

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

INGENIERÍA TÉCNICA INDUSTRIAL
ELECTRICIDAD

DEPARTAMENTO DE INGENIERÍA
ELÉCTRICA

AUTOR: DAVID SÁNCHEZ SANCHIDRIÁN

TUTORA DEL PROYECTO: MÓNICA CHINCHILLA

Septiembre 2008

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

ÍNDICE

ÍNDICE

ÍNDICE

Capítulo 1. Introducción y Objetivos.	1
1.1. Definición del proyecto	1
1.2. Origen del proyecto	3
1.3. Objetivos	4
Capítulo 2. Los Automatas Programables.	5
2.1. La Automatización	5
2.2. Definición y Generalidades	9
2.3. Origen e Historia de los Automatas	13
2.4. Estructura de un Automata Programable	14
2.5. Funcionamiento del Automata	25
2.6. Equipos y lenguajes de programación	30
2.7. Configuración, instalación y puesta a punto	38
Capítulo 3. Descripción del sistema	43
3.1. Descripción de la instalación	43
3.2. Marcha/parada del sistema	48
3.3. Descripción de los modos de operación	50
3.4. Descripción del funcionamiento	50
3.5. Sistema de Alarmas y Avisos	52
3.6. Sistema de Señalización	54
3.7. Operaciones de Mantenimiento	54
Capítulo 4. Diseño del sistema	56
4.1. Introducción	56
4.2. Selección del Automata	57
4.3. Selección del Panel de Operador	64
4.4. Comunicaciones	66
Capítulo 5. Programa del autómata	69
5.1. El Software estándar STEP 7	69
5.2. Configuración del Hardware	73
5.3. Estructura del Programa	75
5.4. Entradas y Salidas de la instalación	109
5.5. Gestión de Avisos y Alarmas	111
Capítulo 6. Programa del panel de control	113
6.1. El Software Protocol.	113
6.2. Estructura del programa	114
Capítulo 7. Proyecto eléctrico	121
7.1. Introducción	121
7.2. Cálculo de la potencia de los motores	123
7.3. Cálculo de secciones de los conductores	133
7.4. Selección de los dispositivos de protección	134
7.5. Estructura del Proyecto Eléctrico	139



7.6. Layout Eléctrico	141
7.6. Fotos de la instalación	144
Capítulo 8. Presupuesto	153
8.1. Oferta	153
Bibliografía	157
Contenido del CD.	158
<i>Presentación.</i>	158
<i>Proyecto.</i>	158
<i>Anexos.</i>	158
Anexo 1. Características técnicas	158
Anexo 2. Esquemas Eléctricos	158
Anexo 3. Programa Autómata	158
Anexo 4. Programa Panel de operación	158
Anexo 5. Manual de Usuario	159
Software.	159

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CAPÍTULO 1

**INTRODUCCIÓN
Y
OBJETIVOS**

1. INTRODUCCIÓN Y OBJETIVOS.

1.1. Definición del proyecto.

Este proyecto consiste en la automatización de una línea de montaje de ordenadores en una empresa del sector informático situada en Portugal.

La instalación está compuesta por dos líneas paralelas de montaje de idénticas características que funcionarán de forma independiente.

Los ordenadores circulan por la instalación sobre unas bandejas de polietileno que tienen un espesor de 20mm y unas dimensiones de 850x850 mm.

Cada línea tiene dos niveles, el nivel superior es un camino de cadenas motorizado por el que circulan las bandejas con carga y se realizan las operaciones necesarias en los ordenadores, y el nivel inferior está compuesto por unos rodillos de gravedad por el que circulan las bandejas vacías.

Existen tres puestos de trabajo por línea (1, 2 y 3). En cada puesto de trabajo existe: una botonera con un pulsador con luz que tiene la función de indicar el fin del trabajo al autómeta, y un accionador neumático para subir una cruz de giro con el fin de poder girar la bandeja que transporta el ordenador.

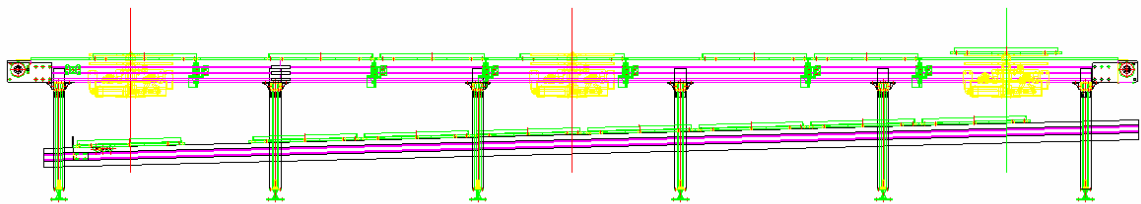
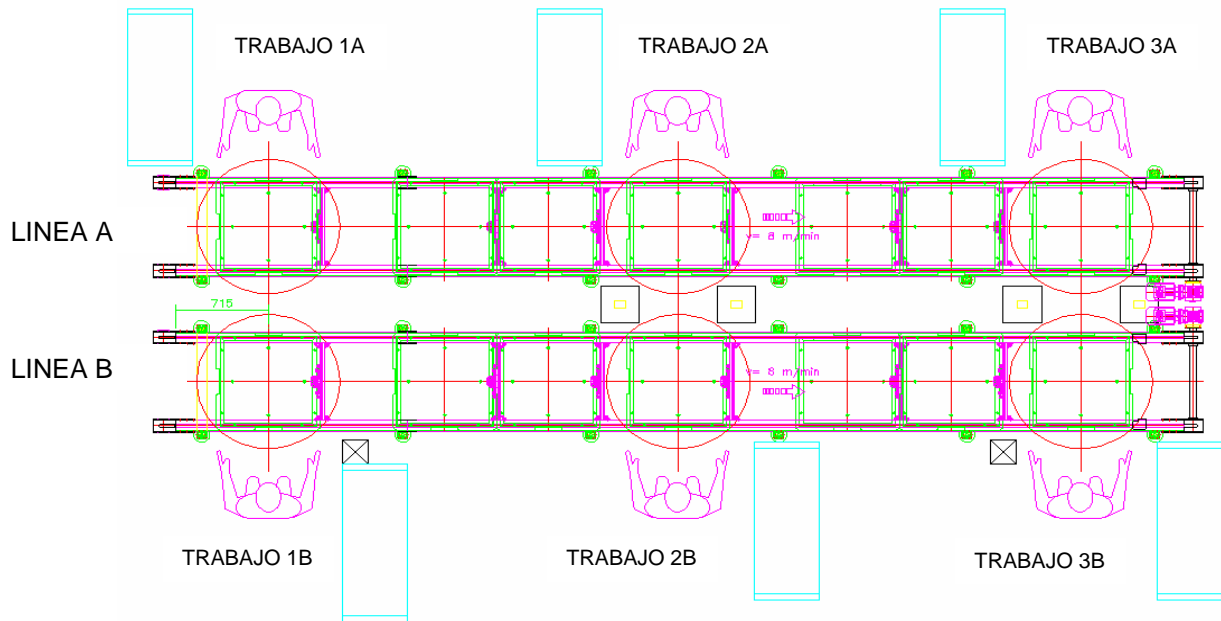
En cada puesto el operador dispone de una estantería de tres niveles y de los elementos necesarios para comprobar el correcto funcionamiento del ordenador (fuente de alimentación, ratón, teclado, pantalla, etc.).

Con el fin de controlar el paso de bandejas se han dispuesto dos topes neumáticos simples en la salida de los puestos 1 y 2, y dos topes neumáticos dosificadores antes de los puestos 2 y 3.

Al tratarse de dos líneas de montaje que funcionan de forma independiente, el sistema estará dividido en dos zonas funcionales:

1. Línea de transporte y montaje A:
Trabajo 1A, Trabajo 2A, Trabajo 3A.
2. Línea de transporte y montaje B:
Trabajo 1B, Trabajo 2B y Trabajo 3B.

El layout de la instalación es el siguiente:





1.2. Origen del proyecto.

En un mundo industrial en plena evolución en que la competitividad es el objetivo esencial, es necesario que una empresa tenga una elevada productividad, una alta calidad en sus productos y una buena organización de la producción para responder a la demanda del mercado, por supuesto al menor coste posible. Estas necesidades de la empresa pueden ser resueltas en parte utilizando sistemas automatizados, ya que el principal objetivo de la automatización es producir mejor a coste más bajo.

En este caso, la empresa en cuestión se dedica al montaje y reparación de ordenadores, venta de componentes informáticos, mantenimiento de equipos, etc.

El proyecto parte de la necesidad que la empresa tiene de disminuir el número de personas que se dedican al montaje de ordenadores para poder recolocar al personal en otros puestos de trabajo como reparación de PCs, mantenimiento de equipos informáticos, etc.

El proyecto completo lo realizará una empresa mecánica, que por su parte subcontrata el presente proyecto que consiste en realizar el sistema de control y gestión mediante el autómeta y el panel de control.

1.3. Objetivos.

Partiendo de que el proyecto es de una instalación real, antes de fijar los objetivos de proyecto es conveniente saber de que partimos, es decir, cual es la información que nos ha proporcionado el cliente sobre dicho proyecto.

En nuestro caso el cliente nos ha suministrado el proyecto eléctrico, un layout de la instalación y un pequeño descriptivo de cómo tiene que funcionar dicha instalación.

Por tanto, el primer objetivo es **realizar un pre-estudio de la documentación entregada** por el cliente con el fin de hacer un presupuesto y ofertar la instalación, ya que es el primer paso para que el cliente nos haga el pedido de proyecto.

Los **objetivos** de este proyecto son:

- Minimizar el tiempo de puesta en marcha de la instalación fuera de la oficina, puesto que cuanto menos tiempo más beneficios.
- Gestionar de forma eficiente y automática los componentes que forman la instalación.
- Indicar el estado de la instalación mediante las señales visuales y el panel de operador.
- Gestión de errores de la instalación (fallos de hardware, manipulación de operario, etc.).
- Gestión de seguridad (paradas de emergencia, etc.).
- Control manual de la instalación para operaciones de mantenimiento.
- Control de ahorro de energía.

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CAPÍTULO 2

**LOS AUTÓMATAS
PROGRAMABLES**

2. LOS AUTÓMATAS PROGRAMABLES.

2.1. LA AUTOMATIZACIÓN.

La automatización es un sistema donde se transfieren tareas de producción, realizadas habitualmente por operadores humanos a un conjunto de elementos tecnológicos.

Un sistema automatizado consta de dos partes principales:

- **Parte Operativa.** Es la parte que actúa directamente sobre la máquina. Son los elementos que hacen que la máquina se mueva y realice la operación deseada. Los elementos que forman la parte operativa son los accionadores de las máquinas como motores, cilindros, etc. y los captadores como detectores inductivos, finales de carrera, etc.
- **Parte de Mando.** Suele ser un autómata programable (tecnología programada), aunque hasta hace bien poco se utilizaban relés electromagnéticos, tarjetas electrónicas o módulos lógicos neumáticos (tecnología cableada). En un sistema de fabricación automatizado el autómata programable está en el centro del sistema. Éste debe ser capaz de comunicarse con todos los constituyentes del sistema automatizado.

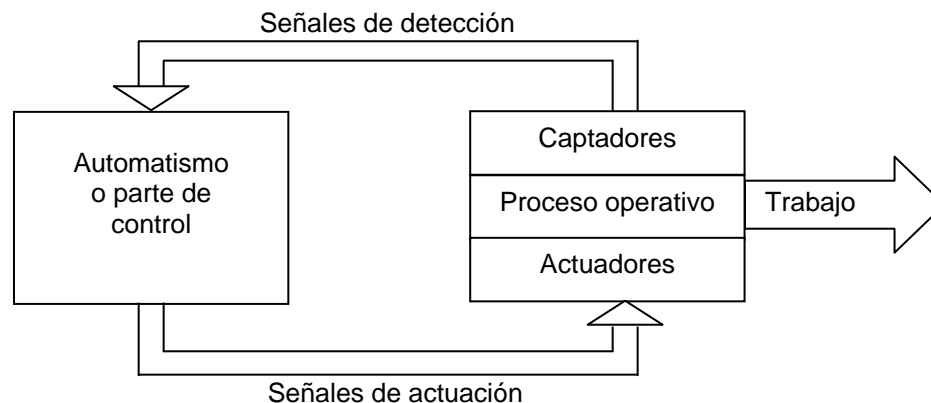


Figura 1.

Parte Operativa.

Es la que opera sobre la máquina y el producto.

➤ Detectores y Captadores.

Al igual que las personas necesitan de los sentidos para percibir lo que ocurre en su entorno, los sistemas automatizados precisan de los transductores para adquirir información de:

- La variación de ciertas magnitudes físicas del sistema (presión, temperatura, etc.).
- El estado físico de sus componentes (posiciones de cilindros, etc.).

Los dispositivos encargados de convertir las magnitudes físicas en magnitudes eléctricas se denominan transductores.

Los transductores se pueden clasificar en función del tipo de señal que transmiten en:

- Transductores todo o nada: Suministran una señal binaria claramente diferenciada. Los finales de carrera son transductores de este tipo.
- Transductores numéricos: Transmiten valores numéricos en forma de combinaciones binarias. Un ejemplo de este tipo de transductores son los encoders.
- Transductores analógicos: Suministran una señal continua que es fiel reflejo de la variación de la magnitud física medida.

Algunos de los transductores más utilizados son: Final de carrera, fotocélulas, pulsadores, encoders, etc.

➤ **Accionadores y Preaccionadores.**

El accionador es el elemento final de control que, en respuesta a la señal de mando que recibe, actúa sobre la variable o elemento final del proceso.

Un accionador transforma la energía de salida del automatismo en otra útil para el entorno industrial de trabajo.

Los accionadores más utilizados en la industria son: Cilindros, motores de corriente alterna, motores de corriente continua, etc.

Los accionadores son gobernados por la parte de mando, sin embargo, pueden estar bajo el control directo de la misma o bien requerir algún preaccionamiento para amplificar la señal de mando. Esta preamplificación se traduce en establecer o interrumpir la circulación de energía desde la fuente al accionador.

Los preaccionadores disponen de parte de mando o de control que se encarga de conmutar la conexión eléctrica, hidráulica o neumática entre los cables o conductores del circuito de potencia.

Los accionadores pueden ser clasificados en:

- Eléctricos: Éstos utilizan directamente la energía eléctrica distribuida en las máquinas y toman diferentes formas: motores de velocidad constante o variable, válvulas eléctricas de flujo, resistencias de calentamiento, electroimanes, cabezas de soldadura por resistencia, por ultrasonidos, cabeza de corte láser, etc.

Los preaccionadores asociados a estos accionadores eléctricos son principalmente los contactores y variadores de velocidad, equipados con las seguridades necesarias.

- **Neumáticos:** Utilizan directamente el aire comprimido distribuido en las máquinas. Sencillos de empleo y presentados bajo formas muy diversas, los cilindros neumáticos se utilizan para numerosos movimientos: transferencias, aprietes, marcados, mantenimientos, ensamblajes, moldeados, eyecciones, etc. Los distribuidores son preaccionadores que están asociados a los mismos. Reciben una señal neumática o una señal eléctrica cuando están equipados con una válvula eléctrica.
- **Hidráulicos:** Cilindros o motores, sólo se utilizan cuando los accionadores eléctricos y neumáticos no dan satisfacción, ya que exigen que se instale en la máquina un grupo generador de presión. Son necesarios cuando los esfuerzos a desarrollar son muy importantes (p.ej. en prensas) o cuando las velocidades lentas se deben controlar con precisión (p.ej. avances de útiles de corte).

➤ **Diálogo Hombre-Máquina.**

Permiten las intervenciones del personal de explotación, de ajuste y de mantenimiento.

Algunos están implantados en el pupitre de mando de la máquina: pulsadores, pilotos, teclados programables y configurables, visualizadores, etc.

Otros son conectables, en caso de necesidad, para un diagnóstico o un reglaje: Terminales de ajuste y de diagnóstico rápido.

Parte de Mando.

Es la que emite órdenes hacia la Parte Operativa y recibe las señales de retorno para coordinar sus acciones. Realizada cada vez más con tecnologías de tratamientos programables.

➤ **Tecnologías cableadas.**

Con este tipo de tecnología, el automatismo se realiza interconectando los distintos elementos que lo integran. Su funcionamiento es establecido por los elementos que lo componen y por la forma de conectarlos.

Ésta fue la primera solución que se utilizó para crear autómatas industriales, pero presenta varios inconvenientes (p.ej. nivel de formación alto del operario, sistema poco flexible, etc.).

Los dispositivos que se utilizan en las tecnologías cableadas para la realización del automatismo son:

- Reles electromagnéticos.
- Módulos lógicos neumáticos.
- Tarjetas electrónicas.

➤ **Tecnologías programadas.**

Los avances en el campo de los microprocesadores de los últimos años han favorecido la generalización de las tecnologías programadas. En la realización de automatismos. Los equipos realizados para este fin son:

- Los ordenadores.
- Los Robots
- Los autómatas programables.

El ordenador, como parte de mando de un automatismo presenta la ventaja de ser altamente flexible a modificaciones de proceso. Pero, al mismo tiempo, y debido a su diseño no específico para su entorno industrial, resulta un elemento frágil para trabajar en entornos de líneas de producción.

Los Robots resultan muy rápidos y precisos, y en la actualidad imprescindibles en instalaciones de soldadura, paletizados, etc. Su programación es similar a la de un PC.

Un autómata programable industrial es un elemento robusto diseñado especialmente para trabajar en ambientes de talleres, con casi todos los elementos del ordenador.

OBJETIVOS DE LA AUTOMATIZACIÓN.

- Mejorar la productividad de la empresa, reduciendo los costes de la producción y mejorando la calidad de la misma.
- Mejorar las condiciones de trabajo del personal, suprimiendo los trabajos penosos e incrementando la seguridad.
- Realizar las operaciones imposibles de controlar intelectual o manualmente.
- Mejorar la disponibilidad de los productos, pudiendo proveer las cantidades necesarias en el momento preciso.
- Simplificar el mantenimiento de forma que el operario no requiera grandes conocimientos para la manipulación del proceso productivo.
- Integrar la gestión y producción.
- Mejorar la calidad del producto mediante controles automatizados.

2.2. Definición y Generalidades.

Basándonos en equipos actuales, un Autómata Programable o PLC (Controlador Lógico Programable) se puede definir como un equipo electrónico el cual realiza la ejecución de un programa de forma cíclica. La ejecución del programa puede ser interrumpida momentáneamente para realizar otras tareas consideradas más prioritarias, pero el aspecto más importante es la garantía de ejecución completa del programa principal [4].

Su manejo y programación puede ser realizado por personal eléctrico o electrónico sin conocimientos informáticos. Realiza funciones lógicas: series, paralelos, temporizaciones, contajes y otras más potentes como cálculos, regulaciones, etc.

El autómata sustituye a los sistemas tradicionales de cableado ahorrando trabajo, espacio, etc., puesto que los elementos tradicionales (como relés auxiliares, de enclavamiento, temporizadores, contadores, etc.) son internos.

La potencia de un PLC está directamente relacionada con la velocidad de ejecución del programa y las variables tratadas. Un PLC del mercado actual tarda unos 0,15 ms por cada Kinstrucciones, o sea que el resultado es perfecto para el control de cualquier automatismo. El fin de dicha ejecución es provocar el cambio de las variables tratadas. Este cambio sobre las variables se realiza antes, durante y al final del programa. Antes del programa se realiza la lectura de las Entradas (inicio de ciclo), al final se realiza la escritura de las Salidas (fin de ciclo y enlace con el inicio). Durante la ejecución del programa se realiza la lectura y/o escritura de las variables internas según el contexto programado [4].

Una visión global de un automatismo nos define varios conceptos [4]:

- Una alimentación principal del sistema
- Una adquisición de datos del estado de la instalación o del proceso (Entradas)
- Un proceso (tratamiento) de esos datos (PLC) (antes relés)
- Un resultado plasmado sobre unos accionadores auxiliares (Salidas)
- La variación real sobre la instalación o el proceso (movimientos, activaciones, cualquier cambio)

De forma paralela existe un diálogo llamado hombre-máquina el cual va a permitir modificar a conveniencia el proceso. Este diálogo se realiza a través de simples pulsadores, interruptores, pedales, etc.; o bien a través de algún terminal (teclas) o/y ordenador [4].

CAMPOS DE APLICACIÓN.

El PLC por sus especiales características de diseño tiene un campo de aplicación muy extenso. La constante evolución del hardware y software amplía constantemente este campo para poder satisfacer las necesidades que se detectan en el espectro de sus posibilidades reales.

Su utilización se da fundamentalmente en aquellas instalaciones en donde es necesario un proceso de maniobra, control, señalización, etc., por tanto, su aplicación abarca desde procesos de fabricación industriales de cualquier tipo a transformaciones industriales, control de instalaciones, etc.

Sus reducidas dimensiones, la extremada facilidad de su montaje, la posibilidad de almacenar los programas para su posterior y rápida utilización, la modificación o alteración de los mismos, etc., hace que su eficacia se aprecie fundamentalmente en procesos en que se producen necesidades tales como:

- Espacio reducido
- Procesos de producción periódicamente cambiantes
- Procesos secuenciales
- Maquinaria de procesos variables
- Instalaciones de procesos complejos y amplios
- Chequeo de programación centralizada de las partes del proceso

VENTAJAS E INCONVENIENTES.

No todos los autómatas ofrecen las mismas ventajas sobre la lógica cableada, ello es debido, principalmente, a la variedad de modelos existentes en el mercado y las innovaciones técnicas que surgen constantemente. Tales consideraciones me obligan a referirme a las ventajas que proporciona un autómata de tipo medio.

➤ **Ventajas.**

- Menor tiempo empleado en la elaboración de proyectos.
- No es necesario dibujar el esquema de contactos.
- No es necesario simplificar las ecuaciones lógicas, ya que, por lo general la capacidad de almacenamiento del módulo de memoria es lo suficientemente grande.
- La lista de materiales queda sensiblemente reducida, y al elaborar el presupuesto correspondiente eliminaremos parte del problema que supone el contar con diferentes proveedores, distintos plazos de entrega, etc.
- Posibilidad de introducir modificaciones sin cambiar el cableado ni añadir aparatos.
- Mínimo espacio de ocupación.

- Menor coste de mano de obra de la instalación.
- Economía de mantenimiento. Además de aumentar la fiabilidad del sistema, al eliminar contactos móviles, los mismos autómatas pueden indicar y detectar averías.
- Posibilidad de gobernar varias máquinas con un mismo autómata.
- Menor tiempo para la puesta en funcionamiento del proceso al quedar reducido el tiempo cableado.
- Si por alguna razón la máquina queda fuera de servicio, el autómata sigue siendo útil para otra máquina o sistema de producción.

➤ **Inconvenientes.**

- Como inconvenientes podríamos hablar, en primer lugar, de que hace falta un programador, lo que obliga a formar a uno de los técnicos en tal sentido, pero hoy en día ese inconveniente está solucionado porque las universidades ya se encargan de dicha formación.
- El coste inicial también puede ser un inconveniente, ya que es mayor, pero en una instalación media el plazo de amortización es bastante corto.

FUNCIONES BÁSICAS DE UN PLC.

- Detección: Lectura de la señal de los captadores distribuidos por el sistema de fabricación.
- Mando: Elaborar y enviar las acciones al sistema mediante los accionadores y preaccionadores.
- Diálogo Hombre-máquina: Mantener un diálogo con los operarios de producción, obedeciendo sus consignas e informando del estado del proceso.
- Programación: Para introducir, elaborar y cambiar el programa de aplicación del autómata. El diálogo de programación debe permitir modificar el programa, incluso con el autómata controlando la máquina.
- Redes de comunicación: Permiten establecer comunicación con otras partes de control. Las redes industriales permiten la comunicación y el intercambio de datos entre diferentes dispositivos (autómatas, servidores, etc.) a tiempo real. En unos cuantos milisegundos pueden enviarse telegramas e intercambiar tablas de memoria compartida.
- Sistemas de supervisión: También los autómatas permiten comunicarse con ordenadores provistos de programas de supervisión industrial. Esta comunicación se realiza por una red industrial o por medio de una simple conexión por el puerto serie del ordenador.

- Control de procesos continuos: Además de dedicarse al control de sistemas de eventos discretos los autómatas llevan incorporadas funciones que permiten el control de procesos continuos. Disponen de módulos de entrada y salida analógicas y la posibilidad de ejecutar reguladores PID que están programados en el autómata.
- Entradas – Salidas distribuidas: Los módulos de entrada salida no tienen porqué estar en el armario del autómata. Pueden estar distribuidos por la instalación, se comunican con la unidad central del autómata mediante un cable de red.
- Buses de campo: Mediante un sólo cable de comunicación se pueden conectar al bus captadores y accionadores, reemplazando al cableado tradicional. De esta forma se pueden distribuir las entradas y salidas por la instalación sin necesidad de que estén centralizadas en el cuadro de control donde se sitúa el autómata, es decir, utilizaremos una periferia descentralizada.

2.3. Origen e historia de los Automatas.

Los autómatas programables aparecieron en los Estados Unidos de América en los años 1969 – 70, y más particularmente en el sector de la industria del automóvil; fueron empleados en Europa alrededor de dos años más tarde. Su fecha de creación coincide, pues con el comienzo de la era del microprocesador y con la generación de la lógica cableada modular [1].

El autómata es la primera máquina con lenguaje, es decir, un calculador lógico cuyo juego de instrucciones se orienta hacia los sistemas de evolución secuencial [2]. Es en este sentido un precursor, y constituye para los automatistas un esbozo de la máquina ideal.

La creciente difusión de aplicaciones de la electrónica junto con la impresionante disminución del precio de los componentes, el nacimiento y el desarrollo de los microprocesadores y, sobretodo, la miniaturización de los circuitos de memoria han permitido una introducción masiva de los autómatas programables en la industria, cuyo precio es atractivo incluso para equipos de prestaciones modestas, en una inmensa gama de campos de aplicación.

El autómata programable satisface las exigencias tanto de procesos continuos como discontinuos. Regula presiones, temperaturas, niveles y caudales así como todas las funciones asociadas de temporización, cadencia, conteo y lógica. También puede incluir tarjetas de comunicación adicionales (serie, ethernet, etc.), que hacen del autómata un poderoso satélite dentro de una red de control distribuida.

El autómata es un aparato electrónico programable por un usuario programador y destinado a gobernar, dentro de un entorno industrial, máquinas o procesos lógicos secuenciales.

Cuando surgieron los autómatas se pensó que la universalidad de los ordenadores tendería a desaparecer, el futuro parecía abrirse hacia esta nueva clase de dispositivos: máquina para proceso de señales, para la gestión de datos, etc. Pero no ha sido así, desde que surgieron los primeros autómatas hasta la actualidad éstos han evolucionado en gran medida debido a los avances en electrónica, microelectrónica, etc., pero no solo han evolucionado los autómatas, sino que también lo han hecho los ordenadores, y debido al nivel de informatización que han alcanzado las empresas, el volumen de datos que se maneja supera en muchos casos las posibilidades de los autómatas y son tratados por ordenadores, pasando a ser estos los que se encargan de manejar los datos y los autómatas los que controlan los elementos físicos de una instalación.

En la actualidad los autómatas siguen teniendo un papel muy importante en el control de las instalaciones industriales, pero cuentan con otros rivales y/o compañeros en su tarea de realizar el control de las instalaciones como pueden ser los robots, ordenadores industriales (más robustos que los normales para trabajar en entornos industriales), etc.

2.4. Estructura de un Autómata Programable.

Este capítulo está dedicado a conocer al autómata en su parte física o hardware, no sólo en su configuración externa, sino también y fundamentalmente en la parte interna.

El autómata esta compuesto de diferentes elementos como CPU, fuente de alimentación, memoria, E/S, etc. que están colocados de diferente forma y modo según la estructura externa del autómata.

En el apartado dedicado a la estructura externa veremos dónde y cómo se colocan los diferentes elementos que componen el autómata programable.

En el apartado dedicado a la estructura interna veremos la función que desempeña cada uno de los diferentes elementos como CPU, E/S, fuente de alimentación, etc.

ESTRUCTURA EXTERNA.

El término estructura externa o configuración externa de un autómata programable industrial se refiere al aspecto físico exterior del mismo, bloques o elementos en que está dividido.

Actualmente son tres las estructuras más significativas que existen en el mercado:

- Estructura compacta.

Este tipo de autómatas se distingue por presentar en un sólo bloque todos sus elementos, esto es, fuente de alimentación, CPU, memorias, entradas/salidas, etc.

Son los autómatas de gama baja o nanoautómatas los que normalmente tienen una estructura compacta. Su potencia de proceso suele ser muy limitada dedicándose a controlar máquinas muy pequeñas o cuadros de mando.

- Estructura semimodular. (Estructura Americana)

Se caracteriza por separar las E/S del resto del autómata, de tal forma que en un bloque compacto están reunidas las CPU, memoria de usuario o de programa y fuente de alimentación y separadamente las unidades de E/S.

Son los autómatas de gama media los que suelen tener una estructura semimodular.

- Estructura modular. (Estructura Europea)

Su característica principal es la de que existe un módulo para cada uno de los diferentes elementos que componen el autómata como puede ser una fuente de alimentación, CPU, E/S, etc. La sujeción de los mismos se hace por carril DIN, placa perforada o sobre RACK, en donde va alojado el BUS externo de unión de los distintos módulos que lo componen.

Son los autómatas de gama alta los que suelen tener una estructura modular, que permiten una gran flexibilidad en su constitución.

ESTRUCTURA INTERNA.

En este apartado vamos a estudiar la estructura interna de cada uno de los diferentes elementos que componen el autómata, las funciones y funcionamiento de cada una de ellas.

El autómata está constituido por diferentes elementos, pero los básicos son tres:

- CPU
- Entradas
- Salidas

Con las partes mencionadas podemos decir que tenemos un autómata, pero para que sea operativo son necesarios otros elementos tales como:

- Fuente de alimentación
- Interfaces
- La unidad o consola de programación
- Los dispositivos periféricos

La estructura básica de un autómata programable es la siguiente:

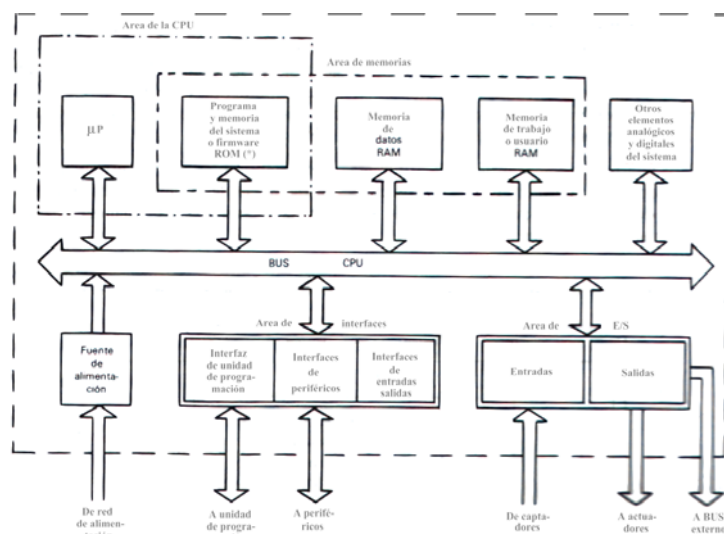


Figura 2. [3]

➤ Fuente de alimentación:

La fuente de alimentación proporciona las tensiones necesarias para el funcionamiento de los distintos circuitos del sistema.

La alimentación a la CPU puede ser de continua a 24 Vcc, tensión muy frecuente en cuadros de distribución, o en alterna a 110/220 Vca. En cualquier caso es la propia CPU la que alimenta las interfaces conectadas a través del bus interno.

La alimentación a los circuitos E/S puede realizarse, según tipos, en alterna a 48/110/220 Vca o en continua a 12/24/48 Vcc.

La fuente de alimentación del autómatas puede incorporar una batería tampón, que se utiliza para el mantenimiento de algunas posiciones internas y del programa usuario en memoria RAM, cuando falla la alimentación o se apaga el autómatas. Aunque lo habitual es que esta batería la lleve la CPU.

➤ Unidad Central de Procesos o CPU:

La CPU (Central Processing Unit) es la parte inteligente del sistema. Interpreta las instrucciones del programa de usuario y consulta el estado de las entradas. Dependiendo de dichos estados y del programa, ordena la activación de las salidas deseadas.

La CPU está constituida por los siguientes elementos:

- Procesador
- Memoria monitor del sistema
- Circuitos auxiliares

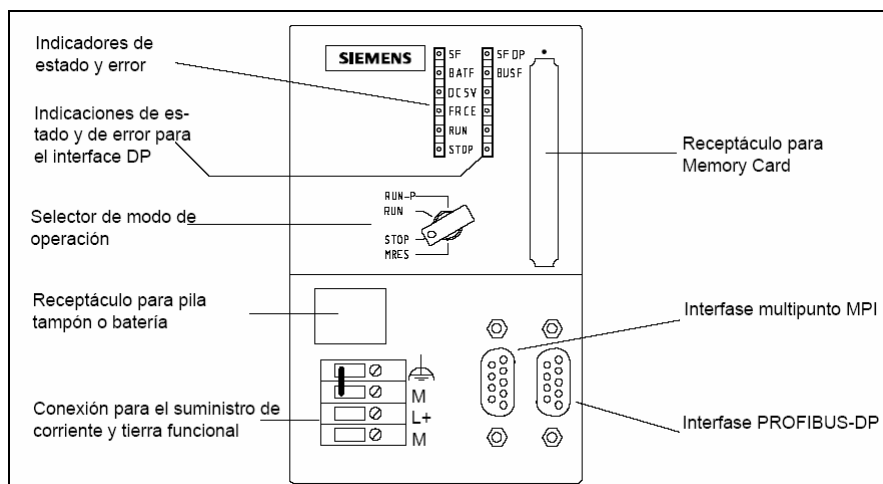


Figura 3. [6]

La figura anterior muestra los elementos de manejo e indicación de una CPU de SIEMENS modelo S7-300. En algunas CPU, los elementos están dispuestos de manera diferente a la representada aquí. Las distintas CPU no poseen siempre todos los elementos representados en la figura.

En cuanto a las CPU de otros fabricantes se podría decir que básicamente son similares, ya que prácticamente todas tienen la opción para seleccionar el modo de operación, indicadores de estados, interfases, etc.

➤ Procesador

Está constituido por el microprocesador, el reloj (generador de onda cuadrada) y algún chip auxiliar.

El microprocesador es un circuito integrado (chip), que realiza una gran cantidad de operaciones, que podemos agrupar en:

- Operaciones de tipo lógico.
- Operaciones de tipo aritmético.
- Operaciones de control de la transferencia de la información dentro del autómata.

Para que el microprocesador pueda realizar todas estas operaciones está dotado de unos circuitos internos que son los siguientes:

- *Circuitos de la unidad aritmética y lógica o ALU:* Es la parte del μp donde se realizan los cálculos y las decisiones lógicas para controlar el autómata.
- *Circuitos de la unidad de control (UC) o Decodificador de instrucciones:* Decodifica las instrucciones leídas en memoria y se generan las señales de control.
- *Acumulador:* Es la encargada de almacenar el resultado de la última operación realizada por el ALU.
- *Flags:* Flags, o indicadores de resultado, que pueden ser consultados por el programa.
- *Contador de programa:* Encargada de la lectura de las instrucciones de usuario.
- *Bus (interno):* No son circuitos en si, sino zonas conductoras en paralelo que transmiten datos, direcciones, instrucciones y señales de control entre las diferentes partes del μp .

➤ Memoria monitor del sistema

Es una memoria de tipo ROM, y además del sistema operativo del autómata contiene las siguientes rutinas, incluidas por el fabricante.

- Inicialización tras puesta en tensión o reset.
- Rutinas de test y de respuesta a error de funcionamiento.
- Intercambio de información con unidades exteriores.
- Lectura y escritura en las interfaces de E/S.

➤ Funciones básicas de la CPU

En la memoria ROM del sistema, el fabricante ha grabado una serie de programas ejecutivos, software del sistema y es a estos programas a los que accederá el μp para realizar las funciones.

El software del sistema de cualquier autómatas consta de una serie de funciones básicas que realiza en determinados tiempos de cada ciclo.

En general, cada autómatas contiene y realiza las siguientes funciones:

- Vigilar que el tiempo de ejecución del programa de usuario no exceda de un determinado tiempo máximo. A esta función se le denomina Watchdog.
- Ejecutar el programa usuario.
- Crear una imagen de las entradas, ya que el programa de usuario no debe acceder directamente a dichas entradas.
- Renovar el estado de las salidas en función de la imagen de las mismas, obtenida al final del ciclo de ejecución del programa usuario.
- Chequeo del sistema.

➤ **Memoria:**

La memoria es el almacén donde el autómatas guarda todo cuanto necesita para ejecutar la tarea de control.

Datos del proceso:

- Señales de planta, entradas y salidas.
- Variables internas, de bit y de palabra.
- Datos alfanuméricos y constantes.

Datos de control:

- Instrucciones de usuario (programa)
- Configuración del autómatas (modo de funcionamiento, número de e/s conectadas, ...)

Existen varios tipos de memorias:

- RAM. Memoria de lectura y escritura.
- ROM. Memoria de solo lectura, no reprogramable.
- EPROM. Memoria de solo lectura, reprogramables con borrado por ultravioletas.
- EEPROM. Memoria de solo lectura, alterables por medios eléctricos.

La memoria RAM se utiliza principalmente como memoria interna, y únicamente como memoria de programa en el caso de que pueda asegurarse el mantenimiento de los datos con una batería exterior.

La memoria ROM se utiliza para almacenar el programa monitor del sistema como hemos visto en el apartado dedicado a la CPU.

Las memorias EPROM se utilizan para almacenar el programa de usuario, una vez que ha sido convenientemente depurado.

Las memorias EEPROM se emplean principalmente para almacenar programas, aunque en la actualidad es cada vez más frecuente el uso de combinaciones RAM + EEPROM, utilizando estas últimas como memorias de seguridad que salvan el contenido de las RAM. Una vez reanudada la alimentación, el contenido de la EEPROM se vuelca sobre la RAM. Las soluciones de este tipo están sustituyendo a las clásicas RAM + batería puesto que presentan muchos menos problemas.

Podemos dividir la memoria en varias secciones:

- *Memoria del programa de usuario:* aquí introducimos el programa que el PLC va a ejecutar cíclicamente.
- *Memoria de la tabla de datos:* es la zona encargada de atribuir las funciones específicas del programa. Se suele subdividir en zonas según el tipo de datos (como marcas de memoria, temporizadores, contadores, etc.).
- *Memoria del sistema:* aquí se encuentra el programa en código de máquina que monitoriza el sistema (programa del sistema o firmware). Este programa es ejecutado directamente por el microprocesador/microcontrolador que posea el PLC.
- *Memoria de almacenamiento:* se trata de una memoria externa que empleamos para almacenar el programa de usuario, y en ciertos casos parte de la memoria de la tabla de datos. Suele ser de uno de los siguientes tipos: EPROM, EEPROM, o FLASH.

Cada PLC divide su memoria de esta forma genérica, haciendo subdivisiones específicas según el modelo y fabricante.

➤ **Periféricos:**

El PLC, en la mayoría de los casos, puede ser ampliable. Las ampliaciones abarcan un gran abanico de posibilidades, que van desde las redes internas (LAN, etc.), módulos auxiliares de E/S, memoria adicional, etc., hasta la conexión con otros autómatas.

Cada fabricante facilita las posibilidades de ampliación de sus modelos, los cuales pueden variar incluso entre modelos de la misma serie.

➤ Entradas y Salidas:

En el control de un proceso automatizado, es imprescindible un dialogo entre operador-máquina junto con una comunicación entre la máquina y el autómatas, estas comunicaciones se establecerán por medio del conjunto de entradas y salidas del citado elemento.

Los autómatas son capaces de manejar tensiones y corrientes de nivel industrial, gracias a que disponen un bloque de circuitos de interfaz de E/S muy potente, que les permite conectarse directamente con los sensores y accionamientos del proceso.

La sección de entradas mediante el interfaz, adapta y codifica de forma comprensible para la CPU las señales procedentes de los dispositivos de entrada o captadores.

Hay dos tipos de entradas:

- Entradas digitales
- Entradas analógicas

La sección de salida también mediante interfaz trabaja de forma inversa a las entradas, es decir, decodifica las señales procedentes de la CPU, y las amplifica y comanda con ellas los dispositivos de salida o actuadores como lámparas, relés, etc. aquí también existen unos interfaces de adaptación a las salidas de protección de circuitos internos.

Hay dos tipos de salidas:

- Salidas digitales
- Salidas analógicas

➤ Entradas digitales

Los módulos de entrada digitales permiten conectar al autómatas captadores de tipo todo o nada como finales de carrera pulsadores, etc.

Los módulos de entrada digitales trabajan con señales de tensión, por ejemplo cuando por una vía llegan 24 voltios se interpreta como un "1" y cuando llegan cero voltios se interpreta como un "0".

El proceso de adquisición de la señal digital consta de varias etapas.

- Protección contra sobretensiones
- Filtrado
- Puesta en forma de la onda
- Aislamiento galvánico o por optoacoplador.

➤ Entradas analógicas

Los módulos de entrada analógicas permiten que los autómatas programables trabajen con accionadores de mando analógico y lean señales de tipo analógico como pueden ser la temperatura, la presión, el caudal, la intensidad, etc.

Los módulos de entradas analógicas convierten una magnitud analógica en un número que se deposita en una variable interna del autómata. Lo que realiza es una conversión A/D, puesto que el autómata solo trabaja con señales digitales. Esta conversión se realiza con una precisión o resolución determinada (numero de bits) y cada cierto intervalo de tiempo (periodo muestreo).

Los módulos de entrada analógica pueden leer tensión o intensidad. El proceso de adquisición de la señal analógica consta de varias etapas:

- Filtrado
- Conversión A/D
- Memoria interna

➤ Salidas digitales

Un módulo de salida digital permite al autómata programable actuar sobre los preaccionadores y accionadores que admitan órdenes de tipo todo o nada.

Según el tipo de proceso a controlar por el autómata, podemos utilizar diferentes módulos de salidas. Existen tres tipos bien diferenciados:

- A relé: son usados en circuitos de corriente continua y corriente alterna. Están basados en la conmutación electromecánica realizada por la bobina de un relé interno del autómata. Son muy útiles cuando en una misma instalación hay elementos comandados con diferentes tensiones de alimentación, ya que nos suministran un contacto libre de potencial.
- A triac: se utilizan en circuitos de corriente continua y corriente alterna que necesitan maniobras de conmutación muy rápidas. Es un módulo de salida estático, y todas las salidas deben trabajar a la misma tensión.
- A transistor: son utilizados en circuitos que necesiten maniobras de conexión / desconexión muy rápidas. El uso de este tipo de módulos es exclusivo de los circuitos de corriente continua, además todas las salidas deben trabajar a la misma tensión.

El proceso de envío de la señal digital consta de varias etapas:

- Puesta en forma
- Aislamiento
- Circuito de mando (relé interno)
- Protección electrónica
- Tratamiento cortocircuitos

➤ Salidas analógicas

Los módulos de salida analógica permiten que el valor de una variable numérica interna del autómatas se convierta en tensión o intensidad.

