

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA INDUSTRIAL



PROYECTO FIN DE CARRERA

“Configuración y puesta en marcha de una red de autómatas programables basada en PROFIBUS, MPI y GSM para el control y monitorización de módulos de fabricación flexible”

Ingeniero Técnico Industrial, especialidad Electrónica Industrial

Departamento de Ingeniería de Sistemas y Automática

Director: Miguel Almonacid Kroeger

Autor: Francisco José Menchón Ruiz

Diciembre, 2006

Agradecimientos

Quisiera agradecer el apoyo y comprensión a mi familia, que siempre ha estado ahí cuando la he necesitado.

Asimismo, expresar mi especial agradecimiento al director de este Proyecto Fin de Carrera, Miguel Almonacid Kroeger, por su incondicional ayuda, dedicación y disponibilidad, sin escatimar en nada.

Por último, he de mencionar al Departamento de Ingeniería de Sistemas y Automática de la Universidad Politécnica de Cartagena así como al técnico de laboratorio por su gran ayuda prestada.

Índice

1	MOTIVACIÓN Y OBJETIVOS	1
1.1	MOTIVACIÓN Y OBJETIVOS	1
1.2	RESUMEN Y FASES DEL PROYECTO.....	1
2	REDES DE COMUNICACIÓN ENTRE AUTÓMATAS. SOLUCIONES ACTUALES	5
2.1	INTRODUCCIÓN.....	5
2.2	SISTEMAS INDUSTRIALES DE CONTROL	6
2.2.1	<i>Control centralizado.....</i>	<i>7</i>
2.2.2	<i>Control distribuido</i>	<i>7</i>
2.2.3	<i>Control híbrido.....</i>	<i>8</i>
2.3	LA PIRÁMIDE CIM	8
2.3.1	<i>Nivel de E/S (Nivel actuador/sensor).....</i>	<i>9</i>
2.3.2	<i>Nivel de Campo y Proceso.....</i>	<i>9</i>
2.3.3	<i>Nivel de Control (Nivel de célula).....</i>	<i>10</i>
2.3.4	<i>Nivel de gestión (Nivel de fábrica).....</i>	<i>11</i>
2.4	REDES DE COMUNICACIÓN INDUSTRIALES	11
2.4.1	<i>Red de Factoría.....</i>	<i>12</i>
2.4.2	<i>Red de Planta</i>	<i>12</i>
2.4.3	<i>Red de Célula</i>	<i>12</i>
2.4.4	<i>Bus de Campo.....</i>	<i>12</i>
2.4.4.1	<i>Panorámica actual de los buses de campo.....</i>	<i>13</i>
2.4.4.2	<i>Buses de Campo y Niveles OSI.....</i>	<i>14</i>
2.4.4.3	<i>Buses Propietarios y Buses Abiertos.....</i>	<i>14</i>
2.4.4.4	<i>Ventajas e Inconvenientes de los Buses de Campo</i>	<i>15</i>
2.4.4.5	<i>Normalización.....</i>	<i>17</i>
2.4.4.6	<i>Buses existentes en el mercado</i>	<i>17</i>
2.4.4.6.1	<i>P-Net.....</i>	<i>19</i>
2.4.4.6.2	<i>PROFIBUS (PROcess FIeld BUS).....</i>	<i>19</i>
2.4.4.6.3	<i>WorldFIP (World Factory Instrumentation Protocol).....</i>	<i>19</i>
2.4.4.6.4	<i>HART (Highway Addressable Remote Transducer).....</i>	<i>20</i>
2.4.4.6.5	<i>Modbus.....</i>	<i>21</i>
2.4.4.6.6	<i>Interbus</i>	<i>22</i>

2.4.4.6.7	<i>Measurement Bus</i>	23
2.4.4.6.8	<i>CAN (Controller Area Network)</i>	23
2.4.4.6.9	<i>SDS (Smart Distributed System)</i>	23
2.4.4.6.10	<i>DeviceNet</i>	24
2.4.4.6.11	<i>Sensoplex</i>	24
2.4.4.6.12	<i>J1939</i>	25
2.4.4.6.13	<i>AS-I (Actuator/Sensor Interface)</i>	25
2.4.4.6.14	<i>NBIP (Nine-Bit Interprocessor Protocol)</i>	25
2.4.4.6.15	<i>M3S (Múltiple Master Múltiple Slave)</i>	26
2.4.4.6.16	<i>Lon Works (Local Operating Network)</i>	26
2.4.4.6.17	<i>ARCNET</i>	27
2.4.4.6.18	<i>M-Bus (Meter-Bus)</i>	27
2.5	EL BUS PROFIBUS	28
2.5.1	<i>Introducción</i>	28
2.5.2	<i>El modelo de referencia OSI y PROFIBUS</i>	29
2.5.3	<i>Tecnología de transmisión</i>	30
2.5.4	<i>Método de acceso</i>	31
2.5.5	<i>El protocolo PROFIBUS-DP</i>	35
2.5.5.1	<i>Tipos de dispositivos</i>	35
2.5.5.2	<i>Configuración del sistema</i>	36
2.5.5.3	<i>Capa física. Transmisión RS 485</i>	36
2.5.5.4	<i>Capa de enlace. FDL</i>	37
2.5.5.5	<i>Nivel de usuario. Funciones DP</i>	38
2.6	LA INTERFACE MPI	39
2.6.1	<i>Datos técnicos</i>	39
2.6.2	<i>Tecnología de transmisión</i>	40
2.7	LA RED DE COMUNICACIÓN GSM	40
2.7.1	<i>Historia</i>	40
2.7.2	<i>Arquitectura y Funcionamiento</i>	41
2.7.3	<i>Envío/Recepción de SMS</i>	42
3	DESCRIPCIÓN DE LOS EQUIPOS UTILIZADOS	45
3.1	INTRODUCCIÓN	45
3.2	AUTÓMATAS PROGRAMABLES SERIE S7	45
3.2.1	<i>Serie S7-200. CPU 224 AC/DC/Relé</i>	46
3.2.2	<i>Serie S7-300. CPU 314C-2DP</i>	47
3.3	FUENTE DE ALIMENTACIÓN PS307-2A	49
3.4	MÓDULO DE COMUNICACIONES EM277	50
3.5	MÓDULO DE COMUNICACIONES TC35	51
3.6	CABLE PC/PPI	53
3.7	CABLE PC ADAPTER	53
3.8	CABLE PROFIBUS	54
3.9	CONECTOR PROFIBUS	55
3.10	MÓDULOS DE FABRICACIÓN FLEXIBLE	55
3.10.1	<i>Módulo doble alimentador por gravedad</i>	56

3.10.2	Módulo cinta transportadora.....	56
4	MONTAJE DE LA RED DE COMUNICACIÓN.....	59
4.1	INTRODUCCIÓN.....	59
4.2	MONTAJE EQUIPO MAESTRO.....	61
4.3	MONTAJE EQUIPOS ESCLAVO DP Y ESCLAVO MPI.....	61
4.4	MONTAJE CABLE PROFIBUS.....	62
4.5	MONTAJE ADAPTADOR MACHO-MACHO.....	65
4.6	MONTAJE EQUIPO MODEM TC35	66
4.7	MONTAJE MÓDULOS DE FABRICACIÓN FLEXIBLE	68
5	CONFIGURACIÓN DE EQUIPOS Y RED DE COMUNICACIÓN	73
5.1	INTRODUCCIÓN.....	73
5.2	FUNCIONAMIENTO “DOBLE ALIMENTADOR”	74
5.3	FUNCIONAMIENTO “CINTA TRANSPORTADORA”	74
5.4	CREACIÓN DE LA APLICACIÓN PRÁCTICA	75
5.5	CONFIGURACIÓN EQUIPO MAESTRO	78
5.5.1	Configuración Hardware	78
5.5.2	Configuración Software.....	83
5.5.2.1	Bloques de programa.....	85
5.5.2.2	Programación para comunicación PROFIBUS DP.....	86
5.5.2.3	Programación para comunicación MPI	90
5.6	CONFIGURACIÓN EQUIPO ESCLAVO DP	96
5.6.1	Configuración módulo EM277	96
5.6.2	Configuración MODEM TC35	105
5.6.3	Configuración CPU 224.....	107
5.6.3.1	Programación para el control del módulo.....	108
5.6.3.2	Programación para comunicación PROFIBUS DP.....	114
5.6.3.3	Programación para comunicación GSM	117
5.7	CONFIGURACIÓN EQUIPO ESCLAVO MPI	133
5.7.1	Configuración módulo EM277	133
5.7.2	Configuración CPU 224.....	135
5.7.2.1	Programación para el control del módulo.....	135
5.7.2.2	Programación para comunicación MPI	138
5.8	CONFIGURACIÓN BUSES DE COMUNICACIÓN	139
5.8.1	Configuración bus PROFIBUS.....	140
5.8.2	Configuración bus MPI.....	142
5.9	CONFIGURACIÓN CABLES DE COMUNICACIÓN.....	144
5.9.1	Cable PC/PPI.....	144
5.9.2	Cable PC Adapter.....	144
5.10	CONFIGURACIÓN EQUIPO SCADA.....	145
5.10.1	Configurar equipo destino y equipo de control	145
5.10.2	Configurar imágenes	150
5.10.2.1	Imagen “Principal”	151

5.10.2.2	Imagen “Alimentador”	153
5.10.2.3	Imagen “Cinta”	154
5.10.3	Configurar variables.....	155
5.10.4	Configurar alarmas	156
5.10.4.1	Puntero de área.....	157
5.10.4.2	Alarmas.....	158
5.11	TRANSFERENCIA DE LA APLICACIÓN A LOS EQUIPOS.....	160
5.11.1	Ajustar interface PG/PC.....	161
5.11.2	Transferir configuración “Maestro”.....	163
5.11.3	Transferir configuración “Esclavo DP”	164
5.11.4	Transferir configuración “Esclavo MPI”	166
6	EXPERIMENTOS REALIZADOS	167
6.1	INTRODUCCIÓN.....	167
6.2	EXPERIMENTO 1: COMUNICACIÓN PROFIBUS	167
6.2.1	Configuración.....	167
6.2.2	Pruebas a realizar	169
6.3	EXPERIMENTO 2: COMUNICACIÓN MPI	169
6.3.1	Configuración.....	170
6.3.2	Pruebas a realizar	171
6.4	EXPERIMENTO 3: COMUNICACIÓN GSM.....	171
6.4.1	Configuración.....	171
6.4.2	Pruebas a realizar.....	172
6.5	EXPERIMENTO 4: PRUEBA FUNCIONAMIENTO MÓDULO “DOBLE ALIMENTADOR POR GRAVEDAD”.....	173
6.4.1	Modo automático.....	173
6.4.2	Modo manual.....	175
6.6	EXPERIMENTO 5: PRUEBA DE FUNCIONAMIENTO MÓDULO “CINTA TRANSPORTADORA”.....	177
7	CONCLUSIONES Y LÍNEAS FUTURAS	181
7.1	CONCLUSIONES	181
7.2	LÍNEAS FUTURAS	182
ANEXO A: DATOS TÉCNICOS DE LOS EQUIPOS	185	
A1. CPU 224 AC/DC/RELÉ.....	185	
A1.1. Dimensiones técnicas	185	
A1.2. Disposición de terminales	186	
A1.3. Datos técnicos	186	
A2. CPU 314C-2DP	189	
A2.1. Dimensiones técnicas	189	
A2.2. Disposición de terminales	189	
A2.3. Datos técnicos	190	
A3. FUENTE DE ALIMENTACIÓN PS307-2A.....	193	

<i>A3.1. Dimensiones técnicas</i>	193
<i>A3.2. Datos técnicos</i>	194
A4. MÓDULO DE COMUNICACIONES EM 277	194
A5. MÓDULO DE COMUNICACIONES TC35	195
A6. CABLE PC/PPI.....	196
A7. CABLE PC ADAPTER.	197
A8. CABLE PROFIBUS.	198
A9. CONECTOR PROFIBUS.	198
8 BIBLIOGRAFÍA Y REFERENCIAS.....	201

Capítulo 1

Motivación y objetivos

1.1 Motivación y objetivos

La motivación para la realización del presente Proyecto Fin de Carrera es la obtención del título de *Ingeniero Técnico Industrial, especialidad en Electrónica Industrial* por la *Universidad Politécnica de Cartagena*.

El trabajo ha sido realizado en el *Departamento de Ingeniería de Sistemas y Automática* de la *Universidad Politécnica de Cartagena*, y el objetivo principal es el estudio y posterior implementación de una red de comunicación industrial basada en tecnologías actuales de comunicación que permita la interconexión de diferentes autómatas programables. En concreto, se desarrollará una aplicación que permita, de forma remota, controlar y monitorizar dos procesos industriales (“módulos de fabricación flexible”) integrados en una red de comunicación basada en PROFIBUS, MPI y GSM.

1.2 Resumen y fases del proyecto

En estos momentos, la automatización de procesos supone, y supondrá aún más en un futuro próximo, uno de los puntos más críticos dentro de un ambiente industrial.

Esta afirmación está avalada por la ingente inversión que realizan las empresas en la adquisición de equipos de control, y por los avances obtenidos en el campo de los semiconductores – electrónica de potencia y microprocesadores -.

Igualmente, existe una clara tendencia a simplificar el cableado en los procesos industriales. Este apartado es donde se centra el presente proyecto. En primer lugar, se realiza una búsqueda de las soluciones en el campo de las comunicaciones industriales, en particular las comunicaciones a nivel de campo. En segundo lugar, se realiza la implementación de dos de ellas en un proceso industrial, a modo de comparación (en concreto empleando bus PROFIBUS y bus MPI). Además se ha realizado la implementación de una comunicación vía red GSM a través de mensajes cortos (SMS) con el objetivo de estudiar la utilidad y posibilidades de este tipo de comunicación.

Por último en el *Departamento de Ingeniería de Sistemas y Automática* se dispone de los equipos necesarios –autómatas programables, redes de comunicación, módulos de fabricación flexible –, que permiten llevar a cabo una aplicación de tipo industrial.

El proyecto se estructura en ocho Capítulos y dos Anexos.

El Capítulo 1 se basa en la introducción del proyecto.

En el Capítulo 2 se realiza una introducción a las redes de comunicación industrial y un análisis del Estado del Arte de dichas redes, con especial atención a los buses de campo.

El Capítulo 3 describe los equipos que se han utilizado en la red de comunicación.

El Capítulo 4 se dedica a la exposición de los pasos a seguir para el montaje de la red de comunicación.

El Capítulo 5 hace referencia al proceso de configuración y puesta en marcha de la red de comunicación.

El Capítulo 6 recoge una serie de experimentos prácticos realizados sobre la red de comunicación.

Las conclusiones y posibles líneas futuras es el tema que aborda el Capítulo 7. En él se exponen las conclusiones obtenidas con el presente trabajo.

En el Anexo A se incluyen una serie de datos técnicos de los equipos utilizados en este proyecto.

Finalmente se hace referencia de la bibliografía utilizada en el Capítulo 8.

Fases del proyecto

- Estudio del Estado del Arte de las comunicaciones industriales entre autómatas programables, en concreto a nivel de campo.
- Estudio del hardware necesario para la puesta en marcha de una red de comunicación empleando buses PROFIBUS y MPI y comunicación vía red GSM.
- Montaje y configuración de dicha red.
- Integración de dos módulos de fabricación en la red.
- Implementación de un sistema SCADA para control y supervisión de ambos módulos a través de la red.
- Conclusiones y líneas futuras.

Capítulo 2

Redes de comunicación entre autómatas. Soluciones actuales

2.1 Introducción

Desde siempre, el hombre ha intentado hacer su vida más fácil y cómoda. Este fenómeno es lo que podemos llamar la “evolución tecnológica” y que en estos últimos siglos ha sido espectacular si comparamos con etapas anteriores.

Hasta el siglo XX, el desarrollo tecnológico había venido de la mano de la revolución industrial en la que los sistemas mecánicos eran piezas clave para la industria y el desarrollo de los procesos industriales. Pero es en el siglo XX cuando los sistemas de automatización surgen con más fuerza, potenciados por los sistemas eléctricos basados en relés electromagnéticos en la primera mitad del siglo, y de los sistemas electrónicos de estado sólido en su segunda mitad. Así, además de sus

objetivos iniciales, el concepto de automatización se extiende a la mejora de producción y calidad, disminución de riesgos laborales, disminución de costes, etc.

La aparición de los autómatas programables marcó un cambio en la industria de ensamblaje y líneas de producción. El concepto de línea de producción automatizada apareció entonces como un hecho real. Este pequeño dispositivo permitió que los procesos industriales fuesen desde entonces más eficientes, precisos, y lo que es más importante, reprogramables, eliminando el gran costo que se producía al reemplazar el complejo sistema de control basado en relés y contactores, tanto por tamaño como por vida útil.

Pero la mejora en los procesos de automatización pasa hoy en día por el desarrollo de las redes de comunicación. La intercomunicación de sistemas y procesos industriales no es un concepto nuevo, pues es ampliamente conocido el uso de sistemas como IEEE-488 y RS485/422 que durante más de 20 años han sido capaces de ofrecer los requerimientos necesarios en las instalaciones de baja y media complejidad en cuanto a las capacidades de intercomunicación se refiere. Este tipo de enlaces entre sistemas se ha empleado esencialmente para equipos de instrumentación y sistemas de automatización, donde es necesaria una baja tasa de transferencia de datos entre equipos, pero que en gran número de casos hoy en día ya no puede responder a las necesidades de intercomunicación entre dispositivos que se demandan. Por ello, redes de comunicación como *PROFIBUS* han conseguido ser indispensables en un entorno de trabajo donde cada día es más necesaria la integración global.

Por tanto, los sistemas de automatización industrial y su funcionamiento distribuido forman parte de un concepto de mayor entidad, encaminado a la optimización global de las industrias mediante la mejora de los procesos de fabricación, la reducción de costes, el incremento en la calidad, la mejora de la eficiencia, la mayor flexibilidad en los procesos de producción y, en general, todos aquellos factores que permitan adaptarse de manera eficiente a las necesidades del mercado al que está orientado el producto.

2.2 Sistemas industriales de control

Se distinguen tres tipos de sistemas de control industrial: control centralizado, control híbrido y control distribuido. La importancia de las tareas a realizar, 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 un tipo u otro de control.

2.2.1 Control centralizado

Esta aproximación es la que se sigue en el caso de sistemas poco complejos, donde un proceso puede ser gestionado directamente mediante un único elemento de control encargado de realizar todas las tareas del proceso de producción y que puede incluir un sistema de monitorización y supervisión. Conforme las necesidades de producción han requerido mayor complejidad, una tendencia ha sido la de emplear elementos de control más complejos y potentes, manteniendo en un único elemento todo el control del proceso, con la complejidad que ello supone, ya que se hace necesario hacer llegar todas las señales de sensores y cablear todos los actuadores allá donde se encuentren.

Como ventajas de esta metodología se tiene que no es necesario planificar un sistema de intercomunicación entre procesos, ya que todas las señales están gestionadas por el mismo sistema. Por otro lado, para sistemas poco complejos, posee un menor coste económico. En cambio, posee numerosas desventajas, ya que si el sistema falla, toda la instalación queda paralizada, siendo necesario un sistema redundante para evitar estas situaciones. También se hace necesario el empleo de unidades de control (generalmente autómatas programables) de mayor capacidad de proceso, dada la complejidad de los problemas que debe abordar y las restricciones de tiempo límite que son habituales en los procesos industriales.

Por otro lado, el cableado puede aumentar notablemente debido a las mayores distancias que pueden existir entre los sensores, actuadores y la unidad de control, aunque este problema se pueda simplificar en cierta medida debido al uso de buses de campo.

2.2.2 Control distribuido

La opción de control distribuido requiere que puedan considerarse procesos, grupos de procesos o áreas funcionales susceptibles de ser definidas por un algoritmo de control que pueda realizarse de forma autónoma. A cada unidad se destinará un autómata (o elemento de control) dimensionado de acuerdo con los requerimientos del proceso considerado. Debido a la interdependencia que existe entre las operaciones que tienen lugar en cada proceso, hay que tener en cuenta que es necesario interconectar los autómatas entre sí mediante entradas y salidas digitales, o a través de una red de comunicaciones para intercambio de datos y estados. Por tanto, el autómata o elemento de control evaluado debe permitir las comunicaciones.

Con esta metodología de control es posible que cada unidad funcional consista en un proceso relativamente sencillo comparado con el proceso global, reduciendo la posibilidad de errores en la programación y permitiendo el empleo de unidades de control (autómatas programables principalmente) más sencillas y, por tanto, más

económicas. Al mismo tiempo, la existencia de fallos en otras unidades de control no implica necesariamente la paralización de todos los procesos que se llevan a cabo en la planta. Como desventaja, es necesario realizar un estudio de implantación previo, ya que se deben identificar los procesos autónomos, asignar elementos a cada proceso y diseñar el modelo de intercomunicación para responder a las necesidades del proceso planteado.

2.2.3 Control híbrido

El control híbrido no está muy bien definido, ya que este tipo de gestión de planta puede considerarse a cualquier estrategia de distribución de elementos de control a medio camino entre el control distribuido y el control centralizado. En numerosas ocasiones no resulta sencillo separar los procesos de manera completamente autónoma, por lo que se debe recurrir a la gestión de varios procesos desde una misma unidad de control, debido a que la complejidad de la separación es mayor que la complejidad que supone su gestión conjunta. Por otro lado, una estrategia de este tipo también conduce a una gestión estructurada, de modo que existen elementos de control de nivel superior que supervisan e intercomunican los procesos autónomos más sencillos, siendo los encargados de gestionar la información común. Para este tipo de gestión también es necesario el uso de redes de comunicación.

2.3 La pirámide CIM

El ideal de factoría completamente automatizada (*Computer Integrated Manufacturing*) se representa como una pirámide (Fig. 2.1.) en la que en los niveles bajos se encuentran los sensores y actuadores; en los niveles intermedios se interconectan estos elementos para funcionar cooperativamente realizando funciones más o menos sincronizadas y finalmente, en el nivel superior aparece la red informática técnico-administrativa donde se recogen informaciones de estado, registros históricos, datos de partida, consignas, etc.

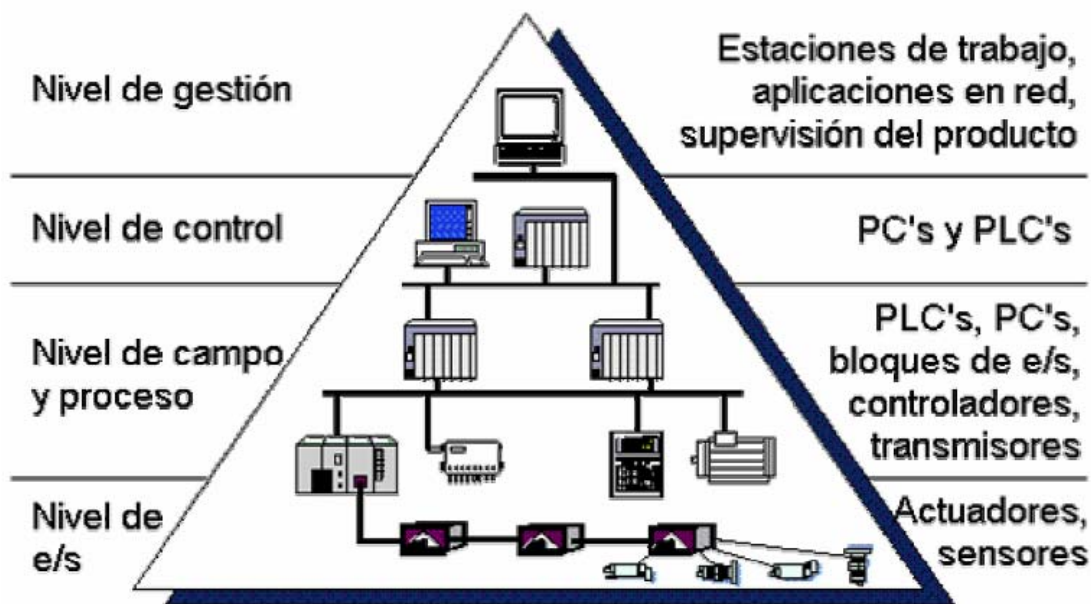


Fig. 2.1. Pirámide CIM de comunicación.

2.3.1 Nivel de E/S (Nivel actuador/sensor)

También llamado nivel de instrumentación. Está formado por los elementos de medida (sensores) y mando (actuadores) distribuidos en una línea de producción. Son los elementos más directamente relacionados con el proceso productivo. Así, los actuadores son los encargados de ejecutar las órdenes de los elementos de control para modificar el proceso productivo y los sensores miden variables en el proceso de producción. Ejemplos de sensores son aquellos que permiten medir nivel de líquidos, caudal, temperatura, presión, posición, etc. Como ejemplos de actuadores se tienen los motores, válvulas, calentadores, taladros, cizallas, etc. Como característica adicional, los sensores y actuadores suelen ser dispositivos que necesitan ser controlados por otros elementos.

2.3.2 Nivel de Campo y Proceso

En este nivel se sitúan los elementos capaces de gestionar los actuadores y sensores del nivel anterior, tales como autómatas programables o equipos de aplicación específica basados en microprocesador como robots, máquinas herramienta o controladores de motor. Estos dispositivos permiten que los actuadores y sensores funcionen de forma conjunta para ser capaces de controlar el proceso industrial deseado.

Los dispositivos de este nivel de control, junto con los del nivel inferior de entrada/salida, poseen entidad suficiente como para realizar procesos productivos por sí mismos. De hecho, gran cantidad de procesos industriales están basados exclusivamente en estos dos niveles, de tal modo que un proceso productivo completo se desglosa en subprocesos de este tipo sin que exista un intercambio de información entre ellos (excepto algunas señales de control para sincronizar el fin de un proceso con el inicio del siguiente). Son pues dispositivos programables, de tal modo que es posible ajustar y personalizar su funcionamiento según las necesidades de cada caso.

No obstante, a pesar de que puedan presentarse como procesos aislados, esto no implica que no se empleen buses de comunicación, ya que para procesos que requieran de un gran número de sensores y actuadores, es recomendable la utilización de buses de campo para leer el estado de los sensores, proporcionar señales de control a los actuadores y conectar diferentes autómatas programables para compartir información acerca de la marcha del proceso completo.

También es importante que estos dispositivos posean unas buenas características de interconexión, para ser enlazados con el nivel superior (Nivel de control), generalmente a través de buses de campo.

2.3.3 Nivel de Control (Nivel de célula)

Todos los dispositivos de control existentes en planta es posible monitorizarlos si existe un sistema de comunicación adecuado, capaz de comunicar estos elementos con otros tipo de dispositivos dedicados a la gestión y supervisión, que habitualmente están constituidos por computadores o sistemas de visualización tales como pantallas industriales.

