



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCUELA TÉCNICA
SUPERIOR INGENIERÍA
INDUSTRIAL VALENCIA

TRABAJO FIN DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

DESARROLLO DE LA AUTOMATIZACIÓN DE UN PROCESO DE CLASIFICACIÓN DE CAJAS DE UNA EMPRESA LOGÍSTICA CON IMPLEMENTACIÓN MEDIANTE LA HERRAMIENTA DE PROGRAMACIÓN IEC 61131-3 CODESYS Y CON SIMULACIÓN DE LA PLANTA A TRAVÉS DEL SOFTWARE FACTORY I/O

AUTOR: RAMÓN PRADES BATALLA

TUTOR: RAÚL SIMARRO FERNÁNDEZ

Selección NOMBRE DEL COTUTOR

Curso Académico: 2019-20



AGRADECIMIENTOS

Me gustaría agradecer a Raúl Simarro Fernández por no solo haberme guiado y ayudado a lo largo de todo el proyecto, alimentando mi interés por la automatización e instruyendo en mí conocimientos que seguro serán valiosos en mi futuro laboral, sino también por haber aceptado la tutoría de este proyecto pese a las condiciones en las que se lo pedí y haber sido uno de los mejores profesores con los que me he topado.

Por otro lado quiero agradecer a mi familia ya que no es sencillo desarrollar un proyecto de estas dimensiones enteramente desde casa y con una pandemia global, y sin su apoyo y paciencia no creo que lo hubiera logrado.



RESUMEN

Los objetivos del proyecto son el diseño y desarrollo de un sistema de clasificación y organización de cajas de distintos tamaños de una empresa logística. La automatización de los procesos que intervienen se ha realizado mediante la herramienta de programación CODESYS, que utiliza el estándar internacional IEC 61131-3. La monitorización e interacción de los operarios con los diferentes procesos se ha efectuado a través de una aplicación de visualización, alojada en el servidor web que trae embebido el PLC de CODESYS. Para probar el funcionamiento del automatismo se ha realizado una virtualización de la planta mediante el software Factory I/O, que permite la validación de la implementación del programa sobre el PLC.

Las tareas realizadas en el proyecto han sido:

- Diseño en planta de los procesos involucrados mediante el software de simulación Factory I/O, que permitirá evaluar el funcionamiento de la automatización del sistema con los PLC's involucrados mediante comunicación Modbus/TCP.
- Diseño de la automatización de los procesos involucrados.
- Implementación del programa sobre el autómata
- Evaluación del automatismo sobre el software de simulación Factory I/O.
- Realización de una aplicación SCADA con la visualización del PLC, distribuido por el servidor web, para monitorizar y controlar el proceso.

Palabras clave: CODESYS; Automatización; Monitorización; SCADA; PLC; Simulación; Modbus/TCP; Factory I/O ; Visualizador Web;



ABSTRACT

The objectives of the project are the design and development of an automatic system capable of classify and organize boxes of different shape on a logistics. The automatic of the processes that take part has been solved by the programming tool named CODESYS, which uses the IEC 61131-c international standard. The monitoring and interaction of the workers with the different processes have been carried out by a display application located in a web server included in the CODESYS' PLC. In order to prove the correct functioning of the automatism a plant a demonstration has taken place using FACTORY I/O, a business simulation tool which allows us to validate the implementation of the software on the PLC.

The tasks made are:

- Plant design of the processes involved through the use of the simulation software Factory I/O that will allow us to evaluate the correct functioning of the system's automation with the involved PLCs using Modbus/TCP communication.
- Design of the automation of the involved processes.
- Implementation of the program on the automaton.
- Evaluation of the automatism using simulation tool Factory I/O.
- Creation of a SCADA application that permits the PLC's display, distributed by web server in order to monitor and control the process.

Key words: CODESYS; Automation; Monitoring; SCADA; PLC; Simulation; Modbus/TCP; Factory I/O, Web display;



ÍNDICE

DOCUMENTOS CONTENIDOS EN EL PROYECTO

- DOCUMENTO N1: MEMORIA
- ANEXO I: DISEÑO DEL SISTEMA AUTOMÁTICO EN GRAFCET
- ANEXO II: VARIABLES
- ANEXO III: MANUAL DE USUARIO
- DOCUMENTO N2: PRESUPUESTOS

ÍNDICE DOCUMENTO N1: MEMORIA

I.INTRODUCCIÓN	15
1.1 - OBJETIVO DEL PROYECTO	15
1.2 - ANTECEDENTES	15
1.3 - MOTIVACIÓN	16
1.4 – NORMATIVA REQUERIDA.....	16
1.5 – PRESTACIONES Y ESPECIFICACIONES FUNCIONALES DEL AUTOMATISMO REQUERIDAS POR EL CLIENTE.....	17
2.ESTUDIO DE ALTERNATIVAS.....	18
2.1 – TIPOS DE LÓGICA	18
2.1.1 – LÓGICA CABLEADA.....	18
2.1.2 – LÓGICA PROGRAMADA	19
2.2 – DISPOSITIVO DE CONTROL.....	21
2.2.1 – MICROCONTROLADOR.....	21
2.2.2 – ATÓMATA PROGRAMABLE PLC	22
2.3 – SOFTWARE	23
2.4 – SOLUCIONES PARA DISEÑAR EL SISTEMA HMI.....	23
2.4.1 – PANTALLA TÁCTIL	23
2.4.2 – SERVIDOR WEB	24
2.5 – ANÁLISIS DEL PROYECTO.....	25

2.5.1 – ANÁLISIS TEÓRICO	25
2.5.2 – ANALISIS REAL	25
2.5.3 – ANALISIS POR SOFTWARE	25
3.JUSTIFICACIÓN DE LA SOLUCIÓN ADOPTADA.....	26
3.1 - ELEMENTOS DEL PROYECTO	27
3.2 - FUNCIONAMIENTO DEL SISTEMA	29
4.DESCRIPCIÓN DETALLADA DE LA SOLUCIÓN ADOPTADA.....	32
4.1 – CONEXIONES DE LOS ELEMENTOS.....	32
4.2 - DISEÑO DEL PROCESO PRODUCTIVO	33
4.2.1 - ZONA 0: ZONA GENERAL.....	34
4.2.1.1 – EMERGENCIAS.....	34
4.2.1.2 - CINTA-CENTRAL	36
4.2.1.3 – DOOR.....	37
4.2.2 – ZONA A: PICKING PREFORMA.....	38
4.2.2.1 – CINTA-BELT-1	38
4.2.2.2 – CINTA-ROLLER-1.....	39
4.2.2.3 – TAM-SEL-1	40
4.2.2.4 – MANUAL1 Y MANUAL2	41
4.2.2.5 – ACUMULADOR-MANUAL1 Y MANUAL-2.....	42
4.2.2.6 – SELECTOR.....	44
4.2.3 – ZONA B: ZONA PALETIZADOR	44
4.2.3.1 – CINTA-BELT-PALETIZADOR.....	45
4.2.3.2 – CINTA-ROLLER-ELEVATOR	46
4.2.3.3 – CLAMP	47
4.2.3.4 – ELEVATOR	48
4.2.3.5 – TAM-SEL-2	49
4.2.3.6 – PALETIZADOR-DIREC	50
4.2.4 – ZONA C: ZONA PICKING ALEATORIO.....	51
4.2.4.1 – ROLLER-AUTOMATICO	51
4.2.4.2 – TAM-SEL-3	51
4.2.4.3 – GRUA-AUTO	52

4.3 – IMPLEMENTACIÓN DEL SISTEMA EN CODESYS	54
4.3.1 – DECLARACIÓN DE VARIABLES.....	54
4.3.2 – LENGUAJES DE PROGRAMACIÓN	55
4.3.3 – IMPLEMENTACIÓN DE LA COMUNICACIÓN CODESYS-FACTORY IO	59
4.3.3.1 – ETHERNET TCP/IP	59
4.3.3.2 – MODBUS TCP	60
4.3.4 – IMPLEMENTACIÓN DE LA INTERFAZ DE USUARIO.....	62
4.3.4.1 – ELEMENTOS FUNCIONALES	63
4.3.4.2 – ELEMENTOS DECORATIVOS	66
5.CONCLUSIONES.....	69
6.BIBLIOGRAFÍA.....	71

ANEXO I: DISEÑO DEL SISTEMA AUTOMÁTICO EN GRAFCET

1 – OBJETO.....	77
2 – ZONA 0: ZONA GENERAL.....	77
2.1 – EMERGENCIAS	77
2.2 – CINTA_CENTRAL	77
2.3 – DOOR	78
3 – ZONA A: PICKING PREFORMA	78
3.1 – CINTA_BELT_1	78
3.2 – CINTA_ROLLER_1.....	79
3.3 – TAM_SEL_1	79
3.4 – MANUAL1.....	80
3.5 – MANUAL2.....	81
3.6 – ACUMULADOR_MANUAL_1.....	82
3.7 – MANUAL_2.....	83
3.8 – SELECTOR	84
4 – ZONA B: ZONA PALETIZADOR.....	85
4.1 – CINTA_BELT_PALETIZADOR.....	85
4.2 – CINTA_ROLLER_ELEVATOR	86



4.3 – CLAMP.....	86
4.4 – ELEVATOR.....	87
4.5 – TALM_SEL_2.....	88
4.6 – PALETIZADOR_DIRECT.....	88
5 – ZONA C: ZONA PICKING ALEATORIO	89
5.1 – ROLLER_AUTOMATICO	89
5.2 – TAM_SEL_3	89
5.3 – GRUA_AUTO.....	90

ANEXO II: VARIABLES

1 – OBJETO.....	95
2 – VARIABLES DE ENTRADA.....	95
2.1 – VARIABLES BOOLEANAS.....	95
2.2 – VARIABLES NO BOOLEANAS.....	96
3 – VARIABLES DE SALIDA.....	97
3.1 – VARIABLES BOOLEANAS.....	97
3.2 – VARIABLES NO BOOLEANAS.....	99

ANEXO III: MANUAL DE USUARIO

1 – OBJETIVO.....	103
2 – PUESTA EN FUNCIONAMIENTO	103
3 – ACCESO AL SERVIDOR WEB	105
4 – INSTRUCCIONES DE USO.....	107

ÍNDICE DOCUMENTO N2: PRESUPUESTOS

1.INTRODUCCIÓN	112
2.CUADRO DE PRECIOS ELEMENTALES	113
2.1 – CUADRO N1: MANO DE OBRA.....	113



2.2 – CUADRO N2: MATERIALES Y MAQUINARIA	113
3.CUADRO DE PRECIOS UNITARIOS.....	114
4.CUADRO DE PRECIOS DESCOMPUESTOS.....	115
5.PRESUPUESTO BASE DE LICITACIÓN.....	117



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCOLA TÈCNICA
SUPERIOR ENGINYERIA
INDUSTRIAL VALÈNCIA

TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

DOCUMENTO I: MEMORIA

Autor: Ramón Prades Batalla

Tutor: Raúl Simarro Fernández

Curso académico: 2019/2020





ÍNDICE DE CONTENIDO DE LA MEMORIA

I.INTRODUCCIÓN	15
1.1 - OBJETIVO DEL PROYECTO	15
1.2 - ANTECEDENTES	15
1.3 - MOTIVACIÓN	16
1.4 – NORMATIVA REQUERIDA.....	16
1.5 – PRESTACIONES Y ESPECIFICACIONES FUNCIONALES DEL AUTOMATISMO REQUERIDAS POR EL CLIENTE.....	17
2.ESTUDIO DE ALTERNATIVAS	18
2.1 – TIPOS DE LÓGICA	18
2.1.1 – LÓGICA CABLEADA.....	18
2.1.2 – LÓGICA PROGRAMADA	19
2.2 – DISPOSITIVO DE CONTROL.....	21
2.2.1 – MICROCONTROLADOR.....	21
2.2.2 – ATÓMATA PROGRAMABLE PLC	22
2.3 – SOFTWARE	23
2.4 – SOLUCIONES PARA DISEÑAR EL SISTEMA HMI.....	23
2.4.1 – PANTALLA TÁCTIL	23
2.4.2 – SERVIDOR WEB.....	24
2.5 – ANÁLISIS DEL PROYECTO.....	25
2.5.1 – ANÁLISIS TEÓRICO	25
2.5.2 – ANALISIS REAL	25
2.5.3 – ANALISIS POR SOFTWARE	25
3.JUSTIFICACIÓN DE LA SOLUCIÓN ADOPTADA	26
3.1 - ELEMENTOS DEL PROYECTO	27
3.2 - FUNCIONAMIENTO DEL SISTEMA	29
4.DESCRIPCIÓN DETALLADA DE LA SOLUCIÓN ADOPTADA	32
4.1 – CONEXIONES DE LOS ELEMENTOS.....	32
4.2 - DISEÑO DEL PROCESO PRODUCTIVO	33
4.2.1 - ZONA 0: ZONA GENERAL.....	34
4.2.1.1 – EMERGENCIAS.....	34

4.2.1.2 - CINTA-CENTRAL	36
4.2.1.3 – DOOR	37
4.2.2 – ZONA A: PICKING PREFORMA.....	38
4.2.2.1 – CINTA-BELT-1	38
4.2.2.2 – CINTA-ROLLER-1.....	39
4.2.2.3 – TAM-SEL-1	40
4.2.2.4 – MANUAL1 Y MANUAL2	41
4.2.2.5 – ACUMULADOR-MANUAL1 Y MANUAL-2	42
4.2.2.6 – SELECTOR.....	44
4.2.3 – ZONA B: ZONA PALETIZADOR	44
4.2.3.1 – CINTA-BELT-PALETIZADOR.....	45
4.2.3.2 – CINTA-ROLLER-ELEVATOR	46
4.2.3.3 – CLAMP	47
4.2.3.4 – ELEVATOR	48
4.2.3.5 – TAM-SEL-2	49
4.2.3.6 – PALETIZADOR-DIREC	50
4.2.4 – ZONA C: ZONA PICKING ALEATORIO	51
4.2.4.1 – ROLLER-AUTOMATICO	51
4.2.4.2 – TAM-SEL-3	51
4.2.4.3 – GRUA-AUTO.....	52
4.3 – IMPLEMENTACIÓN DEL SISTEMA EN CODESYS	54
4.3.1 – DECLARACIÓN DE VARIABLES.....	54
4.3.2 – LENGUAJES DE PROGRAMACIÓN	55
4.3.3 – IMPLEMENTACIÓN DE LA COMUNICACIÓN CODESYS-FACTORY IO	59
4.3.3.1 – ETHERNET TCP/IP.....	59
4.3.3.2 – MODBUS TCP	60
4.3.4 – IMPLEMENTACIÓN DE LA INTERFAZ DE USUARIO.....	62
4.3.4.1 – ELEMENTOS FUNCIONALES	63
4.3.4.2 – ELEMENTOS DECORATIVOS	66
5.CONCLUSIONES.....	69
6.BIBLIOGRAFÍA.....	71

ÍNDICE DE FIGURAS DE LA MEMORIA

Ilustración 1 Automatismo industrial implementado mediante lógica cableada [3]	19
Ilustración 2 Automatismo Industrial implementado mediante lógica programada [4]	20
Ilustración 3 Ejemplo de microcontrolador [5]	21
Ilustración 4 Ejemplo de PLC Haiwell T24S2R [6]	22
Ilustración 5 Pantalla táctil PLC Vision570 [11]	24
Ilustración 6 Tablet con visualizador Web [12]	24
Ilustración 7; Imagen propia, vista aérea del proceso al completo en el software de simulación <i>FACTORY IO</i>	26
Ilustración 8; Imagen propia, vista aérea del proceso simplificado en el software de simulación <i>FACTORY IO</i>	29
Ilustración 9; Imagen propia, esquema de las distintas zonas en el software de simulación <i>FACTORY IO</i>	31
Ilustración 10; Imagen propia, esquema de los distintos elementos que toman parte y sus conexiones.....	32
Ilustración 11; Imagen propia, distribución de los sensores principales vistos desde el <i>FACTORY IO</i>	33
Ilustración 12; Imagen propia, GRAFCET EMERGENCIAS	34
Ilustración 13; Imagen propia, cuadro de mando general	35
Ilustración 15; Imagen propia, GRAFCET CINTA_CENTRAL	36
Ilustración 16; Imagen propia, GRAFCET DOOR	37
Ilustración 17; Imagen propia, GRAFCET CINTA_BELT_1	38
Ilustración 18; Imagen propia, GRAFCET CINTA_ROLLER_1	39
Ilustración 19; Imagen propia, GRAFCET TAM_SEL_1	40
Ilustración 20; Imagen propia, GRAFCET MANUAL1.....	41
Ilustración 21; Imagen propia, GRAFCET ACUMULADOR_MANUAL1	42
Ilustración 23; Imagen propia, GRAFCET SELECTOR	44
Ilustración 24; Imagen propia, GRAFCET CINTA_BELT_PALETIZADOR	45
Ilustración 25; Imagen propia, GRAFCET CINTA_ROLLER_ELEVATOR	46
Ilustración 26; Imagen propia, GRAFCET CLAMP.....	47
Ilustración 27; Imagen propia, GRAFCET ELEVATOR.....	48
Ilustración 28; Imagen propia, GRAFCET PLAETIZADOR_DIREC	50
Ilustración 29; Imagen propia, GRAFCET ROLLER_AUTOMATICO	51
Ilustración 30; Imagen propia, GRAFCET GRUA_AUTO.....	52
Ilustración 31; Imagen propia macroetapa M8.....	53
Ilustración 32; Imagen propia, pestaña con las variables globales en <i>CODESYS</i>	54
Ilustración 33; Imagen propia, estructura de los POU's en el <i>CODESYS</i>	55
Ilustración 34; Imagen propia, ejemplo del lenguaje ST en <i>CODESYS</i>	56
Ilustración 35; Imagen propia, ejemplo del lenguaje LD en <i>CODESYS</i>	56
Ilustración 36; Imagen propia, comparación GRAFCET.....	58

Ilustración 38; Imagen propia, esquema de elementos y conexiones que intervienen en el sistema	59
Ilustración 39; Imagen propia, distribución de las variables y dirección en <i>CODESYS</i>	60
Ilustración 40; Imagen propia, esquema de variables en el <i>FACTORY IO</i>	61
Ilustración 41; Imagen propia, ruta para agregar la pantalla de operador	62
Ilustración 42; Imagen propia, explicación de las pestañas de la pantalla de operador	63
Ilustración 43; Imagen propia, selección de propiedades de los elementos de la pantalla de operador.....	64
Ilustración 44; Imagen propia, tipos de LEDs que ofrece la pantalla de operador	64
Ilustración 45; Imagen propia, interruptor de energía de la pantalla de operador	65
Ilustración 46; Imagen propia, interruptor de presión de la pantalla de operador	65
Ilustración 47; Imagen propia, indicaciones para mostrar variables mediante texto	66
Ilustración 48; Imagen propia, ruta para añadir imágenes externas al software	67
Ilustración 49; Imagen propia, diseño final de la pantalla de operador del proyecto.....	68

ÍNDICE DE TABLAS DE LA MEMORIA

Tabla 1; Elaboración propia, Tabla ventajas y desventajas de la lógica cableada.....	19
Tabla 2; Elaboración propia, Tabla ventajas y desventajas de la lógica programada.....	20
Tabla 3; Elaboración propia, Tabla ventajas y desventajas de los microprocesadores	21
Tabla 4; Elaboración propia, Tabla ventajas y desventajas de los autómatas programables PLC.....	22
Tabla 5; Elaboración propia, explicación y visualización de los elementos del <i>FACTORY IO</i>	29

I. INTRODUCCIÓN

1.1 - OBJETIVO DEL PROYECTO

El objetivo de este trabajo es la automatización un proceso de clasificación para que una empresa logística sea capaz de distribuir su mercancía en de cajas de tres tamaños diferentes con la mayor eficiencia posible mediante distintas subestaciones: grúa con preforma capaz de gestionar distintas configuraciones de pallets, paletizadora automática y grúa aleatoria capaz de formar pallets con independencia del orden de llegada de las cajas.

El segundo objetivo será la visualización, monitorización e interacción de los procesos. Mediante este visualizador seremos capaces de no solo observar el estado de la planta en todo momento sino también de alterar o manipular otra serie de elementos cuya variación pueda resultar de interés, permitiéndonos un mayor control sobre todo el proceso.

1.2 - ANTECEDENTES

En la actualidad la automatización juega un papel fundamental en nuestra vida, no solo a nivel de usuario, por ejemplo el control de gran cantidad de elementos del hogar desde una aplicación móvil, sino también en la industria, y podemos afirmar sin temor a equivocarnos que su presencia aumentará más y más en los años venideros. Y parece lógico que así sea, puesto que los avances en la automatización permiten optimizar los procesos de fabricación, reduciendo costes, mejorando la eficiencia, incrementando la precisión, etc. En resumen, aportan todo aquello que una empresa necesita para mantenerse en el mercado y poder ser competitiva.

A su vez, la automatización nos permite en muchas ocasiones delegar operaciones de riesgo que podrían poner en peligro la vida de los operarios a los autómatas, reduciendo significativamente el factor de riesgo y mejorando las condiciones laborales.

Es por ello que la automatización es un campo imprescindible en la vida moderna, y no solo por las grandes mejoras económicas que genera, sino también por el sinfín de posibilidades para mejorar nuestro nivel de vida que aporta.

Para más detalles acerca de la gran extensión de la automatización en la actualidad se puede consultar la información acerca de la Cuarta Revolución Industrial [1].

1.3 - MOTIVACIÓN

El principal motivo por el cual he escogido este TFG es la fascinación que me genera el sinfín de posibilidades que ofrece la automatización. El hecho de poder realizar tareas titánicas como el movimiento de piezas de cientos de quilos, tareas que requieran de una precisión milimétrica como cortes exactos, o tareas que requieran de una coordinación inimaginable como el funcionamiento de un almacén automático, con un solo botón, es suficiente para incitarme a conocer más de este campo y es por ello que he elegido esta temática para mi trabajo.

Así mismo considero que es una disciplina que estará presente en cualquier industria y por ello creo que es imprescindible empezar mi formación en este ámbito, para poder aportar en un futuro los conocimientos y las soluciones apropiadas en cada situación, independientemente de la tarea que el autómatas vaya a realizar, así como de la dificultad que esta plantee.

Por último, y siguiendo la recomendación de mi tutor, decidí aprender sobre CODESYS, un entorno de programación de controladores bastante popular en la actualidad y con un gran abanico de herramientas, para poder así conocer otro entorno distinto al usado en el aula y mejorar mis conocimientos en este campo.

1.4 – NORMATIVA REQUERIDA

Para la realización de este proyecto de diseño y automatización, es necesario prestar atención a la siguiente normativa vigente en España:

- **UNE-EN IEC 61131:2019:** Estándar Internacional para controladores lógicos programables (PLC). [2]
- **UNE-EN 60848:2013:** Lenguaje de especificación GRAFCET para diagramas funcionales secuenciales. [2]
- **UNE-EN 61508:2011:** Seguridad funcional de los sistemas eléctricos/electrónicos/electrónicos programables relacionados con la seguridad. [2]
- **UNE-EN 60870-5-101:2003:** Norma para la monitorización de sistemas de energía, control y las comunicaciones asociadas a los mismos. [2]
- **UNE-EN ISO 13850:2016:** Seguridad de las máquinas. Función de parada de emergencia. Principios para el diseño. [2]
- **UNE-EN 62439:2018:** Redes de comunicación industrial. Redes de automatización de alta disponibilidad. [2]

1.5 – PRESTACIONES Y ESPECIFICACIONES FUNCIONALES DEL AUTOMATISMO REQUERIDAS POR EL CLIENTE

Nuestro cliente “XXXXX” solicita al equipo de ingeniería que diseñe y desarrolle un proyecto siguiendo las siguientes especificaciones:

- El proyecto se realizará en una nueva nave, por lo que se dispondrá de todo el espacio.
- Se requiere que la formación de los pallets se ajuste a tres tamaños distintos de cajas y por tanto se requiere un sistema para clasificar las cajas.
- Todas las zonas deberán actuar con independencia entre sí para minimizar las pérdidas.
- En la zona del picking con preforma se exigen dos posibilidades distintas.
- La zona donde se encuentra el paletizador deberá admitir únicamente cajas de tamaño pequeño
- La zona de picking automático deberá conformar pallets con independencia del tamaño de la caja y del orden de llegada.
- Se exige un contador que indique el número de pallets generados en cada zona.
- Se necesita un botón de parada junto a cada estación, así como uno de mantenimiento para cada una de ellas.
- Se pide un sistema de alarmas y luces que alerte del estado de la planta en todo momento.
- Se exige que la interfaz de monitorización y control sea lo más sencilla posible para facilitar la comprensión a los operarios.

2. ESTUDIO DE ALTERNATIVAS

Uno de los principales resultados buscados a la hora de realizar un proyecto es obtener la solución más eficiente, tanto a nivel económico como técnico, dentro de las posibilidades y limitaciones que se plantean. Es por ello que resulta de gran importancia realizar un análisis exhaustivo de las diferentes alternativas que existen.

2.1 – TIPOS DE LÓGICA

A la hora de establecer el funcionamiento de un sistema automático se puede optar por dos tipos de lógicas.

2.1.1 – LÓGICA CABLEADA

La lógica cableada o lógica de contactos es una forma de realizar controles en la que el tratamiento de datos se efectúa en conjunto con contactores o relés auxiliares, frecuentemente asociados a temporizadores y contadores.

La principal ventaja que presenta este tipo de lógica es su bajo coste de fabricación y implementación. Así mismo, el tiempo de reacción, es decir, el tiempo que tarda el sistema en reaccionar ante una orden impuesta, de este tipo de sistemas es realmente bajo (del orden de nanosegundos), puesto que el retardo solo viene impuesto por los propios componentes electrónicos. Esto puede resultarnos realmente útil en sistemas que requieran de tiempos de respuesta muy bajos y precisos. Por último estos sistemas pueden resultarnos útiles en instalaciones pequeñas donde la seguridad de las personas y máquinas no pueda depender de la lógica programada.

Por el contrario, estos sistemas ocupan mucho espacio físico, sin contar que cualquier cambio que se quiera realizar en la programación de la instalación pasará por modificar el cableado y los elementos hasta que se cumplan las funciones especificadas. Estos sistemas también presentan limitaciones en cuanto a complejidad de contenido, y son incapaces de realizar funciones muy complejas.



Ilustración 1 Automatismo industrial implementado mediante lógica cableada [3]

TIPO DE LÓGICA	VENTAJAS	INCONVENIENTES
Lógica cableada	Bajo coste de creación e instalación	Ocupa mucho espacio
	Permite operar el plantas donde sea de extrema importancia la seguridad del personal y maquinaria	No permite cambios en la programación (al menos no de forma sencilla)
	Gran tiempo de respuesta	Limitación frente a funciones complejas

Tabla 1; Elaboración propia, Tabla ventajas y desventajas de la lógica cableada

2.1.2 – LÓGICA PROGRAMADA

Logica programada es lo contrario de la lógica cableada, en esta se sustituyen los elementos utilizados en los circuitos de mando (contactos auxiliares de relés electromecánicos, contactores de potencia, relés temporizados, relés contadores, etc.) por PLC's, Autómatas Programables o Relés programables. Esto nos permite realizar cambios en las operaciones de mando, mediante el cambio de la programación, y por ello no tener que modificar el cableado. Aunque los fundamentos de la programación de la lógica programada son similares, cada fabricante utiliza una nomenclatura y un software específico para ello. Según la norma IEC, nos podremos encontrar con los lenguajes KOP (conocido como lenguaje de contactos), FUP (conocido como lenguaje de puertas lógicas) y AWL (conocido como lenguaje en modo texto).

En cuanto a las ventajas de este tipo de lógica son la mayor flexibilidad y facilidad para modificar elementos de la programación, la mayor capacidad para interpretar y realizar funciones complejas, y la capacidad que aporta de conexión con multitud de dispositivos.

Por otro lado los inconvenientes son el elevado precio que esta supone, la necesidad de personal cualificado para su instalación, mantenimiento y modificación y la falta de robustez en aspectos tanto de seguridad como de tiempo de respuesta, aunque estos mejoran diariamente.



Ilustración 2 Automatismo Industrial implementado mediante lógica programada [4]

TIPO DE LÓGICA	VENTAJAS	INCONVENIENTES
Lógica programada	Mayor flexibilidad antes cambios en la programación	Elevado coste
	Mayor capacidad de trabajo con funciones complejas	Necesidad de personal cualificado para su manipulación
	Conectividad con multitud de dispositivos	Falta de robustez en seguridad y tiempo de respuesta

Tabla 2; Elaboración propia, Tabla ventajas y desventajas de la lógica programada

2.2 – DISPOSITIVO DE CONTROL

Son elementos encargados de administrar, ordenar, dirigir o regular el comportamiento de un sistema, con el fin de reducir las probabilidades de fallo y obtener los resultados deseados.

2.2.1 – MICROCONTROLADOR

Un microcontrolador es un circuito integrado programable, capaz de ejecutar las órdenes grabadas en su memoria. Está compuesto de varios bloques funcionales que cumplen una tarea específica. Un microcontrolador incluye en su interior las tres principales unidades funcionales de una computadora: unidad central de procesamiento, memoria y periféricos de entrada/salida.

