión entre los programas de diferentes formas.

Una de ellas consiste en probar directamente el funcionamiento del proyecto. Para ello se tienen que seguir los pasos indicados en el [Apéndice D](#), aunque aquí se van a exponer brevemente.

Si la CPU del PLC no ha sido parada (se ha pulsado el botón de STOP y después RUN), en el HMI deberá aparecer que la planta está en estado de emergencia y no deberá aparecer ningún aviso activo. Si hubiera algún aviso, primero hay que quitarlos. Para ir al estado de reposo, será necesario pulsar el botón de rearme en Factory I/O mientras este programa está en modo run (aparece un cuadrado en vez del botón de play mencionado en la [7.4](#)). Si en el HMI aparece que la planta está en reposo, entonces se puede afirmar que la conexión se ha realizado correctamente.

Si la CPU ha sido parada, para comprobar el funcionamiento habrá que poner en marcha la planta, o generar una emergencia con la seta de Factory I/O. Para poner en marcha la planta, habrá que poner a ON mínimo una nave en el HMI y habrá que activar alguna cinta general, la A o la B o ambas. Después, en Factory I/O hay que darle el botón "Arrancar" del cuadro eléctrico, entonces los elementos necesarios se deberán de haber puesto en marcha. Para generar una emergencia en Factory I/O hay que pulsar una vez la seta del cuadro eléctrico, si en el HMI se activa la alarma de seta y la planta entra en estado de EMERGENCIA, entonces la comunicación se ha establecido correctamente.

Otra forma de comprobar la conexión es mirar a la esquina inferior derecha del Factory I/O si la conexión se ha realizado correctamente aparecerá un tic con fondo verde. En cambio, si no se ha podido realizar, aparecerá una cruz con fondo rojo, además de un *pop-up* indicando cuál es el posible problema. Si el Factory I/O no ha intentado establecer ninguna conexión, no aparecerá ningún símbolo, por lo que, si se desea conectar, tendrá que entrar en la configuración de los drivers, pulsando F4 o sobre el símbolo de controlador en la esquina inferior derecha.



Figura 6: Drivers

Una de las razones por las que puede haber error de conexión entre Factory I/O y PLCSim es que haya más de una instancia de PLCSim abierta, ya que de esta forma Factory I/O desconoce con cuál de las dos instancias tiene que comunicarse. Otra posible razón sea la modificación o bien de la programación del PLC o de Factory I/O.