En este nivel es posible visualizar cómo se están llevando a cabo los procesos de planta y, a través de entornos SCADA (Supervisión, Control y Adquisición de Datos), poseer una “imagen virtual de la planta” de modo que ésta se puede recorrer de manera detallada, o bien mediante pantallas de resumen ser capaces de disponer de un “panel virtual” donde se muestren las posibles alarmas, fallos o alteraciones en cualquiera de los procesos que se llevan a cabo. Mediante este tipo de acciones es posible disponer de acceso inmediato a cada uno de los sectores de la planta. Para ello, resulta imprescindible la conexión con el nivel de control mediante buses de campo o en este caso pueden emplearse redes LAN industriales de altas prestaciones, pues a veces es necesaria la transmisión de importantes cantidades de datos y la conexión con un gran número de elementos de control. Por ejemplo, en un proceso industrial que consta de varias fases para realizar un determinado producto se utilizan varios autómatas para cada proceso, por lo que un sistema de supervisión debe ser capaz de acceder al estado de cada uno de ellos, visualizar el proceso que lleva a cabo y, de manera global, tener información de cómo está trabajando cada uno individualmente, así como poder acceder

a informes generados por el autómata. También es posible modificar los procesos productivos desde los computadores de supervisión. Este nivel sustituye a los grandes paneles y salas de control que durante los años 70 y 80 eran habituales en las grandes empresas.

2.3.4 Nivel de gestión (Nivel de fábrica)

El nivel de gestión estará principalmente constituido por computadores, ya que se encuentra más alejado de los procesos productivos. De hecho, en este nivel no es relevante el estado y la supervisión de los procesos de planta, en cambio, sí adquiere importancia toda la información relativa a la producción y su gestión asociada, es decir, a través del nivel de control es posible obtener información global de todos los niveles inferiores de una o varias plantas. Con esta información, los gestores de la empresa pueden extraer estadísticas acerca de los costes de fabricación, rendimiento de la planta, estrategias de ventas para liberar posibles excesos de producto almacenado y, en general, disponer de datos que permitan a los niveles directivos la toma de decisiones conducentes a una mejor optimización en el funcionamiento de la planta, todo ello de una manera rápida y flexible. Las comunicaciones con este nivel de la pirámide industrial ya no necesitan ser de tipo estrictamente industrial, es decir, muy robustas, de corto tiempo de respuesta, ..., sino que ahora lo verdaderamente importante son los datos que se transmiten, informes que pueden tener un tamaño medio-grande. Por este motivo habitualmente se emplean redes de comunicación menos costosas, como redes *Ethernet*, que se adaptan mejor al tipo de datos que se desean transmitir y, además, permiten la comunicación eficaz entre los diferentes computadores del mismo nivel de gestión.

2.4 Redes de comunicación industriales

Las comunicaciones deben poseer unas características particulares para responder a las necesidades de intercomunicación en tiempo real. Además, deben resistir un ambiente hostil donde existe gran cantidad de ruido electromagnético y condiciones ambientales duras. En el uso de comunicaciones industriales se pueden separar dos áreas principales: una comunicación a nivel de campo, y una comunicación hacia el SCADA. En ambos casos la transmisión de datos se realiza en tiempo real o, por lo menos, con una demora que no es significativa respecto de los tiempos del proceso, pudiendo ser crítico para el nivel de campo.

Según el entorno donde van a ser instaladas, dentro de un ámbito industrial, existen varios tipos de redes:

2.4.1 Red de Factoría

Para redes de oficina, contabilidad y administración, ventas, gestión de pedidos, almacén, etc. El volumen de información intercambiada es muy alto, y los tiempos de respuesta no son críticos.

2.4.2 Red de Planta

Interconexión de módulos y células de fabricación entre sí y con departamentos como diseño o planificación. Suele emplearse para el enlace entre las funciones de ingeniería y planificación con las de control de producción en planta y secuenciamiento de operaciones. Como ejemplo, se tiene la transmisión a un sistema de control numérico del programa de mecanizado elaborado en el departamento de diseño CAD/CAM. Estas redes deben manejar mensajes de cualquier tamaño, gestionar eficazmente errores de transmisión (detección y corrección), cubrir áreas extensas (puede llegar a varios kilómetros), gestionar mensajes con prioridades (gestión de emergencias frente a transferencia de ficheros CAD/CAM), y disponer de amplio ancho de banda para admitir datos de otras subredes como pueden ser voz, vídeo, etc.

2.4.3 Red de Célula

Interconexión de dispositivos de fabricación que operan en modo secuencial, como robots, máquinas de control numérico (CNC), autómatas programables (PLC), vehículos de guiado automático (AGV). Las características deseables en estas redes son: gestionar mensajes cortos eficientemente, capacidad de manejar tráfico de eventos discretos, mecanismos de control de error (detectar y corregir), posibilidad de transmitir mensajes prioritarios, bajo coste de instalación y de conexión por nodo, recuperación rápida ante eventos anormales en la red y alta fiabilidad.

2.4.4 Bus de Campo

Un bus de campo es, en líneas generales, *“un sistema de dispositivos de campo (sensores y actuadores) y dispositivos de control, que comparten un bus digital serie bidireccional para transmitir informaciones entre ellos, sustituyendo a la convencional transmisión analógica punto a punto”*. Permiten sustituir el cableado entre sensores-actuadores y los correspondientes elementos de control. Este tipo de buses debe ser de bajo coste, de tiempos de respuesta mínimos, permitir la transmisión serie sobre un bus digital de datos con capacidad de interconectar controladores con todo tipo de dispositivos de entrada-salida, sencillos, y permitir controladores esclavos inteligentes.

2.4.4.1 Panorámica actual de los buses de campo

Las señales de procesos industriales, originadas a pie de máquina, se transmiten normalmente con un extenso cableado punto a punto, incluso haciendo uso de transmisores “inteligentes”. Esto significa que cada sensor o actuador situado en campo se encuentra conectado a los módulos de entrada/salida de los PLCs, utilizando un par de hilos por instrumento.

Cuando la distancia entre el instrumento y sistema de control comienza a ser considerable o cuando existen en el proceso un gran número de instrumentos, debemos tener en cuenta los costos de cableado, sobre todo cuando se establece la necesidad de un número extenso de conductores de reserva, de cara a futuras ampliaciones. Por estas razones, en la actualidad se está implantando definitivamente la filosofía de bus de campo. Con este sistema es posible la sustitución de grandes haces de conductores por un simple cable bifilar o fibra óptica (Fig. 2.2.), común para todos los sensores y actuadores, con el consiguiente ahorro económico que ello supone. La comunicación de la variable de proceso será totalmente digital.

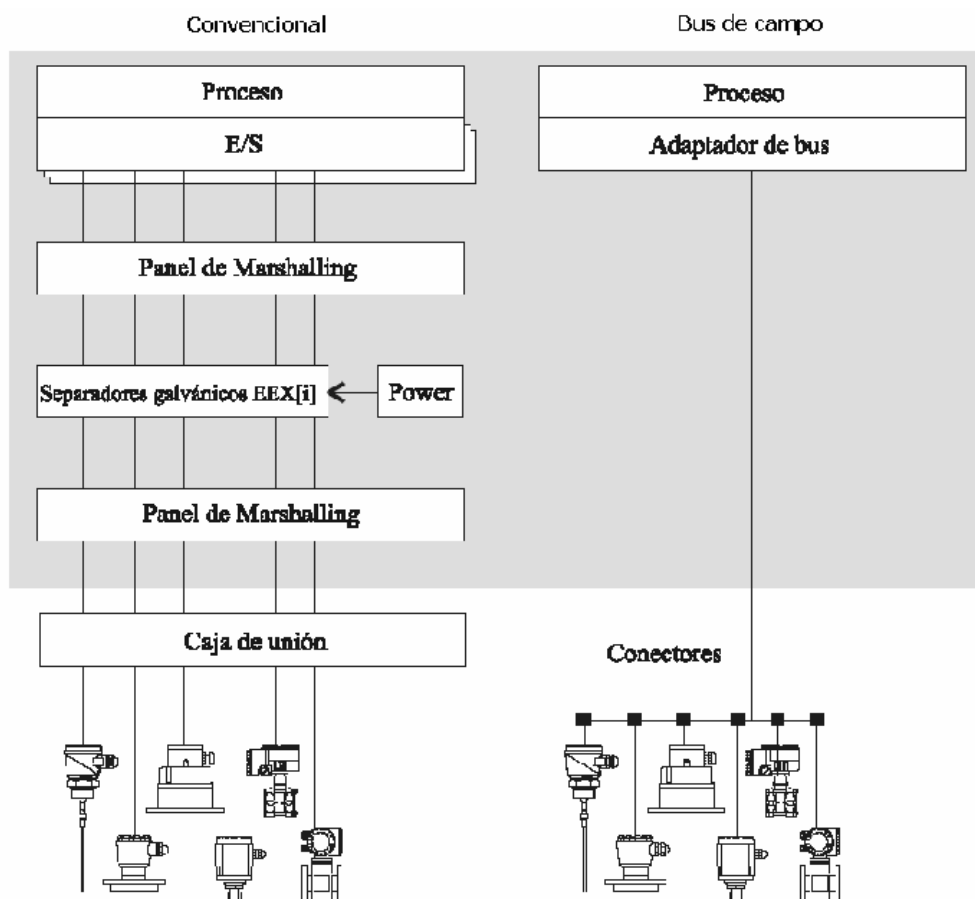


Fig. 2.2. Sistema de cableado convencional versus Bus de Campo.

Inicialmente, los buses de campo están muy poco normalizados, por lo que existe una gran variedad de ellos con diferentes características dependiendo de a qué aplicaciones estén destinados. Lo cierto es que actualmente cabe afirmar que los buses de campo están llegando a un período de madurez, planteándose la convivencia de un número reducido de estándares con posibles soluciones de comunicación entre ellos.

2.4.4.2 Buses de Campo y Niveles OSI.

Idealmente, las especificaciones de un bus de campo deberían cubrir los siete niveles OSI, aunque lo más frecuente es que implementen sólo tres:

- *Nivel físico.*
Especifica el tipo de conexión, naturaleza de la señal, tipo de medio de transmisión, etc. Normalmente, las especificaciones de un determinado bus admiten más de un tipo de medio físico. Los más comunes son de tipo RS485 o con conexiones en bucle de corriente.
- *Nivel de enlace.*
Se especifican los protocolos de acceso al medio (MAC) y de enlace (LLC). En este nivel se definen una serie de funciones y servicios de la red mediante códigos de operación estándar.
- *Nivel de aplicación.*
Es el dirigido al usuario, y permite la creación de programas de gestión y presentación, apoyándose en las funciones estándar definidas en el nivel de enlace. En este nivel se define el significado de los datos. Las aplicaciones suelen ser propias de cada fabricante (no hay un nivel de aplicación estándar para buses de campo).

2.4.4.3 Buses Proprietarios y Buses Abiertos

La existencia de un elevado número de buses de campo diferentes se debe a que cada compañía venía utilizando un sistema propio para sus productos, aunque en los últimos años se observa una cierta tendencia a utilizar buses comunes.

En buses de campo podemos distinguir:

- *Buses propietarios.*
Son propietarios de una compañía o grupo de compañías, y para utilizarlos es necesario obtener una licencia, que es concedida a la empresa que la disfruta con una serie de condiciones asociadas, y a un precio considerable.

- *Buses abiertos.*

Son todo lo contrario:

- Las especificaciones son públicas y disponibles a un precio razonable.
- Los componentes críticos (como por ejemplo microprocesadores) también están disponibles.
- Los procesos de validación y verificación están bien definidos y disponibles en las mismas condiciones anteriores.

Las características fundamentales que el bus de campo debe cumplir, en lo referente a la conexión de dispositivos, son:

- **Interconectividad:** al bus se deben poder conectar de forma segura dispositivos de diferentes fabricantes que cumplan el protocolo. Es el nivel mínimo, y no proporciona, en principio, ninguna ventaja.
- **Interoperatividad:** los dispositivos de diferentes fabricantes funcionan satisfactoriamente en el mismo bus.
- **Intercambiabilidad:** los dispositivos de un fabricante pueden ser sustituidos por otros equivalentes, de otro fabricante, y seguir funcionando. Este es el objetivo final, y sólo se consigue si las especificaciones son completas y se dispone de un sistema de prueba y validación.

2.4.4.4 Ventajas e Inconvenientes de los Buses de Campo

Los buses de campo, si son correctamente elegidos para la aplicación, ofrecen numerosas ventajas, como:

- **Flexibilidad.** El montaje de un nuevo instrumento supone la simple conexión eléctrica al bus y una posterior configuración/programación, normalmente remota (desde la sala de control). Si se trata de buses abiertos, resultará posible la conexión de instrumentos de distintos fabricantes al mismo bus.
- **Seguridad.** Transmisión simultánea de señales de diagnóstico de sensores y actuadores, permitiendo así instalaciones más seguras.
- **Precisión.** Transmisión totalmente digital para variables analógicas.
- **Facilidad de mantenimiento.** Resulta posible diagnosticar el funcionamiento incorrecto de un instrumento y realizar calibraciones de forma remota desde la sala de control. Esto permite localizar rápidamente conexiones erróneas en la instalación, con lo que los errores de conexión son menores y más rápidamente solucionados (reducción de los tiempos de parada y pérdidas de producción).
- **Reducción de la complejidad** del sistema de control en términos de hardware:
 - Reducción drástica del cableado.

- Se elimina la necesidad de grandes armarios de conexiones para el control del equipamiento asociado.
- Reducción del número de PLCs.
- Reducción de tiempo de instalación y personal necesario para ello.

Por el contrario, el principal inconveniente que ofrece la utilización de un bus de campo es la posible **rotura del cable** de bus. Esto conllevaría la caída de todos los elementos que estuvieran conectados al bus y probablemente una parada general del proceso. Hay también que advertir que en la actualidad los buses de campo son muy robustos ante interferencias y entornos agresivos.

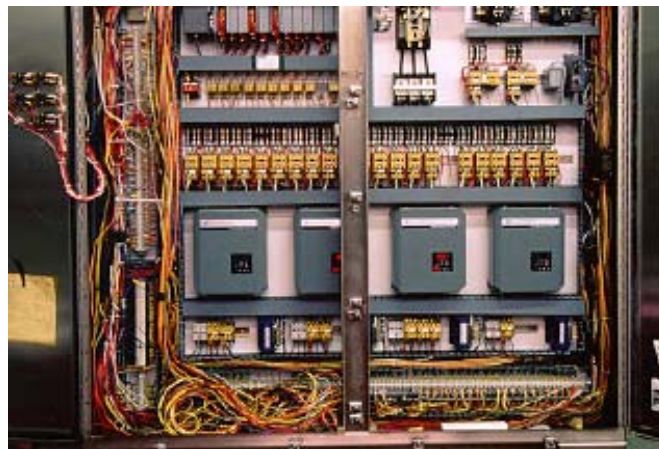


Fig. 2.3. Instalación industrial sin utilización de buses de campo.



Fig. 2.4. Instalación industrial con utilización de buses de campo.

2.4.4.5 Normalización

Se han realizado muchos intentos de normalización de buses de campo. Finalmente se establecieron una serie de reglas genéricas, incluidas en una norma de la IEC (comité TC65C-WG6). Dichas recomendaciones son:

- Nivel físico: bus serie controlado por maestro. Comunicación semidúplex en banda base.
- Velocidades: 1 Mbit/s para distancias cortas y de 64-250 Kbit/s para distancias largas.
- Longitudes: 40 m para la máx. velocidad y 350 m para velocidades más bajas.
- Número de periféricos: máx. de 30 nodos con posibles ramificaciones hasta 60 elementos.
- Cable: par trenzado apantallado.
- Conectores: bornes industriales DB9/DB25.
- Conexión-desconexión en caliente (on-line).
- Topología: bus físico con posibles derivaciones a nodos.
- Longitud máx. de las ramificaciones: 10 m.
- Aislamientos: 500VCA entre elementos de bus y campo.
- Seguridad intrínseca: opción de conectar elementos de campo con tensiones reducidas para atmósferas explosivas.
- Alimentación: opción de alimentación a través del bus.
- Longitud mínima del mensaje: 16 bits.
- Transmisión de mensajes: posibilidad de diálogo entre cualquier par de nodos sin repetidor.
- Maestro flotante: posibilidad de maestro flotante entre nodos.
- Implementación del protocolo: los chips para el protocolo deben estar disponibles comercialmente y no protegidos por patente.

Casi todas las especificaciones que se dan son de nivel físico, y están muy abiertas en los niveles de enlace y de aplicación, de ahí las grandes diferencias en modos de configuración entre unos buses y otros.

2.4.4.6 Buses existentes en el mercado

A continuación se muestra una tabla comparativa entre los buses de campo de más utilización en la industria, haciendo referencia a sus características técnicas más importantes.

BUS DE CAMPO	TOPOLOGÍA	MEDIO FÍSICO	VELOCIDAD	DISTANCIA SEGMENTO	NODOS POR SEGMENTO	ACCESO AL MEDIO
P-NET	Anillo	Par trenzado apantallado	76'8 Kbps	1.200 m	125	Paso de testigo Maestro/esclavo
PROFIBUS	Bus lineal Anillo Estrella Árbol	Par trenzado apantallado Fibra óptica	Hasta 12Mbps	Hasta 9'6 Km y 90 Km	125	Paso de testigo Maestro/esclavo
WORLDFIP	Bus lineal	Par trenzado apantallado Fibra óptica	Hasta 1 Mbps y 5Mbps	Hasta 5 Km y 20 Km	64	Arbitro de bus
HART	Bus lineal	Cable 2 hilos	1'2Kbps	3.000 m	30	Maestro/esclavo
MODBUS	Bus lineal	Par trenzado	Hasta 19'2Kbps	1 Km	248	Maestro/esclavo
INTERBUS-S	Anillo	Par trenzado	500 Kbps	400 m	256	Paso de testigo
BITBUS	Bus lineal	Par trenzado Fibra óptica	Hasta 1'5Mbps	Hasta 1.200m	29	Maestro/esclavo
MEASUREMENT BUS	Bus lineal	Cable 4 hilos	Hasta 1 Mbps	500 m	31	Maestro/esclavo
CAN	Bus lineal	Par trenzado	Hasta 1 Mbps	Hasta 1.000m	127-64	CSMA/CD con arbitraje de bit
SDS	Bus lineal	Cable de 4 hilos	Hasta 1 Mbps	500 m	64	CSMA
DEVICENET	Bus lineal	Par trenzado	Hasta 500 Kbps	Hasta 500 m	64	CSMA/CDBA
CONTROLNET	Bus lineal Árbol Estrella	Coaxial Fibra óptica	5 Mbps	Hasta 3.000m	48	CTDMA
SERIPLEX	Bus lineal	Cable 4 hilos apantallado	98 Kbps	1.500m	300	Maestro/esclavo
AS-i	Bus lineal Árbol Estrella	Cable 2 hilos	167 Kbps	Hasta 200 m	32-62	Maestro/esclavo
LON WORKS	Bus Anillo Libre	Par trenzado Fibra óptica Red eléctrica Coaxial Radio Infrarrojos	Hasta 1'25 Mbps	Hasta 2.700 m	64	CSMA/CA
ARCNET	Bus Estrella	Par trenzado Fibra óptica Coaxial	2'5 Mbps	122 m	255	Paso de testigo
M-BUS	Bus lineal	Cable 2 hilos	Hasta 9'6 Kbps	1.000 m	250	Arbitro de bus
UNI-TELWAY	Bus lineal	Par trenzado apantallado	Hasta 19'2Kbps	20 m	Hasta 28	Maestro/esclavo
COMPOBUS/S	Bus lineal	Cable de 2 ó 4 hilos	Hasta 750 Kbps	Hasta 500 m	32	Maestro/esclavo

2.4.4.6.1 P-Net

El protocolo P-NET tiene su nacimiento en la industria danesa, siendo publicado originalmente por la Internacional P-NET User Organisation ApS y más tarde recogido por el CENELEC.

Para la conexión física usa el estándar RS-485 con transmisión asíncrona a 76.800 bps. Es una norma multiprincipal y multired, es decir, varios principales (másters) pueden conectarse al mismo bus y varios buses pueden interconectarse formando una red mayor mediante pasarelas (gateways). La segmentación hace posible que cada segmento de bus tenga un tráfico local independiente, con lo que se incrementa el ancho de banda del sistema global. Hay 3 tipos de dispositivos que pueden ser conectados a una red P-NET: principales (másters), subordinados (slaves) y pasarelas (gateways). Todas las comunicaciones están basadas en el principio de que un principal envía una petición y la estación subordinada direccionada devuelve una respuesta. Es un sistema muy robusto con respecto a los errores, realizándose continuamente un control de errores por parte de las estaciones subordinadas, siendo notificado el principal si se produce un error.

2.4.4.6.2 PROFIBUS (PROcess FIEld BUS)

Este protocolo ha sido impulsado por los fabricantes alemanes (ABB, AEG, Siemens, Bauer, Danfoss, Klóckner, Móeller, etc.) en el año 1987, y se normalizó como DIN 19245. Los primeros productos que se ajustan a esta norma comenzaron a aparecer en 1989 y se creó un grupo de usuarios denominado PROFIBUS Nutzerorganisation (PNO). En el año 1996 este protocolo fue recogido por



la CENELEC en su norma EN 50170.

Este bus de campo ha sido el implementado en este proyecto.

2.4.4.6.3 WorldFIP (World Factory Instrumentation Protocol)

Este protocolo ha sido impulsado por los fabricantes franceses (Cegelec, Telemecanique, Efisysteme, Gespac, etc.) y se normalizó con el nombre de FIP como la norma francesa NFC 46 601Π605 . En el año 1996, este protocolo fue recogido por la CENELEC en su norma EN 50170 con el nombre de WorldFIP, adoptando para la capa física la norma IEC 1158-2. Es un bus de campo diseñado para establecer comunicaciones entre el nivel de sensores/actuadores y el nivel de unidades de proceso (PLC, consoladores, etc.) en los sistemas automatizados.

Existen estaciones con funcionalidad arbitro de bus y estaciones con funcionalidad consumidora/productora, pudiendo tener una estación simultáneamente las dos funcionalidades, pero sólo una estación puede ejercer la funcionalidad de arbitro de bus en la red en un instante determinado.

2.4.4.6.4 HART (Highway Addressable Remote Transducer)

HART es un protocolo de comunicación digital que opera sobre un bucle de corriente convencional 4-20 mA. Utiliza una onda senoidal de baja frecuencia como portadora analógica de



la información digital, mediante modulación por desplazamiento de frecuencia (FSK: Frequency Shift Keying). Utiliza una frecuencia de 1.200 Hz para codificar un "1" y una frecuencia de 2.200 Hz para codificar un "0". Al tener su valor medio nulo, la señal modulada no afecta a la corriente del bucle. La velocidad de transferencia de este protocolo es de 1.200 bps. y puede alcanzar distancias de 3.000 metros con el uso de cable de par trenzado apantallado.

El método de acceso al medio es un método de sondeo y selección principal-subordinado. Pueden existir hasta dos estaciones principales (primaria y secundaria). En un mensaje HART pueden ser transmitidas hasta cuatro variables de proceso y cada dispositivo puede tener un máximo de 256. Existe un modo de comunicación en ráfaga (burst), mediante el cual un subordinado difunde continuamente un mensaje de respuesta.

Admite configuraciones punto a punto y multipunto (multidrop). En las configuraciones multipunto se puede conectar hasta 30 dispositivos sobre un cable de 2 hilos. En este caso no se utiliza la señal analógica y toda la información es transmitida mediante la comunicación digital HART.

El protocolo HART contempla tres tipos de órdenes: universales, de uso habitual (common practice) y específicas. Las órdenes universales son entendibles por todos los dispositivos HART (por ejemplo, lectura del nombre del fabricante y tipo de dispositivo). Las órdenes de uso habitual son entendibles por la mayoría de los dispositivos HART, pero no por todos. Las órdenes específicas son exclusivas de cada dispositivo HART. Existe un lenguaje de descripción de dispositivos (DDL: Device Description Language) que trata de aumentar la interoperabilidad a un mayor nivel que el proporcionado por las órdenes universales y de uso habitual, pudiendo configurar cualquier dispositivo HART mediante este lenguaje.

2.4.4.6.5 Modbus

El protocolo Modbus fue desarrollado por la firma Modicon (ahora perteneciente al grupo Schneider) y se utiliza principalmente en el continente americano. El bus se compone de una estación activa (principal) y de varias estaciones pasivas (subordinadas). La estación principal es la única que puede tomar la iniciativa de intercambio de información, no pudiendo las estaciones subordinadas comunicarse directamente. Existen dos mecanismos de intercambio:



- **Pregunta/respuesta:** La estación principal transmite preguntas a una subordinada determinada, que a su vez transmite una respuesta a la principal.
- **Difusión:** La estación principal transmite un mensaje a todas las estaciones subordinadas del bus, que ejecutan la orden sin transmitir ninguna respuesta.

En una red Modbus existen 1 estación principal y hasta 247 estaciones subordinadas (direcciones en el rango 1 a 247). Sólo la principal puede iniciar una transacción. Para comunicarse con las estaciones subordinadas, la principal envía unas tramas que llevan: la dirección del receptor, la función a realizar, los datos necesarios para realizar dicha función y un código de comprobación de errores. Cuando la trama llega a la estación subordinada direccionada, ésta lee el mensaje, y si no ha ocurrido ningún error realiza la tarea indicada. Entonces la subordinada envía una trama respuesta formada por: la dirección de la subordinada, la acción realizada, los datos adquiridos como resultado de la acción y un código de comprobación de errores. Si el mensaje enviado por la principal es de tipo difusión (broadcast), o sea, para todas las estaciones subordinadas (se indica con dirección 0), no se transmite ninguna respuesta. Si la estación receptora recibe un mensaje con algún error, contesta a la principal con un código de error (Función ilegal. Datos de direccionamiento ilegales. Datos de valores ilegales. Fallo en el dispositivo, Mensaje rechazado).

En la mayoría de los casos la estación principal puede enviar otro mensaje a cualquier subordinada tan pronto como recibe una respuesta válida, o después de un intervalo de tiempo seleccionable si no recibe respuesta.

En Modbus existen dos posibles modos de transmisión para las estructuras de las unidades de información (caracteres) que forman el mensaje:

- **ASCII (American Standard Code for Information Interchange).** El sistema de codificación es hexadecimal y cada carácter consta de 1 bit de inicio, 7 bits de codificación de los datos, 1 bit de paridad (opcional) y 1 o 2 bits de parada, o sea, un total de 9 a 11 bits por carácter.
- **RTU (Remote Terminal Unit).** El sistema de codificación es binario y cada carácter consta de 1 bit de inicio, 8 bits de codificación de los datos, 1 bit de paridad (opcional) y 1 o 2 bits de parada, o sea, un total de 10 a 12 bits por

carácter. Los dispositivos Modbus usan interfaces serie compatibles con RS-232C y RS-485, siendo el bus capaz de transferir datos a velocidades de 19'2 Kbps y alcanzar distancias de 1 Km.