La principal ventaja de los microcontroladores es que incluyen todos los elementos necesarios para llevar a cabo el control de un proceso a un coste bajo, pero son equipos que no están orientados a trabajar en un ambiente industrial, debido a la complejidad de estos.

Otra de las desventajas de un microcontrolador es que es necesaria un mayor conocimiento en el tipo de programación dado a todas las instrucciones que son necesarias para realizar una tarea en particular.

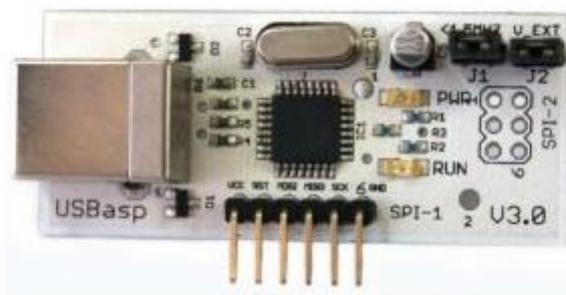


Ilustración 3 Ejemplo de microcontrolador [5]

TIPO DE DISPOSTIVO	VENTAJAS	INCONVENIENTES
MICROCONTROLADOR	Capacidad de emplear el tiempo como variable para sincronizar tareas	Dificultad ante tareas complejas
	Gran variedad de microcontroladores y de softwares para su programación	Requiere de periféricos para controlar un sistema
	Bajo consumo	Complejidad en la programación

Tabla 3; Elaboración propia, Tabla ventajas y desventajas de los mircoprocesadores

2.2.2 – ATÓMATA PROGRAMABLE PLC

Un controlador lógico programable, más conocido por sus siglas en inglés PLC (Programmable Logic Controller) o por autómata programable, es una computadora utilizada en la ingeniería automática o automatización industrial, para automatizar procesos de producción.

Dentro de las ventajas que ofrece el uso de estos dispositivos se encuentra la posibilidad de ahorrar tiempo en la elaboración de proyectos, pudiendo realizar modificaciones sin costos adicionales. Por otra parte, son de tamaño reducido y mantenimiento de bajo costo, además permiten reducir la mano de obra (aspecto realmente útil si esta se ve expuesta a fenómenos que puedan presentar un riesgo para su salud) y posibilitan el control de más de una máquina con el mismo equipo. Finalmente también permiten soportar las vibraciones mecánicas generadas por la maquinaria ya que otros dispositivos son altamente frágiles o propensos a fallas o rupturas.

En caso de las desventajas, encontramos la necesidad de técnicos altamente cualificados para su instalación y mantenimiento, y el elevado coste que supone su adquisición así como su instalación y manipulación.



Ilustración 4 Ejemplo de PLC Haiwell T24S2R [6]

TIPO DE DISPOSITIVO	VENTAJAS	INCONVENIENTES
PLC	Disponibilidad de gran cantidad de E/S	Mayor coste de adquisición
	Posibilidad de añadir módulos adicionales	Necesidad de acceso y conexión a la red eléctrica
	Permite tanto control directo y como continuo	Espacio de memoria relativamente reducido
	Admite gran cantidad de lenguajes de programación	Mayor tamaño físico
	Permite interconexiones entre dispositivos del mismo tipo y monitorización	Necesidad de técnicos especializados para su instalación y mantenimiento

Tabla 4; Elaboración propia, Tabla ventajas y desventajas de los autómatas programables PLC

2.3 – SOFTWARE

En cuanto a la selección del software de programación, dos opciones destacan sobre el resto: *CODESYS* y *UNITY PRO*.

Por un lado el *UNITY PRO* es una herramienta conocida, sencilla y eficiente, hecho que la vuelve una buena solución de cara a afrontar la realización del proyecto. También es una herramienta bastante extendida por la red, de manera que existen una gran cantidad de tutoriales, y permite la inclusión de una pantalla de operador que ofrece a posibilidad de supervisar el manipular el proceso a nuestro antojo. [7]

Por otro lado la herramienta *CODESYS* ofrecía una gran variedad de lenguajes para la implementación de los distintos GRAFCETs parciales, cumpliendo todos con el estándar industrial internacional IEC 61131-3, así como una interfaz cómoda y una curva de aprendizaje sencilla si se ha trabajado previamente con softwares similares. Sin embargo estas no son las cualidades más destacables de la herramienta, puesto que ofrece una gran variedad de drivers para las distintas marcas de PLCs, lo que permite transferir el proyecto a cualquier autómatas siempre que se haya adquirido el driver mencionado y se referencien las variables de forma adecuada, esto ofrece una gran versatilidad ya que da la opción de programar todo el código con el PLC por defecto del *CODESYS* sin tener que preocuparnos por la futura adaptación al PLC requerido. [8]

Por otro lado, en cuanto a la selección de la herramienta de visualización también encontramos dos softwares que presentan un mejor abanico de posibilidades: *FACTORY IO* y *autoSIM-200*.

Por un lado *autoSIM-200* es una herramienta sencilla e intuitiva que nos permite testear los programas diseñados sobre un entorno virtual de forma gratuita. Permite establecer las conexiones entre los distintos elementos que toman parte en el proyecto de forma sencilla y eficiente, pero presenta ciertas dificultades ante proyectos de gran dimensión. [9]

Por otro lado *FACTORY IO* es una popular herramienta realista de simulación de procesos industriales y entrenamiento de PLCs, perfecta para la comprobación del correcto funcionamiento de un proceso productivo. Sus principales ventajas son la gran cantidad de elementos de los que dispone y el amplio abanico de conexiones que permite (Modbus, OPC, etc.), así como la cantidad de información que se puede obtener online. Otro gran atractivo es que dispone de un extenso manual en línea para solventar cualquier duda. [10]

2.4 – SOLUCIONES PARA DISEÑAR EL SISTEMA HMI

Para el control sencillo de su automatismo, se ha realizado el diseño y la programación de sistemas Human Machine Interface. Estos sistemas han ido avanzando a lo largo del tiempo, siendo cada día más intuitivos en el manejo y visualmente muy atractivos.

2.4.1 – PANTALLA TÁCTIL

Una de las formas de mostrar esta interfaz es mediante una pantalla táctil. Esta opción es realmente atractiva debido al gran auge de los “smartphones” y otros dispositivos táctiles como

tabletas electrónicas, ya que todo el mundo está familiarizado con esta forma de actuar y resulta muy visual. Además de la sencillez, esta variante de HMI dispone de gran potencia, ya que si están conectadas al PLC permiten gobernar sobre todo el sistema cambiando distintas variables e incluso permitiendo operaciones tales como registro de datos, visualización de tendencias, routing remoto, etc.

El principal inconveniente recae en su elevado precio debido a que son complejos elementos electrónicos.



Ilustración 5 Pantalla táctil PLC Vision570 [11]

2.4.2 – SERVIDOR WEB

El autómata programable empleado en la realización del proyecto presenta la posibilidad de crear servidores web a los que se puede acceder desde cualquier navegador de un ordenador, tablet o móvil siempre y cuando estos se encuentren dentro de la misma subred que el PLC. Esta característica es llamativa ya que se puede acceder a la información del sistema desde cualquier dispositivo dentro de la empresa amén que ofrece las llamativas cualidades la pantalla táctil.

Una ventaja de esta opción es que el software mencionado anteriormente, *CODESYS*, genera este tipo de sistema HMI de forma automática, por lo que la sencillez aumenta todavía más.



Ilustración 6 Tablet con visualizador Web [12]

2.5 – ANÁLISIS DEL PROYECTO

El principal objetivo es automatizar un proceso de clasificación de forma que se maximice la producción, por tanto, será necesario realizar un análisis.

2.5.1 – ANÁLISIS TEÓRICO

Este método se basa en utilizar los diferentes parámetros que se obtienen de todos los elementos que forman el proyecto, es decir, todos los elementos de cada proceso y una vez obtenidos, se calcula el tiempo de las distintas actividades que lo aúnan: producción, clasificación, almacenamiento... Por lo tanto, es una solución sencilla y económica, ya que un ingeniero puede realizar dichos , sin embargo no resulta una buena solución ya que el análisis planteado no contempla los diferentes errores o variaciones que puede tener el proceso.

2.5.2 – ANALISIS REAL

A diferencia con el análisis teórico, en este método se montaría parte de los procesos para poder analizar y observar los tiempos de producción, clasificación, almacenamiento... Este método supone realizar una gran inversión inicial sin ninguna garantía de beneficio ni funcionamiento, lo que la convierta en la solución más cara y costosa, pese a ser la que más se aproximaría al modelo final.

2.5.3 – ANALISIS POR SOFTWARE

Este método está basado en un modelo virtual del proyecto, llamado digital twin o gemelo virtual. Esta es una de las tecnologías que está cambiando la dinámica del sector industrial. Se trata de réplicas virtuales de objetos o procesos que simulan el comportamiento de sus homólogos reales. La finalidad que tiene es analizar su eficacia o comportamiento en determinados supuestos para mejorar su eficacia. Por lo tanto, una vez obtenido el mejor resultado, se puede pasar a realizar una implementación real y funcionaria igual que se ve en la simulación. Además, esta simulación por software permite realizar cambios una vez instalada la fábrica con el fin de seguir mejorando y probando nuevas soluciones sin tener que realizar grandes inversiones. [13]

3.JUSTIFICACIÓN DE LA SOLUCIÓN ADOPTADA

Una vez estudiadas todas las alternativas se ha optado por realizar el proyecto empleando el software de programación *CODESYS*. Esto es debido a el gran abanico de recursos que aporta este programa, permitiendo distintos lenguajes de programación todos acordes con el estándar industrial internacional IEC 61131-3. Otra de las razones por las que se ha seleccionado este software es por el atractivo que supone poder realizar todo el proyecto empleando el PLC que trae por defecto *CODESYS* sin temer por futuros problemas de compatibilidad, ya que este ofrece una gran variedad de drivers que permiten la migración del programa a los distintos tipos de PLC que podemos encontrar en el mercado. Recalco la importancia de este último punto ya que debido a la situación ocurrida durante esta primera mitad del año, acceder a un PLC físico era tarea compleja y la mejor alternativa era trabajar con este PLC virtual. En cuanto a la HMI, y como se ha mencionado previamente, este software no solo permite, sino que genera automáticamente un servidor web de gran potencia desde donde poder realizar la visualización, lo cual supuso un atractivo más de cara a decantar la balanza hacia esta opción.

En cuanto a la selección del software encargado de la simulación, *FACTORY IO*, la decisión fue si cabe aún más contundente, debido a la infinidad de recursos que ofrece este programa, a la gran variedad de PLCs que admite, y a que dispone de guías para establecer las conexiones entre ambos programas.

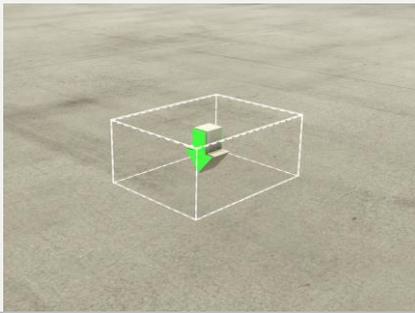
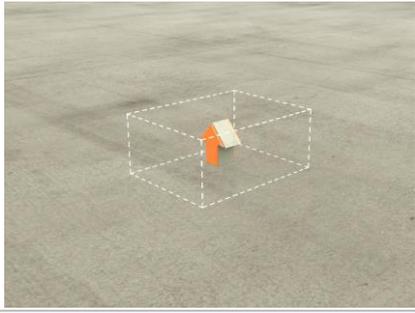
Antes de pasar a la explicación detallada de la solución adoptada, donde se explicará el diseño del proceso en *CODESYS*, se realizará una explicación del *FACTORY IO*, no solo para justificar la selección de este software, sino porque en apartados posteriores se van a hacer referencia a los distintos elementos del proceso productivo, y por tanto conocer de antemano el proceso visualmente y todos sus elementos facilitará la comprensión a posteriori del resto del trabajo.

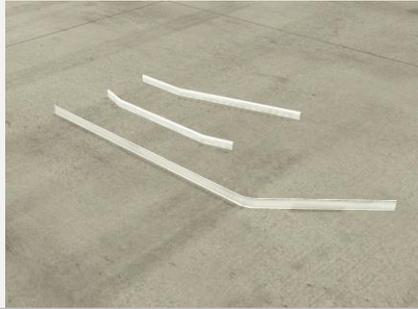
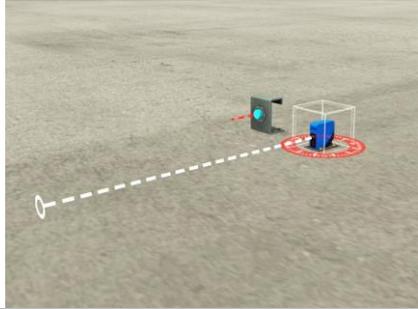


Ilustración 7; Imagen propia, vista aérea del proceso al completo en el software de simulación *FACTORY IO*

3.1 - ELEMENTOS DEL PROYECTO

La planta expuesta es el resultado de la combinación de los distintos elementos que nos ofrece el *FACTORY IO* con la finalidad de poder simular un proceso lo más semejante al resultado real que se quiera obtener. Es por ello que antes de explicar el funcionamiento físico del proceso se deberá conocer cada elemento por separado y que función desempeña de forma individual.

APARIENCIA	NOMBRE	FUNCIÓN
	Emitter	Se encarga de generar y emitir las cajas que se emplearán en el proceso. Permite configurar que tamaños deseamos que emita y la frecuencia de emisión.
	Remove	Se encarga de eliminar cualquier elemento que ingrese en su rectángulo. Se usa principalmente para eliminar pallets o cajas.
	Belt Conveyor	Se encarga de desplazar las cajas. El desplazamiento se efectúa mediante una cinta. Existen variaciones de longitud (6, 4 y 2m) así como inclinados y curvos, pero su función es la misma.
	Roller Conveyor	Se encarga de desplazar pallets. El desplazamiento se efectúa mediante rodillos. Existen variaciones de longitud (6, 4 y 2m) así como inclinados y curvos, pero su función es la misma.

	Conveyor Scale	Es una cinta que permite no solo el desplazamiento de las piezas, sino que también actúa de báscula permitiendo conocer el peso del elemento que la recorre en todo momento.
	Pop Up Wheel Sorter	Es una cinta que permite un giro de 90 ⁰ tanto a la izquierda como a la derecha, así como el desplazamiento en línea recta.
	Aligners	Su único propósito es alinear las cajas para el correcto funcionamiento de las etapas del proceso. No admiten automatización.
	Sensors	Sensores de posición que se activan en cuanto un elemento atraviesa su rango de actuación. Permiten variar tanto el ángulo de incidencia como la distancia a la que llegan.
	Pick & Place	Estación de trabajo que permite mediante un brazo succionador el desplazamiento y reposicionamiento de cajas. Funciona mediante incrementos de coordenadas en los tres ejes.

	<p>Palletizer</p>	<p>Estación de trabajo que permite empujar cajas sobre un techo retráctil. Dispone de un elevador de pallets capaz de colocar el pallet para una correcta colocación de las cajas de la parte superior.</p>
---	--------------------------	---

Tabla 5; Elaboración propia, explicación y visualización de los elementos del *FACTORY IO*

Si se desea una explicación más detallada de las partes se puede obtener en el manual aportado por el software. [14]

3.2 - FUNCIONAMIENTO DEL SISTEMA

Una vez conocidos los distintos elementos que conforman la estructura física del programa, la comprensión del funcionamiento se simplifica, no obstante, para una mayor claridad se ha simplificado el resultado final dejando únicamente aquellos elementos que sean imprescindibles para el funcionamiento del sistema, como las tres subestaciones en las que se basa todo el proceso, las cintas que permiten el suministro de cajas, o los distintos elementos encargados de derivar las mismas. Por tanto se han eliminado los elementos meramente estéticos, y el sistema de seguridad, debido a que no son necesarios para entender como se comporta el sistema. No obstante, estos sí aparecen en el programa final y serán explicados en sus correspondientes apartados.

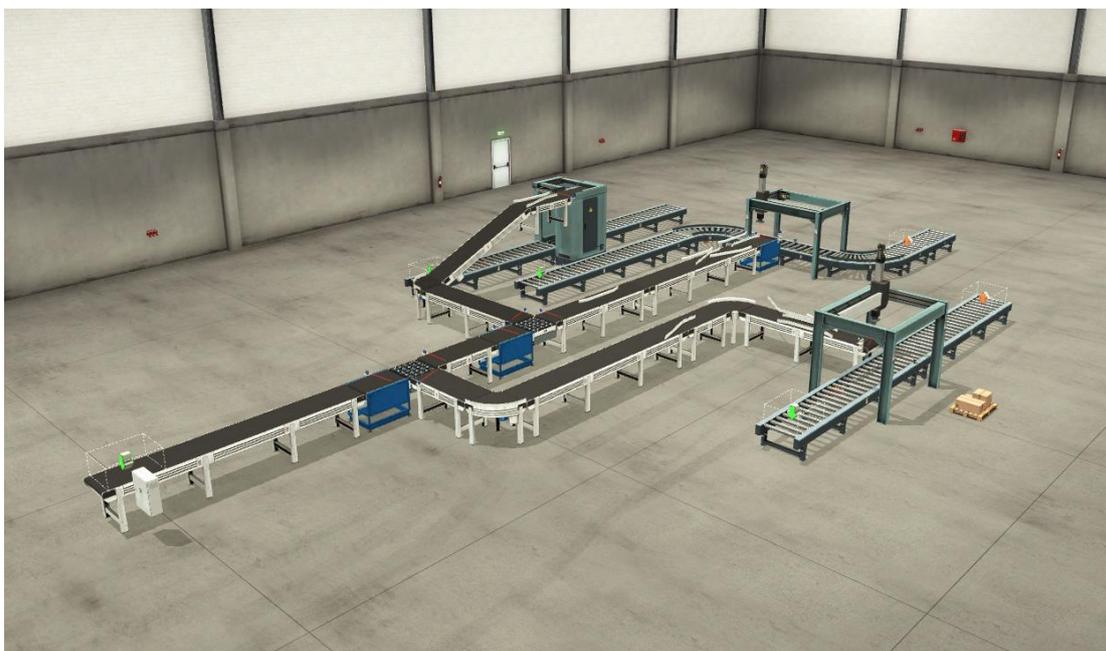


Ilustración 8; Imagen propia, vista aérea del proceso simplificado en el software de simulación *FACTORY IO*

Como podemos observar, se dispone de una cinta principal por la cual viajan las cajas generadas por el *Emitter*. Cabe destacar que se generan piezas de tres tamaños de forma totalmente aleatoria, por tanto no seremos capaces de determinar en que orden van a aparecer. Es por ello que en el final de la primera cinta encontramos el primer *Scale Conveyor*, elemento que nos discretizará el tamaño de la caja en función de su peso. Esta señal será generada en cuanto una caja atraviese el sensor que encontramos justo en medio del elemento. Una vez conocido el peso de la caja se plantean dos situaciones. En primer lugar puede girar a la derecha y entrar en la zona a la denominada “área de picking con preforma”, o puede continuar recto para posteriormente elegir otra rama. Para saber que camino se le asignará es necesario conocer la función de esta rama derecha mencionada anteriormente.

En esta primera bifurcación encontramos una estación de “*Pick & Place*”, cuya particularidad reside en que la forma de apilar las cajas sobre el pallet viene predeterminada; es decir, es una preforma. En este subproceso se podrá elegir entre varias preformas, atendiendo a las necesidades del cliente, en el caso de este proyecto se exigieron dos. Para realizar esta tarea se asigna un orden de entrada de las cajas, y simplemente se van demandando, de manera que si se detecta una caja grande en el “*Scale Conveyor*”, y se requiere una caja grande para formar el pallet, esta será derivada mediante el “*Pop Up Wheel Sorter*”. Sin embargo, no siempre se obtendrá el tamaño de caja deseado, o puede que la entrada a la bifurcación esté saturada porque la estación de “*Pick & Place*” esté funcionando. Entonces, ¿qué ocurre en estos casos?

Si se da la problemática mencionada anteriormente la respuesta es simple: las cajas continúan rectas. En este caso las cajas continuarán avanzando hasta llegar al segundo “*Scale Conveyor*”, en el que de nuevo se volverá a discretizar el tamaño de la caja. Esta operación se repite debido a la complejidad de almacenar los tamaños de una manera sencilla y ordenada. En este caso se podrá de nuevo acceder a dos subramas, sin embargo la elección ahora será bastante más sencilla, y será más fácil de entender una vez explicado el funcionamiento de la rama de la izquierda, o “área del paletizador”. En esta rama las cajas llegarán a la estación de paletizado o “*Palletizer*”, que se encarga de almacenar cajas en techo del elemento. Esta tarea la realiza mediante una pieza en forma de pletina que empuja las cajas una vez estas han llegado al final de la cinta. Tras un breve análisis se determinó que la configuración más sencilla y eficiente era juntar 6 cajas pequeñas de dos en dos. Es por ello que a esta rama se derivará únicamente las cajas pequeñas.

Por ende queda que las cajas medianas y grandes sobrantes avanzarán por la cinta central hasta la segunda estación de “*Pick & Place*”. A esta rama central también se le denominará “área de picking aleatorio”, ya que se conformarán pallets sin importar el orden de llegada de las cajas.

Finalmente resta remarcar que pese a las bondades del software empleado *FACTORY IO* este programa no es perfecto y sufre de ciertas limitaciones que pueden ser un tanto molestas. La más notoria es la baja velocidad que alcanzan los brazos de picking sin tener que renunciar a la precisión. Este inconveniente no imposibilita la ejecución del proyecto, sino que tan solo lo vuelve mínimamente más tedioso, y es por ello que pese a acarrear este tipo de inconvenientes, se eligió como software más apropiado para la simulación.

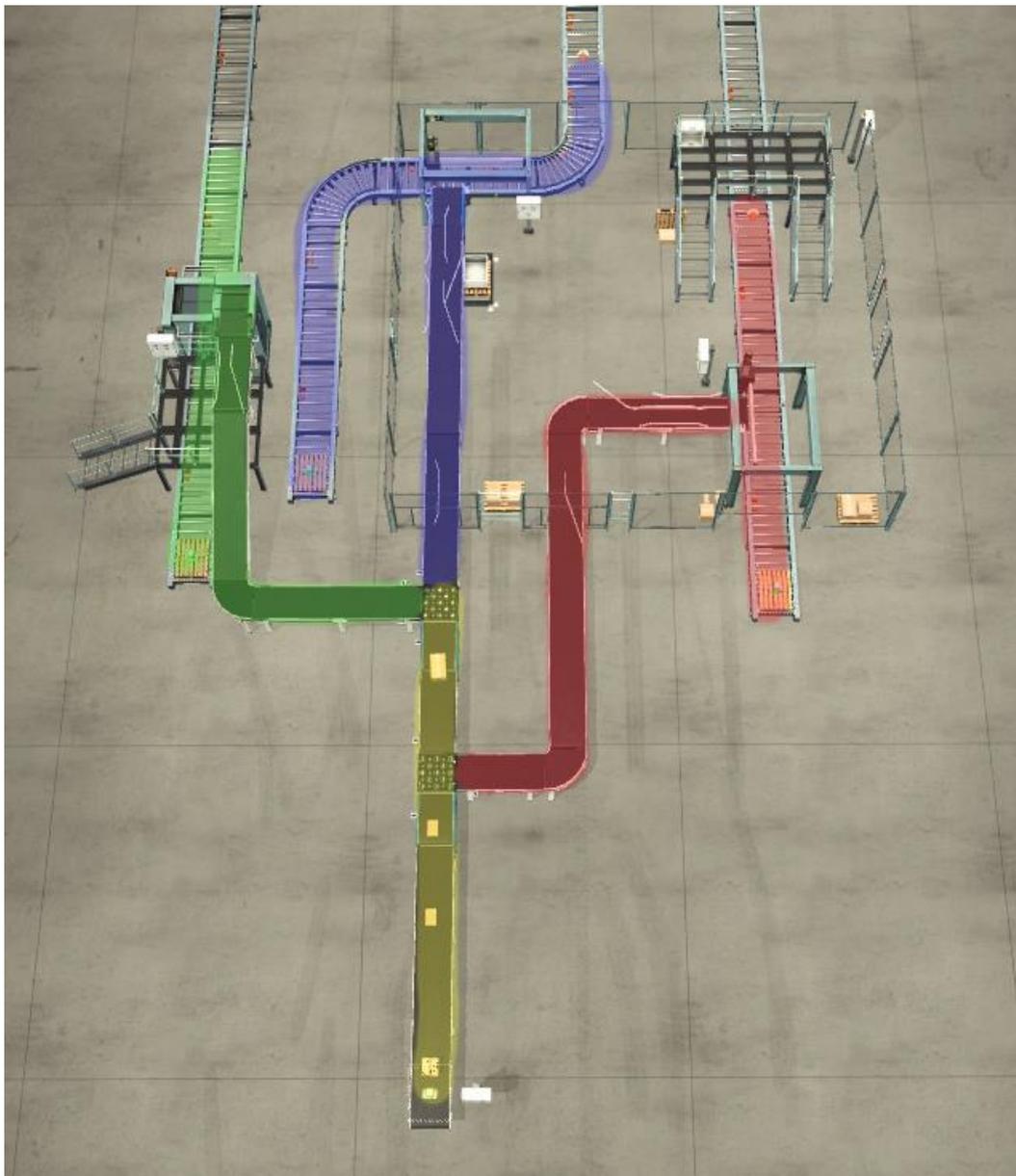


Ilustración 9; Imagen propia, esquema de las distintas zonas en el software de simulación *FACTORY I/O*

En esta imagen se muestra de forma esquemática las distintas zonas mencionadas en la explicación anterior y que integran el proceso:

- Zona 0 (amarillo) es la zona general y compartida por todos los subprocesos.
- Zona 1 (rojo) es la zona del picking con preforma.
- Zona 2 (verde) es la zona del paletizador.
- Zona 3 (azul) es la zona de picking aleatorio.

4. DESCRIPCIÓN DETALLADA DE LA SOLUCIÓN ADOPTADA

En este apartado se detallará la solución obtenida mediante las herramientas mencionadas anteriormente.

4.1 – CONEXIONES DE LOS ELEMENTOS

Para una mejor comprensión de las conexiones que existen entre los distintos elementos del programa se ha dibujado un pequeño esquema en el que se pueden observar de manera directa.

La pieza principal de las conexiones del proyecto es el PLC, el cual actúa de nexo recibiendo y suministrando información. Como ya se ha mencionado el PLC es el que trae por defecto la herramienta *CODESYS*, pero se recuerda que podría ser reemplazados por un gran abanico de PLCs reales siempre y cuando *CODESYS* distribuya los drivers correspondientes. A la izquierda se observa el software de programación donde se desarrolla el proyecto, cuyo funcionamiento va a ser descrito en los próximos apartados. La comunicación entre estos dos elementos tiene lugar por conexión Ethernet con protocolo TCP/IP. En la parte inferior se encuentra el software de simulación *FACTORY IO*, que ha sido explicado anteriormente, y que se conecta con esta pieza central mediante comunicación Modbus TCP. Finalmente se observa a la derecha la pantalla HMI, que se conecta al PLC mediante HTTP siempre y cuando se encuentren en la misma red.