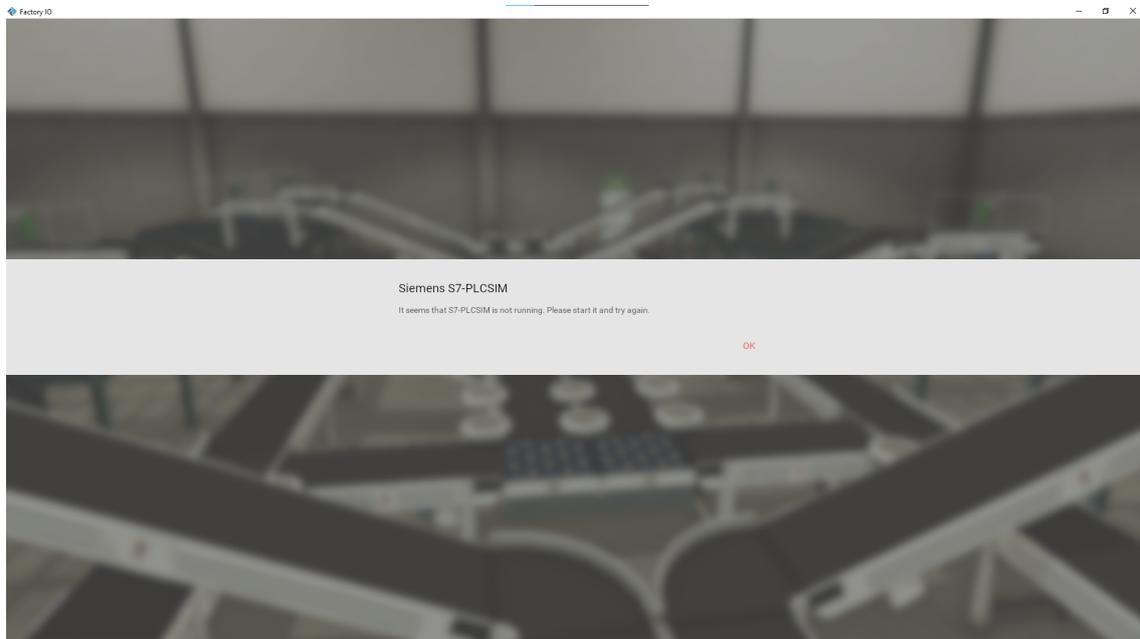


Figura 7: Ejemplo error de conexión

Documento IV

Presupuesto

Índice

Índice	171
Índice de tablas	173
8.1 Precios unitarios	174
8.1.1 Materiales	174
8.1.2 Mano de obra	174
8.1.3 Equipo técnico de oficina y software	174
8.2 Precios descompuestos	175
8.3 Mediciones	176
8.4 Presupuesto	176

Índice de tablas

1	Precios unitarios	174
2	Mano de obra	174
3	Equipo técnico de oficina y software	175
4	d1-Programación PLC y HMI	175
5	d2-Programación Factory I/O	176
6	Mediciones	176
7	Valoración	176

Para la realización del presupuesto del proyecto, se van a tener en cuenta los costes de programación del PLC, del HMI, así como el desarrollo de la simulación.

Ya que se parte de la base de que toda la planta está cableada, a falta de integrar los dispositivos de control, no se van a tener en cuenta los costes de hardware relacionados con la sensorización de la planta o la maquinaria de esta; pero sí que se va a tener en cuenta el precio de la CPU y de los módulos de entradas y salidas, así como el HMI. Además, todas las licencias de los software necesarios también se incluyen.

8.1. Precios unitarios

8.1.1. Materiales

En esta sección se incluyen los materiales que se utilizan íntegramente en este proyecto.

1. Precios unitarios			
1.1. Materiales			
Ref	Unidad	Descripción	Precio (€)
m1	u.	Siemens SIMATIC S7-1500 CPU 1511-1 PN	712,00
m2	u.	Siemens SIMATIC HMI KTP700 Basic color PN	649,00
m3	u.	Cable de Ethernet, Cat5, 2 m, 6.6 ft, RJ45-RJ45	43,70
m4	u.	Siemens SIMATIC S7-1500 DQ 32x24VDC/0.5A HF	432,00
m5	u.	Siemens SIMATIC S7-1500 DI 32x24VDC BA	307,00

Tabla 1: Precios unitarios

8.1.2. Mano de obra

En el coste de mano de obra únicamente se incluyen horas de ingeniería.

1.2. Mano de obra			
Ref	Unidad	Descripción	Precio (€)
h1	h	Ingeniero en electrónica industrial y automática	35,00

Tabla 2: Mano de obra

8.1.3. Equipo técnico de oficina y software

En este apartado se incluye todo el material y licencias que se amortizan a lo largo de los años. Se ha tenido en cuenta el portátil utilizado y las licencias necesarias. Se ha estimado la vida útil de cada componente según los datos de la [Agencia Tributaria](#).

1.5. Equipo técnico de oficina y software					
Ref	Unidad	Descripción	Precio de compra (€)	Vida útil (h)	Precio (€)
t1	h	Legion 5 15IMH05	1205,37	87600,00	0,014
t2	h	Siemens TIA STEP 7 Prof. V16	2340,00	4320,00	0,542
t3	h	Factory I/O Siemens Edition	395,00	52560,00	0,008

Tabla 3: Equipo técnico de oficina y software

8.2. Precios descompuestos

2. Precios descompuestos					
Ref	Uds.	Descripción	Precio (€)	Cant.	Total (€)
d1	u.	Programación PLC y HMI			
2.1.1. Materiales					
m1	u.	Siemens SIMATIC S7-1500 CPU 1511-1 PN	712,00	1	712,00
m2	u.	Siemens SIMATIC HMI KTP700 Basic color PN	649,00	1	649,00
m3	u.	Cable de Ethernet, Cat5, 2 m, 6.6 ft, RJ45-RJ45	43,70	1	43,70
m4	u.	Siemens SIMATIC S7-1500 DQ 32x24VDC/0.5A HF	432,00	2	864,00
m5	u.	Siemens SIMATIC S7-1500 DI 32x24VDC BA	307,00	3	921,00
2.1.2. Mano de obra					
h2	h	Ingeniero en electrónica industrial y automática	35,00	100,00	3500,00
2.1.2. Equipo técnico de oficina y software					
t1	h	Legion 5 15IMH05	0,014	100,00	1,38
t2	h	Siemens TIA STEP 7 Prof. V16	0,542	100,00	54,17
2.1.3. Gastos indirectos					
%			10,00	6745,24	674,52
Precio total de ejecución (€)					7419,77