2.4.4.6.6 Interbus

El bus sensor/actuador Interbus fue desarrollado por la empresa alemana Phoenix Contact a mediados de los años ochenta, y se normalizó como DIN 19258. Phoenix Contact ofrece con este protocolo una familia de soluciones de bus de campo perfectamente acoplables entre sí, aplicándose en una gran diversidad de sectores, como son la industria del automóvil, industria papelera y de impresión, industria alimentaria e industria textil.



Interbus trabaja con un sistema de acceso principal/subordinado, siendo topológicamente un sistema en anillo donde el conductor de datos de ida y de retorno se introduce dentro de un cable que pasa por todos los nodos. En el anillo que parte del principal pueden conectarse subsistemas en anillo subordinados para estructurar el sistema completo. Un sistema subordinado puede tener carácter local (bus periférico) que sirve para formar grupos de estradas/salidas locales dentro de un armario, o puede ser un sistema que acopla participantes descentralizados a lo largo de distancias grandes. El nivel físico se realiza con el estándar RS-485, utilizando cables de par trenzado y precisando 5 hilos debido a la estructura del anillo y a la conducción adicional de una tierra lógica. El empleo de la estructura en anillo aporta dos ventajas determinantes para el sistema: por un lado, ofrece la posibilidad de emisión y recepción simultánea de datos (full dúplex) y, por otro lado, en un sistema en anillo se puede conseguir un mejoramiento del diagnóstico propio del sistema, ya que un sistema en anillo con acoplamiento activo de nodos permite una segmentación de la instalación en sistemas parciales eléctricamente independientes, pudiendo así detectar el punto de un fallo.

La transferencia de datos a los nodos individuales no se efectúa a través de una dirección de bus como sucede en otros sistemas, sino a través de la posición física que tiene en el sistema en anillo. Adicionalmente a este direccionamiento automático físico de los participantes en el bus, puede efectuarse en el principal del bus un direccionamiento lógico de elección libre mediante la elaboración de una lista de asignación de direcciones. Así, las direcciones de estaciones utilizadas por el programa de aplicación son independientes de su posición física, y es posible retirar y añadir nodos en el anillo sin problemas y sin modificar el direccionamiento. Existen más de 400 fabricantes que ofrecen equipos de campo con interfaz para Interbus. Todos los equipos compatibles que realizan este protocolo se verifican en un instituto neutral y se certifican por el InterBus-S Club.

2.4.4.6.7 Measurement Bus

Este bus de campo surge de la cooperación entre diversos fabricantes de dispositivos de medición de alta calidad, empresas del mundo de la industria de automatización y la Physikalisch-Technische Bundesanstalt (autoridad federal alemana para tests, calibración y certificación). Fue normalizado en septiembre de 1989 como DIN 66348. La Association of Measurement Bus Users (ADM e, V.) agrupa a todos aquellos interesados en este bus de campo dando publicidad de este protocolo, de productos y suministradores» de ferias y muestras y de toda la información técnica referente a este bus. Es un bus de bajo coste que se aplica principalmente en test industriales y técnicas de medición. También se aplica en procesos de control de calidad y de monitorización y adquisición de datos de producción. Debido a sus características técnicas, este bus es muy utilizado en sistemas y equipos que están sujetos a continuas calibraciones, como por ejemplo en estaciones de servicio, dispositivos de medición de flujos en general y equipamiento de pesaje.

El Measurement Bus es un bus de 4 hilos que tiene separadas las líneas de recepción y transmisión, pudiendo recibir y transmitir datos al mismo tiempo (full dúplex). Presenta una transmisión de datos fiable y tolerante a fallos y la red es de fácil instalación, mantenimiento sencillo y gran flexibilidad. Es un bus de acceso centralizado que tiene una estación activa (principal) y que actúa también como punto de acceso al servicio para redes de más alto nivel (MAP, MMI).

2.4.4.6.8 CAN (Controller Area Network)

Es un protocolo serie de comunicaciones que soporta control distribuido en tiempo real con un gran nivel de seguridad. Ha sido desarrollado por la firma Bosch en 1985 y soportado desde 1992 por la organización CiA (CAN in Automation). Se recoge como norma en la ISO 11898/11519 y su principal aplicación reside en la industria del automóvil, donde las unidades de control, sensores, sistemas antideslizamiento y otros sistemas a bordo de los coches se conectan usando un bus CAN a velocidades de hasta 1Mbps. También se usa en la electrónica del automóvil, como por ejemplo en los grupos de luces y en las ventanas eléctricas, evitando así un aumento innecesario del cableado.

2.4.4.6.9 SDS (Smart Distributed System)

Este protocolo ha sido desarrollado por Honeywell y normalizado como ISO 11989. Es un sistema de bus basado en CAN y adecuado para comunicar sensores y actuadores inteligentes. Usa un cable de 4 hilos para reunir hasta 64 dispositivos direccionables individualmente con un máximo de 126 direcciones. Se recomiendan 4 posibles velocidades de transmisión, que son 125 Kbps, 250 Kbps, 500 Kbps y 1 Mbps (la velocidad del bus la fija la estación activa con la menor dirección). La distancia que

se puede alcanzar depende de la velocidad de la red, siendo posible llegar hasta 500 metros.

2.4.4.6.10 DeviceNet

El protocolo DeviceNet es un estándar abierto que permite una solución de red económica al nivel de dispositivo. Está basado en la experimentada tecnología de red CAN, que tiene el soporte de los líderes de fabricación de sensores, actuadores y sistemas de control en la industria del automóvil. Fue desarrollado originalmente por Allen-Bradley, hoy en día Rockwell Automation, y actualmente es soportado por la organización ODVA (Open DeviceNet Vendor Association).



El protocolo DeviceNet contempla comunicaciones entre estaciones con la misma funcionalidad (peer-to-peer) y comunicaciones activa-pasiva (máster-esclavo). Puede funcionar a tres velocidades distintas: 125 Kbps (longitud máxima 500 metros), 250 Kbps (longitud máxima 250 metros), y 500 Kbps (longitud máxima 100 metros). La topología es la de un bus lineal, transportando por mismo cable de red los datos y la alimentación de los dispositivos. La longitud máxima de datos en una trama es de 8 octetos. El máximo número de nodos permitidos es de 64.

2.4.4.6.11 Sensoplex

Ofrece una interconexión directa de dispositivos de campo a un controlador vía un cable coaxial. Está pensado sobre todo para la instalación de sensores y actuadores en áreas de atmósfera explosiva y potencialmente peligrosas, como puede ser el motor de un automóvil. Los sensores/actuadores dañados y sus cables pueden ser reemplazados sin afectar al resto del sistema.

Este bus se basa en un acceso determinístico principal-subordinado y puede leer, escribir y monitorizar hasta 180 puntos de entradas y salidas en menos de 5 milisegundos. Para codificar los datos utiliza una modulación por desplazamiento de frecuencia (FSK: Frequency Shift Keying) como en el caso del bus HART. El número de estaciones que se pueden conectar a una red va desde 32 hasta 120, dependiendo del tipo de controlador utilizado en los equipos. Los sistemas Sensoplex son ampliamente utilizados en el sector de la industria del automóvil, siendo instalados por Ford en sus plantas de Genk (Bélgica), Colonia (Alemania) y Windsor (Canadá), por General Motors en Antwerp (Bélgica) y por Chrysler en Toluca (México).

2.4.4.6.12 J1939

Es un bus de alta velocidad diseñado para comunicaciones en tiempo real entre los dispositivos electrónicos de control que están físicamente distribuidos en un vehículo. Su aplicación principal es en la industria del automóvil. Entre las funcionalidades posibles de este bus está el intercambio de información, diagnóstico de fallos y control del sistema.

El J1939 está basado en el protocolo CAN. La transmisión de mensajes es del tipo de difusión (broadcast), careciendo los datos transmitidos de dirección de destino. La velocidad de transmisión del bus J1939 es de 250 Kbps. El tiempo de transmisión de un mensaje que contiene 8 octetos de datos (formato de trama de 128 bits) es de 0'5 milisegundos. La trama más corta es de 64 bits, pudiendo enviarse mensajes cada 250 microsegundos. La topología es de un bus lineal usando cable de pares trenzados apantallados.

2.4.4.6.13 AS-I (Actuator/Sensor Interface)

Este protocolo ha sido desarrollado por la firma Siemens y define la comunicación y la gestión de ésta, entre un dispositivo de control con los sensores y actuadores correspondientes. Se basa en un bus de dos hilos sin apantallar que puede tener una longitud máxima de 100 metros y que interconecta a una estación activa (máster) y un máximo de 31 estaciones pasivas (esclavo), con un máximo de 124 actuadores/sensores binarios (máximo de 4 unidades binarias o 1 unidad digital más compleja por estación pasiva). La estación activa interroga a todas las pasivas sucesivamente y espera la respuesta. AS-I usa mensajes de longitud constante, evitando así el uso de complejos procedimientos para el control de la transmisión y cálculo de las longitudes de los mensajes y formato de los datos, consiguiendo de esta forma que una estación activa consulte a todas sus estaciones pasivas, y actualice los datos en un tiempo máximo de 5 milisegundos a una velocidad de 167 Kbps. Para conectar diferentes actuadores, sensores u otros dispositivos y elementos al bus AS-I, este protocolo define un módulo con tareas de interfaz electromecánica que permite una simple instalación y manipulación de dichos elementos. La alimentación de los nodos conectados al bus se puede realizar a través del propio bus (24 VDC y hasta 100 mA por estación pasiva con un máximo de 2 A en total).



2.4.4.6.14 NBIP (Nine-Bit Interprocessor Protocol)

Este es un protocolo de comunicaciones orientado a carácter (caracteres de 9 bits) y con una configuración principal/subordinada basada en peticiones. Fue diseñado por Intel para comunicar las familias de microcontroladores MCS-51 y MCS-96 a través de un bus serie.

El concepto básico de este protocolo se puede describir brevemente como sigue: cuando el procesador principal desea transmitir o recibir un bloque de datos hacia o desde uno de los subordinados, primero envía un carácter de control de dirección (bit 8 puesto a 1). Este carácter interrumpirá a todos los subordinados para que cada uno de ellos pueda examinar el octeto recibido (bits 0 al 7) y comprobar si esa es su dirección. El subordinado direccionado comprueba en la parte de control del octeto si el principal quiere transmitir o recibir datos y cambia su estado para permitir la comunicación indicada. El nodo que inicia la transmisión envía el mensaje en sucesivos caracteres, y cierra la transmisión con una copia del carácter de control de dirección precedida de un carácter de comprobación (checksum) para asegurarse que no hubo errores en la transmisión.

2.4.4.6.15 M3S (Múltiple Master Múltiple Slave)

M3S es un sistema de comunicaciones diseñado por compañías e instituciones europeas para acceder a los diferentes dispositivos técnicos asistenciales de personas discapacitadas. Su principal aplicación es en las sillas de ruedas para minusválidos, a las cuales se le añaden dispositivos adicionales (brazo robotizado, motores, mando de control, teclado, pantalla, etc.) que ayuden a la persona mejorando su calidad de vida. Es una arquitectura estándar basada en el protocolo CAN, iniciándose cada mensaje con un identificador de 11 bits. El método de acceso al bus es de contienda (CSMA/CD), realizando un arbitraje para el que se utilizan los bits del identificador. Para la detección de errores usa un campo de redundancia cíclica (CRC: Cyclic Redundancy Check) de 15 bits. El bus tiene 2 líneas para comunicación digital (bus CAN), 2 líneas para distribución de la alimentación a los dispositivos y 2 líneas para control de seguridad.

El sistema M3S incluye aspectos adicionales para incrementar la seguridad e integridad de la instalación, como son la llave de encendido (Key) y el DMS (Dead Man Switch).

2.4.4.6.16 Lon Works (Local Operating Network)

LonWorks es el principal bus de campo con aplicación en la domótica en el mercado USA. Fue creado por la firma Echelon pero es un bus totalmente abierto. En abril de 1998 fue reconocido por la EIA como un bus de campo de amplio uso y publicado como una nueva norma para redes de control domótico con la denominación EIA-709. La firma Echelon declara que hay cerca de 5 millones de nodos LonWorks instalados en todo el mundo.

En un principio, Echelon llegó a acuerdos con las firmas Motorola y Toshiba para el desarrollo de circuitos integrados que realicen el protocolo LonTalk en el que se basa

el bus de campo LonWorks. Actualmente existen en el mercado cuatro modelos distintos de Neuron Chip diseñados por estos fabricantes. Una vez que ha llegado a ser una norma EIA, Echelon ha publicado el protocolo para que cualquiera lo pueda desarrollar sobre un microprocesador de libre elección. En 1994 se fundó la LonMark Interoperability Association, que tiene unos 200 miembros, y entre sus tareas principales incluye la realización de pruebas de conformidad para productos LonWorks.

Mientras que las comunicaciones en algunos buses de campo sólo pueden efectuarse sobre pares trenzados (BatiBus, EIB) o sobre la red eléctrica (X10), una red LonWorks puede emplear varios medios físicos (par trenzado, fibra óptica, red eléctrica, radio, infrarrojo, cable coaxial) en una misma red, proporcionando una gran flexibilidad en su aplicación.

2.4.4.6.17 ARCNET

Es un bus doméstico desarrollado por Datapoint Corporation y normalizado como ANSI 878. La ARCNET Trade Association (ATA) es una organización sin ánimo de lucro formada por usuarios y fabricantes, con el propósito de promover el uso de esta norma y dar información sobre ella. Puede presentar una topología en bus lineal o en estrella.

2.4.4.6.18 M-Bus (Meter-Bus)

Es un estándar europeo desarrollado para el cableado en red y lectura remota de medidores y contadores existentes en un domicilio o un edificio, como por ejemplo los contadores de consumo de gas o de agua, siendo también usado por varias clases de sensores y actuadores. Este bus doméstico fue desarrollado por el profesor Dr. Horst Ziegler de la Universidad de Paderborn (Alemania) en cooperación con Texas Instruments, Deutschiand GMBH y Techem Gmbfi. La capa de enlace de datos está basada en la norma IEC 870-5 [46] y la capa de aplicación se corresponde a la norma EN 1434-3.

Como los datos de los contadores son usados para realizar las cuentas de consumo de los usuarios, este bus tiene que tener un alto grado de integridad en la transmisión. El bus tiene que ser insensible a interferencias externas, debiendo estar los dispositivos aislados eléctricamente. Los datos de los contadores y medidores se leen electrónicamente a través de un único bus serie (2 hilos) que conecta a todos los contadores y medidores existentes en el edificio, a un único controlador principal que almacena los datos. El método de acceso al medio es centralizado por la estación principal, la transmisión es asíncrona alternada (half dúplex) y a velocidades entre 300 y 9600 bps. Todos los dispositivos se direccionan individualmente. La lectura de datos es rápida y sin errores, teniendo los datos un formato que hacen que su procesamiento sea sencillo.

2.5 El bus PROFIBUS

2.5.1 Introducción

PROFIBUS (PROcess FIeld BUS) es el bus líder en Europa. Se trata de un bus de campo *abierto*, que puede implementarse en diversas áreas como pueden ser fabricación, proceso y automatización de edificios. Se encuentra *estandarizado* en la norma DIN 19245 (a principios de 1991) para más tarde, en 1996, ser incluido por el CENELEC en la norma europea EN 50170 (volumen 2), lo que le proporciona un gran nivel de confianza de cara al usuario, así como la posibilidad de comunicación entre equipos de diferentes fabricantes sin necesidad de ajustes especiales de interface.

La familia PROFIBUS está formada por tres versiones o protocolos de comunicación, a saber:

- PROFIBUS-DP (*Periferia Descentralizada*, DIN E 19245, parte 3), perfil de protocolo para el acoplamiento de la periferia descentralizada, con rápidos tiempos de reacción.
- PROFIBUS-PA (*Process Automation*) es la ampliación de PROFIBUS-DP compatible en comunicación con una tecnología que permite aplicaciones en áreas con riesgo de explosión.
- PROFIBUS-FMS (*Field Message Specification*, DIN 19245, tomo 2) es aplicable para la comunicación de autómatas en pequeñas células y para la comunicación con dispositivos de campo con interface FMS. En esta versión, la funcionalidad es más importante que conseguir un tiempo de reacción pequeño.

En la Tabla 2.1. se observan las características más importantes para cada versión.

	PROFIBUS-FMS	PROFIBUS-DP	PROFIBUS-PA
Aplicación	Nivel de campo y proceso	Nivel de E/S	Nivel de E/S
Estándar	EN 50 170/IEC 61158	EN 50 170/IEC 61158	IEC 1158-2
Dispositivos conectables	PLC, PG/PC, Dispositivos de campo	PLC, PG/PC, Dispositivos de campo, accionamientos, OPs	Dispositivos de campo para áreas con riesgo de explosión
Tiempo respuesta	< 60 ms	1-5 ms	< 60 ms
Tamaño red	<= 150 Km	<= 150 Km	Máx. 1.9 Km
Velocidad	9.6 Kbit/s -12Mbit/s	9.6 Kbit/s -12Mbit/s	31.25 Kbit/s

Tabla 2.1. Diferentes versiones PROFIBUS y principales características.

En este proyecto se ha optado por una configuración PROFIBUS-DP, debido a que aporta una velocidad y un tiempo de respuesta más que suficientes para la red de comunicación.

De la tabla anterior cabe destacar que PROFIBUS es un bus que puede implementarse tanto a nivel de E/S (nivel actuador/sensor) como a nivel de campo y proceso, gracias a sus características de velocidad y tiempo de respuesta. PROFIBUS especifica las características técnicas y funcionales de un sistema de bus de campo serie al cual pueden conectarse controladores digitales descentralizados desde el nivel de E/S hasta el nivel de campo y proceso. En PROFIBUS se distingue entre dispositivos maestros y esclavos:

- **MAESTROS.**

Determinan la comunicación de datos en el bus. Un maestro puede enviar mensajes, sin necesidad de una petición o solicitud externa, cuando posee los derechos del bus. En el protocolo PROFIBUS también se les da el nombre de *estaciones activas*.

- **ESCLAVOS.**

Son elementos de periferia. Los esclavos típicos incluyen dispositivos de entrada/salida, válvulas, transmisores de medida y accionamientos. Un esclavo no tiene derecho de acceso al bus y sólo pueden acusar los mensajes recibidos o enviar mensajes al maestro cuando éste así lo requiere. A los esclavos también se les denomina *estaciones pasivas*.

La velocidad de transmisión del bus PROFIBUS puede oscilar desde los 9.6 Kbit/s hasta alcanzar un máximo de 12 Mbit/s. La longitud máxima de la red también es variable, dependiendo del medio físico utilizado (eléctrico o fibra óptica). El número máximo de estaciones es de 127 (direcciones de la 0 a 126).

2.5.2 El modelo de referencia OSI y PROFIBUS.

La norma PROFIBUS está orientada al modelo de referencia OSI (Open System Interconnection) acorde con el estándar internacional ISO 7498. En la Fig. 2.5. se observa la arquitectura del protocolo PROFIBUS, así como las capas que implementa cada versión.

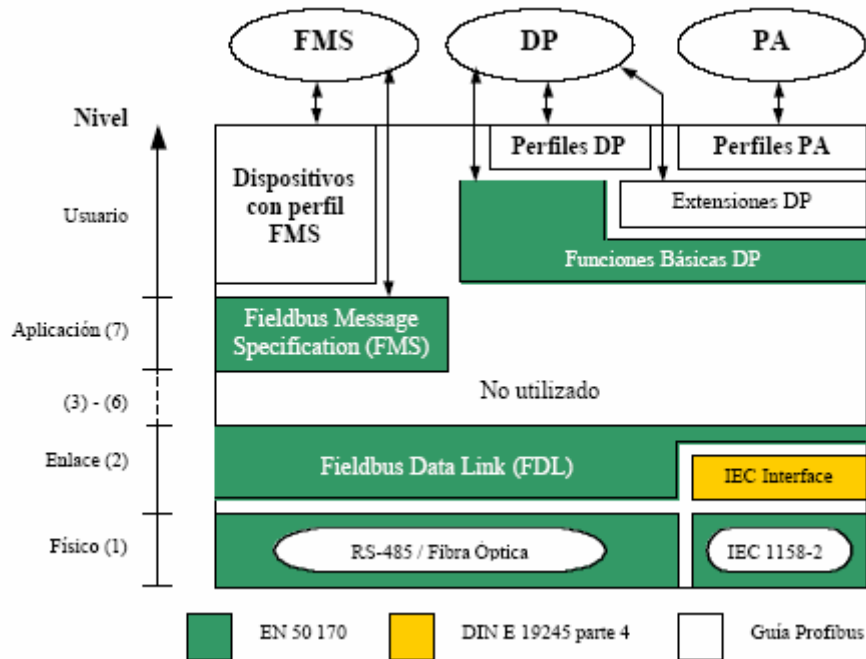


Fig. 2.5. Arquitectura del protocolo PROFIBUS.

- **Capa física:** se detallan las especificaciones eléctricas y mecánicas para garantizar la transmisión de los datos entre equipos. Incluye las características de la tecnología de transmisión del bus, como pueden ser voltajes empleados, forma de codificación, sincronización, tipo de medio físico, tipos de conectores, etc.
- **Capa de enlace:** tiene por misión garantizar la fiabilidad de la transmisión entre equipos. Realiza funciones de detección y corrección de errores así como de regulación y control de flujo de datos.
- **Capa de aplicación:** proporciona servicios y funciones para que los procesos de aplicación tengan acceso al entorno de comunicación y se puedan comunicar con otras aplicaciones.
- **Interface de usuario:** se especifican las funciones de aplicación que están disponibles para el usuario así como el comportamiento del sistema y de los distintos tipos de dispositivos PROFIBUS.

2.5.3 Tecnología de transmisión

Dada la imposibilidad de satisfacer todos los requisitos de las diferentes versiones de PROFIBUS (Tabla 2.1.), se proporcionan tres variantes:

- Transmisión IEC 1158-2 para PA.
- Fibra óptica (FO).
- Transmisión RS 485 para DP y FMS.

2.5.4 Método de acceso

El acceso a la red en PROFIBUS se controla en base a la combinación de dos métodos de acceso, el principio de “**paso de testigo**” (*Token Bus*), definido en la norma EN 50170, para comunicación entre estaciones maestras, y el principio de “**maestro-esclavo**” (*Master-Slave*) para sistemas centralizados. Con este método de acceso se satisfacen dos requisitos imprescindibles en un bus de campo:

- Durante la comunicación entre maestros debe asegurarse que cada una de las estaciones tiene tiempo suficiente para ejecutar sus tareas de comunicación en un intervalo de tiempo definido con precisión.
- Transmisión de datos en tiempo real, cíclica, tan rápida y simple como sea posible, para la comunicación entre maestros y esclavos.

El método de acceso es independiente del medio de transmisión. Todas las estaciones maestro disponen de control de acceso al bus, mientras que las estaciones esclavo son neutras en este aspecto. La estación maestra, activa en un momento determinado, tendrá posibilidad de comunicarse con una determinada estación esclava. La Fig. 2.6. muestra el mencionado *método híbrido*, con usuarios de bus activos y pasivos.

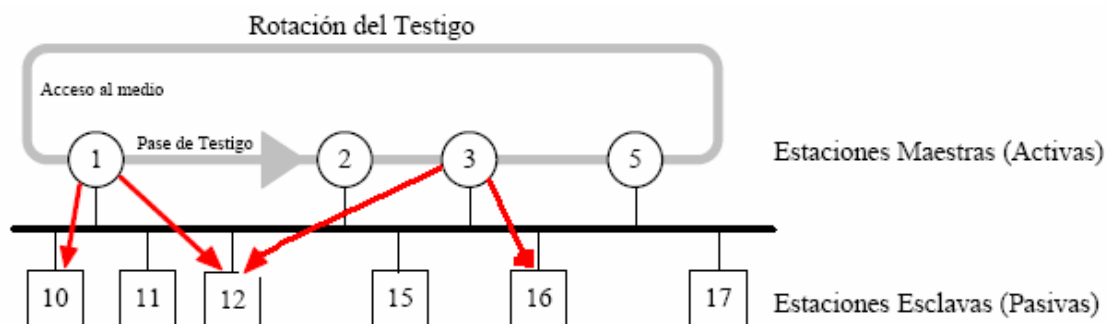


Fig. 2.6. Método de acceso al bus.

Todas las estaciones activas constituyen, en un orden definido por las “direcciones PROFIBUS”, el *anillo lógico con paso de testigo*. Cada una de ellas conoce a los restantes usuarios activos, así como su orden en el anillo lógico, que es independiente de la disposición topológica de las estaciones activas en la red.

El testigo o *token* se pasa de una estación activa a la siguiente en orden numérico ascendente de dirección de estación. La estación con dirección más alta pasará el testigo a aquella con menor dirección, cerrándose de este modo el anillo lógico. Cada una de las estaciones conoce la *estación previa* (PS, Previous Station, de la que recibe el testigo), la *próxima estación* (NS, Next Station, a la que pasará el testigo) y *su propia dirección* (TS, This Station). Estas direcciones están almacenadas en una *Lista de Estaciones Activas* (LAS, List of Active Stations), lista que se genera en la fase de escucha después del arranque de la CPU y que se actualiza dinámicamente. En la Fig. 2.7. puede verse como se llevaría a cabo el paso de testigo en una red con las características que se muestran.

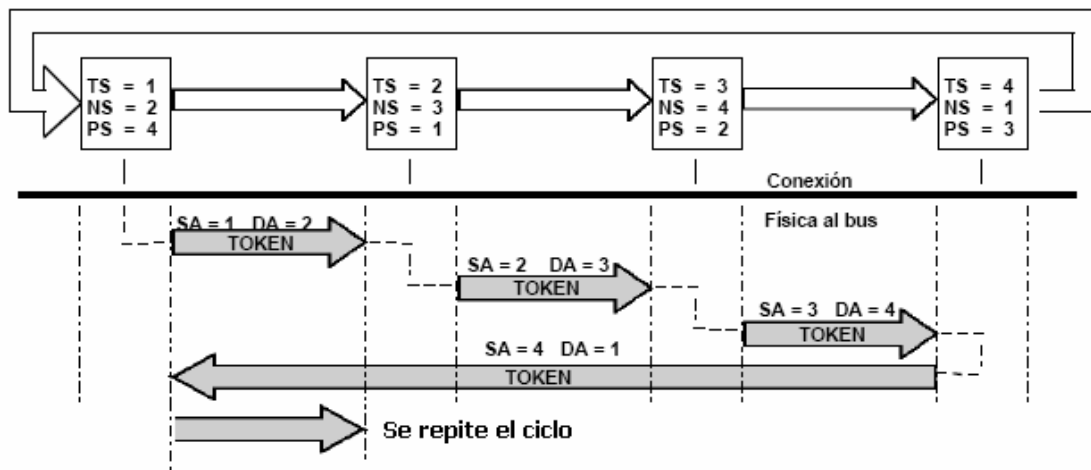


Fig. 2.7. Ejemplo de ruta de testigo.