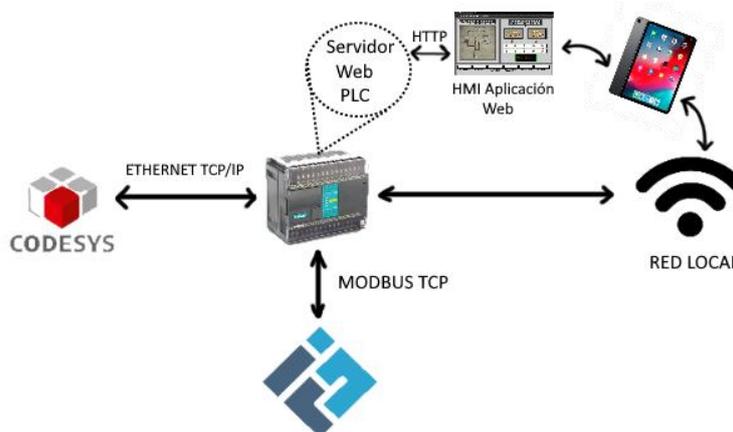


Ilustración 10; Imagen propia, esquema de los distintos elementos que toman parte y sus conexiones

Para un conocimiento detallado de como establecer todas las comunicaciones expuestas anteriormente, visitar el manual ofrecido por *FACTORY IO*, apartado *CODESYS*. [15]

4.2 - DISEÑO DEL PROCESO PRODUCTIVO

Para realizar el diseño del automatismo del proceso productivo se ha utilizado la metodología GRAFCET (Grphe Fonctionnel de Commande Etape Transition) que significa diagrama de control con etapas y transiciones. Trata de una representación gráfica de los sucesivos comportamientos de un sistema lógico predefinido por sus entradas y salidas. Permite realizar un modelo de proceso a automatizar compuesto por entradas, acciones a realizar y procesos intermedios que provocan dichas acciones. Además, se pueden establecer relaciones jerárquicas entre varios GRAFCETs mediante opciones técnicas como órdenes de forzado, encapsulado y macroetapa, lo cual resulta muy útil para incorporar sistemas de seguridad y modos de funcionamiento. Todos los GRAFCETs representados cumplen la norma UNE-EN 60848:2013. [16]

Antes de comenzar con la explicación uno por uno los distintos GRAFCETs que integran la solución al automatismo del proceso, y puesto a que va a ser necesario, se realizará una breve introducción a las variables que conforman el proceso:

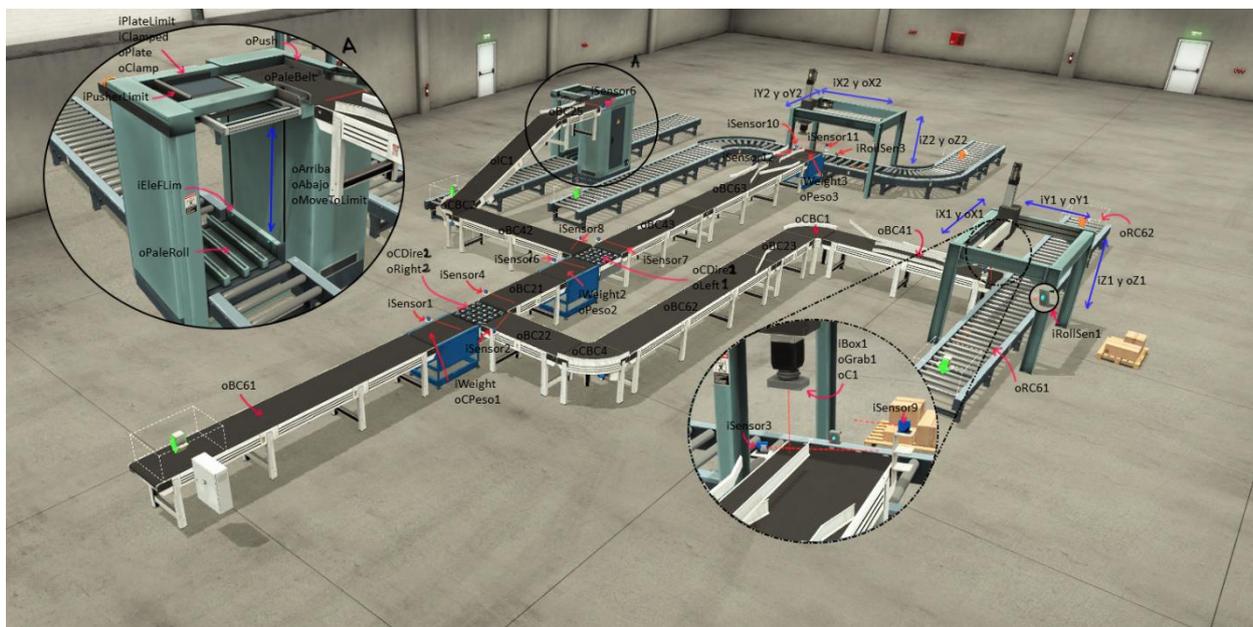


Ilustración 11; Imagen propia, distribución de los sensores principales vistos desde el *FACTORY IO*

Destacar que la imagen es meramente introductoria, y por tanto las variables se explicarán en detalle en los GRAFCETs que las empleen. También mencionar que no se han incluido todas las variables, solo se muestran las más determinantes.

Para encontrar la lista de todas las variables, su identificador y su breve descripción visitar el Anexo II: Variables.

Previamente a la explicación detallada de todos los GRAFCETs, añadir que todos ellos se muestran de manera ininterrumpida en el Anexo I: GRAFCETs.

4.2.1 - ZONA 0: ZONA GENERAL

La zona engloba tanto el funcionamiento de las cintas centrales como el del sistema de alarmas, de paradas y mantenimientos. Esta zona dispone de tres GRAFCETS.

4.2.1.1 – EMERGENCIAS

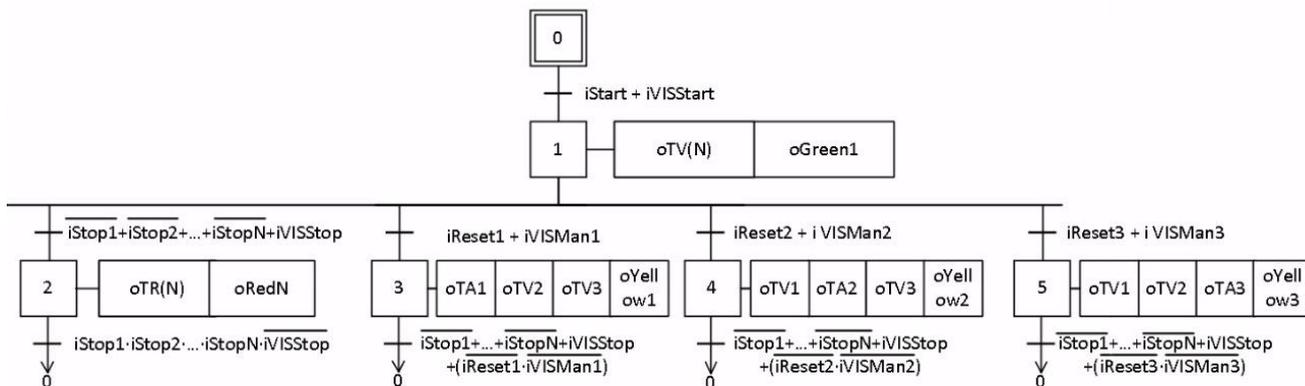


Ilustración 12; Imagen propia, GRAFCET EMERGENCIAS

El primer GRAFCET es el encargado de mantener la seguridad de toda la planta. Es quien gobierna sobre todos los dispositivos lumínicos y quien activa los distintos estados del proceso. Para inicializar el GRAFCET se deberá activar uno de las dos variables de comienzo, que son el botón de Start situado físicamente en la planta (iStart) o el botón ficticio que se encuentra en la pantalla de operador. Ambos dos tienen la misma función y para omitir colocar ambos en todos los GRAFCETS posteriores se resumirán como una única variable denominada START. Una vez se ha inicializado el proceso se avanzará a la etapa 1, que corresponde a la etapa de correcto funcionamiento del GRAFCET. En esta etapa se activarán los dispositivos luminosos oTV(N) que corresponde al estado de luz verde de los tres semáforos de luces dispuestos en cada una de las subestaciones siendo N los números (1,2,3), y oGreen1 que corresponde al led verde situado junto al botón de Start físico que indica que el proceso está funcionando.

No obstante pueden surgir altercados que requieran de la parada total del proceso y para ello se dispondrá de la etapa 2, que corresponde al estado de parada de la planta. Para acceder a este estado será necesario desactivar uno de los distintos botones (en concreto 4) de parada situados a lo largo de la planta (iStopN) o bien activar el botón de Stop virtual de la pantalla de operador. Este estado fuerza el cese de actividad en toda la planta debido a que todas las transiciones del proceso tienen como condición que este GRAFCET no se encuentre en esta etapa en particular, amén de las condiciones que se requieran en el estado de funcionamiento habitual. Como se ha mencionado esta condición se encontrará a lo largo de todas las transiciones de todos los GRAFCETS, pero para no sobrecargar los diagramas visualmente se omitirá. En esta etapa al igual que en la etapa de funcionamiento, se activarán la luz roja de los tres semáforos lumínicos así como el led rojo correspondiente a cada botón de parada. Para

abandonar el estado de parada y retomar el correcto funcionamiento de la planta será necesario que todos los botones de parada estén activos y el virtual desactivado.

Además del estado de funcionamiento habitual y de parada, también existen los estados de mantenimiento de las distintas subestaciones. Estos estados están separados del de parada debido a que en los estados de mantenimiento de una subestación en particular, las otras dos deben seguir funcionando con normalidad para no perder eficiencia. Se observan tres de estos estados (etapa 3, 4 y 5) que corresponden a cada una de las tres subestaciones. Todas tienen el mismo funcionamiento, activándose con el correspondiente botón físico de mantenimiento o con su equivalente virtual. Todas ellas activan la luz amarilla del semáforo lumínico de su estación y el verde de los otros dos, así como la luz amarilla que corresponde al botón. Para abandonar este estado no solo bastará con que tanto el botón de mantenimiento físico como virtual estén desactivados, sino que si se desactiva cualquier botón físico de parada o se activa el botón de parada virtual se forzará el cese del estado de mantenimiento para acceder al estado de parada. Las condiciones de parada también se encontrarán como condiciones a lo largo de todos los GRAFCETs del proceso, y así como las de parada se omitirán para facilitar la comprensión.

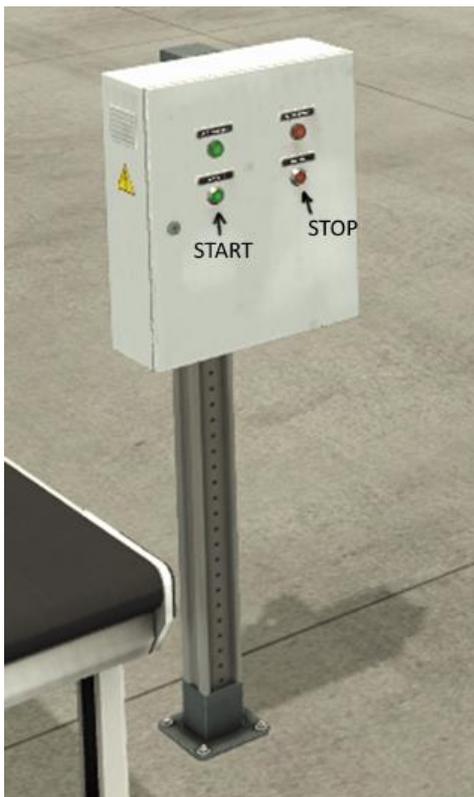


Ilustración 13; Imagen propia, cuadro de mando general



Ilustración 14; Imagen propia, cuadro de mando A

4.2.1.2 - CINTA-CENTRAL

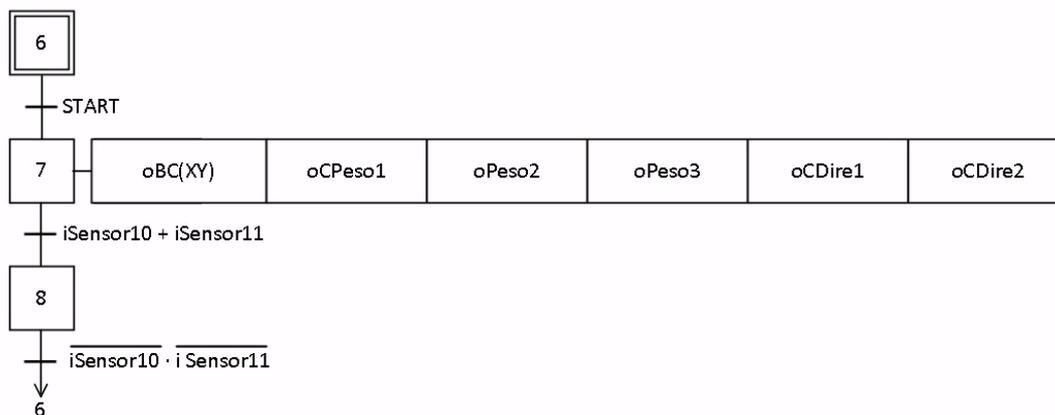


Ilustración 15; Imagen propia, GRAFCET CINTA_CENTRAL

El funcionamiento de este diagrama es relativamente sencillo. El botón de **Start** se encarga de poner en marcha todas las cintas citadas en la imagen anterior. Seguidamente, si una caja llega al Sensor 10 o 11 (los sensores situados al final de la cinta de la zona de “Pick & Place” aleatorio), detendrá el movimiento de todas las cintas centrales para evitar colapsos. Una vez se ha manipulado la caja y se desactivan ambos sensores las cintas se vuelven a poner en funcionamiento hasta que una nueva pieza las vuelva a detener.

Para una mayor claridad se explicará brevemente que efecto causa cada acción. Se dispone de variables de tres tipos: las oBC(XY) que hacen referencia a las cintas transportadoras siendo la primera letra el número de la distancia en metros que recorren y el segundo el orden de esta dentro de su grupo, las oPeso que hacen referencia al movimiento de la cinta de todos los “Scale Conveyor”, y finalmente encontramos los oCDire que hacen referencia al movimiento en línea recta de los dos “Pop Up Wheel Sorter”.

4.2.1.3 – DOOR

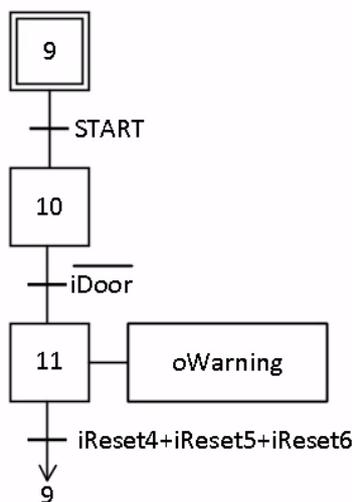


Ilustración 16; Imagen propia, GRAFCET DOOR

El GRAFCET Door es muy sencillo y se encarga de controlar si alguien entra en el perímetro vallado de la planta. Este perímetro dispone de una puerta con un sensor que se encuentra activo cuando la puerta está cerrada, de este modo si alguien abre la puerta para ingresar en esta zona se activará la alarma (oWarning). Para poder detener la alarma se deberá presionar uno de los tres botones correspondientes, dos de ellos situados en el interior del perímetro y uno en el exterior, para poder desactivar la alarma tanto al entrar como al salir. Si la alarma no cesa significa que el operario no ha sido capaz de presionar el botón y puede ser indicativo de que ha sufrido algún tipo de problema. Es un GRAFCET cuya finalidad es garantizar la seguridad de los operarios.

Estos son los tres GRAFCETs que componen la Zona 0.

4.2.2 – ZONA A: PICKING PREFORMA

En este apartado se introducirán los distintos GRAFCETs que componen la zona del picking con preforma. Esta es la zona que más GRAFCETs emplea con un total de ocho GRAFCETs. Aunque no todos se detallarán debido a que varios comparten estructura y funcionamiento.

4.2.2.1 – CINTA-BELT-1

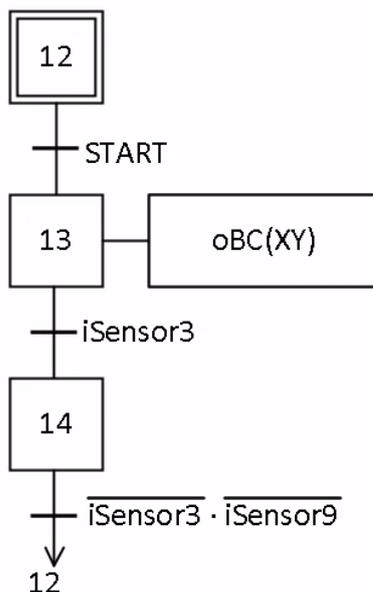


Ilustración 17; Imagen propia, GRAFCET CINTA_BELT_1

Este primer GRAFCET es el encargado de controlar el funcionamiento de las cintas de la zona A, y su funcionamiento es muy sencillo. El diagrama se inicializa con la variable START, que pone en marcha todas las cintas de este subproceso. Estas estarán activas hasta que una caja llegue hasta el sensor dispuesto en el final de las cintas, el sensor (iSensor3), y se mantendrán desactivadas hasta que los dos sensores iSensor3 e iSensor9 (sensor dispuesto encima del sensor iSensor3 para aumentar el tiempo que las cintas están paradas, ya que eventualmente se producían colapsos debido a la lentitud con la que se subían las cajas) se desactiven. Cabe destacar la razón por la que estas cintas están separadas de las cintas que se gobiernan en la sección global, y es debido a que es realmente conveniente dotar de independencia a las distintas regiones del proceso, para que todas puedan trabajar de forma independiente sin depender del funcionamiento del resto.

Este GRAFCET se verá repetido en todas las zonas ya que el control de las cintas de los distintos subprocesos es bastante similar.

4.2.2.2 – CINTA-ROLLER-1

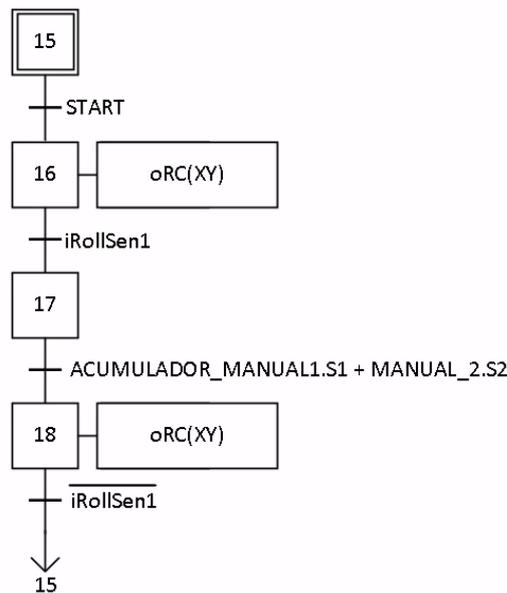


Ilustración 18; Imagen propia, GRAFCET CINTA_ROLLER_1

Este diagrama se encarga esencialmente del funcionamiento de las cintas de rodillos, que son las que suministran y transportan los pallets. Estas se ponen a funcionar en cuanto se activa la variable START, y se detienen cuando los pallets llegan al sensor (iRollSen1) situando frente a la estación de Picking. El movimiento de las cintas se reanuda en cuanto todas las cajas de la preforma estén sobre el pallet, por tanto se colocará como condición para reanudar la marcha las etapas correspondientes al estado de finalización de la preforma (ACUMULADOR_MANUAL1 para la primera preforma y MANUAL_2 para la segunda preforma). Las cintas por tanto reanudarán su marcha hasta que el pallet abandone el sensor y vuelta a empezar.

Finalmente añadir que la condición de finalización de preforma será explicada en más detalle en el correspondiente GRAFCET, y que al igual que el diagrama anterior, este se verá repetido en las distintas zonas del proyecto controlando las respectivas cintas de rodillos de cada zona.

4.2.2.3 – TAM-SEL-1

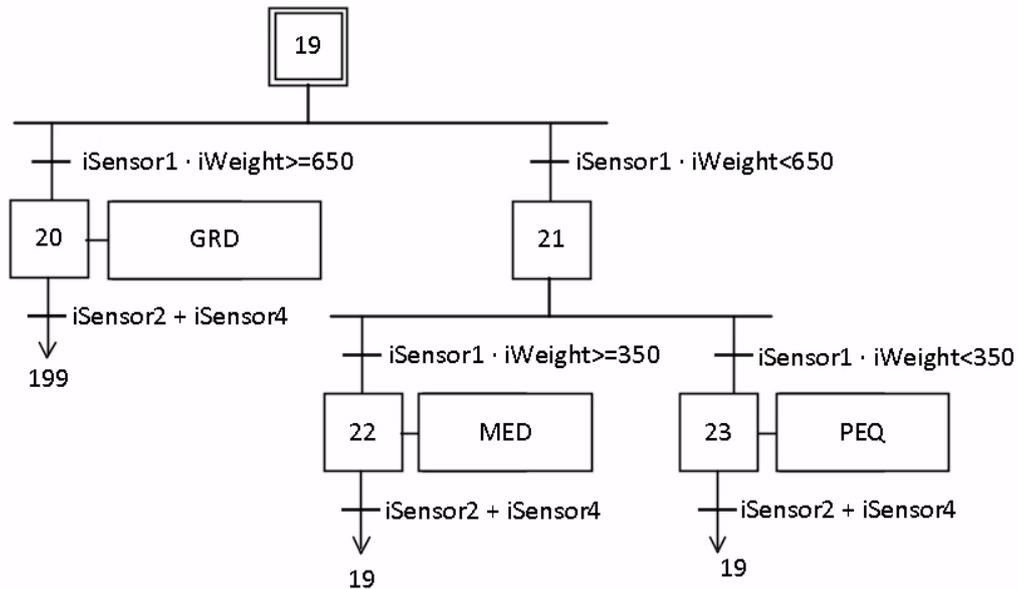


Ilustración 19; Imagen propia, GRAFCET TAM_SEL_1

Este diagrama se encarga de identificar el peso de cada caja cuando atraviesan el primer “Scale Conveyor”. El funcionamiento comienza cuando una caja atraviesa el sensor (iSensor1) colocado junto a la primera báscula. En este punto si la caja pesa más de 6’5kg, es decir, si la variable iWeight1 es igual o superior a 650 activará la etapa 1 manteniendo la variable GRD (grande) activada hasta que la caja atraviese el sensor iSensor2 (situado en la entrada de la zona A o el sensor iSensor4 (situado en la salida del “Pop Up Wheel Sorter”). Esto es así debido a que esta señal será empleada para indicar que la caja es grande y poder por tanto elegir su destino. Una vez la caja esté en el destino seleccionado y haya atravesado el correspondiente sensor el diagrama retorna a su posición inicial a la espera de una nueva caja que clasificar. El caso de MED y PEQ es exactamente el mismo, con la particularidad que se activan con un pesaje distinto, siendo ($6'5\text{kg} > \text{MED} > 3'5\text{kg}$) en el caso de la mediana y ($> 3'5\text{kg}$) en el caso de la pequeña. Los valores empleados para la medición son del orden de centenas debido a que en la conexión de ambos softwares los valores tales como los pesos o los voltajes utilizados en el “Pick & Place” se ven multiplicados por “100” para una mayor precisión. Esta escala se puede configurar dentro del propio Factory IO.

El hecho de poder conocer el peso de la caja que llega a cada estación es necesario en las tres zonas, por ello se dispondrá de tres GRAFCETs paricales idénticos a este cada uno para la zona solicitada, y con la única diferencia siendo las variables “iWeight(1,2,3)”, ya que estas son suministradas por cada una de las básculas o “Scale Conveyors” situados a lo largo del proceso.

4.2.2.4 – MANUAL1 Y MANUAL2

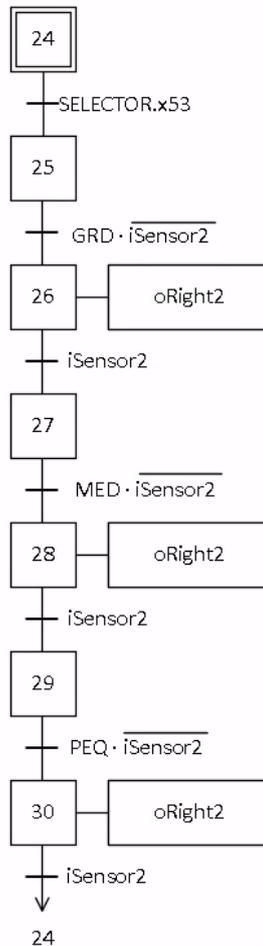


Ilustración 20; Imagen propia, GRAFCET MANUAL1

En este apartado veremos los GRAFCETs que permiten seleccionar el destino de las cajas, concretamente los que permiten derivar las cajas a la zona 1 para el posterior apilamiento sobre el pallet. Este apartado puede contener tantos GRAFCETs parciales como preformas se desee añadir, aunque en nuestro caso solo se han diseñado dos. No obstante la estructura es idéntica y solo cambia el tamaño de caja que se desee y el orden en que se precise, por tanto solo se explicará la estructura de uno de ellos, en este caso MANUAL1. Añadir que el tipo de preforma se seleccionará con un GRAFCET llamado SELECTOR que se explicará posteriormente.

Como se ha mencionado anteriormente este diagrama permite derivar las cajas del tamaño que se desee. Esto es posible debido a que el elemento que permite el giro de noventa grados de las cajas (*“Pop Up Wheel Sorter”*) está situado inmediatamente tras la primera báscula, y por tanto se conoce el tipo de caja que se va a manipular. Es por ello que, una vez se haya seleccionado el tipo de preforma, solo se tendrá que esperar a que la caja deseada llegue al *“Pop Up”* para derivarla. Siguiendo el ejemplo mostrado en la imagen, en el momento en que se tenga una caja

grande, y el sensor iSensor2 situado a la entrada de la zona A no esté activo (esto es así para evitar colapsos) el elemento encargado de derivar las cajas activará la variable oRight2 resultando en un giro de noventa grados. Una vez las cajas hayan entrado en el subproceso deseado activando por tanto el sensor iSensor2 se procederá a esperar el siguiente tamaño de caja deseado. Una vez se haya completado la preforma y siempre que no se haya cambiado el tipo de preforma el proceso volverá a empezar.

4.2.2.5 – ACUMULADOR-MANUAL1 Y MANUAL-2

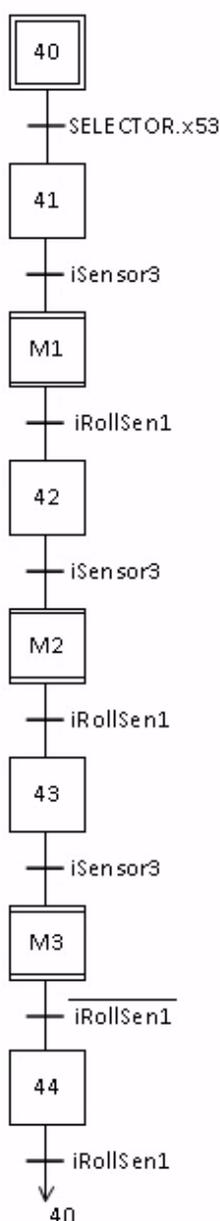


Ilustración 21; Imagen propia, GRAFCET ACUMULADOR_MANUAL1

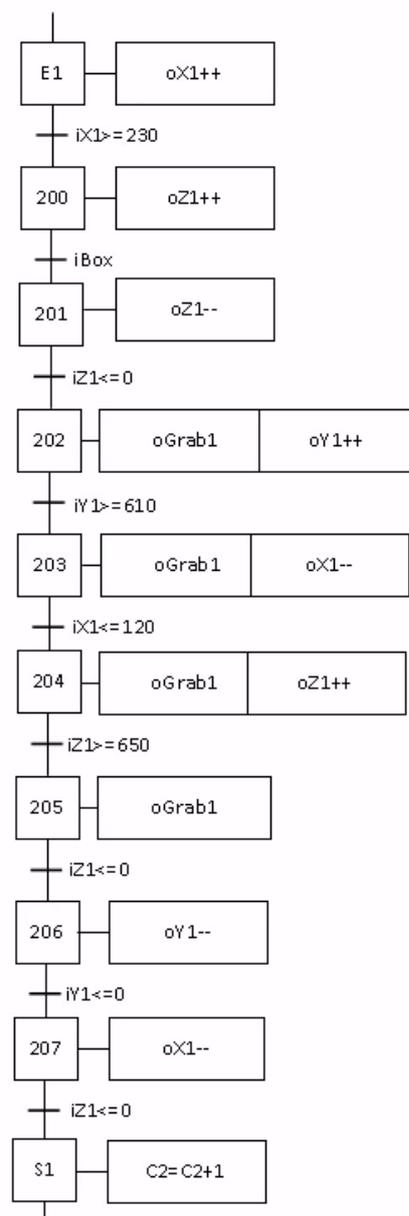


Ilustración 22; Imagen propia, MACROETAPA M1

Este GRAFCET se encarga de controlar el funcionamiento de la estación de picking, y al igual que en el apartado anterior, existen tantos GRAFCETs parciales de este tipo como preformas dispongamos, y son idénticos entre sí, únicamente variando las coordenadas de depósito de las cajas. Es por ello que únicamente se explicará un ejemplo, en este caso el de la preforma primera. Para empezar comentar que este GRAFCET dispone de tantas macros como cajas contenga la preforma, y cada una de las macros incluye el diagrama necesario para la correcta colocación de cada caja. Así mismo, las macros tienen la misma estructura entre sí, y por tanto solo se explicará el ejemplo de la primera, que corresponde a la colocación de una caja grande.

Al igual que en el apartado anterior, el GRAFCET da comienzo en cuanto se active la correspondiente preforma, y continuará en cuanto el sensor iSensor3 dispuesto al final de las cintas de la zona A detecte una caja, puesto que esto significará que esta está en posición de ser manipulada. Una vez esto ocurra se avanzará a la etapa dos, donde se encuentra la macro, y por tanto esta dará comienzo. Como se ha mencionado anteriormente, las macros se encargan del movimiento del brazo de la estación de Picking. Estos movimientos se efectúan aumentando o disminuyendo las coordenadas de los tres ejes cartesianos (oXYZ++ y oXYZ--), en función del eje en el que se desee efectuar el movimiento, y los cambios quedarán registrados en las variables (iXYZ) que indican la posición actual del brazo. Una vez comprendido el funcionamiento del brazo la comprensión del proceso es muy sencilla. Inicialmente se aumenta la posición del brazo en el eje X para situarlo sobre la caja hasta que este alcance la posición deseada. A continuación se aumentará la coordenada Z hasta que el sensor situado en la base de aspiración del brazo (iBox) detecte la presencia de una caja, y se activará la succión (oGrab). Seguidamente se reducirá Z para levantar la caja y se llevará en X e Y a la posición deseada. Una vez alcanzada esta se aumentará Z de nuevo para que el brazo descienda y que la caja no tenga que soportar una gran caída, y una vez este lo suficientemente cerca del pallet se desactivará la succión colocando la caja sobre el pallet. Finalmente el brazo retornará a la posición inicial y se abandonará la macro.

Este mismo proceso se realiza tantas veces como cajas se desee apilar sobre el pallet, con la particularidad que será la última etapa del diagrama de la última macro la señal que permitirá la reanudación del movimiento de las cintas de rodillos. Esto es así debido a que al ser la última etapa de la última macro, el pallet ya está perfectamente conformado y listo para ser distribuido. Cabe añadir que en esta última etapa está colocado el contador que indica la cantidad de pallets generados en cada preforma.