Lo que realiza es una conversión D/A, puesto que el autómatas solo trabaja con señales digitales. Esta conversión se realiza con una precisión o resolución determinada (número de bits) y cada cierto intervalo de tiempo (periodo muestreo).

Esta tensión o intensidad puede servir de referencia de mando para actuadores que admitan mando analógico como pueden ser los variadores de velocidad, las etapas de los tiristores de los hornos, reguladores de temperatura, etc. permitiendo al autómatas realiza funciones de regulación y control de procesos continuos.

El proceso de envío de la señal analógica consta de varias etapas:

- Aislamiento galvánico
- Conversión D/A
- Circuitos de amplificación y adaptación
- Protección electrónica de la salida

Como hemos visto las señales analógicas sufren un gran proceso de adaptación tanto en los módulos de entrada como en los módulos de salida. Las funciones de conversión A/D y D/A que realiza son esenciales. Por ello a los módulos de E/S analógicos se les considera módulos de E/S especiales.

➤ **Interfaces:**

Todo PLC, salvo casos excepcionales, posee la virtud de poder comunicarse con otros dispositivos (como un PC).

Lo normal es que posea una interfase serie del tipo RS-232 / RS-422.

A través de esta línea se pueden manejar todas las características internas del controlador, incluida la programación del mismo, y suele emplearse para monitorización del proceso en otro lugar separado.

El PLC elegido para este proyecto (Siemens S7-313) tiene un interfase MPI.

La MPI es la interfase de la CPU para el equipo PG/OP (ordenador programación/panel de operador) o para la comunicación en una subred MPI.

Se pueden enviar datos a 4 distintos aparatos al mismo tiempo y utilizando siempre el mismo puerto a una velocidad de 187,5 Kbits / seg o 187,5 Kbaudios. Para pequeñas redes de comunicación o pequeños volúmenes de datos la CPU ofrece el servicio de Datos Globales, que permite intercambiar cíclicamente cantidades de datos en paquetes de hasta 22 bytes como máximo.

Distancia máxima entre dos estaciones o nodos de red de MPI adyacentes: 50 metros (sin repetidores); 1100 metros (con dos repetidores); 9100 metros (con más de 10 repetidores en serie); por encima de los 500 Km. (cable de fibra óptica, con módulos de conexión ópticas).

La CPU envía por la interfase MPI automáticamente los parámetros de bus que tiene ajustados (p.ej. la velocidad en baudios). De esta manera una unidad de programación puede conectarse automáticamente en una subred MPI.

Equipos conectables a una subred MPI:

- PG/PC y OP
- Automatas S7 con interfase MPI (S7-300, M7-300, S7-400, M7-400, C7-6xx)
- S7-200 (Nota: sólo 19,2 kbaudios)

Puesto que, el autómata de la serie S7-300 es modular puede tener diferentes interfases de comunicación utilizando tarjetas específicas de SIEMENS del tipo CP-xxx.

- Procesadores de comunicación CP 343-5, CP 343-1 y CP 343 TCP para conexión al PROFIBUS y sistemas bus de Ethernet Industrial.
- Procesador de comunicaciones CP 340 para conexión a sistemas punto a punto.

➤ **Terminal de programación:**

El terminal de programación es el que permite al automatista programar el PLC.

Las funciones básicas de éste son las siguientes:

- Transferencia y modificación de programas.
- Verificación de la programación.
- Información del funcionamiento de los procesos.



Como terminal de programación pueden ser utilizadas las consolas de programación construidas específicamente para el autómata o bien un ordenador personal (normalmente portátil), que soporte un software específicamente diseñado por el fabricante para llevar a cabo la programación y supervisión del Autómata.

En la actualidad básicamente se utiliza para programar un ordenador, casi siempre, portátil con el fin de depurar el programa desarrollado en la propia instalación. La consola de programación nos va a permitir una mayor autonomía que el PC, pero el mayor problema estriba en que estas consolas, hoy día, están pensadas para programar PLC pequeños (de hasta 48 E/S). Es evidente que en PLC superiores una programación con estas consolas se convierte en un proceso tedioso ya que se visualizan, normalmente, una o dos líneas del programa escrito. Estas consolas sí tienen utilidad, para modificar datos, bien de autómatas pequeños como de un calibre superior.

2.5. Funcionamiento del Autómata.

Los autómatas programables son máquinas secuenciales que ejecutan correlativamente las instrucciones indicadas en el programa de usuario almacenado en su memoria, generando unas órdenes o señales de mando a partir de las señales de entrada leídas de la planta (aplicación): al detectarse cambios en las señales, el autómata reacciona según el programa hasta obtener las órdenes de salida necesarias. Esta secuencia se ejecuta continuamente para conseguir el control actualizado del proceso.

La secuencia básica de operación del autómata se puede dividir en tres fases principales:

- Lectura de señales desde la interfaz de entradas.
- Procesado del programa para obtención de las señales de control.
- Escritura de señales en la interfaz de salidas.

A fin de optimizar el tiempo, la lectura y escritura de las señales se realiza a la vez para todas las entradas y salidas. Entonces, las entradas leídas de los módulos de entrada se guardan en una memoria temporal (imagen entradas). A ésta acude la CPU en la ejecución del programa, y según se van obteniendo las salidas, se guardan en otra memoria temporal (imagen de salida). Una vez ejecutado el programa completo, estas imágenes de salida se transfieren todas a la vez al módulo de salida.

El autómata realiza también otra serie de acciones que se van repitiendo periódicamente, definiendo un ciclo de operación. Dichas acciones se pueden observar en el diagrama de bloques de la figura 4.

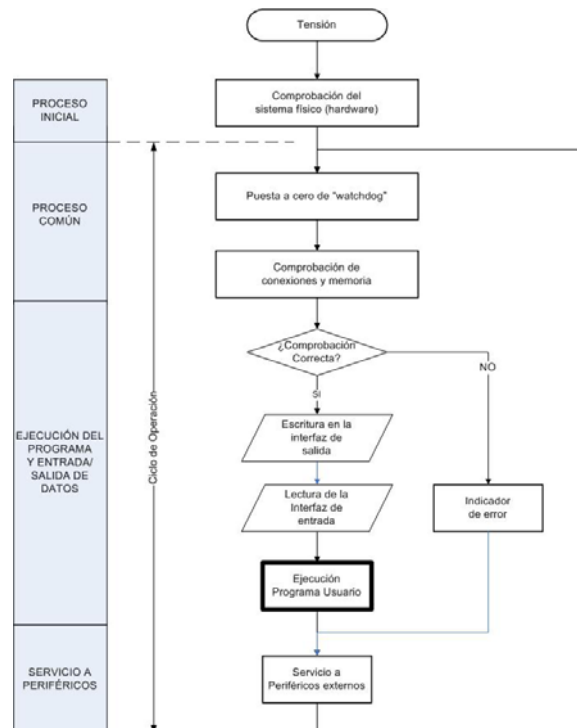


Figura 4.

MODO DE FUNCIONAMIENTO.

El Autómata que comanda la instalación de este proyecto (SIEMENS serie S7-300) tiene 4 modos de operación seleccionables mediante el interruptor llave como se muestra en la figura 3, y puede trabajar de tres formas diferentes:

- *RUN-P (Modo RUN-PROGRAM)*: La CPU ejecuta el programa de aplicación. En este modo de operación no se puede sacar la llave y permite programar la CPU mientras ejecuta el programa sin alterar el funcionamiento de la instalación.
- *RUN (Modo RUN)*: La CPU ejecuta el programa de aplicación. En este modo de operación se puede sacar la llave para que nadie pueda modificar el modo de operación sin autorización. Con este modo seleccionado podríamos monitorizar el funcionamiento de la CPU, pero no alterar el programa (sólo lectura).
- **STOP (Modo de operación STOP)**: La CPU no ejecuta ningún programa de aplicación. El interruptor llave se puede colocar en esta posición para que nadie pueda modificar el modo de operación sin autorización. En este modo si se puede programar (lectura/escritura).
- **MRES (Borrado Total)**: Posición no estable del selector de modo de operación, para el borrado total de la CPU. El borrado total a través del selector de modo de operación requiere una secuencia de manejo especial.

CICLO DE FUNCIONAMIENTO.

El funcionamiento del autómata es, salvo el proceso inicial que sigue a un Reset, de tipo secuencial y cíclico, es decir, las operaciones tienen lugar una tras otra, y se van repitiendo continuamente mientras el autómata esté bajo tensión.

La figura 4 muestra esquemáticamente la secuencia de operaciones que ejecuta el autómata, siendo las operaciones del ciclo de operación las que se repiten indefinidamente.

El ciclo de funcionamiento se divide en dos partes como se puede observar en el esquema de diagrama de la figura 4 llamados Proceso Inicial y Ciclo de Operación.

➤ **Proceso Inicial.**

Como se muestra en la figura 4, antes de entrar en el ciclo de operación el autómata realiza una serie de acciones comunes, que tratan fundamentalmente de inicializar los estados del mismo y chequear el hardware.

Estas rutinas de chequeo comprueban:

- El bus de conexiones de las unidades de E/S.
- El nivel de la batería, si ésta existe.
- La conexión de las memorias internas del sistema.
- El módulo de memoria exterior conectado, si existe.
- Etc.

Si se encontrara algún error en el chequeo, se activaría el LED de error y quedaría registrado el código del error.

Comprobadas las conexiones, se inicializan las variables internas:

- Se ponen a OFF las posiciones de memoria interna, excepto las mantenidas o protegidas contra pérdidas de tensión (memoria remanente).
- Se borran todas las posiciones de memoria imagen E/S.
- Se borran todos los contadores y temporizadores, excepto los mantenidos o protegidos contra pérdidas de tensión.

Transcurrido el Proceso Inicial y si no han aparecido errores o éstos han sido tratados por programa (algunos errores pueden ser controlados por programa y hacer que el autómatas continúe funcionando obviando el error), el autómatas entra en el Ciclo de Operación.

➤ **Ciclo de operación.**

Este ciclo puede considerarse dividido en tres bloques tal y como se puede observar en la figura 4, dichos bloques son:

- Proceso Común
- Ejecución del programa
- Servicio a periféricos

➤ Proceso común:

En este primer bloque se realizan los chequeos cíclicos de conexiones y de memoria de programa, protegiendo el sistema contra:

- Errores de hardware (conexiones E/S, etc.).
- Errores de sintaxis (programa imposible de ejecutar).
- Niveles de tensión de alimentación.
- Estado de la batería si existe.
- Buses de conexión con las interfaces.

➤ Ejecución del programa:

En este segundo bloque se consultan los estados de las entradas y de las salidas y se elaboran las órdenes de mando o de salida a partir de ellos.

El tiempo de ejecución de este bloque de operaciones es la suma del:

- Tiempo de acceso a interfaces de E/S.
- Tiempo de escrutación de programa.

Y, a su vez, esto depende respectivamente de:

- Número y ubicación de las interfaces de E/S.
- Longitud del programa y tipo de CPU que lo procesa.

➤ Servicio a periféricos:

Este tercer y último bloque es únicamente atendido si hay pendiente algún intercambio con el exterior. En caso de haberlo, la CPU le dedica un tiempo limitado, de 1 a 2 ms, en atender el intercambio de datos. Si este tiempo no fuera suficiente, el servicio queda interrumpido hasta el siguiente ciclo.

TIEMPO DE EJECUCIÓN Y CONTROL EN TIEMPO REAL.

El tiempo total que el autómata emplea para realizar un ciclo de operación se llama "Tiempo de ciclo (Scan time)".

Dicho tiempo depende de:

- El número de E/S involucradas.
- La longitud del programa usuario (cantidad y tipo de instrucciones, etc.).
- El número y tipo de periféricos conectados al autómata.

Los tiempos totales de ciclos son entonces la suma de tiempos empleados en realizar las distintas operaciones del ciclo:

- Autodiagnóstico
- Actualización de E/S.
- Ejecución de programa.
- Servicio a periféricos.

La ejecución cíclica del programa y, con ello, la del programa de usuario se realiza en segmentos de tiempo. Para mostrar de una forma más clara su ejecución, partiremos del supuesto de que cada segmento de tiempo dura exactamente 1 ms.

Para que la CPU disponga de una imagen coherente de las señales de proceso durante la ejecución cíclica del programa, las señales de proceso se leen o escriben antes de esta ejecución. A continuación, durante la ejecución del programa, la CPU direcciona las áreas de operandos Entradas (E) y Salidas (S) sin acceder directamente a los módulos de señal, sino al área de memoria del sistema de la CPU, donde se encuentra la imagen de proceso de las entradas y las salidas.

La siguiente tabla y la figura muestran las fases de la ejecución cíclica del programa.

Paso	Secuencia
(1)	El sistema operativo inicia la vigilancia de tiempo de ciclo.
(2)	La CPU escribe los valores de la imagen de proceso de las salidas en los módulos de salida.
(3)	La CPU lee el estado de las entradas en los módulos de entrada y actualiza la imagen de proceso de las entradas.
(4)	La CPU procesa el programa de usuario en segmentos de tiempo y ejecuta las operaciones indicadas en el programa.
(5)	Al final de un ciclo, el sistema operativo realiza las tareas pendientes, p. ej. carga y borrado de bloques.
(6)	A continuación, la vuelve al comienzo del ciclo y reinicia la vigilancia de tiempo de ciclo.

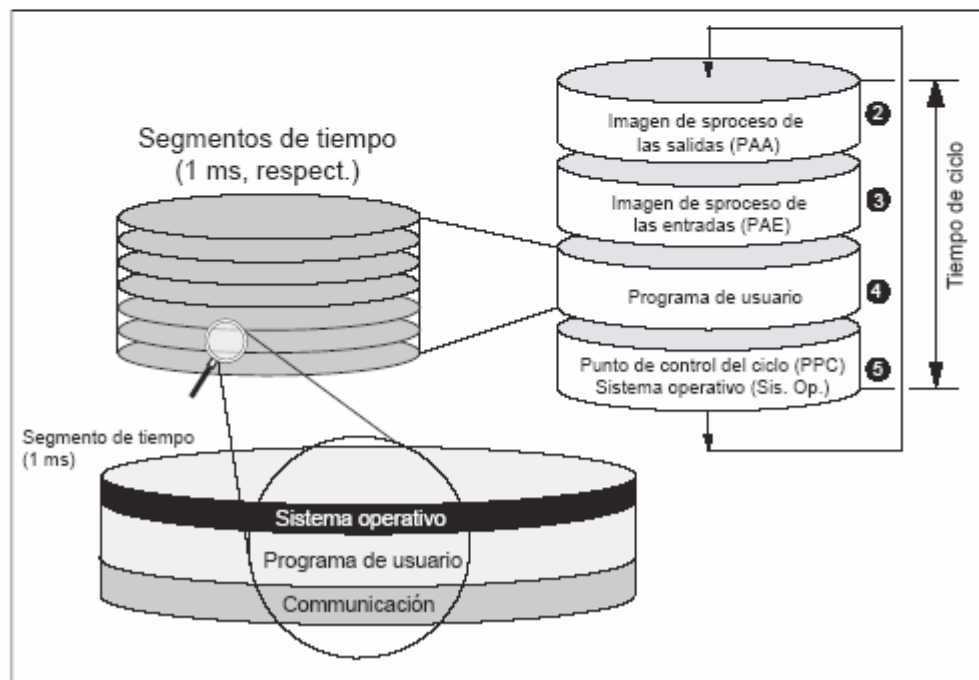


Figura 5. [6]

2.6. Equipos y Lenguajes de Programación.

EQUIPOS DE PROGRAMACIÓN

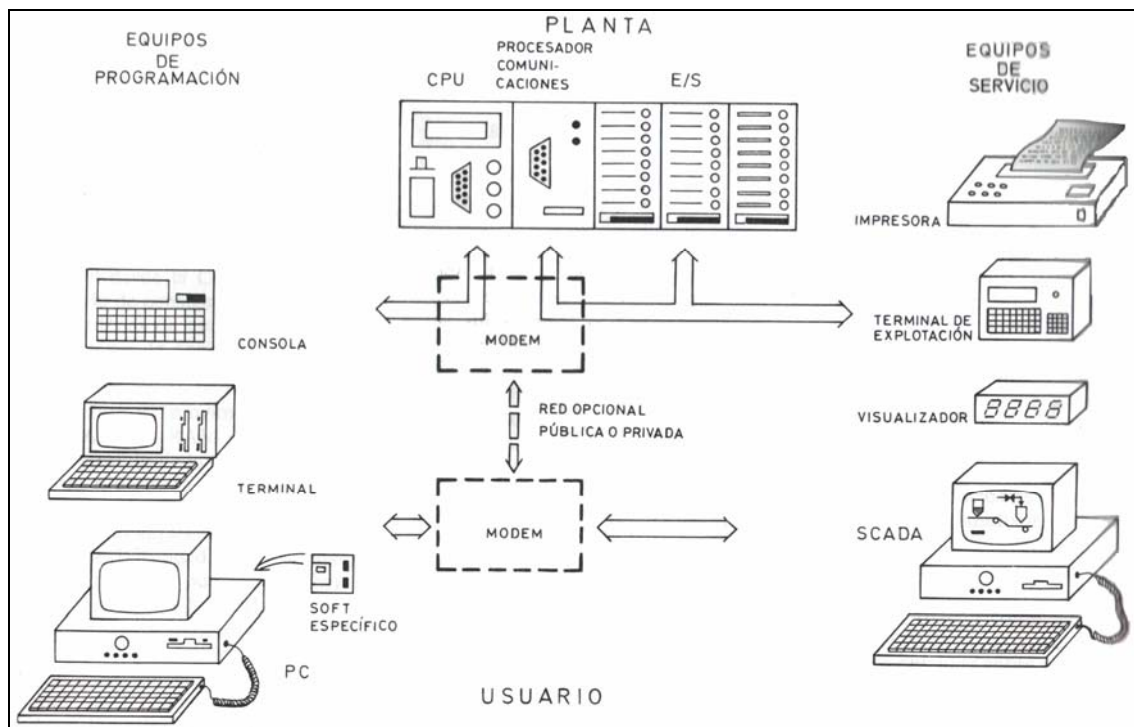


Figura 6. [1]

El equipo de programación de un autómata tiene por misión configurar, estructurar, programar, almacenar y verificar las diferentes funciones del automatismo. Se define entonces el equipo de programación como el conjunto de medios hardware y software mediante los cuales el programador introduce y depura las secuencias de instrucciones (en uno u otro lenguaje) que constituyen el programa a ejecutar y posteriormente se transfiere a la memoria de programa de usuario del PLC.

Son funciones específicas de los tipos de programación las siguientes:

- Creación, edición y documentación del programa o aplicación.
- Almacenamiento y gestión del programa o bloques del programa.
- Transferencias de programas desde y hacia el autómata.
- Gestión de errores del autómata, con identificación de los mismos, ayudas para su localización y corrección, y reinicialización del sistema.
- Puesta en marcha y detención del autómata (RUN / STOP).
- Monitorización del funcionamiento, sobre variables seleccionadas o sobre las propias líneas del programa.
- Forzado de variables binarias o numéricas y preselección de contadores, temporizadores y registros de datos.

Además de la confección del programa, el programador debe introducir una configuración adicional sobre el equipo, a fin de que el sistema pueda verificar el mapa de memoria que se está utilizando, el hardware utilizado en el equipo, etc.

Por tanto, en una aplicación con autómatas programables el usuario actual (programador u operador) necesita comunicación con la máquina para programar y depurar el programa, para acceder a los estados de planta y para forzar secuencias de mando sobre el sistema.

Estas funciones son cubiertas por las unidades de programación, equipos y entornos software inteligentes dedicados, con interfaces sencillas para el usuario, y con canales y protocolos de conexión con el autómata ya resueltos.

Con ellos, el usuario accede a las funciones que necesita (programación, depuración, visualización, forzado, etc.) de forma rápida y flexible, según procedimientos interactivos que facilitan la aplicación del autómata a cualquier proceso industrial.

En general, los equipos de programación podrán ser de dos tipos:

- Específicos, bajo la forma de una consola o terminal conectado directamente a la entrada de programación del autómata.
- Implementados sobre un PC, mediante un software dedicado que corre bajo un entorno operativo actual tipo WINDOWS.

Entre los equipos específicos, y según su complejidad, podemos distinguir dos variantes:

- Consola o unidad de programación.
- Terminales de programación.

➤ **Consolas de programación**

Las consolas son pequeños dispositivos de bolsillo que permiten la programación, ajuste y diagnóstico del autómata, con un visualizador apto para observar una (o pocas) líneas de programa, y un teclado que incluye:

- Teclas de comandos.
- Teclas de instrucciones.
- Teclas alfanuméricas.

Son dispositivos portátiles y de bajo precio, especialmente útiles para las intervenciones de ajuste en planta (forzado de variables, modificación de valores numéricos de preselección, etc.), aunque para aplicaciones más ambiciosas presentan los problemas inherentes a su simplicidad:

- Dificultades para trabajar con más de un programa simultáneamente,
- Dificultades para el almacenamiento y/o impresión de programas.

- Muy baja o inexistente capacidad de documentación del programa (uso de símbolos, textos de ayuda, etc.).

➤ **Terminales de programación.**

El terminal de programación se distingue de la consola por su teclado tipo QWERTY (teclado actual) y su pantalla de gran tamaño, que permite la visualización de bloques completos de programa, con identificación simbólica de las variables y menús de ayuda “on line”.

Los terminales constituyen verdaderas estaciones autónomas de trabajo dedicadas a la programación de autómatas. Permiten programación combinada en distintos lenguajes, edición del programa por bloques, manejo de librerías, simulación del programa resultante, conexión a otros ordenadores por red informática, o a redes de autómatas específicas del fabricante, etc.

El manejo de estos terminales es muy simple, gracias al empleo de teclas funcionales y ayuda en línea que facilitan el acceso a todas sus funciones (programación “off/on line”, búsqueda y corrección de instrucciones, visualización del estado de las señales E/S, mando de variables, archivo en disco, etc.).

En programación “off line”, los programas se confeccionan sin conexión entre el aparato y el autómata, y una vez finalizados se transfieren al autómata.

En programación “on line” los programas se confeccionan, prueban y corrigen sobre el mismo autómata, unido directamente a la unidad de programación.

➤ **Software para la programación.**

Los paquetes de software para programación de autómatas convierten un ordenador personal en un equipo de programación específico, aprovechando sus potentes recursos de interfaz con otros sistemas (impresoras, otros PC) y con el usuario (teclado, monitor), y el bajo precio del hardware debido a la estandarización y generalización de uso de los ordenadores compatibles. En la actualidad esta opción (PC + software) constituye prácticamente la totalidad de equipos de programación utilizados por los programadores de autómatas.

Los requisitos de hardware y software exigidos para instalar sobre el PC un entorno de programación de autómatas suelen ser soportados por cualquier PC actual del mercado, por simple que sea.

El paquete de programación se completa con la unidad externa de conexión que convierte y hace compatibles las señales físicas entre la salida estándar de PC (RS-232C, USB, Ethernet, etc.) y el puerto de conexión del autómata.

LENGUAJES DE PROGRAMACIÓN

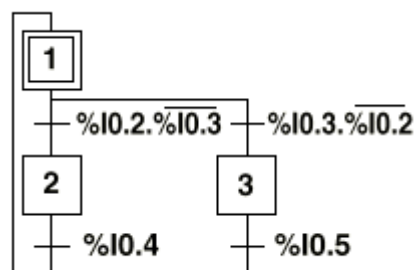
Cuando surgieron los autómatas programables, lo hicieron con la necesidad de sustituir a los enormes cuadros de maniobra construidos con contactores y relés. Por lo tanto, la comunicación hombre-máquina debería ser similar a la utilizada hasta ese momento. El lenguaje usado, debería ser interpretado, con facilidad, por los mismos técnicos electricistas que anteriormente estaban en contacto con la instalación. Estos lenguajes han evolucionado, en los últimos tiempos, de tal forma que algunos de ellos ya no tienen nada que ver con el típico plano eléctrico a relés. [5]

La creciente complejidad en la programación de los autómatas programables hizo que se intentaran estandarizar los lenguajes de programación, y esto dio lugar, en Agosto de 1992 bajo la dirección de la IEC (Comisión Electrónica Internacional), al estándar IEC 1131-3 (IEC 65) para la programación de PLC's. Los lenguajes gráficos y textuales definidos en el estándar son una fuerte base para entornos de programación potentes en PLC's. Con la idea de hacer el estándar adecuado para un gran abanico de aplicaciones, cinco lenguajes han sido definidos en total.

- Gráfico secuencial de funciones (grafcet).
- Lista de instrucciones (IL o AWL).
- Texto estructurado.
- Diagrama de contactos.
- Diagrama de funciones.

➤ Gráfico secuencial de funciones (grafcet).

El gráfico secuencial de funciones (SFC o Grafcet) es un lenguaje gráfico que proporciona una representación en forma de diagrama de las secuencias del programa. Soporta selecciones alternativas de secuencia y secuencias paralelas. Los elementos básicos son pasos y transiciones. Los pasos consisten en piezas de programa que son inhibidas hasta que una condición especificada por las transiciones es cierta. Como consecuencia de que las aplicaciones industriales funcionan en forma de pasos, el SFC es la forma lógica de especificar y programar un PLC.



➤ Elementos Gráficos del Grafset.

▪ Las Etapas

Las etapas representan cada uno de los estados del sistema.

El símbolo empleado para representar una etapa es un cuadrado con un número o símbolo en su interior que la identifica. Las etapas iniciales se representan por un cuadrado con doble línea. Cuando se recorre el gráfico de evolución, por cualquier camino posible, deben alternarse siempre una etapa y una transición.

Las acciones que llevan asociadas las etapas se representan con un rectángulo donde se indica el tipo de acción a realizar. Una etapa puede llevar asociadas varias acciones.

REPRESENTACION DE ETAPAS:



REPRESENTACION DE ETAPAS INICIALES:



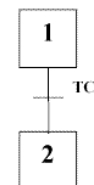
▪ Las Líneas de Evolución

Las líneas de evolución unen entre si las etapas que representan actividades consecutivas. Las líneas se entenderán siempre orientadas de arriba abajo, a menos que se represente una flecha en sentido contrario. Dos líneas de evolución que se crucen debe de interpretarse, en principio que no están unidas.



▪ Las Transiciones

Las transiciones, representan las condiciones lógicas necesarias para que finalice la actividad de una etapa y se inicie la de la etapa o etapas inmediatamente consecutivas. Gráficamente se representan por una línea cruzada sobre las líneas de evolución.



▪ Los Reenvíos

Son símbolos en forma de flecha que indican la procedencia o destino de las líneas de evolución.



➤ Reglas de Evolución.

El proceso se descompone en etapas, que serán activadas de forma secuencial.

Una o varias acciones se asocian a cada etapa. Estas acciones sólo están activas cuando la etapa está activa.

Una etapa se hace activa cuando la precedente lo está y la condición de transición entre ambas etapas ha sido activada.

La activación de una condición de transición implica la activación de la etapa siguiente y la desactivación de la etapa precedente.

La etapa inicial tiene que ser activada antes de que se inicie el ciclo del GRAFCET, un ciclo está formado por todas las etapas posteriores a la etapa inicial.

➤ Estructuras del Grafcet.

Existen procesos que requieren estructuras más complejas, en las que se representan bucles, tomas de decisiones o tareas simultáneas que deben sincronizarse. Para estos casos el GRAFCET dispone de otras estructuras básicas a partir de las cuales pueden generarse los diagramas de dichos progresos.

Las tres estructuras básicas del GRAFCET, de las cuales pueden derivarse todas las demás, son:

- Secuencia Lineal
- Convergencia y divergencia en o (subprocesos alternativos)
- Convergencia y divergencia en y (subprocesos simultaneos)

➤ **Lista de instrucciones.**

La lista de instrucciones (IL o AWL) es un lenguaje de bajo nivel, similar al lenguaje ensamblador. Con IL sólo una operación es permitida por línea (ej. almacenar un valor en un registro). Las instrucciones pueden variar de un fabricante a otro.

000	LD	%I0.1	Bp. inicio ciclo	U	%I0.1	Bp. inicio ciclo
	AND	%I0.0	Dp. presencia vehículo	U	%I0.0	Dp. presencia vehículo
	AND	%M3	Bit autorización reloj calendario	U	%M3.0	Bit autorización reloj calendario
	AND	%I0.5	Fc. alto rodillo	U	%I0.5	Fc. alto rodillo
	AND	%I0.4	Fc. detrás pórtico	U	%I0.4	Fc. detrás pórtico
005	S	%M0	Memo inicio ciclo	S	%M0.0	Memo inicio ciclo
	LD	%M2		U	%M2.0	
	AND	%I0.5		U	%I0.5	
	OR	%I0.2	Bp. parada ciclo	O	%I0.2	Bp. parada ciclo
	R	%M0		R	%M0.0	
010	LD	%M0		U	%M0.0	
	ST	%Q0.0	Piloto ciclo	=	%Q0.0	Piloto ciclo

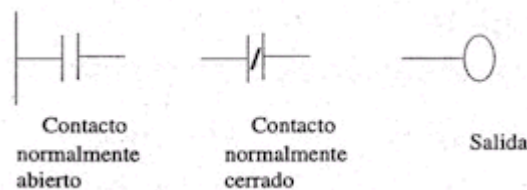
➤ Texto estructurado.

El texto estructurado (structured text o ST) es un lenguaje de alto nivel estructurado por bloques que posee una sintaxis parecida al PASCAL. El ST puede ser empleado para realizar rápidamente sentencias complejas que manejen variables con un amplio rango de diferentes tipos de datos, incluyendo valores analógicos y digitales. También se especifica tipos de datos para el manejo de horas, fechas y temporizaciones, algo importante en procesos industriales. El lenguaje posee soporte para bucles iterantes como REPEAT UNTIL, ejecuciones condicionales empleando sentencias IF-THEN-ELSE y funciones como SQRT() y SIN().

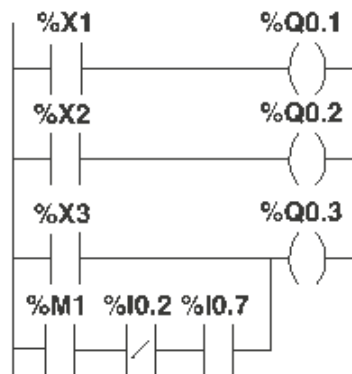
➤ Diagrama de contactos.

Es un lenguaje gráfico, derivado del lenguaje de relés, que mediante símbolos representa contactos, solenoides, etc. Su principal ventaja es que los símbolos básicos están estandarizados por el IEC y son empleados por todos los fabricantes.

Los símbolos básicos son:



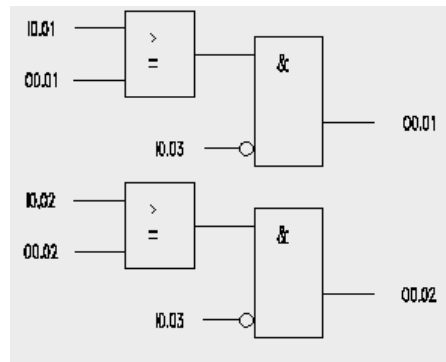
Los elementos básicos que configuran la función se representan entre dos líneas verticales que simbolizan las líneas de alimentación.



Para las líneas de función más complejas como temporizadores, registros de desplazamiento, etc., se emplea el formato de bloques. Estos no están formalizados, aunque guardan una gran similitud entre sí para distintos fabricantes y resultan mucho más expresivos que si se utiliza para el mismo fin el lenguaje en lista de instrucciones o mnemónico.

➤ Diagrama de funciones.

El diagrama de funciones (function block diagram o FBD) es un lenguaje gráfico que permite programar elementos que aparecen como bloques para ser cableados entre si de forma análoga al esquema de un circuito. FBD es adecuado para muchas aplicaciones que involucren el flujo de información o datos entre componentes de control.



➤ Organización de tareas.

El estándar también define una nueva arquitectura para la organización e interacción de tareas con PLC's. Una tarea controla la ejecución de un programa ejecutándolo periódicamente o en respuesta a un específico evento. Para optimizar los recursos del controlador, una aplicación puede ser fragmentada en un número de pequeños programas concretos. Cada programa está bajo el control de una tarea que se ejecuta a la velocidad que requiera la E/S asociada.

➤ Bloques de funciones.

Los bloques de funciones (FB's) son bloques estándar que ejecutan algoritmos como reguladores PID. El estándar IEC asegura que los FB's son definidos empleando una metodología estándar. Hay controles empleando parámetros externos, mientras que los algoritmos internos permanecen ocultos empleando Programación Orientada a Objetos.

➤ El estándar IEC 1131 [4].

El estándar de programación es la tercera parte del estándar 1131 para autómatas programables. Las partes son:

- Parte 1 - Información general (IS)
- Parte 2 - Requerimientos de equipo y test (IS)
- Parte 3 - Lenguajes de programación (IS)
- Parte 4 - Líneas de guía al usuario (CD)
- Parte 5 - Servicios de comunicación (CD)

2.7. Configuración, instalación y puesta a punto.

Configuración

La configuración del autómatas es un proceso mediante el que se determina cómo y dónde se sitúan los distintos componentes del sistema de control.

La configuración dependerá de la tarea de control propiamente dicha y del tipo de control que se haya decidido y contempla tanto los elementos del autómatas como sus periféricos.

Durante la elaboración del algoritmo de control, se han determinado las entradas y salidas, tanto discretas como numéricas, y éstas se han relacionado mediante diagramas o esquemas lógicos: la cantidad y tipo de las E/S determina qué componentes son necesarios.

La mejor manera de realizar la configuración es confeccionar un mapa de direccionamiento, en el que mediante una representación de las estructuras de E/S se indica qué componentes se ubican en el local junto a la unidad central y cuáles se sitúan en posiciones remotas. Concluida la configuración del sistema, pueden comenzar simultáneamente dos trabajos: la programación y la instalación.

Instalación

Dadas las características constructivas y de diseño de los autómatas programables, su instalación es viable en prácticamente cualquier ambiente industrial siempre que no se sobrepasen las especificaciones dadas por el fabricante. No obstante, existen ciertas recomendaciones prácticas para asegurar un correcto funcionamiento del sistema, que atañen principalmente a las condiciones de temperatura y humedad y a la inmunidad frente a interferencias eléctricas.

En general, el autómatas se montará en un armario de maniobra de dimensiones adecuadas para contener con holgura los componentes del equipo y el resto de elementos, como interruptores/seccionadores, fuentes de alimentación, circuitos de protección, conductos de cableado, etc. [2] Se recomienda el empleo de armarios metálicos ya que minimizan los efectos de la radiación electromagnética generada por equipos de conmutación instalados en las inmediaciones. Para la instalación, se seguirán las normas y reglamentos vigentes de aplicación habitual en cualquier instalación eléctrica de control.

La convección natural es suficiente ya que la mayoría de los fabricantes preparan los autómatas para que trabajen a una temperatura máxima de 60°, pero en la mayoría de los casos se utilizará algún sistema de refrigeración, bien sea un ventilador o un climatizador, según las condiciones ambientales y los dispositivos montados en el armario eléctrico, ya que algunos dispositivos disipan bastante calor (p.ej. resistencias de frenado, variadores de frecuencia, etc.).

➤ Situación de los componentes

Los componentes del autómatas se montarán siguiendo las recomendaciones del fabricante y, en todo caso, se pueden seguir las siguientes pautas de aplicación general:

- Es recomendable el montaje vertical de los componentes para facilitar la convección y disipación del calor.
- Las fuentes de alimentación deberán ocupar una posición por encima del resto de componentes y en la parte superior del armario, ya que son generadores de calor.
- La unidad central ocupará una posición adyacente o por debajo de las fuentes de alimentación, en la zona superior del armario, quedando a una altura que facilite su inspección.
- Los racks de E/S estarán dispuestos de la forma más conveniente para el acceso y cableado, en el espacio libre.
- Se dejarán espacios suficientes entre los componentes y entre éstos y la envolvente para una adecuada disipación del calor.
- Para el resto de componentes del sistema, se recomienda su instalación en posiciones lo más alejadas del equipo que sea posible, principalmente si se trata de componentes electromecánicos, para minimizar las interferencias electromagnéticas.

➤ Cableado

Siempre que sea posible, en la configuración del sistema, se intentará agrupar los módulos por categorías en cuanto a entradas / salidas, tensión alterna o continua y señales discretas o analógicas.

Una configuración por grupos permite un cableado racional y una necesaria segregación de los cables de señal débil respecto a los que alimentan cargas, y de los de comunicaciones. Siempre que sea posible se separarán los cables de CC de los de CA, para minimizar las interferencias producidas por la conmutación de cargas y también los cables de interconexión de racks y de comunicaciones se separan completamente de otros.

➤ Puesta a tierra

Se seguirá lo especificado en la normativa vigente y las recomendaciones de los fabricantes, pero hay que recordar que cada una de las estructuras (racks) del autómatas, debe estar unida mediante un cable independiente de sección adecuada, a la pletina de tomas de tierra del armario. Nunca deben compartirse circuitos de tierra entre racks o con otros componentes del sistema.

➤ **Circuitos de seguridad**

Los dispositivos de parada de emergencia se instalarán con independencia del autómatas, para permitir la parada del sistema aún en caso de avería del mismo; en general, deben actuar sobre un contactor de potencia que corte la alimentación a las cargas de la instalación, e incluso que corte las señales de mando.

➤ **Circuitos de disposición de E / S**

En general, o por lo menos para los dispositivos de salida, es deseable que exista un contactor de maniobra o una protección que permita cortar la alimentación de esos elementos y que hará posible trabajar con seguridad en la puesta a punto o investigación de averías, con el autómatas alimentado.

➤ **Alimentación**

Se recomienda el empleo de transformadores separadores de alimentación ya que proporcionan una buena protección frente a interferencias introducidas en las líneas por la conmutación de cargas importantes existentes en la instalación. Además es deseable que los dispositivos de E/S se alimenten de la misma línea que el autómatas, ya que la fuente de alimentación del mismo posee circuitos de detección de nivel de tensión que provocan la secuencia de parada del equipo en caso de anomalía en la red, y de este modo se evitarán las falsas lecturas de señal de entrada.

Algunos autómatas incorporan una fuente auxiliar de 24 Vcc para uso externo de los dispositivos de entrada sobre módulos de entrada a 24 Vcc.

Hay que vigilar que no supere la capacidad de esta fuente, particularmente cuando se alimentan de ella dispositivos estáticos (detectores inductivos, fotoeléctricos, etc.) y deben seguirse las recomendaciones de cableado del fabricante para minimizar la posibilidad de interferencia sobre estos circuitos.

En caso de que se prevea la existencia de variaciones de tensión en la línea de alimentación que puedan superar los márgenes de trabajo especificados para el equipo, habrá que instalar transformadores estabilizadores, para evitar frecuentes paradas del sistema; en estas circunstancias es mejor alimentar las salidas del autómatas directamente desde la línea de entrada para descargar el transformador permitiendo que sea de una menor potencia.

➤ **Consideraciones sobre la instalación de E/S.**

Cuando se emplean dispositivos electrónicos de detección como elementos de entrada, hay que tener en cuenta la corriente residual de los mismos (detectores de 2 hilos de corriente alterna). En general, el problema se reduce a que el indicador de entrada se ilumina tenuemente, pero en ocasiones, cuando la corriente residual es elevada, o dependiendo de los umbrales de disparo del circuito de entrada pueden darse señales falsas.

Cuando los dispositivos de entrada trabajan a niveles de señal débil como analógicas, termopares, etc., hay que realizar conducciones de cableado separadas para evitar el problema de la inducción. Además, para evitar las interferencias electromagnéticas, se recomienda la instalación mediante cables trenzados y apantallados.

Los circuitos de salida controlan habitualmente cargas inductivas (solenoides), que provocan la aparición de picos de tensión cuando se interrumpe el circuito de alimentación (descarga del circuito inductivo). Estas crestas, que pueden alcanzar varios centenares de voltios, deben ser suprimidas, ya que pueden averiar los circuitos de salida (estáticos) y provocar interferencias en todo el sistema. Los fabricantes suelen incorporar supresores de transitorios en los circuitos de los módulos de salida pero a veces no son suficientes para evitar anomalías.

En general, los módulos de salida incorporan circuitos fusibles de protección dimensionados adecuadamente a las características nominales de la salida (transistor, triac); si no es así, hay que instalarlos en el exterior (regleta de bornes) teniendo en cuenta las especificaciones del fabricante ya que no protegerán adecuadamente la salida en caso de sobrecarga si no están bien dimensionados.

Puesta a punto

Una vez montado e instalado el equipo y cargado el programa en la memoria de la Unidad Central, hay que poner en marcha el sistema para comprobar que responde adecuadamente a la descripción de la tarea de control original, y en su caso, realizar las correcciones y mejoras oportunas.

Antes de dar alimentación, hay que hacer una serie de comprobaciones rutinarias pero importantes:

- Comprobar que todos los componentes del Autómata están en su lugar (el que corresponde a la configuración) perfectamente insertados en sus conectores y asegurados.
- Comprobar que la línea de alimentación está conectada a los correspondientes terminales de la fuente de alimentación del equipo, y que se distribuye adecuadamente a los módulos de entrada y salida (si procede).
- Verificar que los cables de interconexión entre racks están correctamente instalados.
- Verificar que los cables de conexión a periféricos están correctamente instalados.
- Verificar que las conexiones de los bornes de E/S están firmes y corresponden al esquema de eléctrico de la instalación y al esquema suministrado por el fabricante.

Previo al ensayo de funcionamiento según lo programado, hay que comprobar que los dispositivos de E/S funcionan correctamente:

- Con el equipo en PARO (selector de operación en posición STOP) aplicar tensión al sistema.
- Verificar que los indicadores de diagnóstico de la Unidad Central reflejan una situación correcta.
- Comprobar que los paros de emergencia actúan correctamente.
- Accionar los dispositivos de entrada manualmente y verificar que su estado es registrado por el autómata; el funcionamiento se puede seguir en los indicadores de los módulos y también se puede seguir visualizando la tabla de E/S mediante un equipo de programación.

Para la comprobación de los dispositivos de salida, hay que cortar la alimentación de las cargas que pudieran dar lugar a situaciones peligrosas y verificar con el procesador en MARCHA (selector de operación en posición RUN o RUN-P) que las salidas se activan. Esta comprobación resulta más fácil si se utiliza un terminal de programación en el modo "forzado de E/S" para activar o desactivar las salidas una a una.

Una vez finalizadas todas las comprobaciones anteriores, hay que introducir el programa en la memoria de la Unidad Central y dar alimentación al sistema. Se recomienda que siempre que sea posible, las pruebas de funcionamiento se hagan por áreas, particularmente si se trata de sistemas grandes, dejando fuera de servicio los componentes de las áreas que no se prueban; esto puede realizarse cortando la alimentación de campo de los racks de E/S o inhibiendo su funcionamiento, incluyendo las oportunas instrucciones en el programa que se eliminarán una vez concluidas las pruebas.

Verificadas y corregidas las distintas secuencias, el sistema puede arrancar en automático debiendo funcionar correctamente si todas las comprobaciones se han efectuado con éxito. Las correcciones efectuadas, tanto en la instalación como en el programa deben ser documentadas inmediatamente, y se obtendrán copias del programa definitivo tan pronto como sea posible.