Recepción del testigo. Cuando una estación TS recibe una *trama de testigo* (token frame) puede ejecutar ciclos de mensajes. Si el transmisor del testigo no está registrado como la estación PS, el receptor inicialmente considera que se ha producido un error y no acepta el testigo. No lo acepta hasta que la misma estación PS lo intente de nuevo; es entonces cuando el receptor acepta y asume que se ha producido un cambio en el anillo lógico. El tiempo del que la estación TS dispone para transmitir viene definido por el llamado *tiempo de retención del testigo*. Una vez expirado, la estación solo puede emitir un mensaje de prioridad alta.

Emisión del testigo. Cuando una estación TS ha completado sus ciclos de mensaje, pasa el testigo a la próxima estación (NS) mediante el envío de la trama de testigo (token frame). Si el emisor del testigo recibe una trama de confirmación válida, considera que su NS ha cogido el testigo y está ejecutando sus ciclos de mensaje. Si por el contrario recibe una trama defectuosa, interpreta que otra estación está enviando. En ambos casos, detiene la comprobación y pasa a estado "Active-Idle" (activa en proceso). Si el emisor del testigo no detecta ninguna actividad en el bus en el *slot time*, envía la trama de testigo de nuevo y espera otro *slot time*. Si vuelve a no detectar actividad,

envía una última vez la trama de testigo a su NS y vuelve a esperar otro *slot time*. Si tras este tercer intento todavía no hay actividad en el bus, el emisor intentará pasar el testigo a la estación posterior a su NS. Se repite este proceso hasta que encuentra una estación NS. Si no tiene éxito, interpretará que es la única estación que existe en el anillo lógico y mantendrá el testigo para sí misma si no se solicitan ciclos de mensaje. Todo este funcionamiento se resume en la Fig. 2.8.

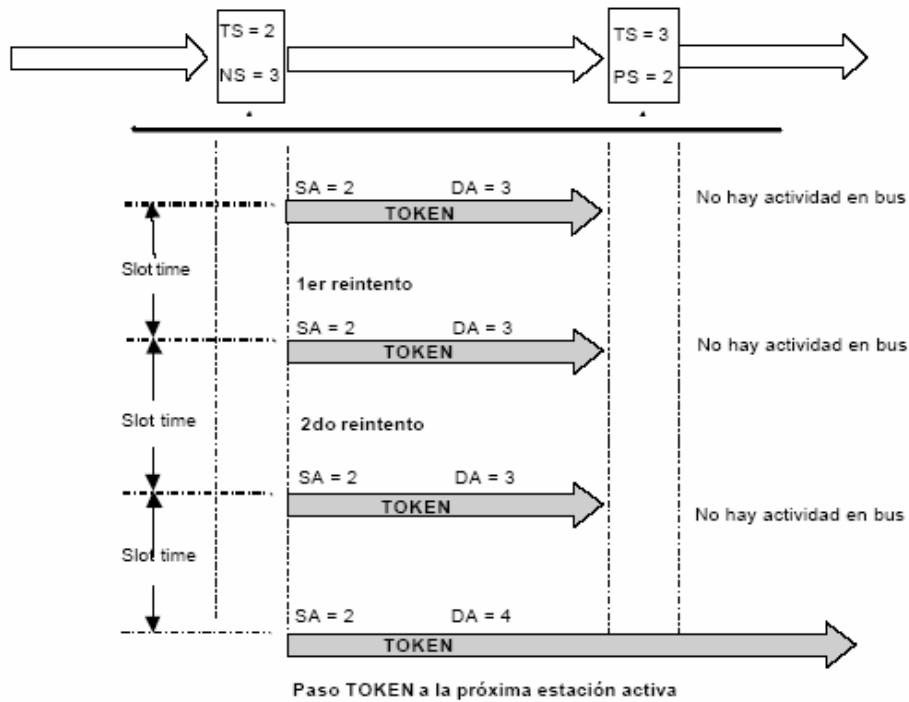


Fig. 2.8. Proceso de emisión del testigo.

Añadir y eliminar estaciones (Actualización de la LAS). Todas las estaciones activas que constituyen el anillo lógico son responsables de añadir nuevas estaciones y eliminar las viejas. Cada estación es responsable del área comprendida entre su propia dirección (TS) y la de su próxima estación (NS). Esta área de direcciones se denomina lista de estación GAP (GAPL). Cada estación activa del anillo comprueba cíclicamente si se han producido cambios en estaciones activas y pasivas en su área de direcciones (todas las direcciones GAP). Cuando una estación posee el testigo comienza a actualizar el GAP inmediatamente después de haber procesado todos los ciclos de mensaje correspondientes. Si no dispone de tiempo (tiempo de retención del testigo), esta acción tendrá lugar la próxima vez que reciba el testigo. En la Fig. 2.9. se ilustra este proceso.

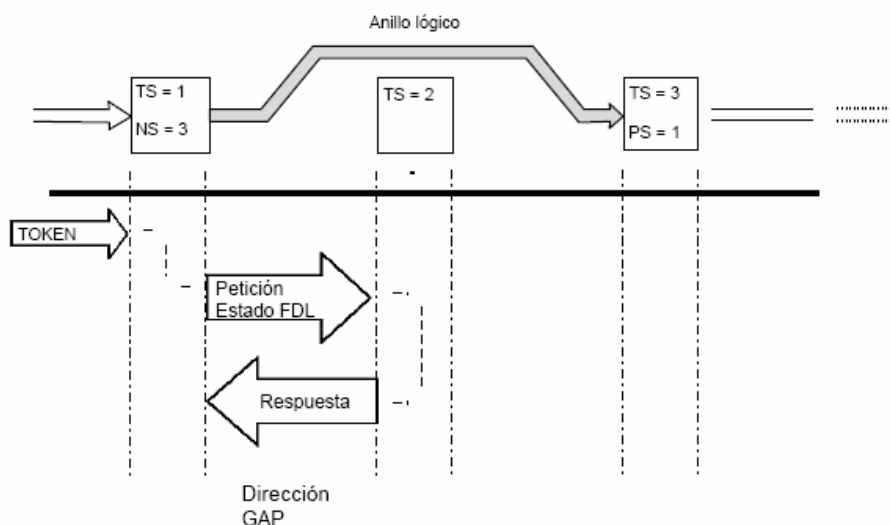


Fig. 2.9. Proceso de actualización de la LAS.

Las direcciones GAP se comprueban en orden numéricamente ascendente. Si una estación responde con un acuse positivo y el estado “no preparada” o “estación pasiva”, se introduce en la GAPL como tal, y se interroga a la próxima estación. Si una estación responde con “preparada para el anillo”, el poseedor del testigo modifica su GAPL y pasa el testigo a la nueva próxima estación (NS). La estación añadida al anillo ya habrá creado su LAS y determinará su propio área GAP. La estación que anteriormente poseía el testigo también actualizará su Lista de Estaciones Activas (LAS) y su área GAP.

Parámetros del bus. En este apartado se definirán los principales parámetros de tiempo para el bus PROFIBUS. Para ello, es necesario mencionar que una estación activa puede ejecutar dos tipos de mensajes: de prioridad alta y de prioridad baja.

- **Tiempo de Rotación Real (T_{RR}).** Tan pronto como una estación activa recibe el testigo comienza la medición del tiempo de rotación del testigo. Cuando la misma estación acepta el testigo de nuevo se detiene la medición del tiempo para el ciclo ejecutado, y el resultado es el tiempo de rotación real.
- **Tiempo Teórico de Rotación de Testigo (T_{TR}).** Es el tiempo máximo disponible para una rotación del testigo. Tras este tiempo, todas las estaciones han recibido una vez el derecho de emisión.
- **Tiempo de Retención de Testigo (T_{TH}).** Es el tiempo que determina si una estación activa puede o no realizar ciclos de mensaje. Se define como la diferencia entre T_{TR} y T_{RR} .
- **Tiempo de Respuesta del Sistema (T_{RS}).** Intervalo máximo entre dos ciclos de mensaje de prioridad alta sucesivos en una estación activa.

- Tiempo de Rotación de Testigo Teórico Mínimo (T_{TRmin}). Se calcula a partir de la siguiente fórmula:

$$T_{TRmin} = na * (T_{TC} + alta T_{MC}) + k * baja T_{MC} + mt * RET T_{MC}$$

na.	Número de estaciones activas.
k.	Nº probable de ciclos de mensaje de prioridad baja por rotación.
T_{TC} .	Tiempo de ciclo de testigo.
T_{MC} .	Tiempo de ciclo de mensaje, función de la longitud de la trama.
mt.	Nº de ciclos de reintento de mensaje por rotación.
RET T_{MC} .	Tiempo de un ciclo de reintento de mensaje.

- Tiempo de espera a la recepción (Slot Time). Determina el tiempo máximo que espera el emisor a recibir una respuesta de su interlocutor.
- Factor de actualización GAP. Determina después de cuántas rotaciones del testigo se aceptará una nueva estación en el anillo lógico.

2.5.5 El protocolo PROFIBUS-DP

PROFIBUS-DP se dispuso en la parte 3 de la norma DIN 19245 y se integró en la norma europea de bus de campo EN 50170. Se ajusta a los requisitos de intercambio de datos más rápido y eficiente, entre los elementos de automatización y los elementos distribuidos, tales como módulos de entradas/salidas, actuadores, o incluso PLCs.

2.5.5.1 Tipos de dispositivos

En PROFIBUS-DP se presentan tres tipos de dispositivos:

- **Maestro DP de clase 1 (DPM1)**.
Se trata de un controlador central que intercambia información con los equipos distribuidos (esclavos DP) en un ciclo de mensajes específico. Incluye funciones maestro-esclavo como por ejemplo recopilación de información de diagnóstico por parte de los esclavos DP, operación cíclica de datos de usuario, parametrización y configuración de esclavos DP, etc. Estos dispositivos pueden ser PLCs, controles numéricos computerizados (CNC) o control de robots (RC).
- **Maestro DP clase 2 (DPM2)**
Los elementos de este tipo son los dispositivos de programación, configuración y diagnóstico. Son parametrizados en la puesta en marcha para especificar la configuración del sistema DP, como el número de dispositivos, direcciones de las estaciones del bus, etc. Las funciones de este tipo de maestro incluyen carga y descarga de programas de usuario, activación y desactivación de estaciones, lectura/escritura de registros, etc.

- **Esclavo DP.**

Cada elemento de la periferia se identifica como esclavo DP cuando los datos de entrada son leídos y los datos de salida son suministrados a la periferia. Se trata de dispositivos de entrada/salida, dispositivos actuadores, o PLCs configurados como esclavos.

2.5.5.2 Configuración del sistema

PROFIBUS-DP soporta sistemas mono y multi-maestro. Esto permite un alto nivel de flexibilidad en la configuración del sistema de bus.

- En un sistema mono-maestro sólo hay un maestro activo en el bus. El maestro DP es el componente central de control, a través del cual se acoplan los esclavos DP. Representa un proceso limpio de acceso maestro-esclavo.
- En el caso de haber más de un maestro en el bus, se trata de un sistema multi-maestro. Se puede construir subsistemas de bus independientes, asignados a cada maestro y sus esclavos asociados. Las entradas/salidas de los esclavos pueden ser leídas por todos los maestros. Los maestros pueden intercambiar datos entre sí.

2.5.5.3 Capa física. Transmisión RS 485

Para la transmisión en PROFIBUS-DP se dispone tanto de la tecnología RS 485 como de la fibra óptica. En este proyecto se ha utilizado transmisión RS 485, por motivos de disponibilidad.

El método de transmisión RS 485 responde a la transmisión simétrica de datos según el EIA Standard RS 485, y también se conoce con el nombre de H2. Este método de transmisión está prescrito con carácter *obligatorio* en la norma PROFIBUS EN 50170 para la transmisión de datos por líneas bifilares. Sus principales características son:

- Medio físico: cable bifilar de cobre trenzado y apantallado.
- Velocidad de transmisión: desde 9.6 Kbit/s hasta 12 Mbit/s.
- Resistencia de terminación obligatoria en extremos del bus.
- Longitud de cable máxima en función de la velocidad de transmisión.

Velocidad (kbit/s)	9.6	19.2	93.75	187.5	500	1500	12000
Distancia/segmento (m)	1200	1200	1200	1000	400	200	100

- Máximo número de estaciones: 32 estaciones (maestros o esclavos) por segmento. Posible enlace de segmentos por medio de repetidores RS 485, con un máximo de 127 estaciones y un máximo de 9 repetidores.
- Estructuras de red en línea o en árbol.
- Tipo de conexión: conectores Sub-D de 9 polos.
- El Standard RS 485 utiliza Codificación NRZ (Non-Return-to-Zero, 0 lógico no produce cambio al principio del bit, 1 lógico produce cambio de nivel a principio del bit). Se trata de una *transmisión asíncrona, orientada a caracteres* (11 bits).

2.5.5.4 Capa de enlace. FDL

PROFIBUS-DP emplea un protocolo de acceso al bus que se implementa en el nivel de enlace datos, denominado Fieldbus Data Link, FDL. El método de acceso al medio (MAC) ya fue descrito en el Apartado 2.5.4.

En lo que respecta a la seguridad de los datos, todos los telegramas tienen una Distancia de Hamming, $H_d = 4$. Esto se logra gracias a unos delimitadores especiales inicial y final y a un bit de paridad por cada octeto, como se define en el estándar internacional IEC 870-5-1.

El nivel FDL opera en un modo sin enlaces. Además de la transmisión de datos punto a punto, proporciona comunicaciones tipo *Broadcast* (mensaje para todas las estaciones del bus) y *Multicast* (mensaje para un grupo determinado de estaciones del bus).

Todas las tramas elaboradas por FDL están compuestas por un cierto número de caracteres UART (Universal Asynchronous Receiver/Transmitter). Estos caracteres se emplean en transmisiones serie asíncronas, y cada uno de ellos consta de 11 bits distribuidos del siguiente modo:

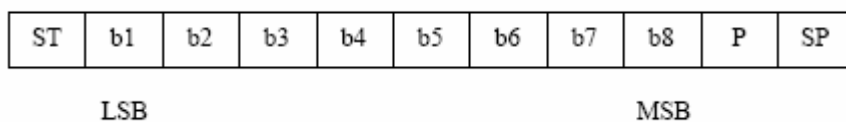


Fig. 2.10. Estructura de un carácter UART.

Desglosando esta trama nos encontramos:

- **ST** (*Start Bit*): bit de inicio, siempre toma el valor binario “0”.
- **B1 a B8**: bits de información. Pueden tomar valor “0” o “1”.

- **P**: bit de paridad, en este caso par.
- **SP** (*Stop Bit*): bit de parada, siempre tiene el valor “1”.

En PROFIBUS DP, los principales tipos de tramas de mensaje están referidos a: asignación de parámetros, configuración, diagnóstico, comandos de control y datos de usuario. Existe una trama de cabecera, la cual será configurada de una forma u otra por el protocolo dependiendo del tipo de mensaje al que corresponda. La estructura de la trama cabecera es la que se representa en la Fig. 2.11.

SD	LE	LEr	SD	DA	SA	FC	DSAP	SSAP	FCS	ED
<i>SD (Start Delimiter)</i>										
<i>LE (frame LEnght)</i>										
<i>LEr (repetition of the frame LEnght)</i>										
<i>DA (Destination Address)</i>										
<i>SA (Source Address)</i>										
<i>FC (Function Code)</i>										
<i>DSAP (Destination Service Access Point)</i>										
<i>SSAP (Source Service Access Point)</i>										
<i>FCS (Frame Check Sequence)</i>										
<i>ED (End Delimiter)</i>										

<i>SD (Start Delimiter)</i>	⇔ Delimitador inicial.
<i>LE (frame LEnght)</i>	⇔ Longitud de la trama.
<i>LEr (repetition of the frame LEnght)</i>	⇔ Repetición de la longitud de la trama.
<i>DA (Destination Address)</i>	⇔ Dirección de destino.
<i>SA (Source Address)</i>	⇔ Dirección de origen (fuente).
<i>FC (Function Code)</i>	⇔ Código de función.
<i>DSAP (Destination Service Access Point)</i>	⇔ Punto de acceso al servicio destino.
<i>SSAP (Source Service Access Point)</i>	⇔ Punto de acceso al servicio origen.
<i>FCS (Frame Check Sequence)</i>	⇔ Secuencia de comprobación de trama.
<i>ED (End Delimiter)</i>	⇔ Delimitador final.

Fig. 2.11. Trama cabecera de PROFIBUS DP. Significado de las abreviaturas.

Llegados a este punto, hay que advertir que pueden explicarse de forma precisa los distintos tipos de trama posibles en el nivel de enlace de datos de PROFIBUS-DP pero esta información no resulta imprescindible a la hora de configurar una red de este tipo, por lo que se ha optado por no detallarla en este proyecto. Esta información se encuentra recogida en la *Referencia Bibliográfica* [5].

2.5.5.5 Nivel de usuario. Funciones DP

El protocolo de comunicación PROFIBUS-DP aporta múltiples funciones de comunicación para realizar diversas tareas en el bus, tales como escribir/leer datos, diagnosticar esclavos, sincronizar grupos de esclavos, y un largo etcétera. Estas funciones están implementadas en el software de programación STEP 7, a través del cual se configura la comunicación PROFIBUS. A grandes rasgos, son funciones tipo “caja negra”, es decir, en función de unas entradas parametrizadas por el usuario y/o obtenidas del proceso, proporcionan unas salidas acordes a la función. En este proyecto no ha sido necesaria la utilización de este tipo de funciones.

Las funciones de comunicación DP se encuentran recogidas en la **Referencia Bibliográfica [4]**.

2.6 La interface MPI

Este sistema de bus se desarrolló como una interface de programa para los autómatas programables de las series S7. La MPI (Multi Point Interface) sirve además como medio de comunicación entre los componentes que se utilizan como “Interface Hombre-Maquina” y como medio homogéneo de comunicación entre elementos de automatización. Esta interface de comunicación se encuentra disponible en todos los productos de la serie S7 de Siemens.

2.6.1 Datos técnicos

La interface MPI puede utilizarse para uniones simples en redes y permite las siguientes formas de comunicación:

- Con el servicio de Datos Globales, las CPUs interconectadas en red y de las series S7-300 y S7-400 pueden intercambiar datos cíclicamente unas con otras.
- La Programación y Diagnóstico de Errores puede ejecutarse vía MPI desde otros elementos de programación/PCs a todos los PLCs conectados a la red.
- La conexión desde los paneles de operador/estaciones de operador al PLC es muy simple con MPI. Los servicios de comunicación son soportados por defecto y no son necesarios bloques de función especiales.

Los principales datos técnicos para la interface de comunicación MPI se resumen en los siguientes puntos:

- Máx. de 32 participantes MPI.
- Máx. de 8 conexiones para comunicación básica con otros PLCs.
- Máx. de 4 conexiones para comunicación estática con PG/PC, sistemas HMI.
- Velocidad de transmisión de datos desde 187.5 Kbit/s hasta 12 Mbit/s.
- Posibilidades flexibles de configuración en el bus o estructura árbol (con repetidores).
- Máx. longitud del cable: 10 Km.
- Interface: RS 485.

2.6.2 Tecnología de transmisión

La red MPI utiliza un cable bifilar y apantallado. La longitud máxima del cable no podrá superar los 50 m, a no ser que se utilicen repetidores. La distancia máxima entre dos repetidores es de 1000 m, siempre y cuando no se conecte otro nodo entre esos dos repetidores. Se pueden colocar hasta 10 repetidores.

Los componentes que intervienen en una configuración MPI, esto es, conectores, cable y repetidores RS 485, son los mismos que los utilizados en una red PROFIBUS.

2.7 La red de comunicación GSM

En el presente apartado se hará una rápida introducción al estándar de comunicación móvil GSM, destacando cómo se lleva a cabo el proceso de intercambio de datos mediante mensajes cortos o SMS (*Short Message Service*).

2.7.1 Historia

El estándar de comunicación GSM (*Global System for Mobile*) surge en Europa a principios de los '80, ante dos claras necesidades. En primer lugar, la demanda potencial para los servicios móviles era mayor que la capacidad de las redes analógicas existentes. En segundo lugar, los diferentes sistemas de aquel momento no ofrecían compatibilidad entre ellos, es decir, un terminal diseñado para una determinada red no podía acceder a una red diferente a la suya propia. Todas estas circunstancias apuntaban hacia el diseño de un sistema nuevo, desarrollado en común entre varios países.

El mundo de las telecomunicaciones en Europa siempre estuvo dominado por la CEPT (*Conférence Européenne des Postes et Télécommunications*), un foro de estandarización que, en los primeros '80, incluía las Administraciones europeas de Correos y Telecomunicaciones de más de 20 países. En el año 1982 surge en la CEPT un nuevo organismo de estandarización: GSM (*Groupe Spécial Mobile*). Su misión era la de especificar un sistema único de telecomunicaciones para Europa, en 900 MHz. La elaboración del estándar GSM llevó casi toda una década, que se resume en la Tabla 2.2.

Fecha	Logro
1982	Se crea el "Groupe Spécial Mobile" en la CEPT.
1986	Se establece un núcleo permanente.
1987	Se seleccionan las técnicas básicas de transmisión por radio, basadas en la evaluación de prototipos llevadas a cabo a lo largo de 1986.
1989	El GSM se convierte en un comité técnico de ETSI.
1990	Se congelan las especificaciones técnicas fase 1 del GSM900 escritas entre 1987 y 1990.
1991	Funcionan los primeros sistemas (exposición Telecom'91) Se congelan las especificaciones DCS1800.
1992	Los principales operadores GSM900 europeos inician la operación comercial del servicio.

Tabla 2.2. Principales hitos en la elaboración del estándar GSM.

2.7.2 Arquitectura y Funcionamiento

El sistema consiste en una red de radio-células contiguas para cubrir una determinada área de servicio. Cada célula tiene una BTS (*Base Transceiver Station*) que opera con un conjunto de canales diferentes de los utilizados por las células adyacentes.

Un determinado conjunto de BTSs es controlado por una BSC (*Base Station Controller*). Un grupo de BSCs es a su vez controlado por una MSC (*Mobile Switching Centre*) que enruta llamadas hacia y desde redes externas (*R.T.B., R.D.S.I., etc.*) públicas o privadas. La arquitectura descrita queda representada en la Fig. 2.12.

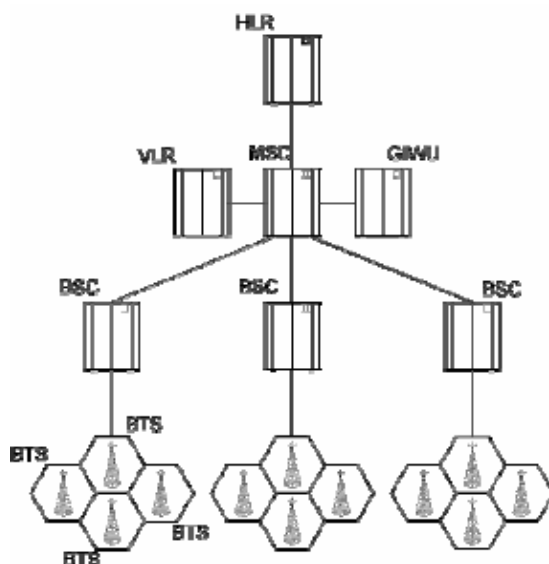


Fig. 2.12. Arquitectura de una red GSM.

- *Base Transceiver Station (BTS)*
La función principal de una BTS es proporcionar un número de canales radio a la zona a la que da servicio. Se trata de una antena omnidireccional o sectorial.
- *Base Station Controller (BSC)*.
La función primaria de una BSC es el mantenimiento de la llamada, así como la adaptación de la velocidad del enlace radio al estándar de 64 Kbit/s, utilizado por la red.
- *Mobile Switching Centre (MSC)*
La MSC es el centro de control de llamadas, responsable del establecimiento, enrutamiento y terminación de cualquier llamada, control de los servicios suplementarios, recogida de información para tarificación, etc.
- *Home Location Register (HLR)*
El HLR contiene información de estado de cada usuario asignado al mismo.
- *Visitor Location Register (VLR)*
El VLR contiene información de estado de todos los usuarios que en un momento dado están registrados dentro de su zona de influencia. Esta información ha sido requerida y obtenida a partir de los datos contenidos en el HLR del que depende el usuario.

2.7.3 Envío/Recepción de SMS

Dado que en el presente proyecto se ha implementado una aplicación de control y monitorización mediante envío/recepción de SMS, en este apartado se expondrá cómo en una red GSM se envían/reciben mensajes cortos. En la Fig. 2.13. se observa la ruta a seguir por un mensaje SMS.

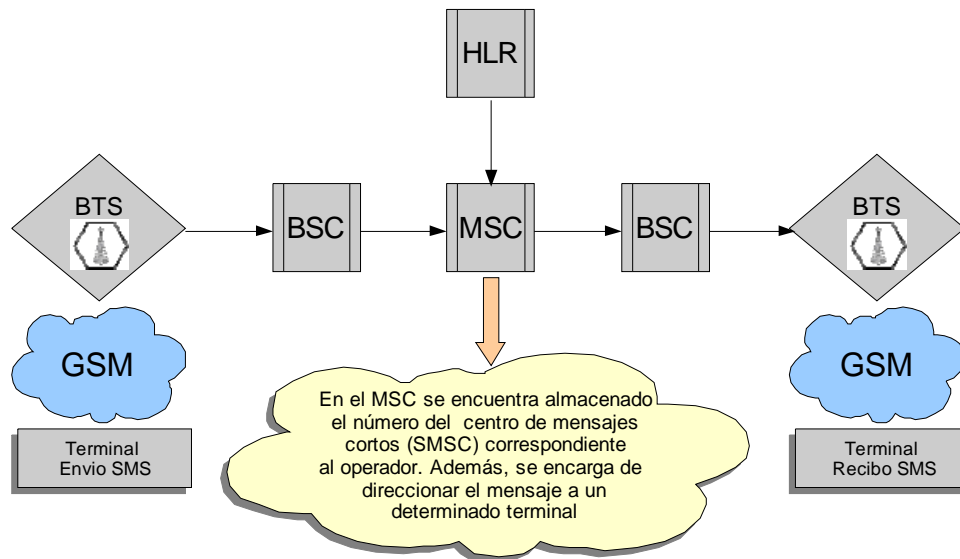


Fig. 2.13. Ruta a seguir por un mensaje SMS.