Nota: en el GRAFCET de ejemplo de ha añadido en la M1 para mostrar la etapa en la que se encuentra (la última de la macro) sin embargo el contador se encuentra en la última macro, es decir, en la "S3".

Finalmente mencionar que estos incrementos en las coordenadas expresados como acciones han sido programados de forma distinta en el programa *CODESYS* debido a que este permite distintos lenguajes de programación y la resolución se simplificaba al emplear una estructura de texto (ST).

4.2.2.6 – SELECTOR

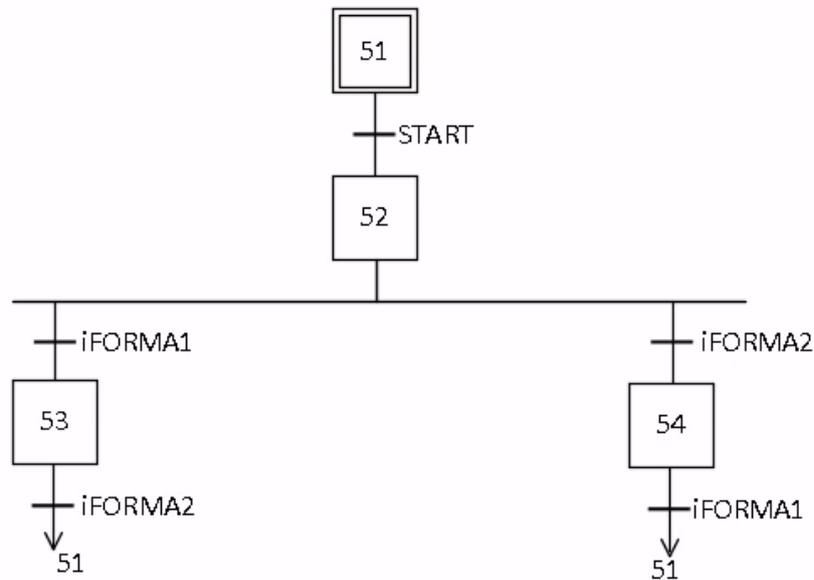


Ilustración 23; Imagen propia, GRAFCET SELECTOR

Este GRAFCET es el que ha sido mencionado en apartados anteriores que permite seleccionar el tipo de preforma a realizar en la estación de picking.

El diagrama da comienzo en cuanto se activa la variable de START y espera hasta que se seleccione una de las dos preformas disponibles. Para seleccionarlas se deberá pulsar el correspondiente botón en el manual de usuario (iFORMA1 o iFORMA2). Una vez estemos en la etapa que corresponde a cada preforma la única forma de abandonarla será pulsando el botón correspondiente a la otra preforma forzando así el cambio de modo con tan solo apretar un botón.

Con este GRAFCET concluye la explicación de todos los GRAFCETS parciales que componen la zona A. Como recordatorio añadir que este no es un número finito, ya que se pueden incluir tantas preformas (con sus diagramas correspondientes) como se requiera sin alterar el funcionamiento del programa.

4.2.3 – ZONA B: ZONA PALETIZADOR

A continuación se expondrán todos los GRAFCETs correspondientes a la zona de paletización. Esta zona dispone de un total de seis GRAFCETs, de los cuales algunos se explicarán de forma muy resumida puesto que son muy similares a otros introducidos anteriormente.

4.2.3.1 – CINTA-BELT-PALETIZADOR

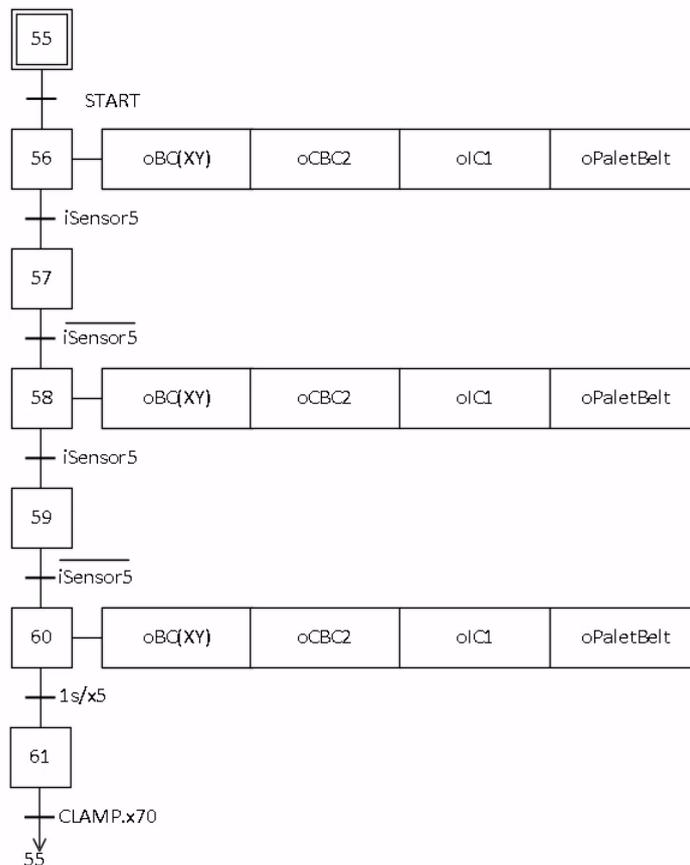


Ilustración 24; Imagen propia, GRAFCET CINTA_BELT_PALETIZADOR

Este primer GRAFCET es el encargado de hacer funcionar las distintas cintas de la zona B, pero como se puede apreciar, es algo más complejo que sus análogos de otras zonas. Esto se debe a que por temas de espacio, caben justo dos cajas pequeñas en serie, y la forma más sencilla de apilarlas es dejando pasar ambas, que se coloquen en fila de forma automática por el simple movimiento de las cintas, y deteniendo las cintas para poder efectuar las operaciones de la parte superior del paletizador. Una vez conocida la explicación veamos como se recrea en el diagrama.

El proceso comienza con la variable START, hecho que pone en marcha todas las cintas del proceso (en este caso, amén de las Belt Conveyor “oBC”, se dispone de cintas curvas Curved Belt Conveyor “oCBC”, cintas inclinadas Inclined Conveyor “oIC” y una cinta integrada en el propio paletizador “oPaleBelt”). Como se ha mencionado previamente se necesitan dos cajas antes de realizar las operaciones necesarias, o lo que es lo mismo, que se active y desactive el sensor “iSensor5” (situado justo al final de las cintas) dos veces. Una vez realizado este proceso se mantendrá el funcionamiento de las cintas durante un segundo más para asegurar que su movimiento enfila ambas cajas. Finalmente se avanzará a un estado de reposo en el que se

esperará hasta que el GRAFCET encargado de la zona alta del paletizador (CLAMP) termine de realizar las operaciones necesarias y permita que el proceso continúe.

4.2.3.2 – CINTA-ROLLER-ELEVATOR

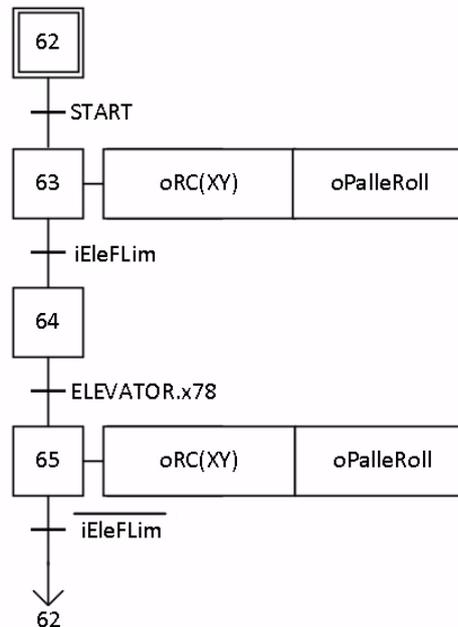


Ilustración 25; Imagen propia, GRAFCET CINTA_ROLLER_ELEVATOR

Este GRAFCET es el encargado del funcionamiento de las cintas de rodillos de esta zona. Su funcionamiento es muy similar a sus análogos en otras zonas, con un par de pequeños detalles.

En primer lugar el paletizador dispone de su propia cinta de rodillos en la base, y por tanto será necesario añadir su funcionamiento (oPalleRoll). Así mismo también dispone de un sensor (iEleFLim) que detecta cuando el pallet está en la posición adecuada para ser manipulado, y por tanto no se requerirá de otro sensor externo. Por último aclarar que la condición para reanudar las cintas viene dada por una etapa del GRAFCET ELEVATOR que se introducirá posteriormente.

4.2.3.3 – CLAMP

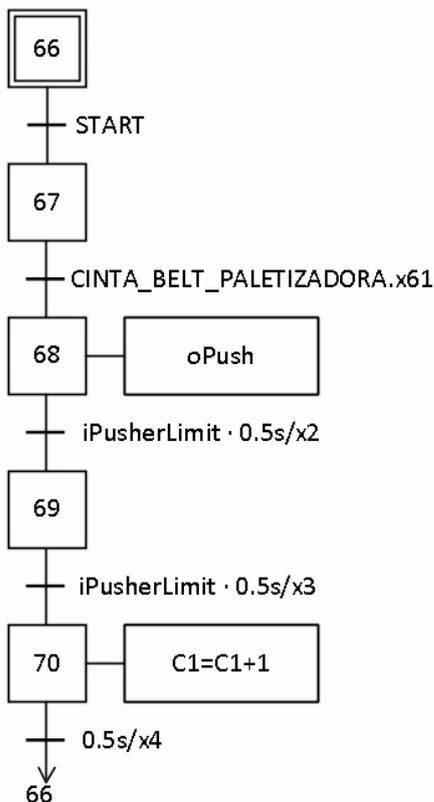


Ilustración 26; Imagen propia, GRAFCET CLAMP

Este diagrama se encarga de efectuar las operaciones de la zona superior del paletizador, es decir, se encarga esencialmente de empujar las cajas sobre la plataforma retráctil. El funcionamiento es muy sencillo; el proceso comienza cuando se activa la variable START y pasa a una etapa de reposo que espera a que el GRAFCET que controla las cintas deje pasar dos cajas y llegue a la etapa sesenta y uno. Una vez las dos cajas estén en posición se activa una pletina que se encarga de empujar las cajas sobre la plataforma (oPush). Cabe añadir que solo existe un sensor para expresar que la pletina ha llegado a su límite (iPusherLim), y este se activa en las dos posiciones extremas. Es por ello que si este se pusiera como condición para dejar de realizar la acción de empujar, no se iniciaría la acción, ya que en la posición de reposo, completamente replegado, el sensor estaría activo; por tanto se necesitará añadir una condición temporal que fuerce a la acción a comenzar desactivando así el sensor y reactivándolo en el otro extremo. Esto es válido tanto para el avance como para el retroceso. Finalmente se esperará 0.5s a que la pletina se repliegue completamente antes de posibilitar el reinicio del proceso. En esta última etapa se ha dispuesto un contador que indica la cantidad de veces que se realiza la acción de empujar, y que será de gran ayuda en el GRAFCET ELEVATOR expuesto a continuación.

4.2.3.4 – ELEVATOR

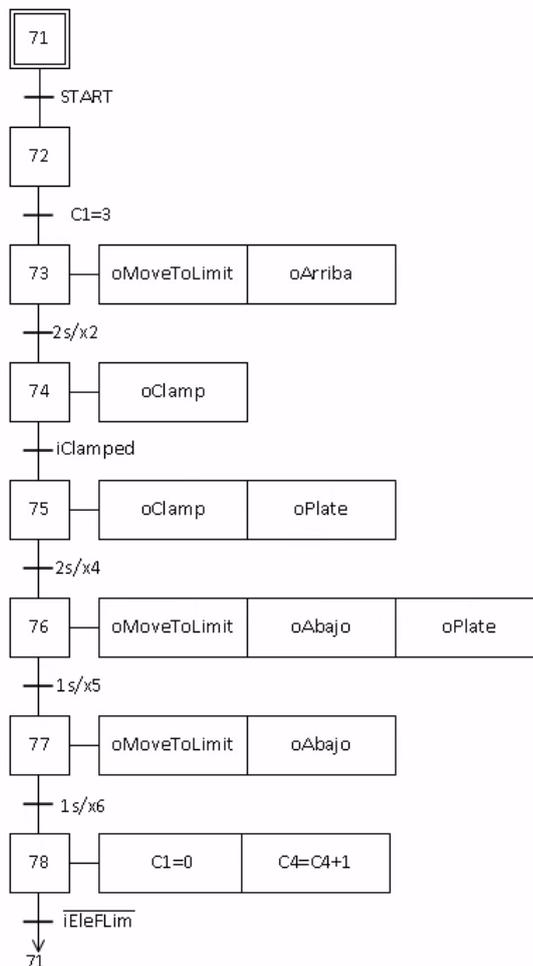


Ilustración 27; Imagen propia, GRAFCET ELEVATOR

El funcionamiento de este GRAFCET da comienzo cuando se activa la variable START, pasando a una etapa de reposo. Para comprender porque se requiere de esta etapa de reposo se necesitará comprender la forma en la que se conforman los pallets en esta zona. Como es conocido, a esta zona solo llegan cajas pequeñas, que posteriormente son empujadas de dos en dos sobre la plataforma retráctil de la parte superior del paletizador. El máximo de filas de dos cajas de se pueden colocar es de tres, resultando en un total de seis cajas pequeñas, o lo que es lo mismo, se tendrá que realizar tres iteraciones del GRAFCET encargado de las cintas. Esta cuenta de tres la lleva el contador mencionado en el anterior apartado, que reanudará el funcionamiento del GRAFCET en cuanto la complete indicando que las cajas están listas para el depósito. Es por ello que se deberá enviar el pallet hacia la parte superior mediante el elevador integrado en el propio paletizador, mediante las acciones “oArriba” que indican el sentido del desplazamiento y “oMoveToLimit” que indica que se desea que el pallet viaje hasta la parte más alta del paletizador. Como no se dispone de ningún sensor que indique que el pallet ha llegado a la



posición deseada la condición establecida será temporal, de 2s exactamente, puesto que es el tiempo aproximado que tarda el pallet en subir. Una vez el pallet esté en el lugar deseado se activará la señal “oClamp” que fuerza a una pinzas a apretar las cajas entre sí para evitar una colocación inadecuada de estas. Una vez las pinzas lleguen a su límite (iClamped) y sin que éstas dejen de presionar las cajas levemente, se abrirá la plataforma retráctil permitiendo que todas las cajas se posen sobre el pallet. Al igual que con el elevador, la plataforma no dispone de ningún sensor que nos indique que se ha completado la acción de manera que colocaremos como condición que transcurran 2s, que es el tiempo aproximado que tarda en abrirse la plataforma. Una vez la plataforma esté completamente abierta y las cajas se encuentren sobre el pallet bastará con bajar el pallet mediante las acciones “oAbajo” (dirección) y “oMoveToLimit”. Cabe destacar que durante el primer segundo de bajada se mantendrá la plataforma abierta para evitar ningún pinzamiento con las cajas.

Finalmente se alcanzará la etapa setenta y ocho que es la que permite reanudar la marcha de las cintas de rodillos, forzando así la desactivación del sensor “iEleFLim” y permitiendo que se reanude el proceso. Además esta etapa es la que reinicia la cuenta de tres que indica que las cajas están llenas, y también es la encargada de llevar la cuenta de cuantos pallets se han generado en esta estación.

4.2.3.5 – TAM-SEL-2

Como es bien sabido, a esta zona únicamente deben llegar cajas pequeñas, ya que el paletizador está configurado para tratar únicamente con estas. Es debido a esto que se requerirá de un GRAFCET que permita distinguir los distintos pesos de las cajas para poder así derivar las que lo requieran. Este GRAFCET es idéntico al mostrado en la zona A y también lo es su explicación y funcionamiento, por tanto se omitirá la explicación en este apartado.

4.2.3.6 – PALETIZADOR-DIREC

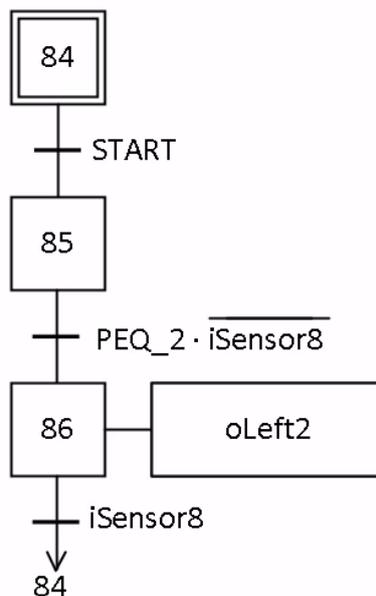


Ilustración 28; Imagen propia, GRAFCET PALETIZADOR_DIREC

Solo resta por tanto explicar el GRAFCET mediante el cual se derivan las cajas pequeñas a esta segunda zona. El funcionamiento de este GRAFCET es realmente sencillo, ya que únicamente cambia la dirección del “Pop Up Wheel Sorter” (oLeft2) cuando detecta una caja pequeña y siempre y cuando no haya una caja justo en la entrada de las cintas de la zona a la que se quiera acceder. El proceso se reinicia cuando la caja atraviesa el sensor colocado en la entrada de la zona puesto que indica la correcta incorporación de esta.

Con este GRAFCET se da por concluida la zona B o zona del paletizador.

4.2.4 – ZONA C: ZONA PICKING ALEATORIO

A continuación se expondrá todos los GRAFCETs correspondientes a la zona de picking automático. Esta zona es la más breve de explicar debido a que solo contiene tres GRAFCETs de los cuales dos se explicarán de forma muy resumida puesto que son muy similares a otros introducidos anteriormente.

4.2.4.1 – ROLLER-AUTOMATICO

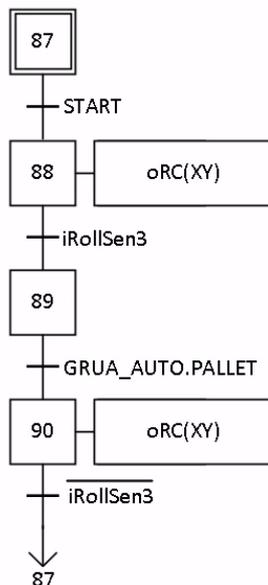


Ilustración 29; Imagen propia, GRAFCET ROLLER_AUTOMATICO

Sigue el mismo formato que los GRAFCETs análogos de las otras zonas: las cintas se detienen cuando se activa el sensor (iRollSen3) y se reactivan cuando el pallet ha sido completado, permitiendo avanzar a este y reanudar el proceso.

4.2.4.2 – TAM-SEL-3

La estructura y la explicación del GRAFCET es idéntica a la realizada en la primera zona por lo que se omitirá la imagen y la explicación.

No obstante sí que presenta una diferencia, no en cuanto a la estructura sino en cuanto a la finalidad. Mientras que las otras dos básculas indicaban el peso de la caja para poder derivarla, en esta zona llegan las cajas “sobrantes”, es decir, llegan las cajas que no han sido derivadas previamente puesto que no cumplían los requisitos. Es por ello que no se requiere de un elemento de derivación, pero sí de uno de pesaje que indique el tipo de caja que es para saber que proceso de apilamiento debe seguir, como se verá a continuación.

4.2.4.3 – GRUA-AUTO

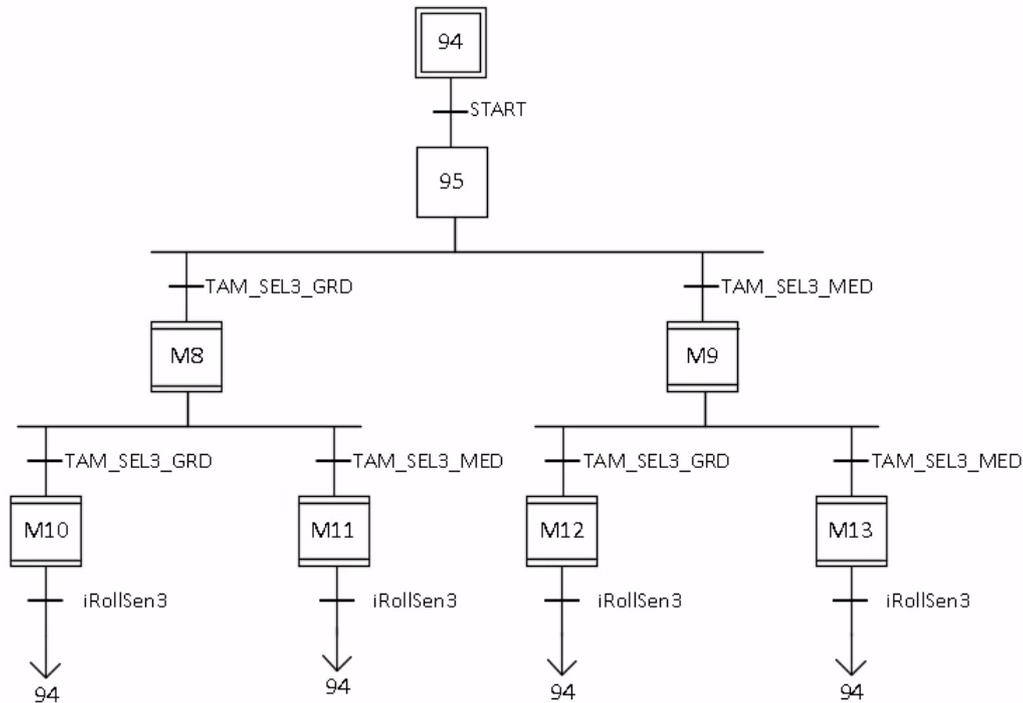


Ilustración 30; Imagen propia, GRAFCET GRUA_AUTO

Se podría definir este GRAFCET como la fusión de los dos GRAFCETS de la primera zona que se encargaban de derivar las cajas del tamaño adecuado y de mover el brazo mediante coordenadas para conformar los pallets. Este GRAFCET por tanto tiene la función tanto de escoger la mejor configuración para el pallet en base al tipo de caja que le llega como de ubicar las cajas sobre el pallet mediante el brazo de picking. Antes de pasar a la explicación del diagrama y para facilitar su comprensión cabe mencionar que los pallets de este proceso admiten cuatro configuraciones posibles en función del orden de llegada de las cajas, siendo estas G-G, G-M, M-G, M-M. Podemos proceder a la explicación del funcionamiento del proceso.

Una vez activada la variable de START se avanza a una etapa donde se esperará la llegada de una caja. Como es bien sabido a esta zona únicamente pueden llegar cajas grandes y medianas ya que todas las pequeñas han sido derivadas previamente a la zona del paletizador, por tanto únicamente se deberá plantear dos ramas posibles, siendo la primera el caso que llegue primero una caja grande y la segunda que llegue primero una caja mediana. En ambos casos se iniciarán las macros encargadas del funcionamiento del brazo de la segunda estación de picking y de su correcta disposición. Estas macros son idénticas a las mostradas en la primera zona únicamente variando las coordenadas de cada una de ellas puesto que la disposición va a ser distinta. Una vez esta primera caja haya sido satisfactoriamente colocada se plantean dos posibles escenarios (de ahí que de nuevo se observen dos nuevas ramas partiendo de las dos ya existentes): que la

segunda caja en llegar sea grande o mediana. En función del tamaño de esta, se ingresará a una macro u otra, y se ejecutarán los comandos necesarios para colocar las cajas en el lugar adecuado. Cabe destacar que en esta segunda fase de macros se activará una variable denominada "PALLET" en la última etapa que permitirá reanudar la marcha de las cintas de rodillos forzando al pallet a avanzar. Así mismo, en esta última etapa también se colocará un contador que indique el número de pallets realizados en esta estación.

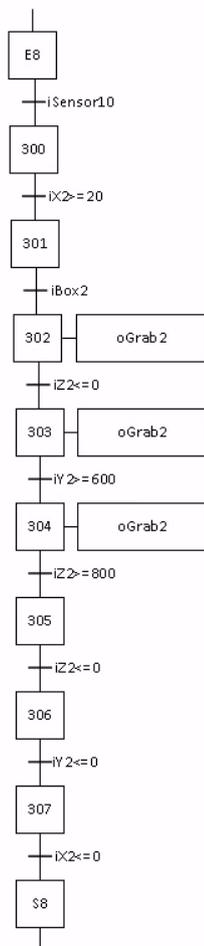


Ilustración 31; Imagen propia macroetapa M8

Una vez este proceso haya tenido lugar y el pallet haya abandonado la zona de picking, se esperará hasta la llegada de un nuevo pallet y la activación del su correspondiente sensor (iRollSen3) antes de permitir una nueva iteración del proceso, para evitar así que se empiece el conformado de un nuevo pallet si este no está en posición.

Con esto concluye la explicación de la zona C y por tanto de todos los GRAFCETs existentes en el proceso.

4.3 – IMPLEMENTACIÓN DEL SISTEMA EN CODESYS

Una vez diseñado todo el sistema mediante la metodología de GRAFCET, el siguiente paso será implementarlo. Para ello, en este apartado se explicará la metodología empleada y los pasos a seguir para realizarlo.

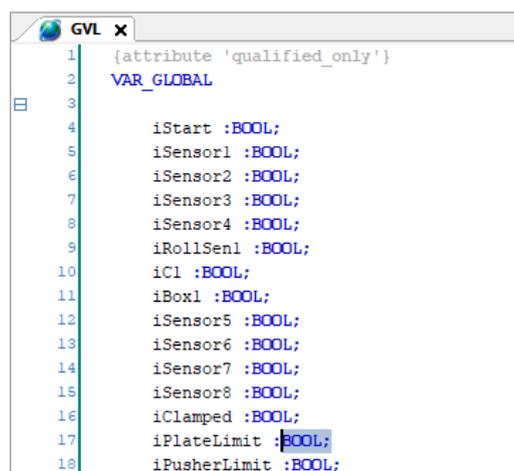
4.3.1 – DECLARACIÓN DE VARIABLES

El primer paso para la implementación de los distintos GRAFCETs en *CODESYS* será la declaración de todas y cada una de las variables que tomen parte en el proceso.

Estas variables serán las entradas y salidas del proceso, así como variables internas utilizadas en la programación. En el anexo II se encuentra la definición de cada variable utilizada. Estas requieren un calificativo global que permita así poder referenciarlas en todas las secciones de programación (POUs). Para ello simplemente se añadirá en el programa una lista de variables globales. En este apartado y previo a la escritura de todas las variables se deberá escoger un prefijo que defina su status de variable global. En el caso en particular de este proceso se ha empleado el prefijo por defecto: GVL. Para la llamada de estas variables bastará con colocar el prefijo que requieran seguido de punto (".") y del nombre de la variable, por ejemplo GVL.iStart.

Nota: mencionar que para facilitar la detección de si una variable es una entrada (sensor) o una salida (actuador), están han sido precedidas de la letra "i" en caso de las entradas, y de la letra "o" en caso de las salidas. Esto no es obligatorio pero facilita la comprensión de todos los procesos.

Finalmente es necesario indicar el tipo de variable (INT, BOOL, WORD) debido a que en función del tipo de dato se nos permitirá hacer distintas acciones.



```
GVL x
1 {attribute 'qualified_only'}
2 VAR_GLOBAL
3
4     iStart :BOOL;
5     iSensor1 :BOOL;
6     iSensor2 :BOOL;
7     iSensor3 :BOOL;
8     iSensor4 :BOOL;
9     iRollSen1 :BOOL;
10    iCl :BOOL;
11    iBox1 :BOOL;
12    iSensor5 :BOOL;
13    iSensor6 :BOOL;
14    iSensor7 :BOOL;
15    iSensor8 :BOOL;
16    iClamped :BOOL;
17    iPlateLimit :BOOL;
18    iPusherLimit :BOOL;
```

Ilustración 32; Imagen propia, pestaña con las variables globales en *CODESYS*

4.3.2 – LENGUAJES DE PROGRAMACIÓN

Para la implementación en CODESYS de los distintos GRAFCETs se han empleado unos elementos denominados POU (*Program Organization Unit*). Estas unidades básicas conformarán toda la estructura del proyecto y permitirán recrear todos los GRAFCETs mostrados previamente mediante el uso de numerosos lenguajes de programación, todos conforme con el estándar industrial internacional IEC 61131-3. [17]

Para el caso de este proyecto se han empleado principalmente tres, el lenguaje SFC en el que se han recreado la gran mayoría de GRAFCETs, el lenguaje LD para la implementación de contadores, y el lenguaje ST para la programación de los incrementos en las coordenadas de los distintos brazos de picking.

Estos POU se pueden observar en la columna de la izquierda del programa y se pueden organizar para facilitar el trabajo y la comprensión.