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CAPÍTULO 3

**DESCRIPCIÓN
DEL SISTEMA**

3. DESCRIPCIÓN DEL SISTEMA.

3.1. Descripción de la instalación.

Características.

La instalación realizada está destinada al montaje y verificación de ordenadores a través de un sencillo sistema automatizado.

Descripción de los equipos.

La instalación está compuesta por dos líneas paralelas de montaje de idénticas características que funcionarán de forma independiente, por tanto la descripción de los equipos se realizará por línea.

El control de las dos líneas se realiza por el mismo cuadro de control, en el que se encuentran situados:

1. Seccionador, Interruptor magnetotérmico y protección diferencial generales de la instalación.
2. Fuente de alimentación de 24 Vcc para el mando.
3. Autómata.
4. Panel de operación.
5. Relé de seguridad, junto con relé de corte de mando y contactor de corte de potencia.
6. Protecciones de mando y de potencia.
7. Contactores de potencia.
8. Borneros de mando y de potencia.
9. Sistema de refrigeración del armario eléctrico.
10. Enchufes para terminal de programación e iluminación del cuadro de control.
11. Baliza de señalización del estado de la instalación.
12. Pulsadores de mando en la puerta para el control general de la instalación.

Cada línea comprende:

1. Un camino de cadenas motorizado en un nivel superior por el que circulan bandejas con carga.
2. Un camino de rodillos de gravedad en un nivel inferior destinado para el retorno de bandejas vacías.
3. Tres puestos de trabajo (1, 2 y 3) donde podemos encontrar:
 - Una cruz de giro para manipular la bandeja
 - Una botonera con un pulsador luminoso como interface hombre-máquina con el autómata
 - Un accionamiento neumático manual para elevar la cruz de giro.
 - Una estantería donde se sitúan los componentes de los ordenadores.
 - Un monitor, un ratón, un teclado y una fuente de alimentación ininterrumpida (S.A.I.) para comprobar los PCs.
4. Dos topes neumáticos automatizados a la salida de los puestos de trabajo 1 y 2 con el fin de controlar el flujo de bandejas y referenciar las mismas para poder manipularlas con la cruz de giro.
5. Un tope mecánico al final de la línea superior (puesto de trabajo 3) con el fin de referenciar la bandeja para que pueda ser manipulada por la cruz de giro.
6. Dos topes dosificadores entre los puestos 1 y 2, y los puestos 2 y 3 para dosificar el paso de bandejas a los puestos.
7. Una parada de emergencia general.

Descripción de los aparatos de diálogo Hombre-máquina.

La instalación comprende:

- Puerta armario X0.
- Botonera X1B1 en puesto 1A.
- Botonera X1B2 parada de emergencia.
- Botonera X1B3 en puesto 2A.
- Botonera X1B4 en puesto 3A.
- Botonera X1B5 en puesto 1B.
- Botonera X1B6 en puesto 2B.
- Botonera X1B7 parada de emergencia.
- Botonera X1B8 en puesto 3B.

A continuación se muestran unas imágenes de los elementos listados anteriormente:

- Puerta del armario:



- Botoneras X1B1, X1B3, X1B4, X1B5, X1B6 y X1B8:
La figura muestra la botonera X1B3, el resto son iguales entre si, tan solo cambia el texto del pulsador.



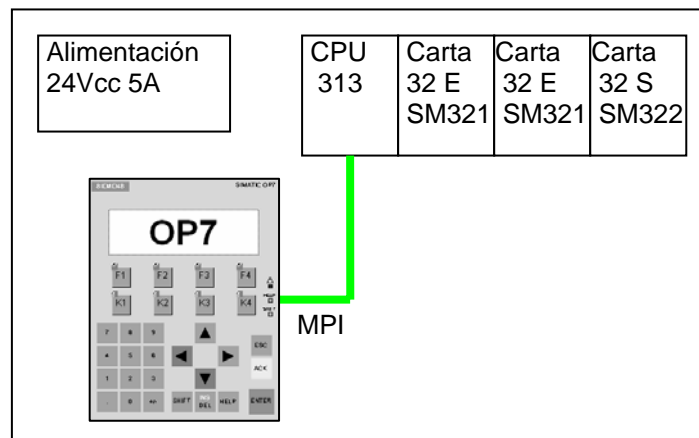
- Botoneras X1B2 y X1B7.



Arquitectura de los aparatos de mando.

La gestión del proceso está controlada por un autómata SIEMENS S7 que comprende:

- 1 fuente alimentación 24 Vcc (5A): 6EP1 333-2AA00
- 1 raíl de montaje ref.: 6ES7 390-1AF30-0AA0.
- 1 procesador CPU 313 ref.: 6ES7 313-1AD03-0AB0.
- 2 cartas 32 entradas SM321 ref.: 6ES7 321-1BL00-0AA0.
- 1 carta 32 salidas SM322 ref.: 6ES7 322-1BL00-0AA0.
- 1 Panel de operador OP7 ref.: 6AV3607-1JC20-0AX1



El único interfase utilizado en este proyecto es la conexión de un terminal de operador OP7 de Siemens conectado a la red MPI como se puede ver en la imagen anterior.

Desde el panel de operador se podrán seleccionar el modo de funcionamiento para las diferentes zonas de la instalación, visualizar las alarmas de la instalación y realizar los movimientos manuales para operaciones de mantenimiento.

Descripción de las seguridades.

Lista de los dispositivos de seguridad.

- La protección del personal.
 - Al nivel de armario eléctrico.
 - Interruptor seccionador y diferencial para fuerza motriz.
 - Interruptor seccionador para 230 Vac.
 - Interruptor seccionador alimentación 24 Vcc entradas.
 - Interruptor seccionador alimentación 24 Vcc salidas.
 - Parada de emergencia.
 - Relé de seguridad Pilz.

- Entorno de máquina.
 - Dos botoneras con una seta de emergencia, seriadas con el relé de emergencia ubicado en el armario eléctrico.
- La protección del material.
 - Al nivel de armario eléctrico
 - Disyuntores de protección para los motores.
 - Enclavamiento eléctrico de motores.
 - Interruptores seccionadores para las fuentes de alimentación ubicadas en los puestos de trabajo.
 - Al nivel de las zonas funcionales.
 - Detectores de presencia y parada.
 - Topes de control de flujo.

Tratamiento de seguridades.

- Los disyuntores de protección de los motores.
 - Un contacto normalmente abierto para una entrada de autómata.
- En funcionamiento normal, durante una parada de emergencia:
 - La alimentación de los motores es cortada.
 - La alimentación del mando es cortada.
 - Los motores efectúan un frenado inmediato.
 - El autómata de gestión de la instalación corta también la alimentación de 24 Vcc para mandos de relés.

3.2. Marcha/parada del sistema.

Puesta en marcha del sistema.

En primer lugar se debe verificar que el armario eléctrico está en tensión, lo podemos ver en la lámpara blanca situada en la puerta del armario. Si el armario está en tensión la lámpara blanca estará encendida, en caso contrario, debemos poner el armario en tensión mediante el seccionador ubicado a la derecha del armario.



Una vez tenemos el armario en tensión presionar el pulsador de MARCHA situado en el panel, el sistema arranca pasado unos segundos, durante los cuales sonará de forma continua la sirena y parpadeará la baliza verde y el pulsador luminoso de marcha general, con el fin de indicar el arranque de la máquina.



Pasado el periodo de arranque la baliza verde y el pulsador luminoso de marcha general se iluminarán de forma fija. Si se pulsa una emergencia este pulsador parpadeará indicando que se ha interrumpido el funcionamiento del sistema pero que tiene marcha general.

Si el sistema no se pone en marcha, piloto verde parpadeando o baliza roja iluminada, revisar que no este pulsada alguna parada de emergencia. (Se han instalado varias paradas de emergencias en la instalación, una en el armario principal, y una situada en cada una de las botoneras instaladas por cada línea).

Si no arranca puede ser por defecto del relé de seguridad a través del que actúan las emergencias o por falta de presión de aire en la instalación, revisar los textos de alarmas activas en el panel de operador OP7.

Comprobar que todas las zonas se encuentran en automático, si la baliza roja está intermitente alguna está en manual y no arrancará hasta pasarla a automático.

Nota: Si el sistema arranca sin tensión, por defecto el relé de seguridad estará sin rearmar, para rearmar el mismo se debe presionar el pulsador de rearme situado en la puerta del armario principal.



Parada del sistema.

Presionar el pulsador de PARO situado en la puerta del armario eléctrico, automáticamente la instalación se parará si el sistema está en el estado de reposo y no responderá a las actuaciones programadas en modo automático, desactivando todas las salidas e ignorando las señales de entrada.



Si el sistema no está en reposo espera a terminar todos los trabajos en curso, y una vez realizado esto se parará. Si se presiona el pulsador de marcha durante la secuencia de paro, ésta se interrumpirá y el sistema continuará normalmente.

Si se vuelve a arrancar la instalación, y no se ha manipulado el estado de las piezas en los distintos puestos, todo continuará normalmente, en caso contrario, es decir, se han manipulado algunas piezas, bien retirándolas o bien introduciendo otras manualmente, debido a que se memorizan los estados de la instalación, se podrían realizar respuestas erróneas.

3.3. Descripción de los modos de operación.

Para esta instalación se han previsto dos modos de operación:

- Modo automático.

Este es el modo de funcionamiento normal de la instalación. Con este modo seleccionado los elementos de la instalación se mueven automáticamente cuando es necesario.

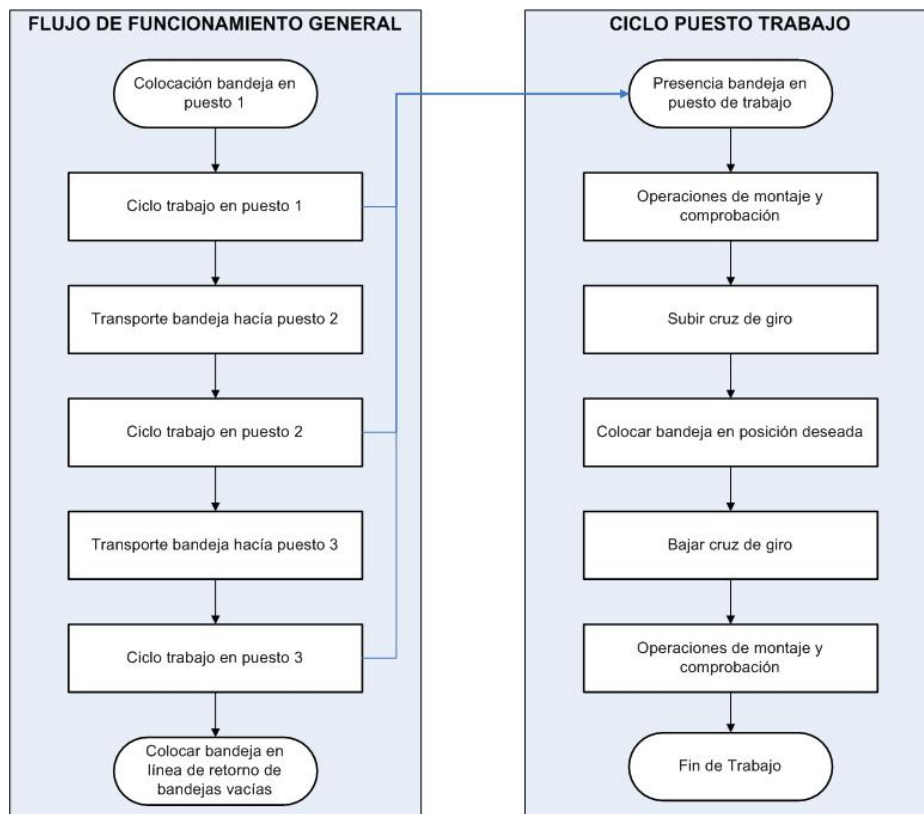
- Modo manual.

El modo manual está previsto para realizar las operaciones de mantenimiento en la instalación y poder verificar el correcto funcionamiento de los diferentes elementos de la instalación, todo ellos pueden ser accionados manualmente desde el terminal de operador OP7 dispuesto en el armario principal.

3.4. Descripción de Funcionamiento.

Como ya se había comentado con anterioridad en el capítulo 1, la instalación está dividida en zonas de funcionamiento independientes (Línea A y Línea B). El funcionamiento de ambas es exactamente igual, por tanto de aquí en adelante hablaremos una sola línea (A).

En la siguiente figura se muestra un diagrama de flujo con el funcionamiento general de la línea.



El flujo de funcionamiento completo de la línea debería ser el siguiente:

1. El operario coge una bandeja vacía del nivel inferior (retorno de bandejas vacías) y la coloca sobre el puesto de trabajo 1.
2. Cuando el inductivo detecte la bandeja en posición de trabajo (bandeja referenciada contra el tope, inductivo B132), se realiza la elevación de la cruz de giro +CGA1 (operación manual por el operario) para poder realizar la manipulación y las operaciones de montaje necesarias.
3. Cuando el operario baja la cruz de giro el pulsador de fin de trabajo en puesto 1 parpadea indicando que espera que se pulse el fin de trabajo.
4. El operario presiona el pulsador de fin de trabajo en puesto 1 indicando que ha acabado la manipulación del equipo en el puesto, la lámpara del pulsador se encenderá de forma fija indicando que el PLC se ha enterado de la situación.
5. Si el camino tras el puesto 1 está libre (inductivo B133 o B134 libres) el tope +TNA01 baja dejando salir la bandeja hacia el puesto 2.
6. Si el puesto de trabajo 2 está libre (inductivo B135) el tope dosificador +TNA02 deja pasar una bandeja hacia el puesto.
7. Cuando la bandeja está en el puesto 2 (inductivo B135) se realiza la elevación de la cruz de giro +CGA2 para poder realizar la manipulación y las operaciones de montaje necesarias.
8. Cuando el operario baja la cruz de giro el pulsador de fin de trabajo en puesto 2 parpadea indicando que espera que se pulse el fin de trabajo.
9. El operario presiona el pulsador de fin de trabajo en puesto 2 indicando que ha acabado la manipulación del equipo en el puesto, la lámpara del pulsador se encenderá de forma fija indicando que el PLC se ha enterado de la situación.
10. Si el camino tras el puesto 2 está libre (inductivo B136 o B137 libres) el tope +TNA04 baja dejando salir la bandeja hacia el puesto 3.
11. Si el puesto de trabajo 3 está libre (inductivo B140) el tope dosificador +TNA05 deja pasar una bandeja hacia el puesto.
12. Cuando la bandeja está en el puesto 3 (inductivo B140) se realiza la elevación de la cruz de giro +CGA3 para poder realizar la manipulación y las operaciones de montaje necesarias.
13. Cuando el operario acaba las operaciones de montaje debe retirar la bandeja colocándola en la línea de retorno de bandejas vacías y bajar la cruz de giro. El pulsador de fin de trabajo en puesto 3 parpadea indicando que espera que se pulse el fin de trabajo al detectar la cruz de giro abajo.
14. El operario presiona el pulsador de fin de trabajo en puesto 3 indicando que ha acabado la manipulación del equipo en el puesto y el puesto queda libre para la recepción de una nueva bandeja.

Con el fin de hacer la instalación lo más flexible posible se ha pensado para que pueda trabajar una sola persona o varias, dependiendo en cada momento de las necesidades de la empresa. Por este motivo todos los puestos de trabajo son exactamente iguales, pudiéndose realizar la mismas operaciones en todos.

El operario puede terminar la labor de montaje y verificación del equipo en cualquier puesto de trabajo, sin ser necesariamente en el último (puesto 3). De esta forma si el operario acaba el montaje del ordenador puede retirar el equipo de la bandeja y colocarla en la línea de retorno de bandejas vacías. Al presionar el pulsador de fin de trabajo el puesto quedará libre esperando otra bandeja.

Si estuvieran trabajando varios operarios en la máquina y uno de ellos tuviera un problema en un puesto de trabajo podría dejar sin bandejas a los otros operarios. Esta situación se puede solventar de una forma sencilla, ya que en cualquier momento un operario podría coger una bandeja vacía y colocarla en su puesto de trabajo, sin necesidad de esperar a que le lleguen bandejas de los puestos anteriores.

3.5. Sistemas de alarmas y avisos.

Se ha diseñado un sistema de alarmas y avisos para advertir al operador de las posibles anomalías que se produzcan en la instalación.

Este sistema será gestionado por el autómata y se utilizará el panel de operador para mostrar el texto de la alarma correspondiente.

Cuando una alarma o aviso se dispara el piloto rojo de la baliza del armario central luce de forma intermitente. El texto de la alarma activada aparecerá parpadeando reflejado en el terminal de operador OP7, situado en el cuadro principal. También parpadeará el led rojo de la OP7 indicando alarma activa no aceptada. Como aviso acústico la bocina sonará de manera intermitente hasta que se acepte la alarma.

Se debe consultar la alarma y después presionar el botón "ACK" del terminal de operador. De esta manera la bocina dejará de sonar, el texto del terminal de operador dejará de parpadear y se mantendrá fijo y el led rojo se quedará fijo hasta que se rearme dicha alarma.

Una vez solucionado el problema se debe resetear la alarma, para ello, se pulsará el botón de rearme situado en el armario.

Estando en la pantalla inicial del terminal, podemos visualizar las alarmas actuales presionando la tecla ESC. Si existen varias, podemos desplazarnos para visualizarlas con las teclas "▲" o "▼". Para volver a la pantalla principal presionar la tecla ENTER.

Ayudas para resolver problemas.



Texto de alarma en terminal	Solución
RELE DE SEGURIDAD NO OK	Comprobar el origen de la alarma. El relé de seguridad general no está rearmado. Desenclavar emergencias, y pulsar rearme del armario principal.
EMERGENCIA CUADRO	Desenclavar la emergencia del cuadro descrita en la alarma y pulsar rearme del armario principal.
EMERGENCIA BOTONERA XXBX	Desenclavar la emergencia de la botonera descrita en la alarma y pulsar rearme del armario principal.
DEFECTO TÉRMICO XXXXXX	Comprobar el térmico que dio origen a la alarma en el cuadro principal, rearmarlo y pulsar rearme de la instalación en el armario principal.
FALLO BATERÍA DE LA CPU	Defecto en el nivel de carga de la batería de la CPU. Cambiar la batería y rearmar en el armario principal.
DEFECTO PRESIÓN DE AIRE EN LA INSTALACIÓN	Defecto en la presión de aire en la instalación. Solucionar el problema y rearmar la alarma.
FALLO EN TRANSFERENCIA TNAXX>>>TNAXX ZONA X	Verificar que no hay atasco o si lo hay solucionarlo y rearmar la alarma. Acusar y rearmar la alarma.
ATASCO EN DETECTOR BXXX ZONA X	Fallo o atasco en el detector indicado. Solucionar el problema y rearmar en el armario principal.
DEFECTO MOVIMIENTO TOPE TNAXX ZONA X.	Fallo en el movimiento del tope indicado. Solucionar el problema y rearmar la alarma.
DEFECTO POSICIÓN CRUZ DE GIRO CGAX ZONA X.	Fallo provocado por mal posicionamiento de la cruz de giro indicada. Solucionar el problema y rearmar la alarma.



3.6. Sistemas de señalización.

Balizas

En la parte superior del armario eléctrico hay dos balizas, una verde de marcha y otra roja para avisar que existe una alarma u otras emergencias generales. En el siguiente cuadro se muestran los posibles estados de la baliza verde y roja:

BALIZA VERDE	BALIZA ROJA	DESCRIPCIÓN
Apagado	Apagado	Sistema parado.
Encendido fijo	Apagado	Sistema en marcha.
Encendido parpadeando a Frecuencia 250 ms.	-----	La instalación está en proceso de arranque.
Encendido	Encendido parpadeando a Frecuencia 1 s.	Aviso de avería o la instalación en manual.
Encendido	Encendido fijo	Parada de emergencia general activada, no rearmada o alarma general.
Encendido parpadeando a Frecuencia 1 ms.	-----	La instalación está en proceso de parada.

3.7. Operaciones de mantenimiento.

Se ha previsto un modo de funcionamiento manual que se controlará desde el panel de operador OP7 tal y como se ha comentado con anterioridad para realizar las operaciones de mantenimiento.

A continuación se muestra una lista con los movimientos manuales que se pueden realizar en la instalación:

Número de Movimiento	Zona 1 (Línea A)	Zona 2 (Línea B)
1	MARCHA MOTOR M11	MARCHA MOTOR M12
2	BAJAR TOPE TNA1	BAJAR TOPE TNA7
3	BAJAR TOPE TNA2	BAJAR TOPE TNA8
4	BAJAR TOPE TNA4	BAJAR TOPE TNA10
5	BAJAR TOPE TNA5	BAJAR TOPE TNA11



Además de los movimientos manuales hay un pulsador de prueba lámparas en el armario principal previsto para que el personal de mantenimiento pueda chequear el funcionamiento de todos los elementos luminosos de la instalación.

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CAPÍTULO 4

**DISEÑO DEL
SISTEMA**

4. DISEÑO DEL SISTEMA.

4.1. Introducción.

En primer lugar antes de continuar con el presente capítulo destacar que como se había mencionado anteriormente en el apartado 1.2 (origen del proyecto) el alcance de este proyecto es realizar el sistema de control y gestión mediante el autómatas y el panel de control, y partiendo de un proyecto eléctrico realizado por la empresa mecánica que nos subcontrata, en el cual ya están definidos los elementos de la instalación.

No obstante, elegir los elementos que compondrán el sistema de control de una instalación es algo importante a la hora de realizar un proyecto de este tipo, y por ello nos encontramos en este capítulo.

Antes de continuar con las pautas o puntos en los que debemos prestar especial atención a la hora de decantarnos por los elementos a instalar en el sistema, hay que destacar que debido a restricciones del cliente final (empresa que compra la instalación) o en este caso de un cliente intermedio (empresa que vende la instalación y nos subcontrata) los materiales que se eligen deben ser de determinadas marcas.

Este tipo de restricciones en la elección de materiales para las instalaciones son algo habitual debido a que las empresas ya constituidas cuentan con multitud de instalaciones automáticas, y de cara a llevar a cabo el mantenimiento de las mismas se deben de unificar los materiales utilizados en ellas. Además, en la actualidad los departamentos de mantenimiento tienen personal con formación en autómatas programables y otros sistemas de control (como variadores de frecuencia, etc.), debido a que los propios fabricantes imparten cursos gratuitamente a las grandes empresas con el fin de abrirse mercado.

En nuestro caso, los elementos del sistema de control vienen fijados por la empresa que nos subcontrata, ya que esta empresa cuenta con un departamento de Software que habitualmente trabaja con sistema de automatización de SIEMENS, y siempre que el cliente final lo permita, se utilizarán elementos de esta marca en concreto.

4.2. Selección del Autómata.

Debido a la amplia oferta del mercado surge la necesidad de seleccionar el equipo más adecuado. La decisión debe basarse en un análisis sistemático de una serie de factores, pero considerando no sólo las características actuales de la tarea de control, sino también las necesidades futuras en función de los objetivos de la empresa.

El análisis se divide en *factores cuantitativos* y *factores cualitativos*.

Factores cuantitativos

Se refieren a la capacidad del equipo para soportar todas aquellas especificadas para el sistema de control y se pueden agrupar en las siguientes categorías.

- *Entradas/Salidas (E/S)*: cantidad, tipo, prestaciones, etc.
- *Tipo de control*: una o varias máquinas, proceso, etc.
- *Memoria*: cantidad, tecnología, expansibilidad, etc.
- *Software*: conjunto de instrucciones, módulos de programa, etc.
- *Periféricos*: equipos de programación, diálogo hombre-máquina, etc.
- *Físicos y ambientales*: características constructivas, temperatura, etc.

Entradas/Salidas (E/S)

El primer trabajo a realizar al iniciar la implementación del sistema de control es determinar la cantidad de entradas y salidas que debe ser capaz de tratar el equipo. Debemos contar el número de dispositivos cuyo estado hay que leer o gobernar, y una vez obtenidas estas cantidades es recomendable reservar espacio para futuras ampliaciones (entre un 10% y un 20%). En ocasiones el porcentaje de entradas y salidas de reserva los fija el cliente en sus especificaciones.

Los fabricantes ofrecen una gran diversidad de soluciones en cuanto a las características constructivas y funcionales de los elementos del sistema de entrada/salida. Por lo que además de saber el número de Entradas/Salidas que necesitamos en la instalación debemos elegir el tipo de tarjetas que utilizaremos según las características de nuestro sistema.

En el caso de señales de entrada discretas hay que procurar que las entradas incorporen filtros para evitar lecturas falsas en caso de señales erróneas (p.ej. rebote de un contacto).

Para las salidas discretas es conveniente que incorporen una protección de sobrecarga, que en caso de ser un fusible, es mejor que sea de acceso frontal (evitará dejar fuera de servicio todas las salidas del módulo al retirarlo para cambiar el fusible), y además es deseable que incorporen un indicador de fusible fundido para su inmediata localización.

En cuanto a las E/S de señal analógica se encuentran las adecuadas para el tratamiento de señales, procedentes de instrumentación de campo (caudal, temperatura, presión, etc.) y para la regulación (variación de velocidad, válvulas motorizadas, etc.) con las bandas de trabajo más usuales. En las características del módulo, el fabricante debe especificar los parámetros de precisión de la conversión.

En el caso de grandes sistemas de control, es de particular importancia disponer de E/S remotas. La ubicación de estructuras de E/S junto a los elementos de campo y unidas a la Unidad Central con un solo cable de comunicaciones (bus de campo), disminuye los costes de cableado, tanto en material como en trabajo de instalación. Por otra parte, facilita las tareas de puesta a punto y mantenimiento, ya que se pueden realizar por aéreas funcionales sin afectar al funcionamiento del resto del sistema.

En nuestro caso, **el sistema estará compuesto únicamente por entradas/salidas discretas**, concretamente tiene 2 tarjetas de 32 entradas digitales (64 en total, de las cuales hay un 26% de reserva) y una tarjeta de 32 salidas digitales (de las cuales hay un 34% de reserva).

Tipo de control

En aplicaciones en las que se pretende el control de varias áreas o máquinas independientes, pero con funciones autónomas, se plantea el problema de optar por un control centralizado o distribuido.

La importancia de dichas funciones por si solas, o la posibilidad de subdividir la tarea de control del proceso o conjunto de máquinas en esas funciones autónomas determinará en muchos casos la elección de uno u otro tipo de control.

El control centralizado presenta el inconveniente de que si el autómatas falla se produce una parada total de la instalación, por tanto en este tipo de sistemas si la disponibilidad de funcionamiento es lo bastante importante se opta por utilizar unidades redundantes. Una unidad está en activo controlando las entradas y salidas, mientras la otra está en la reserva, de forma que si la primera deja de funcionar la segunda asume el control de las entradas y salidas.

El control distribuido requiere que puedan considerarse máquinas o grupos de máquinas o áreas funcionales del proceso susceptibles de ser definidas por un algoritmo de control. A cada una de ellas se destinará un autómatas dimensionado de acuerdo con los requerimientos de aquella área.

Debido a la dependencia que puede existir entre las operaciones que tienen lugar en cada área, hay que tener en cuenta que puede ser necesario interconectar los autómatas entre si a través de una red de comunicaciones, por tanto el autómatas utilizado debe permitir las comunicaciones.

En nuestro caso el **sistema es centralizado** con un sólo autómata. El principal motivo de la elección de este sistema es el económico, ya que para el cliente final no es tan fundamental la disponibilidad de funcionamiento como para afrontar la diferencia económica que supondría un sistema centralizado redundante o un sistema descentralizado para controlar las dos zonas funcionales de la instalación.

Memoria.

En cuanto a la memoria es necesario considerar dos características principales: tamaño y tipo de la memoria.

Las unidades centrales suelen incorporar una memoria de tamaño acorde con su capacidad de control y la potencia del conjunto de instrucciones con las que opera. Aunque para adaptarse mejor a cada aplicación por razones económicas los fabricantes suelen sacar al mercado en un mismo equipo distintas opciones de cantidad de memoria. En cualquier caso la posibilidad de expansión de memoria futura suele existir para no encontrarse con la necesidad de sustituir toda la unidad.

No existe una regla fija para la evaluación de la cantidad de memoria necesaria para una determinada aplicación, aunque existen ciertas fórmulas de aproximación (p.ej. multiplicar el número de E/S por un factor entre 5 y 10). El valor obtenido debe ser considerablemente incrementado en el caso de que el programa incluya cálculos de cierta complejidad (p.ej. con variables numéricas y datos la fórmula de aproximación se obtendría multiplicando el total de variables numéricas por un factor entre 15 y 30).

Al igual que ocurre con las E/S, para la memoria es recomendable dejar un espacio adicional de reserva. Ya que debido a la flexibilidad del autómata es frecuente que el usuario una vez resuelto el problema de control se plantee obtener determinada información del proceso u optimizar alguna de las operaciones del proceso.

El tipo de la memoria empleada dependerá de la aplicación concreta:

- Memoria RAM o Flash EPROM. Ideal para aplicaciones en que es necesario introducir cambios en el funcionamiento con cierta frecuencia con la posibilidad de que el proceso no se detenga.
- Memoria EPROM o EEPROM. Proporciona un medio muy fiable de almacenamiento del programa. Ideal para empresas que se dedican a realizar máquinas estándar que tienen siempre el mismo funcionamiento.

En algunos equipos se ofrece la posibilidad de disponer de ambos tipos de memoria (permanente y volátil), de forma que el usuario tiene la posibilidad de modificar con facilidad el funcionamiento de la instalación.

En nuestro caso el autómata **Siemens S7-313 tiene una memoria RAM de 32 Kbytes**. En el anexo 1 podemos ver con más detalle las características técnicas de esta CPU.

Software.

Una vez está claro y definido el funcionamiento de la instalación, el programador tendrá una referencia clara del tipo de instrucciones que podrán ser necesarias para programar las secuencias lógicas definidas, incluso para aquellas funciones especiales que no afectan directamente al proceso, como son cálculos y tratamientos de datos, comunicaciones, regulación, etc., que requieren de instrucciones especiales. Un conjunto de instrucciones potente facilitará la tarea de programación, y en general reducirá el tiempo de respuesta del sistema, además del tiempo de programación.

También hay que considerar las instrucciones que permiten el control del ciclo de ejecución, la posibilidad de organización del programa en módulos funcionales para facilitar la organización del programa y la existencia de una librería de funciones preprogramadas suministrada por el fabricante, en la que simplemente con personalizar parámetros y direcciones de variables se puedan emplear en nuestro propio programa.

En nuestro caso el autómatas se programa con el software “**SIMATIC STEP 7**”, el cual permite la utilización de módulos de organización, módulos de función, funciones, módulos de datos, tipos de datos y tablas de variables. Además el fabricante proporciona multitud de librerías de funciones estándar para movimiento de datos, regulaciones PID, etc.

Periféricos.

Los fabricantes ofrecen distintos niveles de equipos de programación, cuya utilidad depende del tipo de empleo a que se destinen. Así los pequeños terminales tipo calculadora son de gran utilidad y económicos cuando se emplean como unidad de monitorización y para muy pequeñas modificaciones, sin embargo trabajar con ellos en programas complejos resulta poco útil y engorroso. Existen también consolas con pantalla CRT, que aportan mayor comodidad así como un gran número de opciones de interconexión a otros periféricos, como son impresoras o unidades de cintas, y algunos de ellos incluso permiten la programación off-line del autómatas.

Actualmente se ofrecen elementos de programación mediante software para PC, lo que da la posibilidad de tener en un solo equipo de programación el software de diferentes fabricantes a un coste aceptable, y si además añadimos la posibilidad de ordenadores portátiles entendemos el porqué de que hayan desplazado a las consolas de programación.

Respecto a otros periféricos en cada caso hay que comprobar que el autómatas permite la interconexión a los que se considere formen parte del sistema de control: impresoras, monitores, visualizadores, unidades de disco, paneles de control, etc.

En nuestro caso el sistema se programa con un **PC portátil** y lleva asociado un panel de control **Siemens OP7**.

Físicos y ambientales.

Las características en cuanto a los materiales empleados, formas de presentación y dimensiones, deben ser analizadas en función de las condiciones mecánicas de la aplicación (p.ej. aspectos como la forma de realizar el conexionado de los dispositivos de E/S pueden ser importantes en relación al personal que debe realizar la instalación). En cada caso hay que valorar las condiciones ambientales de la instalación, polvo, humedad, temperatura y considerar la necesidad de tomar precauciones al respecto (p.ej. climatizar el armario).

Todos los fabricantes realizan una serie de pruebas, cuyos resultados se reflejan en las características técnicas de los equipos: banda de temperatura de trabajo y almacenaje, vibración soportada, nivel de interferencia, etc.

En nuestro caso pese a que el autómata trabaja adecuadamente en las condiciones ambientales de la nave donde se realiza la instalación se ha dotado al armario eléctrico de un **ventilador** para la refrigeración del mismo.

Factores cualitativos.

En muchas ocasiones la decisión se basa en criterios comerciales y en general limitados al aspecto económico de la adquisición.

Una vez evaluados los factores correspondientes a las características técnicas y constructivas de los componentes del autómata y equipos periféricos, hay que tener en cuenta otros aspectos que en definitiva tendrán una mayor influencia a medio plazo.

- Ayudas al desarrollo del programa
- Fiabilidad del producto
- Servicios del suministrador
- Normalización en planta

Ayudas al desarrollo del programa.

Son las proporcionadas por las herramientas de programación ofrecidas por los fabricantes. En un entorno de programación hay que valorar ciertas características que pueden facilitar el desarrollo del programa:

- Detección de errores de sintaxis de la programación en curso.
- Potencia de los mandatos de edición y modificación.
- Listado de referencias cruzadas.
- Información de ocupación de memoria (tanto E/S como memoria interna).

- Lenguajes de programación ofrecidos por el entorno (hay fabricantes que venden sus software de programación con un lenguaje en una versión básica, y añadir otros lenguajes tiene un coste adicional).
- Posibilidades de documentación del programa (simbólicos, comentarios, etc.).
- Ayuda del entorno de programación y listado de instrucciones.
- Posibilidad de simular. La mayoría de los fabricantes ofrecen un simulador de autómatas con el fin de poder probar el programa sin necesidad de tener el autómata. Esta herramienta es fundamental cuando en el autómata se trabaja con instrucciones complejas (p.ej. tablas de datos, etc.)

En las fases de puesta a punto es necesario disponer de la opción de visualización y forzado de estado, tanto en E/S como en variables de trabajo internas. Ésta opción permite determinar el estado o valor de las variables independientemente del que le corresponda por el desarrollo del programa. La posibilidad de forzar es una gran ayuda, ya que nos permite probar todas las salidas de la instalación de forma independiente.

Estas ayudas reducen los tiempos de programación, y puesta a punto que constituyen siempre un coste significativo en los sistemas programables.

Fiabilidad del producto.

Este es un factor de particular importancia, si tenemos en cuenta que una falta de fiabilidad se traduce directamente en tiempos de parada y por tanto, costes de producción.

Un indicador de la fiabilidad lo constituyen los parámetros del tiempo medio entre fallos, que el fabricante debe proporcionar, al igual que otros datos, como su control de calidad en curso de fabricación.

Otro indicador lo constituyen las experiencias de otros usuarios y la existencia de otras instalaciones similares en las que el equipo ha probado su valía. Es por esto, que normalmente una empresa con cierta experiencia en instalaciones automáticas cuando contrata una instalación suele restringir la elección de componentes a ciertas marcas con las que tiene experiencia.

Servicios del suministrador.

Ya en los primeros contactos con los potenciales suministradores se puede obtener una idea de la capacidad y voluntad del soporte técnico que el vendedor está en condiciones de ofrecer. Es particularmente indicativa la respuesta que el suministrador es capaz de dar a cuestiones preliminares, que no se hallan de forma explícita en los catálogos.

La formación del personal de programación o de mantenimiento, es un punto importante, ya que cuanto mejor se conozcan y comprendan las características y prestaciones del equipo, mayor rendimiento se obtendrá de él.

Los cursos suelen contratarse, aunque en ocasiones se ofrecen gratis con la compra del equipo. También suelen ofrecer cursos gratuitos cuando lanzan un nuevo autómata para dar a conocer el producto.

Una vez instalado el equipo, adquiere gran importancia la asistencia técnica. Esta asistencia puede abarcar desde un servicio de consulta telefónica hasta un técnico de servicio en el lugar de la instalación.

Otro elemento de gran valía es la disponibilidad de una buena información técnica que cubra tanto los aspectos de instalación y programación como el mantenimiento. La información debe estar bien organizada, debe ser clara y con ilustraciones de calidad y es deseable que incluya ejemplos y notas de aplicación (p.ej. para una mejor explotación de las instrucciones de programación).

Por último, pero no menos importante, hay que considerar la disponibilidad del producto y de recambios. La disponibilidad de recambios en un tiempo mínimo es fundamental, ya que aunque deben existir recambios en el almacén de mantenimiento, no es posible ni rentable cubrir la totalidad de componentes.

Normalización en planta

Pueden considerarse dos posturas respecto a la normalización de un determinado fabricante para cubrir todas las necesidades de empleo de autómatas en una empresa:

1. Actualmente, los fabricantes ofrecen familias de productos compatibles entre si que cubren todas las necesidades, desde pequeños sistemas hasta aquellos capaces de controlar miles de E/S, que pueden comunicar a través de redes locales, y se configuran empleando componentes de E/S comunes a todos los modelos. En este aspecto la adopción de una sola marca cubre todas las necesidades y presenta las siguientes ventajas:
 - La formación del personal respecto a nuevos componentes es simplemente una ampliación de conocimientos previos.
 - Se reduce el stock de piezas de recambio.
 - Una unidad que se amplíe, en todo caso requerirá simplemente el cambio de la unidad central por otra más potente o añadir más memoria a la existente y reprogramar.
2. Por otra parte existe un factor de riesgo en la dependencia única de un solo suministrador. Además, los productos de distintos fabricantes no siempre son intercambiables, lo que complica la situación. Hay que pensar con más de una marca de forma que el

personal técnico esté formado en el empleo de un tipo de equipo, y tener una alternativa cada vez que se presenta una nueva aplicación.

Las desventajas originales que representa tener que conocer dos o tres sistemas distintos, probablemente con lenguajes distintos y equipos de programación propios, van desapareciendo por la mayor similitud de los equipos actuales, las opciones de programación a través del ordenador, la disponibilidad de módulos de interconexión entre unidades de distintos fabricantes y la utilización de redes locales.

4.3. Selección del Panel de Operador.

En la actualidad casi todas las instalaciones por pequeñas que sean requieren la existencia de un terminal de operador como interface hombre-máquina entre el operador y el autómata.

La mayoría de los paneles del mercado permiten su interconexión con autómatas de diferentes fabricantes, no obstante es preferible que el panel de operador sea del mismo fabricante que el autómata, ya que de este modo evitaremos problemas a la hora de realizar la interconexión entre ambos dispositivos, obtendremos mejor documentación y asistencia técnica, al depender de un solo fabricante. Además de las ventajas anteriores hay que sumar que según sea el tamaño del pedido a un fabricante el descuento que se obtiene en la compra puede ser más favorable.

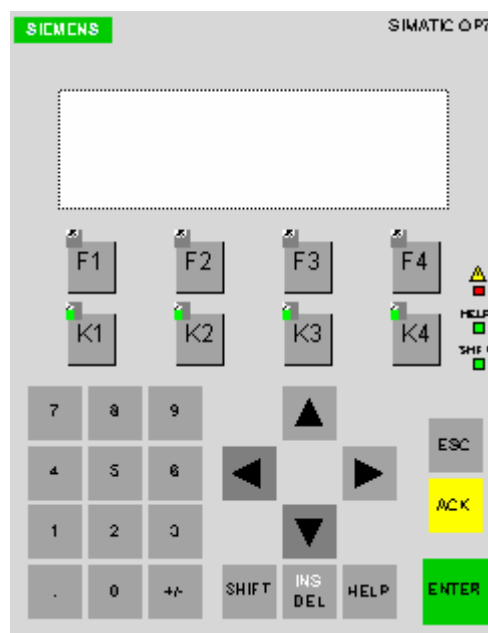
La elección del panel de operador también es un punto importante, ya que existen multitud de paneles diferentes dentro de un mismo fabricante, e incluso otras opciones más potentes que los paneles de operador según las necesidades de la instalación. Por ejemplo SIEMENS ofrece las siguientes soluciones como interface Hombre-Máquina (SIMATIC HMI):

- *SIMATIC WinCC.* Sistema SCADA para supervisión de procesos escalable para cualquier requisito, del sistema monousuario al sistema multiusuario redundante así como posibilidad de supervisión y manejo de plantas por Internet.
- *SIMATIC Push Button Panels.* Paneles de operador con conectividad por bus, preconfeccionados listos para su conexión, para mando simple y directo de máquinas.
- *SIMATIC Micro Panels.* Paneles para máquinas de pequeño tamaño, especialmente para aplicaciones asociadas al SIMATIC S7-200.
- *SIMATIC Mobile Panels.* Paneles móviles para el mando directo de instalaciones y máquinas desde cualquier punto.
- *SIMATIC Panel PC.* Las plataformas industriales para la visualización local en PC para variadas tareas de automatización basadas en PC.

- *SIMATIC Panels.* La solución compacta y robusta para su aplicación directa en la máquina, gama finamente escalonada en cuanto a prestaciones y comodidad, presentándose en las versiones visualizadores de textos (Text Displays), paneles de operador y paneles táctiles.
- *SIMATIC Multi Panels.* Plataformas multifuncionales que además de las tareas de visualización ejecutan otras tareas de automatización como por ejemplo funciones de control.
- *SIMATIC ProTool.* El software de configuración unificado y homogéneo bajo Windows para todos los paneles SIMATIC HMI.
- *SIMATIC ProTool/Pro.* Este software de visualización para manejo y visualización basado en PC a nivel de máquina permite breves tiempos de reacción y una conducción segura del proceso.
- *SIMATIC WinCC flexible.* El nuevo software HMI homogéneo y flexible para todos los paneles de operador de pantalla gráfica, del Micro Panel hasta la plataforma de PC. Ideal para manejo y visualización a nivel de máquina con paneles estacionarios, móviles o distribuidos, inclusive función de manejo, servicio técnico y diagnóstico por la web.

Dentro de cada grupo de paneles existen diferentes modelos de panel con diferentes características, por lo tanto a la hora de elegir un panel lo primero que hay que decidirse es por el tipo de panel más adecuado para el sistema.

En nuestro caso debido a la sencillez de la instalación se ha elegido un **panel de operador OP7**, que tiene una pantalla de 4 líneas de texto, un teclado de funciones, un teclado numérico y leds de señalización.



Panel de Operador SIMATIC OP7.

4.4. Comunicaciones.

En esta instalación tendremos tres tipos de comunicaciones distintas, que serán:

- *PC de programación con Autómata:* para escribir el programa y supervisar el funcionamiento.
- *PC de programación con Panel de Control:* para escribir el programa.
- *Autómata con Panel de Control:* para poder realizar el control y supervisión de la instalación a través del Panel de Control.

PC de programación con Autómata.

En nuestro caso el autómata se programa a través de la **interface multipunto MPI**. El Software necesario para realizar la programación del Autómata es "SIMATIC Step 7" de Siemens. Para realizar la conexión entre el Autómata y el PC de programación a través de la interface MPI es necesario utilizar un cable de comunicación específico.