Supongamos que uno de los terminales envía un mensaje SMS. Dicho Terminal se encuentra conectado a través de la red GSM a una estación BTS, que a su vez está conectada con una estación BSC. El SMS llega así a la estación MSC. Esta estación, de acuerdo a una base de datos que se actualiza dinámicamente, redirige el mensaje a su destinatario, a través de nuevo de una estación BSC y de una estación BTS. Si el Terminal al que va dirigido el mensaje no se encuentra disponible en ese momento, éste queda almacenado en la estación HLR, enviándose cuando el Terminal entra en servicio. Si se supera el tiempo límite de espera o también llamado “tiempo de validez”, el mensaje es desechado, por lo que no se produce su envío.

Otra característica a destacar en este sistema de comunicación es que no existe una secuencia de mensajes, o lo que es lo mismo, el último mensaje enviado puede ser el primero recibido.

Capítulo 3

Descripción de los equipos utilizados

3.1 Introducción

En este capítulo se llevará a cabo una descripción a nivel técnico de los diferentes dispositivos que han sido utilizados en este proyecto. Con el objetivo de no engrosar de forma innecesaria el capítulo, se incluirán solamente datos técnicos relevantes para el montaje de la red de comunicación. El resto de características se recogen en el *Anexo A “Datos técnicos de los equipos”*.

Primeramente se realiza una descripción de los autómatas programables, seguida de los diferentes módulos que han sido necesarios, para terminar hablando sobre los elementos de comunicación utilizados.

3.2 Autómatas programables serie S7

La familia de autómatas programables serie S7 pertenece a la marca alemana SIEMENS, y abarcan desde pequeños controladores lógicos para soluciones de control simples hasta autómatas de gama alta para soluciones de sistema en la industria manufacturera y de procesos. Comprenden los llamados S7-200, S7-300 y S7-400.

3.2.1 Serie S7-200. CPU 224 AC/DC/Relé

Se han utilizado dos autómatas programables de la serie S7-200, el modelo CPU 224. Los autómatas programables pertenecientes a la serie S7-200 son PLCs de gama baja (de ahí que reciban el nombre de Micro-PLCs). Gracias a su diseño compacto, su capacidad de ampliación, su bajo coste y su amplio juego de operaciones están especialmente indicados para solucionar tareas de automatización sencillas. En la Fig. 3.1. se muestra una CPU de la serie S7-200.

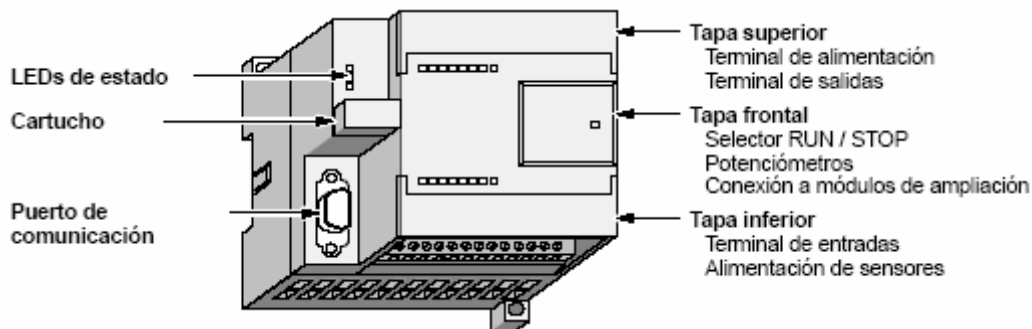


Fig. 3.1. Vista general de una CPU224.

La CPU 224 AC/DC/Relé se alimenta a tensión alterna de 100 a 230 V, proporciona tensión continua 24V y 5V y sus salidas son de tipo Relé (de ahí la terminología AC/DC/Relé). Consta de los siguientes elementos:

- Una unidad central de procesamiento o CPU.
- Una fuente de alimentación integrada con salida 24VDC para sensores y actuadores y 5VDC para alimentación de módulos de ampliación.
- Leds de estado. Se contemplan tres estados:
 - RUN
 - STOP
 - FALLO
- Selector de estado. Tres posiciones:
 - RUN: ejecución del programa de usuario.
 - STOP: la CPU no procesa ningún programa de usuario.
 - TERM: posición para controlar el estado de la CPU por software.
- Ranura para cartucho de memoria.
- Puerto de comunicación integrado.
- 14 entradas digitales a 24VDC.
- 10 salidas digitales tipo relé a 24VDC/24 a 230 VAC.

- 2 potenciómetros analógicos con resolución 8 bits asociados a dos áreas de memoria internas

En cuanto a características técnicas de esta CPU hay que destacar los siguientes puntos importantes para este proyecto:

- *Comunicación integrada.*
Puerto de comunicación integrado para comunicaciones PPI/MPI/Freeport. En este proyecto se utilizan la comunicación modo MPI y la comunicación modo Freeport.
- *Mapa de memoria.*
Número de entradas, salidas, contadores, temporizadores, marcas, bloques de datos, funciones integradas. son datos relevantes a la hora de programar una CPU 224.
- *Marcas especiales.*
Las marcas especiales permiten intercambiar datos entre la CPU y el programa. Dichas marcas se puede utilizar para seleccionar y controlar algunas funciones especiales de la CPU 224. En este proyecto se utilizan:
 - SM0.0. – Marca en estado “1” siempre.
 - SM0.1. – Marca que pasa a estado “1” sólo en el primer ciclo de la CPU.
 - SMB30. – Registros de control modo Freeport.
 - SMB87, SMB88, SMB92, SMB94. – Control de recepción de mensajes.
 - SMB226, SMB228. – Datos del estado módulo EM 277.

3.2.2 Serie S7-300. CPU 314C-2DP

En este proyecto se ha utilizado un autómatas serie S7-300 de Siemens, modelo CPU 314C-2DP. Los autómatas de esta familia son PLCs de gama media, rápidos, de alto rendimiento y versátiles (CPU 31xC con funcionalidad integrada); indicados para aplicaciones con un grado de automatización medio-alto. En la Fig. 3.2. se observan los elementos y la estructura de una CPU de esta gama.

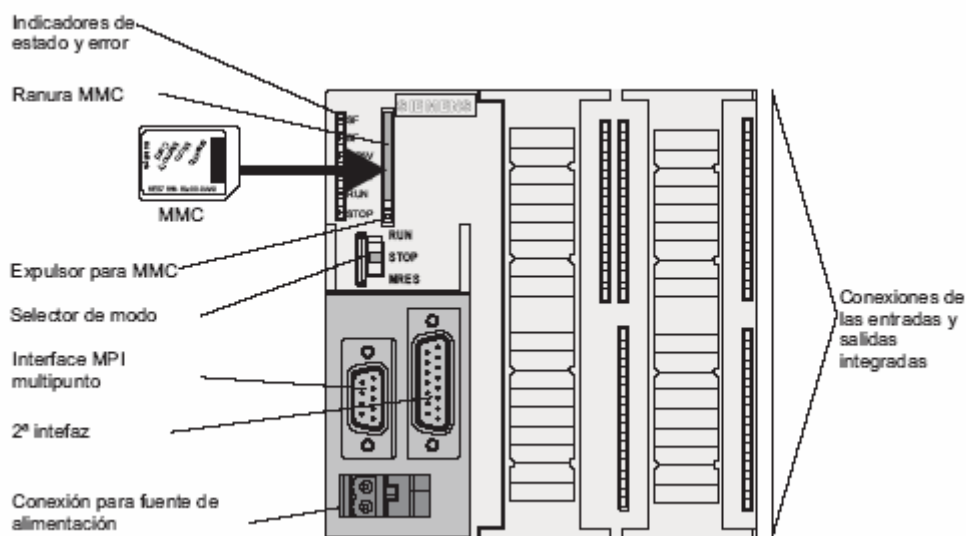


Fig. 3.2. Vista general de una CPU314C-2DP.

La CPU 314C-2DP se alimenta a tensión alterna de 100 a 230 V desde una fuente de alimentación externa. Los elementos que constituyen la CPU son:

- Unidad central de procesamiento o CPU.
- Indicadores de estado y de error. Posee varios indicadores con varios significados (Tabla 3.1.).

Indicadores de la CPU 314C-2DP	
SF (rojo)	Fallo de hardware o software
BF (rojo)	Fallo de bus
DC5V (verde)	Alimentación de 5 V para CPU y bus S7-300 funciona correctamente
FRCE (amarillo)	Petición de forzado activa
RUN (verde)	CPU en RUN; Led parpadea en ARRANQUE a 2 Hz; en PARADA a 0.5 Hz
STOP (amarillo)	CPU en STOP o en PARADA o en ARRANQUE. Led parpadea a 0.5 Hz al solicitar borrado, durante el borrado a 2 Hz

Tabla 3.1. Indicadores de la CPU 314C-2DP.

- Selector de modo de operación. Son posibles tres posiciones:
 - RUN: la CPU procesa el programa de usuario.
 - STOP: la CPU no procesa ningún programa de usuario.
 - MRES: borrado total. Posición no enclavable del selector para el borrado total de la CPU.

- Ranura para cartucho de memoria. El tipo de memoria es Micro Memory Card SIMATIC (MMC). Es necesaria para el funcionamiento de la CPU, pues este modelo no dispone de memoria de carga integrada.
- Puertos de comunicaciones. La CPU 314C-2DP posee dos interfaces de comunicación integrados que son:
 - Interface MPI (Multi Point Interface).
 - Interface PROFIBUS DP (Periferia Descentralizada).
- 24 entradas digitales integradas a 24VDC.
- 16 salidas digitales integradas a 24VDC
- 5 entradas analógicas integradas.
- 2 salidas analógicas integradas.

En cuanto a características técnicas de esta CPU hay que destacar los siguientes puntos importantes para este proyecto:

- *Comunicación integrada.*
La CPU 314C-2DP dispone de dos puertos de comunicación integrados: un interface MPI y un interface PROFIBUS DP para periferia descentralizada. En este proyecto se hace uso de ambos interfaces.
- *Mapa de memoria.*
Número de entradas, salidas, contadores, temporizadores, marcas, bloques de datos, funciones integradas. son datos relevantes a la hora de programar una CPU de la serie S7-300.

3.3 Fuente de alimentación PS307-2A

Esta fuente de alimentación es necesaria para alimentar la CPU 314C-2DP y los sensores/actuadores con 24VDC. Se dispone dentro del sistema S7-300 como un módulo de alimentación. La Fig. 3.3. muestra una fuente de alimentación de este tipo.

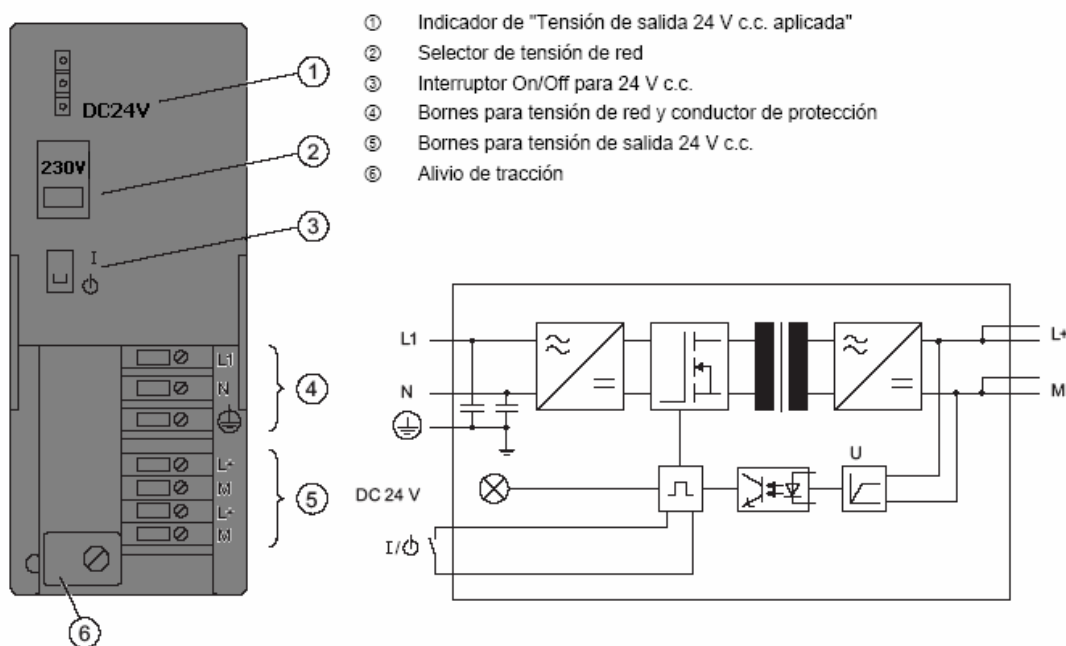


Fig. 3.3. Vista general de Fuente de Alimentación PS307-2A. Componentes internos.

La fuente de alimentación PS307-2A se alimenta a 120/230 VAC y proporciona una salida a 24 VDC, de intensidad 2A.

3.4 Módulo de comunicaciones EM277

El módulo EM 277 PROFIBUS-DP pertenece a la serie S7-200. Es un módulo de comunicación para conectar la CPU 224 a una red PROFIBUS DP (como esclavo) y a una red MPI. Soporta ambas conexiones al mismo tiempo. El puerto de comunicaciones se adapta al estándar RS 485. Es un conector del tipo 9-Pin Sub D I/O, hembra. En la Fig. 3.4. está representado un módulo de este tipo.

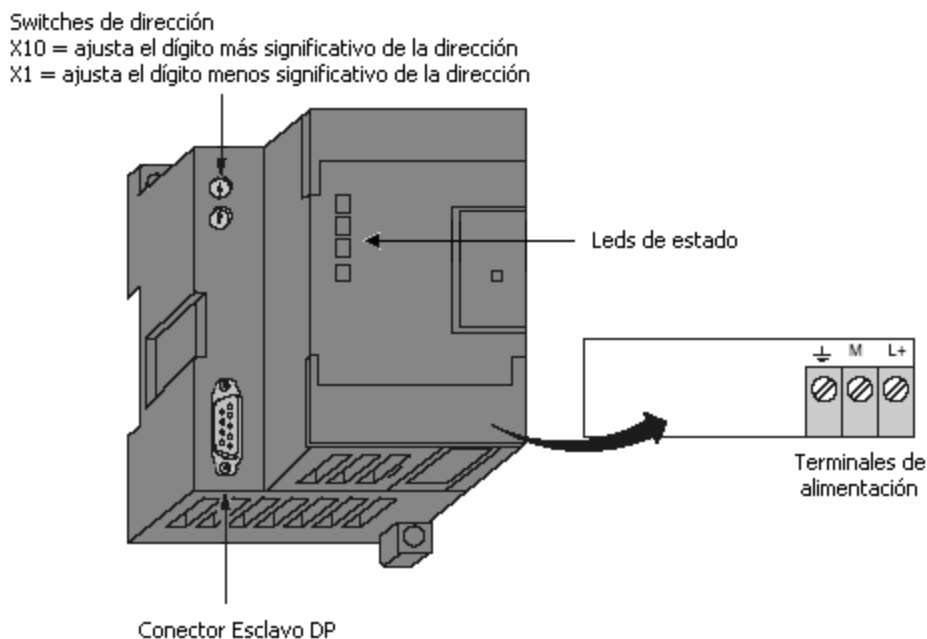


Fig. 3.4. Vista general modulo EM277.

El módulo EM 277 cuenta con 4 leds de estado en el panel frontal, los cuales indican:

LED	OFF	ROJO	PARPADEO ROJO	VERDE
CPU FAULT	No fallo	Fallo interno	-	-
POWER	No 24VDC	-	-	Alimentación OK
DP ERROR	No error	Datos perdidos	Error parametrización	-
DX MODE	No hay intercambio de datos	-	-	En modo intercambio de datos

3.5 Módulo de comunicaciones TC35.

El módulo de comunicaciones TC35 es un compacto MODEM GSM para la transferencia de datos, voz, mensajes cortos y faxes a través de la red GSM. Las principales características para este MODEM son:

- Soporta GSM900/1800.
- Sistema autobaudío para la comunicación serie.
- Entradas y salidas analógicas para micrófono y altavoz
- Antena magnética externa independiente.

- El puerto de comunicaciones del módulo TC35 es un interface RS 232, 9-pin D-Sub, conector hembra.
- Programación mediante comandos AT. Los comandos AT son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un terminal módem.

La Fig. 3.5. representa una visión general del modulo TC35.

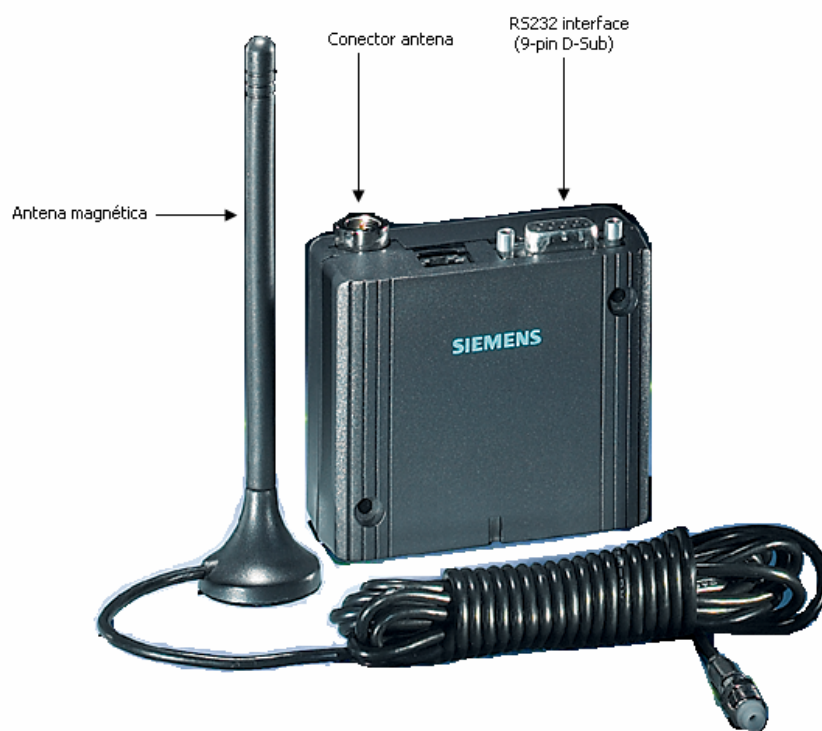


Fig. 3.5. Vista general modulo TC35.

El módulo TC35 puede encontrarse en diferentes estados de funcionamiento, indicados éstos por el led de estado de la siguiente forma:

LED	ESTADO
OFF	Modulo sin alimentación
Parpadeo intervalos 2 s	El módulo está alimentado
Parpadeo intervalos 2 ms	- Buscando red - Tarjeta SIM no insertada o tarjeta incorrecta - Nº PIN no insertado o incorrecto - Red no disponible
Parpadeo intervalos 3 s	Módulo registrado en red

3.6 Cable PC/PPI.

Es el cable de comunicación para la serie S7-200. Transforma RS 485 en RS 232, es decir, transforma el interface PPI (Point to Point Interface) del autómeta en RS 232 para comunicación con un PC, impresora, MODEM, etc. En la Fig. 3.6. puede verse un cable de este tipo, así como las diferentes configuraciones y velocidades de comunicación que soporta.

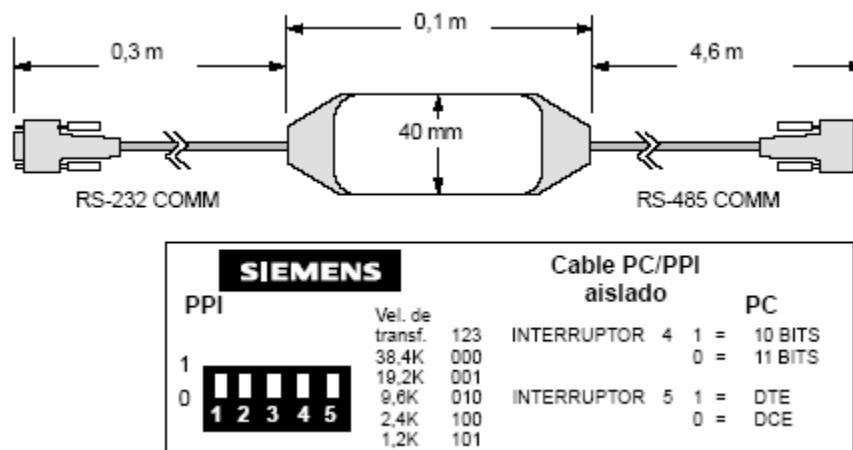


Fig. 3.6. Cable PC/PPI.

Las principales características para este tipo de cable, en relación con este proyecto son:

- Velocidades de transferencia desde 1.2 Kbit/s hasta 38.4 Kbit/s.
- Posibilidad de conexión en modo DTE ó modo DCE.
- Posibilidad de conexión a MODEM 10/11 bits.
- NO soporta conexión MPI/PROFIBUS, por lo que solo puede utilizarse con CPUs de la serie S7 200.

3.7 Cable PC Adapter.

El cable PC Adapter permite enlazar un PC a través de su puerto serie COM con el interface MPI/PROFIBUS DP de un sistema S7. Es el cable de comunicación para la serie S7-300. La Fig. 3.7. muestra un cable de este tipo.

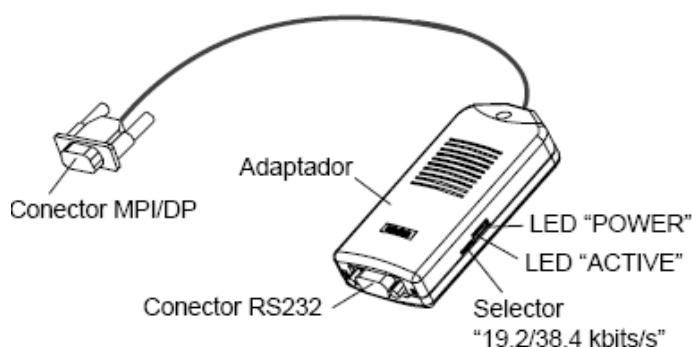


Fig. 3.7. Cable PC Adapter

Según se observa en la Fig. 3.7., este cable soporta dos velocidades de transferencia hacia el PC (lado RS 232): **19.2 y 38.4 Kbits/s**. En el lado MPI/DP soporta velocidades de 9.6 Kbit/s a 1.5 Mbit/s.

3.8 Cable PROFIBUS.

El cable utilizado en este proyecto recibe el nombre de PROFIBUS FC Standard Cable (Fig. 3.8.). Se trata de un cable **bifilar de cobre apantallado y trenzado** con sección circular, que se ajusta al método de transmisión RS 485. Cabe destacar los siguientes puntos:

- Alta inmunidad contra perturbaciones gracias al *doble apantallamiento*.
- *Puesta a tierra homogénea* realizable gracias a la pantalla exterior del cable.
- *Marcas impresas cada metro*.

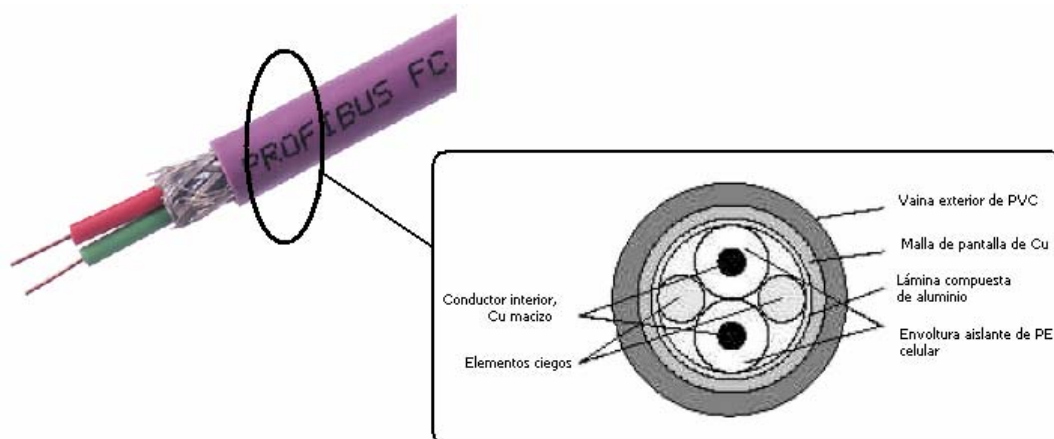


Fig. 3.8. Cable PROFIBUS FC Standard.

3.9 Conector PROFIBUS.

La conexión del cable PROFIBUS a los diferentes equipos utilizados se realiza mediante conectores de bus RS 485 para PROFIBUS (figura 3.9).

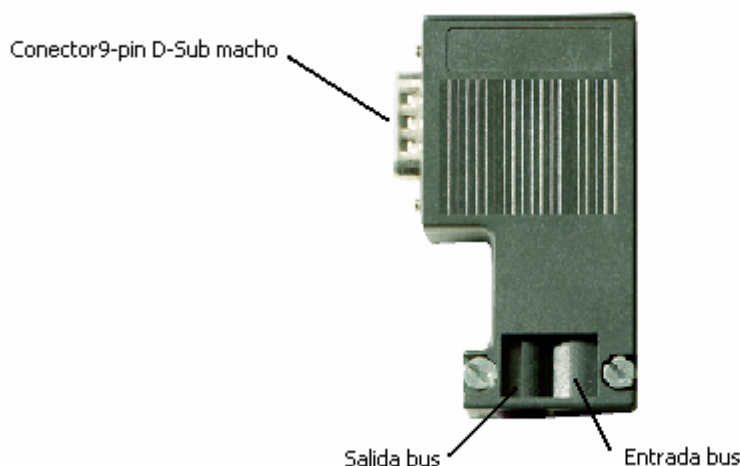


Fig. 3.9. Conector PROFIBUS.

Por la ranura entrada de bus se conecta el segmento PROFIBUS que procede del anterior dispositivo del bus. Por la ranura salida de bus saldrá el segmento PROFIBUS destinado a la conexión del siguiente dispositivo del bus. Si tras el dispositivo actual no hay más conexiones, estos conectores disponen de una **resistencia de terminación de bus** para evitar interferencias en el mismo. Esta resistencia se activa mediante un selector en la parte superior del conector, seccionando el bus saliente. En cuanto al tipo de conexión, estos conectores son del tipo **9-pin D-Sub macho**.

3.10 Módulos de fabricación flexible

Un sistema de fabricación flexible es aquel que es capaz de producir una gran variedad de productos sin perder tiempo de intercambio entre un producto y otro. No hay pérdida de tiempo en la producción al variar de producto. Estos sistemas están compuestos por unidades o módulos de fabricación flexible, cada uno de ellos con una determinada función dentro del sistema. Las dos unidades utilizadas en la presente aplicación han sido un “*doble alimentador por gravedad*” y una “*cinta transportadora*”. A continuación se describen de forma breve ambos módulos.