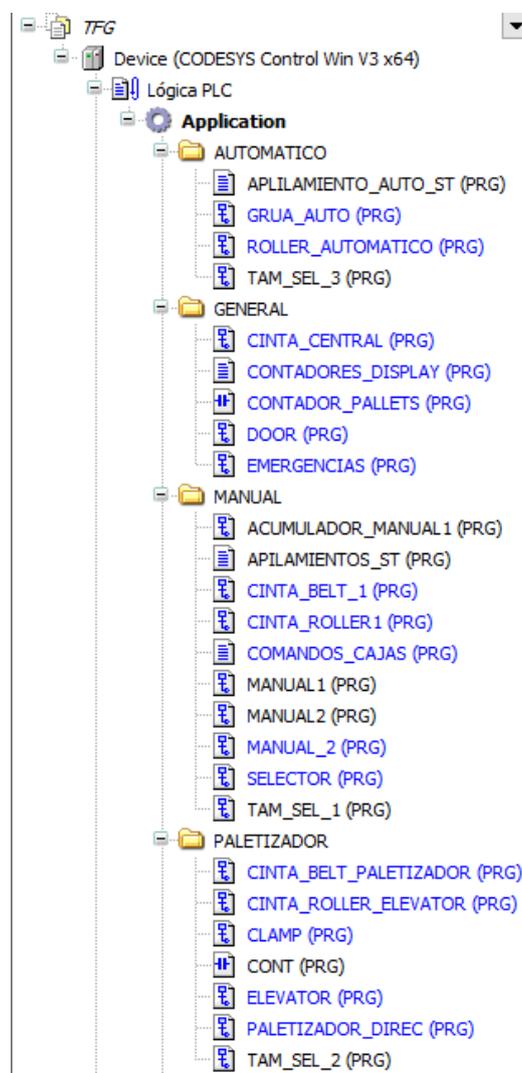


Ilustración 33; Imagen propia, estructura de los POU en el CODESYS

Antes de seguir con el desarrollo del lenguaje SFC y de como este permite recrear fácilmente los GRAFCETs, se explicará de forma muy breve los otros dos lenguajes empleados para recalcar la versatilidad de ofrece *CODESYS*.

TEXTO ESTRUCTURADO (ST):

El texto estructurado ha sido empleado para programas los incrementos y decrementos del brazo de ambas estaciones de picking. Este lenguaje facilita la implementación de estas acciones debido a que nos permite ajustar la velocidad, entre otras ventajas.

```
IF (MANUAL_2.x32_1.x=TRUE OR MANUAL_2.x25_1.x OR MANUAL_2.x25_5.x=TRUE OR MANUAL_2.x28_4.x=TRUE OR MANUAL_2.x31_4.x=TRUE) THEN
  GVL.oX1:=gvl.iX1 +20;
END_IF

IF (MANUAL_2.x32_5.x=TRUE OR MANUAL_2.x32_9.x=TRUE OR MANUAL_2.x25_9.x=TRUE OR MANUAL_2.x28_8.x=TRUE OR MANUAL_2.x31_8.x=TRUE) THEN
  GVL.oX1:=gvl.iX1 -20;
END_IF

IF (MANUAL_2.x32_4.x=TRUE OR MANUAL_2.x25_4.x=TRUE OR MANUAL_2.x28_3.x=TRUE OR MANUAL_2.x31_3.x=TRUE) THEN
  gvl.oY1:=gvl.iY1 +20;
END_IF

IF (MANUAL_2.x32_8.x=TRUE OR MANUAL_2.x25_8.x=TRUE OR MANUAL_2.x28_7.x=TRUE OR MANUAL_2.x31_7.x=TRUE) THEN
  gvl.oY1:=gvl.iY1 -50;
END_IF

IF (MANUAL_2.x32_2.x=TRUE OR MANUAL_2.x25_2.x=TRUE OR MANUAL_2.x28_1.x=TRUE OR MANUAL_2.x31_1.x=TRUE) THEN
  gvl.oZ1:=gvl.iZ1 +20;
END_IF

IF (MANUAL_2.x32_6.x=TRUE OR MANUAL_2.x25_6.x=TRUE OR MANUAL_2.x28_5.x=TRUE OR MANUAL_2.x31_5.x=TRUE) THEN
  gvl.oZ1:=gvl.iZ1 +50;
END_IF

IF (MANUAL_2.x32_3.x=TRUE OR MANUAL_2.x32_7.x=TRUE OR MANUAL_2.x25_3.x=TRUE OR MANUAL_2.x25_7.x=TRUE OR MANUAL_2.x28_2.x=TRUE OR MANU
  gvl.oZ1:=gvl.iZ1 -50;
END_IF
```

Ilustración 34; Imagen propia, ejemplo del lenguaje ST en *CODESYS*

DIAGRAMA DE CONTACTOS (LD):

El diagrama de contactos es el lenguaje de programación más cómodo y sencillo para la programación de contadores, debido a que permite, entre otros, añadir variables de reseteo, variables en las que almacenar la cuenta, etc.

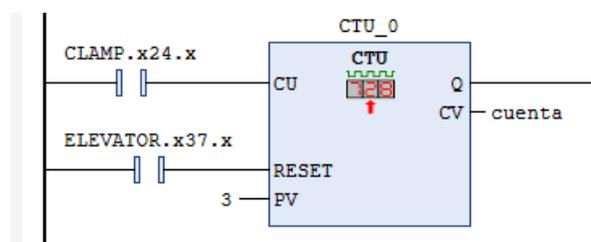


Ilustración 35; Imagen propia, ejemplo del lenguaje LD en *CODESYS*

Una vez mencionados estos dos lenguajes, se explicará el lenguaje SFC.

Como ya se ha mencionado, el lenguaje principal para la implementación de los GRAFCETS será el lenguaje SFC, pero ¿qué es exactamente este lenguaje?

El diagrama funcional secuencial (SFC) es un lenguaje orientado gráficamente que describe el orden cronológico de acciones concretas en un programa. Estas acciones están disponibles como objetos de programación independientes, y están escritas en cualquier lenguaje de programación disponible. En SFC, esas acciones se asignan a elementos de paso y los elementos de transición controlan la secuencia de procesamiento. [18]

Como se puede apreciar por su definición es un lenguaje muy conveniente para poder representar GRAFCETS, sin embargo y pese a las similitudes visuales existen una serie de diferencias que es menester mostrar.

En primer lugar se debe saber que no importa si el número de varias etapas coincide siempre y cuando estén en distintas unidades, debido a que para hacer referencia a cualquier elemento de un diagrama (una etapa, una variable propia del diagrama, etc.) será necesario precederlo del nombre de la unidad. La mecánica es la misma que la usada para llamar variables globales, por ejemplo "GRUA_AUTO.PALLET". En este caso se está llamando a la variable PALLET propia de la unidad GRUA_AUTO. Lo mismo ocurre cuando se quiere emplear una etapa como condición de transición, pero la sintaxis es un poco distinta; en este caso se empleará el nombre la unidad seguido de un punto ("."). A continuación se deberá indicar el nombre de la etapa seguido de un punto equis ("x") para indicar que se hace referencia al estado de la etapa. Un ejemplo sería: "EMERGENCIAS.x2.x", que indica que se empleará el estado de la etapa "x2" de la unidad EMERGENCIAS. Esta sintaxis resulta realmente útil cuando se está trabajando con gran cantidad de programas, para no olvidar a que unidad hace referencia cada elemento.

La siguiente diferencia reside a la hora de expresar una condición de transición temporal, aunque es simplemente una diferencia en cuanto a la sintaxis. Para expresar una condición temporal se deberá escribir primeramente el nombre de la etapa en la que se quiera que se ejecute esta condición de tiempo, seguido de punto te (".t") para indicar que se hace referencia no al estado sino al tiempo. A continuación se establecerá si se desea que sea una igualdad, una superioridad o una inferioridad, y finalmente se colocará "t#" acompañado del número de unidades que se quieren y en que unidad temporal está expresado. Con un ejemplo se comprende mucho mejor: "x34.t >= t#5s".

Finalmente mencionar que para expresar negación se precede la variable de "not" y para expresar una operación lógica de multiplicidad se emplea "AND" en lugar del signo de multiplicar, y lo mismo ocurre para la operación lógica de suma, donde se empleará "OR" en lugar del signo de suma.

Una vez conocidas las diferencias, se mostrará el mismo diagrama representado en GRAFCET y en SFC para apreciarlas de forma visual, y poder empezar la familiarización con este nuevo sistema.

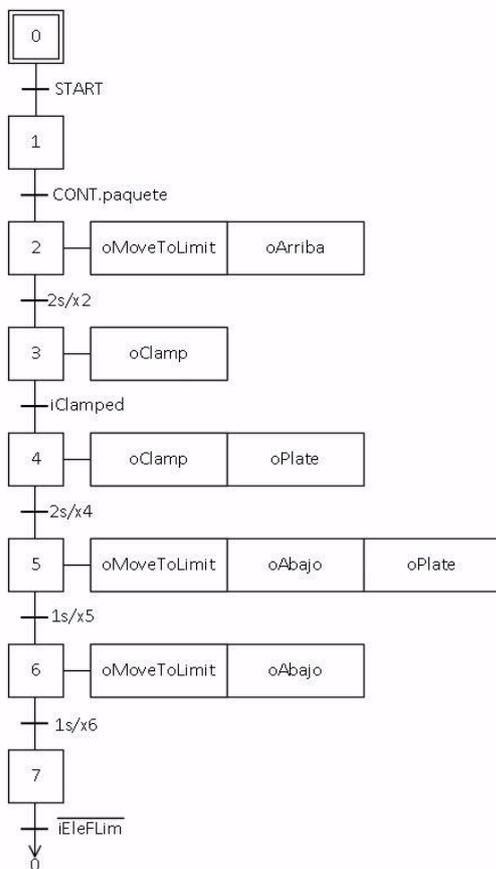


Ilustración 36; Imagen propia, comparación GRAFCET

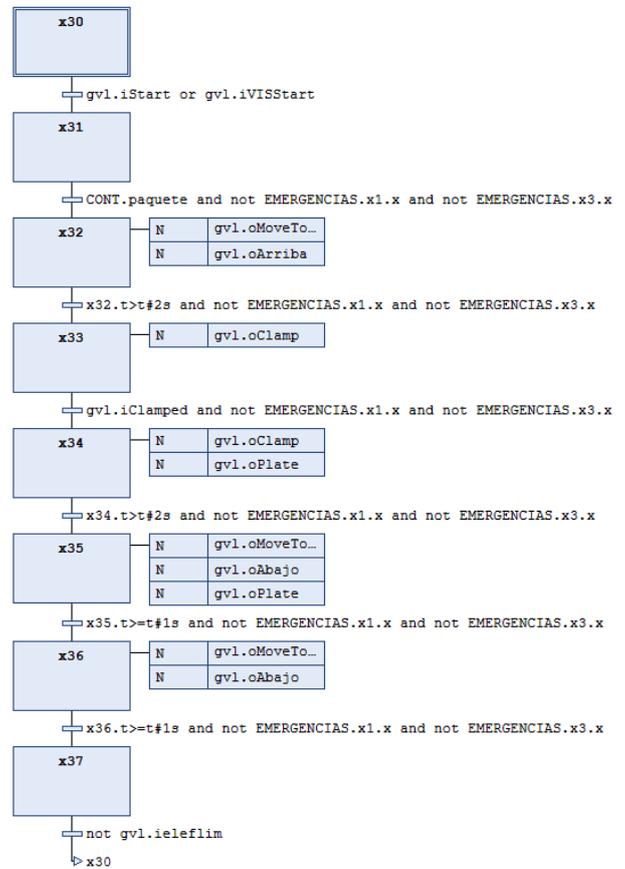


Ilustración 37; Imagen propia, comparación POU

Por otro lado pero siguiendo con la temática de los POU, es necesario añadir que la creación de un POU no implica que este vaya a tomar parte en el proyecto. Para que este se implique en el proceso será necesario añadirlo a la lista de tareas principales, o “Main Tasks”, simplemente arrastrando el POU a esta categoría. Esto puede parecer que no suponga una gran diferencia, incluso puede llegar a parecer inútil, pero dado el caso de estar generando un proyecto puede llegar a ser increíblemente útil ya que facilita la detección de errores, o la simplicidad en las pruebas realizadas para comprobar el correcto funcionamiento debido a que se pueden eliminar elementos que relentsen el proceso o que lo impidan. Durante la realización de este proyecto fue de mucha utilidad ya que permitía desactivar todos los POU referentes a la zona A para poder testear tranquilamente que todo funcionara correctamente en la zona B. Además esta categoría permite elegir como se va a desarrollar el proyecto (cíclico, libre...), la duración de los intervalos e incluso la posibilidad de añadir un Watchdog. Sin embargo en este proyecto no se han modificado.

4.3.3 – IMPLEMENTACIÓN DE LA COMUNICACIÓN CODESYS-FACTORY IO

Para poder realizar la explicación de como funcionan las distintas conexiones se volverá a exponer la imagen que las resumen, de nuevo como referencia visual.

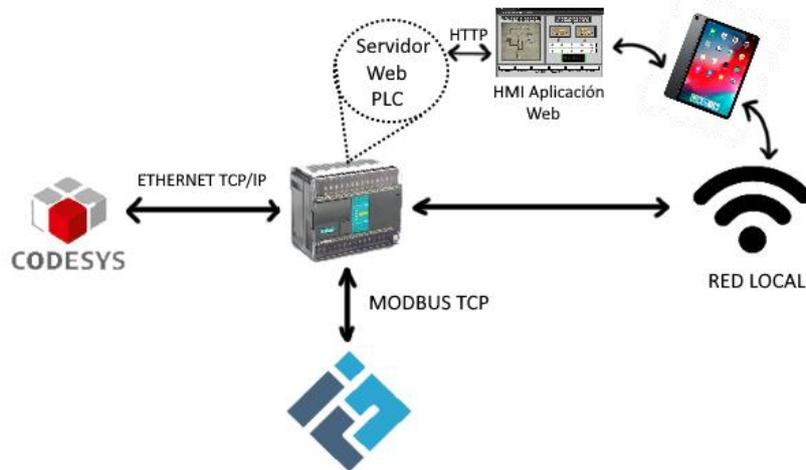


Ilustración 38; Imagen propia, esquema de elementos y conexiones que intervienen en el sistema

4.3.3.1 – ETHERNET TCP/IP

Ethernet/IP es un protocolo de red en niveles para aplicaciones de automatización industrial. Basado en los protocolos estándar TCP/IP, utiliza los ya bastante conocidos hardware y software Ethernet para establecer un nivel de protocolo para configurar, acceder y controlar dispositivos de automatización industrial. En este proyecto esta conexión se ha empleado para establecer las configuraciones entre el software de programación *CODESYS* y el PLC virtual que trae implementado el propio programa. Dentro de esta opción, será necesaria la correcta disposición de las variables, ya que deben coincidir con la distribución realizada en *FACTORY IO* en cuanto este trate de leerlas desde el PLC. [19]

Variable	Asignación	Canal	Dirección	Tipo	Unidad	Descripción
		Entradas	%IW0	ARRAY [0..11] OF WORD		Registros Holding Modbus
		Entradas[0]	%IW0	WORD		
		Entradas[1]	%IW1	WORD		
		Entradas[2]	%IW2	WORD		
		Entradas[3]	%IW3	WORD		
		Entradas[4]	%IW4	WORD		
		Entradas[5]	%IW5	WORD		
		Entradas[6]	%IW6	WORD		
		Entradas[7]	%IW7	WORD		
		Entradas[8]	%IW8	WORD		
		Entradas[9]	%IW9	WORD		
		Entradas[10]	%IW10	WORD		
		Entradas[11]	%IW11	WORD		
		Salidas	%QW0	ARRAY [0..11] OF WORD		Registros Input Modbus
		Salidas[0]	%QW0	WORD		
		Salidas[1]	%QW1	WORD		
		Salidas[2]	%QW2	WORD		
		Salidas[3]	%QW3	WORD		
		Salidas[4]	%QW4	WORD		
		Salidas[5]	%QW5	WORD		
		Salidas[6]	%QW6	WORD		
		Salidas[7]	%QW7	WORD		
		Salidas[8]	%QW8	WORD		
		Salidas[9]	%QW9	WORD		
		Salidas[10]	%QW10	WORD		
		Salidas[11]	%QW11	WORD		

Ilustración 39; Imagen propia, distribución de las variables y dirección en CODESYS

Nota: la disposición está distribuida de forma muy intuitiva, manteniendo todas las entradas en la parte superior, y todas las salidas en la parte inferior. Además, será necesario que el tipo de dato coincida con la dirección en la que se encuentra.

4.3.3.2 – MODBUS TCP

Modbus es un protocolo de comunicaciones basado en la arquitectura maestro/esclavo (RTU) o cliente/servidor (TCP/IP). Convertido en un protocolo de comunicaciones estándar de facto en la industria, es el que goza de mayor disponibilidad para la conexión de dispositivos electrónicos industriales. Es el tipo de comunicación empleado para conectar el PLC con el software de validación *FACTORY IO*. [20]

Como se ha mencionado previamente, es un tipo de conexión basado en la arquitectura cliente/servidor, donde el *FACTORY IO* tomará el rol de cliente, y el PLC de *CODESYS* tomará el papel de servidor. Esto significa que será el cliente, en este caso el *FACTORY IO*, será quien iniciará las conexiones, demandando una lectura o escritura de las variables en sus correspondientes espacios de memoria, provenientes del PLC de *CODESYS*, que toma el rol de Servidor.

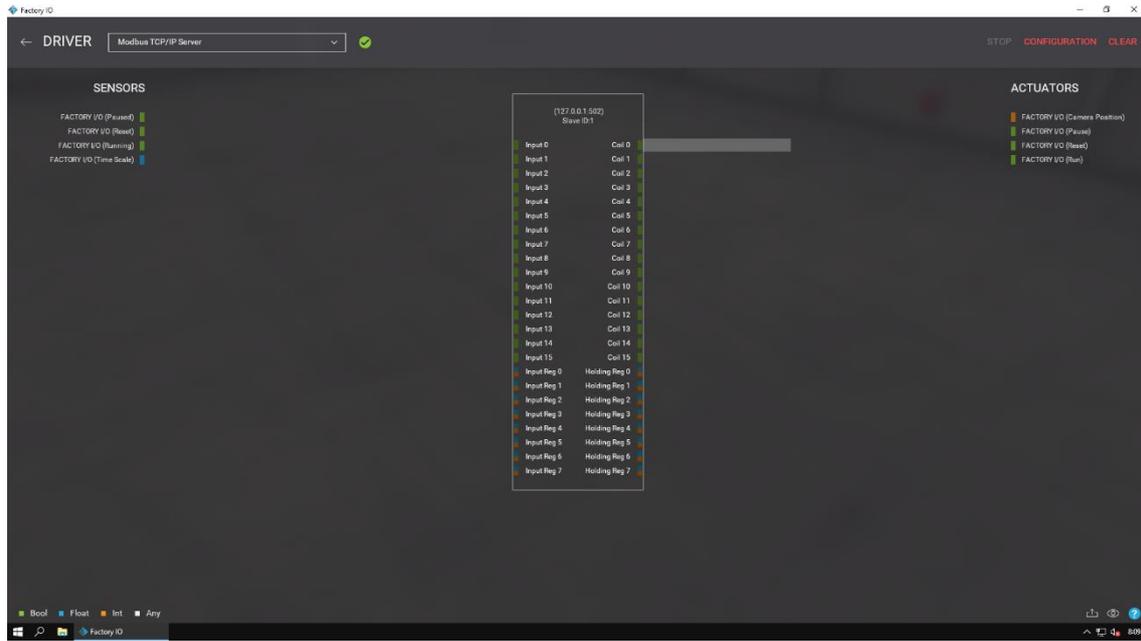


Ilustración 40; Imagen propia, esquema de variables en el *FACTORY IO*

4.3.4 – IMPLEMENTACIÓN DE LA INTERFAZ DE USUARIO

Uno de los principales atractivos que tiene *CODESYS* es su interfaz de usuario, la cual se aloja en un servidor web que trae embebido en el firmware del PLC. Esto es realmente cómodo puesto que nos permite acceder a esta interfaz y manipularla a nuestro antojo desde cualquier dispositivo conectado a la misma subred que el PLC, así como desde el propio programa.

En el presente apartado se explicarán los distintos pasos seguidos para realizar una interfaz de usuario, así como los objetos funcionales y configuraciones que se han implementado para conseguir que esta sea lo más sencilla, intuitiva y útil posible.

El proceso para añadir esta aplicación de visualización es muy sencillo, y bastará con hacer click derecho sobre la opción de “Application” -> Agregar objeto -> Visualización. Esto generará automáticamente el editor donde se podrá crear y modificar tanto la interfaz como la aplicación alojada en el servidor web.

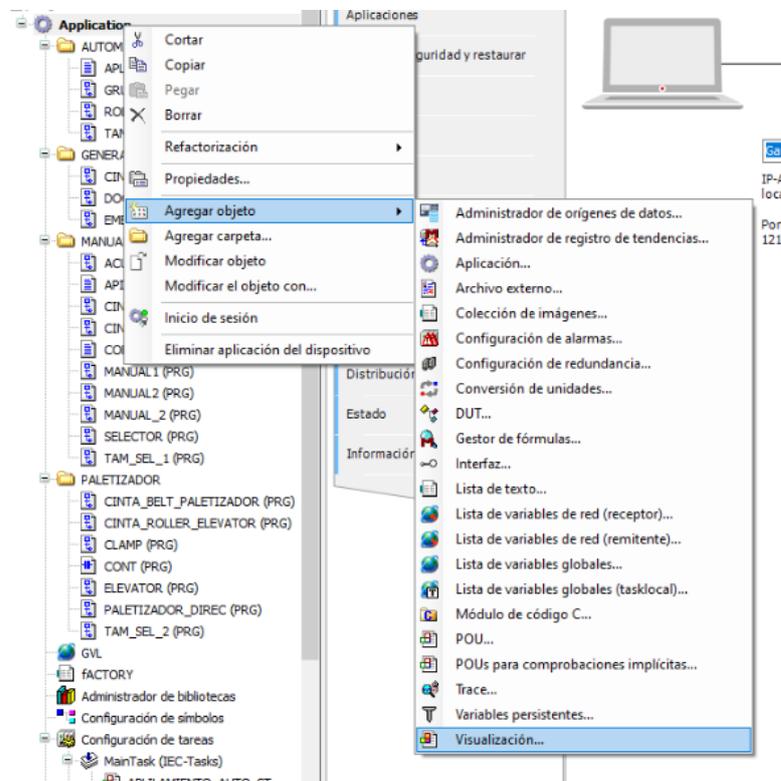


Ilustración 41; Imagen propia, ruta para agregar la pantalla de operador

Una vez dentro del apartado de visualización se puede observar como a la derecha aparece un menú con tres pestañas distintas en la parte inferior. De estas se trabajará principalmente con dos: “Herramientas” y “Cuadro de herramientas Visualizador”.

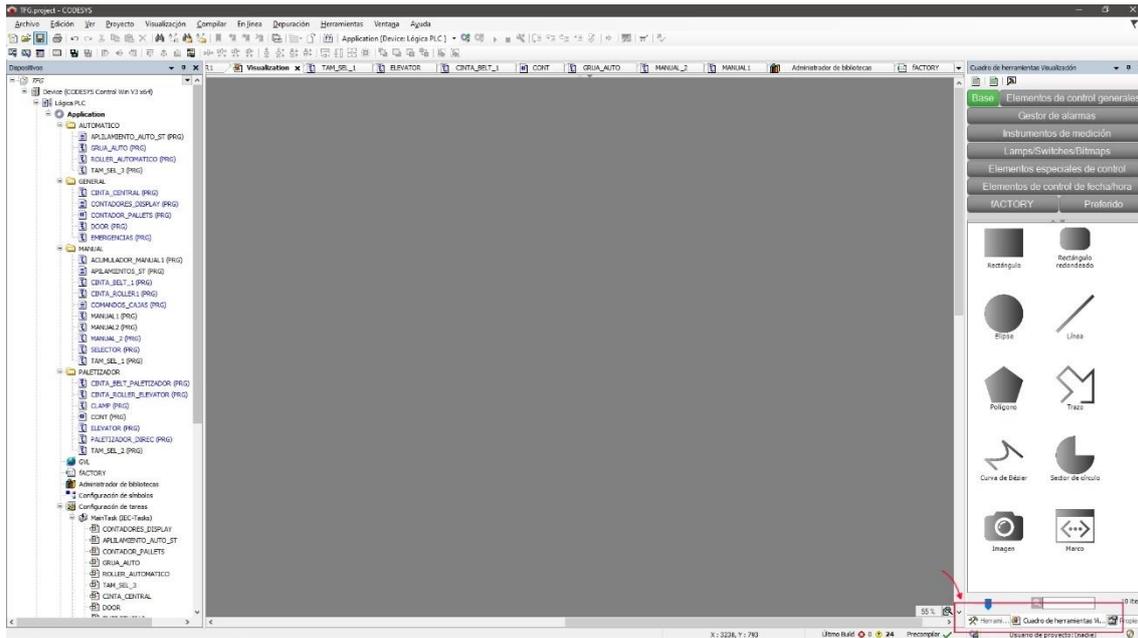


Ilustración 42; Imagen propia, explicación de las pestañas de la pantalla de operador

Por un lado la pestaña de “Cuadro de herramientas Visualizador” permite el acceso a decenas de elementos, agrupados en las distintas pestañas que aparecen en la parte superior derecha, y que se detallarán en profundidad al explicar los elementos que componen el visualizador del proyecto. Por el otro lado la pestaña “Herramientas” permite modificar todas las propiedades de estos elementos; desde el color, hasta el tamaño, la posición, e incluso la variable que los gobierna si el elemento lo permite.

En el caso del visualizador de este proyecto se van a explicar únicamente los elementos funcionales, mencionando únicamente aquellos que son puramente estéticos.

4.3.4.1 – ELEMENTOS FUNCIONALES

Se entiende por elementos funcionales aquellos que tienen un impacto directo sobre la interfaz de usuario y pueden cambiar su estado. Cabe destacar que a todos estos objetos funcionales se les puede asignar el nombre que se desee o no asignarle ninguno.

Lámparas

Esta función permite mostrar información de una dirección determinada mediante el valor de un bit en concreto. Para ello se leerá el valor del bit del que se quiera mostrar información y se encenderá o no la bombilla en función del mismo. Trabajan siempre con variables booleanas.

Esta función permite conocer en cada momento el estado de una variable y es realmente útil para saber el todo momento si ciertos elementos están actuando correctamente.

En el visualizador de este proyecto se han empleado estos LEDs lumínicos para conocer el estado de todas las cintas de la planta, así como el estado de los botones tanto de los distintos modos (parada, mantenimiento, etc.) como de las distintas preformas.

Para poder atribuir una variable a estos elementos se deberá ir a la pestaña “Propiedades” mencionada anteriormente y añadir el nombre de la variable en la casilla “Variable”. Esto es igual en todos los elementos que se mostrará.

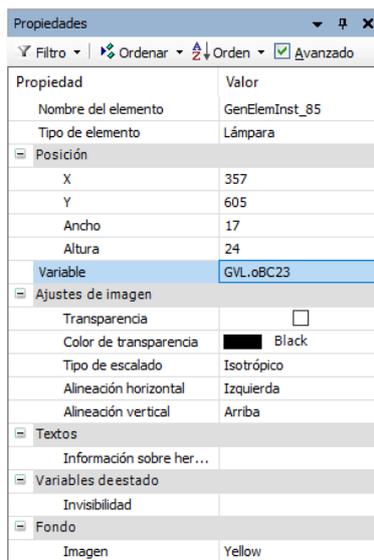


Ilustración 43; Imagen propia, selección de propiedades de los elementos de la pantalla de operador

Finalmente añadir que se puede cambiar el color de estas bombillas en el apartado Imagen, función muy útil visualmente para poder distinguir distintos modos de funcionamiento, por ejemplo.

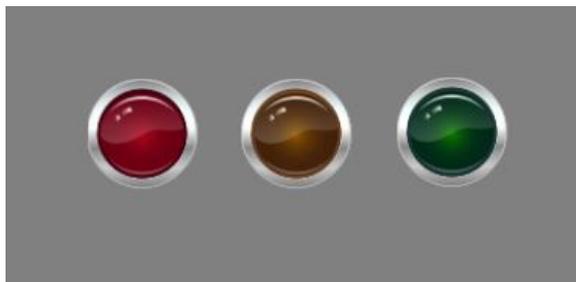


Ilustración 44; Imagen propia, tipos de LEDs que ofrece la pantalla de operador

Interruptor de energía

Esta función permite activar o desactivar un bit escribiendo en la variable a la que se está refiriendo. Para realizar este cambio bastará con clicar encima del interruptor deseado. De cierto modo, combina las acciones de lectura y escritura de una variable, ya que el estado “Off” representa que el valor de esa variable es “cero” mientras que, por el contrario, el interruptor está en estado “On”, el valor de la variable es “uno”.

Por lo tanto, al igual que la bombilla, los interruptores hacen referencia siempre a variables de tipo boolean ya que solamente disponen de dos estados.

En este caso los interruptores de energía han sido empleados para seleccionar los distintos estados del proceso, ya que es conveniente mantener este estado hasta que se desee desactivarlo manualmente.



Interruptor de ener...

Ilustración 45; Imagen propia, interruptor de energía de la pantalla de operador

Interruptor de presión

Esta función es similar a la anterior, pero con la diferencia que no mantiene la posición de “On” de manera indefinida, sino que la activa de forma momentánea y luego la desactiva; es decir, escribe un uno en la variable a la que haga referencia durante tan solo un instante.

Esta función es muy útil para la selección de preformas, ya que de este modo se permite el cambio entre ellas con una sola pulsación, y sin necesidad de desconectar el interruptor que vaya a dejarse de usar previamente.