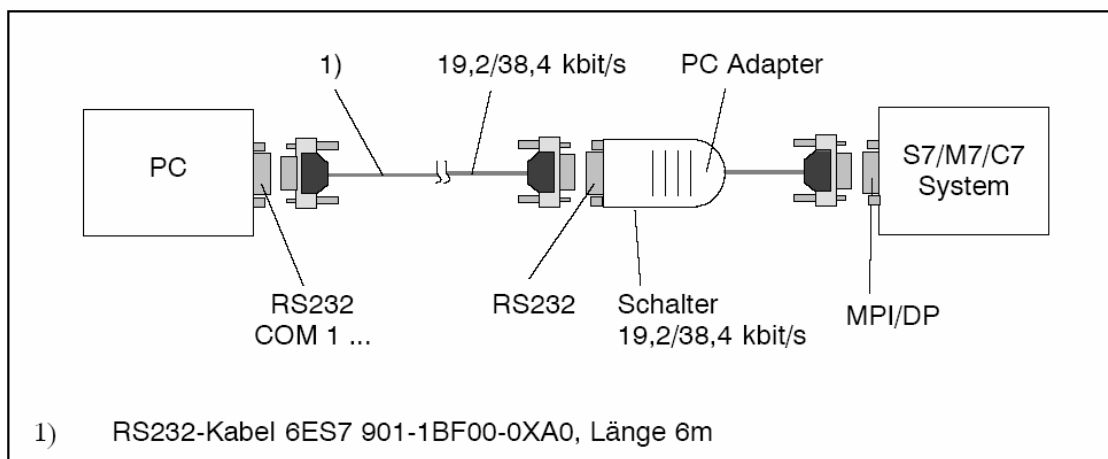


Bild 1-1 PC Adapter im System

El fabricante del Autómata ofrece diferentes soluciones para comunicar este con el PC de programación. En nuestro caso se utilizó un "**PC ADAPTER**". Este cable se conecta por un extremo al puerto serie (RS-232) del PC y por el otro al puerto MPI del autómata.

Este cable permite la conexión con autómata a dos velocidades diferentes (19200 y 38400 bits/s). Esta velocidad de transmisión indica cuántos datos se pueden transmitir en un determinado período, concretamente en nuestro caso cuántos bits se transmiten en un segundo.

Todos los aparatos que conforman una red se deben configurar de manera que transfieran datos a una misma velocidad.

La elección de este cable de comunicación se debe básicamente al precio, ya que es el dispositivo de comunicación más económico que ofrece el fabricante del Automata.

- **PC con Panel de control.**

En nuestro caso el panel de control tiene dos interfaces de comunicación:

- RS232/TTY para la conexión de autómata/PC/impresora.
- RS422/485 para la conexión de autómata/PC.

Debido a que el PC de programación no dispone de puertos RS422 se ha usado el puerto **RS232** del panel para llevar a cabo la programación.



[8]

- **Autómata con Panel de control.**

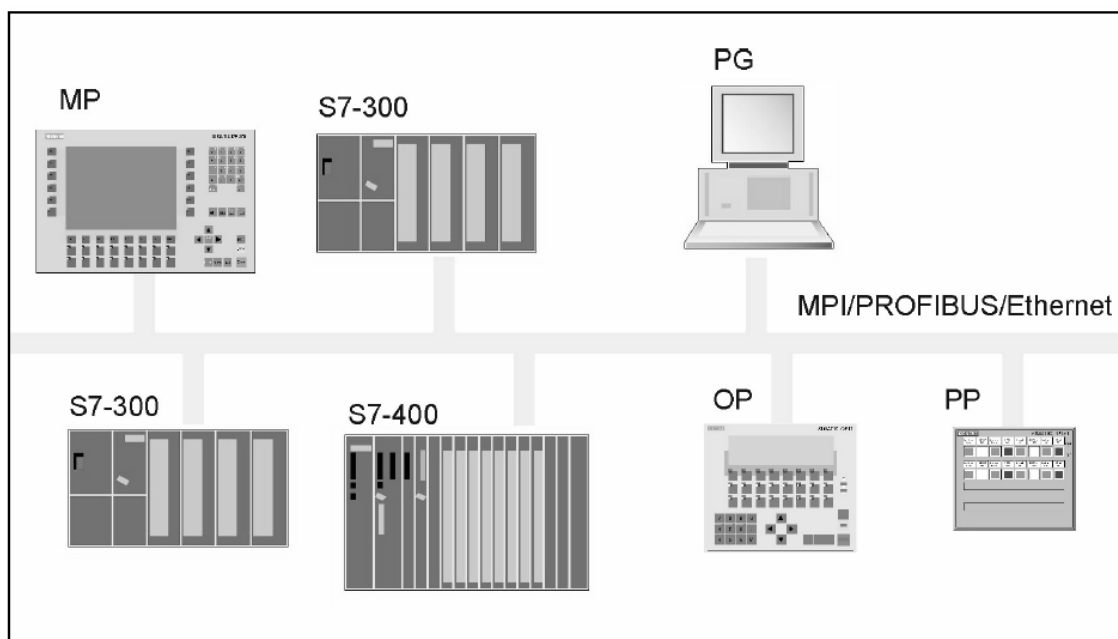
El panel de Control se conecta al autómata a través de la red MPI. El cable de conexión utilizado es un cable suministrado por el fabricante especial para este tipo de red, que tiene 2 hilos y está apantallado. La conexión en los extremos del cable se realiza con dos conectores especiales suministrados por el fabricante que incorporan una resistencia, con el fin de poder indicar, activándola o no, mediante un interruptor, el fin de red MPI.

La comunicación entre SIMATIC HMI (panel de control) y SIMATIC S7/M7/C7 (Autómata) se lleva a cabo a través de funciones OP, y es soportada plenamente por el sistema operativo de la CPU S7/M7. Por consiguiente, no se requieren a tal efecto módulos funcionales en SIMATIC S7/M7.

La comunicación OP permite intercambiar datos con varios interlocutores a la vez (SIMATIC HMI <-> SIMATIC S7/M7).

SIMATIC HMI comprende familias de productos de equipos de operación y observación, como p.ej. Push Button Panel (PP), Text Display (TD), Operator Panel (OP), Touch Panel (TP), plataforma multifuncional (MP) y software de sistema HMI para SIMATIC Panel PC (FI) o PC, que son los productos ProTool/PRO y WinCC.

Los sistemas mencionados ofrecen posibilidades de conexión a SIMATIC S7/M7-300/400 a través de MPI/PROFIBUS y con WinCC también a través de Industrial Ethernet. Los SIMATIC Push Button Panels operan en PROFIBUS como esclavos DP.



Equipos conectadas a una subred

Cada enlace entre SIMATIC HMI y un SIMATIC S7/M7/C7 ocupa en el CPU S7/M7/C7 un recurso de enlace. Cada CPU S7/M7/C7 tiene reservado por defecto un recurso de enlace para SIMATIC HMI. Por cada otro enlace hacia un SIMATIC HMI se requiere un nuevo recurso de enlace.

Para la configuración del enlace basta con seleccionar el software de programación "SIMATIC Protool" el interlocutor.

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CAPÍTULO 5

**PROGRAMA DEL
AUTÓMATA**

5. PROGRAMA DEL AUTÓMATA.

5.1.- El Software estándar STEP 7.

Para desarrollar el programa del Autómata de la instalación se necesita un software que suministra el fabricante, y que en nuestro caso es el “SIMATIC STEP 7”.

Por tanto, antes de seguir con la descripción del programa de autómata vamos a realizar una breve introducción al software de programación con el que se va a llevar a cabo.

Estándares utilizados

Los lenguajes de programación SIMATIC integrados en STEP 7 cumplen con la norma DIN EN 6.1131-3. [7] El software estándar se ejecuta bajo los sistemas operativos MS Windows 95, MS Windows 98, MS Windows Millenium, MS Windows NT 4.0 Workstation, MS Windows 2000 Professional y MS Windows XP Professional, estando adaptado a su funcionamiento gráfico y orientado a los objetos.

Funciones del software estándar [7]

El software estándar le asiste en todas las fases de creación de soluciones de automatización, tales como

- crear y gestionar proyectos
- configurar y parametrizar el hardware y la comunicación
- gestionar símbolos
- crear programas, p.ej. para sistemas de destino S7
- cargar programas en sistemas de destino
- comprobar el sistema automatizado
- diagnosticar fallos de la instalación

El interface de usuario del software STEP 7 ha sido diseñado siguiendo los criterios ergonómicos más avanzados, lo que permite conocer rápidamente sus funciones.

Herramientas auxiliares [7]

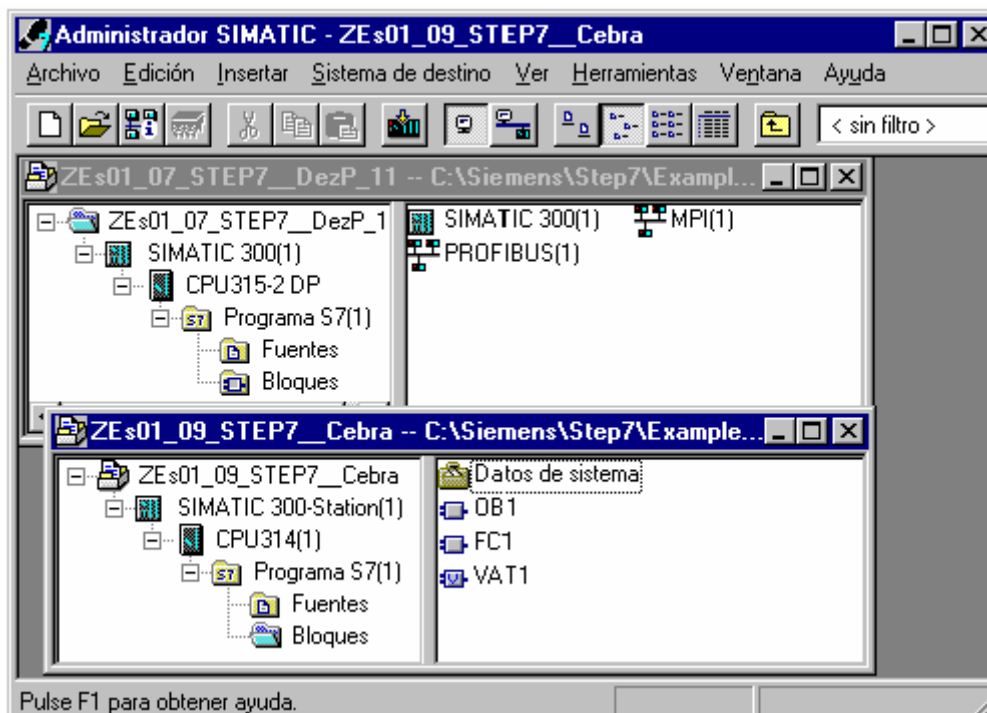
El software estándar STEP 7 ofrece toda una serie de herramientas:



Las herramientas no se deben llamar por separado, puesto que arrancan automáticamente al seleccionarse una determinada función o al abrirse un objeto.

Administrador SIMATIC [7]

El Administrador SIMATIC gestiona todos los datos pertenecientes al proyecto de automatización, independientemente del sistema de destino (S7/M7/C7) donde se encuentren. El Administrador SIMATIC arranca automáticamente las herramientas necesarias para tratar los datos seleccionados.



Editor de símbolos [7]

Con el editor de símbolos se gestionan todas las variables globales. Se dispone de las siguientes funciones:

- definir nombres simbólicos y comentarios para las señales del proceso (entradas y salidas), las marcas y los bloques,
- funciones de ordenación,
- importación/exportación de/hacia otros programas de Windows.

Todas las herramientas pueden acceder a la tabla de símbolos creada. Por consiguiente, detectan automáticamente si se ha modificado un parámetro de un símbolo.

Diagnóstico del hardware [7]

El diagnóstico del hardware permite visualizar el estado del sistema de automatización, mostrando una vista general en la que aparece un símbolo cuando alguno de los módulos presenta un fallo o no. Con un doble clic en el módulo averiado se visualizan información detallada sobre el error. El volumen de información disponible depende del módulo en cuestión:

- visualización de informaciones generales sobre el módulo (p.ej. número de referencia, versión, denominación) y sobre su estado (p.ej. fallo),
- visualización de los fallos del módulo (p.ej. errores de canal) de la periferia centralizada y de los esclavos DP,
- visualización de los avisos del búfer de diagnóstico.

En el caso de las CPUs se visualizan además las siguientes informaciones:

- causas de una ejecución errónea del programa de usuario,
- duración del ciclo (máximo, mínimo y último),
- características y grado de utilización de la comunicación MPI,
- datos característicos (cantidad de entradas y salidas, marcas, contadores, temporizadores y bloques posibles).

Lenguajes de programación [7]

Los lenguajes de programación KOP, AWL y FUP para S7-300/400 son parte integrante del software estándar.

- KOP (esquema de contactos) es un lenguaje de programación gráfico. La sintaxis de las instrucciones es similar a la de un esquema de circuitos. KOP permite observar la circulación de la corriente a través de contactos, elementos complejos y bobinas.
- AWL (lista de instrucciones) es un lenguaje de programación textual orientado a la máquina. En un programa creado en AWL, las instrucciones equivalen en gran medida a los pasos con los que la CPU ejecuta el programa. Para facilitar la programación, AWL se ha ampliado con estructuras de lenguajes de alto nivel (tales como accesos estructurados a datos y parámetros de bloques).
- FUP (diagrama de funciones) es un lenguaje de programación gráfico que utiliza los cuadros del álgebra booleana para representar la lógica. Asimismo, permite representar funciones complejas (p.ej. funciones matemáticas) mediante cuadros lógicos.

Además de estos lenguajes se ofrecen otros lenguajes de programación opcionales.

HW-Config: Configuración del hardware [7]

Esta herramienta se utiliza para configurar y parametrizar el hardware de un proyecto de automatización. Se dispone de las siguientes funciones:

- Para configurar el sistema de automatización, se eligen primero los bastidores (racks) de un catálogo electrónico y luego se asignan los módulos seleccionados a los slots de los bastidores.
- La configuración de la periferia descentralizada se efectúa del mismo modo. También se asiste la periferia canal a canal (granular).
- Al parametrizar la CPU se pueden ajustar mediante menús propiedades tales como el comportamiento en el arranque y la vigilancia del tiempo de ciclo. Se asiste el modo multiprocesador. Los datos introducidos se depositan en bloques de datos del sistema.
- Al configurar los módulos, todos los datos se pueden ajustar en cuadros de diálogo. La parametrización de los módulos se efectúa automáticamente durante el arranque de la CPU. Por consiguiente se puede p.ej. sustituir un módulo sin necesidad de repetir la parametrización.
- La parametrización de módulos de función (FMs) y de procesadores de comunicaciones (CPs) se efectúa con la misma herramienta de configuración del hardware de forma idéntica a como se parametrizan los demás módulos. Para los FM y CP se dispone de cuadros de diálogo específicos de los módulos (que forman parte del volumen de suministro del paquete de funciones FM/CP). El sistema impide que se efectúen entradas incorrectas, ya que los cuadros de diálogo sólo ofrecen las entradas admisibles.

NetPro [7]

Con NetPro, los datos se pueden transferir de forma cíclica y temporizada a través de MPI, permitiendo

- seleccionar las estaciones que intervienen en la comunicación e
- introducir la fuente y el destino de los datos en una tabla. La creación de todos los bloques a cargar (SDB) y su transferencia completa a todas las CPUs se efectúa de forma automática.

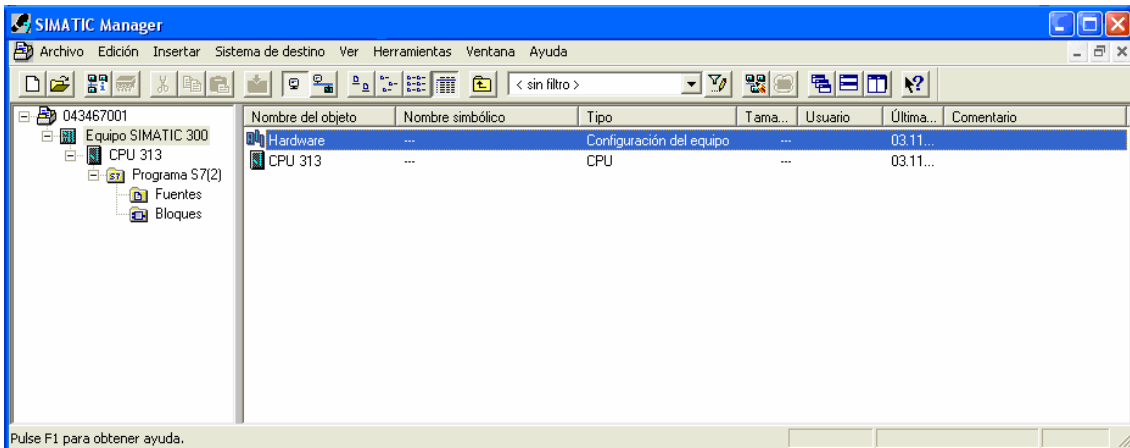
Además, existe la posibilidad de transferir los datos de forma controlada por eventos, pudiéndose

- definir los enlaces de comunicación,
- seleccionar los bloques de comunicación o de función de la librería de bloques integrada,
- parametrizar, en el lenguaje de programación habitual, los bloques de comunicación o de función seleccionados.

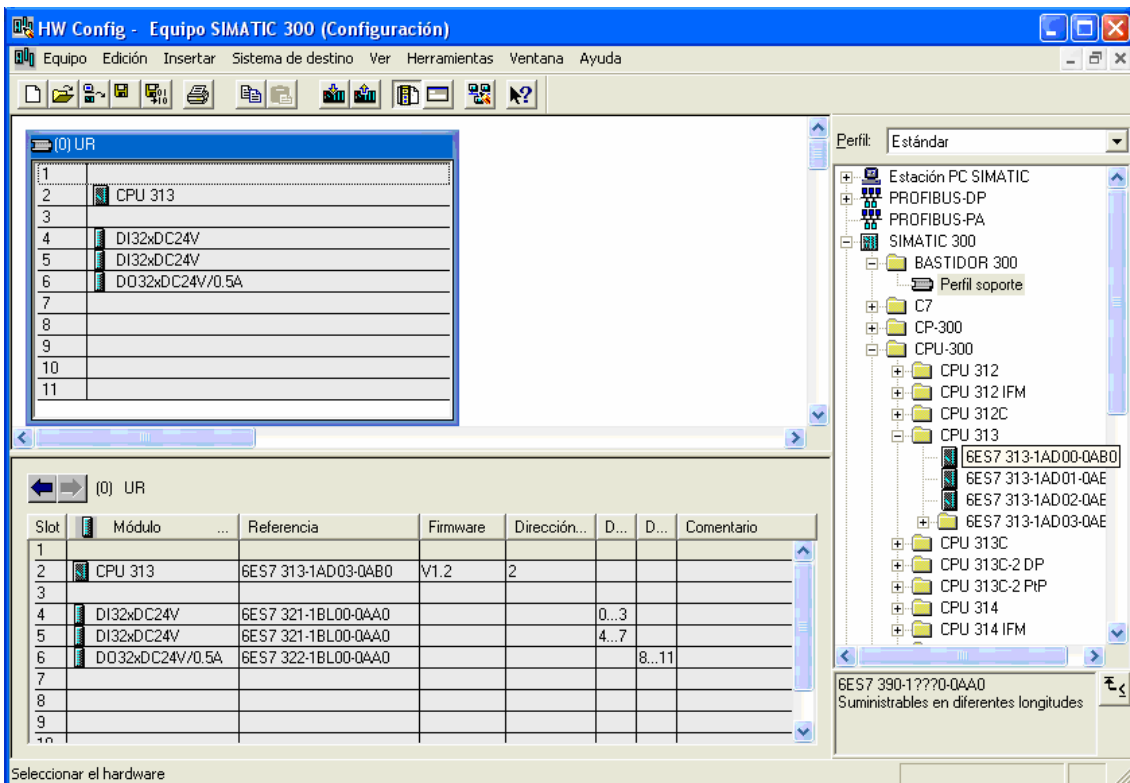
5.2.- Configuración del Hardware.

Uno de los primeros pasos antes de realizar el programa del autómatas es realizar la configuración del Hardware del automatismo, es decir, mediante el software de programación configurar los elementos que tendrá el automatismo con el fin de la que la CPU del autómatas pueda gestionarlos.

Esto se realiza con la herramienta HW-CONFIG a la cual se puede acceder desde el Administrador SIMATIC.



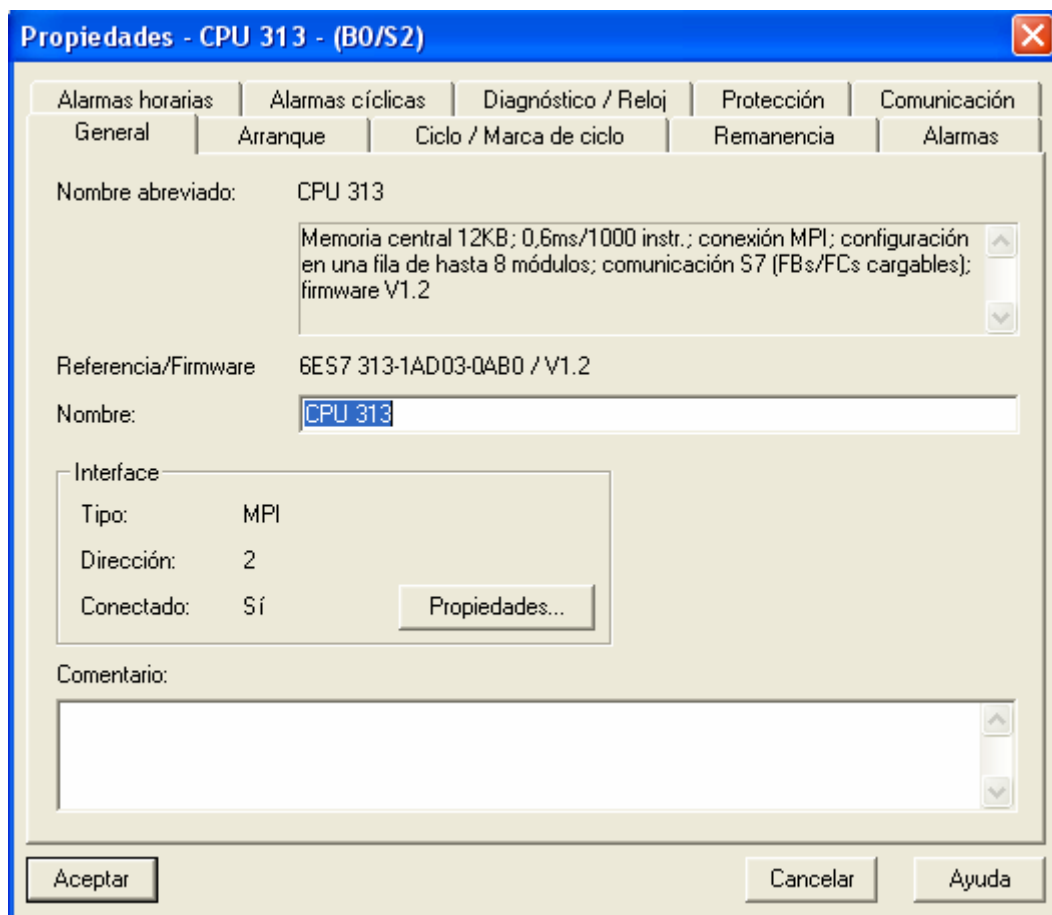
Una vez entramos en la herramienta HW-CONFIG debemos seleccionar los elemento del sistema, para lo cual disponemos de un catálogo electrónico del que podemos seleccionar los elementos.



En la figura anterior podemos ver como es la configuración de Hardware de nuestra instalación. Está compuesta por un Rack o perfil de soporte sobre el cual se colocan la CPU, las dos tarjetas de 32 entradas digitales y la tarjeta de 32 salidas digitales que componen el automatismo de nuestro proyecto.

Pinchando dos veces con el ratón sobre cualquiera de los elementos podremos ver las propiedades y opciones de configuración de los mismos. Además si estamos conectados en línea con el autómatas podemos ver también el diagnóstico de los diferentes elementos.

Si pinchamos dos veces sobre la CPU por ejemplo, veremos la siguiente ventana:



En esta ventana podemos ver la referencia de la CPU y sus características, tales como memoria, velocidad, dirección de red MPI, etc. Además tenemos las opciones de configuración de remanencia, dirección y velocidad de red MPI, alarmas horarias, etc. Y si estamos conectados con el autómatas el Diagnóstico, tiempo de ciclo, etc.

En nuestro caso todos los elementos tienen la configuración por defecto que ofrece el Software STEP 7, salvo la CPU en la cual se han modificado las opciones de remanencia de marcas de trabajo, temporizadores y contadores, para evitar que ninguno de ellos mantenga su estado tras una pérdida de tensión o un rearranque de la CPU.

5.3.- Estructura del programa.

Bloques del programa de usuario

El software de programación STEP 7 permite estructurar el programa de usuario, es decir, subdividirlo en distintas partes. Esto aporta las siguientes ventajas:

- los programas de gran tamaño se pueden programar de forma clara
- se pueden estandarizar determinadas partes del programa
- se simplifica la organización del programa
- las modificaciones del programa pueden ejecutarse más fácilmente
- se simplifica el test del programa, ya que puede ejecutarse por partes
- se simplifica la puesta en marcha.

Las partes de un programa de usuario estructurado equivalen a las distintas tareas y se definen como los bloques del programa.

Tipos de bloques

En un programa de usuario S7 se pueden utilizar diversos tipos de bloques:

- Bloques de organización (OB): definen la estructura del programa de usuario.
- Bloques de función del sistema (SFBs) y funciones del sistema (SFCs): están integrados en la CPU S7, permitiéndole acceder a importantes funciones del sistema.
- Bloques de función (FB): son bloques con “memoria” que puede programar el mismo usuario.
- Funciones (FC): contienen rutinas de programa para funciones frecuentes.
- Bloques de datos de instancia (DBs de instancia): Al llamarse a un FB/SFB, los DBs de instancia se asocian al bloque. Los DBs de instancia se generan automáticamente al efectuarse la compilación.
- Bloques de datos (DB): son áreas de datos para almacenar los datos de usuario. Adicionalmente a los datos asociados a un determinado bloque de función, se pueden definir también datos globales a los que pueden acceder todos los bloques.
- Tipos de datos (UDT): se utiliza para crear una estructura de datos definida por el usuario.

Los OBs, FBs, SFBs, FCs y SFCs contienen partes del programa, por lo que se denominan también bloques lógicos. El número permitido de bloques de cada tipo y su longitud admisible dependen de la CPU.

Organización del programa.

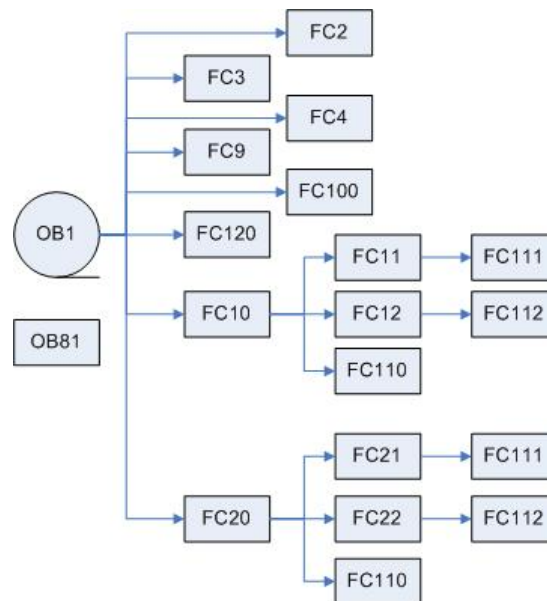
Una vez explicado para que sirve cada tipo de bloque vamos a ver los que se usaran en nuestro programa y como están organizados.

Se utilizarán bloques de organización (OB), funciones (FC), bloques de datos (DB) y tipos de datos (UDT).

A continuación se muestra una tabla con todos los bloques utilizados en el programa.

Bloques de Organización (OB)	
OB1	Cycle Execution
OB81	Fallos de Alimentación
OB121	Error de Programación
Funciones (FC)	
FC2	Arranque y Parada Instalación
FC3	Gestión Baliza 2 colores
FC4	Defectos Generales
FC9	Control leds Panel de Operador
FC10	Organización Zona 1
FC11	Control Flujo Zona 1
FC12	Gestión Alarmas Zona 1
FC20	Organización Zona 2
FC21	Control Flujo Zona 2
FC22	Gestión Alarmas Zona 2
FC100	Tratamientos Temporizadores
FC110	Activaciones Manuales
FC111	Control Puesto Trabajo
FC112	Gestión de Atascos
FC120	Tratamientos de Alarmas
Bloques de datos (DB)	
DB50	Remanencia
DB100	Valores Preselección Temporizadores
DB110	Movimientos manuales Zona 1
DB111	Movimientos manuales Zona 2
DB112	Datos Atascos
DB113	Consigna Tiempos Atascos
DB114	Memoria de Atascos
DB120	Alarmas
DB121	Acuses Alarmas
DB122	Memoria de Alarmas
DB123	Áreas intercambio panel operador
DB124	Mensajes de Servicio
Tipo de Datos (UDT)	
UDT1	Estructura Datos Movimientos Manuales

En la siguiente figura se muestra un esquema de la estructura básica del programa, en la que podemos ver como se van llamando los diferentes bloques de programa.



Bloques de Organización (OBs).

Los bloques de organización (OB) constituyen el interface entre el sistema operativo y el programa de usuario. Son llamados por el sistema operativo y controlan el procesamiento cíclico y controlado por alarmas del programa, el comportamiento de arranque del sistema de automatización y el tratamiento de los errores. Programando los bloques de organización se define el comportamiento de la CPU.

Los bloques de organización determinan la secuencia (eventos de arranque) en la que habrán de ejecutarse las diferentes partes del programa. La ejecución de un OB puede ser interrumpida por la llamada de otro OB. Que OB puede interrumpir a otro OB depende de su prioridad. Los OB's de mayor prioridad pueden interrumpir a los de menor prioridad. La prioridad más baja la tiene el OB de tarea no prioritaria.

En un PLC el funcionamiento normal del programa es cíclico, es decir, el programa del PLC se ejecuta una vez tras otra de forma cíclica. En nuestro autómatas el sistema operativo llama cíclicamente al OB1 desde el cual realizaremos las llamadas a las diferentes funciones del programa.

La ejecución cíclica del programa puede ser interrumpida por:

- una alarma
- una orden STOP (selector de modo de operación, comando de menú desde la PG, SFC 46 STP, SFB 20 STOP)
- un corte de tensión de red (alimentación)
- el fallo de un aparato o por un error del programa

En nuestro programa se han usado los siguientes bloques de organización:

➤ **OB1 – Cycle Execution.**

Segmento 1. Arranque y Parada del Sistema.

En este segmento se activan las marcas auxiliares de control de arranque y se llama a la función estándar FC2 que controla el arranque del sistema.

```
// Agrupación de alarmas para llamada a FC2
U   "Alarmas".Emergencia_general
O   "Alarmas".Emergencia_X1B2
O   "Alarmas".Emergencia_X1B7
O   "Alarmas".RELE_SEGURIDAD_NO_OK
=   "Agrupacion emergencias"

// Agrupación de Termicos (solo si paran toda la instalación)
U   "Alarmas".Termico_Q301
U   "Alarmas".Termico_Q302
U   "Alarmas".Termico_F51
U   "Alarmas".Termico_F71
=   "Agrupacion termicos"

// Agrupación de presostatos
U   "Alarmas".Presostato
=   "Agrupacion presostatos"

// Agrupación de reposos de la instalación
U   "En reposo zona 1"
U   "En reposo zona 2"
=   "Agrupacion reposos"

// Agrupación de zonas en manual
U   "Manual zona 1"
U   "Manual zona 2"
=   "En manual general"

// Agrupación de zonas en automatico
U   "Automatico zona 1"
U   "Automatico zona 2"
=   "En automatico general"

// Llamada modulo de arranque FC2
CALL "Arranque / Parada inst."
pulsador_marcha:="PM"
pulsador_paro  := "PP"
reposo         := "Agrupacion reposos"
def_emergencias:= "Agrupacion emergencias"
def_termicos   := "Agrupacion termicos"
def_presostato := "Agrupacion presostatos"
sirena         := "H784"

//Activación Lámpara Marcha
UN   "Marcha general"
UN   "Agrupacion emergencias"
U    "Int_500ms"
O
U    "Marcha general"
O
U    "Prueba_Lamparas"
=    "SH901"
```

```
//Activación Electroválvula paso de aire.
U      "Fin secuencia inicial"
=      "Y1100"
```

Segmento 2. Activación Baliza del armario principal.

En este segmento se realiza la llamada a la función estándar que controla el funcionamiento de la baliza de 2 colores situada en el armario general:

```
//Llamada función estándar control Baliza 2 colores
CALL  "Baliza 2 colores"
      alarmas      := "Alarma activada general"
      emergencias := "Agrupacion emergencias"
      manual       := "En manual general"
      baliza_verde:= "H782"
      baliza_roja := "H783"
```

Segmento 3. Tratamiento de Alarmas.

En este segmento se realiza la llamada al módulo de tratamiento de alarmas como se muestra a continuación:

```
//Llamada función estándar Tratamiento de Alarmas
CALL  "Tratamiento alarmas"
      pulsador_rearme      := "REA"
      pulsador_enterado    := "Areas".OP[1].Teclado_sistema.ACK
      direccion_inicio     := 0
      cantidad_word        := 25
      Db_alarmas           := "Alarmas"
      Db_acuses            := "Acuses"
      Db_mem_alarmas       := "Memoria_alarmas"
      sirena               := "H784"
      Bit_alarma_activa    := "Alarma activada general"
      Bit_nueva_alarma_activa:= "Nueva alarma general"
```

Segmento 4. Llamada a módulos de programa.

En este segmento se realiza la llamada a los diferentes módulos de programa. En primer lugar se llaman a las funciones generales comunes a todas las zonas funcionales de la instalación y posteriormente se llaman a las funciones que se encargan de la organización de las zonas funcionales.

```
//Llamada a Módulos generales
UC    "Defectos generales"
UC    "Trat. temporizadores"
UC    "Leds y Pantallas OP7"

//Llamada a Módulos Zona 1
UC    "Organización Zona 1"

//Llamada a Módulos Zona 2
UC    "Organización Zona 2"
```


➤ OB81 – Fallos alimentación.

El sistema operativo de la CPU llama al OB 81 cuando se presenta un evento activado por un fallo en la alimentación normal (sólo en el S7-400) o de respaldo en tampón (tanto en caso de un evento entrante como de uno saliente).

Este módulo tiene una serie de datos locales que hay que interpretar para saber qué error nos devuelve. Esta información se puede obtener en la ayuda del Software STEP 7, ya que esté OB está realizado por Siemens. A continuación se muestra una imagen con los datos locales del bloque:

Nombre	Tipo de datos	Dirección	Comentario
OB81_EV_CLASS	Byte	0.0	16#39, Event class 3, Entering event state, Internal fault event
OB81_FLT_ID	Byte	1.0	16#XX, Fault identification code
OB81_PRIORITY	Byte	2.0	26/28 (Priority of 1 is lowest)
OB81_OB_NUMBR	Byte	3.0	81 (Organization block 81, OB81)
OB81_RESERVED_1	Byte	4.0	Reserved for system
OB81_RESERVED_2	Byte	5.0	Reserved for system
OB81_MDL_ADDR	Int	6.0	Reserved for system
OB81_RESERVED_3	Byte	8.0	Reserved for system
OB81_RESERVED_4	Byte	9.0	Reserved for system
OB81_RESERVED_5	Byte	10.0	Reserved for system
OB81_RESERVED_6	Byte	11.0	Reserved for system
OB81_DATE_TIME	Date_And...	12.0	Date and time OB81 started

Segmento 1. Gestión fallo pila de respaldo programa.

En este segmento se activa una marca si hay fallo en la pila de respaldo de la CPU para poder dar una alarma por pantalla.

```
//#OB81_EV_CLASS = 39 --> Evento Entrante
//#OB81_EV_CLASS = 38 --> Evento Saliente
//#OB81_FLT_ID = 22 --> Fallo tensión de respaldo en el aparato
//
//                               central

//Activación marca fallo bateria CPU
//con evento entrante del fallo
    U(
    L    #OB81_EV_CLASS
    L    B#16#39
    ==I
    )
    U(
    L    #OB81_FLT_ID
    L    B#16#22
    ==I
    )
    S    "Fallo bateria CPU"

//Desactivación marca fallo bateria CPU
//con evento saliente del fallo
    U(
    L    #OB81_EV_CLASS
    L    B#16#38
    ==I
    )
    U(
    L    #OB81_FLT_ID
    L    B#16#22
    ==I
    )
    R    "Fallo bateria CPU"
```

➤ **OB121 – Error de programación.**

El sistema operativo de la CPU llama al OB 121 cuando aparece un evento activado por un error durante la ejecución del programa.

El OB 121 funciona en la misma prioridad que el bloque que ha sido interrumpido.

Si no está programado el OB 121, la CPU pasará de RUN a STOP.

Este bloque de organización normalmente no se suele cargar salvo durante el periodo de pruebas, ya que nos da la posibilidad de no cargar el programa completo y probarlo por partes. Es decir, si no cargamos el programa completo la CPU se pondría en STOP si llamamos a un módulo que no está cargado, y con este OB impedimos que la CPU se pondrá en STOP.

Módulos de función (FCs).

Las funciones son bloques programables. Una función es un bloque lógico "sin memoria". Las variables temporales de las FC's se memorizan en la pila de datos locales. Estos datos se pierden tras el tratamiento de las FC's. Para fines de memorización de datos, las funciones pueden utilizar bloques de datos globales.

Como una FC no tiene asignada ninguna memoria, se han de indicar siempre parámetros actuales. A los datos locales de una FC no se pueden asignar valores iniciales.

La FC contiene un programa que se ejecuta cada vez que la FC es llamada por otro bloque lógico. Las funciones se pueden utilizar para:

- devolver un valor de función al bloque invocante (ejemplo: funciones matemáticas)
- ejecutar una función tecnológica (ejemplo: control individual con combinación binaria).

A estas funciones se les puede pasar parámetros, algo muy útil si hay varias partes en la instalación que sean iguales, ya que basta con hacer el programa para alguna de ellas y mediante estos parámetros se puede generalizar para todas, ahorrando tiempo y trabajo.

El parámetro formal es un comodín para el parámetro real, es decir, el parámetro actual. Los parámetros actuales sustituyen a los parámetros formales al efectuar la llamada a una FC. A los parámetros formales de una FC se han de asignar siempre parámetros actuales. Los parámetros de entrada, de salida y de entrada/salida utilizados por la FC se depositan en forma de punteros en los parámetros actuales del bloque lógico que ha llamado a la FC.

A diferencia con los bloques de función (FB), al acceder a parámetros se utiliza la copia del parámetros actual del DB de instancia. Si en la llamada de un FB no se transfiere un parámetro de entrada o si en el bloque no se escribe un parámetro de salida, se continúan utilizando los valores antiguos todavía existentes en el DB de instancia (DB de instancia = memoria del FB).

Las funciones (FC) no tienen memoria. Por eso, al contrario que en el caso del FB, el hecho de proporcionar los parámetros formales no es opcional, sino necesario. El acceso a parámetros de FC se realiza a través de direcciones (indicadores de varias áreas). Si se utiliza como parámetro actual un operando del área Datos (bloque de datos) o una variable local del bloque que realiza la llamada, para la transmisión de parámetros se guarda temporalmente una copia del parámetro actual en los datos locales del bloque que realiza la llamada.

Por ello debe tener en cuenta los siguientes puntos:

- Si es posible inicialice parámetros OUTPUT.
- Los comandos de activar y desactivar dependen de RLO. Si con estos comandos se determina el valor de un parámetro OUTPUT, si RLO = 0 no se forma ningún valor.
- Tenga en cuenta que independientemente de todas las rutas de programa posibles dentro del bloque los parámetros OUTPUT se describen siempre. Debe prestar especial atención a comandos de salto y a la salida ENO en KOP y FUP. Tampoco debe olvidar BEB y el efecto de los comandos MCR.

➤ FC2 – Arranque y Parada instalación.

Mediante este FC se realiza el protocolo de marcha y parada de la instalación, así como el control de los enclavamientos generales de la instalación.

En la página 78 hemos visto como se realiza la llamada a esté módulo desde el OB1, que es como sigue:

```
// Llamada modulo de arranque FC2
CALL "Arranque / Parada inst."
pulsador_marcha:="PM"
pulsador_parado := "PP"
reposito       := "Agrupacion reposos"
def_emergencias := "Agrupacion emergencias"
def_termicos   := "Agrupacion termicos"
def_presostato := "Agrupacion presostatos"
sirena         := "H784"
```

En la tabla de abajo podemos ver las variables locales con las que trabaja la función:

Nombre	Tipo de Datos	Tipo Parametro
Pulsador_marcha	BOOL	Entrada
Pulsador_paro	BOOL	Entrada
Reposo	BOOL	Entrada
Def_emergencias	BOOL	Entrada
Def_termicos	BOOL	Entrada
Def_presostatos	BOOL	Entrada
SIRENA	BOOL	Entrada / Salida

Este módulo controla el arranque de la instalación. Al presionar el pulsador de marcha se inicia el proceso de arranque de la instalación. Durante este proceso la sirena sonará de forma interrumpida durante el tiempo T1. Una vez finalizado el proceso de arranque la sirena parará y mediante la luz fija del pulsador de marcha se indicará que la instalación está en marcha.

Segmento 1. Arranque Inicial.

En este segmento se activa una marca que indica que hay petición de marcha (M2.0) con la que se inicia el proceso de arranque. La bocina suena de forma continua durante el tiempo de arranque (T1). Una vez finaliza el proceso de arranque se memoriza activando la marca M2.1 (FIN_SECUENCIA_INICIAL).

```
//Flanco pulsador de Marcha
U    #pulsador_marcha
UN   "Aux Flanco PM"
=    "FP Pulsador Marcha"
U    #pulsador_marcha
=    "Aux Flanco PM"

//Activación Petición de Marcha
U    "FP Pulsador Marcha"
UN   "Agrupacion emergencias"
S    "Peticion de marcha"

//Si no ha finalizado el arranque activamos bocina
UN   "Fin secuencia inicial"
SPBN JUMP

U    "Peticion de marcha"
UN   "Fin secuencia inicial"
=    "Activacion T1"

U    "Peticion de marcha"
UN   "Fin secuencia inicial"
UN   T    1
=    #sirena
JUMP: NOP    0

//Memorizamos fin de arranque de la instalación
U    "Peticion de marcha"
U    T    1
S    "Fin secuencia inicial"
```

Segmento 2. Parada General.