Tabla 4: d1-Programación PLC y HMI

d2	u.	Programación Factory I/O			
2.2.1. Mano de obra					
h1	h	Ingeniero en electrónica industrial y automática	35,00	100,00	3500,00
2.2.2. Equipo técnico de oficina y software					
t1	h	Legion 5 15IMH05	0,014	100,00	1,38
t3	h	Factory I/O Siemens Edition	0,008	100,00	0,75
2.2.3. Gastos indirectos					
%			10,00	3502,13	350,21
Precio total de ejecución (€)					3852,34

Tabla 5: d2-Programación Factory I/O

8.3. Mediciones

3. Mediciones			
Ref	Unidades	Descripción	Cantidad
d1	u.	Programación PLC y HMI	1,00
d2	u.	Programación Factory I/O	1,00

Tabla 6: Mediciones

8.4. Presupuesto

4. Valoración					
Ref	Unidad	Descripción	Precio (€)	Cantidad	Total (€)
d1	u.	Programación PLC y HMI	7419,77	1,00	7419,77
d2	u.	Programación Factory I/O	3852,34	1,00	3852,34
4.1 Beneficio Industrial					
%			10,00	11272,11	1127,21
4.2 IVA					
%			21,00	12399,32	2603,86
Presupuesto total de ejecución (€)					15003,17

Tabla 7: Valoración

El presupuesto total **con IVA** asciende a la cantidad de QUINCE MIL TRES EUROS Y DIECISIETE CÉNTIMOS.

Bibliografía

-
- [1] 46.52.9.024.0020 FINDER - Relé: electromagnético / DPDT; 24VCC; miniatura, industrial; 5mm / TME - *Elektroniikka komponentit*, es. dirección: <https://www.tme.eu/es/details/46.52.9.024.0020/reles-electromagn-industriales/finder/> (visitado 29-08-2022).
- [2] es, abr. de 2018. dirección: <https://www.talleresoliver.com/maquinaria/cinta-transportadora> (visitado 01-09-2022).
- [3] *Productos*, es-AR. dirección: <https://sctpatagonia.com.ar/producto/> (visitado 01-09-2022).
- [4] *Phoenix Contact Relay Module - RIF-0-RPT-24DC/ 1 - 2903361*, en. dirección: <https://marketplace.realpars.com/en-us/products/relay-module-rif-0-rpt-24dc-1-2903361> (visitado 27-08-2022).
- [5] *Siemens SIMATIC S7-1500 CPU 1516F-3 PN/DP - 6AG1516-3FN01-2AB0*, en. dirección: <https://marketplace.realpars.com/products/simatic-s7-1500-cpu-1511-1-pn-central-processing-unit-with-working-memory-150-kb-for-program-and-1-mb-for-data-1-interface-profinet-irt-with-2-po-6es7511-1ak02-0ab0> (visitado 27-08-2022).
- [6] *Siemens SIMATIC S7-1500 DI 32x24VDC BA - 6ES7521-1BL10-0AA0*, en. dirección: <https://marketplace.realpars.com/en-us/products/simatic-s7-1500-digital-input-module-di-32x24-v-dc-ba-32-channels-in-groups-of-16-input-delay-typ-3-2ms-input-type-3-iec-61131-incl-push-in-6es7521-1bl10-0aa0> (visitado 27-08-2022).
- [7] *Siemens SIMATIC S7-1500 DQ 32x24VDC/0.5A HF - 6ES7522-1BL01-0AB0*, en. dirección: <https://marketplace.realpars.com/en-us/products/simatic-s7-1500-digital-output-module-dq-32x24-v-dc-0-5-a-hf-32-channels-in-groups-of-8-4-a-per-group-single-channel-diagnostics-substitute-value-6es7522-1bl01-0ab0> (visitado 27-08-2022).
- [8] T. Gross, K. Anderson y N. Tsuchiya, «Programmable Automation Controller Mechatronic Experiment,» *IAES International Journal of Robotics and Automation (IJ-RA)*, vol. 6, pág. 39, mar. de 2017. DOI: 10.11591/ijra.v6i1.pp39-48. dirección: https://www.researchgate.net/publication/332579079_Programmable_Automation_Controller_Mechatronic_Experiment.
- [9] Mr_crazy61, Reddit Post, sep. de 2020. dirección: www.reddit.com/r/trainsimworld/comments/ikzjee/horn_sequencer_on_ac4400cw/ (visitado 01-09-2022).
- [10] Albeitt, Reddit Post, mayo de 2022. dirección: www.reddit.com/r/electricians/comments/urd28x/here_is_a_siemens_climate_control_plc_cabinet_i/ (visitado 01-09-2022).
- [11] 783525-03 *Kit de Desarrollo*, es. dirección: <https://es.farnell.com/ni/783525-03/placa-des-zynq-7020-fpga-arm-cortex/dp/3620527> (visitado 27-08-2022).
- [12] National Instruments, *SPECIFICATIONS NI sbRIO-9651h*, <https://www.farnell.com/datasheets/3750667.pdf>, Accessed: 2022-09-01, ago. de 2015.
- [13] SCHWEITZER ENGINEERING LABORATORIES, *SEL-2240 Axion*, <https://selinc.com/api/download/10595/>, Accessed: 2022-09-01, jul. de 2022.
- [14] *Investing in an RTU*. dirección: <https://www.dpstele.com/insights/2019/06/25/why-price-remote-terminal-unit-rtu-worth-upfront-cost/> (visitado 27-08-2022).