Interruptor de presión

Ilustración 46; Imagen propia, interruptor de presión de la pantalla de operador

Campo de texto

Esta función permite mostrar en formato texto una variable de tipo INT o WORD. No provoca ningún cambio sobre la variable, tan solo muestra su estado actual.

Esta función viene bien para mostrar por pantalla un seguimiento del número de pallets que se van generando en cada zona. Esta información puede ser muy valiosa por ejemplo si se exigen únicamente diez pallets de cada preforma.

Para que este elemento muestre la variable que nosotros deseemos por pantalla se deberá ir a la pestaña “Herramientas” y a continuación a “Variables de texto”, donde se introducirá la variable que se quiera mostrar por pantalla.

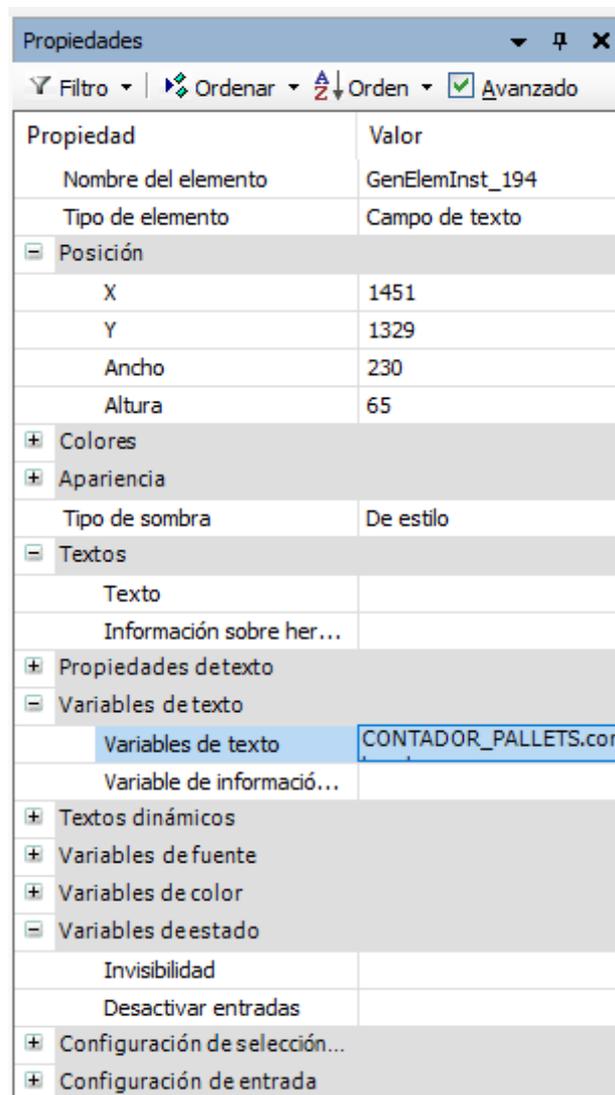


Ilustración 47; Imagen propia, indicaciones para mostrar variables mediante texto

4.3.4.2 – ELEMENTOS DECORATIVOS

Para realizar la pantalla de operador se han añadido gran cantidad de elementos para mejorar la estética de esta. Estos son principalmente rectángulos o líneas, que se pueden añadir fácilmente desde el menú de visualización. Sin embargo para añadir imágenes el proceso no es tan sencillo.

Si se desea añadir una imagen cualquiera a la pantalla de operador se tendrá que añadir previamente una “Colección de imágenes”.

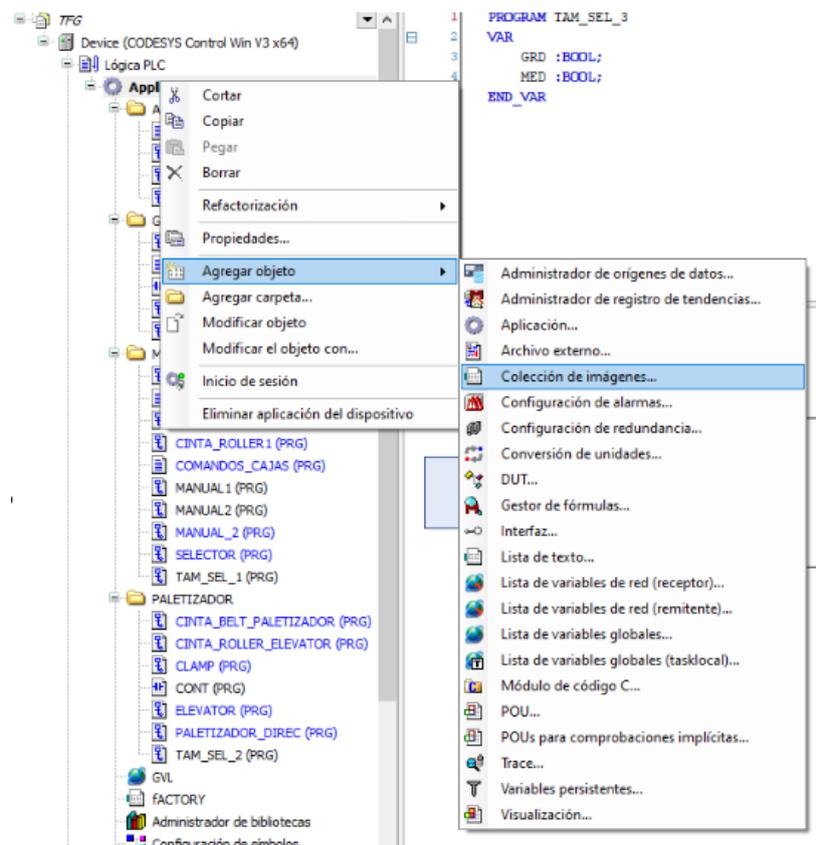


Ilustración 48; Imagen propia, ruta para añadir imágenes externas al software

Una vez creado este elemento, bastará con clickar dos veces sobre él, y posteriormente clickar en la casilla “Nombre de archivo”, donde se permitirá subir cualquier imagen almacenada en el ordenador.

Una vez finalizada la explicación de los distintos elementos que conforman la pantalla de operador solo resta ver el resultado creado para este proyecto, fruto de la combinación de todos ellos.

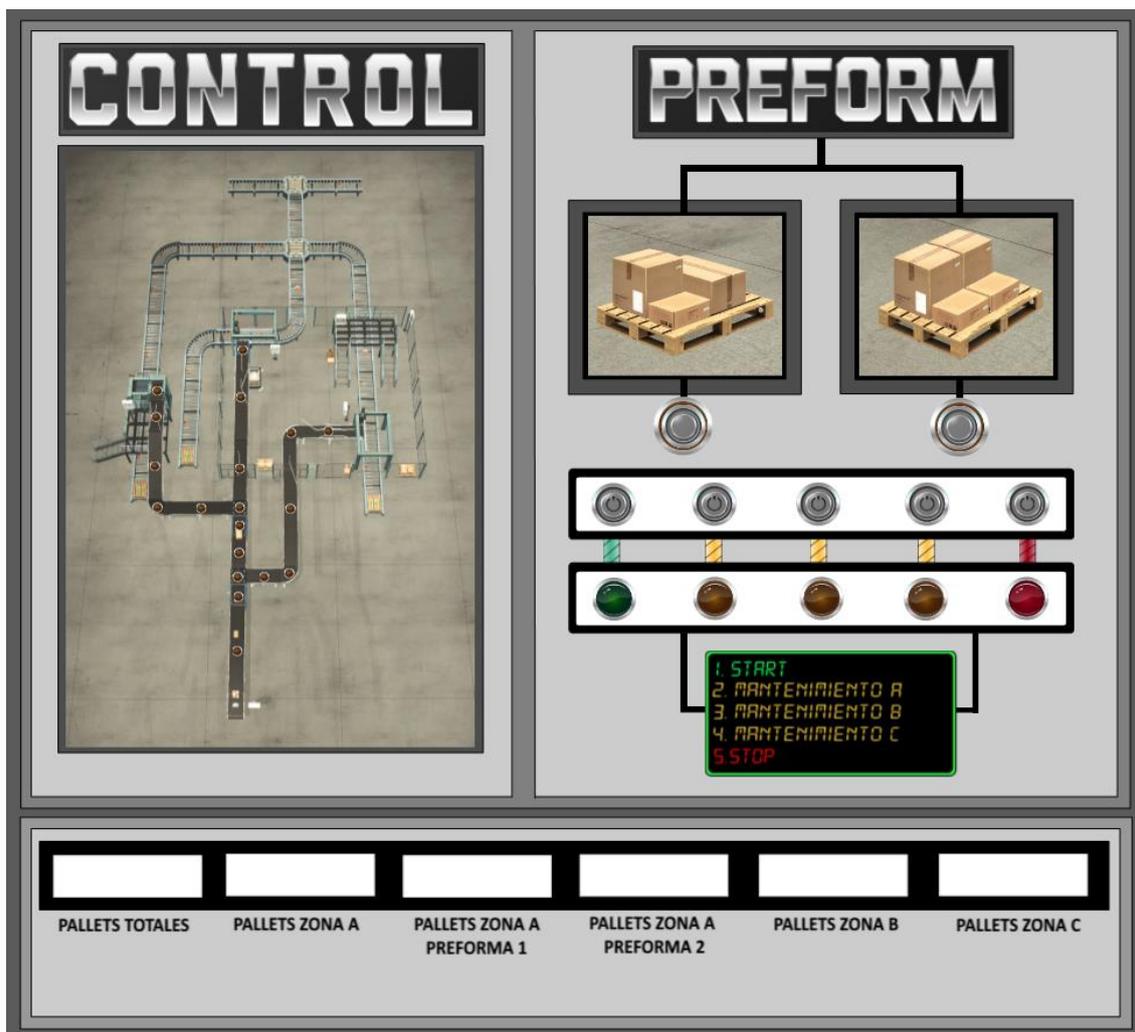


Ilustración 49; Imagen propia, diseño final de la pantalla de operador del proyecto

Como se puede observar están presentes todos los elementos descritos anteriormente: las lámparas que indican los distintos estados del proceso, así como la preforma seleccionada (debajo del interruptor de presión) y del estado de todas las cintas del proceso, los interruptores de presión que permiten controlar la preforma seleccionada, y los de energía que permiten activar y desactivar de una forma cómoda los distintos modos de funcionamiento, y finalmente los campos de texto que permiten conocer a tiempo real el número de pallets que ese está realizando en cada una de las distintas zonas del proyecto.

Así mismo también se puede observar los elementos decorativos que dotan de personalidad y originalidad a la pantalla de operador.

Para concluir mencionar que las instrucciones para inicializar de forma segura el proyecto así como los pasos necesarios para abrir el visualizador Web está incluidos en el Anexo III: Pantalla de Operador.

5.CONCLUSIONES

Tras la finalización del proyecto, con todo lo que ello ha conllevado, se puede corroborar que se han cumplido los objetivos y que se ha llegado a unas valiosas conclusiones.

Estas conclusiones reflejan tanto la adquisición de conocimientos como la evolución personal que ha tenido lugar a lo largo de todo este trabajo académico de final de grado.

En cuanto a conclusiones técnicas:

- Es importante conocer los aspectos físicos y funcionales de los elementos a automatizar además de las conexiones realizadas para la buena asignación de las variables de E/S del sistema.
- Hay una amplia gama de PLCs y programas para automatizarlos, por ello, es muy importante elegir aquel que cumpla los requisitos y cuyo lenguaje se ajuste más a las características del proceso, optimizando siempre la solución.
- Es de vital importancia la comunicación entre dispositivos ya que todos deben de estar conectados y coordinados perfectamente para la transferencia de datos.
- Siempre hay que realizar la automatización en vistas a la seguridad del equipo y de las personas
- La realización de una aplicación SCADA resulta muy útil para visualizar el proceso productivo, historiar acontecimientos y cambiar el valor de parámetros.
- Se corrobora que *CODESYS* es una herramienta muy potente y eficaz para tratar este tipo de proyectos.
- El conocimiento de las leyes y normativas vigentes es un apartado que se debe tener muy en cuenta para la realización de proyectos.
- La redacción del mismo debe ser concisa y clara para una buena implementación y puesta en marcha.

A nivel personal:

- El diseño corresponde con una parte clave en la realización del proyecto siendo tan importante como la implementación, por ello, se debe invertir el tiempo suficiente en diseñar el proyecto.
- Se requiere de tiempo y atención para lograr que todos los procesos funcionen correctamente al unísono, y es importante afrontar los problemas con calma.
- Es importante tomar anotaciones e ir recolectando información según avanza el proyecto puesto que esto aportará facilidad en la posterior explicación.
- Un proceso productivo se puede automatizar de varias maneras y es por ello que se debe conocer el funcionamiento y especificaciones que el cliente desea.



- Se debe tener presente en todo momento que se está realizando un trabajo que puede herir a las personas si se realiza de forma incorrecta, y por tanto hay que tener extrema precaución a la hora de tomar cualquier decisión.
- Este proyecto me ha ayudado a consolidar muchos de los conocimientos adquiridos durante la carrera y ha conseguido aumentar mi interés por los procesos de automatización.

6. BIBLIOGRAFÍA

- [1] Cuarta revolución industrial https://es.wikipedia.org/wiki/Revoluci%C3%B3n_industrial_etapa_cuatro
- [2] «AENOR - Buscador de normas», s. f. <https://www.aenor.com/normas-y-libros/buscador-de-normas/>
- [3] Lógica cableada <https://dissenyproducte.blogspot.com/2010/12/logica-cableada-y-logica-programada.html>
- [4] Lógica programada <https://polaridad.es/logica-programada-programable-que-es-pld-fpga-hdl-cpld/>
- [5] Microcontrolador <https://es.wikipedia.org/wiki/Microcontrolador>
- [6] Controlador Lógico Programable (PLC) https://es.wikipedia.org/wiki/Controlador_l%C3%B3gico_programable
- Carlos Valvidia Miranda, (2019). Comunicaciones industriales
- [7] Software *UNITY PRO* https://store.unity.com/products/unity-pro?gclid=CjwKCAjwi_b3BRAGEiwAemPNU_oVWT57yTSvPsJkwcSuVrOGD-Ull2Ind4EHnrQL4b-9o0CCvUFH6RoC0GgQAvD_BwE
- [8] Software *CODESYS* <https://www.codesys.com/>
<https://es.wikipedia.org/wiki/CoDeSys>
- [9] Software *autoSIM-200* <https://www.smctraining.com/es/webpage/indexpage/108>
- [10] Software *FACTORY IO* <https://factoryio.com/features>
- [11] Human-Machine Interface (HMI) <http://aumezar.es/ES/aumezar cms.php?tkn=16#1>
- [12] Servidor Web <https://www.novatechweb.com/substation-automation/webserver-hmi/>
- [13] Gemelo virtual (digital twin) <https://www.technologyreview.es/s/10696/digital-twin-un-gemelo-virtual-para-aconsejar-la-industria-40>
- [14] Partes *FACTORY IO* <https://docs.factoryio.com/manual/parts/>
- [15] Conexiones *FACTORY IO* – *CODESYS* <https://docs.factoryio.com/tutorials/codesys/setting-up-codesys-mb/index.html>
- [16] GRAFCET <https://es.wikipedia.org/wiki/GRAFCET>
- [17] Program Organization Unit (POU) <https://www.motioncontroltips.com/what-are-program-organization-units-in-plc-programming/>
- [18] Sequential Function Chart (SFC) https://es.wikipedia.org/wiki/Sequential_Function_Chart



[19] Ethernet TCP/IP https://www.siemon.com/la/white_papers/03-10-13-ethernet-ip.asp#:~:text=Ethernet%2FIP%20es%20un%20protocolo,controlar%20dispositivos%20de%20automatizaci%C3%B3n%20industrial.

[20] Modbus TCP <https://es.wikipedia.org/wiki/Modbus>

Antón Girod Fortuño, (2017). Guía teórico práctica de Modbus en Arduino



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCOLA TÈCNICA
SUPERIOR ENGINYERIA
INDUSTRIAL VALÈNCIA

TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

ANEXO I: DISEÑO DEL SISTEMA AUTOMÁTICO EN GRÁFCET

Autor: Ramón Prades Batalla

Tutor: Raúl Simarro Fernández

Curso académico: 2019/2020



ÍNDICE DE CONTENIDOS DEL ANEXO I

1 – OBJETO.....	77
2 – ZONA 0: ZONA GENERAL.....	77
2.1 – EMERGENCIAS.....	77
2.2 – CINTA_CENTRAL.....	77
2.3 – DOOR.....	78
3 – ZONA A: PICKING PREFORMA.....	78
3.1 – CINTA_BELT_1.....	78
3.2 – CINTA_ROLLER_1.....	79
3.3 – TAM_SEL_1.....	79
3.4 – MANUAL1.....	80
3.5 – MANUAL2.....	81
3.6 – ACUMULADOR_MANUAL_1.....	82
3.7 – MANUAL_2.....	83
3.8 – SELECTOR.....	84
4 – ZONA B: ZONA PALETIZADOR.....	85
4.1 – CINTA_BELT_PALETIZADOR.....	85
4.2 – CINTA_ROLLER_ELEVATOR.....	86
4.3 – CLAMP.....	86
4.4 – ELEVATOR.....	87
4.5 – TALM_SEL_2.....	88
4.6 – PALETIZADOR_DIREC.....	88
5 – ZONA C: ZONA PICKING ALEATORIO.....	89
5.1 – ROLLER_AUTOMATICO.....	89
5.2 – TAM_SEL_3.....	89
5.3 – GRUA_AUTO.....	90

ÍNDICE DE FIGURAS DEL ANEXO I

Ilustración 1; Imagen propia, GRAFCET EMERGENCIAS.....	77
Ilustración 2; Imagen propia, GRAFCET CINTA_CENTRAL.....	77
Ilustración 3; Imagen propia, GRAFCET DOOR.....	78

Ilustración 4; Imagen propia, GRAFCET CINTA_BELT_1.....	78
Ilustración 5; Imagen propia, GRAFCET CINTA_ROLLER_1.....	79
Ilustración 6; Imagen propia, GRAFCET TAM_SEL_1	79
Ilustración 7; Imagen propia, GRAFCET MANUAL1	80
Ilustración 8; Imagen propia, GRAFCET MANUAL2	81
Ilustración 9; Imagen propia, GRAFCET ACUMULADOR_MANUAL1.....	82
Ilustración 11; Imagen propia, GRAFCET MANUAL_2.....	83
Ilustración 12; Imagen propia, GRAFCET SELECTOR	84
Ilustración 13; Imagen propia, GRAFCET CINTA_BELT_PALETIZADOR	85
Ilustración 14, Imagen propia, GRAFCET CINTA_ROLLER_ELEVATOR.....	86
Ilustración 15; Imagen propia, GRAFCET CLAMP.....	86
Ilustración 16; Imagen propia, GRAFCET ELEVATOR.....	87
Ilustración 17; Imagen propia, GRAFCET TAL_SEL_2	88
Ilustración 18, Imagen propia, GRAFCET PLAETIZADOR_DIRECT.....	88
Ilustración 19; Imagen propia, GRAFCET ROLLER_AUTOMATICO	89
Ilustración 20; Imagen propia, GRAFCET TAM_SEL_3	89
Ilustración 21; Imagen propia, GRAFCET GRUA_AUTO.....	90
Ilustración 22; Imagen propia, macroetapa M8.....	91

1 – OBJETO

En este documento se muestran de forma ininterrumpida todos los GRAFCETs parciales que conforman el proyecto. Mencionar que solo se ha incluido una macro en cada zona debido a que su estructura y funcionamiento es idéntico.