En este segmento se realiza el proceso de paro de la instalación. Al pulsar el botón de parada se memoriza el proceso de paro con el bit M2.2 (PETICION_PARADA). Se debe de usar dicho bit para bloquear los diferentes ciclos o procesos en sus puntos de reposo conocidos. Cuando se alcancen los reposos de todas las zonas funcionales el programa pondrá a cero los bits M2.1 y M2.2. Si durante el proceso de parada, se acciona el pulsador de marcha, el proceso se detendrá dejando el sistema activo.

```
//Flanco Pulsador de parada
U   #pulsador_paro
UN  "Aux Flanco PP"
=   "FP Pulsador Paro"
U   #pulsador_paro
=   "Aux Flanco PP"

//Memoriza petición de parada
U   "FP Pulsador Paro"
U   "Fin secuencia inicial"
S   "Petición de parada"

//Si tenemos reposo de la instalación reseteamos marcas de marcha
U   "Petición de parada"
U   #reposo
R   "Fin secuencia inicial"
R   "Petición de parada"
R   "Petición de marcha"

//Si tenemos pulsador de marcha se interrumpe petición de parada
U   "Petición de parada"
U   "Fin secuencia inicial"
U   "FP Pulsador Marcha"
R   "Petición de parada"
```

Segmento 3. Condición Inicial General.

En este segmento se activa la marca de "MARCHA GENERAL" (M2.5) una vez finalizado el arranque de la instalación siempre y cuando las emergencias de la instalación, los térmicos y los presostatos estén OK.

```
//Activación de "Marcha general"
U   "Fin secuencia inicial"
UN  #def_emergencias
UN  #def_termicos
UN  #def_presostato
=   "Marcha general"
```

➤ **FC3 – Baliza de dos colores.**

Esta función es la que se encarga de controlar el funcionamiento de la baliza de dos colores (verde/rojo) situada en el armario general.

A continuación se muestra una tabla con la descripción del funcionamiento de la baliza, en la que podemos ver el estado de la instalación según esté iluminada la baliza:

BALIZA VERDE	BALIZA ROJA	DESCRIPCIÓN
Apagado	Apagado	Sistema parado.
Encendido fijo	Apagado	Sistema en marcha.
Encendido parpadeando a Frecuencia 250 ms.	-----	La instalación esta en proceso de arranque.
Encendido	Encendido parpadeando a Frecuencia 1 s.	Aviso de avería o la instalación en manual.
Encendido	Encendido fijo	Parada de emergencia general activada, no rearmada o alarma general.
Encendido parpadeando a Frecuencia 1 ms.	-----	La instalación esta en proceso de parada.

En la página 78 hemos podido observar como se realiza la llamada a esta función:

```
//Llamada función estándar control Baliza 2 colores
CALL "Baliza 2 colores"
  alarmas      := "Alarma activada general"
  emergencias := "Agrupacion emergencias"
  manual       := "En manual general"
  baliza_verde := "H782"
  baliza_roja  := "H783"
```

En la tabla siguiente podemos ver las variables locales con las que trabaja la función:

Nombre	Tipo de Datos	Tipo Parametro
Alarmas	BOOL	Entrada
Emergencias	BOOL	Entrada
Manual	BOOL	Entrada
Baliza_verde	BOOL	Entrada / Salida
Baliza_roja	BOOL	Entrada / Salida

Una vez sabemos lo que hace este módulo vamos a ver cómo es el código que se encarga de controlar la baliza.

Segmento 1. Lámpara verde de estado en servicio.

```
//Activación Baliza verde.
U      "Fin secuencia inicial"
UN     "Petición de parada"
O
U      "Fin secuencia inicial"
U      "Petición de parada"
U      "Int_1s"
O
U      "Petición de marcha"
UN     "Fin secuencia inicial"
UN     "Petición de parada"
U      "Int_250ms"
O
U      "Prueba_Lamparas"
=      #baliza_verde
```

Segmento 2. Lámpara roja de emergencia.

```
//Activación Baliza roja.
U      "Petición de marcha"
U(
O(
U      "Int_1s"
U(
O(
U      #alarmas
UN     #emergencias
)
O      #manual
)
)
O      #emergencias
)
O
U      "Prueba_Lamparas"
=      #baliza_roja
```

➤ **FC4 – Defectos Generales.**

Esta función a diferencia con las anteriores no es estándar, si no que depende de la instalación, pero si se procura seguir una estructura. El motivo de utilizar una estructura determinada es que cualquier otro programador de la empresa pueda trabajar con un programa que no ha realizado sin mayores problemas. La estructura es la siguiente:

- Segmento 1: Gestión fallos de emergencia.
- Segmento 2: Gestión de disparos Térmicos.
- Segmento 3: Gestión de fallos de presostatos.
- Segmento 4: Gestión de fallo batería.
- Segmento 5: Gestión de otros fallos.

Todas las alarmas serán memorizadas en el bloque de datos DB120 que veremos con más detalle a lo largo del capítulo.

A continuación vemos un ejemplo de cómo se memoriza una alarma:

```
//Alarma Rele de Seguridad no OK
UN    "Alarmas".Emergencia_general
UN    "Alarmas".Emergencia_X1B2
UN    "Alarmas".Emergencia_X1B7
UN    "RELE_GENERAL_OK"
UN    "Alarmas".RELE_SEGURIDAD_NO_OK
S     "Alarmas".RELE_SEGURIDAD_NO_OK
```

➤ FC9 – Leds y pantallas OP7.

Esta función se encarga de controlar el estado de los Leds del panel de operador según en que pantalla se encuentre el operador.

El código de esta función no estándar, sino que depende de la instalación. La activación de los leds se hace a través del área de interface entre el PLC y el OP (DB123 que veremos con más detalle a lo largo del capítulo).

A continuación veremos un ejemplo de cómo se hace la activación de un led:

```
//ACTIVACIÓN DEL LED K1
//EN PANTALLA 12
U(
L    "Areas".OP[1].Imagen.Num_de_Pantalla
L    12
==I
)
SPBN K1_1

//LED PARPADEA 0,5HZ
U(
O    "Led_K1_Int_Pant_12"
O    "Led_K1_Fijo_Pant_12"
)
=    "Areas".OP[1].Imagen_de_los_led.Led_tecla_K1_bit_1

//LED FIJO
U    "Led_K1_Fijo_Pant_12"
=    "Areas".OP[1].Imagen_de_los_led.Led_tecla_K1_bit_2

K1_1: NOP    0
```

Podemos ver en el ejemplo que la activación del led se hace a través de dos bit, de forma que si activamos el bit 1 el led se ilumina de forma intermitente, y si activamos el bit 1 y 2 el led se ilumina de forma fija. También podemos observar que la activación se hace con un salto, debido a que estos led son activados también en la función estándar FC110 (movimientos manuales), por tanto no podemos dejar que se ejecute la instrucción = salvo cuando sea necesario.

➤ FC10/FC20 – Módulos de organización.

Estás funciones se encargan de la organización de cada zona funcional de la instalación, es decir, gestionan el estado de cada zona, llaman a los módulos que controlan la zona en cuestión, etc.

La llamada a estos módulos se realiza desde el OB1. Veamos a continuación con más detalle el código de uno de ellos (FC10. Organización Zona 1):

Segmento 1. Tratamiento Alarmas Zona 1.

En este segmento se realiza la llamada a la función de gestión de alarmas de la zona 1 y se activa la marca auxiliar de “Alarma Zona 1” si existe alguna alarma en la zona.

```
//Llamada a la función de gestión de alarmas
CALL "Gestion Alarmas Z1"

//Genera marca de alarma zona
AUF "Alarmas"

O(
L DBW 0
L 0
<>I
)
O DBX 2.0
O DBX 2.2
O DBX 2.3
O(
L DBD 4
L L#0
<>D
)
O(
L DBD 8
L L#0
<>D
)
O(
L DBW 12
L 0
<>I
)
= "Alarma zona 1"
```

Segmento 2. Marcha y Parada de Zona 1.

En este segmento se gestionan las marcas de marcha, parada y reposo de la zona 1. Estas marcas son importantes, ya que se utilizarán como condiciones de movimiento y parada de los diferentes elementos que componen la zona 1. Así mismo estas marcas dependen directamente de las de marcha y parada general de la instalación. Además, la marca de marcha puede tener diferentes condiciones de seguridad tales como emergencias, etc.

```
//Activación Marca Reposo Zona
UN  "REMANENCIA".TR_TNA02_TNA03
UN  "REMANENCIA".TR_TNA03_CGA02
UN  "REMANENCIA".TR_TNA05_TNA06
UN  "REMANENCIA".TR_TNA06_CGA03
UN  "REMANENCIA".Salida_1A
UN  "REMANENCIA".Salida_2A
UN  "REMANENCIA".Salida_3A
O
U   "En reposo zona 1"
UN  "Peticion de marcha"
=   "En reposo zona 1"

//Activación marca Parada Zona
U   "Peticion de parada"
U   "En reposo zona 1"
=   "Parada zona 1"

//Activación marca Marcha Zona
U   "Marcha general"
UN  "Agrupacion emergencias"
UN  "Parada zona 1"
=   "Marcha zona 1"
```

Segmento 3. Tratamiento de Manual y Automático Zona 1.

En este segmento se activan las marcas de Automático y Manual de la zona. El estado se selecciona desde el panel de Operador con la zona en Macha. Estas marcas se usarán posteriormente para el control de los diferentes elementos de la instalación.

```
//Manual Zona 1
U   "Marcha zona 1"
U   "Areas".Zona_A.Manual_OP
S   "Manual zona 1"
R   "Automatico zona 1"

//Automático Zona 1
U   "Marcha zona 1"
U   "Areas".Zona_A.Automatico_OP
S   "Automatico zona 1"
R   "Manual zona 1"
```

Segmento 4. Visualización Modo funcionamiento en la OP.

En este segmento se escribe un valor en un byte que se usará posteriormente en el panel de operador para visualizar el estado de la instalación mediante una lista de textos.

La lista de textos que se muestra en el panel es la siguiente:

Valor Byte	Texto
0	PARADA
1	EN MANUAL
2	EN AUTOMÁTICO
3	PARADA
4	CON DEFECTO

```
//Gestión Estado Zona 1
  U   "Manual zona 1"
  SPBN M001
  L   1
  T   "Areas".Zona_A.Modos_Funcionamiento
  SPA  FIN2

M001: U   "Alarma zona 1"
      SPBN M002
      L   4
      T   "Areas".Zona_A.Modos_Funcionamiento
      SPA  FIN2

M002: U   "Automatico zona 1"
      SPBN M003
      L   2
      T   "Areas".Zona_A.Modos_Funcionamiento
      SPA  FIN2

M003: U   "Parada zona 1"
      SPBN M004
      L   3
      T   "Areas".Zona_A.Modos_Funcionamiento
      SPA  FIN2

M004: NOP  0
      L   0
      T   "Areas".Zona_A.Modos_Funcionamiento

FIN2: NOP  0
```

Segmento 5. Tratamiento de Movimientos Manuales.

En este segmento se realiza la llamada a la función que gestiona los movimientos manuales de la zona.

```
//Llamada a la función que gestiona los movimientos manuales
CALL "Activacion manuales"
  Numero_pantalla_OP_req :=12
  Cantidad_de_movimientos:=5
  Bit_Manual              :="Manual zona 1"
  DB_movimientos          :="Mov_manuales_Zona_1"
  DB_areas_OP             :="Areas"
  Aux_bit_manual          :="FN_manual_Zona_1"
  Aux_bit_op              :="FN_cambio_pantalla_Z1"
  Aux_FP_K3               :="FP_K3_Z1"
  Aux_FP_K4               :="FP_K4_Z1"
  Aux_Fn_K3               :="FN_K3_Z1"
```

En la llamada a la función debemos pasarle una serie de parámetros para definir por ejemplo la cantidad de movimientos manuales. No entraremos en más detalle, ya que más adelante veremos el funcionamiento de esta función.

Segmento 6. Activación de los Leds de la OP.

En este segmento se activan las marcas auxiliares que serán utilizadas en la función FC9 que ya vimos con anterioridad para activar los leds del panel de operador relacionados con las pantallas de la zona en cuestión (zona 1).

En concreto las marcas que se activarán en este segmento están relacionadas con la imagen del OP que tiene como función la selección de modo de funcionamiento (Automático/Manual). De forma que el led estará fijo indicando qué modo está seleccionado y se iluminará de forma intermitente indicando qué se puede seleccionar.

```
//Led K1 pantalla 11 (Automático)
UN      "Automatico zona 1"
=       "Led_K1_Int_Pant_12"

U       "Automatico zona 1"
=       "Led_K1_Fijo_Pant_12"

//Led K2 pantalla 11 (Manual)
UN      "Manual zona 1"
=       "Led_K4_Int_Pant_12"

UN      "Manual zona 1"
=       "Led_K4_Fijo_Pant_12"
```

Segmento 7. Llamadas a Módulos de la Zona 1.

En este segmento se realizan las llamadas a los módulos que controlan el funcionamiento de la zona 1, que en nuestro caso es sólo el módulo de control del flujo de funcionamiento.

```
//Llamada al módulo de control del flujo de funcionamiento
UC      "Control Flujo Zona 1"
```

➤ **FC11/FC21 – Módulos de flujo.**

Estas funciones se utilizan para tener controlado el movimiento de la zona. Se realiza la gestión de todos los accionamientos de la zona con las condiciones de manual, automáticos y alarmas siguiendo el flujo de la instalación.

Cada accionamiento está ubicado en un segmento y teniendo en cuenta que en estos módulos se sigue el flujo de la instalación será más fácil encontrar el accionamiento buscado.

La llamada a estos módulos se realiza desde el módulo de organización de su zona como vimos anteriormente. Pese a que existan accionamientos diferentes (p.ej. topes, motores, etc.), en todos ellos se trata de mantener una estructura similar dentro de cada segmento, que sería como sigue:

1. *Alarmas.* Generar alarmas del accionamiento.
2. *Condiciones de funcionamiento.* Según la complicación del accionamiento se activarán las marcas auxiliares de condiciones de movimiento.
3. *Activación del accionamiento.* Activación de las salidas del PLC que comandan el accionamiento.
4. *Llamada a módulos auxiliares de control.* Según la complejidad del accionamiento.

A continuación, veremos un par de ejemplos del código que gestiona estos accionamientos:

1. MOTOR M11 (+CAC2).

```
//*****          ALARMAS          *****
//   Defecto Termico
//   UN   "Q301"
//   UN   "Alarmas".Termico_Q301
//   S    "Alarmas".Termico_Q301

//*****  Activacion Salidas  *****
//   U    "Marcha general"
//   UN   "Agrupacion emergencias"
//   U(
//   Automático
//   U    "Automatico zona 1"
//   UN   "Alarmas".Termico_Q301
//   UN   "Alarmas".TR_TNA01_TNA02
//   U(
//   L    DB120.DBW 9 //Palabra de alarmas de transf. y atascos
//   L    0
//   ==I
//   )
//   O
//   Manual
//   U    "Manual zona 1"
//   U    "Mov_manuales_Zona_1".Activacion_mov[0]
//   )
//   =    "K301"
```

2. Control Puesto 1A. Cruz de giro CGA1.

```
//*****          ALARMAS          *****
//   U    "REMANENCIA".Salida_1A
//   UN   "B123"
//   S    "Alarmas".Cruz_Giro_CGA1

//*****  CONTROL PUESTO  *****
CALL "Control_puesto"
Fin_Trabajo           := "Fin_ciclo_puesto_1A"
Presencia_en_puesto   := "B132"
Cruz_Giro_abajo       := "B123"
Tope_abajo            := "B124"
Presencia_Tope_siguiete := "B133"
Tope_Siguiente_abajo := "B125"
Tipo_de_puesto        := FALSE
Consig_Tim_reset_puesto := 10000
Puesto_Trabajo_Ocupado := "REMANENCIA".Ocupado_1A
Salida_Puesto_Trabajo := "REMANENCIA".Salida_1A
Borrado_Puesto_Trabajo := "REMANENCIA".Borrado_1A
Lampara_Pulsador      := "SH111"
Mem_Cruz_Giro_ha_subido := "REMANENCIA".Mem_Cruz_Giro_arriba_1A
Mem_fin_trabajo       := "REMANENCIA".Mem_fin_trabajo_1A
Tim_reset_puesto      := "Tim_reset_1A"
```

Debido a que todos los puestos de la instalación son prácticamente iguales se ha realizado una función parametrizable para realizar el control del puesto, de forma que se minimiza el tiempo de desarrollo de programa y puesta en marcha. Más adelante veremos esta función con más detalle.

➤ FC12/FC22 – Gestión Alarmas Zona.

Estas funciones se utilizan para gestionar las alarmas por atasco en detector y los fallos de transferencia de la zona. A continuación veremos un ejemplo de cada una de ellas:

1. Alarmas por atasco en detectores.

Una alarma por atasco se produce si hay algo que se esté detectando durante demasiado tiempo, siempre y cuando las condiciones de la instalación no impidan el movimiento de lo que se esté detectando. En nuestro caso por ejemplo si una bandeja no está parada por un tope y el motor está en marcha, un detector no debe de estar viendo la bandeja más que el tiempo necesario para que la bandeja sobrepase el detector.

```
//Alarma por fallo de orientacion del detector B132
U      "B132"
U      "B124"
=      #Bit_Aux

CALL  "Gestion Atascos"
      detector      :=#Bit_Aux
      motor         :="K301"
      word_atasco   :=DBW0
      consigna      :=DBW0
      db_atasco     :="DB_Atasco"
      db_consigna   :="DB_Consigna_Atasco"
      defecto       :="Alarmas".ATASCO_B132
      memoria_defecto:= "Memoria_Atascos".Atascos[1]
```

2. Alarmas por transferencia.

En este caso, si una bandeja está en movimiento del tope TNA01 al tope TNA02 se realiza una memoria de transferencia, y la alarma relacionada consiste en que la marca de transferencia esté activa más tiempo que el que la bandeja necesita para llegar a su destino.

```
//Alarma por fallo de Transferencia de TNA01 A TNA02
U      "B124"
U      "K301"
UN     "Alarmas".ATASCO_B132
UN     "Alarmas".TR_TNA01_TNA02
=      "Activacion T14"

U      T      14
=      "Alarmas".TR_TNA01_TNA02
```

➤ FC100 – Tratamiento Temporizadores.

Esta función se encarga de activar los temporizadores del autómeta. Para llevar a cabo la activación de los mismos se utilizan marcas auxiliares, concretamente las marcas de la M100.0 a M115.7, que son exactamente 128 marcas. Por tanto esta función está preparada para la activación de 128 temporizadores (T0 a T127).

El tiempo de preselección de estos temporizadores estará almacenado en el bloque de datos DB100, de forma que la variación de la consigna resultará sencilla. Además al guardar estas consignas en un DB nos permitirá poder modificar las mismas desde el panel de operador.

La estructura del FC100 se divide en segmentos con diez temporizadores cada uno, a excepción de los segmentos 1 y 2 en los cuales se activa el T0 en el segmento 1 y los temporizadores T1 a T9 en el segmento 2.

Los temporizadores T0 y T1 están reservados, el resto son de uso libre. T0 se utiliza para generar bits de intermitencias en el byte de marcas MB1, y T1 se utiliza como tiempo de arranque de la instalación.

Segmento 1. T0 TRATAMIENTO DE INTERMITENCIAS A LA MB1.

```

AUF  "Val. preseleccion timer"
U    T      0
SPBN M001
L    "Byte de intermitencias"
INC  1
T    "Byte de intermitencias"
M001: NOP  0
UN   T      0
L    S5T#250MS
SE   T      0

```

Segmento 2. ACTIVACION TEMPORIZADORES T1-T9.

```

U    "Activacion T1"    //Reservado Arranque Instalación
L    DBW    2
SE   T      1
U    "Activacion T2"
L    DBW    4
SE   T      2
U    "Activacion T3"
L    DBW    6
SE   T      3
U    "Activacion T4"
L    DBW    8
SE   T      4
U    "Activacion T5"
L    DBW   10
SE   T      5
U    "Activacion T6"
L    DBW   12
SE   T      6
U    "Activacion T7"
L    DBW   14
SE   T      7
U    "Activacion T8"
L    DBW   16
SE   T      8
U    "Activacion T9"
L    DBW   18
SE   T      9

```

Las marcas utilizadas para activar los temporizadores (M100.0 a M115.7) como hemos podido ver en el código anterior tienen nombre simbólico "Activación Tx", siendo x el número de temporizador a activar.

La forma de activar un temporizador será por tanto tan simple como activar la marca que corresponda al temporizador deseado. A continuación mostramos un ejemplo:

```
//Alarma por fallo de Transferencia de TNA01 A TNA02
U      "B124"
U      "K301"
UN     "Alarmas".ATASCO_B132
UN     "Alarmas".TR_TNA01_TNA02
=      "Activacion T14"

U      T      14
=      "Alarmas".TR_TNA01_TNA02
```

➤ FC110 – Activación Manuales.

Esta función se encarga de realizar el control de la activación de los movimientos manuales. Desde el panel de operador se activan los movimientos manuales, y esta función se encarga de gestionar el funcionamiento de los led del panel de operador y de activar las marcas correspondientes a los movimientos manuales seleccionados en panel de operador.

En la tabla de abajo podemos ver las variables locales con las que trabaja la función:

Nombre	Tipo de Datos	Tipo Parametro
Número pantalla OP req	INT	Entrada
Cantidad de Movimientos	INT	Entrada
Bit Manual	BOOL	Entrada
DB_movimientos	Block_DB	Entrada
DB_areas_OP	Block_DB	Entrada
Aux_bit_manual	BOOL	Entrada / Salida
Aux_bit_OP	BOOL	Entrada / Salida
Aux_FP_K3	BOOL	Entrada / Salida
Aux_FP_K4	BOOL	Entrada / Salida
Aux_FN_K3	BOOL	Entrada / Salida
Bucle	Dword	Temp

Segmento 1. Control límite cantidad de movimientos.

```
U(
L      #Cantidad_de_movimientos
L      64
>I
)
O(
L      #Cantidad_de_movimientos
L      0
<I
)
BEB
```


El código anterior controla que el número de movimientos manuales sea el correcto, ya que el máximo número de movimientos que puede controlar la función es 64.

Segmento 2. Control estado manual.

Este segmento se encarga de poner a 0 los bits de activación de los movimientos manuales si la zona no está en manual.

```

U      #Bit_Manual
FN     #Aux_bit_manual
SPBN   Manu

L      0
LAR1

L      #Cantidad_de_movimientos
Next:  T      #Bucle
AUF    #DB_movimientos
R      DBX [AR1,P#6.0]
+AR1   P#0.1
L      #Bucle
LOOP   Next
Manu:  NOP    0

```

Segmento 3. Control cambio de pantalla.

El código de éste segmento tiene como función resetear los bit de activación de los led del OP al salir de la pantalla configurada en la llamada al módulo.

```

U      #Bit_Manual
U(
AUF    #DB_areas_OP
L      DBB    33                // Numero actual de pantalla en OP
L      #Numero_pantalla_OP_req
<>I
)
FP     #Aux_bit_op
SPBN   COP
R      DBX    42.4              // Bit 1 activación led K3 OP
R      DBX    42.5              // Bit 2 activación led K3 OP
R      DBX    42.6              // Bit 1 activación led K4 OP
R      DBX    42.7              // Bit 2 activación led K4 OP
COP:   NOP    0

```

Segmento 4. Control numero de pantalla.

Este segmento controla que el número de pantalla del OP sea el adecuado, y si no es así interrumpe la ejecución del resto del código del módulo.

```

U(
L      DBB    33                // Numero actual de pantalla en OP
L      #Numero_pantalla_OP_req
<>I
)
BEB

```

Segmento 5. Texto del movimiento.

Este segmento copia el valor del movimiento manual a la palabra de datos del DB que controla la lista de textos de movimientos manuales del OP.

```
AUF #DB_movimientos
L DBW 0
T DBW 2
```

Segmento 6. Acotamiento Word de movimientos.

Este segmento se encarga de hacer que la lista de movimientos se muestre en el OP de forma circular, ya que desde el OP se selecciona el número de movimiento introduciendo un valor, o sumando 1 o restando 1. Por tanto, este código tiene la función de acotar la lista entre los valores adecuados.

```
L #Cantidad_de_movimientos
L 1
-I
AUF #DB_movimientos
L DBW 0 // Numero de movimiento
<I
SPBN CM

L 0
T DBW 0 // Numero de movimiento
CM: NOP 0

U(
L DBW 0 // Numero de movimiento
L 0
<I
)
SPBN Cm

L #Cantidad_de_movimientos
L 1
-I
T DBW 0 // Numero de movimiento
Cm: NOP 0
```

Segmento 7. Activación modo mantenido.

Existen dos modos de funcionamiento manual:

- Mantenido: El movimiento se realiza mientras se mantiene presionado el pulsador.
- Impulsional: El movimiento se comienza al pulsar, y se finaliza al volver a pulsar.

Este segmento se encarga de activar o desactivar la marca que indica tipo de movimiento según la selección desde el OP.



```

AUF #DB_areas_OP
U DBX 40.6 // Tecla K3 en OP
FP #Aux_FP_K3 //FP tecla K3
SPBN M001

AUF #DB_movimientos
UN DBX 4.0 // Bit de modo mantenido
S DBX 4.0 // Bit de modo mantenido
SPB M001

U DBX 4.0 // Bit de modo mantenido
R DBX 4.0 // Bit de modo mantenido
M001: NOP 0

```

Segmento 8. Activación/Desactivación movimiento en función del índice.

Este segmento se encarga de setear o resetear el bit de activación del movimiento manual en función del número de movimiento que se encuentre seleccionado en el OP.

```

// Con FP de K4
AUF #DB_areas_OP
U DBX 40.7 // Tecla K4 en OP
AUF #DB_movimientos
FP #Aux_FP_K4
SPBN M002

L DBW 0 // Numero de movimiento
LAR1
UN DBX [AR1,P#6.0]
S DBX [AR1,P#6.0]
SPB M002
U DBX [AR1,P#6.0]
R DBX [AR1,P#6.0]
M002: NOP 0

// Con FN de K4
AUF #DB_areas_OP
U DBX 40.7 // Tecla K4 en OP
AUF #DB_movimientos
FN #Aux_Fn_K3 //FN tecla K4
SPBN M003

L DBW 0 // Numero de movimiento
LAR1
U DBX 4.0 // Bit de modo mantenido
U DBX [AR1,P#6.0]
R DBX [AR1,P#6.0]
M003: NOP 0

```

Segmento 9. Activación del Led K3.

Este segmento se encarga de activar el led K3 cuando nos encontramos en la pantalla configurada indicando de forma intermitente que tenemos seleccionado el modo de funcionamiento mantenido y de forma fija que el modo seleccionado es impulsional.

```

AUF #DB_movimientos
UN  DBX 4.0 // Bit de modo mantenido
AUF #DB_areas_OP
=   DBX 42.4 // Bit 1 activación led K3 OP

SET
=   DBX 42.5 // Bit 2 activación led K3 OP

```

Segmento 10. Activación del Led K4.

Este segmento se encarga de activar el led K4 cuando nos encontramos en la pantalla de movimientos manuales indicando de forma intermitente que el movimiento está permitido, de forma fija que el movimiento se está realizando y si está apagado que el movimiento no está permitido.

```

AUF #DB_movimientos
U   DBX [AR1,P#6.0] // Movimiento activado
AUF #DB_areas_OP
=   DBX 42.6 // Bit 1 activación led K4 OP

AUF #DB_movimientos
U   DBX [AR1,P#16.0] // Permiso movimiento
AUF #DB_areas_OP
=   DBX 42.7 // Bit 2 activación led K4 OP

AUF #DB_movimientos
UN  DBX [AR1,P#16.0] // Permiso movimiento
AUF #DB_areas_OP
R   DBX 42.6 // Bit 1 activación led K4 OP
R   DBX 42.7 // Bit 2 activación led K4 OP

```

➤ FC111 – Control puesto.

Esta función se encarga de realizar el control de los puestos de trabajo de la instalación.

En la tabla de abajo podemos ver las variables locales con las que trabaja la función:

Nombre	Tipo de Datos	Tipo Parámetro
Fin de Trabajo	BOOL	Entrada
Presencia en Puesto	BOOL	Entrada
Cruz de giro abajo	BOOL	Entrada
Tope Abajo	BOOL	Entrada
Presencia Tope Siguiente	BOOL	Entrada
Tope Siguiente Abajo	BOOL	Entrada
Tipo de Puesto	BOOL	Entrada
Consigna Tiempo Reset Puesto	INT	Entrada
Puesto Trabajo Ocupado	BOOL	Entrada / Salida
Salida Puesto Trabajo	BOOL	Entrada / Salida
Borrado Puesto Trabajo	BOOL	Entrada / Salida
Lámpara Pulsador	BOOL	Entrada / Salida
Memoria Cruz Giro ha Subido	BOOL	Entrada / Salida
Memoria Fin de Trabajo	BOOL	Entrada / Salida
Tiempo Reset Puesto	INT	Entrada / Salida

De los 6 puestos de trabajo que existen en la instalación hay dos que son diferentes, concretamente los del final de cada zona, ya que en lugar de tener un tope neumático tras ellos, hay un tope mecánico. Con el fin de que la función sea capaz de controlar todos los puestos pese a que no sean iguales existe un parámetro de entrada que indica al módulo el tipo de puesto.

Veamos ahora el código de la función:

Segmento 1. Gestión de memorias de trabajo.

En este segmento se memoriza que el puesto de trabajo está ocupado, de forma que se impida el paso de nuevas bandejas al puesto de trabajo, y se memoriza que la cruz de giro ha subido con el fin de indicar que el operario ha realizado el trabajo.

```
//Memoria de puesto de trabajo ocupado
U   #Presencia_en_puesto
UN  #Puesto_Trabajo_Ocupado
UN  #Tope_abajo
U   #Cruz_Giro_abajo
S   #Puesto_Trabajo_Ocupado

//Memoria Cruz de giro ha subido
U   #Puesto_Trabajo_Ocupado
UN  #Cruz_Giro_abajo
UN  #Mem_Cruz_Giro_ha_subido
S   #Mem_Cruz_Giro_ha_subido
```

Segmento 2. Control Fin de Trabajo.

En este segmento se memoriza el fin de trabajo y se activa las marcas auxiliares de salida de bandeja del puesto si hay condiciones de salida de bandeja, y borrado de memorias auxiliares del puesto si la bandeja ha terminado de salir.

```
//Activacion memoria fin trabajo
U   #Fin_Trabajo
U   #Mem_Cruz_Giro_ha_subido
U   #Cruz_Giro_abajo
UN  #Mem_fin_trabajo
S   #Mem_fin_trabajo

//Activacion marca de salida puesto trabajo
U   #Mem_fin_trabajo
U   #Presencia_en_puesto
UN  #Tipo_de_puesto
U   #Cruz_Giro_abajo
UN  #Presencia_Tope_siguiete
UN  "Peticon de parada"
UN  #Salida_Puesto_Trabajo
S   #Salida_Puesto_Trabajo

//Activacion marca de borrado puesto trabajo
U   #Mem_fin_trabajo
UN  #Presencia_en_puesto
U   #Cruz_Giro_abajo
UN  #Borrado_Puesto_Trabajo
S   #Borrado_Puesto_Trabajo
```

Segmento 3. Limpiar marcas puesto trabajo.

En este segmento se resetean las marcas auxiliares si la bandeja avanza al siguiente puesto, si han quitado la bandeja del puesto por haber finalizado las operaciones o si se realiza un reset del puesto.

```
//Si el puesto queda limpio
U      #Mem_fin_trabajo
UN     #Presencia_en_puesto
U      #Borrado_Puesto_Trabajo
U      #Cruz_Giro_abajo
UN     #Tope_abajo
SPB    Borr

//Si la bandeja sale hacia el siguiente tope
U      #Mem_fin_trabajo
UN     #Presencia_en_puesto
UN     #Tipo_de_puesto
U      #Cruz_Giro_abajo
U      #Presencia_Tope_siguiete
U      #Salida_Puesto_Trabajo
SPB    Borr

//Si quieren resetar puesto trabajo con pulsador
//Presionar el pulsador durante 10 segundos
U      #Fin_Trabajo
SPBN   Tim1
U(
L      #Tim_reset_puesto
L      #Consig_Tim_reset_puesto
<I
)
SPBN   Borr

L      "Ciclo PLC"
L      #Tim_reset_puesto
+I
T      #Tim_reset_puesto

SPA    Tim2
Tim1:  NOP    0
L      0
T      #Tim_reset_puesto

Tim2:  NOP    0

//Borrado marcas auxiliares puesto de trabajo
SPA    Fin1
Borr:  NOP    0
SET
R      #Puesto_Trabajo_Ocupado
R      #Salida_Puesto_Trabajo
R      #Borrado_Puesto_Trabajo
R      #Mem_Cruz_Giro_ha_subido
R      #Mem_fin_trabajo

Fin1:  NOP    0
```

Segmento 4. Control Lámpara pulsador Fin ciclo.

Este segmento controla la activación de la lámpara del pulsador de Fin de Ciclo de forma que si la lámpara está iluminada intermitentemente indica que se debe o se puede subir la cruz de giro, y si está iluminada de forma fija indica que se debe pulsar para validar el fin de Trabajo.

```
//Activación Lámpara Pulsador Fin de Ciclo
U      #Puesto_Trabajo_Ocupado
U      #Mem_Cruz_Giro_ha_subido
U(
UN     #Mem_fin_trabajo
U      #Cruz_Giro_abajo
U      "Int_500ms"
O
U      #Mem_fin_trabajo
)
O      "Prueba_Lamparas"
=      #Lampara_Pulsador
```

➤ **FC112 – Gestión de Atascos.**

Esta función se encarga de gestionar las alarmas de atasco de la instalación.

En la tabla de abajo podemos ver las variables locales con las que trabaja la función:

Nombre	Tipo de Datos	Tipo Parámetro
Detector	BOOL	Entrada
Motor	BOOL	Entrada
Word atasco	WORD	Entrada
Consigna	WORD	Entrada
DB Atasco	Block_DB	Entrada
DB Consigna	Block_DB	Entrada
Defecto	BOOL	Entrada / Salida
Memoria Defecto	BOOL	Entrada / Salida

Está función debe llevar asociada dos bloques de datos, concretamente el DB112 y DB113 que veremos con más detalle a lo largo del capítulo.

El funcionamiento de la función es muy simple. Si tenemos un detector y su motor, se cuenta el tiempo que el detector está viendo algo, y si se sobrepasa el tiempo de consigna, se activa una alarma y se memoriza.

Cuando hablamos de detector y de motor, no tiene por qué ser exactamente eso, sino que se puede pasar al módulo como parámetro una combinación de condiciones a través de una marca auxiliar. Por ejemplo, si tenemos un detector que ve una bandeja y el motor en marcha puede que no tengamos que dar fallo de atasco si existe un tope que pare la bandeja, por tanto la condición motor pasaría a ser una combinación de motor + tope abajo.

La memorización de la alarma se realiza para obligar al operario a liberar el detector y así evitar los daños que se pudieran causar por rearmar la alarma

si verificar cual es el problema, ya que cada rearme si no memorizamos la alarma supondría que el motor seguiría en marcha durante el tiempo de consigna pudiendo dañar si realmente existe el atasco algún elemento de la instalación.

Segmento 1. Cuenta Tiempo.

```
//Cuenta tiempo detector con presencia y motor en marcha
U      #detector          // = 1 presencia
U      #motor
SPBN   MT1
AUF    #db_atasco
L      #word_atasco
L      "Ciclo PLC"          //Ciclo del PLC
+I
T      #word_atasco
MT1:   NOP    0
```

Segmento 2. Pone a 0 el tiempo.

```
//Poner a 0 el tiempo si se pierde detector o se para motor
UN     #detector
ON     #motor
SPBN   MT2
AUF    #db_atasco
L      0
T      #word_atasco
MT2:   NOP    0
```

Segmento 3. Activa Alarma.

```
//Activación de Alarma si tiempo es mayor que consigna
AUF    #db_atasco
L      #word_atasco
AUF    #db_consigna
L      #consigna
>=I
O      #memoria_defecto
S      #defecto
S      #memoria_defecto
```

Segmento 4. Iguala Tiempo a Consigna.

```
//Si tiempo es mayor que consigna iguala tiempo a consigna
AUF    #db_atasco
L      #word_atasco
AUF    #db_consigna
L      #consigna
>I
SPBN   MT3
AUF    #db_consigna
L      #consigna
AUF    #db_atasco
T      #word_atasco
MT3:   NOP    0
```


Segmento 5. Reset Memoria Defecto.

```
//Si se libera el detector se resetea memoria defecto
UN    #detector
R     #memoria_defecto
```

➤ FC120 – Tratamiento de alarmas.

Esta función se encarga de realizar el tratamiento de alarmas de la instalación.

En la tabla de abajo podemos ver las variables locales con las que trabaja la función:

Nombre	Tipo de Datos	Tipo Parámetro
Pulsador Rearme	BOOL	Entrada
Pulsador Enterado	BOOL	Entrada
Dirección Inicio	WORD	Entrada
Cantidad word	WORD	Entrada
DB Alarmas	Block_DB	Entrada
DB Acuses	Block_DB	Entrada
DB Memoria Alarmas	Block_DB	Entrada
Sirena	BOOL	Entrada / Salida
Bit Alarma Activa	BOOL	Entrada / Salida
Bit Nueva Alarma Activa	BOOL	Entrada / Salida
Puntero_word	DWORD	Temp
Puntero_bluble	WORD	Temp

Esta función debe llevar asociada tres bloques de datos:

- DB120. Alarmas.
- DB121. Acuses de alarmas.
- DB122. Memorización de alarmas.

Esta función se suele llamar varias veces según las zonas que tengamos, aunque en nuestro caso debido a que la instalación es pequeña se llama una sola vez desde el OB1:

```
//Llamada función estándar Tratamiento de Alarmas
CALL "Tratamiento alarmas"
pulsador_rearme      := "REA"
pulsador_enterado    := "Areas".OP[1].Teclado_sistema.ACK
direccion_inicio     := 0
cantidad_word        := 25
db_alarmas           := "Alarmas"
db_acuses             := "Acuses"
db_mem_alarmas       := "Memoria_alarmas"
sirena                := "H784"
bit_alarma_activa    := "Alarma activada general"
bit_nueva_alarma_activa := "Nueva alarma general"
```

Segmento 1. Marca de Alarma Activa.

Este segmento tiene la función de buscar si existe alguna alarma dentro del área establecida entre la dirección inicial y la cantidad de palabras a explorar del DB de alarmas (DB120), y si es así activar una marca auxiliar “Alarma Activa”.

```
//Abrimos DB de alarmas
    AUF    #Db_alarmas

//Inicialización de punteros
    L     #direccion_inicio
    L     P#1.0
    *D
    T     #puntero_word

//Bucle de Búsqueda
    L     #cantidad_word
Next: T   #puntero_bucle
    L     DBW [#puntero_word]
    L     0
    <>I
    =     #Bit_alarma_activa
    SPB   Exit

    L     #puntero_word
    L     P#2.0
    +D
    T     #puntero_word

    L     #puntero_bucle
    LOOP Next
Exit: NOP 0
```

En este segmento podemos ver el uso de un puntero y de la instrucción “LOOP” combinados para realizar una búsqueda. El uso combinado de ambos se convierte en una herramienta muy potente a la hora de manejar cantidades grandes de datos. En este caso el puntero nos permite acceder de forma indirecta a una dirección del word del DB de alarmas y la instrucción LOOP hacerlo para la cantidad de palabras que debemos comparar.

Segmento 2. Tratamiento de sirenas.

En este segmento se activa la bocina cuando existe alguna alarma sin acusar y la instalación está en marcha. Además de la bocina también se activa una marca auxiliar que indica que existe una nueva alarma.

Para saber si existe una alarma nueva en este segmento se usa un DB auxiliar (Memoria de Alarmas, DB122) en que están memorizadas todas las alarmas existentes en la instalación en el anterior ciclo de Scan. Para realizar la comparación se invierte el estado de los bit de la palabra del DB122 y hace un AND_WORD (UW, y lógico con palabra) con la misma palabra del DB120, el resultado debe ser 0, de otro modo el estado de la memoria y el estado actual serían diferentes, y por tanto existiría una nueva alarma.

```

    U    "Fin secuencia inicial"
    SPBN JUMP
//Inicialización de punteros
    L    #direccion_inicio
    L    P#1.0
    *D
    T    #puntero_word

    L    #cantidad_word
NEXT:  T    #puntero_bucle
    AUF  #Db_mem_alarmas
    L    DBW [#puntero_word]
    INVI
    AUF  #Db_alarmas
    L    DBW [#puntero_word]
    UW
    L    0
    <>I
    S    #Bit_nueva_alarma_activa

    L    DBW [#puntero_word]
    AUF  #Db_mem_alarmas
    T    DBW [#puntero_word]

    U    #Bit_nueva_alarma_activa
    SPB  JUMP

    L    #puntero_word
    L    P#2.0
    +D
    T    #puntero_word

    L    #puntero_bucle
LOOP  NEXT
JUMP: NOP  0

    U    "Fin secuencia inicial"
    SPBN J2
    U    #Bit_nueva_alarma_activa
    UN   "Int_500ms"
    U    "Int_4s"
    =    #sirena

    U    #Bit_nueva_alarma_activa
    U    #pulsador_enterado
    R    #Bit_nueva_alarma_activa
J2:   NOP  0

```

Segmento 3. Desactivación de alarmas.

En este segmento se realiza el rearme de las alarmas, es decir, la puesta a cero de las mismas si se ha presionado el pulsador de rearme.

Para que una alarma se pueda borrar no basta con sólo presionar el pulsador de rearme. Por seguridad se obliga a que esta alarma haya sido enterada desde el panel de operador, de forma que nos aseguramos de que el operario ha leído el texto de la alarma.

Para saber si se ha enterado o no una alarma utilizaremos el DB auxiliar de acuses (DB121). Este DB es un área de interface entre el panel de operador y el PLC, y además tiene la misma estructura que el DB120 y que el DB122.

```

    U      #pulsador_rearme
    SPBN  RM
//Inicialización de punteros
    L      #direccion_inicio
    L      P#1.0
    *D
    T      #puntero_word

    L      #cantidad_word
NExt:  T      #puntero_bucle

    AUF   #Db_acuses
    L     DBW [#puntero_word]
    INVI
    AUF   #Db_alarmas
    L     DBW [#puntero_word]
    UW
    T     DBW [#puntero_word]

    L     #puntero_word
    L     P#2.0
    +D
    T     #puntero_word

    L     #puntero_bucle
    LOOP NExt
RM:    NOP  0

```

Bloques de Datos Globales (DBs).

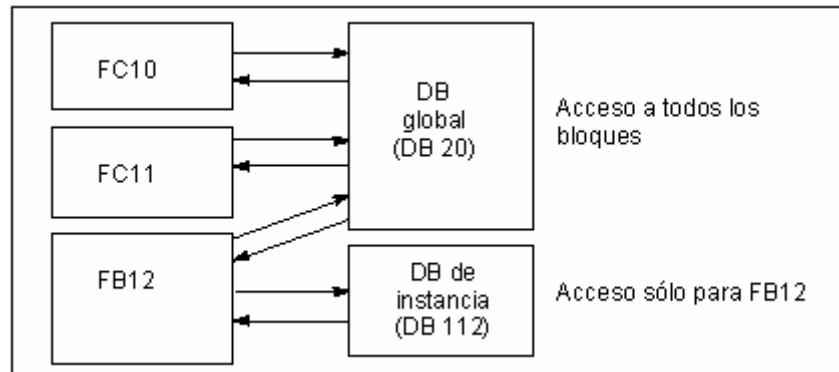
Al contrario de los bloques lógicos, los bloques de datos no contienen instrucciones STEP 7. En cambio, sirven para depositar datos de usuario, es decir, que los bloques de datos contienen datos variables con los que trabaja el programa de usuario. Los bloques de datos globales contienen datos de usuario utilizables desde otros bloques.