3.10.1 Módulo doble alimentador por gravedad

El módulo “doble alimentador por gravedad” (Fig. 3.10.) consta de dos torres de alimentación de piezas colocadas sobre una unidad lineal neumática sin vástago y comandada por una válvula 5/2. Un cilindro expulsor de simple efecto realiza la extracción de piezas. La pieza extraída es situada sobre un soporte de aluminio para su posterior manipulación. Sobre este soporte se sitúa un microinterruptor o sensor electro-mecánico. De igual forma, tanto la unidad lineal como el cilindro extractor disponen de ranuras donde se ubican los sensores magnéticos REED finales de carrera.

El “doble alimentador por gravedad” dispone de un panel de control con un interruptor, un pulsador, una seta de emergencia con señalización naranja y tres lámparas de iluminación blanca, verde y amarilla. Este panel de mando se utilizará para accionar el módulo. Además la unidad dispone de un inteface para comunicación con el PLC.

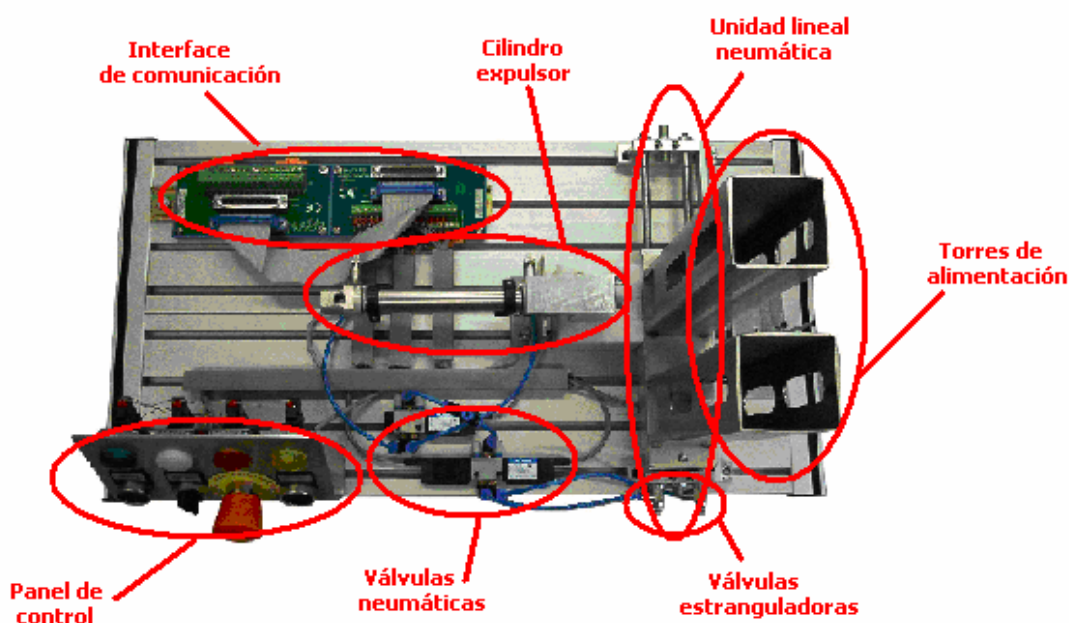


Fig 3.10. Módulo “doble alimentador por gravedad”

3.10.2 Módulo cinta transportadora

La unidad funcional “cinta transportadora” (Fig. 3.11.) consta de una cinta transfer de goma verde movida por un motor de corriente continua de 24V y 1.8W de potencia, capaz de obtener un par de 4Nm. El motor dispone de un sistema reductor y de un encoder magnético que proporciona un pulso de 24V por revolución del eje del

motor. El motor cambia de sentido de giro gracias a la acción de un sistema de relés que invierten la polaridad de la alimentación.

La cinta transportadora dispone de un panel de control con un interruptor de selección Manual / Automático, un pulsador de marcha, una seta de emergencia con luz y dos lámparas de iluminación de colores verde y blanca. Además la unidad dispone de un interface para comunicación con el PLC.

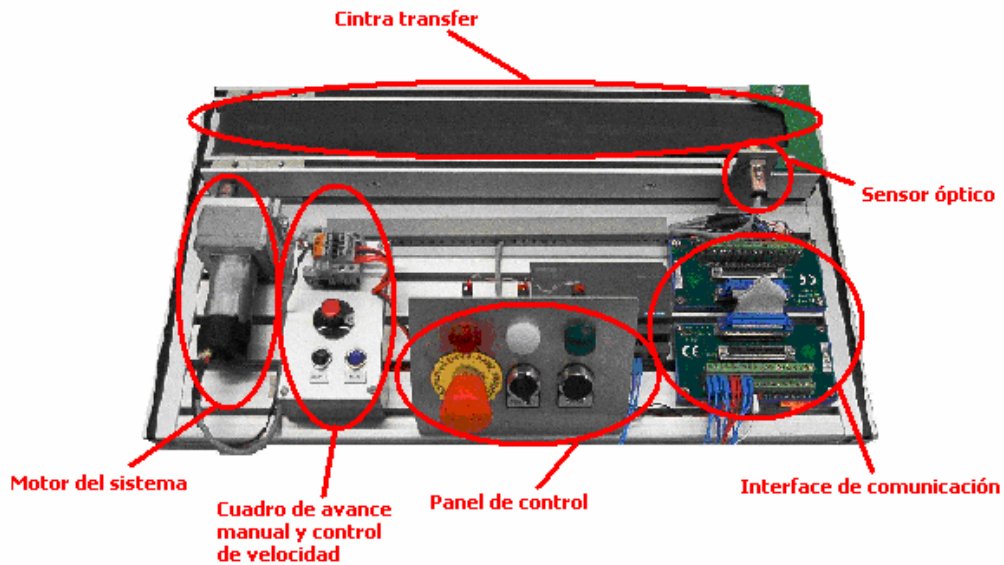


Fig. 3.11. Módulo "cinta transportadora"

Capítulo 4

Montaje de la red de comunicación

4.1 Introducción

En este capítulo se describen los pasos a seguir para el montaje de una red de comunicación industrial basada en PROFIBUS, MPI y GSM con los equipos que se han descrito en el Capítulo 3. La red de comunicación está formada por los siguientes elementos:

- CPU 314C-2DP.
Es el maestro de la red de comunicación. Lee/escribe datos en los esclavos y en el SCADA. De aquí en adelante, se denominará **“Maestro”**.
- CPU 224 AC/DC/Relé + EM 277.
Esclavo conectado al “Maestro” vía PROFIBUS DP. Su función es la de controlar el módulo *“doble alimentador por gravedad”*. De aquí en adelante, se identificará como **“Esclavo DP”**.
- CPU 224 AC/DC/Relé + EM 277.
Esclavo conectado al “Maestro” vía MPI, encargado de controlar el módulo *“cinta transportadora”*. Además de comunicarse vía MPI con el equipo “Maestro”, también tiene implementado un enlace PROFIBUS DP con el

objetivo de hacer más versátil al sistema. De aquí en adelante, se identificará como “**Esclavo MPI**”.

- MODEM GSM TC35.
Se encuentra conectado al PLC configurado como “Esclavo DP” a través de su puerto integrado. Su función es la de enviar/recibir determinados mensajes SMS a un Terminal de telefonía móvil gracias a una tarjeta SIM que deberá insertarse en el equipo para la supervisión y el control del “*doble alimentador por gravedad*”.
- SCADA.
Equipo de supervisión y control del sistema. Lee/escrbe datos en el equipo “Maestro”. Se encuentra implementado en un PC.
- Cable PROFIBUS.
Es el bus de comunicación utilizado. Soporta la conexión vía PROFIBUS DP entre “Maestro”, “Esclavo DP” y “Esclavo MPI” y la conexión vía MPI entre “Maestro” y “Esclavo MPI”.
- Cable PC Adapter.
Cable de comunicación encargado de conectar SCADA con “Maestro”.
- Cable PC/PPI.
Cable de comunicación para conectar “Esclavo DP” con MODEM TC35. **Importante:** la salida RS 232 del cable PC/PPI es un conector DB9 hembra, al igual que el puerto de conexión del MODEM TC35. Por esta razón, ha sido necesario crear un *adaptador macho-macho* para la comunicación. El montaje de este adaptador está descrito en el *Apartado 4.2.4.* de este Capítulo.
- Módulos de fabricación flexible.
Son los dos procesos a controlar.

La Fig. 4.1. muestra una ilustración del montaje realizado.

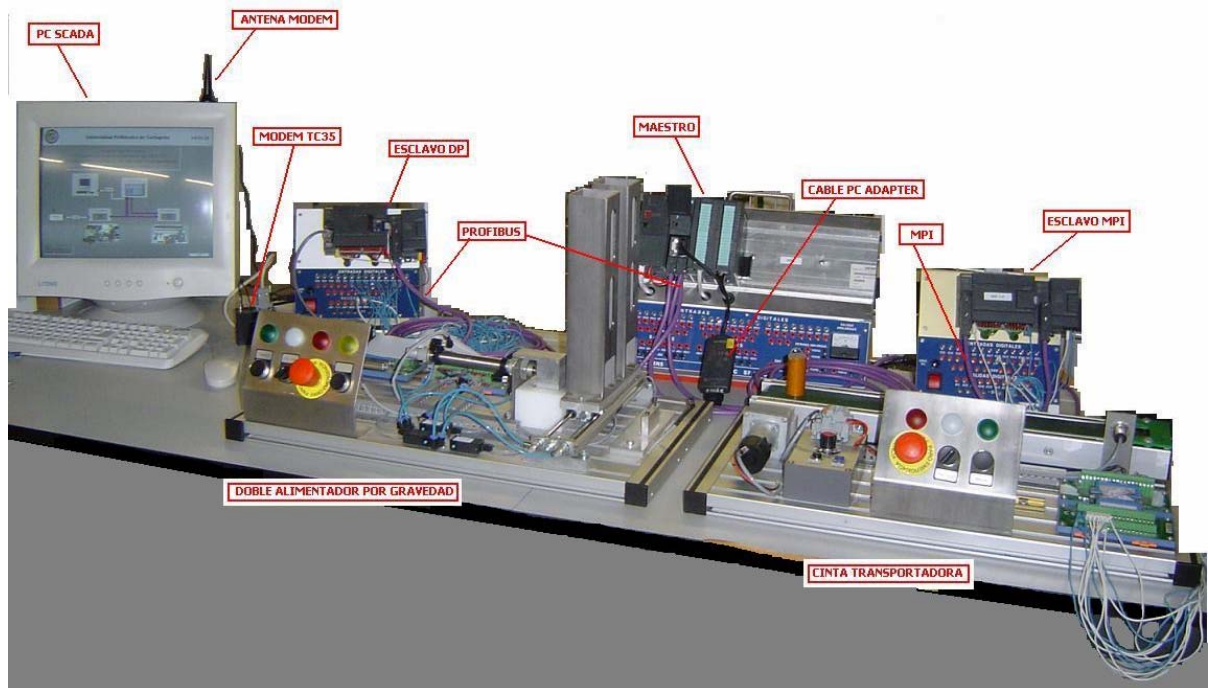


Fig 4.1. Ilustración del montaje realizado

4.2 Montaje equipo Maestro

El equipo “Maestro” (CPU 314C-2DP), de acuerdo a la aplicación que se ha implementado y a las características de la CPU, no necesita ser modificado en cuanto a hardware, por lo que lo único necesario en cuanto a montaje es su conexión a la red de alimentación (230VAC).

4.3 Montaje equipos Esclavo DP y Esclavo MPI

Los equipos “Esclavo DP” y “Esclavo MPI” (CPUs 224), para ser integrados en la red PROFIBUS, necesitan del módulo de comunicaciones EM 277. Para realizar este montaje es necesario:

- En primer lugar, conectar el módulo EM 277 con la CPU 224 a través del **bus de E/S serie**. (Fig. 4.2.). A través de esta conexión CPU y módulo intercambian datos.

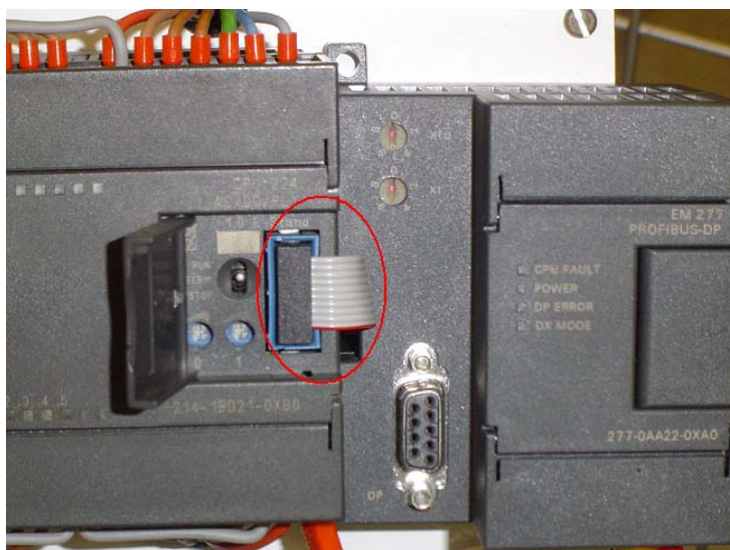


Fig. 4.2. Conexión bus de E/S serie

- A continuación, **alimentar el módulo EM 277**. En este caso, se ha optado por alimentarlo desde la CPU 224. (Fig. 4.3.).

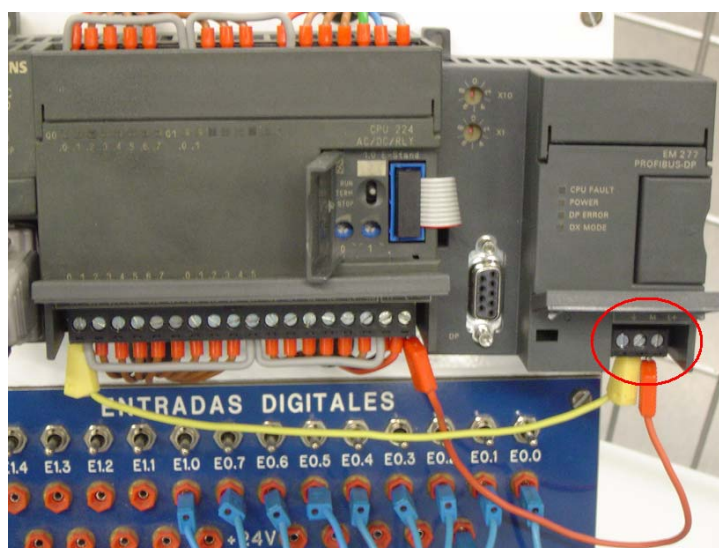


Fig. 4.3. Alimentación del módulo desde CPU (+L y M).

4.4 Montaje cable PROFIBUS

Como se comentó al inicio de este capítulo, soporta la conexión entre “Maestro”, “Esclavo DP” y “Esclavo MPI” (vía PROFIBUS) y la conexión entre “Maestro” y “Esclavo MPI” (vía MPI). En las Fig. 4.4. y Fig. 4.5. se observan el segmento de cable PROFIBUS y el segmento de cable MPI, respectivamente.



Fig. 4.4. Segmento de cable PROFIBUS



Fig. 4.5. Segmento de cable MPI

A la hora de crear el cable es de especial importancia la forma en la que se conectan los conectores. Se establecen dos tipos de conexiones:

- Conexión intermedia de bus. Corresponde con el terminal conectado en equipo “Maestro”, en su interfaz PROFIBUS. Su montaje se observa en la Fig. 4.6. El conector posee un conector superior que permite conectarse a este punto de la red para realizar acciones como monitorización de red, configuración de equipos, etc.

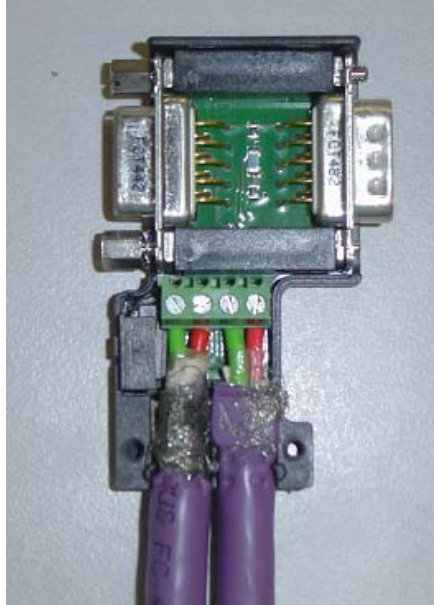


Fig. 4.6. Conexión intermedia de bus con terminal de conexión superior

- Conexión final de bus. Corresponde con los demás terminales (equipo “Maestro”, interfaz MPI; equipos “Esclavo DP” y “Esclavo MPI”, interfaz PROFIBUS DP a través de modulo EM277; equipo “Esclavo MPI”, interfaz MPI a través de puerto integrado). Su montaje se observa en la Fig. 4.7.

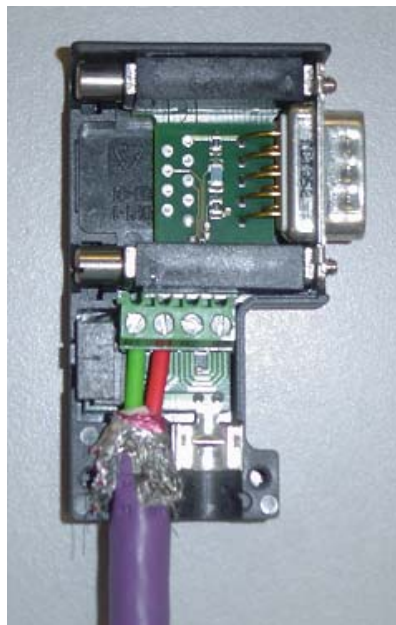


Fig. 4.7. Conexión final de bus

4.5 Montaje adaptador macho-macho

Este cable de conexión es necesario por la incompatibilidad de conectores entre el cable PC/PPI (extremo RS 232, hembra) y el puerto de conexión del MODEM TC35 (también hembra). Para solucionar este problema es necesario crear un adaptador macho-macho que permita conectar ambos interfaces y así comunicar el PLC y el MODEM.

Este adaptador esta formado por 2 conectores machos tipo DB9 (9 pines) y un cable de 9 hilos. La asignación de pines para estos conectores es uno a uno, es decir, pin 1 con pin 1, pin 2 con pin 2; así sucesivamente excepto los pines 7 y 8 que se encuentran puenteados. El pin 7 corresponde a la señal de salida RTS (Solicitud de envío), mientras que el pin 8 se trata de la señal de entrada CTS (Libre para envío). Por tanto, la razón de este puente no es otra que la de conseguir que cuando el MODEM quiera enviar datos al PLC no necesite de un permiso para ello, y lo haga automáticamente (al solicitar el envío activa el pin 7 que, al estar puenteado con pin 8, proporciona la señal “libre para envío” al MODEM). La Fig. 4.8. representa el adaptador macho-macho montado.



Fig. 4.8. Adaptador macho macho

4.6 Montaje equipo MODEM TC35

La instalación del equipo MODEM TC35 resulta relativamente sencilla. En primer lugar se deben asegurar unas premisas para el buen funcionamiento del equipo. Estas son:

- Anotar los parámetros código PIN y número del centro de mensajes de la tarjeta SIM.
- Insertar la tarjeta SIM en un teléfono móvil compatible y confirmar el servicio.
- Borrar todos los mensajes SMS almacenados en la tarjeta (*)
- Borrar todas las entradas almacenadas en la agenda de la tarjeta (*)

(*) Es muy importante cumplir las premisas anteriores. De lo contrario, la comunicación puede no funcionar bien o no realizarse.

Una vez hecho esto, se procede a la instalación del MODEM TC35 siguiendo los siguientes pasos:

- 1) Utilizando un objeto punzante, tal como un bolígrafo, se presiona el botón amarillo situado a la derecha de la ranura de la tarjeta SIM, para así extraer la plataforma donde la tarjeta SIM es colocada (Fig. 4.9.).



Fig. 4.9. Extracción soporte tarjeta SIM

- 2) Se coloca la tarjeta SIM de forma que el chip de la tarjeta quede orientado hacia arriba y se insertará de nuevo correctamente la plataforma SIM en la ranura (Fig. 4.10.).



Fig. 4.10. Posición tarjeta SIM en plataforma

- 3) Se conecta la antena magnética (Fig. 4.11.).



Fig. 4.11. Conexión antena magnética a MODEM

- 4) Se conecta el cable de comunicación al conector DB9 hembra (Fig 4.12.).



Fig. 4.12. Conexión cable de comunicación

- 5) Se alimenta el equipo a través del transformador suministrado con el mismo (Fig. 4.13.).



Fig. 4.13. Conexión alimentación al MODEM

4.7 Montaje módulos de fabricación flexible

Hablar del montaje de los módulos de fabricación flexible se refiere a realizar todo el conexionado de entradas / salidas entre el módulo y el PLC que lo va a controlar, así como alimentar el propio módulo. Los esquemas de conexionado de

ambas unidades se han obtenido a partir de la **Referencia Bibliográfica [1]**. En dicha documentación se explican las arquitecturas y los componentes de los módulos y se recogen esquemas mecánicos y eléctricos de las mismas.

En nuestro caso, se han seguido los siguientes esquemas de conexionado:

- Unidad funcional “Cinta transportadora” (Tabla 4.1.).

Pin	Descripción	Color de cable	Vble PLC
1	Alimentación 24V	azul	24V
2	24V Señal de emergencia	Verde	E0.5
3	24V para motor	Negro	
4	24V para encoder	Marrón	
5	24V para sensor	Marrón	
6			
7	0V para sensor	Azul	
8	0V para encoder	Negro	
9	0V para lámparas	Blanco/verde	
10	0V para motor	Marrón	0V
20	Pulsador marcha	Marrón	E0.0
21	Interruptor Manu/Auto	Naranja	E0.1
22	Lámpara verde	Blanco/azul	A0.0
23	Lámpara blanca	Blanco/naranja	A0.1
24	Motor derecha	Azul	A0.2
25	Motor izquierda	Marrón	A0.3
26	Señal encoder	Azul	E0.6
27	Señal sensor	negro	E0.3

Tabla 4.1. Mapa de conexiones para cinta transportadora.

- Unidad funcional “Doble alimentador por gravedad” (Tabla 4.2.).

Pin	Descripción	Color de cable	Vble PLC
1	Alimentación 24V	azul	24V
2	24VDC Señal de emergencia	Verde	E0.0
3			
4			
5			
6	24VDC para Microinterruptor		
7	24VDC para REED Cilindro atrás		
8	24VDC para REED Cilindro delante		
9	24VDC para REED Unidad lineal Izq.		
10	24VDC para REED Unidad lineal Drcha.		
11			
12			
13			
14	Válvula 5/2 Cilindro expulsor		A0.0
15	Válvula 5/2 Unidad lineal izquierda		A0.1
16	Válvula 5/2 Unidad lineal derecha		A0.2
17	Interruptor Manu/Auto	Naranja	E0.1
18	Pulsador marcha	Marrón	E0.2
19	Pulsador preparado	Azul	E0.3
20	0V válvula cilindro expulsor		0V
21	0V válvula Unidad lineal izquierda		
22	0V válvula Unidad lineal derecha		
23	0V REED Cilindro expulsor atrás		
24	0V REED Cilindro expulsor delante		
25	0V REED Unidad lineal izquierda		
26	0V REED Unidad lineal derecha		
27	0V Lámparas		
28			
29	Microinterruptor		E0.4
30	Cilindro expulsor atrás		E0.5
31	Cilindro expulsor delante		E0.6
32	Unidad lineal izquierda		E0.7
33	Unidad lineal derecha		E1.0
34	Lámpara blanca		A0.3
35	Lámpara verde		A0.4
36	Lámpara amarilla		A0.5
37			

Tabla 4.2. Mapa de conexiones para doble alimentador por gravedad.

Nota: Las flechas representadas en el cuadro se refieren a los puentes existentes en el bornero eléctrico de la maqueta

Una vez realizado el montaje de los módulos, ya están listos para funcionar de acuerdo a un programa de usuario introducido en el PLC que los controla.

Capítulo 5

Configuración de equipos y red de comunicación

5.1 Introducción

La **aplicación práctica** llevada a cabo en este proyecto consiste en el control y la monitorización de dos módulos de fabricación flexible gracias a una red de comunicación basada en PROFIBUS, MPI y GSM. Por este motivo, tanto la red de comunicación como los equipos que la forman se configurarán orientados a dicha aplicación práctica.

Un primer paso antes de comenzar con la configuración de los diferentes equipos es conocer el funcionamiento de los módulos de fabricación flexible que van a controlarse. Seguidamente se describirán los distintos pasos a seguir para configurar la red de comunicación y los equipos integrados en dicha red.

5.2 Funcionamiento “doble alimentador”

El funcionamiento de la unidad “doble alimentador por gravedad” queda resumido en la Tabla 5.1., en la que se observan los dos modos de funcionamiento del módulo (Automático y Manual), así como las condiciones que determinan un modo u otro.

	Modo Automático	Modo Manual
Condiciones	Interruptor Manual/Automático de panel de control local y de panel de control SCADA en Automático Accionar Marcha en panel de control local o panel de control SCADA, o a través de mensaje SMS	Interruptor Manual/Automático de panel de control local y de panel de control SCADA en Manual Accionar Marcha o Preparado en panel de control local o panel de control SCADA
Funcionamiento	La unidad realiza ciclos completos de extracción de piezas	Si Marcha, avance/retroceso de cilindro Si Preparado, izquierda o derecha torres de alim.

Tabla 5.1. Resumen del funcionamiento unidad doble alimentador

Ciclo completo de extracción de piezas. Si se pulsa marcha, el ciclo empieza y si no están las condiciones iniciales definidas por el cilindro extractor retraído y las torres de alimentación a la derecha, se situarán con el pulsador de preparado. Una vez puesto en marcha el sistema, el cilindro expulsor realizará la extracción de la primera pieza siempre que el sensor electromecánico (microinterruptor) no detecte la presencia de ningún objeto. Logrado el tope del cilindro, éste se retrae hasta su posición inicial, momento en el cual se desplazarán las torres hacia la izquierda. Cuando las torres lleguen a su destino, el cilindro extractor realizará la extracción de la segunda pieza siempre y cuando el sensor electromecánico no detecte ningún objeto. El ciclo se repetirá de forma continuada hasta que se seleccione modo de funcionamiento Manual ó se origine una parada de emergencia. Además se establecen las siguientes condiciones:

- Parada de emergencia del sistema: provocada por la activación de la seta desde el panel de control local producirá la caída de todo el sistema. Esta parada de emergencia se notificará como alarma en el SCADA y mediante SMS.
- Activación sensor electromecánico: la activación del microinterruptor producirá una alarma en el SCADA y una notificación mediante SMS.