2 – ZONA 0: ZONA GENERAL

2.1 – EMERGENCIAS

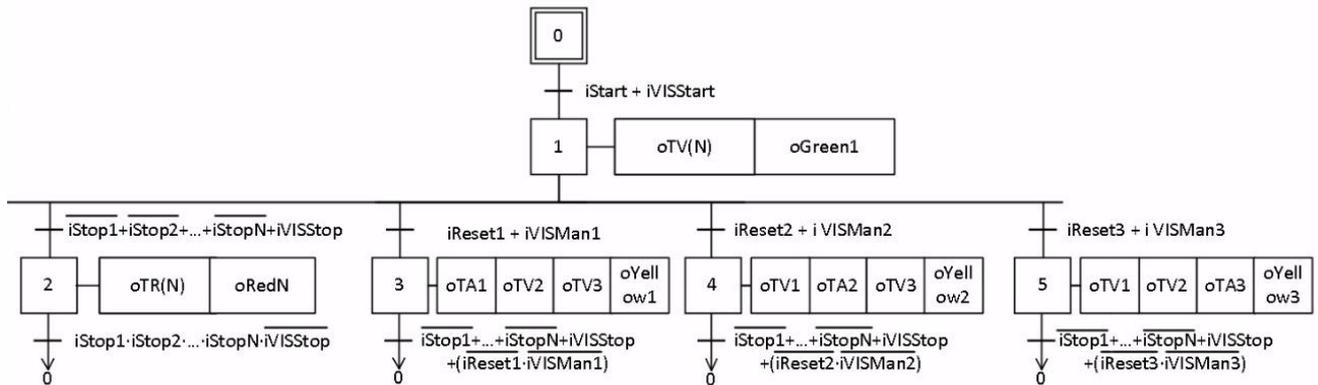


Ilustración 1; Imagen propia, GRAFCET EMERGENCIAS

2.2 – CINTA_CENTRAL

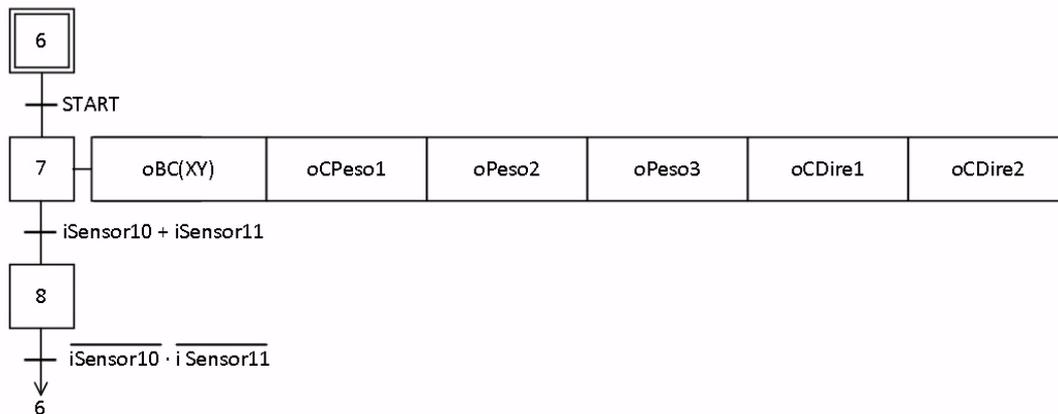


Ilustración 2; Imagen propia, GRAFCET CINTA_CENTRAL

2.3 – DOOR

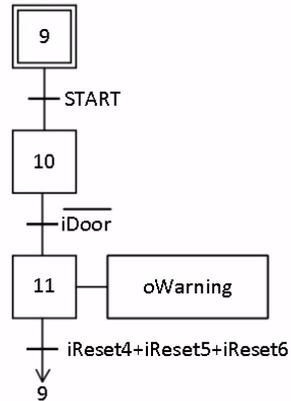


Ilustración 3; Imagen propia, GRAFCET DOOR

3 – ZONA A: PICKING PREFORMA

3.1 – CINTA_BELT_1

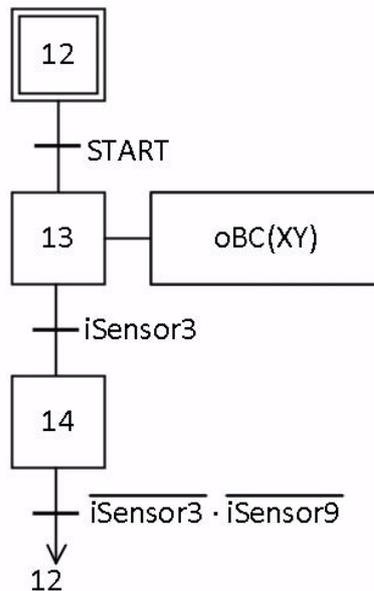


Ilustración 4; Imagen propia, GRAFCET CINTA_BELT_1

3.2 – CINTA_ROLLER_1

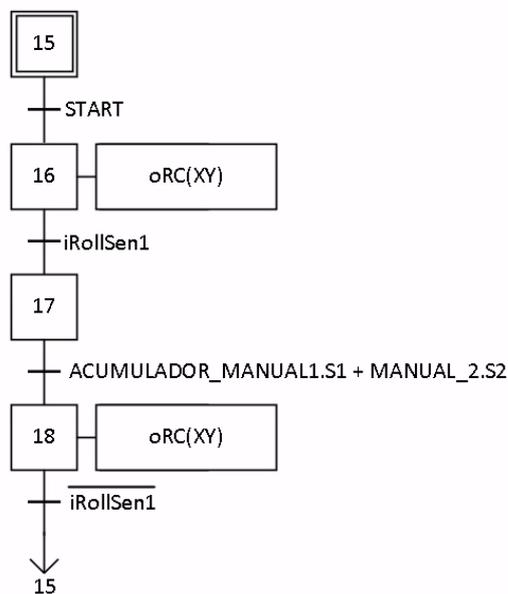


Ilustración 5; Imagen propia, GRAFCET CINTA_ROLLER_1

3.3 – TAM_SEL_1

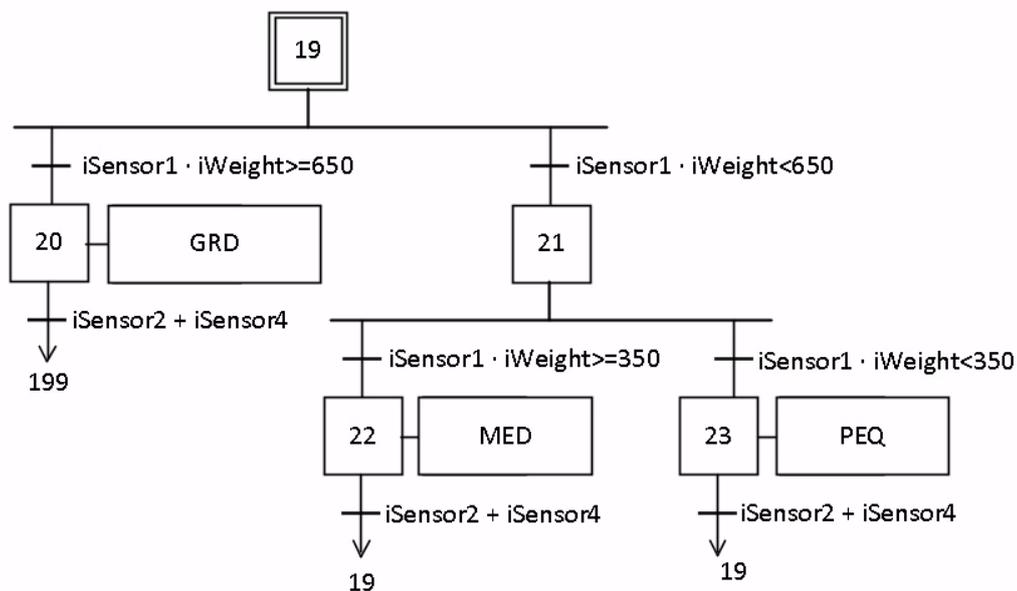


Ilustración 6; Imagen propia, GRAFCET TAM_SEL_1

3.4 – MANUAL1

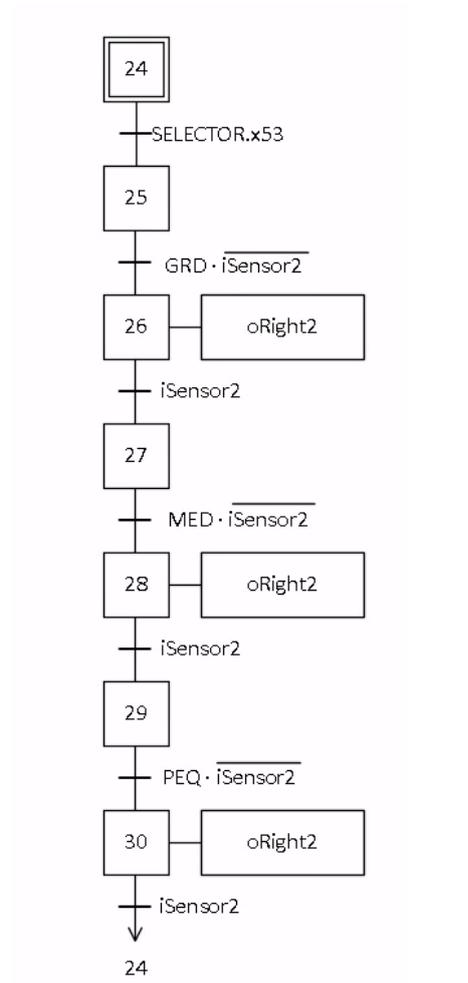


Ilustración 7; Imagen propia, GRAFCET MANUAL1

3.5 – MANUAL2

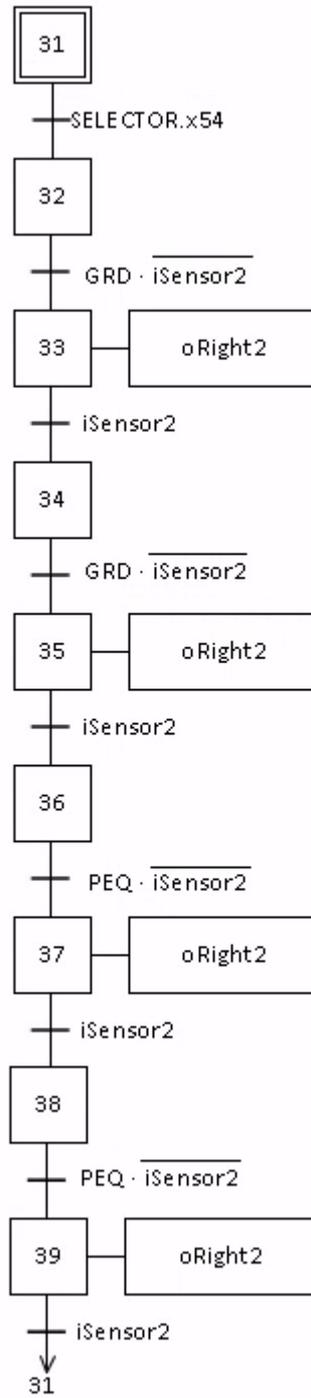


Ilustración 8; Imagen propia, GRAFCET MANUAL2

3.6 – ACUMULADOR_MANUAL_1

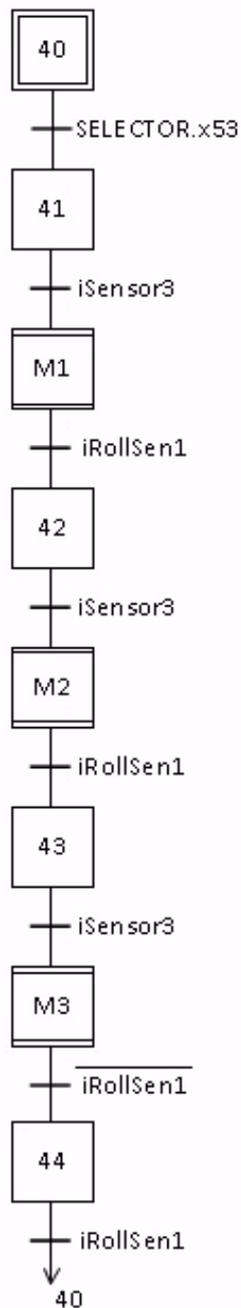


Ilustración 9; Imagen propia, GRAFCET ACUMULADOR_MANUAL1

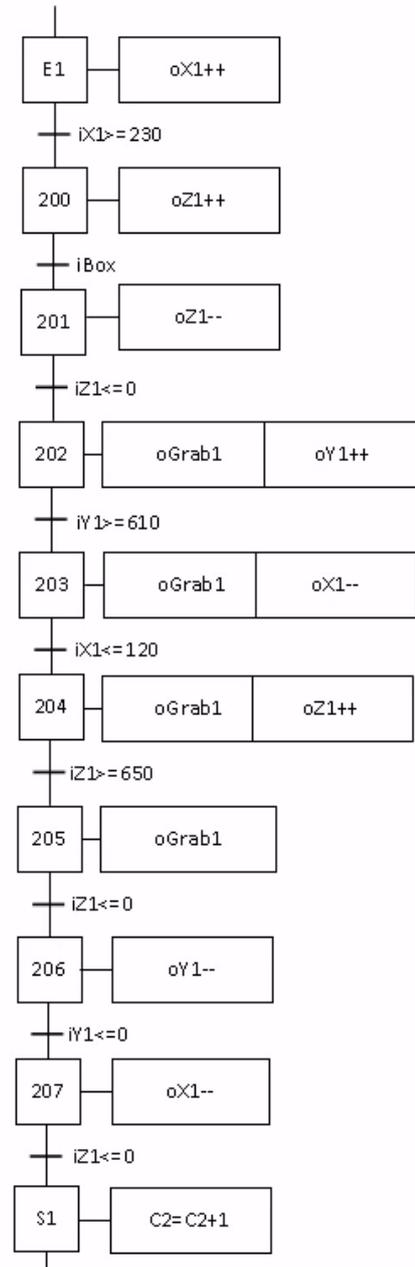


Ilustración 10; Imagen propia, macroetapa M1

3.7 – MANUAL_2

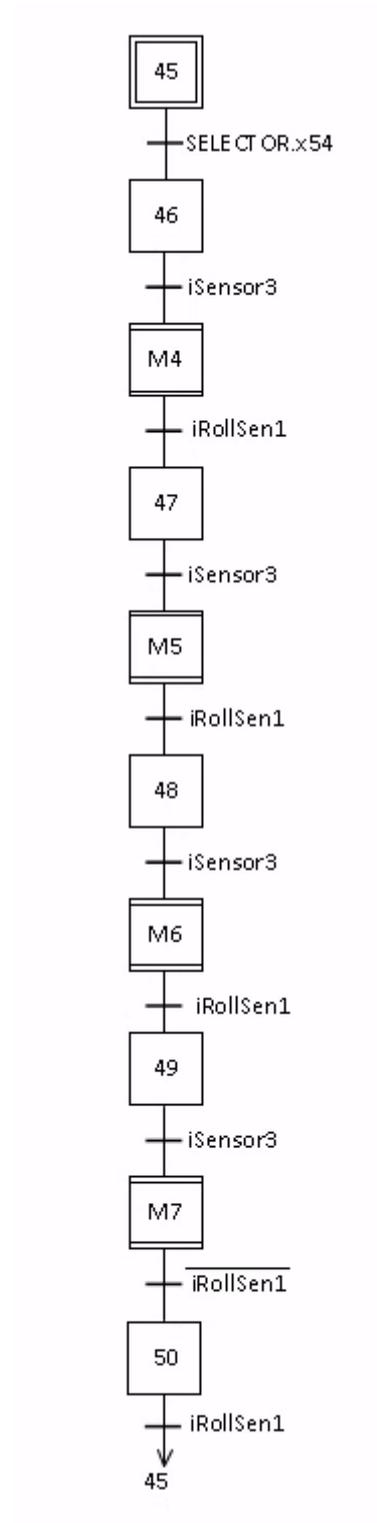


Ilustración 11; Imagen propia, GRAFCET MANUAL_2

3.8 – SELECTOR

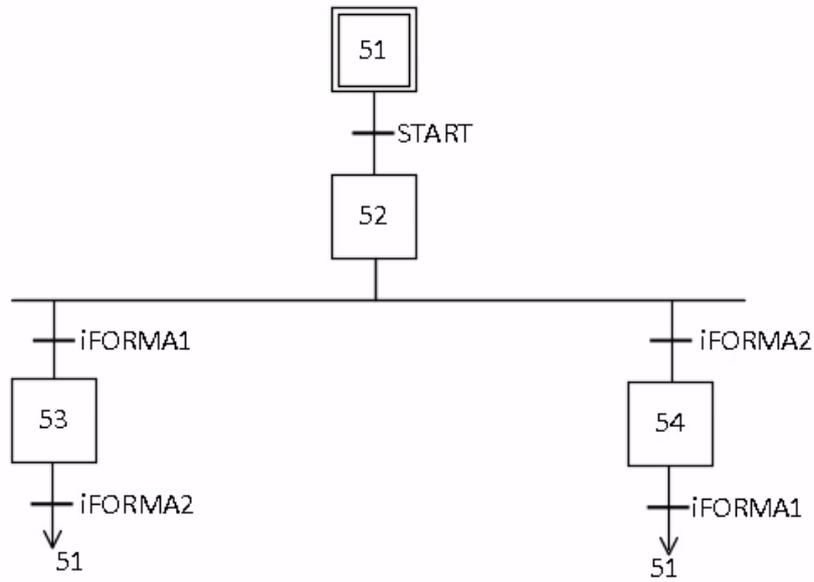


Ilustración 12; Imagen propia, GRAFCET SELECTOR

4 – ZONA B: ZONA PALETIZADOR

4.1 – CINTA_BELT_PALETIZADOR

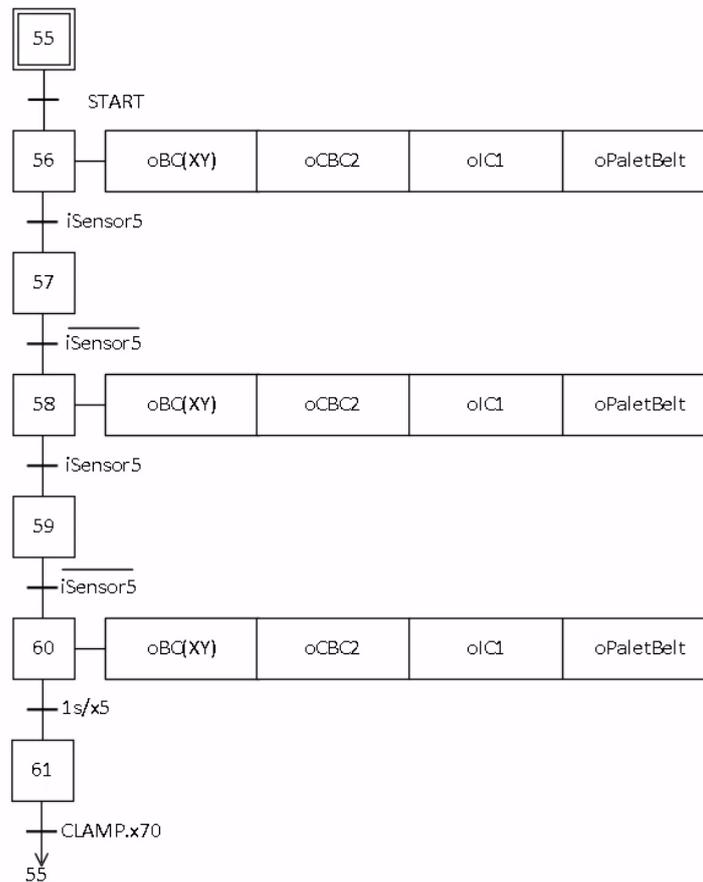


Ilustración 13; Imagen propia, GRAFCET CINTA_BELT_PALETIZADOR

4.2 – CINTA_ROLLER_ELEVATOR

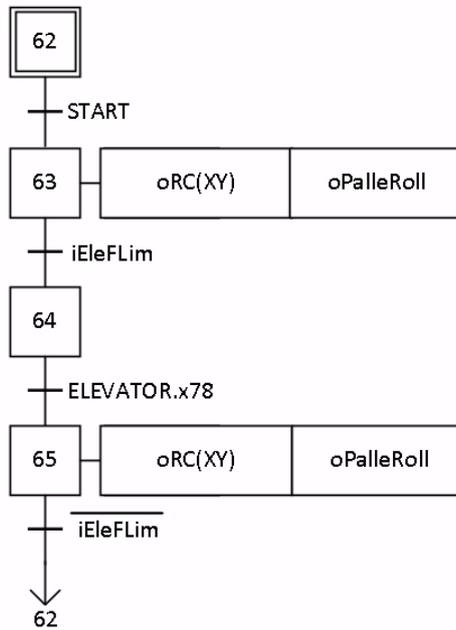


Ilustración 14, Imagen propia, GRAFCET CINTA_ROLLER_ELEVATOR

4.3 – CLAMP

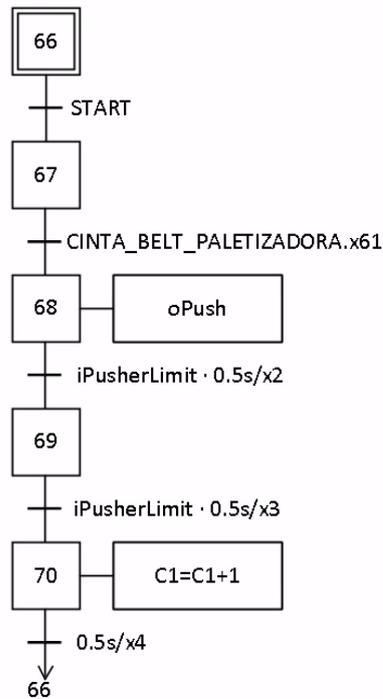


Ilustración 15; Imagen propia, GRAFCET CLAMP

4.4 – ELEVATOR

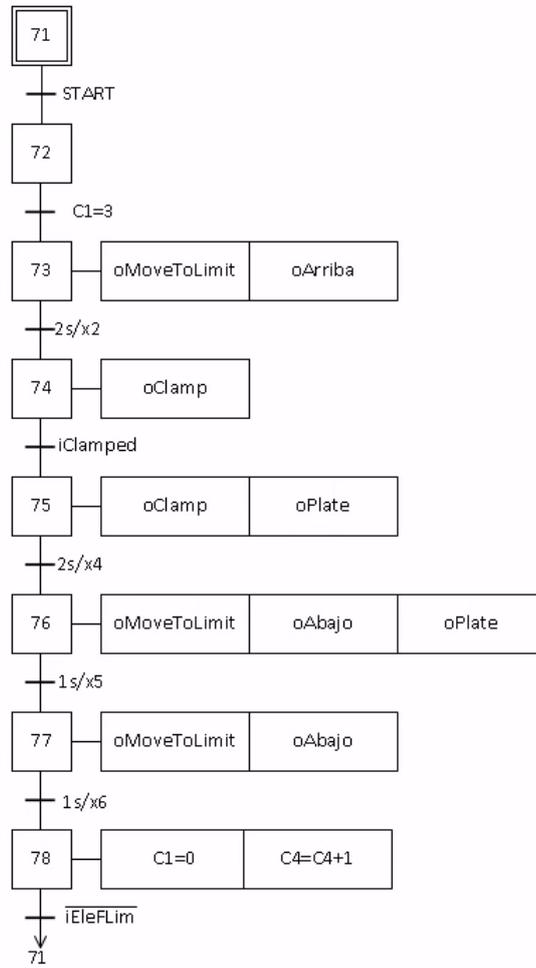


Ilustración 16; Imagen propia, GRAFCET ELEVATOR

4.5 – TALM_SEL_2

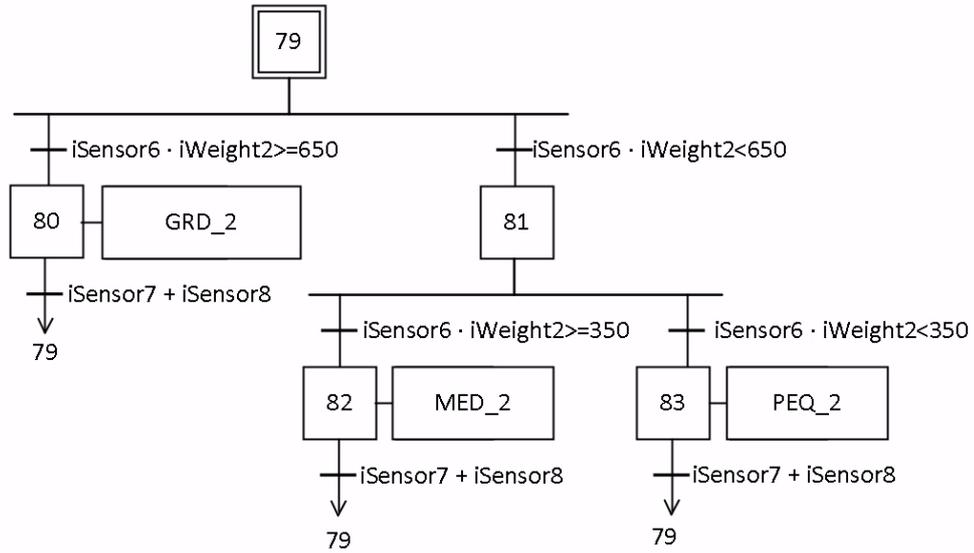


Ilustración 17; Imagen propia, GRAFCET TAL_SEL_2

4.6 – PALETIZADOR_DIRECT

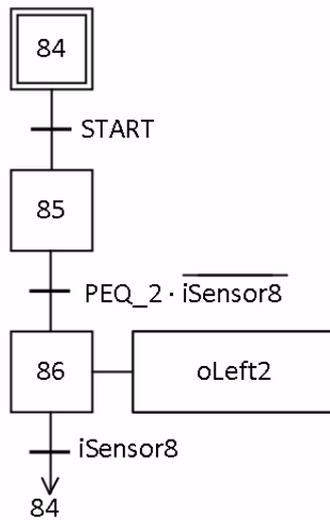


Ilustración 18, Imagen propia, GRAFCET PALETIZADOR_DIRECT

5 – ZONA C: ZONA PICKING ALEATORIO

5.1 – ROLLER_AUTOMATICO

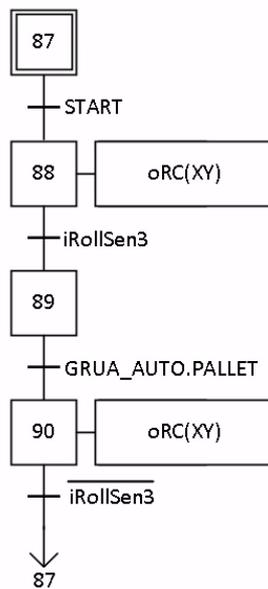


Ilustración 19; Imagen propia, GRAFCET ROLLER_AUTOMATICO

5.2 – TAM_SEL_3

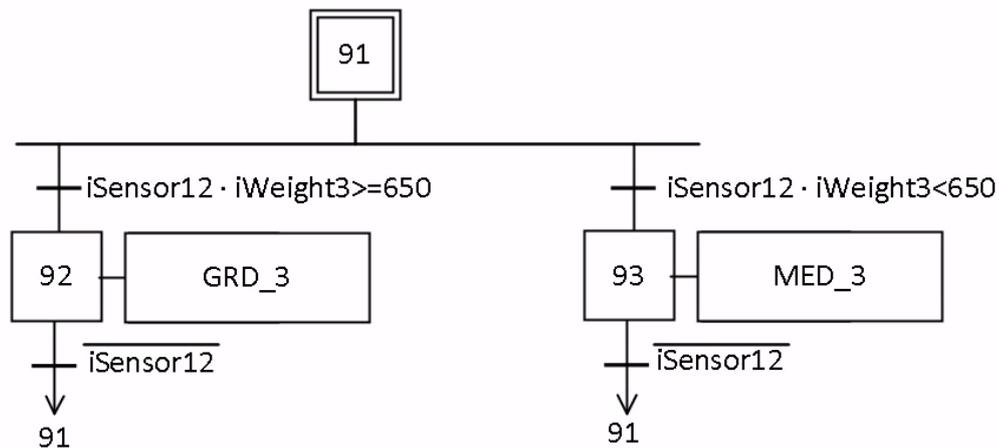


Ilustración 20; Imagen propia, GRAFCET TAM_SEL_3

5.3 – GRUA_AUTO

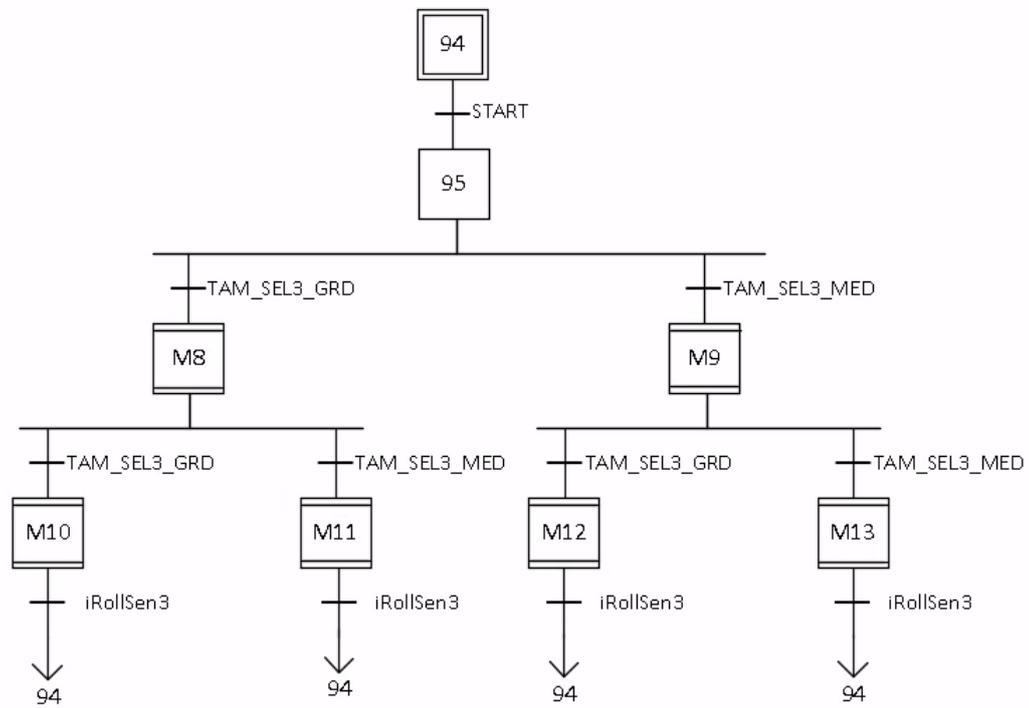


Ilustración 21; Imagen propia, GRAFCET GRUA_AUTO

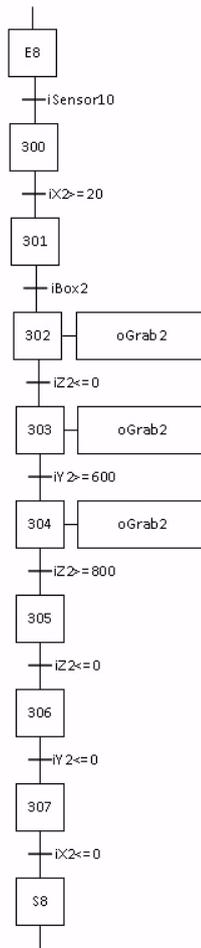


Ilustración 22; Imagen propia, macroetapa M8



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCOLA TÈCNICA
SUPERIOR ENGINYERIA
INDUSTRIAL VALÈNCIA

TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

ANEXO II: VARIABLES

Autor: Ramón Prades Batalla

Tutor: Raúl Simarro Fernández

Curso académico: 2019/2020



ÍNDICE DE CONTENIDOS DEL ANEXO II

1 – OBJETO.....	95
2 – VARIABLES DE ENTRADA.....	95
2.1 – VARIABLES BOOLEANAS.....	95
2.2 – VARIABLES NO BOOLEANAS.....	96
3 – VARIABLES DE SALIDA.....	97
3.1 – VARIABLES BOOLEANAS.....	97
3.2 – VARIABLES NO BOOLEANAS.....	99

ÍNDICE DE TABLAS DEL ANEXO II

Tabla 1; Elaboración propia, Tabla de variables booleanas de entrada	96
Tabla 2; Elaboración propia, Tabla de variables no booleanas de entrada	97
Tabla 3; Elaboración propia, Tabla de variables booleanas de salida	99
Tabla 4; Elaboración propia, Tabla de variables no booleanas de salida	99

1 – OBJETO

En este documento se muestran las variables del proceso que utiliza el PLC que ofrece *CODESYS* para la automatización del sistema. Para facilitar su comprensión se ha añadido una pequeña descripción de funcionamiento de cada variable. Además, se ha dividido en documento en dos partes, por un lado, están las entradas y por el otro lado las salidas.

2 – VARIABLES DE ENTRADA

2.1 – VARIABLES BOOLEANAS

Nombre	Dirección	Tipo de dato	Descripción
iStart	%IX0.0	BOOL	Botón que permite el comienzo del proceso en el cuadro general 1
iSensor1	%IX0.1	BOOL	Sensor situado sobre la primera báscula
iSensor2	%IX0.2	BOOL	Sensor situado a la entrada de la Zona 1
iSensor3	%IX0.3	BOOL	Sensor situado al final de la Zona 1
iSensor4	%IX0.4	BOOL	Sensor situado tras el primer elemento derivador de cajas
iRollSen1	%IX0.5	BOOL	Sensor situado en las cintas de rodillos de la zona 1
iC1	%IX0.6	BOOL	Permite saber si el brazo de la estación de picking 1 ha cambiado el ángulo
iBox1	%IX0.7	BOOL	Detecta si hay una caja bajo la grúa de la estación de picking 1
iSensor5	%IX1.0	BOOL	Sensor situado al final de la Zona 2
iSensor6	%IX1.1	BOOL	Sensor situado sobre la segunda báscula
iSensor7	%IX1.2	BOOL	Sensor situado tras el segundo elemento derivador de cajas
iSensor8	%IX1.3	BOOL	Sensor situado a la entrada de la Zona 2
iClamped	%IX1.4	BOOL	Detecta si las pinzas del paletizador están funcionando
iPlateLimit	%IX1.5	BOOL	Detecta si la plataforma del paletizador se está abriendo o cerrando
iPusherLimit	%IX1.6	BOOL	Detecta las posiciones extremas de la pletina del paletizador
iEleMoving	%IX1.7	BOOL	Detecta si el elevador se está moviendo
iEleBLim	%IX2.0	BOOL	Sensor trasero de la base del paletizador
iEleFLim	%IX2.1	BOOL	Sensor delantero de la base del paletizador

iSensor9	%IX2.2	BOOL	Sensor de apoyo para el picking de la zona 1
iSensor10	%IX2.3	BOOL	Sensor situado al final de la zona 3
iSensor11	%IX2.4	BOOL	Sensor de apoyo para el picking de la zona 3
iRollSen3	%IX2.5	BOOL	Sensor situado en las cintas de rodillos de la zona 3
iC2	%IX2.6	BOOL	Permite saber si el brazo de la estación de picking 2 ha cambiado el ángulo
iSensor12	%IX2.7	BOOL	Sensor situado sobre la tercera báscula
iBox2	%IX3.0	BOOL	Detecta si hay una caja bajo la grua del picking 2
iReset1	%IX3.1	BOOL	Mantenimiento zona 1
iReset2	%IX3.2	BOOL	Parada de la sirena desde la zona 1
iReset3	%IX3.3	BOOL	Mantenimiento zona 2
iReset5	%IX3.4	BOOL	Mantenimiento zona 2
iReset6	%IX3.5	BOOL	Para de la sirena desde la zona 3
iReset7	%IX3.6	BOOL	Parada de la sirena desde fuera del perímetro restringido
iDoor	%IX3.7	BOOL	Sensor que detecta cuando se abre la puerta del perímetro restringido
iStop1	%IX4.0	BOOL	Parada desde cuadro general
iStop2	%IX4.1	BOOL	Parada desde zona 1
iStop3	%IX4.2	BOOL	Parada desde zona 2
iStop4	%IX4.3	BOOL	Parada desde zona 3

Tabla 1; Elaboración propia, Tabla de variables booleanas de entrada

2.2 – VARIABLES NO BOOLEANAS

Nombre	Dirección	Tipo de dato	Descripción
iWeight	%IW3	WORD	Muestra el peso que obtiene la báscula 1
iX1	%IW4	WORD	Muestra la posición en el eje X del brazo de la estación de picking 1
iY1	%IW5	WORD	Muestra la posición en el eje Y del brazo de la estación de picking 1
iZ1	%IW6	WORD	Muestra la posición en el eje Z del brazo de la estación de picking 1

iWeight2	%IW7	WORD	Muestra el peso que obtiene la báscula 2
iWeight3	%IW8	WORD	Muestra el peso que obtiene la báscula 3
iX2	%IW9	WORD	Muestra la posición en el eje X del brazo de la estación de picking 2
iY2	%IW10	WORD	Muestra la posición en el eje Y del brazo de la estación de picking 2
iZ2	%IW11	WORD	Muestra la posición en el eje Z del brazo de la estación de picking 2