El tamaño de los DBs puede variar. El tamaño máximo admisible depende de la CPU utilizada.

Si se llama un bloque lógico (FC, FB o OB), éste puede ocupar determinada capacidad de memoria en el área de datos locales (pila L). Además de este área de datos locales, un bloque lógico puede abrir un área de memoria en forma de un DB. Al contrario de los datos en el área de datos locales, los datos contenidos en un DB no son borrados al cerrar el DB o al concluir el tratamiento del correspondiente bloque lógico.

Cada FB, FC o OB puede leer los datos de un DB global o escribir datos en un DB global. Estos datos se conservan en el DB incluso al abandonar dicho DB.

Un DB global y un DB de instancia pueden estar abiertos al mismo tiempo. La figura siguiente ilustra diferentes accesos a bloques de datos.



Los bloques de datos usados en el programa se pueden ver completos en el Anexo 3, no obstante a continuación realizaremos una breve descripción de su función:

➤ **DB50 – Remanencia.**

Este bloque se utiliza básicamente para memorizar las transferencias y otras marcas auxiliares que es importante mantengan su estado tras una pérdida de tensión, y por lo cual, tal y como está realizada la configuración de Hardware no pueden usarse marcas de programa normales.

➤ **DB100 – Valores Preselección Tiempos.**

Como ya habíamos comentado al hablar de la función FC100, este bloque de datos guarda los valores de preselección de los temporizadores del programa.

➤ **DB110 – Movimientos Manuales Zona 1.**

En este bloque de datos se guardan todos los datos necesarios para realizar los movimientos manuales de la zona 1.

➤ **DB111 – Movimientos Manuales Zona 2.**

En este bloque de datos se guardan todos los datos necesarios para realizar los movimientos manuales de la zona 2.

➤ **DB112 – Atascos.**

Este bloque tiene los datos referentes al tiempo actual que lleva un sensor detectando para poder dar el atasco al alcanzar el tiempo de consigna.

➤ **DB113 – Consignas Atascos.**

Este bloque tiene los datos referentes a la consigna de tiempo para considerar que hay un atasco.

➤ **DB114 – Memoria de Atascos.**

Este bloque tiene la función de memorizar los atascos que se han producido para evitar que se rearme la instalación sin solventar el problema que ha producido dicho atasco.

➤ **DB120 – Alarmas.**

Este bloque contiene todas las alarmas de la instalación.

➤ **DB121 – Acuses.**

Este bloque contiene las alarmas que han sido acusadas.

➤ **DB122 – Memoria Alarmas.**

Este bloque tiene la función de memorizar las alarmas existentes en la instalación con el fin de diferenciar en el siguiente ciclo de Scan si una alarma es nueva o ya existía.

➤ **DB123 – Áreas OP.**

Este DB hace de interface entre el panel de operador y el PLC guardando datos como el número de pantalla en el que se encuentra el OP, etc.

➤ **DB124 – Mensajes de servicio.**

Este bloque de datos guarda los mensajes de servicio que se dan en el panel de Operador.

5.4.- Entradas y Salidas de la Instalación.

Una parte muy importante antes de confeccionar el programa es definir el direccionamiento de entradas y salidas de los diferentes elementos de control de la instalación. A continuación, se muestra una tabla con las entrada y salidas de la instalación:

➤ **Salidas.**

Símbolo	Dirección	Tipo de Datos	Comentario
SH901	A 8.0	BOOL	Lámpara pulsador marcha instalación
H782	A 8.1	BOOL	Baliza verde marcha
H783	A 8.2	BOOL	Baliza roja de avería
H784	A 8.3	BOOL	Bocina
SH111	A 9.0	BOOL	Lámpara fin ciclo puesto 1A
SH131	A 9.1	BOOL	Lámpara fin ciclo puesto 2A
SH141	A 9.2	BOOL	Lámpara fin ciclo puesto 3A

Símbolo	Dirección	Tipo de Datos	Comentario
SH151	A 9.3	BOOL	Lámpara fin ciclo puesto 1B
SH161	A 9.4	BOOL	Lámpara fin ciclo puesto 2B
SH181	A 9.5	BOOL	Lámpara fin ciclo puesto 3B
Y1100	A 10.0	BOOL	Válvula general aire
K301	A 10.1	BOOL	Marcha motor M11 Línea A (CAC2)
K302	A 10.2	BOOL	Marcha motor M12 Línea B (CAC1)
Y1110	A 11.0	BOOL	BAJAR TOPE TNA 1
Y1111	A 11.1	BOOL	BAJAR TOPE TNA 2
Y1112	A 11.2	BOOL	BAJAR TOPE TNA 4
Y1113	A 11.3	BOOL	BAJAR TOPE TNA 5
Y1114	A 11.4	BOOL	BAJAR TOPE TNA 7
Y1115	A 11.5	BOOL	BAJAR TOPE TNA 8
Y1116	A 11.6	BOOL	BAJAR TOPE TNA 10
Y1117	A 11.7	BOOL	BAJAR TOPE TNA 11

➤ **Entradas.**

Símbolo	Dirección	Tipo de Datos	Comentario
PM	E 0.0	BOOL	Pulsador Marcha
PP	E 0.1	BOOL	Pulsador Parada
REA	E 0.2	BOOL	Rearme
Emergencia_Armario	E 0.3	BOOL	Emergencia Armario
Rele_General_Ok	E 0.4	BOOL	Relé Seguridad General
Prueba_Lamparas	E 0.5	BOOL	Pulsador Prueba Lámparas
Emergencia_X1B2	E 0.6	BOOL	Emergencia Botonera X1B2
Emergencia_X1B7	E 0.7	BOOL	Emergencia Botonera X1B7
Q301	E 1.0	BOOL	Térmico M11 Línea A (Cac2)
Q302	E 1.1	BOOL	Térmico M12 Línea B (Cac1)
F51	E 1.2	BOOL	Térmico Salidas
F71	E 1.3	BOOL	Térmico Entradas Campo
Presostato_general	E 1.7	BOOL	Presión De Aire
Fin_ciclo_puesto_1A	E 2.0	BOOL	Fin Ciclo Puesto 1a (X1b1)
Fin_ciclo_puesto_2A	E 2.1	BOOL	Fin Ciclo Puesto 2a (X1b3)
Fin_ciclo_puesto_3A	E 2.2	BOOL	Fin Ciclo Puesto 3a (X1b4)
B123	E 2.3	BOOL	Cruz Giro 1a Abajo (Cga1)
B124	E 2.4	BOOL	Tope Tna1 Abajo
B125	E 2.5	BOOL	Tope Tna2 Abajo, Tna3 Arriba
B126	E 2.6	BOOL	Cruz Giro 2a Abajo (Cga2)
B127	E 2.7	BOOL	Tope Tna4 Abajo
B130	E 3.0	BOOL	Tope Tna5 Abajo, Tna6 Arriba
B131	E 3.1	BOOL	Cruz Giro 3a Abajo (Cga3)
B132	E 3.2	BOOL	Presencia En Tope Tna1
B133	E 3.3	BOOL	Presencia En Tope Tna2
B134	E 3.4	BOOL	Presencia En Tope Tna3
B135	E 3.5	BOOL	Presencia En Tope Tna4
B136	E 3.6	BOOL	Presencia En Tope Tna5
B137	E 3.7	BOOL	Presencia En Tope Tna6
B140	E 4.0	BOOL	Presencia Cruz De Giro Cga3
Fin_ciclo_puesto_1B	E 5.0	BOOL	Fin Ciclo Puesto 1b (X1b5)

Simbolo	Dirección	Tipo de Datos	Comentario
Fin_ciclo_puesto_2B	E 5.1	BOOL	Fin Ciclo Puesto 2b (X1b6)
Fin_ciclo_puesto_3B	E 5.2	BOOL	Fin Ciclo Puesto 3b (X1b8)
B153	E 5.3	BOOL	Cruz Giro 1b Abajo (Cga4)
B154	E 5.4	BOOL	Tope Tna7 Abajo
B155	E 5.5	BOOL	Tope Tna8 Abajo, Tna9 Arriba
B156	E 5.6	BOOL	Cruz Giro 2b Abajo (Cga5)
B157	E 5.7	BOOL	Tope Tna10 Abajo
B160	E 6.0	BOOL	Tope Tna11 Abajo, Tna12 Arriba
B161	E 6.1	BOOL	Cruz Giro 3b Abajo (Cga6)
B162	E 6.2	BOOL	Presencia En Tope Tna7
B163	E 6.3	BOOL	Presencia En Tope Tna8
B164	E 6.4	BOOL	Presencia En Tope Tna9
B165	E 6.5	BOOL	Presencia En Tope Tna10
B166	E 6.6	BOOL	Presencia En Tope Tna11
B167	E 6.7	BOOL	Presencia En Tope Tna12
B170	E 7.0	BOOL	Presencia Cruz De Giro Cga6

5.5.- Gestión de Avisos y Alarmas.

La Gestión de Avisos y Alarmas es muy importante en cualquier instalación de automatización. Un buen sistema de automatización debe informar en todo momento de cualquier anomalía en la instalación que impida su funcionamiento total o parcialmente, y si es posible informar incluso de la solución al problema.

En nuestra instalación cuando una alarma o aviso se dispara el piloto rojo de la baliza del armario central luce de forma intermitente. El texto de la alarma activada aparecerá parpadeando reflejado en el terminal de operador OP7, situado en el cuadro principal. También parpadeará el led rojo de la OP7 indicando alarma activa no aceptada. Como aviso acústico la bocina sonará de manera intermitente hasta que se acepte la alarma.

Se debe consultar la alarma y después presionar el botón "ACK" del terminal de operador. De esta manera, la bocina dejará de sonar, el texto del terminal de operador dejará de parpadear y se mantendrá fijo y el led rojo se quedará fijo hasta que se rearme dicha alarma.

Una vez solucionado el problema se debe resetear la alarma, para ello, se pulsará el botón de rearme situado en el armario.

Estando en la pantalla inicial del terminal, podemos visualizar las alarmas actuales presionando la tecla ESC. Si existen varias, podemos desplazarnos para visualizarlas con las teclas "▲" o "▼". Para volver a la pantalla principal presionar la tecla ENTER.

A continuación podemos ver una tabla con las alarmas que puede darse en la instalación y las posibles soluciones:



Texto de alarma en terminal	Solución
RELE DE SEGURIDAD NO OK	Comprobar el origen de la alarma. El relé de seguridad general no está rearmado. Desenclavar emergencias, y pulsar rearme del armario principal.
EMERGENCIA CUADRO	Desenclavar la emergencia del cuadro descrita en la alarma y pulsar rearme del armario principal.
EMERGENCIA BOTONERA XXBX	Desenclavar la emergencia de la botonera descrita en la alarma y pulsar rearme del armario principal.
DEFECTO TÉRMICO XXXXXX	Comprobar el térmico que dio origen a la alarma en el cuadro principal, rearmarlo y pulsar rearme de la instalación en el armario principal.
FALLO BATERIA DE LA CPU	Defecto en el nivel de carga de la batería de la CPU. Cambiar la batería y rearmar en el armario principal.
DEFECTO PRESION DE AIRE EN LA INSTALACION	Defecto en la presión de aire en la instalación. Solucionar el problema y rearmar la alarma.
FALLO EN TRANSFERENCIA TNAXX>>>TNAXX ZONA X	Verificar que no hay atasco o si lo hay solucionarlo y rearmar la alarma. Acusar y rearmar la alarma.
ATASCO EN DETECTOR BXXX ZONA X	Fallo o atasco en el detector indicado. Solucionar el problema y rearmar en el armario principal.
DEFECTO MOVIMIENTO TOPE TNAXX ZONA X.	Fallo en el movimiento del tope indicado. Solucionar el problema y rearmar la alarma.
DEFECTO POSICION CRUZ DE GIRO CGAX ZONA X.	Fallo provocado por mal posicionamiento de la cruz de giro indicada. Solucionar el problema y rearmar la alarma.



UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CAPÍTULO 6

**PROGRAMA DEL
PANEL DE CONTROL**

6. PROGRAMA DEL PANEL DE CONTROL.

6.1.- El Software Protool.

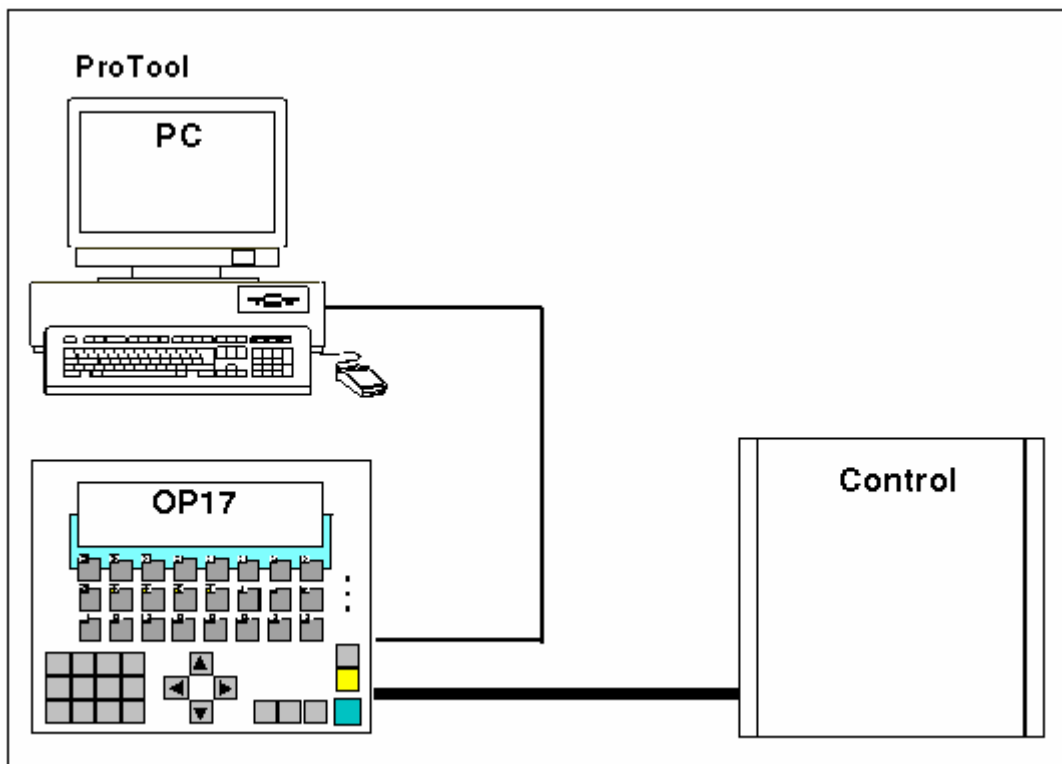
Para desarrollar el programa del Panel de Control de la instalación se necesita un software que suministra el fabricante, y que en nuestro caso es el "SIMATIC ProTool".

Por tanto, antes de seguir con la descripción del programa del panel de operador vamos a realizar una breve introducción al software de programación con el que se va a llevar a cabo. Nos referiremos en todo caso a la configuración de equipos de display de líneas ya que nuestro panel es de este tipo.

¿Qué es ProTool?

ProTool es el software de configuración para los equipos de líneas de la gama de equipos SIMATIC HMI. Todos los equipos pueden proyectarse con el mismo software de configuración independientemente del equipo de destino para el que cree su proyecto. ProTool es una aplicación de Windows con la superficie operativa totalmente gráfica, sencillamente haciendo clic con el ratón se crean proyectos orientados hacia el objeto. No son necesarios conocimientos especiales de programación.

A continuación, podemos ver una imagen de un ejemplo de estructura:



Ejemplo: PC como ordenador de configuración, OP 17 como unidad de operación

Los editores ofrecidos en ProTool pueden llamarse al mismo tiempo. Asimismo se pueden abrir al mismo tiempo diversos proyectos, también equipos de destino distintos, y transferir datos de un proyecto a través del portapapeles a otro proyecto.

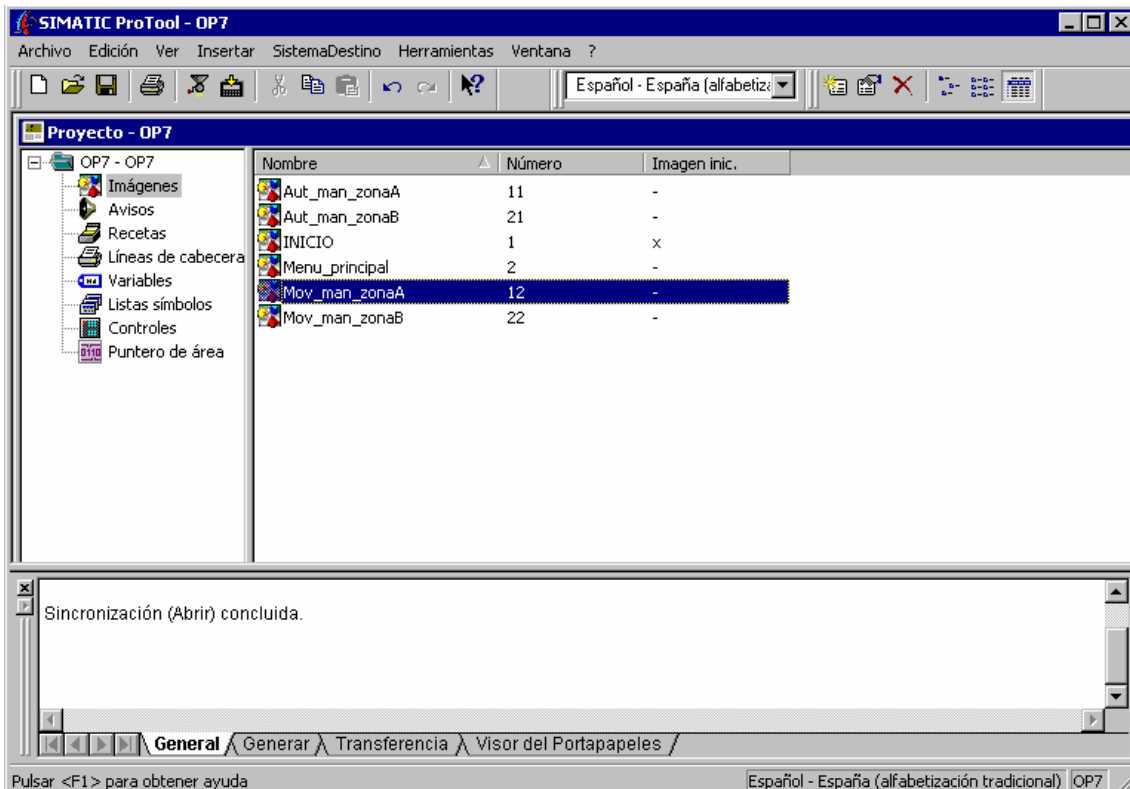
Además, ProTool se puede integrar en el software de configuración SIMATIC STEP 7. Esto permite seleccionar en ProTool símbolos y módulos de datos de SIMATIC STEP 7 como variables. De este modo no sólo se ahorra tiempo y dinero sino que también se evitan los posibles errores por la entrada repetida.

Con ProTool se crean y editan los proyectos offline. Por tanto se puede trabajar sin tener todavía disponible el equipo de destino. El ordenador de configuración representa los datos del proceso configurados del modo en que se visualizarán posteriormente en el equipo de destino.

Una vez concluida la fase de configuración se puede transferir el archivo del proyecto ejecutable desde el ordenador de configuración al equipo de destino.

6.2.- Estructura del Programa.

El programa del panel de control se estructura en 8 apartados distintos como podemos ver en la siguiente figura:

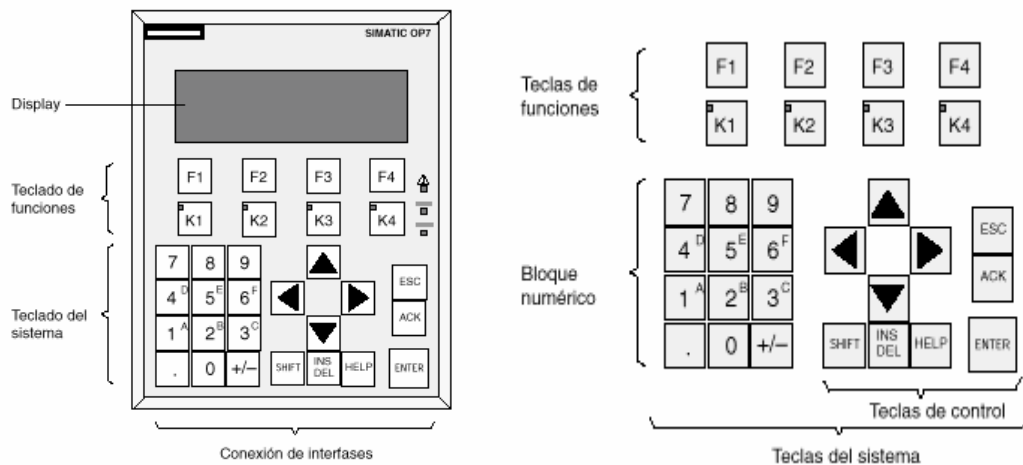


Veamos con más detalle para qué sirve cada uno de estos apartados:

- **Imágenes:** En este apartado se declaran las pantallas del panel de operador, dentro de las cuales se visualizan o modifican las variables del autómata.
- **Avisos:** En esta sección se declaran las alarmas y los avisos de servicio de la instalación.
- **Recetas:** En este apartado se crean recetas. La finalidad de las recetas es transferir al control varios datos interrelacionados "juntos" y de forma "síncrona".
- **Líneas de cabecera:** En este apartado se crean los pies de página y las cabeceras de impresión desde el panel de control.
- **Variables:** En esta sección se declaran las variables de los diferentes controles o del propio panel.
- **Lista de símbolos:** Se declaran listas de símbolos, en las que se asigna a un valor decimal de un bit o palabra un texto. En función del valor de una variable aparecerá un texto u otro.
- **Controles:** Se especifica aquí el tipo o los tipos de autómata que comunicarán con el panel de control.
- **Punteros de área:** En este apartado se declaran áreas de interface entre el panel de operador y el autómata. Por ejemplo, alarmas, teclado del sistema, etc.

A continuación se muestra una figura del software

En la siguiente imagen se puede ver la estructura del panel de operador OP7 que se utilizará en la instalación.



A continuación, veremos más detenidamente algunos apartados de los que se compone el programa:

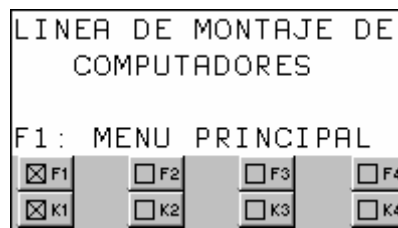
Imágenes.

El programa del panel de control de nuestra instalación está constituido por 6 imágenes:

Nombre	Número
Inicio	1
Menu_principal	2
Aut_man_zonaA	11
Mov_man_zonaA	12
Aut_man_zonaB	21
Mov_man_zonaB	22

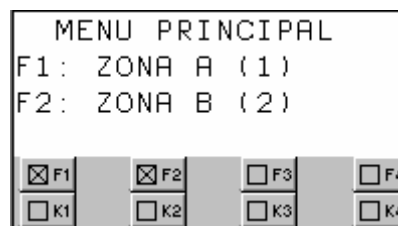
➤ Imagen 1 – Inicio.

Desde esta imagen se accede al menú principal presionando la tecla de función “F1” y a la pantalla de alarmas presionando la tecla de control “ESC”.



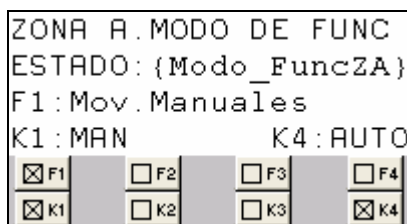
➤ Imagen 2 – Menú Principal.

Desde esta imagen se accede a las pantallas de control de las diferentes zonas o a la pantalla anterior presionando la tecla “ESC”.



➤ Imagen 11/21 – Automático / Manual Zona A/B.

Desde esta imagen podemos ver el estado de la zona a través de una lista de textos asociada a una variable del PLC (Modo_FuncZA), y además se puede observar el modo de funcionamiento seleccionado a través de los leds de las teclas “K1” y “K4”.



➤ Imagen 12/22 – Movimientos Manuales Zona A/B.

Desde esta imagen podemos realizar los movimientos manuales de la zona seleccionada.

MOV. MANUALES ZONA A			
MOV: {N_mov_ZA}			
MOV.Num: {N_mov_ZA}			
-1	+1	M/I	ACTV
<input checked="" type="checkbox"/> F1	<input checked="" type="checkbox"/> F2	<input type="checkbox"/> F3	<input type="checkbox"/> F4
<input type="checkbox"/> K1	<input type="checkbox"/> K2	<input type="checkbox"/> K3	<input type="checkbox"/> K4

Nota: Para más información sobre el panel de control y su manejo referirse al “Anexo 4. Programa Panel de Operación” y “Anexo 5. Manual de Usuario”.

Avisos.

En este apartado se introducen los textos de las alarmas y de los avisos de servicios, en nuestro caso únicamente de las alarmas, ya que no se ha utilizado ningún aviso de servicio. El texto se corresponde a un bit del “puntero de área alarmas” que coincidirá con un bit del bloque de datos de alarmas “DB120” utilizado en el PLC.

Variables.

Como habíamos mencionado con anterioridad en este apartado se definen las variables que se utilizarán como interconexión con el autómatas, que en nuestro caso son:

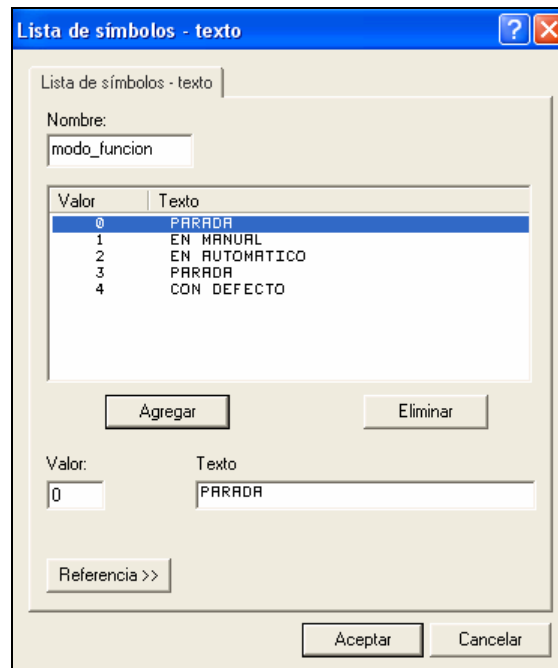
Nombre	Tipo	Co...	Dirección	Tiempo polling	Lectura continua	Número de elementos	Conversión	Funciones
Automatico_OP_ZONA_A	BOOL	PLC_1	DB 123 DBX 100.1	1.0	-	1	-	-
Automatico_OP_ZONA_B	BOOL	PLC_1	DB 123 DBX 120.1	1.0	-	1	-	-
Manual_OP_ZONA_A	BOOL	PLC_1	DB 123 DBX 100.0	1.0	-	1	-	-
Manual_OP_ZONA_B	BOOL	PLC_1	DB 123 DBX 120.0	1.0	-	1	-	-
Modo_Funcionamiento_ZONA_A	BYTE	PLC_1	DB 123 DBB 101	1.0	-	1	-	-
Modo_Funcionamiento_ZONA_B	BYTE	PLC_1	DB 123 DBB 121	1.0	-	1	-	-
Modo_FuncZA	INT	PLC_1	DB 1 DBW 0	1.0	-	1	-	-
N_mov_ZA	INT	PLC_1	DB 1 DBW 0	1.0	-	1	-	-
N_mov_ZONA_A	INT	PLC_1	DB 1 DBW 0	1.0	-	1	-	-
Num_mov_ZONA_A	INT	PLC_1	DB 1 DBW 0	1.0	-	1	-	-
Num_movimiento_ZONA_A	WORD	PLC_1	DB 110 DBW 0	1.0	-	1	-	-
Num_movimiento_ZONA_B	WORD	PLC_1	DB 111 DBW 0	1.0	-	1	-	-
Numero_movimiento_ZONA_A	WORD	PLC_1	DB 110 DBW 0	1.0	-	1	-	-
Numero_movimiento_ZONA_B	WORD	PLC_1	DB 111 DBW 0	1.0	-	1	-	-

Listas símbolos.

En nuestro proyecto se ha definido 3 listas de símbolos:

Nombre	Tipo
modo_funcion	Decimal
MovManZ1	Decimal
MovManZ2	Decimal

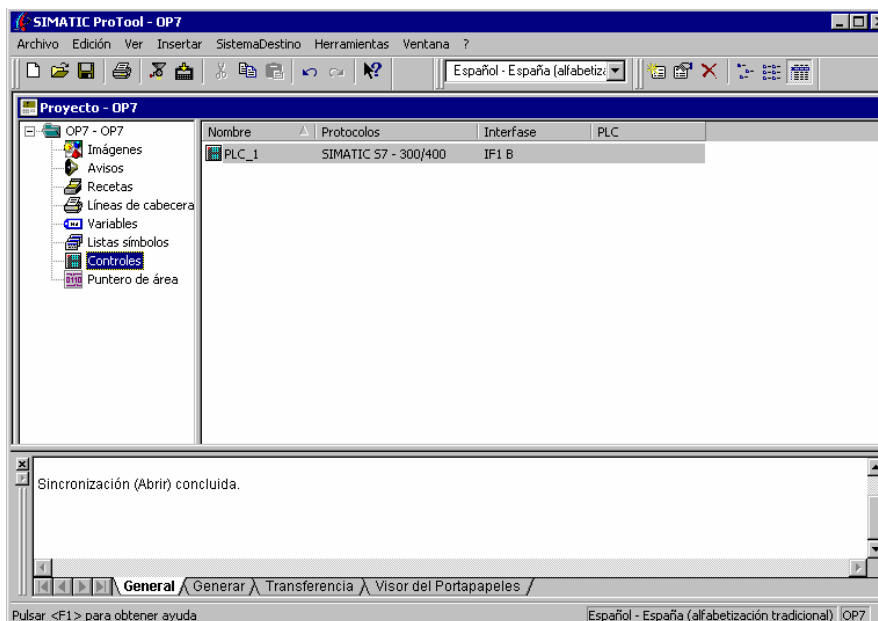
Si pinchamos dos veces con el ratón sobre una lista de símbolos se nos abre una ventana como la de la siguiente figura desde la cual podemos editar, ver o modificar los textos y el valor decimal que actúa como índice para mostrar dicho texto en el panel de operador.



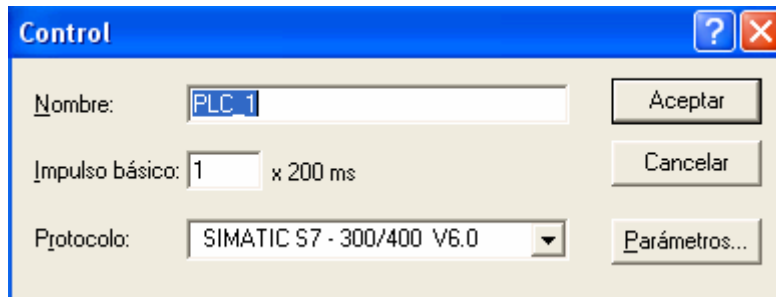
Controles.

Unos de los primeros pasos a la hora de realizar el programa del panel de operador es definir cual es el controlador (PLC) con el que debe comunicar. Esto se realiza desde esta sección.

Un panel de operador tiene la posibilidad de comunicar con varios controles a la vez, aunque en nuestro caso sólo tenemos un control, que hemos llamado "PLC_1" tal y como se muestra en la siguiente figura:



Si pinchamos dos veces sobre el control “PLC_1” se nos abrirá una pantalla desde la cual podemos editar las propiedades del mismo:



Control

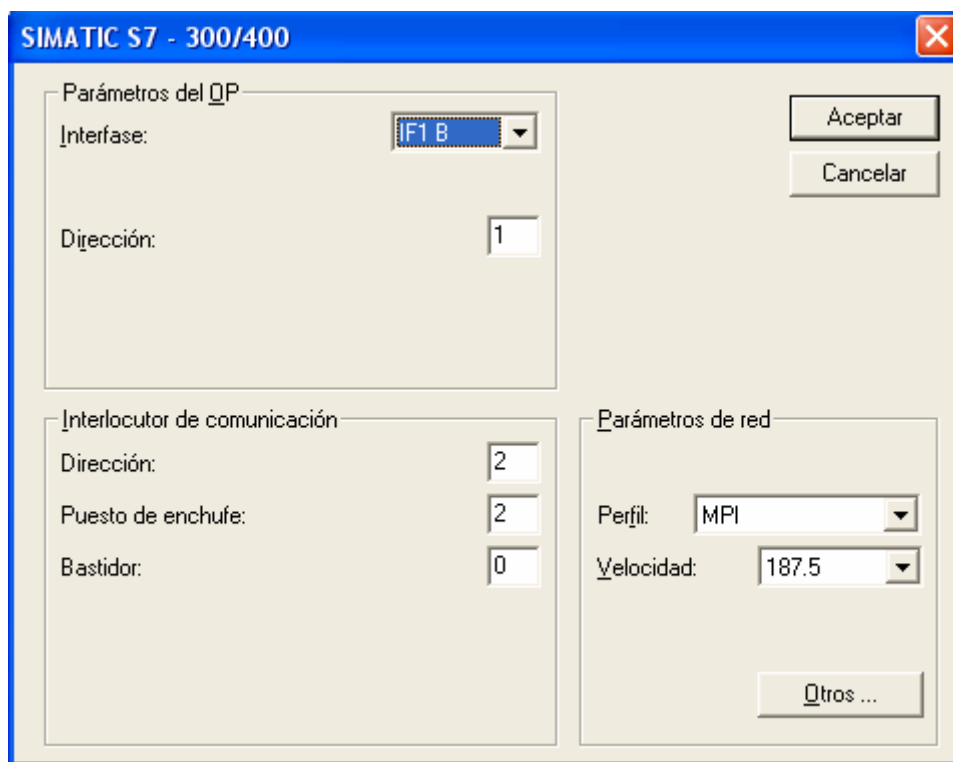
Nombre:

Impulso básico: x 200 ms

Protocolo:

En nuestro caso el control se debe configurar con el protocolo “SIMATIC S7 – 300/400 V6.0” puesto que nuestro autómeta es un S7-300.

Pero para que se realice correctamente la comunicación no basta con seleccionar el protocolo, si no que se deben definir otros parámetros, para lo cual se debe pinchar sobre el pulsador “Parámetros” de esta pantalla, accediendo a la configuración de los parámetros del control:



SIMATIC S7 - 300/400

Parámetros del OP

Interfase:

Dirección:

Interlocutor de comunicación

Dirección:

Puesto de enchufe:

Bastidor:

Parámetros de red

Perfil:

Velocidad:

En esta pantalla debemos seleccionar el Interfase (puerto del OP en el que se conectará el PLC), la dirección del OP, la dirección del PLC, el puesto de enchufe del PLC (posición que ocupa en el Bastidor o Rack), Bastidor o Rack donde se encuentra el PLC, el tipo de red de comunicación y la velocidad de la misma.

Punteros de Área.

En la siguiente figura se pueden ver los punteros de área declarados en el panel de operador:

Tipo	Nº	Control	Dirección	Long. [Palabras]	Ciclo de registro [s]	Avisos
Acuses OP	1	PLC_1	DB 121 DBW 0	32	-----	1-499
Acuses PLC	1	PLC_1	DB 120 DBW 64	32	1.0	1-499
Alarmas	1	PLC_1	DB 120 DBW 0	32	1.0	1-499
Avisos de servicio	1	PLC_1	DB 124 DBW 0	32	1.0	1-500
Imagen de los LED		PLC_1	DB 123 DBW 42	1	1.0	-----
Número de la imagen		PLC_1	DB 123 DBW 32	2	-----	-----
Teclado de funciones	1	PLC_1	DB 123 DBW 40	1	-----	-----
Teclado del sistema		PLC_1	DB 123 DBW 36	2	-----	-----
Área de interfase		PLC_1	DB 123 DBW 0	16	-----	-----

De todos los punteros declarados realmente en nuestra instalación solo usaremos 4, el resto están declarados por si fuese necesario usarlos en un futuro. Veamos a continuación cuales son los punteros usamos:

➤ Acuses OP.

De esta área hemos hablado en el capítulo anterior al comentar el funcionamiento de la función FC120, pero resumiendo sirve de interface entre el panel de operador y el PLC para indicar el estado de las alarmas activas en el panel de operador, es decir, si estas alarmas han sido vistas (acusadas con la tecla "ACK") o no.

➤ Alarmas.

Define la dirección de memoria inicial del PLC y la longitud en palabras donde deben estar ubicados los bit de alarma.

➤ Imagen de los leds.

Define la dirección de memoria del PLC que se debe activar desde el mismo para controlar el estado de los leds del panel de control.

➤ Número de la imagen.

En este área el panel de control escribe en número de imagen qué se está mostrando por pantalla en cada momento.

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CAPÍTULO 7

**PROYECTO
ELÉCTRICO**

7. PROYECTO ELÉCTRICO.

7.1.- Introducción.

Pese a que como, se había comentado con anterioridad, el alcance del proyecto no incluye el proyecto eléctrico, ya que es uno de los datos de entrada entregados por el cliente, se ha creído conveniente incluir este capítulo.

En los siguientes apartados veremos brevemente los cálculos de dimensionamiento de los motores, de secciones de conductores, de protecciones de la instalación, una descripción de la estructura del proyecto, el software con el que se ha desarrollado, el layout eléctrico y algunas fotos de la instalación.

Para estos cálculos se tendrán en cuenta las siguientes normas previstas en el actual Reglamento Electrotécnico de Baja Tensión establecido en el Real Decreto 842/2002 [10].

Caídas de Tensión

La instrucción BT 019, según la cual la sección de los conductores a utilizar será tal que entre el origen de la instalación y cualquier punto de ella, la caída de tensión será menor del 3 % en el caso de alumbrado, y menor del 5 % para los demás usos.

Cálculo de secciones y Caídas de tensión

Para el cálculo de estos circuitos se tendrán en cuenta las siguientes premisas:

- Conductor de Cu.
- Caída de tensión menor del porcentaje indicado anteriormente.
- Tensión de servicio 400 V.
- Frecuencia 50 Hz.

Fórmulas utilizadas para un Circuito Trifásico

$$P = \sqrt{3} \cdot U \cdot I \cdot \cos \varphi \quad \Rightarrow \quad I = \frac{P}{\sqrt{3} \cdot U \cdot \cos \varphi}$$

$$S = \frac{\sum P \cdot L}{\gamma_{Cu} \cdot \Delta U \cdot U_N} \quad I_{N_{\max}} = \frac{P}{\sqrt{3} \cdot (U_N - \Delta U_{\max}) \cdot \cos \varphi}$$

Fórmulas utilizadas para un Circuito Monofásico

$$P = U \cdot I \cdot \cos \varphi \quad \Rightarrow \quad I = \frac{P}{U \cdot \cos \varphi}$$

$$S = \frac{2 \cdot \sum P \cdot L}{\gamma_{Cu} \cdot \Delta U \cdot U_N} \quad I_{N_{\max}} = \frac{P}{(U_N - \Delta U_{\max}) \cdot \cos \varphi}$$

Intensidades máximas admisibles en los conductores

La instrucción BT 019, según la cual, las intensidades máximas admisibles en las distintas partes de la instalación y naturaleza de los conductores, serán las indicadas en la tabla 1.

Conductores de protección

La instrucción BT 019, según la cual, para los conductores de protección que estén constituidos por el mismo metal que los conductores de fase o polares, tendrán una sección mínima igual a la fijada en la tabla 2, en función de la sección de los conductores de fase o polares de la instalación.

Protecciones contra sobrecargas y cortocircuitos

La instrucción BT 022, según la cual, la protección contra posibles defectos, se hará mediante interruptores automáticos con sistema de corte omnipolar.

Estos dispositivos se instalarán en el origen de cada circuito.

Protecciones contra contactos directos

La instrucción BT 024, según la cual, la protección contra contactos directos se realizará tal que la instalación y receptores estén dispuestos de tal forma que se impida el contacto con las partes activas de los mismos, recubriendo las partes activas con aislantes eficaces o instalándolas en zonas difícilmente accesibles.

Protecciones contra contactos indirectos

La instrucción BT 024, según la cual, la protección contra contactos indirectos tendrá en cuenta la naturaleza del local, la extensión e importancia de la instalación y la tensión de la instalación con respecto a tierra.

Instalación de receptores. Motores.

La instrucción BT 047, según la cual, los conductores de conexión que alimentan a un solo motor deben estar dimensionados para una intensidad del 125 % de la intensidad a plena carga del motor.

7.2.- Cálculo de la potencia de los motores.

Antes de continuar con el cálculo de los motores, cabe destacar que normalmente no suele realizarse, ya que los fabricantes de motores ponen a disposición de sus clientes herramientas de software con una base de datos de sus motores, que se encarga de realizar la elección de motor y reductores adecuados a las necesidades la instalación por nosotros.

Debido a esto es posible que los datos del motor que realmente se ha montado no coincidan con los cálculos teóricos.

Nuestra instalación tiene únicamente **dos motores de la marca “SEW”**. A continuación realizaremos el cálculo de uno de ellos puesto que la función de ambos motores es la misma y por tanto deben tener las mismas características.

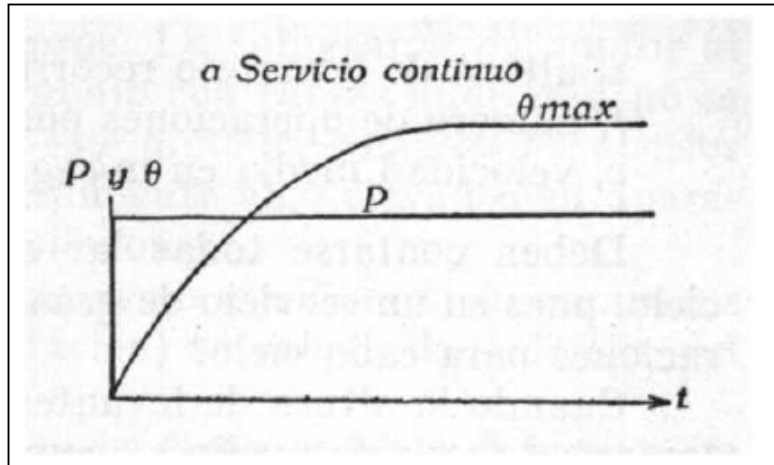
Antes de determinar la potencia debe escogerse el **tipo de motor en función del tipo de servicio** que vaya a realizar; el primer paso en la elección del motor consistirá en determinar si el motor debe ser: abierto, protegido o cerrado [9].