5.3 Funcionamiento “cinta transportadora”

El funcionamiento del módulo es el siguiente: si el interruptor Manual / Automático del panel de control local y del panel de control SCADA en Automático y accionamos Marcha en panel de control local o panel de control SCADA, la cinta arrancará hacia la derecha. Cuando el sensor fotoeléctrico detecte presencia de pieza, se detendrá la cinta y pasados 2 segundos, se pondrá en marcha hacia la izquierda.

Transcurridos 4.5 segundos, la cinta se detendrá y tras 1 segundo reiniciará la marcha hacia la derecha, repitiéndose el ciclo de vaivén de forma continuada, a menos que se seleccione en el interruptor Manual / Automático la posición Manual o se produzca un paro de emergencia. La función Manual no ha sido implementada. La parada de emergencia del sistema estará señalizada como una alarma en el SCADA.

5.4 Creación de la aplicación práctica

La configuración de la aplicación se ha realizado sobre un PC de sobremesa convencional, provisto del sistema operativo Windows 2000 y del paquete de software SIMATIC. Condición necesaria e imprescindible es que el PC incorpore puerto de comunicación serie, ya que a través del mismo se realiza la comunicación con los PLCs.

Se han utilizado 3 programas diferentes pertenecientes a SIEMENS, incluidos en el paquete de software SIMATIC. Cada uno de ellos sirve para configurar determinados elementos de la aplicación. Estos programas son:

- SIMATIC Manager V5.2.
Es el software necesario para la configuración del equipo **Maestro**, del **bus PROFIBUS** y del **bus MPI**, aunque englobará a todos los elementos de la aplicación. Este software se compone de una serie de aplicaciones o herramientas que permiten implementar soluciones parciales. Las utilizadas en el presente proyecto son:
 - *NETPro*. Configurar redes de comunicación.
 - *HWConfig*. Configuración hardware de equipos.
 - *KOP/AWL/FUP Editor*. Crear y parametrizar bloques de programa en el PLC.
- STEP 7 Microwin 32.
En este software se ha implementado la configuración para los equipos **Esclavo DP**, **Esclavo MPI** y **MODEM TC35**.
- ProTooL Pro V5.2.
Es el software utilizado para la implementación del sistema **SCADA**.

A continuación se describen los pasos a seguir que son necesarios para realizar la aplicación descrita.

Creación de un proyecto nuevo.

En primer lugar, se ha creado un proyecto nuevo en SIMATIC Manager V5.2., con el nombre “PFC”. En la Fig. 5.1. se representa la ventana resultante. De esta forma, se crea un nuevo proyecto en el cual se insertarán todos los componentes que conforman la aplicación (redes de comunicación, equipos, etc.).

Archivo >> Nuevo>> PFC

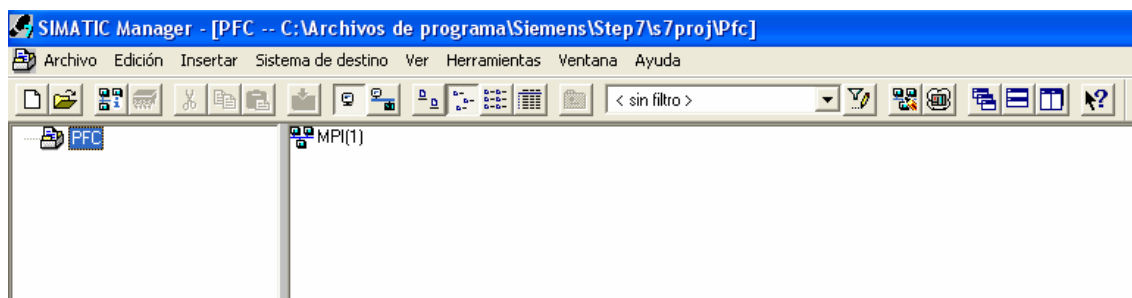


Fig. 5.1. Creación nuevo proyecto

Por defecto, el nuevo proyecto incorpora un bus MPI. Más adelante, este bus será modificado de acuerdo a la aplicación.

Insertar equipo “Maestro”

A continuación se inserta en el proyecto el equipo “Maestro” de la red. Para ello se utilizan los comandos: Insertar >> Equipo >> SIMATIC 300 (Fig. 5.2.). Una vez insertado, se cambia el nombre dado por defecto por el nombre “Maestro”.

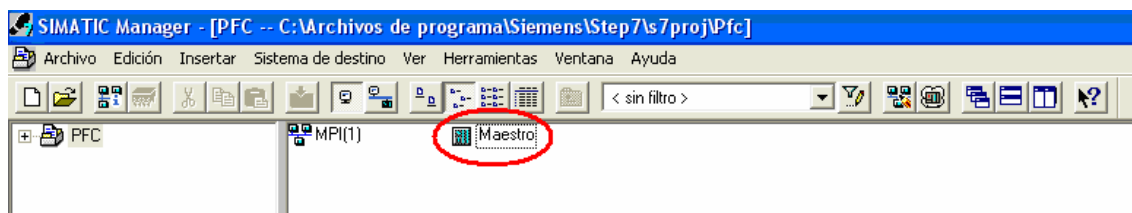


Fig. 5.2. Equipo Maestro

Insertar Bus PROFIBUS

Seguidamente, se inserta el bus de comunicación PROFIBUS. La forma de realizar esta acción es la siguiente: Insertar >> Subred >> PROFIBUS (Fig. 5.3.).

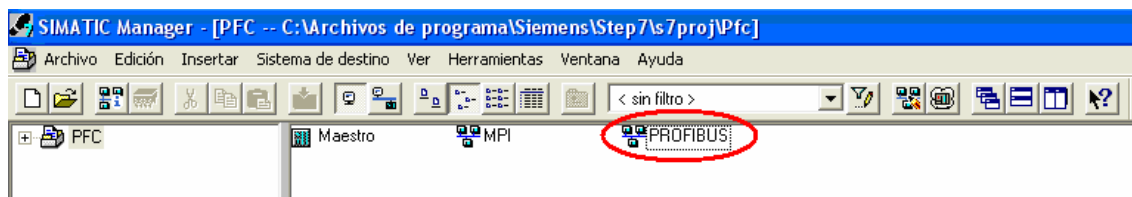


Fig. 5.3. Bus PROFIBUS

Insertar equipo SCADA

Para insertar el equipo SCADA se siguen los siguientes comandos: Insertar >> Equipo >> SIMATIC OP (Fig. 5.4.).

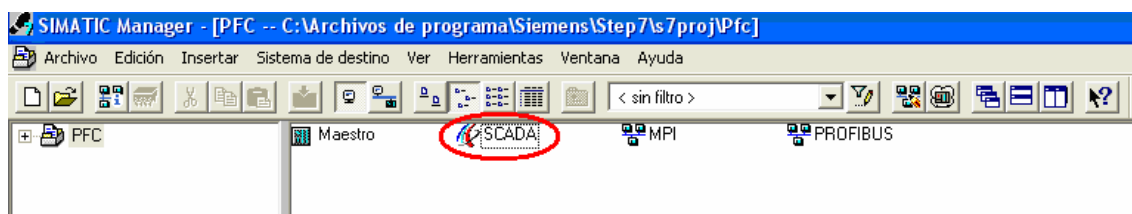


Fig. 5.4. Equipo SCADA

Insertar equipos "Esclavo DP" y "Esclavo MPI"

Como se comento al iniciar el presente apartado, la configuración de estos equipos no se realiza desde este software, sino que se lleva a cabo con "STEP 7 Microwin 32". El incluir dichos equipos en el proyecto responde a una razón de **comodidad** para el usuario. En otras palabras: insertando estos equipos se consigue un acceso directo a su software de programación, de forma que con un simple clic sobre el icono correspondiente se inicia el software *STEP 7 Microwin 32* para la configuración de "Esclavo DP" y "Esclavo MPI". El motivo por el cual estos equipos no pueden configurarse con *SIMATIC Manager V5.2.* se debe a razones de la marca SIEMENS.

La inserción de estos equipos en el proyecto sigue los comandos Insertar >> Equipo >> SIMATIC 200 Station. Una vez insertadas las dos estaciones que son necesarias se modificarán los nombres por defecto por los de "Esclavo DP" y "Esclavo MPI" (Fig. 5.5.).

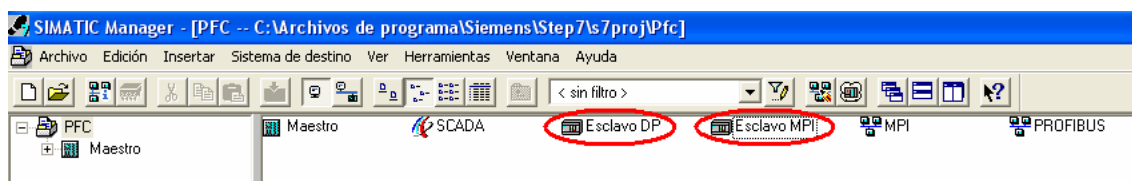


Fig. 5.5. Equipos “Esclavo DP” y “Esclavo MPI”

5.5 Configuración equipo Maestro

El “Maestro” de la red de comunicación es una CPU 314C-2DP. Este es un dato importante ya que para configurar este equipo es necesario saber el **modelo exacto** y todos los módulos de los que dispone. Para evitar confusiones, la mejor opción consiste realizar la configuración con el equipo físicamente “al lado”, para así no cometer fallos y obtener una configuración en el software acorde al hardware del que se dispone.

5.5.1 Configuración Hardware

El primer paso para configurar el equipo es realizar su configuración hardware. Los pasos a seguir son:

Entrar en la configuración hardware del equipo

En la ventana general de configuración del proyecto (Fig. 5.6.) se despliega el Menú “PFC”, se selecciona “Maestro” y se realiza la acción: Doble clic en el icono Hardware. De esta forma, se inicia la herramienta *HWConfig* (*configurar hardware*) mediante la cual se configurará módulo por módulo el equipo.

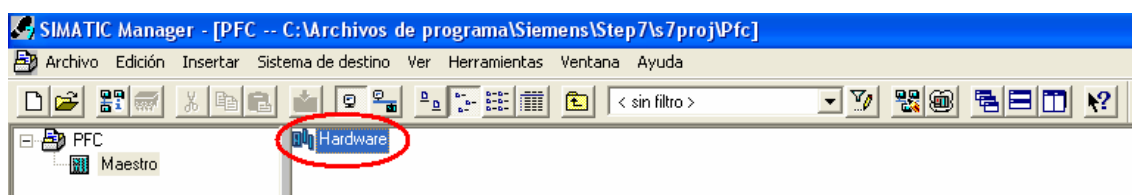


Fig. 5.6. Iniciar configuración hardware para equipo “Maestro”

Configurar hardware del equipo

Al iniciar la herramienta *HWConfig* la ventana que resulta es la representada en la Fig. 5.7. Como se observa, la herramienta dispone de diferentes ventanas para configurar el equipo.

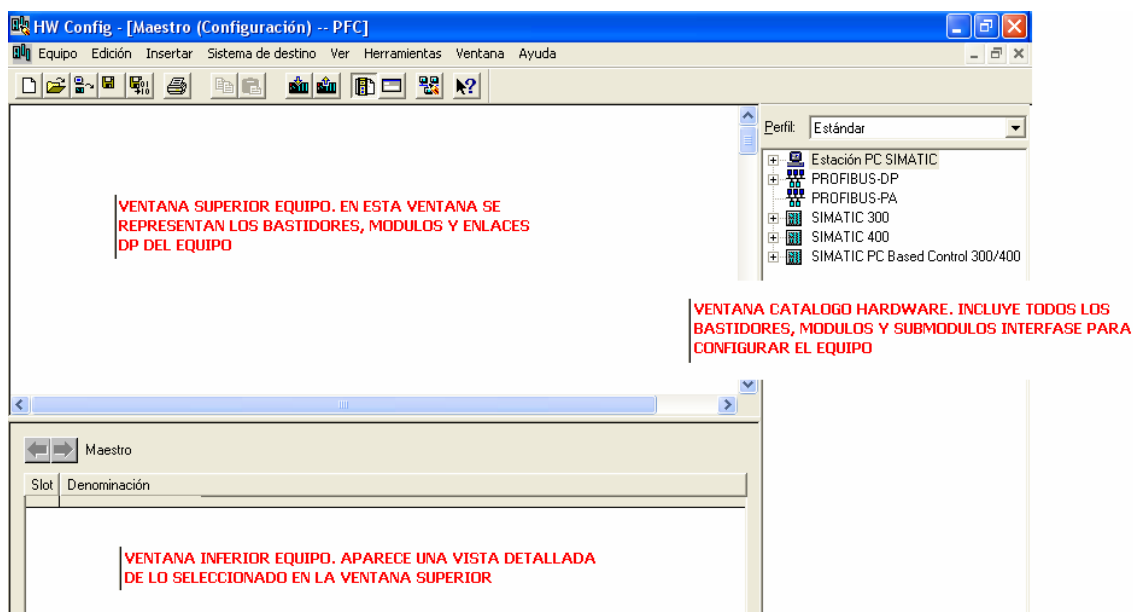


Fig. 5.7. Herramienta HWConfig

Procedimiento para configurar equipos

Localizar en la ventana Catálogo Hardware el módulo acorde y haciendo clic sobre el mismo arrastrarlo con el ratón a su correspondiente posición en la ventana superior (todo esto con el botón izquierdo del ratón pulsado).

Los pasos a seguir para configurar el equipo “Maestro” son los que a continuación se indican:

- 1) Se inserta un **Perfil Soporte**: Catálogo hardware >> SIMATIC 300 >> Bastidor 300 >> Perfil Soporte (Fig. 5.8.). Hay que advertir que tras haber insertado el perfil soporte, aparece automáticamente una tabla de configuración para el primer bastidor (*UR 0*).

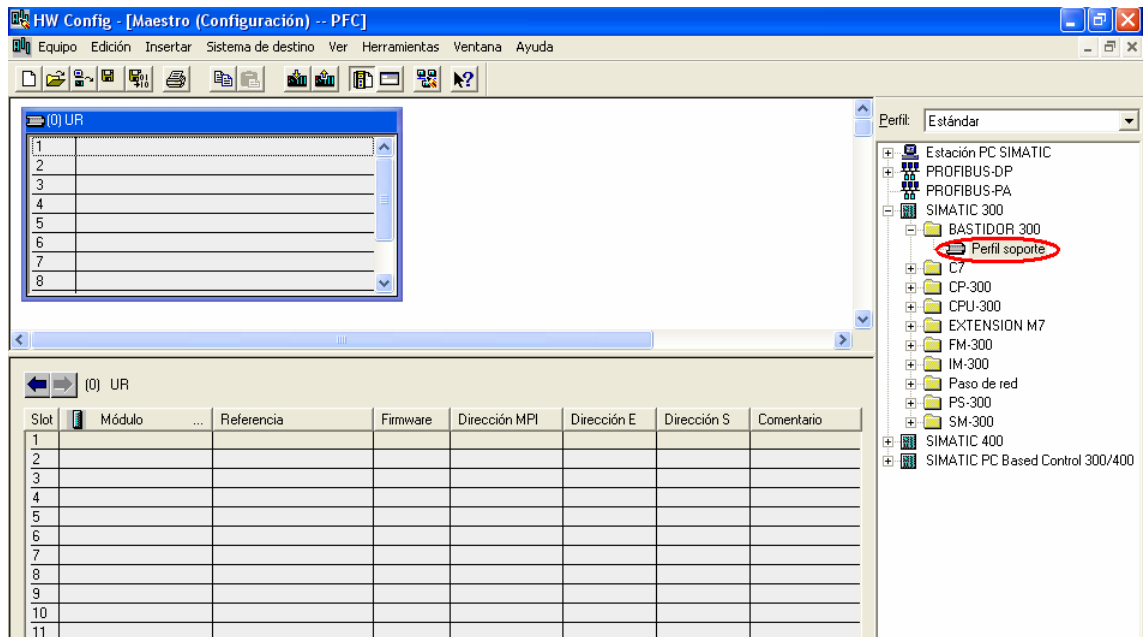


Fig. 5.8. Insertar perfil soporte

Ahora se pueden seleccionar todos los módulos en el catálogo hardware e insertarlos en la tabla de configuración del equipo, en su correspondiente posición en el bastidor (*la posición en el bastidor ha de corresponderse con la disposición física del equipo, y se denomina Slot*).

- 2) Se inserta la **Fuente de Alimentación PS 307 2A**: Catálogo hardware >> SIMATIC 300 >> PS 300 >> PS 307 2A (Fig. 5.9.). La fuente de alimentación ocupará en el bastidor el Slot 1.

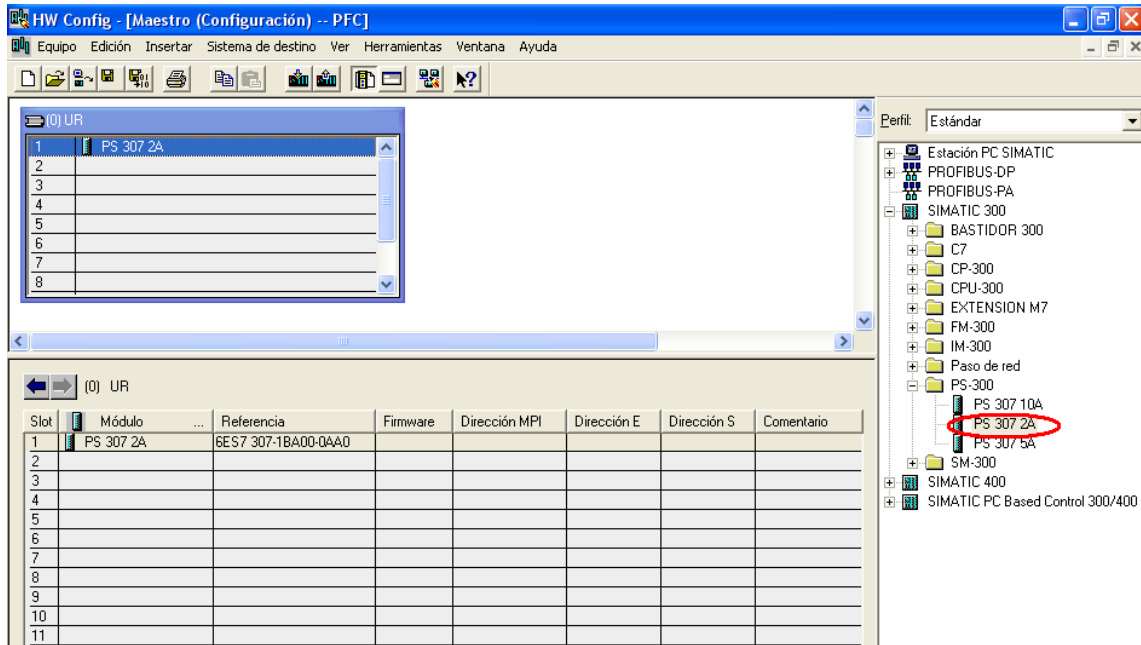


Fig. 5.9. Insertar fuente de alimentación PS307 2A

- 3) Se inserta la **CPU 314C-2DP**: *Catálogo hardware >> SIMATIC 300 >> CPU 300 >> CPU 314C-2DP* (Fig. 5.10.). La CPU ocupará el Slot 2 en el bastidor.

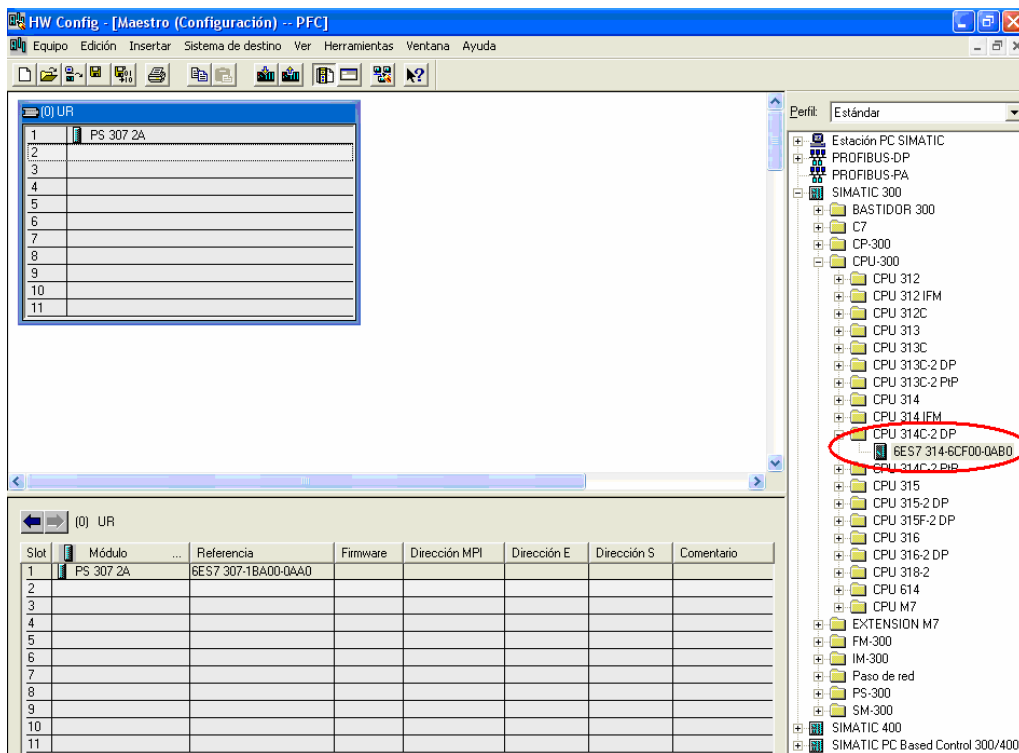


Fig. 5.10. Insertar CPU 314C-2DP

Al arrastrar la CPU hacia el Slot 2 del bastidor, se muestra la pantalla de la Fig. 5.11., en la que se pide asignar una dirección PROFIBUS. Para el caso de la presente aplicación, “Maestro”, se asigna la **dirección PROFIBUS 1** y se selecciona la **red PROFIBUS**.

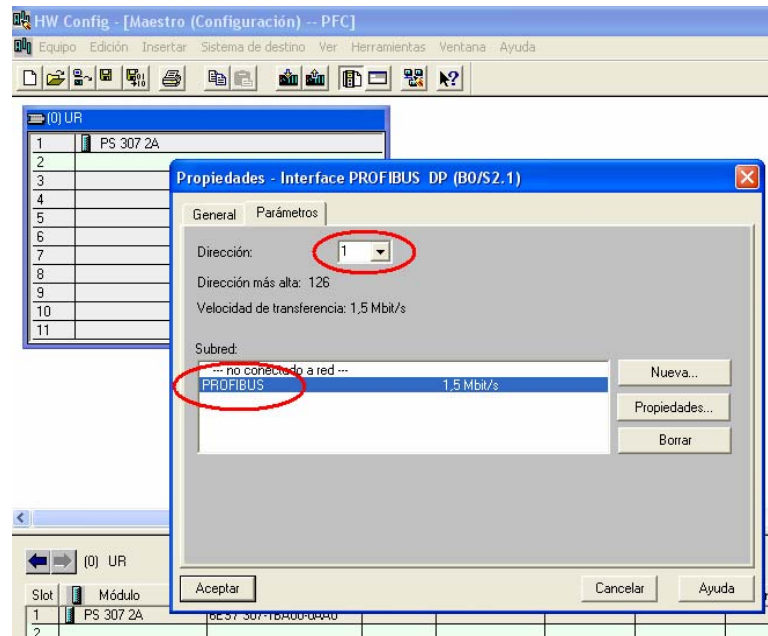


Fig. 5.11. Propiedades del interface PROFIBUS de la CPU 314C-2DP

Por defecto, el equipo queda configurado automáticamente como Maestro de la red PROFIBUS a la que ha sido conectado. La pantalla resultante muestra el equipo “Maestro” configurado (Fig. 5.12.).

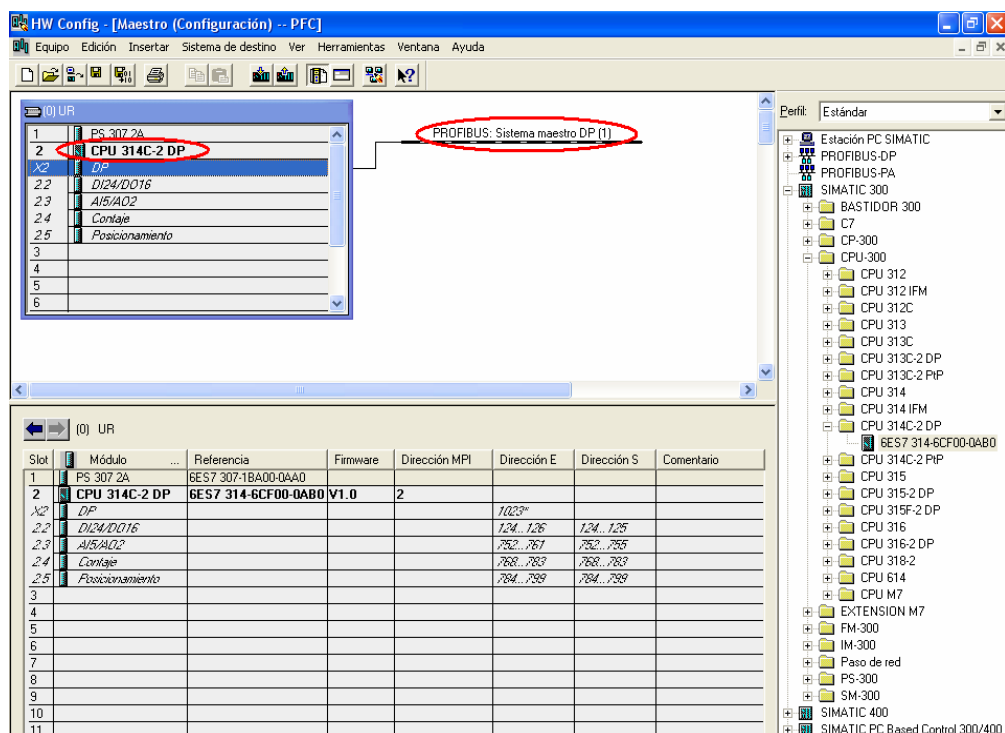


Fig. 5.12. Configuración hardware equipo "Maestro"

5.5.2 Configuración Software

La programación del equipo "Maestro" deberá diseñarse de forma que solucione los siguientes aspectos: comunicación PROFIBUS, comunicación MPI e interacción con SCADA.