-
- [15] *SEL-2440 Controlador discreto de automatización programable*. dirección: <https://selinc.com/es/products/2440/> (visitado 27-08-2022).
- [16] en-US. dirección: <https://www.johnhcarter.com/products/measurement-instrumentation/fabricated-solutions/panel-systems/remote-terminal-unit-rtu-panels/> (visitado 01-09-2022).
- [17] *Siemens SIMATIC HMI KTP700 Basic color PN- 6AV2123-2GB03-0AX0*, en. dirección: <https://marketplace.realpars.com/collections/hmi-panel/products/simatic-hmi-ktp700-basic-basic-panel-key-touch-operation-7-tft-display-65536-colors-profinet-interface-configurable-from-wincc-basic-v13-step-6av2123-2gb03-0ax0> (visitado 01-09-2022).
- [18] Dirección: <https://www.holivera.es/home/CUADRO-EL%20CENTRICO-CON-ORDENADOR-CLIMATICO-SAI-Y-SINOPTICO-PERSONALIZABLE-p303470163> (visitado 01-09-2022).
- [19] Dirección: <https://ph.parker.com/mx/es/on-line-sensor-suite-hmi-parker-kittiwake> (visitado 01-09-2022).
- [20] *Robotics Simulation: simulation of complex conveyor systems in CoppeliaSim*, es-ES. dirección: <https://www.youtube.com/watch?v=FmMEJctOgkM> (visitado 01-09-2022).
- [21] *MHJ-Software*. dirección: <https://www.mhj-tools.com/?page=plc-lab> (visitado 01-09-2022).
- [22] *Release Notes - FACTORY I/O*. dirección: <https://docs.factoryio.com/release-notes/#version-230-20180615> (visitado 01-09-2022).
- [23] *Documentación didáctica/para cursos de formación*, es. dirección: <https://www.automation.siemens.com/sce-static/learning-training-documents/tia-portal/advanced-programming-s7-1200/sce-051-201-scl-s7-1200-r1709-es.pdf> (visitado 05-09-2022).
- [24] C. R. Fernández, *Unit 5: STRUCTURED DESIGN. Guide D'Étude des Modes de Marches et d'Arrêts (GEMMA) <Study Guide of the Start and Stop Modes* (Industrial Automation). Valencia, España: Universidad Politécnica de Valencia, 2021.
- [25] S. García-Nieto, *Introducción a Grafset* (Instalaciones de Control Industrial (12174)). Valencia, España: Universidad Politécnica de Valencia, 2022.