1. **Construcción abierta:** es más barata y tiene la ventaja de una fácil conservación ya que inducido, cojinetes y escobillas son fácilmente accesibles. El módulo de inercia es mínimo. Estos motores no se pueden utilizar ni en intemperie ni en atmósferas poco favorables (húmedas o polvorientas).
2. **Construcción protegida:** mismas ventajas que la construcción abierta, se suele proteger a los motores contra goteo, se pueden emplear en intemperie si la protección contra la lluvia es total incluso suponiendo una dirección de lluvia casi horizontal.
3. **Construcción cerrada:** ideales para intemperie o interiores con atmósferas desfavorables (polvorientas, húmedas o cargadas de ácidos), zonas donde puedan tener lugar proyecciones de agua u objetos o siempre que se requiera una construcción especialmente robusta.

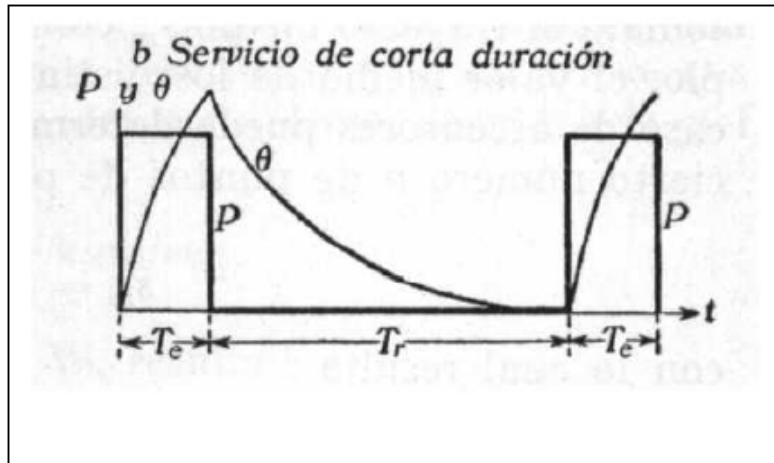
Para la **elección del motor** deberá también tenerse en cuenta el número de revoluciones, habrá que elegir velocidades normales de serie, las velocidades anormales encarecen la instalación y dificultan posteriormente las sustituciones. Como regla general la potencia de un motor es tanto más grande cuanto mayor sea el número de revoluciones [9].

Para la **elección del tamaño** de los motores deberá tenerse en cuenta el tipo de servicio que van a realizar, según la normativa para máquinas eléctricas, se distinguen las tres formas de trabajo siguientes [9]:

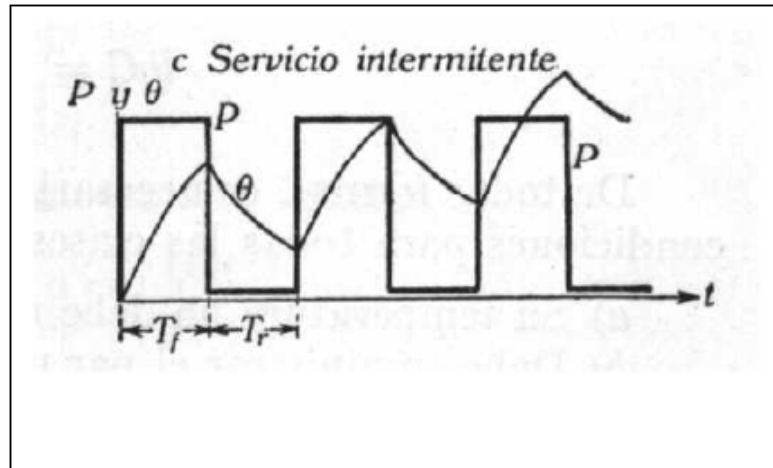
1. **Servicio permanente o continuo:** El motor está funcionando constantemente o por lo menos durante algunas horas con plena carga, alcanzando así su temperatura final, T_{max} . Esta temperatura no debe sobrepasar el límite fijado por la normativa.



2. **Servicio de corta duración:** La carga actúa con toda intensidad durante un corto tiempo, a este estado le sigue una marcha en vacío o la desconexión, que da tiempo al enfriamiento del motor. La potencia nominal en este tipo de servicio será aquella que puede suministrar el motor durante el tiempo convenido sin calentamiento excesivo (Por ejemplo 50 Kw. en 15 min.).



3. **Servicio intermitente:** Alternan el tiempo de funcionamiento que llamaremos t_f , y el tiempo de reposo que llamaremos t_r , el tiempo de ciclo que llamaremos t_c , será la suma de ambos ($t_c = t_f + t_r$), dicho t_c no debe sobrepasar de un tiempo t que permita al motor enfriarse completamente, de esta forma la temperatura va aumentando escalonadamente hasta el valor final que tardará más tiempo en alcanzarse que en el régimen continuo.



En este tipo de servicio una magnitud muy importante es el tiempo de conexión **TC**, que nos da la relación entre el tiempo de funcionamiento y el tiempo de ciclo.

$$TC = \frac{t_f}{t_f + t_r} * 100 \quad (\%)$$

El valor exacto de TC sólo puede determinarse mediante un diagrama de trabajo. Para aparatos de elevación puede aproximarse el valor de TC con la siguiente fórmula:

$$TC = \frac{f \cdot s}{36 \cdot v}$$

siendo:

s, altura de elevación o recorrido en m. en cada ciclo.

f, número de operaciones por hora.

v, velocidad media en m/seg.

Deben contarse todas las operaciones realizadas durante cada ciclo, así por ejemplo en un servicio de grúa se estarán efectuando varias operaciones por cada ciclo.

Cuando la altura de elevación o la longitud del trayecto varían, se toma **s_m** (trayecto medio) como valor de **s**; en las grúas por ejemplo **s_m** será el valor medio de los distintos levantes durante un ciclo, en el caso de ascensores puede determinarse aproximadamente **s_m** para un cierto número **n** de puntos de parada, mediante la fórmula:

$$s_m = \frac{n \cdot s}{2 \cdot (n - 1)}$$

con lo cual resulta:

$$TC = \frac{f \cdot n \cdot s}{72 \cdot v \cdot (n-1)}$$

El motor para todas las clases de servicio debe cumplir siempre que:

- Su temperatura nunca debe rebasar la fijada por los reglamentos.
- Debe suministrar el par motor requerido por el dispositivo que se pretende accionar.

Según el reglamento: {

UN MOTOR PARA SERVICIO CONTINUO
Debe suministrar un par máximo: $M_{\max}=1,6 \cdot M_n$

UN MOTOR PARA SERVICIO INTERMITENTE
Debe suministrar un par máximo: $M_{\max}=2 \cdot M_n$

PROCEDIMIENTO PARA DETERMINAR LA POTENCIA DEL MOTOR

Existen diferentes procedimientos para calcular la potencia del motor, dependiendo de la exactitud que necesitemos.

- exactitud ↓
- a. Cálculo de la potencia para plena carga.
 - b. Determinación de la potencia mediante diagrama de trabajo teórico o normalizado.
 - c. Determinación de la potencia mediante diagrama de trabajo exacto.
- + exactitud ↓

Estos procedimientos están ordenados según el grado de exactitud de los resultados.

a. CÁLCULO DE LA POTENCIA DEL MOTOR PARA PLENA CARGA [9]

Dependiendo del uso que se le vaya a dar al motor existen diferentes fórmulas para calcular la potencia.

A.1. APARATOS DE ELEVACIÓN SIN CONTRAPESO

El problema que se plantea es el de elevar una carga de Q [Kg.], desplazándose con una velocidad de v [m/seg.]; la potencia que debe desarrollar la máquina en esta situación es $P = v \cdot Q$ [Kg m/seg.]. Si llamamos η al rendimiento de la máquina, la potencia del motor se puede escribir como:

$$P = \frac{Q \cdot v}{\eta} \text{ [Kg} \cdot \text{m/seg]}$$

como 1 [Kg m/seg.] equivale a 9,81 [w.], resulta:

$$P = \frac{Q \cdot v}{102 \cdot \eta} \text{ [Kw.]}$$

el par motor será igual a:

$$P = M \cdot n \quad P = M \cdot \frac{2 \cdot \pi}{60} \cdot n$$

$$M = \frac{P}{n \cdot \frac{\pi}{30}} \text{ [N} \cdot \text{m.]} = \frac{P}{n \cdot \left(\frac{\pi}{30} \cdot 9,81 \right)} \text{ [m} \cdot \text{Kg.]}$$

$$M = 975 \cdot \frac{P}{n} \text{ [m} \cdot \text{Kg.]} = 9,55 \cdot \frac{Q \cdot v}{n \cdot \eta} \text{ [m} \cdot \text{Kg.]}$$

donde **P** es la potencia del motor en [Kw.] y **n** es la velocidad de giro en [r.p.m.].

A.2. APARATOS DE ELEVACIÓN CON CARGA EQUILIBRADA

Ascensores por ejemplo, en los que la carga aparece equilibrada con otra que se denomina contrapeso.

$$\text{Potencia del motor: } P = \frac{Q \cdot v}{102 \cdot \eta} \cdot (1 - a)$$

$$\text{Par motor: } M = 9,55 \cdot \frac{Q \cdot v}{n \cdot \eta} \cdot (1 - a)$$

siendo $a = \frac{G_1 - G}{Q}$ el factor de equilibrado de la carga.

G₁: representa el contrapeso.

G: representa el peso muerto del lado de la carga (en un ascensor, el peso de la cabina + el marco de suspensión).

A.3. MECANISMO DE TRASLACIÓN

Siendo **G** el peso total de las masas en movimiento y **R** $\left[\frac{\text{Kg}}{\text{Kg}} \right]$ la resistencia de rodamiento, el esfuerzo de tracción necesario, será igual a:

$$F_t = G \cdot R \text{ [Kg.]}$$

Suponiendo una velocidad de desplazamiento v [m/seg.], se obtienen las fórmulas siguientes:

$$\text{Potencia del motor: } P = \frac{F_t \cdot v}{102 \cdot \eta} \text{ [Kw.]}$$

$$\text{Par motor: } M = 975 \cdot \frac{P}{n} = 9,55 \cdot \frac{F_t \cdot v}{n \cdot \eta} \text{ [m \cdot Kg.]}$$

b. CÁLCULO DE LA POTENCIA DEL MOTOR TOMANDO COMO BASE UN DIAGRAMA DE TRABAJO TEÓRICO O NORMALIZADO [9]

Este método se suele utilizar para mecanismos de elevación y de traslación de grúas, eligiendo la aceleración (que dará lugar a la elección del motor) con arreglo a los siguientes supuestos:

- a. El frenado se efectúa siempre desconectando el aparato de mando, o sea, siempre con frenado mecánico, de forma que el motor queda sin corriente.
- b. Para la aceleración de las masas se supone el doble del par nominal.
- c. Las curvas de velocidades corresponden a la mayor parte de los tipos de motores tanto para corriente continua como para alterna trifásica.
- d. Se han considerado recorridos iguales para la elevación y el descenso de la carga, para el cálculo se necesita además la relación α entre cargas.

$$\alpha = \frac{\text{Par motor para: (elevación ó desplazamiento de la carga)}}{\text{Par motor para: (elev. ó desplaz. de la carga + carga en vacío)}}$$

La carga en vacío estará constituida por el gancho, la cuchara, el recipiente de carga, etc. La aceleración está representada por un factor ε que tiene en cuenta el efecto de las masas. El valor de ε puede ser determinado por la fórmula siguiente:

$$\varepsilon = \frac{0,28}{1000} * \frac{\sum GD^2 \cdot n^2 \cdot \eta}{Q \cdot s}$$

Como estamos utilizando la carga Q debemos suponer que hablamos de un movimiento de elevación, en el caso de que el movimiento fuera de traslación, se sustituirá la carga Q por el esfuerzo de tracción $F_t = G \cdot R$ que es función de la resistencia de rodamiento.

Una vez determinado ε , debemos entrar en las curvas de γ constante para la determinación del tamaño del motor; estas curvas que aparecen en la página siguiente, corresponden: las dos superiores a un movimiento de elevación y las dos inferiores a un movimiento de traslación. Debemos deducir el coeficiente γ que nos permitirá calcular el par motor suficiente desde el punto de vista del calentamiento, así como de la potencia media en función de las siguientes fórmulas:

$$M = 9,55 \cdot \frac{Q \cdot v}{n \cdot \eta} \cdot \gamma \quad [m \cdot Kg.]$$

$$P = \frac{Q \cdot v}{102 \cdot \eta} \cdot \gamma \quad [Kw.]$$

Estas fórmulas únicamente se distinguen de las anteriormente vistas en el factor de corrección γ , dicho factor aparece en el eje de ordenadas de las curvas de la figura que aparece en la hoja siguiente.

Entramos en las curvas con el valor de ε en el eje de abscisas (x) nos desplazamos hasta la curva de α y obtenemos el valor de γ sobre el eje de ordenadas (y).

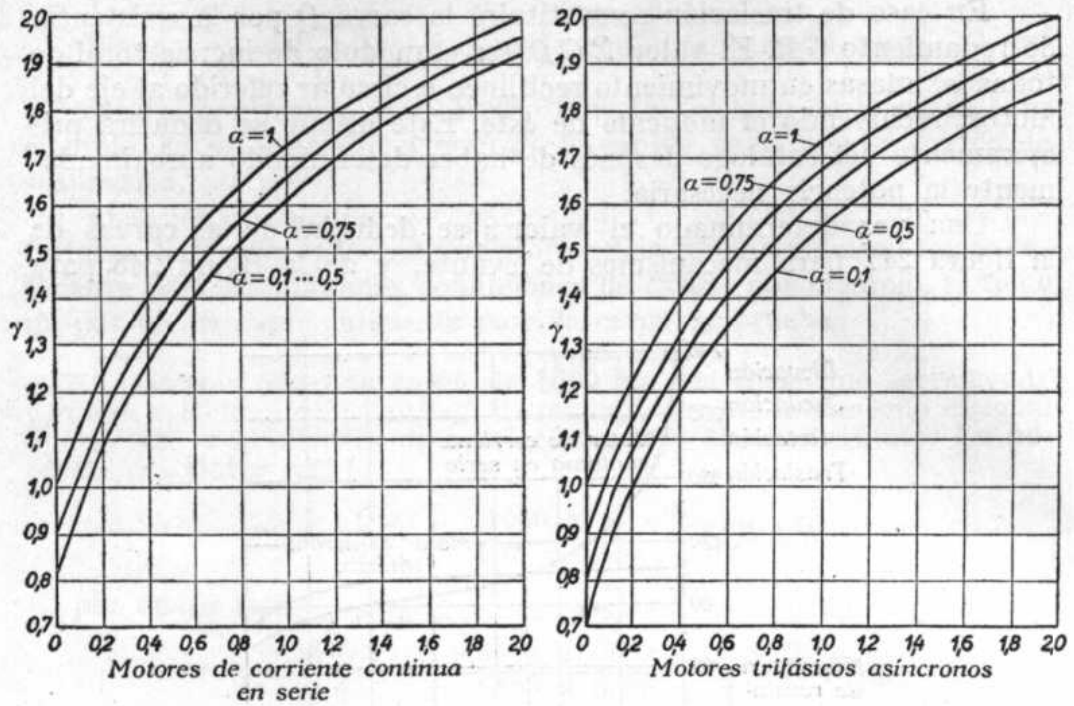
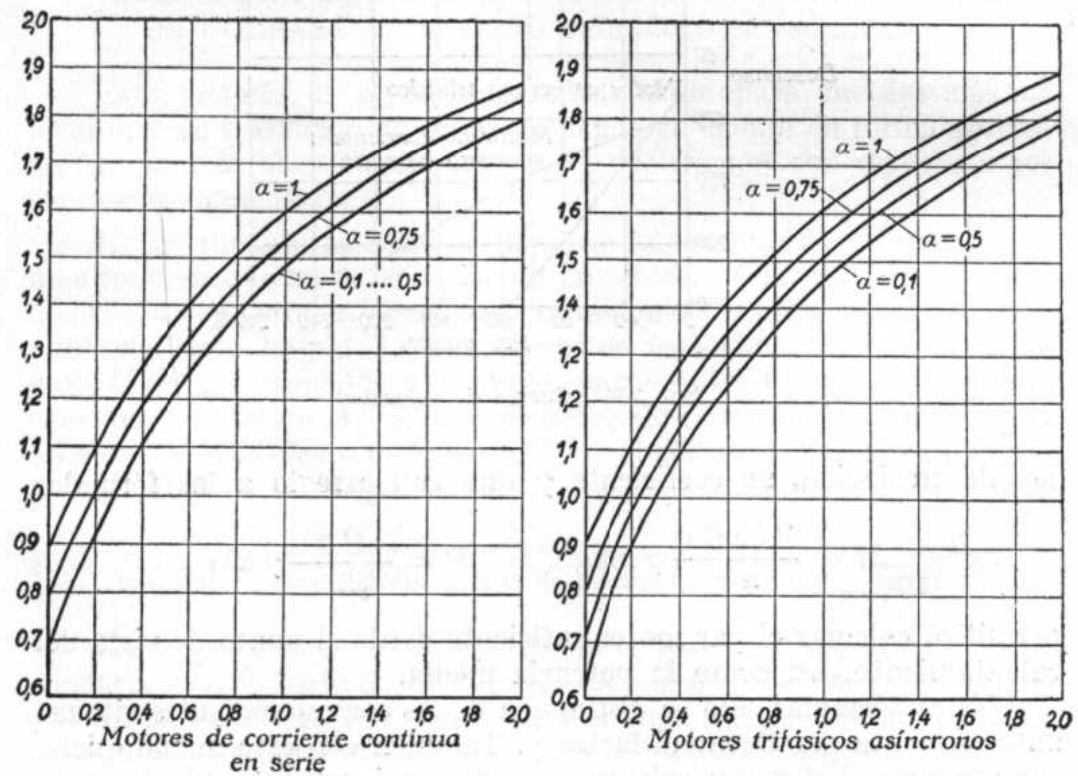


FIG. 247



c. CÁLCULO DE LA POTENCIA DEL MOTOR PARTIENDO DE UN DIAGRAMA DE TRABAJO EXACTO [9]

Es sin duda el procedimiento más exacto, ya que tenemos en cuenta todos los factores que influyen en el diagrama de trabajo.

El diagrama de trabajo se suele dividir en cuatro partes:

- a. Diagrama de velocidad.
- b. Diagrama de recorrido.
- c. Diagrama de par.
- d. Diagrama de potencia.

Vamos a explicar el procedimiento refiriéndonos a un **motor de elevación**. El funcionamiento de un motor de elevación tiene cuatro momentos diferentes:

- Motor de elevación
- | | |
|---|---|
| { | <ol style="list-style-type: none">1. Elevación con carga.2. Bajada con carga.3. Elevación en vacío.4. Bajada en vacío. |
|---|---|

No tienen por que darse siempre los cuatro estados, puede faltar alguno parcialmente o en su totalidad.

Para un **mecanismo de traslación** sólo pueden presentarse dos estados:

- Mecanismo de traslación
- | | |
|---|--|
| { | <ol style="list-style-type: none">1. Desplazamiento con carga2. Desplazamiento en vacío |
|---|--|

En nuestro caso, realizaremos el cálculo de la potencia del motor para plena carga en un mecanismo de translación horizontal en base a los siguientes datos de partida:

Parámetro	Valor
Velocidad del transporte	0,13 m/s
Peso máximo a transportar	200 kg
Coefficiente de rozamiento	0,2
Rendimiento mecánico motor	0,70

$$P = \frac{F_t * v}{102 * \eta} = 0,07 kW$$

Por tanto, según estos cálculos, **nuestro motor debería ser de construcción protegida, con un servicio de funcionamiento permanente o continuo y la siguiente potencia normalizada superior a 0,07 kW.**

En realidad el motor que se ha montado en la instalación de la marca SEW tiene una potencia **0,37 kW** acoplado a una reductora. En la figura siguiente se puede ver la placa de características del motor.



Como ya se había comentado al inicio de este punto, el motor que realmente se ha montado era posible que no coincidiera con los datos teóricos, esto puede deberse a un motivo económico, o a una estandarización en los materiales. También es habitual en muchas ocasiones, que para motores de baja potencia se sobredimensione ligeramente el motor a fin de evitar problemas de calentamiento del motor o sobreesfuerzos en el arranque a plena carga.

7.3.- Cálculo de secciones de los conductores.

En este apartado realizaremos el cálculo de secciones de cable para los elementos de campo más significativos.

Motores eléctricos. 2 Motores eléctricos de 0,37 kW.

Datos: $U_N = 400 \text{ V}$
 $I_N = 1,15 \text{ A}$
 $P_N = 0,37 \text{ kW}$
 $\Delta U = 5\% U_N$
 $L_{AB} = 16 \text{ m}$
 $\gamma_{20^\circ\text{C}} = 56 \text{ S} \cdot \text{m}/\text{mm}^2$
 $\gamma_{70^\circ\text{C}} = 48 \text{ S} \cdot \text{m}/\text{mm}^2$

Siguiendo lo que dice la instrucción BT 47 los cálculos se realizan tomando como valores el cálculo un 125% de los valores nominales.

$$S_{AB} = \frac{\sum P_K \cdot L_K}{\gamma \cdot \Delta U_{AB} \cdot U_N} = \frac{462,5 \cdot 16}{48 \cdot 20 \cdot 400} = 0,019 \text{ mm}^2$$

$$I_{N_{\max}} = \frac{P}{\sqrt{3} \cdot (U_N - \Delta U_{\max}) \cdot \cos \varphi} = \frac{462,5}{\sqrt{3} \cdot (400 - 20) \cdot 0,85} = 0,83 \text{ A}$$

El criterio más restrictivo es de corriente nominal máxima que nos da una sección normalizada de:

$$S_{AB} = 1,5 \text{ mm}^2 \Rightarrow I = 16 \text{ A}$$

$$\Delta U_{AB} = \frac{\sum P_K \cdot L_K}{\gamma \cdot S_{AB} \cdot U_N} = \frac{41 \cdot 10^3 \cdot 17}{56 \cdot 35 \cdot 380} = 3,9 \cdot 10^{-3} \text{ V}$$

Pese a que con una sección de $1,5 \text{ mm}^2$ sería suficiente, en realidad se usará una sección normalizada de $2,5 \text{ mm}^2$ para la alimentación de los motores.

El **conductor de protección**, según la tabla 2 de la instrucción BT 019 tendrá una sección de **$2,5 \text{ mm}^2$** .

7.4. Selección de los dispositivos de protección.

Se entiende por dispositivo de protección al encargado de detectar y/o eliminar las posibles averías o incidentes que se puedan producir en los receptores o instalaciones eléctricas y en sus automatismos eléctricos o circuitos de maniobra.

Los receptores pueden ser el origen de multitud de averías de tipo mecánico o eléctrico. Para evitar que dichas averías deterioren o destruyan los receptores, así como el automatismo eléctrico o circuito de maniobra que los gobiernan, es necesario protegerlos.

Estas averías o incidentes pueden ser las siguientes:

➤ *Sobreintensidad o sobrecarga.*

Es un aumento del consumo de corriente que sobrepasa la corriente de servicio del circuito. Si es permanente se considera como una sobreintensidad no admisible y, por tanto hay que eliminarla. En el caso de que sea una sobreintensidad de corta duración, se considera como una sobreintensidad admisible, por lo que no es necesaria su eliminación.

➤ *Cortocircuitos.*

Es un aumento del consumo de corriente que sobrepasa, considerablemente, a la corriente de servicio, cuando se produce la unión de varios conductores activos. Es obligatoria la eliminación de este tipo de avería en un tiempo inferior a cinco segundos.

➤ *Defecto de aislamiento.*

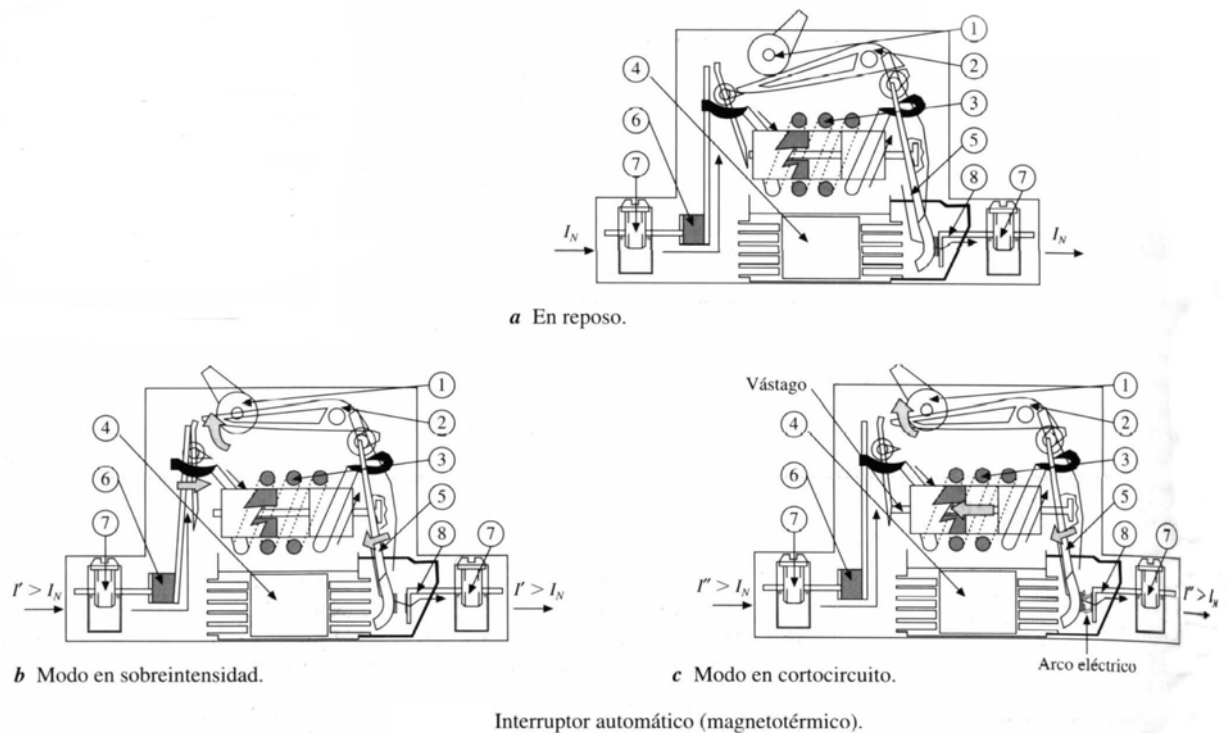
Es la unión entre partes conductoras no activas y partes conductoras activas sometidas a tensión nominal. Es necesario tomar las medidas de protección y de seguridad de los circuitos, para evitar que estos defectos de aislamiento o contactos indirectos produzcan daños importantes.

Los tipos de protecciones utilizadas en el presente proyecto son interruptores magnetotérmicos, guardamotors e interruptor diferencial. A continuación se hará una descripción de su funcionamiento.

Interruptor Automático Magnetotérmico.

Un interruptor magnetotérmico es un dispositivo de protección con capacidad de cortar, por si sólo, las sobreintensidades no admisibles y los cortocircuitos que se producen en los receptores, sin abrir los contactos, cuando la sobreintensidad sea admisible. Este tipo de dispositivo no debe usarse nunca como elemento de maniobra, ya que su estado de reposo es con sus contactos cerrados.

En la siguiente figura se muestra un interruptor magnetotérmico en sus diferentes estados.



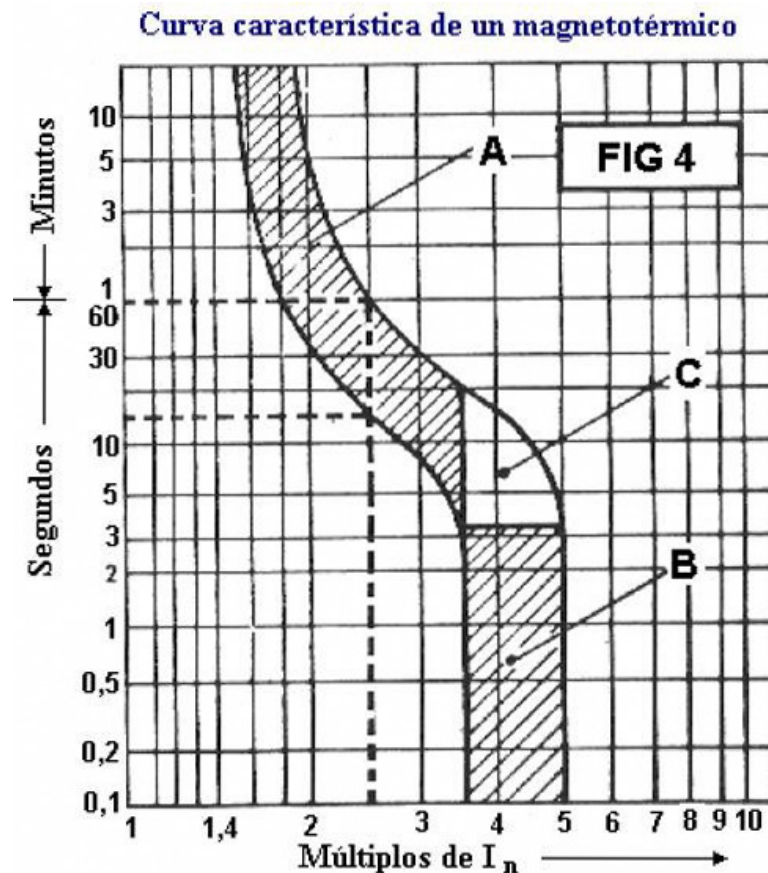
Las partes básicas de un interruptor magnetotérmico son:

- **Palanca.** Es la encargada de accionar manualmente el interruptor magnetotérmico.
- **Trinquete.** Juego de palancas y resortes que mantienen en su posición de reposo cuando la palanca está en la situación ON.
- **Bobina magnética.** Es la encargada de abrir el contacto móvil cuando la corriente que atraviesa el interruptor automático supera la corriente de magnético.
- **Cámara de extinción del arco eléctrico.** Es el elemento encargado de extinguir el arco eléctrico que se produce cuando se abre el contacto móvil.
- **Contacto móvil.** Es el elemento que se abre por la acción de la bobina magnética o por la acción del bimetálico.
- **Bimetálico.** Es la lámina encargada de abrir el contacto móvil cuando la corriente que atraviesa al interruptor automático es superior a su corriente nominal. El tiempo de apertura depende del valor de la corriente que lo atraviesa.

- Bornas de conexión. Son los elementos en donde se conectan los conductores eléctricos, tanto de la alimentación como del receptor.
- Contacto fijo. Este contacto es solidario a la borna donde se conecta el conductor eléctrico del receptor.

Este tipo de interruptor tienen dos modos de funcionamiento, que dependen del valor de la corriente que lo atraviesa, y se pueden representar mediante la curva de disparo, que relaciona su tiempo de intervención en función de la corriente. Si la corriente que atraviesa al magnetotérmico es inferior a la corriente de magnético y superior a la corriente nominal, estamos en el modo de funcionamiento de sobreintensidad. Si esta corriente es superior a la de magnético, estamos en el modo de funcionamiento de cortocircuito.

En la figura siguiente se muestra la curva de disparo característica de un interruptor magnetotérmico.

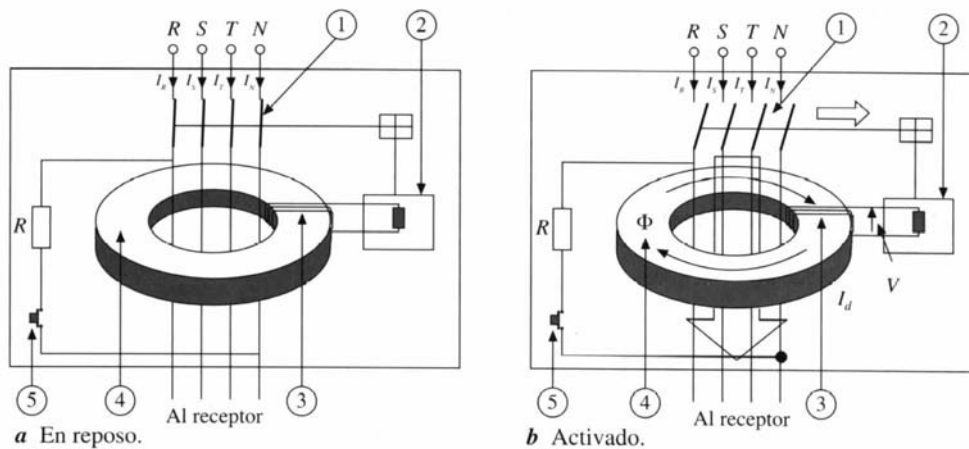


La parte del dispositivo que interviene cuando se produce una sobreintensidad funciona como un relé térmico, liberando un trinquete y abriendo el contacto móvil, mientras que cuando se produce un cortocircuito la parte que interviene es la bobina magnética, que desplaza un vástago abriendo el contacto móvil.

Interruptor diferencial.

Un interruptor diferencial es un dispositivo de protección con capacidad de detectar y eliminar, por sí solo, los defectos de aislamiento que se producen en los receptores. Un defecto de aislamiento producido entre la parte activa y la no activa que está conectada a una puesta a tierra hace circular una corriente de defecto hacia la misma, pero en algunas ocasiones puede cerrarse a través del cuerpo humano. Por este motivo, este dispositivo de protección tiene una gran importancia en las instalaciones eléctricas y es obligado colocar, al menos, uno al principio de la misma. Este tipo de dispositivo necesita estar protegido frente a intensidades y cortocircuitos, colocando un magnetotérmico antes del mismo.

En la siguiente figura se muestra un esquema de interruptor diferencial en sus diferentes estados.



Las partes básicas de un interruptor magnetotérmico son:

- **Contactos principales.** Son los encargados de eliminar el defecto de aislamiento, y están gobernados por el relé diferencial.
- **Relé diferencial.** Es el elemento encargado de accionar la apertura de los contactos principales, mediante un sistema de enclavamiento mecánico.
- **Bobina secundaria.** Es el elemento encargado de detectar la corriente de defecto, cuando se produce un defecto de aislamiento.
- **Toroide.** Es el encargado de canalizar el flujo magnético producido por la corriente de defecto.
- **Pulsador de test.** Mediante este pulsador creamos un defecto de aislamiento entre la fase R y el neutro N, activando el interruptor diferencial.

Cuando no existe ningún defecto de aislamiento en la instalación, la suma vectorial de las corrientes de cada una de las líneas es nula, por lo que no existe corriente de defecto. Pero cuando se produce un defecto de aislamiento,

la suma vectorial de estas corrientes no es nula, apareciendo una corriente de defecto que atraviesa el toroide creando un flujo magnético. Este flujo magnético atraviesa la bobina secundaria, produciéndose una tensión en bornes del relé diferencial, el cual activa la apertura de los contactos principales del interruptor diferencial.

Una vez visto el funcionamiento de las principales protecciones, a continuación se muestra una tabla con todas las protecciones usadas en el proyecto.

Tipo de Protección	Fabricante	Referencia	Nombre	Circuito protección
Automático 4P 16A 3F+N	MERLIN GERIN	21597	F11	Circuito Principal
Diferencial 4P 25A 300mA	MERLIN GERIN	21756	F12	Circuito Principal
Magnetotérmico 3P 1-1,6A	TELEMECANIQUE	GV2-ME06	Q21	Fuente 24Vdc
Automático 2P 10A	MERLIN GERIN	24336	F13	S.A.I. 1A y 1B
Automático 2P 10A	MERLIN GERIN	24336	F14	S.A.I. 2A y 2B
Automático diferencial VIGI 2P 10A 30mA	MERLIN GERIN	21615	F41	Alimentación 230Vac armario eléctrico
Automático 1P 3A	MERLIN GERIN	20533	F51	24Vdc Salidas PLC
Automático 1P 2A	MERLIN GERIN	20532	F61	24Vdc Armario
Automático 1P 3A	MERLIN GERIN	20533	F71	24Vdc Campo
Automático 1P 2A	MERLIN GERIN	20532	F81	24Vdc OP7
Automático 1P 2A	MERLIN GERIN	20532	F91	24Vdc PLC
Magnetotérmico 3P 1-1,6A	TELEMECANIQUE	GV2-ME06	Q301	Motor M11
Magnetotérmico 3P 1-1,6A	TELEMECANIQUE	GV2-ME06	Q302	Motor M12

7.5. Estructura del proyecto eléctrico.

El proyecto eléctrico ha sido realizado con un software de diseño específico. El software en cuestión es **EPLAN**, que es un **CAD Eléctrico**, de forma que se facilita la tarea al desarrollar un proyecto eléctrico.

Este software, entre otras cosas nos permite generar de forma automática los siguientes elementos:

- Índice de Páginas.
- Referencias Cruzadas.
- Resumen de datos de PLC (Listado entradas y salidas).
- Lista de materiales
- Lista de pedidos de materiales.

Sólo con esto, a la hora de realizar un proyecto el tiempo que se ahorra es considerable, por lo que esta herramienta nos permite realizar proyectos eléctricos a un menor coste.

Pero, no es ésta su única ventaja, además ofrece otras características muy importantes:

- Símbolos. El software trae una librería de símbolos estándar, con los elementos más comunes, de forma, que resulta muy fácil insertar cualquier elemento.
- Macros de Símbolos. Con esta herramienta podemos crear macros con elementos que no estén en la librería de símbolos para utilizarlos posteriormente en otros proyectos, de forma, que sólo debemos dibujar un elemento que no esté en el proyecto una sola vez. Además, la mayoría de los fabricantes (por ejemplo, SIEMENS) ponen a disposición de sus clientes una librería de macros con sus componentes para EPLAN.
- Lugares. Esta herramienta nos permite desglosar el proyecto en diferentes partes, de forma que a la hora de buscar cualquier cosa se facilita la tarea. Por ejemplo, en este proyecto se han definido 14 lugares (armario eléctrico, botonera emergencia, etc.).
- Verificación de errores. Es muy importante, no sólo ahorrar tiempo a la hora de realizar un proyecto, si no también que el proyecto esté perfectamente realizado, de forma que si el proyecto no tiene errores, la instalación se realizará sin errores, y evitaremos modificar el proyecto después de la instalación. Algunos errores detectados por el software son: elementos duplicados, errores de conexión, etc.

Pasaremos ahora a ver la estructura del proyecto eléctrico.

Como ya habíamos comentado se han definido 14 lugares a la hora de realizar este proyecto. En la tabla siguiente se pueden ver cada uno de ellos.

Identificador de Lugar	Descripción
X0	Armario eléctrico principal
X1M1	Caja de mando Línea A
X1M2	Caja de mando Línea B
X1B1	Botonera X1B1
X1B2	Botonera X1B2
X1B3	Botonera X1B3
X1B4	Botonera X1B4
X1B5	Botonera X1B5
X1B6	Botonera X1B6
X1B7	Botonera X1B7
X1B8	Botonera X1B8
XUM1	Unidad de mantenimiento (Neumática)
X100	Lista de materiales
X101	Lista de pedidos

De esta forma, el proyecto queda bastante organizado, de forma que de cara al instalador, incluso se pueden repartir los planos si trabajan varias personas, y realizar la instalación de una forma más rápida y eficaz, ya que si un eléctrico va a realizar la conexión de una botonera, solo tendrá que buscar los esquemas eléctricos relacionados con la misma.

Además de utilizar los identificadores de lugar para organizar el proyecto, este software permite crear hojas con la numeración que se quiera, y automáticamente las relaciona, de forma que en cada hoja muestra cuál sería la siguiente. De esta forma hay proyectistas, que de cara a estandarizar los proyectos se reservan un número de hojas para cada parte de proyecto, pudiendo así facilitar nuevamente la tarea a los instaladores, e incluso a los programadores de PLC, ya que se buscarán las cosas más rápidas.

En nuestro caso, el departamento eléctrico de la empresa contratista tiene una estructura claramente definida para la realización del lugar X0 (armario eléctrico principal). En la tabla siguiente se muestra la estructura para este proyecto:

Lugar	Página Inicial	Descripción
X0	1	Protección y distribución de potencia
X0	20	Relé de seguridad y emergencias
X0	30	Distribución potencia motores
X0	60	PLC
X0	90	Puerta de armario eléctrico

Una vez visto esto, en el anexo 2 podemos ver el proyecto eléctrico completo.

7.6. Layout eléctrico.

Un elemento muy importante es el layout eléctrico. Con un layout eléctrico bien diseñado el instalador sabrá en todo momento dónde están ubicados los diferentes elementos que debe cablear.

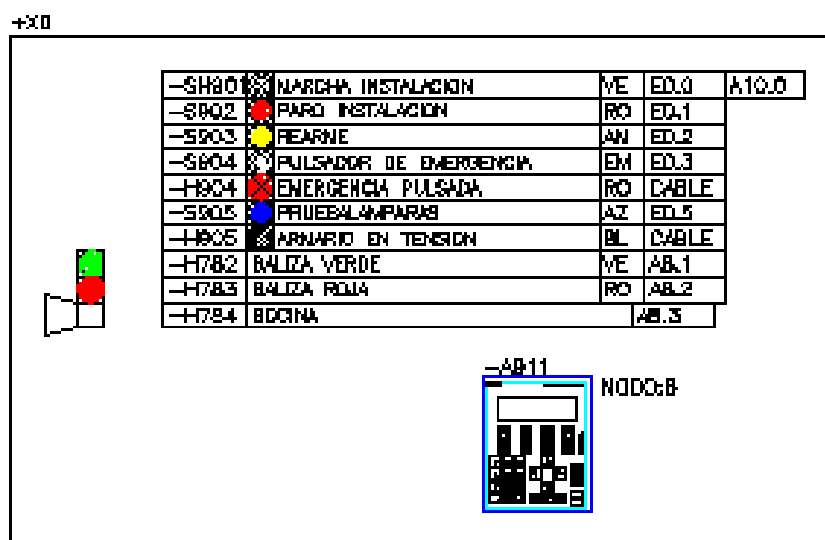
El **layout eléctrico** ha sido realizado con **Autocad**. Si el layout ha sido realizado a escala y el instalador dispone de la herramienta de diseño podrá incluso calcular los metros necesarios de cable para realizar la instalación.

Además de ayuda para el instalador también será de gran utilidad para el programador de la instalación, ya que de un vistazo se podrá hacer una idea de donde están los elementos que debe controlar, sus entradas del autómatas, sus salidas, etc. Y posteriormente, también servirá de ayuda al personal de mantenimiento, puesto que si por ejemplo, las alarmas de la instalación hacen referencia a los nombres de los componentes dibujados en el layout podrá localizar la avería en menor tiempo.

Vamos a ver algún ejemplo de los diferentes elementos que aparecen en el layout:

- **Armario eléctrico.** En primer lugar, veremos que el detalle del armario eléctrico está dibujado dentro de un caja y que tiene el nombre +X0. Este nombre como habíamos visto en el apartado anterior corresponde a un lugar definido en Eplan, por lo que si necesitamos más detalle nos está indicando donde buscar en los planos.

Dentro de esta caja aparecen los elementos más significativos instalados en el exterior del armario (por ejemplo en la puerta) con su nombre eléctrico, el color de pulsadores, entradas del PLC, salidas del PLC, etc.



- Detector inductivo. El detector inductivo se representa con un cuadrado, y en nuestro ejemplo es de color marrón. El color indica donde va conectado este elemento, de forma que si vemos el layout completo veremos que está conectado en la caja de mando +X1M1 que tiene el mismo color.