Tabla 2; Elaboración propia, Tabla de variables no booleanas de entrada

3 – VARIABLES DE SALIDA

3.1 – VARIABLES BOOLEANAS

Nombre	Dirección	Tipo de dato	Descripción
oBC61	%QX0.0	BOOL	Cinta 6m zona general
oBC21	%QX0.1	BOOL	Cinta 2m zona general
oBC63	%QX0.2	BOOL	Cinta 6m zona 3
oBC22	%QX0.3	BOOL	Cinta 2m zona 1
oBC62	%QX0.4	BOOL	Cinta 2m zona 1
oBC23	%QX0.5	BOOL	Cinta 2m zona 1
oBC41	%QX0.6	BOOL	Cinta 2m zona 1
oCBC1	%QX0.7	BOOL	Cinta curva zona 1
oCBC4	%QX1.0	BOOL	Cinta curva zona 1
oCPeso1	%QX1.1	BOOL	Cinta báscula 1
oCDire1	%QX1.2	BOOL	Cinta recta derivador 1
oCDire2	%QX1.3	BOOL	Cinta recta derivador 2
oRight2	%QX1.4	BOOL	Cinta derecha derivador 1
oEmmitter4	%QX1.5	BOOL	Elemento emisor de cajas/pallets
oRC61	%QX1.6	BOOL	Cinta rodillos 6m zona 1
oRC62	%QX1.7	BOOL	Cinta rodillos 6m zona 1
oC1	%QX2.0	BOOL	Cambiar el ángulo brazo picking zona 1
oGrab1	%QX2.1	BOOL	Activar la succión del brazo picking zona 1
oBC42	%QX2.2	BOOL	Cinta 4m zona 2
oBC25	%QX2.3	BOOL	Cinta 2m zona 2

oCBC2	%QX2.4	BOOL	Cinta curva zona 2
oIC1	%QX2.5	BOOL	Cinta inclinada zona 2
oRC63	%QX2.6	BOOL	Cinta rodillos 6m zona 2
oRC65	%QX2.7	BOOL	Cinta rodillos zona 6m 2
oPeso2	%QX3.0	BOOL	Cinta báscula 2
oLeft2	%QX3.1	BOOL	Cinta izquierda derivador 2
oPush	%QX3.2	BOOL	Movimiento de empuje de la pletina del paletizador
oClamp	%QX3.3	BOOL	Movimiento de pinzado del paletizador
oPaleBelt	%QX3.4	BOOL	Cinta del paletizador
oPaleRoll	%QX3.5	BOOL	Cinta rodillos del paletizador
oPlate	%QX3.6	BOOL	Abrir plataforma paletizador
oArriba	%QX3.7	BOOL	Marcar dirección hacia arriba del elevador
oAbajo	%QX4.0	BOOL	Marcar dirección hacia abajo del elevador
oMoveToLimit	%QX4.1	BOOL	Forzar el movimiento del elevador
oBC43	%QX4.2	BOOL	Cinta 4m zona 3
oRC24	%QX4.3	BOOL	Cintaro rodillos 2m zona 3
oRC64	%QX4.4	BOOL	Cinta rodillos 6m zona 3
oCRC2	%QX4.5	BOOL	Cinta rodillos curva zona 3
oRC23	%QX4.6	BOOL	Cinta rodillos 2m zona 3
oRC22	%QX4.7	BOOL	Cinta rodillos 2m zona 3
oRC21	%QX5.0	BOOL	Cinta rodillos 2m zona 3
oCRC1	%QX5.1	BOOL	Cinta curva rodillos zona 3
oRC25	%QX5.2	BOOL	Cinta rodillos 2m zona 3
oC2	%QX5.3	BOOL	Cambiar el ángulo brazo picking zona 3
oGrab2	%QX5.4	BOOL	Activar la succión del brazo picking zona 3
oPeso3	%QX5.5	BOOL	Cinta báscula 3
oGreen1	%QX5.6	BOOL	LED verde cuadro general
oRed2	%QX5.7	BOOL	LED rojo cuadro general
oRed3	%QX6.0	BOOL	LED rojo cuadro zona 1
oRed5	%QX6.1	BOOL	LED rojo cuadro zona 2
oRed7	%QX6.2	BOOL	LED rojo cuadro zona 3

oYellow4	%QX6.3	BOOL	LED amarillo cuadro zona 1
oYellow6	%QX6.4	BOOL	LED amarillo cuadro zona 2
oYellow8	%QX6.5	BOOL	LED amarillo cuadro zona 3
oWarning	%QX6.6	BOOL	Señal lumínica perímetro restringido
oTR1	%QX6.7	BOOL	Luz roja semáforo zona 1
oTV1	%QX7.0	BOOL	Luz verde semáforo zona 1
oTA1	%QX7.1	BOOL	Luz amarilla semáforo zona 1
oTR2	%QX7.2	BOOL	Luz roja semáforo zona 2
oTV2	%QX7.3	BOOL	Luz verde semáforo zona 2
oTA2	%QX7.4	BOOL	Luz amarilla semáforo zona 2
oTR3	%QX7.5	BOOL	Luz roja semáforo zona 3
oTV3	%QX7.6	BOOL	Luz verde semáforo zona 3
oTA3	%QX7.7	BOOL	Luz amarilla semáforo zona 3

Tabla 3; Elaboración propia, Tabla de variables booleanas de salida

3.2 – VARIABLES NO BOOLEANAS

Nombre	Dirección	Tipo de dato	Descripción
oX1	%QW4	WORD	Posición que se requiere alcanzar en el eje X del picking 1
oY1	%QW5	WORD	Posición que se requiere alcanzar en el eje Y del picking 1
oZ1	%QW6	WORD	Posición que se requiere alcanzar en el eje Z del picking 1
oX2	%QW7	WORD	Posición que se requiere alcanzar en el eje X del picking 2
oY2	%QW8	WORD	Posición que se requiere alcanzar en el eje Y del picking 2
oZ2	%QW9	WORD	Posición que se requiere alcanzar en el eje Z del picking 2

Tabla 4; Elaboración propia, Tabla de variables no booleanas de salida



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCOLA TÈCNICA
SUPERIOR ENGINYERIA
INDUSTRIAL VALÈNCIA

TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

ANEXO III: MANUAL DE USUARIO

Autor: Ramón Prades Batalla

Tutor: Raúl Simarro Fernández

Curso académico: 2019/2020





ÍNDICE DE CONTENIDOS DEL ANEXO III

1 – OBJETIVO.....	103
2 – PUESTA EN FUNCIONAMIENTO.....	103
3 – ACCESO AL SERVIDOR WEB.....	105
4 – INSTRUCCIONES DE USO.....	107

ÍNDICE DE FIGURAS DEL ANEXO III

Ilustración 1; Elaboración propia, conexión con el PLC para la puesta en funcionamiento	103
Ilustración 2; Elaboración propia, icono para el encendido del PLC virtual.....	104
Ilustración 3; Elaboración propia, icono para la compilación del código	104
Ilustración 4; Elaboración propia, icono para la conexión con el autómeta.....	104
Ilustración 5; Elaboración propia, icono para ejecutar el comando “run”	104
Ilustración 6; Elaboración propia, ruta de acceso a la consola de comandos	105
Ilustración 7; Elaboración propia, dirección IP mostrada en la consola de comandos	106
Ilustración 8; Elaboración propia, Esquema pantalla operador	107

1 – OBJETIVO

El objetivo de este anexo es acabar de complementar la información ya otorgada acerca de la pantalla de operador explicando detalladamente la correcta puesta en funcionamiento del programa, el acceso al servidor Web y un breve manual para el correcto uso de la pantalla propia de este proyecto.

2 – PUESTA EN FUNCIONAMIENTO

Toda la información que se va a exponer es este apartado ha sido obtenida del manual que el *FACTORY IO* aporta y aquí solo se resumirán las que afecten directamente a la puesta en funcionamiento.

El primer paso que se deberá realizar es el de iniciar la comunicación con el PLC. Para ello se deberá realizar un doble click sobre el elemento “Device”, a continuación se deberá seleccionar “Examinar Redes” y finalmente haremos click sobre el PLC que tengamos disponible y posteriormente a “Aceptar”.

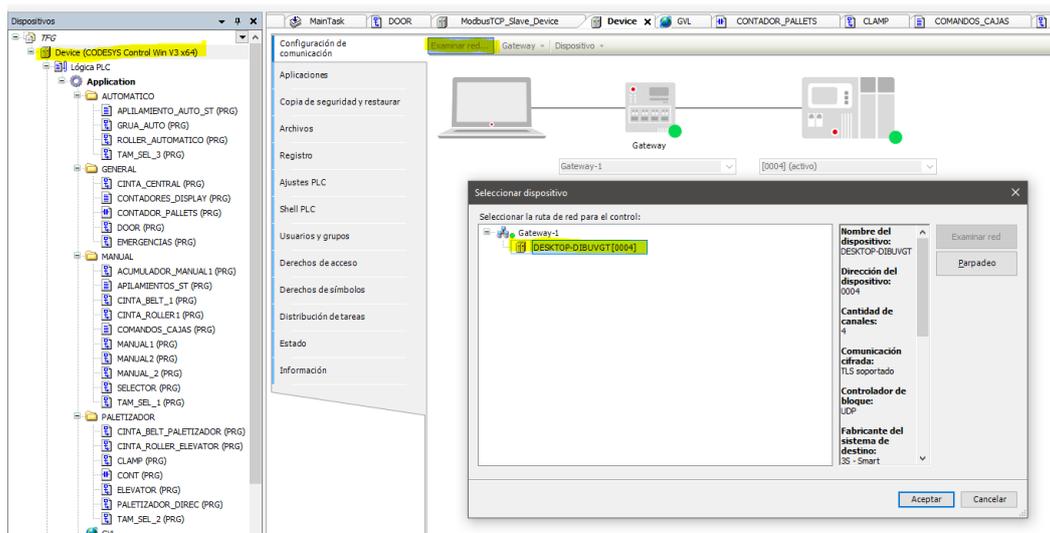


Ilustración 1; Elaboración propia, conexión con el PLC para la puesta en funcionamiento

Si se está empleando el PLC que trae incorporado el propio CODESYS será necesario comprobar que esté esta activo. El icono para conectarlo se encuentra en la barra de tareas, en la pestaña de iconos ocultos.

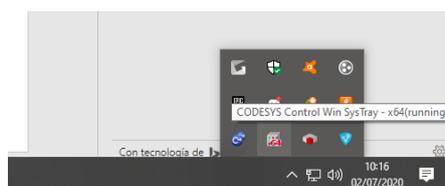


Ilustración 2; Elaboración propia, icono para el encendido del PLC virtual

Una vez se haya establecido la comunicación con el PLC, será necesario en caso de ser la primera vez o en caso de haber modificado el programa, compilar el código en búsqueda de errores que puedan alterar el correcto funcionamiento del proceso. Para ello o bien se clickará sobre el icono para compilar situado en la barra de herramientas de CODESYS, o bien se pulsará el atajo para realizar esta tarea: F11.



Ilustración 3; Elaboración propia, icono para la compilación del código

Una vez el programa haya sido compilado exitosamente, se procederá a iniciar la conexión con el autómatas. Para ello o bien se clicka sobre el correspondiente icono, o se emplea el atajo para realizar esta función: ALT+F8.



Ilustración 4; Elaboración propia, icono para la conexión con el autómatas

Si bien no han surgido fallos en este último proceso, el último paso será poner el programa a funcionar, bien mediante su icono o bien mediante su atajo: F5.



Ilustración 5; Elaboración propia, icono para ejecutar el comando "run"

Una vez realizados todos los pasos y si estos no han generado errores, el programa debería estar funcionando de manera satisfactoria, así como Interfaz de Usuario. Cabe destacar que el correcto funcionamiento del proyecto también va a depender de la correcta asignación de todas las variables compartidas con el *FACTORY IO*, de manera que pese a no existir ningún error de programación, puede haber otro tipo de problemas que impidan que la planta se comporte como se había planeado.

3 – ACCESO AL SERVIDOR WEB

Como se ha mencionado en varias ocasiones, *CODESYS* permite que la monitorización de la planta efectuada a través de la HMI se realice en línea mediante una aplicación Web, accesible desde cualquier dispositivo electrónico conectado en la misma subred en la que se aloje el sistema Web embebido en el PLC.

Esta característica es realmente interesante debido a que permite el control remoto del proceso desde nuestro propio teléfono móvil. Además el acceso a esta aplicación es muy sencillo.

Para poder acceder al sitio web donde se encuentra alojada la pantalla de operador deberemos conocer su URL. Esta URL será la combinación de la IP desde la que se realice la conexión, del puerto en el que se aloja, y del nombre que se le ha dado a esta aplicación. Por tanto el primer paso que se deberá realizar es averiguar la IP del dispositivo desde el que se quiera realizar la conexión. En el caso de un ordenador, la obtención de este número es realmente sencilla; para ello se escribirá en el buscador de la barra de tareas las letras “cmd” para acceder a la consola de comandos.

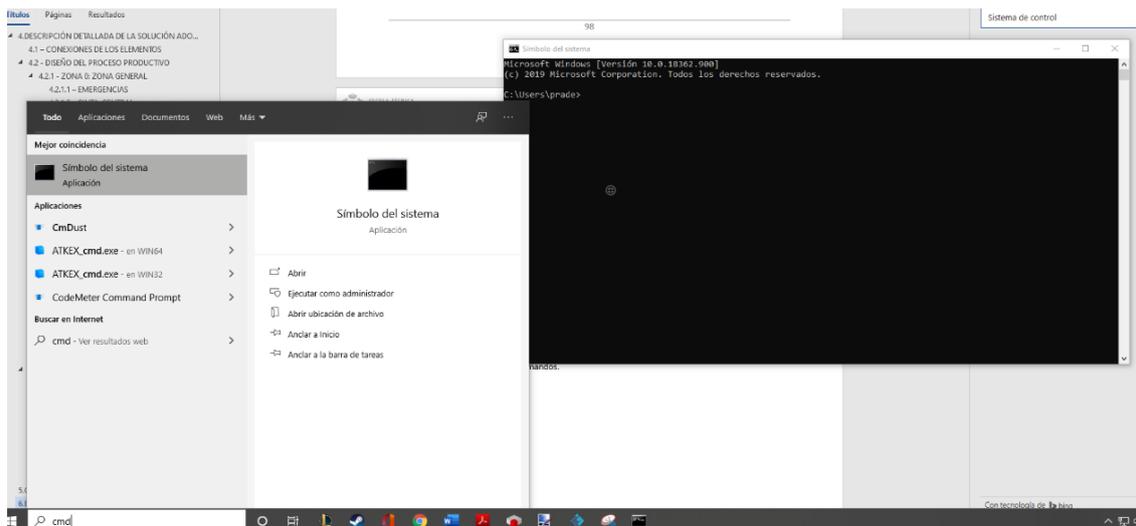


Ilustración 6; Elaboración propia, ruta de acceso a la consola de comandos

Una vez se haya abierto la consola, se deberá escribir la palabra “ipconfig”, y la IP que se requerirá para acceder a la aplicación Web es la siguiente:

```
Selecionar Símbolo del sistema

adaptador de Ethernet Radmin VPN:

  Sufijo DNS específico para la conexión. . . :
  Dirección IPv6 . . . . . : fdfd::1a3a:d50c
  Vínculo: dirección IPv6 local. . . : fe80::c10c:6491:7751:b292%12
  Dirección IPv4. . . . . : 26.58.213.12
  Máscara de subred . . . . . : 255.0.0.0
  Puerta de enlace predeterminada . . . . . : 26.0.0.1

adaptador PPP VPN a la UPV:

  Sufijo DNS específico para la conexión. . . :
  Dirección IPv4. . . . . : 158.42.237.175
  Máscara de subred . . . . . : 255.255.255.255
  Puerta de enlace predeterminada . . . . . : 0.0.0.0

adaptador de Ethernet Conexión de red Bluetooth 2:

  Estado de los medios. . . . . : medios desconectados
  Sufijo DNS específico para la conexión. . . :

adaptador de túnel Teredo Tunneling Pseudo-Interface:
```

Ilustración 7; Elaboración propia, dirección IP mostrada en la consola de comandos

Una vez obtenido este número bastará con acceder a algún navegador de Internet y escribir la siguiente URL: IP:PUERTO/webvisu.htm

En cuanto a la IP se colocará la obtenida en el paso anterior, y en cuanto al puerto este es común para todos y es 8080. Por tanto la dirección quedaría:

<http://158.42.237.175:8080/webvisu.htm>

4 – INSTRUCCIONES DE USO

El funcionamiento de la pantalla de operador es realmente sencillo e intuitivo, y se podría dividir en cuatro funciones distintas.

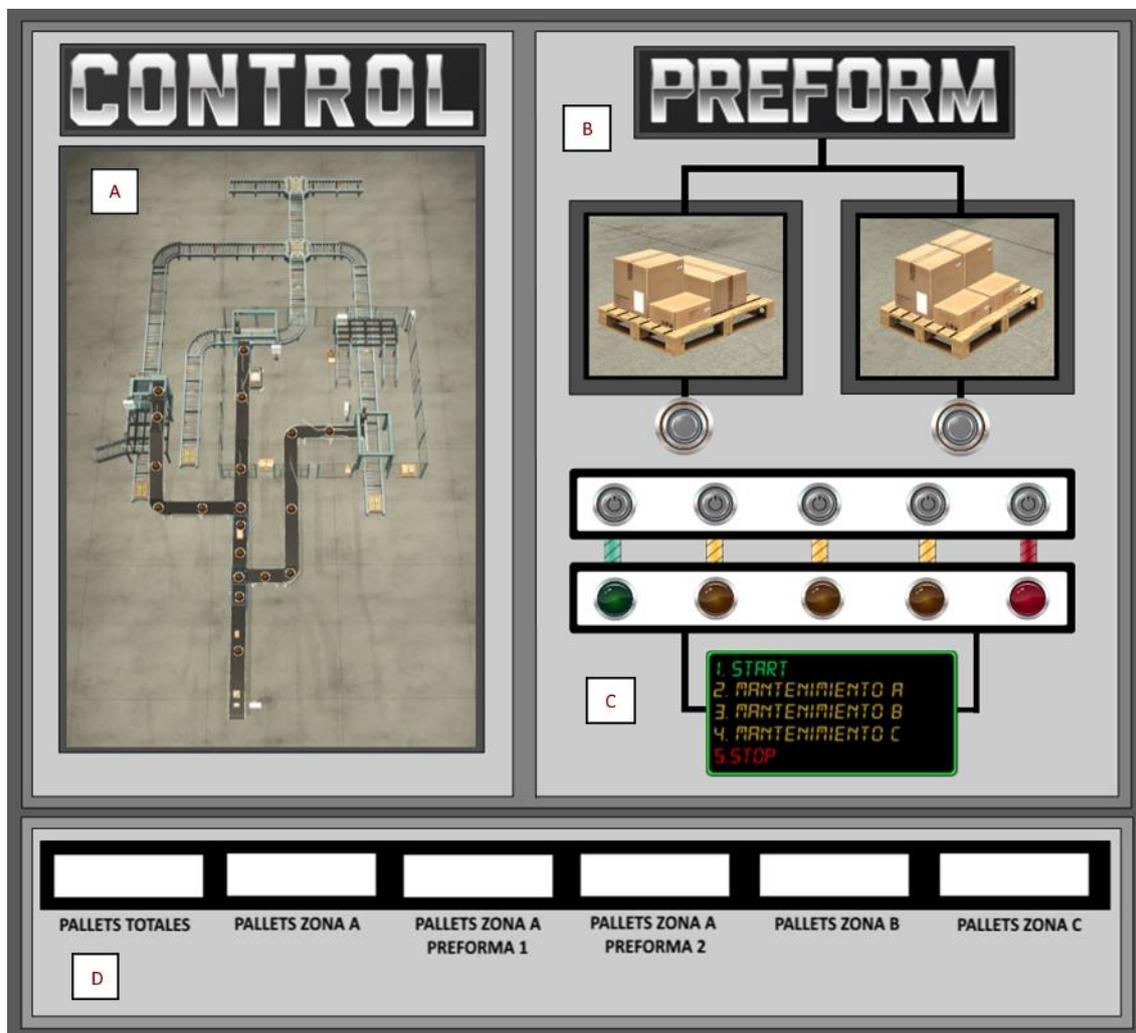


Ilustración 8; Elaboración propia, Esquema pantalla operador

FUNCIÓN A:

Permite observar el estado de todas las cintas que conforman el proceso automatizado, de manera que se pueda comprobar en todo momento el correcto funcionamiento de todas y cada una de ellas. También permite verificar si las acciones de parada y mantenimiento detienen las cintas deseadas.

FUNCIÓN B:

Permite elegir el tipo de preforma que se conformará en la estación de picking de la zona A. Esta diseñado para permitir el cambio entre ellas simplemente presionando el botón de la preforma que no se encuentra en uso.

FUNCIÓN C:

Es el mando de control asignado al manual de operador. Mediante la pulsación de los distintos botones se puede acceder a los distintos estados del proceso (funcionamiento, mantenimiento y parada) sin necesidad de presionar los botones físicos situados en los cuadros de mandos que se encuentran en la planta, y su función es la misma. Los LEDs situados justo debajo permiten conocer si el estado está activado, tanto si se ha activado a través de la pantalla de operador, como si se ha realizado mediante su variable física en planta.

La leyenda situada justo debajo de las lámparas contiene la información necesaria para saber que función desempeña cada color.

FUNCIÓN D:

Se trata de una serie de elementos que muestran por pantalla la cantidad de pallets generados en cada subestación. Además contiene la información segmentada de los pallets producidos por cada preforma así como los totales, obtenidos a partir de la suma de todas las zonas.



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



ESCOLA TÈCNICA
SUPERIOR ENGINYERIA
INDUSTRIAL VALÈNCIA

TRABAJO FINAL DE GRADO EN INGENIERÍA EN TECNOLOGÍAS INDUSTRIALES

DOCUMENTO 2: PRESUPUESTOS

Autor: Ramón Prades Batalla

Tutor: Raúl Simarro Fernández

Curso académico: 2019/2020





ÍNDICE DE CONTENIDOS DE LOS PRESUPUESTOS

1.INTRODUCCIÓN.....	112
2.CUADRO DE PRECIOS ELEMENTALES	113
2.1 – CUADRO N1: MANO DE OBRA.....	113
2.2 – CUADRO N2: MATERIALES Y MAQUINARIA	113
3.CUADRO DE PRECIOS UNITARIOS.....	114
4.CUADRO DE PRECIOS DESCOMPUESTOS.....	115
5.PRESUPUESTO BASE DE LICITACIÓN.....	117

ÍNDICE DE TABLAS DE LOS PRESUPUESTOS

Tabla 1; Elaboración propia, distribución de las horas empleadas en la realización del proyecto	112
Tabla 2; Elaboración propia, Tabla con las especificaciones de la mano de obra.....	113
Tabla 3; Elaboración propia, Tabla precio de licencia de <i>FACTORY IO</i>	113
Tabla 4; Elaboración propia, Tabla con las especificaciones de material y maquinaria	114
Tabla 5; Elaboración propia, Tabla con el precio de cada Ud de obra	114
Tabla 6; Elaboración propia, Tabla precios descompuestos de la primera ud. de obra	115
Tabla 7; Elaboración propia, Tabla precios descompuestos de la segunda ud. de obra	115
Tabla 8; Elaboración propia, Tabla precios descompuestos de la tercera ud. de obra	116
Tabla 9; Elaboración propia, Tabla precios descompuestos de la cuarta ud. de obra	116
Tabla 10; Elaboración propia, Tabla presupuestos parciales de cada una de las uds. de obra.....	117
Tabla 11; Elaboración propia, Tabla presupuesto total del proyecto	117

I. INTRODUCCIÓN

En este presente documento se desarrollará el presupuesto del proyecto, tanto de los costes de diseño, implementación de los sistemas de control y SCADA. Para ello se hará un análisis económico de todo lo que ha intervenido en el mismo.

Este presupuesto se realizará en vista a su implementación real y partiendo de la base que una empresa dispusiera de todos los elementos físicos empleados a lo largo del proyecto, por tanto no se tendrá en cuenta el precio de la maquinaria industrial utilizada pero sí de la utilización del programa empleado para simular el proceso: *FACTORY IO*.

Es conveniente recordar que este proyecto se ha automatizado sobre el PLC virtual que ofrece el propio *CODESYS*, y que por tanto no supondrá ningún coste; no obstante, si se quisiera automatizar sobre un PLC real, opción completamente viable como se ha indicado en varias ocasiones, se debería obtener los drivers pertenecientes a dicho PLC y estos sí cuestan dinero.

Finalmente y previo a la obtención del presupuesto se indicará las distintas actividades en las que se ha descompuesto el trabajo (que posteriormente serán las distintas Ud. De Obra) y el total de horas dedicadas a cada una de ellas, así como la duración total del proyecto, que no es más que la suma de estas.

ACTIVIDAD	DURACIÓN (h)
Diseño del proceso	40
Implementación y validación del proceso	120
Implementación de la interfaz de usuario	30
Elaboración de la redacción del proyecto	60
Tiempo total del proyecto	250

Tabla 1; Elaboración propia, distribución de las horas empleadas en la realización del proyecto

2. CUADRO DE PRECIOS ELEMENTALES

2.1 – CUADRO N1: MANO DE OBRA

Código	Unidad	Descripción	Salario (€/h)
MO.GITI	h	Ingeniero en Tecnologías Industriales	18

Tabla 2; Elaboración propia, Tabla con las especificaciones de la mano de obra

2.2 – CUADRO N2: MATERIALES Y MAQUINARIA

En este apartado se realizará la distinción entre softwares y hardware:

Hardware:

El proyecto al completo se ha desarrollado con un ordenador de sobremesa, el cual ha sido empleado 8 horas diarias (jornada laboral) durante 225 días laborales que tiene un año. Se estima que la vida útil de este ordenador será de entorno a 5 años:

$$Amortización MT.OS = \frac{\text{base de amortización (€)}}{\text{vida útil (h)}} = \frac{1250 - 0}{5 \text{ años} \cdot \frac{1800h}{1 \text{ año}}} = 0.1389 \text{ €/h}$$

Softwares:

En este proyecto se han empleado dos softwares: el CODESYS y el FACTORY IO. No obstante la licencia para adquirir CODESYS es completamente gratuita por lo que no se añadirá en el cuadro de precios. Por otro lado FACTORY IO sí dispone de una licencia de pago, por lo que se tendrá en cuenta una jornada laboral de 8h durante 225 días laborales al año:

Nombre del software	Precio de licencia (€)	Duración de licencia (años)
FACTORY IO	253	1

Tabla 3; Elaboración propia, Tabla precio de licencia de FACTORY IO

$$Amortización MT.FIO = \frac{\text{base de amortización (€)}}{\text{vida útil (h)}} = \frac{253}{1 \text{ año} \cdot \frac{1800h}{1 \text{ año}}} = 0.1406 \text{ €/h}$$

Código	Unidad	Descripción	Salario (€/h)
<i>MT.OS</i>	h	Ordenador de sobremesa	0.1389
<i>MT.FIO</i>	h	Software <i>FACTORY IO</i>	0.1406

Tabla 4; Elaboración propia, Tabla con las especificaciones de material y maquinaria

3.CUADRO DE PRECIOS UNITARIOS

Código	Unidad	Descripción	Precio (€/Ud)
<i>UD.01</i>	Ud	Diseño del proceso	723.47
<i>UD.02</i>	Ud	Implementación y validación del proceso	2193.54
<i>UD.03</i>	Ud	Implementación de la interfaz de usuario	546.28
<i>UD.04</i>	Ud	Elaboración de la redacción del proyecto	1088.33

Tabla 5; Elaboración propia, Tabla con el precio de cada Ud de obra

4. CUADRO DE PRECIOS DESCOMPUESTOS

Código	Cantidad	Descripción	Rendimiento	Precio	Importe (€)
<i>Ud.01</i>	Ud	Diseño del proceso		723.47	
<i>MO.GITI</i>	h	Ingeniero en Tecnologías Industriales	40	18	720
<i>MT.OS</i>	h	Ordenador de Sobremesa	25	0.1389	3.4725

Tabla 6; Elaboración propia, Tabla precios descompuestos de la primera ud. de obra

Código	Cantidad	Descripción	Rendimiento	Precio	Importe (€)
<i>Ud.02</i>	Ud	Implementación y validación del proceso		2193.54	
<i>MO.GITI</i>	h	Ingeniero en Tecnologías Industriales	120	18	2160
<i>MT.OS</i>	h	Ordenador de Sobremesa	120	0.1389	16.668
<i>MT.FIO</i>	h	Software <i>FACTORY IO</i>	120	0.1406	16.872

Tabla 7; Elaboración propia, Tabla precios descompuestos de la segunda ud. de obra

Código	Cantidad	Descripción	Rendimiento	Precio	Importe (€)
<i>Ud.03</i>	Ud	Implementación de la Interfaz de usuario		546.28	
<i>MO.GITI</i>	h	Ingeniero en Tecnologías Industriales	30	18	540
<i>MT.OS</i>	h	Ordenador de Sobremesa	30	0.1389	4.167
<i>MT.FIO</i>	h	Software <i>FACTORY IO</i>	15	0.1406	2.109

Tabla 8; Elaboración propia, Tabla precios descompuestos de la tercera ud. de obra

Código	Cantidad	Descripción	Rendimiento	Precio	Importe (€)
<i>Ud.04</i>	Ud	Elaboración de la redacción del proyecto		1088.33	
<i>MO.GITI</i>	h	Ingeniero en Tecnologías Industriales	60	18	1080
<i>MT.OS</i>	h	Ordenador de Sobremesa	60	0.1389	8.334

Tabla 9; Elaboración propia, Tabla precios descompuestos de la cuarta ud. de obra

5.PRESUPUESTO BASE DE LICITACIÓN

Código	Unidad	Descripción	Rendimiento	Precio	Presupuestos parciales (€)
UD.01	Ud	Diseño del proceso	1	723.47	723,47
UD.02	Ud	Implementación y validación del proceso	1	2193.54	2193,54
UD.03	Ud	Implementación de la Interfaz de usuario	1	546.28	546,28
UD.04	Ud	Elaboración de la redacción del proyecto	1	1088.33	1088,33

Tabla 10; Elaboración propia, Tabla presupuestos parciales de cada una de las uds. de obra

<i>Presupuesto total de ejecución material</i>	4551,62
<i>Gastos generales (13%)</i>	591,7106
<i>Beneficio industrial (6%)</i>	273,0972
<i>Presupuesto total de ejecución por contrata</i>	5416,4378
<i>I.V.A. (21%)</i>	1137,4493
<i>Presupuesto Base de Licitación</i>	6553,89 €

Tabla 11; Elaboración propia, Tabla presupuesto total del proyecto

El presupuesto final asciende a:

SEIS MIL QUINIENTOS CINCUENTA Y TRES EUROS CON OCHENTA Y NUEVE CÉNTIMOS.