Además también podemos saber a que entrada está conectado, ya que nombre del inductivo está formado por B+nº sinóptico+entrada PLC. Por tanto de un vistazo, vemos que el elemento pertenece al sinóptico 1 y está conectado a la entrada E3.3.

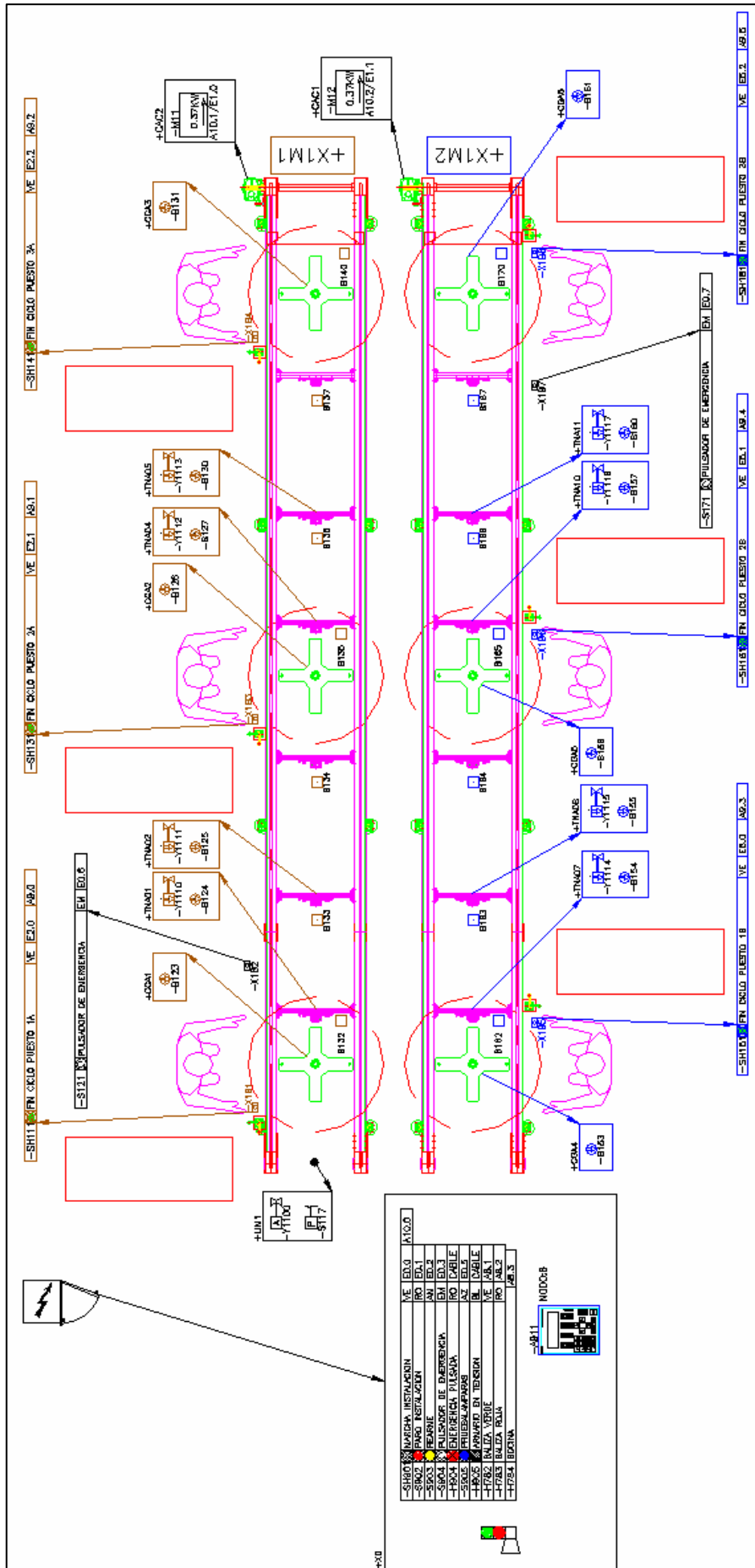


- Tope. En primer lugar vemos que la información aparece dentro de una caja con el nombre del tope. Y dentro de esta caja aparecen varios elementos.

El primer elemento que aparece es la electroválvula que hace moverse al tope y el segundo es un detector magnético, que nos indica la posición. De un vistazo podremos que electroválvula es de simple efecto por el símbolo y que la operación que realiza es bajar el tope por la flecha que lleva en su interior, además sabremos que la salida que la hace actuar es la A11.3, ya que la filosofía para nombrar el elemento es la misma que vimos para el inductivo, pero cambiando la B por Y y la entrada por la salida. Con el magnético de posición del tope ocurre lo mismo, podemos ver que nos indicará que el tope está bajado (por la flecha en su interior) y que corresponde a la entrada E3.0.



Una vez vistos estos ejemplos podemos entender mejor el layout que se muestra en la siguiente página.



7.7. Fotos de la instalación.

A continuación, se muestran una serie de fotografías en las que podremos ver los diferentes elementos de la instalación.



Figura 1. Puerta Armario Eléctrico.



Figura 2. Interior Puerta Armario Eléctrico.



Figura 3. Interior Armario Eléctrico.

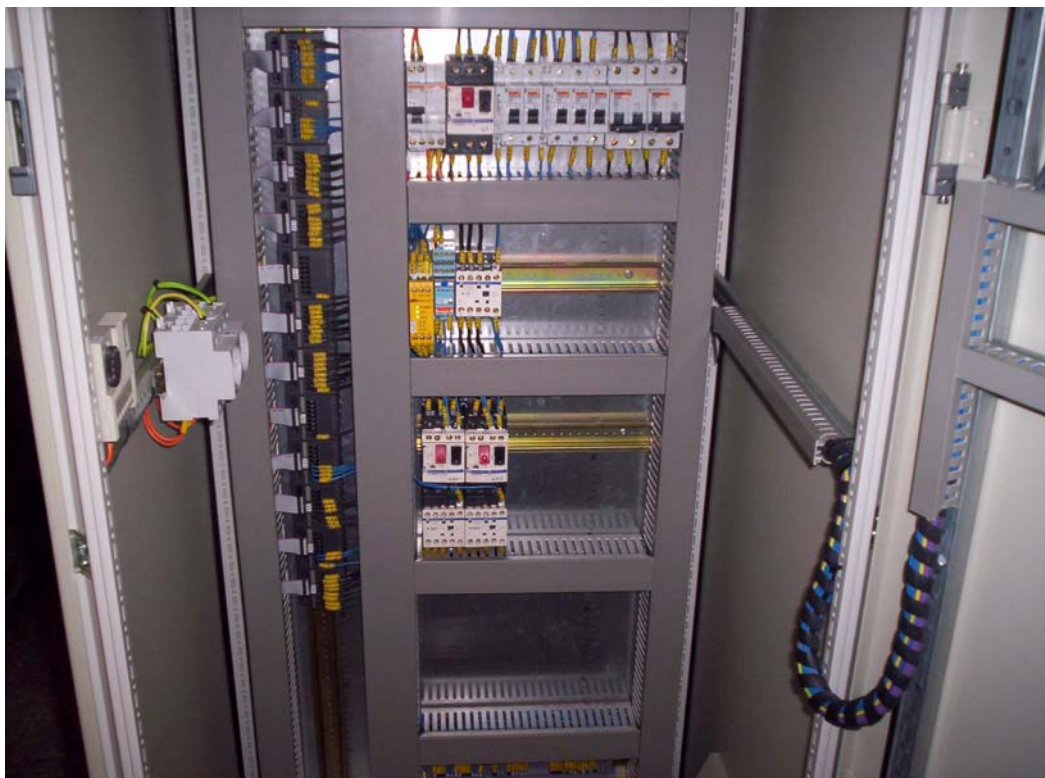


Figura 4. Interior Armario Eléctrico.

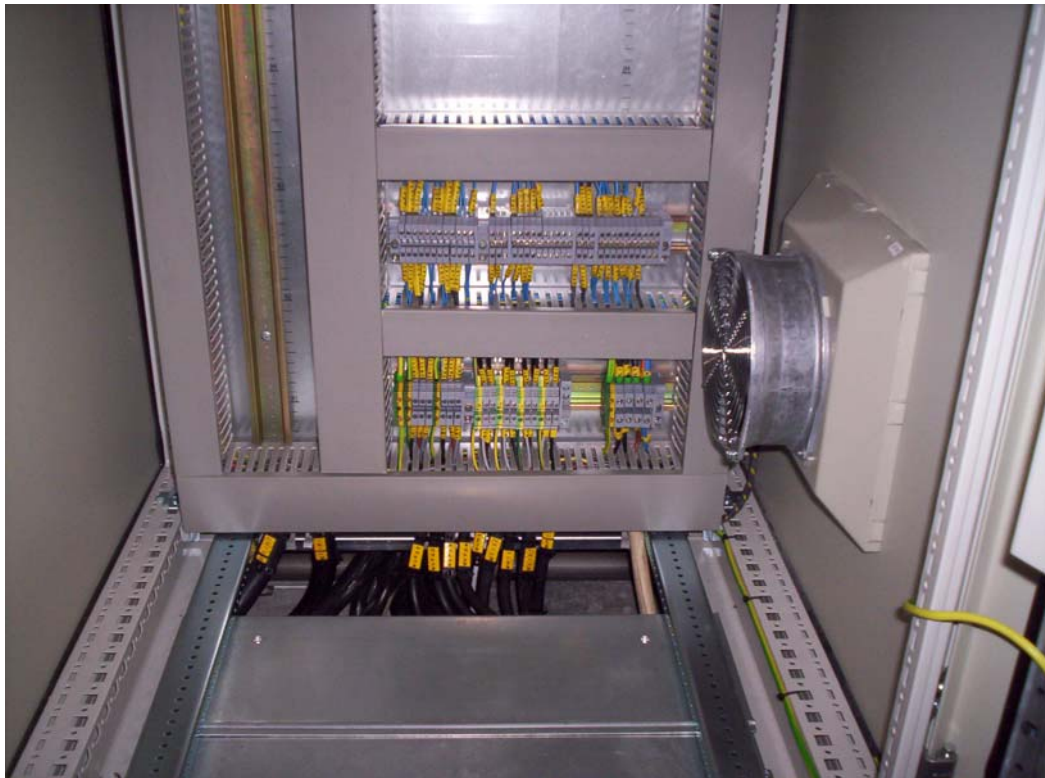


Figura 5. Interior Armario Eléctrico.

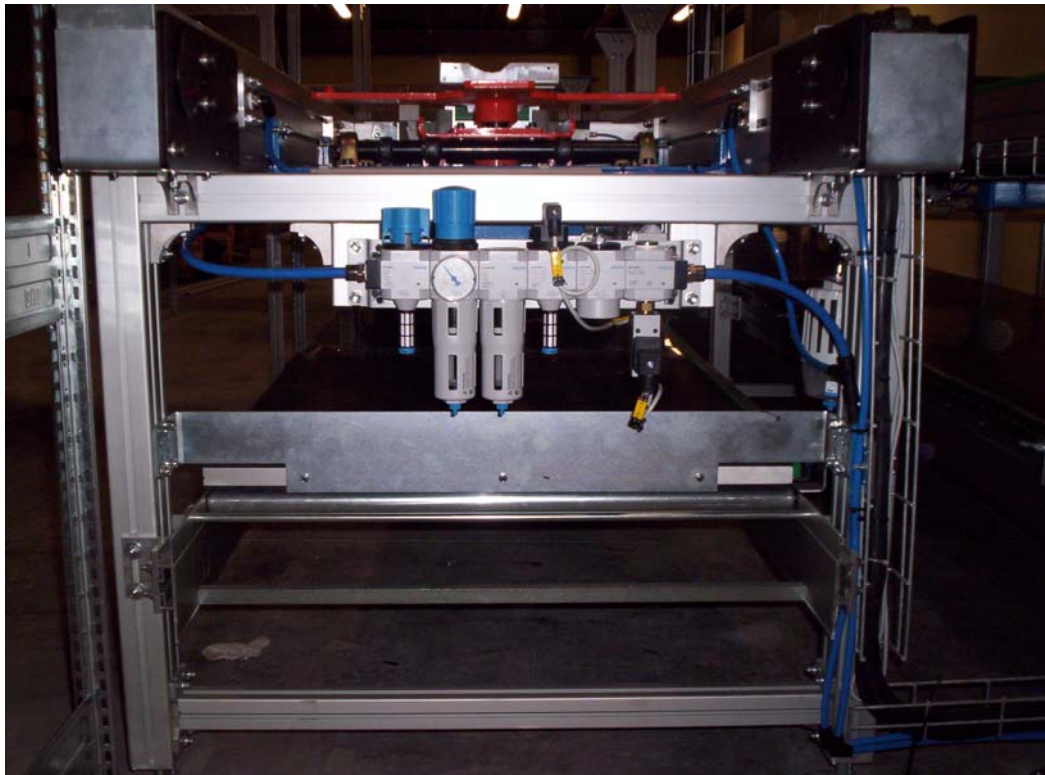


Figura 6. Unidad mantenimiento Neumática.



Figura 7. Cajas de Mando.

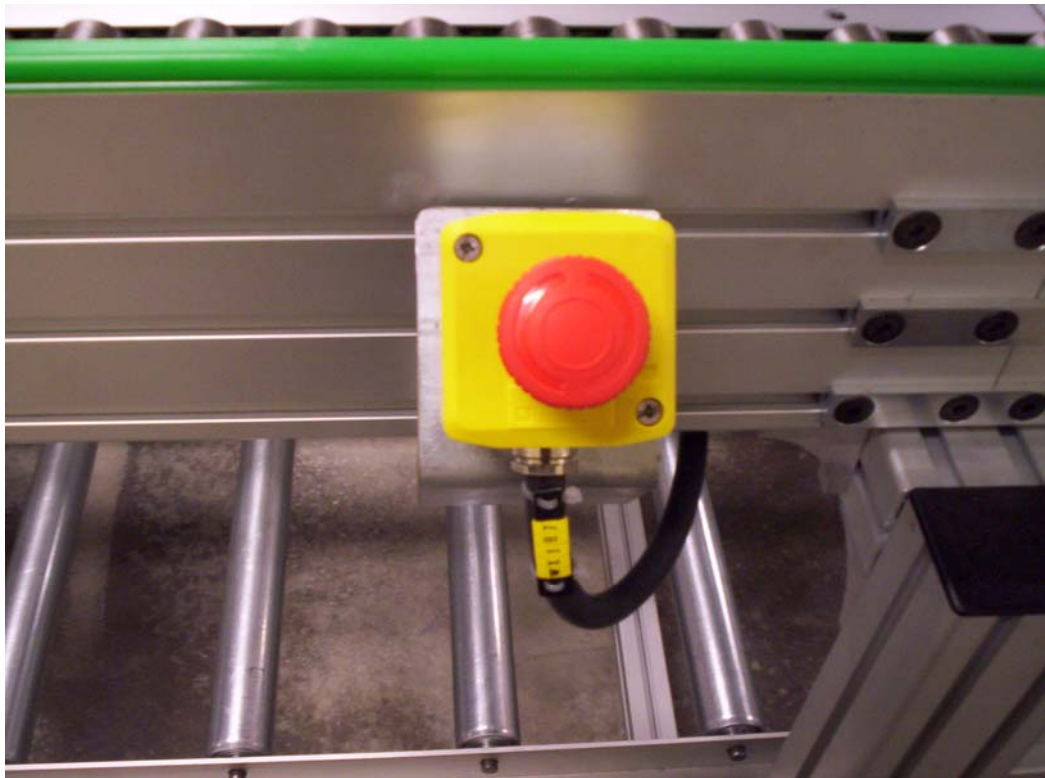


Figura 8. Parada de Emergencia.

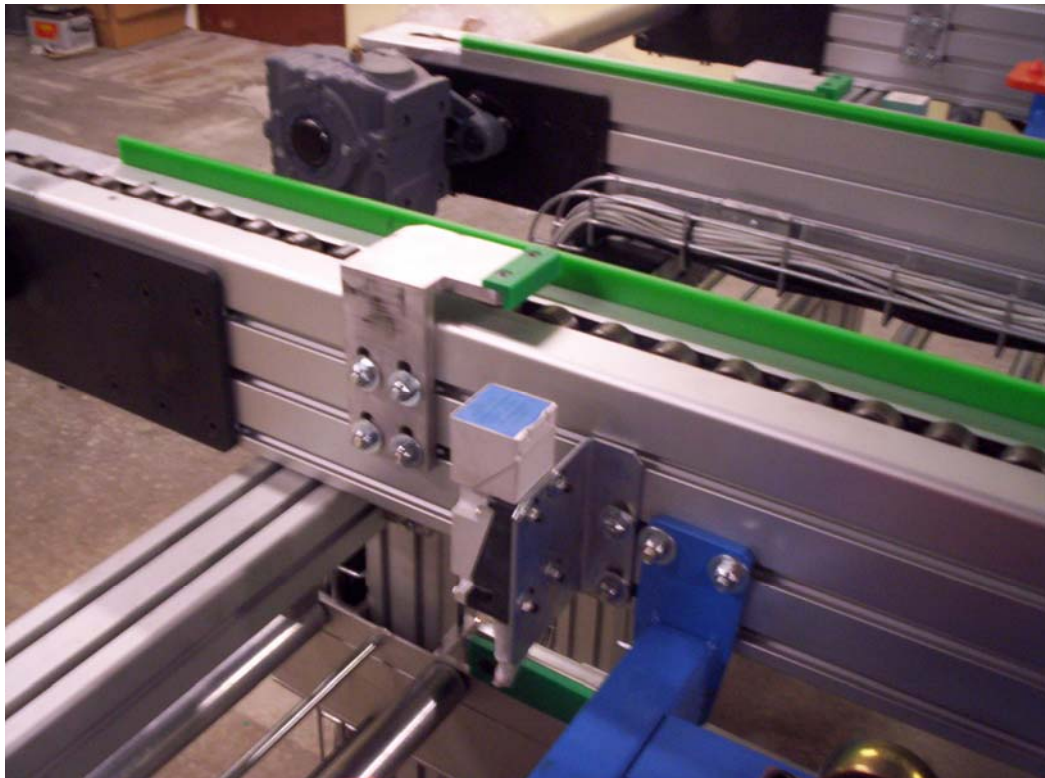


Figura 9. Inductivo presencia en puesto trabajo.

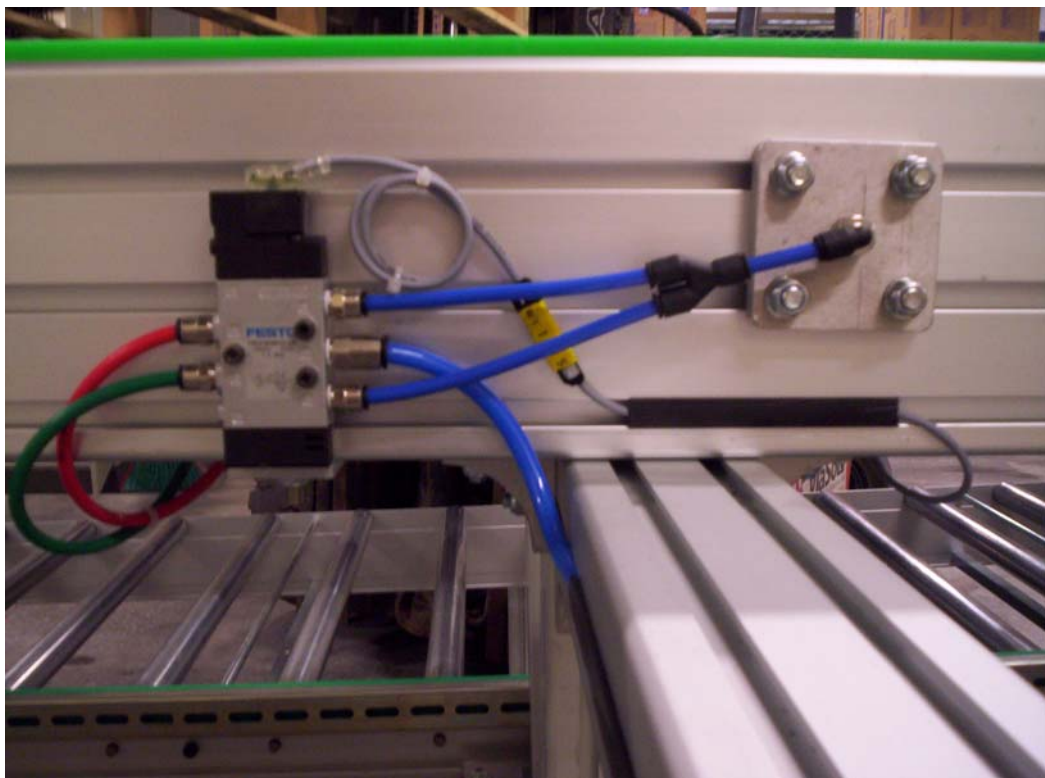


Figura 10. Electro válvula control tope neumático.

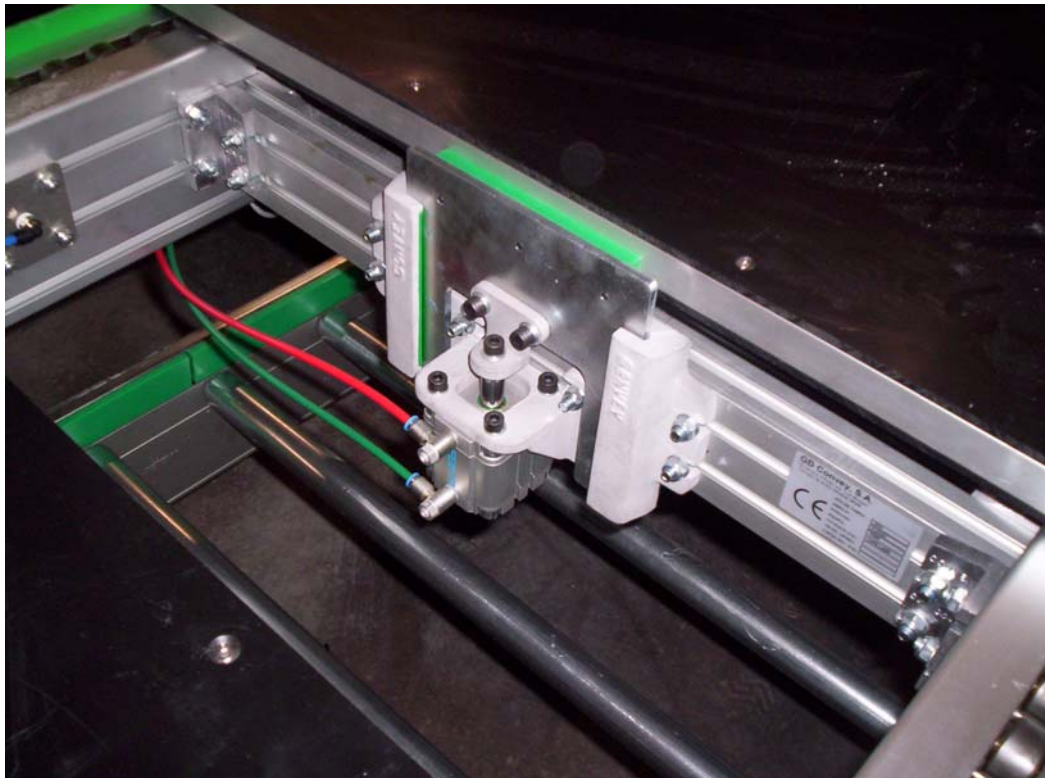


Figura 11. Tope neumático.

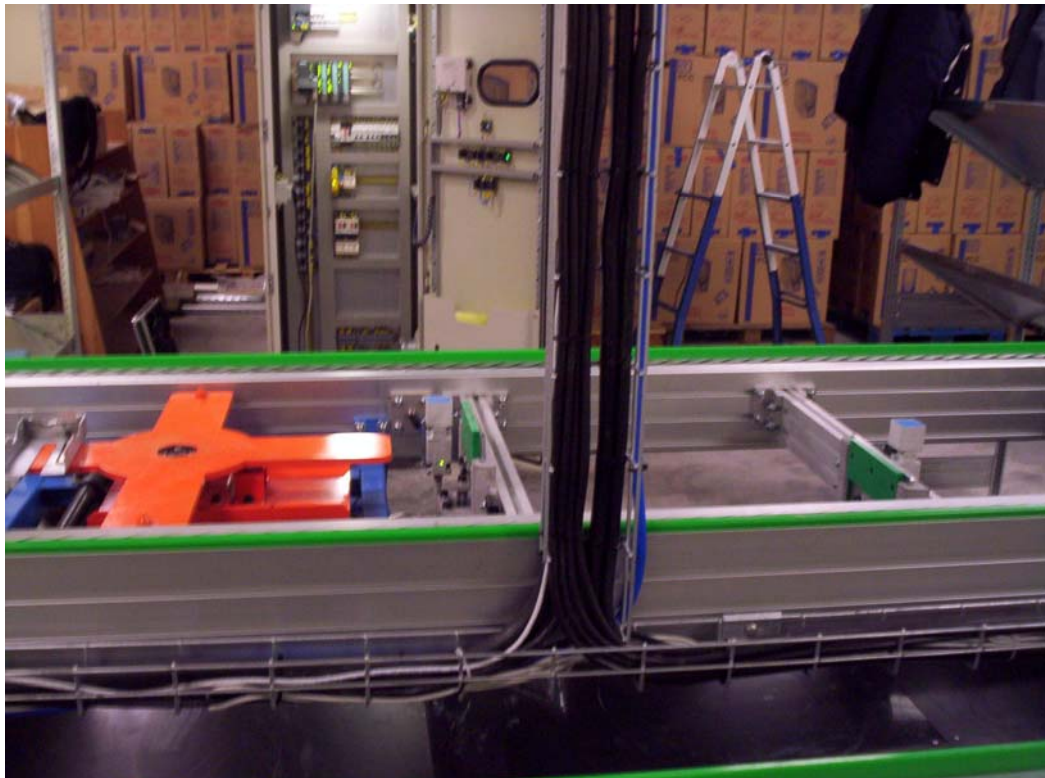


Figura 12. Bandeja Regiband.



Figura 13. Motor tracción cadenas arrastre bandejas.



Figura 14. Parte inferior bandejas de transporte.

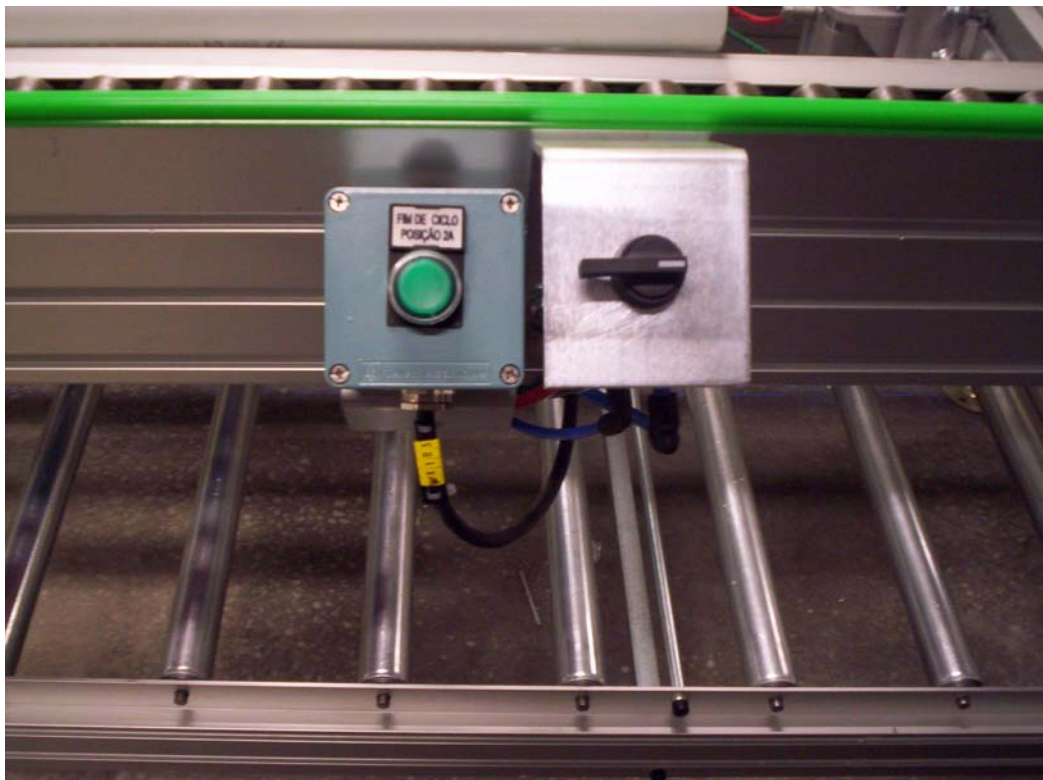


Figura 15. Botonera control puesto trabajo.

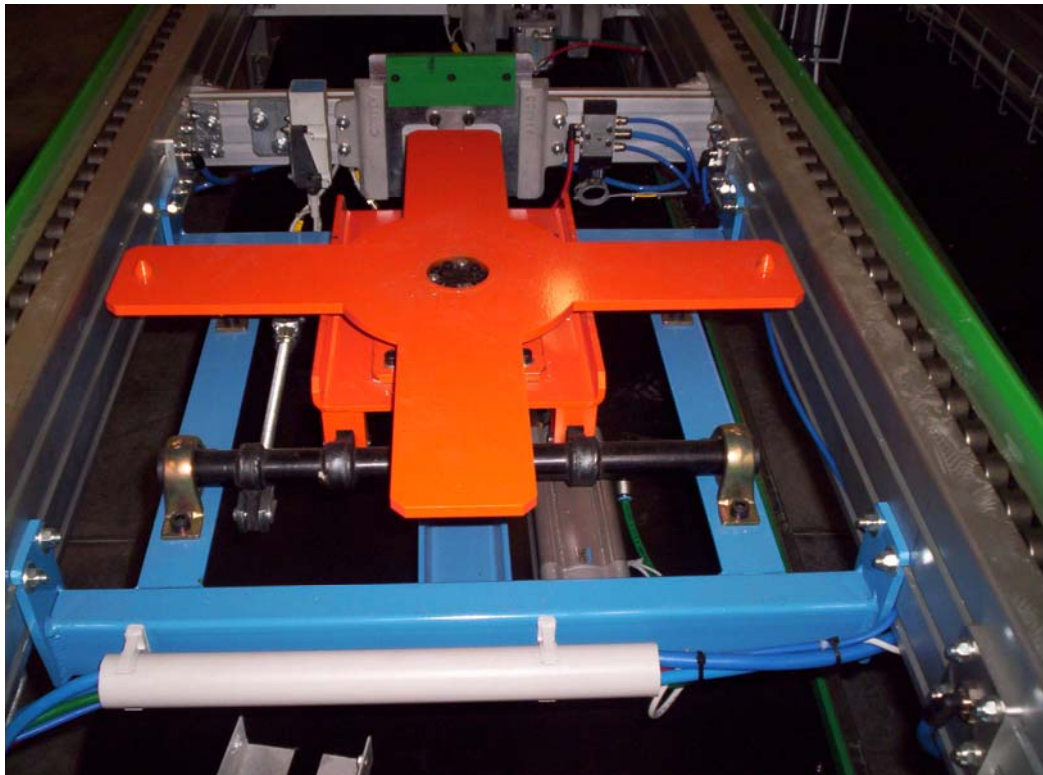


Figura 16. Cruz de giro puesto de trabajo.

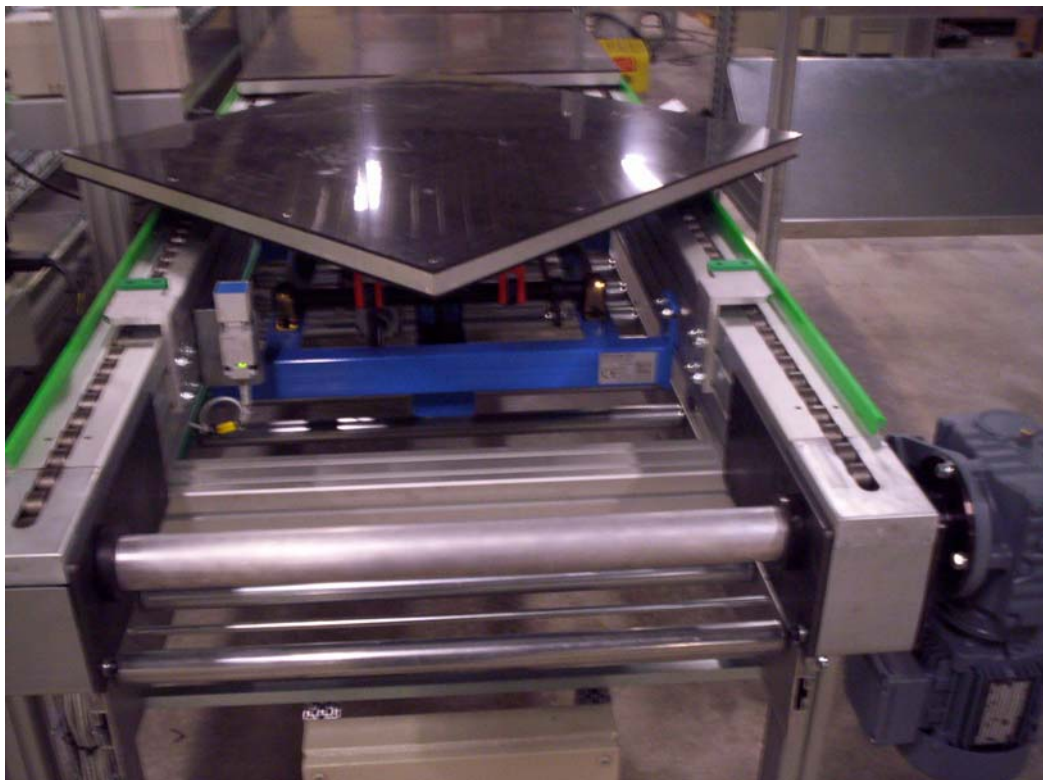


Figura 17. Manipulación bandeja en puesto de trabajo.

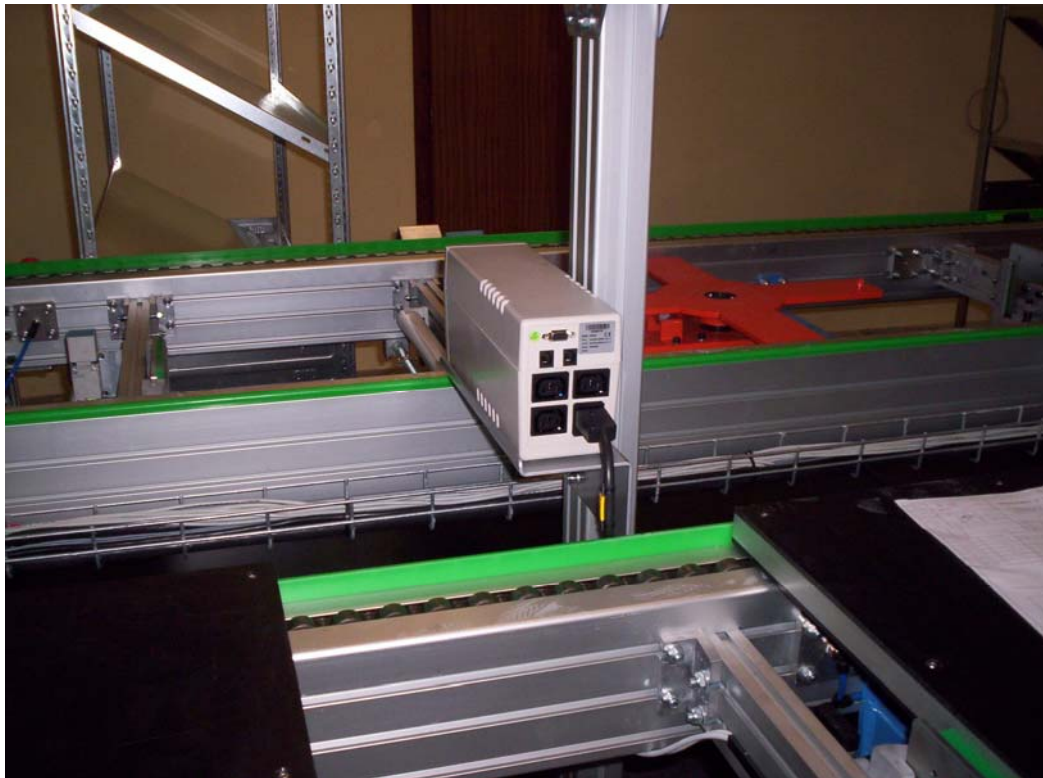


Figura 18. UPS para alimentación PC en puesto de trabajo.

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CAPÍTULO 8

PRESUPUESTO

8. PRESUPUESTO.

En este capítulo hablaremos de los puntos que se deben tener en cuenta a la hora de realizar una oferta de un pedido. Concretamente hablaremos de los puntos que se han tenido en cuenta para la realización de la oferta que se puede ver en el siguiente apartado.

En primer lugar debemos recopilar toda la información que sea posible sobre la instalación. En nuestro caso el cliente nos proporcione un layout eléctrico, el proyecto eléctrico, el listado de materiales y un pequeño descriptivo del funcionamiento de la instalación.

Debido a que el suministro del proyecto atañe únicamente al desarrollo de la programación del autómatas y del panel de control, así como la puesta en marcha de los mismos, los puntos que se han tenido en cuenta para realización de la oferta son los siguientes:

1. Número de horas de programación de PLC.
2. Número de horas de programación del Panel de Operador.
3. Número de horas de Documentación.
4. Número de horas de Puesta en Marcha.
5. Dietas por desplazamiento y alojamiento.

El cálculo del número de horas que puede llevar la realización del proyecto no depende sólo del tamaño de la instalación, sino también de la complejidad de la instalación.

En el apartado siguiente se puede ver la oferta realizada para este proyecto.



8.1. Oferta.

OFERTA Nº : 00001/08

PROYECTO

AUTOMATIZACIÓN DE UNA LÍNEA DE
MONTAJE DE ORDENADORES

1.- OBJETO

El objeto de esta oferta es el de valorar los trabajos de desarrollo de la programación de autómatas, programación de panel operador y puesta en marcha para el proyecto arriba indicado.

2.- ALCANCE DEL SUMINISTRO

El alcance de suministro de la siguiente oferta contempla los siguientes puntos.

2.1.- INGENIERIA.

2.1.1.- PROGRAMACION AUTOMATA Y OP.

- Programación automática.
- Programación panel operador OP7.
- Documentación actualizada.

2.1.2.- PUESTA EN MARCHA EN MADRID.

- Puesta en marcha de la instalación durante el montaje en las instalaciones del cliente en Madrid.

2.1.3.- PUESTA EN MARCHA EN PORTUGAL.

- Puesta en marcha de la instalación durante el montaje en PORTUGAL.
- Cursos de formación.
- Asistencia a la producción durante dos días.

3.- OFERTA ECONOMICA

La oferta económica incluidos los gastos de viaje y dietas asciende al siguiente importe:

3.1.- INGENIERIA.

3.1.1.- PROGRAMA AUTOMATA Y OP.	2.000.- €
3.1.2.- P. EN MARCHA EN GD CONVEY.	600.- €
3.1.3.- P. EN MARCHA EN PORTUGAL.	2.400.- €

TOTAL : 5.000.- €

4.- CONDICIONES DE PAGO

30% Al contado con el pedido.
70% Restante mediante pagaré a 120 días f.f.



5.- PLAZO DE ENTREGA

A confirmar por el cliente.

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

BIBLIOGRAFÍA

BIBLIOGRAFÍA

BIBLIOGRAFÍA

- [1]. Joseph Balcells y José Luis Romeral. *Autómatas programables*. Editorial Marcombo, Barcelona. Edición 1997.
- [2]. Albert Mayol i Badía. *Autómatas programables*. Editorial Marcombo. Edición 1987.
- [3]. Alejandro Porras Criado y A. P. Montanero. *Autómatas programables*. Editorial McGraw-Hill, Madrid. Edición 1997.
- [4]. www.automatas.org.
- [5]. Revista de Electricidad, Electrónica y Automática. (www.reea.6x.to).
- [6]. Siemens. *Sistema de automatización S7-300. Datos de las CPU 312 IFM a 318-2 DP*. Referencia: 6ES7398-8FA10-8DA0. Edición 10/2001.
- [7]. Siemens. *Programar con STEP 7 V5.1*. Referencia: 6ES7 810-4CA05-8DA0. Edición 08/2000.
- [8]. Siemens. *Panel de operador OP7, OP17*. Referencia: 6AV3991-1AE05-1AE0. Edición 04/1999.
- [9]. www.diee.unican.es. Cálculo de potencia de motores (Tracción Eléctrica).
- [10]. Reglamento Electrotécnico de Baja Tensión. Real Decreto 842/2002
- [11]. www.siemens.es.

UNIVERSIDAD CARLOS III DE MADRID
ESCUELA POLITÉCNICA SUPERIOR



PROYECTO FIN DE CARRERA

**AUTOMATIZACIÓN DE UNA LÍNEA
DE MONTAJE DE ORDENADORES**

CONTENIDO DEL CD

CONTENIDO DEL CD

CONTENIDO DEL CD.

En el proyecto se ha incluido un CD con toda la información impresa en el proyecto, y además algunos documentos que no han sido impresos, ya que pueden ser mejor visualizados en formato pdf, puesto que permite realizar búsquedas.

Presentación.

La presentación del proyecto está incluido en formato PowerPoint y en pdf en el siguiente directorio dentro del CD: [\\000. Presentación.](#)

Esta presentación sólo está incluida en el CD y no de forma impresa.

Proyecto.

El proyecto está incluido en formato pdf en el siguiente directorio dentro del CD: [\\001. Proyecto.](#)

Anexos.

Esta parte, al igual que la presentación no está incluida de forma impresa, sólo se encuentra en CD. Los anexos están incluidos en el siguiente directorio dentro del CD: [\\002. Anexos.](#)

Dentro de la carpeta anexos nos encontraremos los siguientes:

➤ *Anexo 1. Características técnicas.*

En este anexo se incluyen las hojas de características, catálogos o manuales de los principales elementos electrónicos de la instalación.

➤ *Anexo 2. Esquemas eléctricos.*

Este anexo contiene los esquemas eléctricos completos impresos en pdf desde Eplan.

➤ *Anexo 3. Programa del Autómata.*

Aquí podemos ver el programa del PLC, tanto configuración de Hardware, como listado entradas y salidas, programa, referencias cruzadas y simbólico.

➤ *Anexo 4. Programa del Panel de Operación.*

En este anexo encontraremos el programa del panel de operador, las pantallas, listado de alarmas, etc.



➤ *Anexo 5. Manual de Usuario.*

Este anexo contiene el manual de usuario que fue entregado al cliente final de la instalación, con toda la información necesaria para manejar la instalación.

Software.

Se ha incluido como software el Adobe Acrobat Reader en el siguiente directorio del CD: [\\Software\AcroReader](#), con el fin de poder visualizar los documentos en cualquier PC en caso de no tener el software.