En la programación de la CPU 314C-2DP se han utilizado cinco tipos de bloques de programa:

- **Bloque de datos del sistema (SDB):** es un área de datos localizada dentro del módulo central (CPU). Contiene ajustes del sistema y parámetros de configuración de módulos. Se crea y modifica de forma *automática* al configurar el hardware, los enlaces, las redes, etc.
- **Bloques de organización (OBs):** los bloques de organización constituyen el interface entre el sistema operativo de la CPU S7 y el programa de usuario. En los OBs se determina el orden de ejecución que deberá seguir el programa de usuario.
- **Bloques de datos (DBs):** los bloques de datos son áreas residentes en el programa de usuario. A diferencia de los demás bloques de programa, no contienen instrucciones, sólo datos.

- **Funciones (FCs):** las funciones son bloques lógicos sin memoria. Una función ofrece la posibilidad de transferir parámetros dentro del programa de usuario. Por esta razón, resultan idóneas para programar funciones complejas que se repitan con frecuencia.
- **Funciones del sistema (SFCs):** una SFC es una función integrada en el sistema operativo de la CPU S7 que se puede llamar en el programa de usuario cuando se necesite.

Más información sobre cada uno de estos bloques puede obtenerse a partir de la **Referencia Bibliográfica [4]**. Cada uno de estos bloques (a excepción del bloque SDB) necesita ser creado en el proyecto SIMATIC y posteriormente programado si es el caso. El método a seguir es el siguiente:

- 1) En la pantalla principal del proyecto PFC se selecciona el apartado “Programa S7” para el equipo “Maestro” y la carpeta “Bloques”. La Fig. 5.13. muestra esta operación.

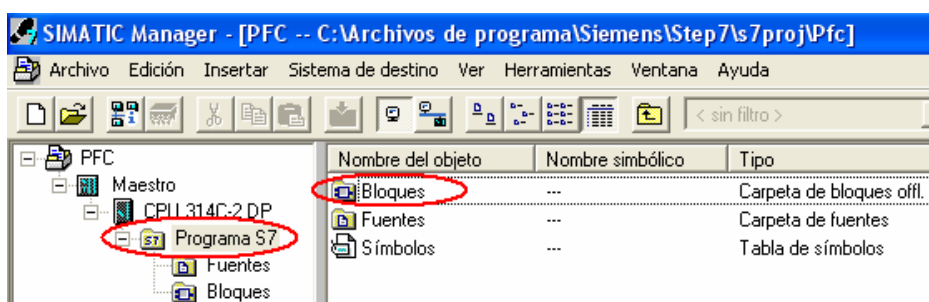


Fig. 5.13. Programa S7, carpeta Bloques

- 2) Una vez dentro, se realiza la acción **Botón derecho ratón >> Insertar nuevo objeto** y se selecciona el tipo de bloque de programa a insertar (Fig. 5.14.).

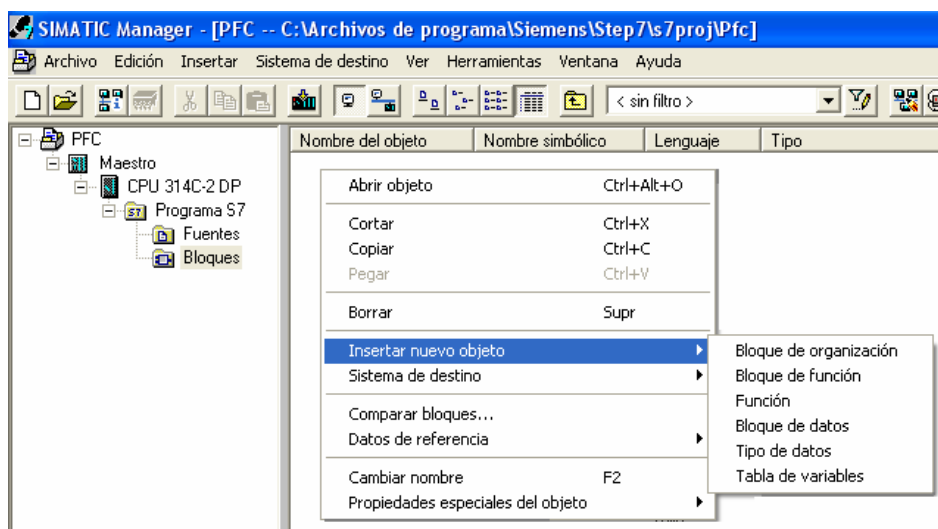


Fig. 5.14. Insertar bloques de programa

- 3) Para editar la programación de un bloque basta con realizar Doble clic sobre el bloque. Se iniciará la herramienta “KOP/AWL/FUP – Programar bloques S7”.

5.5.2.1 Bloques de programa

En la Fig. 5.15. se muestran todos los bloques incluidos en el programa de usuario para el equipo “Maestro”, con un breve comentario de la función que cumplen.

Nombre del objeto	Nombre simbólico	Lenguaje	Tipo	Tamaño...	Usuario	Nombre (encabezado)
Datos de sistema	SDB
DB1	Datos Globales	DB	Bloque de datos	40
DB2	Datos MPI	DB	Bloque de datos	40
DB3	Datos Profibus	DB	Bloque de datos	42
FC1	Tratamiento Profibus	KOP	Función	62
FC2	Tratamiento MPI	KOP	Función	346
OB1	Principal	KOP	Bloque de organización	170
OB80	CYCL_FLT	KOP	Bloque de organización	38
OB82	I/O_FLT1	KOP	Bloque de organización	38
OB85	OBNL_FLT	KOP	Bloque de organización	38
OB86	RACK_FLT	KOP	Bloque de organización	38
OB87	COMM_FLT	KOP	Bloque de organización	38
OB121	PRG_ERR	KOP	Bloque de organización	38
OB122	MOD_ERR	KOP	Bloque de organización	38
SFC67	X_GET	AWL	SFC	...	SIMATIC	X_GET
SFC68	X_PUT	AWL	SFC	...	SIMATIC	X_PUT

Fig. 5.15. Estructura del programa para equipo Maestro

La Tabla 5.2. recoge una breve descripción de los bloques de programa implementados. A lo largo del Capítulo y dependiendo de la función que desempeñen, se describirán más detalladamente.


Nombre del bloque	Nombre simbólico	Comentario
OB1	Principal	Contiene las llamadas a funciones. Se ejecuta de forma cíclica
OB80	CYCL_FLT	OB de error de tiempo (*)
OB82	I/O_FLT1	Alarma de diagnóstico de hardware (*)
OB85	OBNL_FLT	Error de ejecución del programa (*)
OB86	RACK_FLT	Fallo en el bastidor (*)
OB87	COMM_FLT	Error en la comunicación (*)
OB121	PROG_ERR	Error de programación (*)
OB122	MOD_ERR	Error de acceso a la periferia (*)
DB1	Datos Globales	Contiene datos globales que se utilizan en el programa
DB2	Datos MPI	Contiene los datos que se leen / escriben vía MPI
DB3	Datos PROFIBUS	Contiene los datos que se leen / escriben vía PROFIBUS DP
FC1	Tratamiento PROFIBUS	Contiene las operaciones necesarias para la comunicación PROFIBUS DP
FC2	Tratamiento MPI	Contiene las operaciones necesarias para la comunicación MPI
SFC67	X_GET	Función lectura de datos de un interlocutor situado fuera de la estación S7 propia (**)
SFC68	X_PUT	Función escritura de datos en un interlocutor situado fuera de la estación S7 propia (**)

Tabla 5.2. Bloques de programa implementados en “Maestro”

(*) No se encuentra programado pero es necesario para el funcionamiento de la CPU. Razón impuesta por SIEMENS.

(**) Bloque protegido por SIEMENS, por lo que no es posible visualizar su código de programa. Se considera bloque tipo “caja negra”, es decir, en función de unas entradas proporciona unas salidas.

Importante

La relación entre estos bloques se realiza mediante llamadas desde un bloque a otro. A la hora de insertar la llamada a la función debe de seleccionarse el segmento, pulsar el icono “Cuadro vacío”  y escribir el nombre de la función a llamar. La función debe de estar creada previamente.

5.5.2.2 Programación para comunicación PROFIBUS DP

La programación para la comunicación PROFIBUS DP se ha implementado en los bloques de programa OB1 “Principal”, FC1 “Tratamiento PROFIBUS” y DB3 “Datos PROFIBUS”.

OB1 “Principal”

En este bloque de programa se realiza la llamada a la función para tratamiento de datos comunicación PROFIBUS. La condición de llamada es cierta siempre. La Fig. 5.16. muestra esta acción.

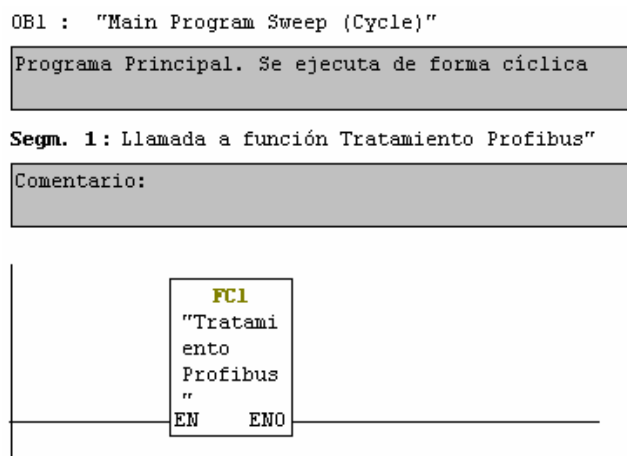


Fig. 5.16. Llamada desde OB1 a FC1

FC1 “Tratamiento PROFIBUS”

En esta función se realiza la adaptación tanto de los datos a enviar vía PROFIBUS DP, así como de los datos que se reciben. Para implementar esta función hay que tener en cuenta:

- Los datos recibidos se almacenan en los registros PEB0....PEB7.
- Los datos enviados se almacenan en los registros PAB0....PAB7.

La localización de los datos viene impuesta por la configuración del módulo EM 277 configurado como Esclavo DP. La Fig. 5.17. representa cómo quedarán los datos recibidos por PROFIBUS tras ser tratados por FC1. Se observa que de 8 bytes en el buzón receptor solamente son considerados los 4 primeros bytes recibidos, ya que son los que contienen los datos provenientes del esclavo.

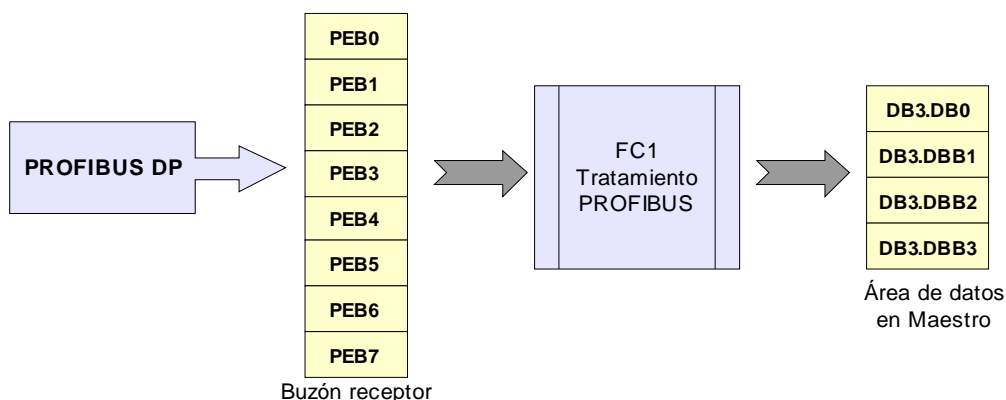


Fig. 5.17. Tratamiento de datos recibidos desde “Esclavo DP”

La localización de los datos a enviar por PROFIBUS viene representada en la Fig. 5.18. Se observa que únicamente son cargados 2 bytes de datos, que se almacenarán en PAB0 y PAB1.

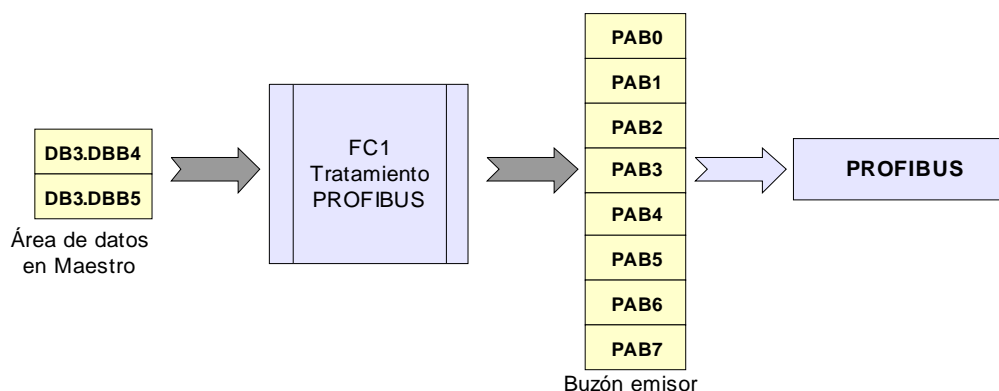


Fig. 5.18. Tratamiento de datos a enviar a “Esclavo DP”

Los segmentos de programación de la función FC1 se muestran en la Fig. 5.19.

- En el Segmento 1 se realiza un movimiento de datos (*Operación MOVE*) de la palabra DBW4 (DBB4+DBB5) perteneciente al DB3 a la palabra PAW0 (PAB0+PAB1). Estos son los datos que se transmiten al “Esclavo DP”.
- En el Segmento 2 se realiza un movimiento de datos (*Operación MOVE*) de la doble palabra PED0 (PEB0+PEB1+PEB3+PEB4) a la doble palabra DBD0 (DBB0+DBB1+DBB2+DBB3) perteneciente al DB3. Estos son los datos que se reciben del “Esclavo DP”.

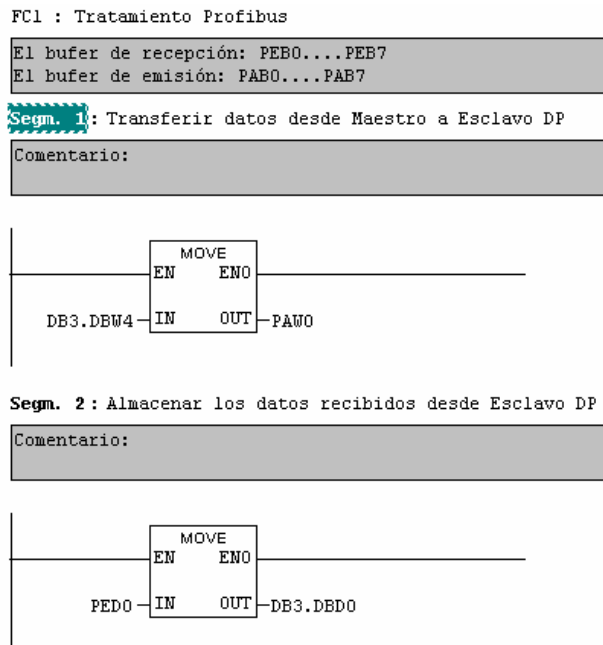


Fig. 5.19. Código de programa para FC1

DB3 “Datos PROFIBUS”

Es el bloque de datos en el cual se almacenan todos los datos que se envían / reciben por PROFIBUS DP. Tiene un tamaño de 6 bytes, de los cuales 4 bytes corresponden a datos de envío (DBB0, DBB1, DBB2, DBB3) y 2 bytes a datos de recepción (DBB4, DBB5). En la Fig. 5.20. está representado el DB3, con un pequeño comentario de la función de cada bit. El término **DCS** (*Digital Control System*) hace referencia a señales procedentes del sistema de control y supervisión (SCADA).

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	E0_0	BOOL	FALSE	Señal de emergencia
+0.1	E0_1	BOOL	FALSE	Interrupcor Manu/Auto
+0.2	E0_2	BOOL	FALSE	Pulsador Marcha
+0.3	E0_3	BOOL	FALSE	Pulsador Preparado
+0.4	E0_4	BOOL	FALSE	Microinterruptor
+0.5	E0_5	BOOL	FALSE	Cilindro expulsor atras
+0.6	E0_6	BOOL	FALSE	Cilindro expulsor delante
+0.7	E0_7	BOOL	FALSE	Unidad lineal izquierda
+1.0	E1_0	BOOL	FALSE	Unidad lineal derecha
+1.1	E1_1	BOOL	FALSE	LIBRE
+1.2	E1_2	BOOL	FALSE	LIBRE
+1.3	E1_3	BOOL	FALSE	LIBRE
+1.4	E1_4	BOOL	FALSE	LIBRE
+1.5	E1_5	BOOL	FALSE	LIBRE
+1.6	E1_6	BOOL	FALSE	LIBRE
+1.7	E1_7	BOOL	FALSE	LIBRE
+2.0	A0_0	BOOL	FALSE	Valvula 5/2 cilindro expulsor
+2.1	A0_1	BOOL	FALSE	Valvula 5/2 unidad lineal izquierda
+2.2	A0_2	BOOL	FALSE	Valvula 5/2 unidad lineal derecha
+2.3	A0_3	BOOL	FALSE	Lampara blanca
+2.4	A0_4	BOOL	FALSE	Lampara verde
+2.5	A0_5	BOOL	FALSE	Lampara amarilla
+2.6	A0_6	BOOL	FALSE	LIBRE
+2.7	A0_7	BOOL	FALSE	LIBRE
+3.0	A1_0	BOOL	FALSE	LIBRE
+3.1	A1_1	BOOL	FALSE	LIBRE
+3.2	A1_2	BOOL	FALSE	LIBRE
+3.3	A1_3	BOOL	FALSE	LIBRE
+3.4	A1_4	BOOL	FALSE	LIBRE
+3.5	A1_5	BOOL	FALSE	LIBRE
+3.6	A1_6	BOOL	FALSE	LIBRE
+3.7	A1_7	BOOL	FALSE	LIBRE
+4.0	DCS_0	BOOL	FALSE	Interrupcor Manu/Auto desde DCS
+4.1	DCS_1	BOOL	FALSE	Pulsador Marcha desde DCS
+4.2	DCS_2	BOOL	FALSE	Pulsador Preparado desde DCS
+4.3	DCS_3	BOOL	FALSE	LIBRE
+4.4	DCS_4	BOOL	FALSE	LIBRE
+4.5	DCS_5	BOOL	FALSE	LIBRE
+4.6	DCS_6	BOOL	FALSE	LIBRE
+4.7	DCS_7	BOOL	FALSE	LIBRE
+5.0	DCS_8	BOOL	FALSE	LIBRE
+5.1	DCS_9	BOOL	FALSE	LIBRE
+5.2	DCS_10	BOOL	FALSE	LIBRE
+5.3	DCS_11	BOOL	FALSE	LIBRE
+5.4	DCS_12	BOOL	FALSE	LIBRE
+5.5	DCS_13	BOOL	FALSE	LIBRE
+5.6	DCS_14	BOOL	FALSE	LIBRE
+5.7	DCS_15	BOOL	FALSE	LIBRE
=6.0		END_STRUC		

Fig. 5.20. Bloque de datos DB3 "Datos PROFIBUS"

5.5.2.3 Programación para comunicación MPI

La programación para la comunicación vía MPI se ha implementado en los bloques de programa OB1 "Principal", FC2 "Tratamiento MPI" y DB2 "Datos MPI". Además se han utilizado dos funciones de sistema, SFC67 "X_GET" y SFC68 "X_PUT".

OB1 "Programa principal"

En este bloque de programa se realiza la llamada a la función para tratamiento de datos comunicación MPI. La condición de llamada es cierta siempre. La Fig. 5.21. representa esta acción.

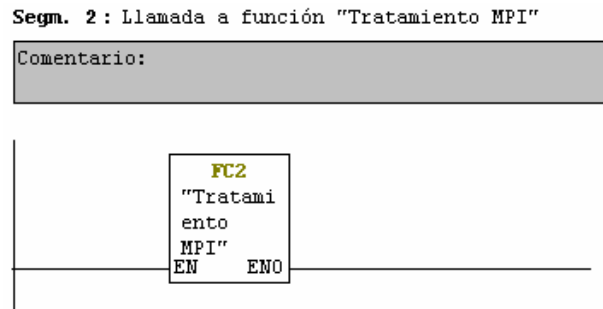


Fig. 5.21. Llamada desde OB1 a FC2

FC2 "Tratamiento MPI"

En esta función se realiza el proceso de comunicación MPI. Se utilizan dos funciones de sistema (SFC) que son llamadas desde esta función. La condición de llamada a estas funciones es siempre cierta. Hay que tener en cuenta que:

- Los datos recibidos se almacenan en el bloque de datos DB2 (DBB0 y DBB1).
- Los datos enviados están almacenados en el bloque datos DB2 (DBB2).

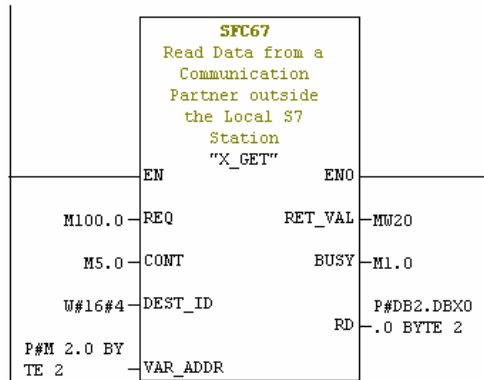
El código de programa para la rutina FC2 queda representado en la Fig. 5.22.

FC2 : Tratamiento MPI

En esta rutina se incluyen los bloques de función X GET (Leer datos de un equipo externo) y X PUT (Escribir datos en un equipo externo). Los datos son leídos y escritos desde la CPU 224 a través de estas funciones y depositados en unas zonas de datos determinadas.

Segm. 1 : Función Leer de Esclavo

Comentario:



Segm. 2 : Función Escribir en Esclavo

Comentario:

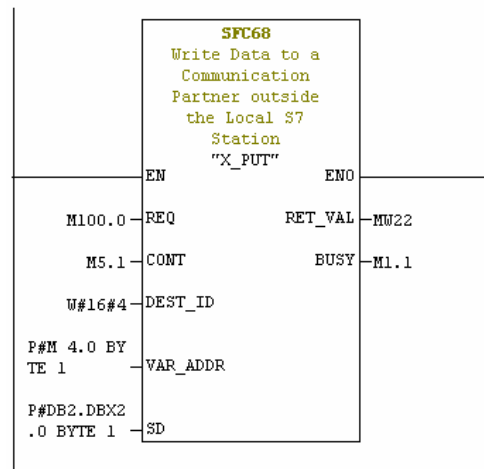


Fig. 5.22. Código de programa para FC2

SFC 67 “X GET”

La función SFC 67 permite leer datos de un interlocutor situado fuera del equipo S7 propio. En el interlocutor (“Esclavo MPI”) no existe ninguna SFC asociada. Es necesario atender que el área de recepción en la CPU “Maestro” (receptora) sea como mínimo tan largo como el área de lectura en la CPU “Esclavo MPI” (interlocutor). Además deberán coincidir los tipos de datos de ambas áreas. Los parámetros que definen a la función SFC 67 se recogen en la Tabla 5.3.

Parámetro	Tipo	Valor	Descripción
REQ	BOOL	M100.0	Con un flanco positivo en REQ se activa la lectura de datos
CONT	BOOL	M5.0	Parámetro de control "continue". Determina si tras realizar la lectura, el enlace permanece (valor 1) o se interrumpe (valor 0)
DEST_ID	WORD	W#16#4	Dirección MPI de la CPU que hay que leer. "Esclavo MPI" = 4
VAR_ADDR	ANY	P#M 2.0 Byte 2	Referencia sobre el área de datos a leer en interlocutor ("Esclavo MPI")
RET_VAL	INT	MW22	Si durante la ejecución de la función aparece un error, incluye su código de error asociado. De lo contrario, incluye la longitud en bytes del paquete de datos copiado
BUSY	BOOL	M1.1	BUSY=1: la recepción aún no ha terminado BUSY=0: la recepción ha terminado o no está activada
SD	ANY	P#DB2.DBX0.0 Byte 2	Referencia sobre el área de recepción en CPU receptora ("Maestro")

Tabla 5.3. Parámetros de la SFC67

SFC 68 "X PUT"

La SFC 68 permite escribir en un interlocutor situado fuera de la estación S7 propia. En el interlocutor no existe ninguna SFC asociada. El área de emisión en la CPU "Maestro" (emisora) debe tener la misma longitud que el área de recepción en la CPU "Esclavo MPI" (interlocutora). Además deberán coincidir los tipos de datos de ambas áreas. Los parámetros que definen a la función SFC 68 se recogen en la Tabla 5.4.

Parámetro	Tipo	Valor	Descripción
REQ	BOOL	M100.0	Con un flanco positivo en REQ se activa la lectura de datos
CONT	BOOL	M5.1	Parámetro de control "continue". Determina si tras realizar la lectura, el enlace permanece (valor 1) o se interrumpe (valor 0)
DEST_ID	WORD	W#16#4	Dirección MPI de la CPU a escribir. "Esclavo MPI" = 4
VAR_ADDR	ANY	P#M 4.0 Byte 1	Referencia sobre el área de datos en la que se desea escribir ("Esclavo MPI")
RET_VAL	INT	MW22	Si durante la ejecución de la función aparece un error, incluye su código de error asociado. De lo contrario, incluye la longitud en bytes del paquete de datos copiado
BUSY	BOOL	M1.1	BUSY=1: la recepción aún no ha terminado BUSY=0: la recepción ha terminado o no está activada
RD	ANY	P#DB2.DBX2.0 Byte 1	Referencia sobre el área que contiene los datos a emitir ("Maestro")

Tabla 5.4. Parámetros de la SFC68

Notas importantes

- Como se ha comentado, para activar la SFC se requiere de un flanco positivo en la entrada REQ. Surge así la necesidad de disponer de un registro que se active / desactive cada cierto tiempo. Esto se ha conseguido mediante la creación de una **Marca de Ciclo**. Una marca de ciclo es una marca que modifica su estado

binario periódicamente con un ciclo de trabajo. Se define como un byte, donde cada bit tiene asignada una frecuencia (Fig. 5.23.).

Bit del byte de la marca de ciclo	7	6	5	4	3	2	1	0
Duración del período (s)	2,0	1,6	1,0	0,8	0,5	0,4	0,2	0,1
Frecuencia (Hz)	0,5	0,625	1	1,25	2	2,5	5	10

Fig. 5.23. Byte marca de ciclo

En esta aplicación se ha definido como marca de ciclo la MB100; y se ha empleado el bit M100.0, que modifica su estado binario cada 0.1 s.

Para definir una marca de ciclo

En la pantalla principal de la aplicación se selecciona equipo CPU 314C-2DP y se realiza la acción *Sistema destino >> Diagnóstico/Configuración >> Información del módulo* (Fig. 5.24.). En la ventana resultante se establece MB100 como marca de ciclo. **Esta operación debe realizarse ONLINE, es decir, con la CPU conectada al PC.**

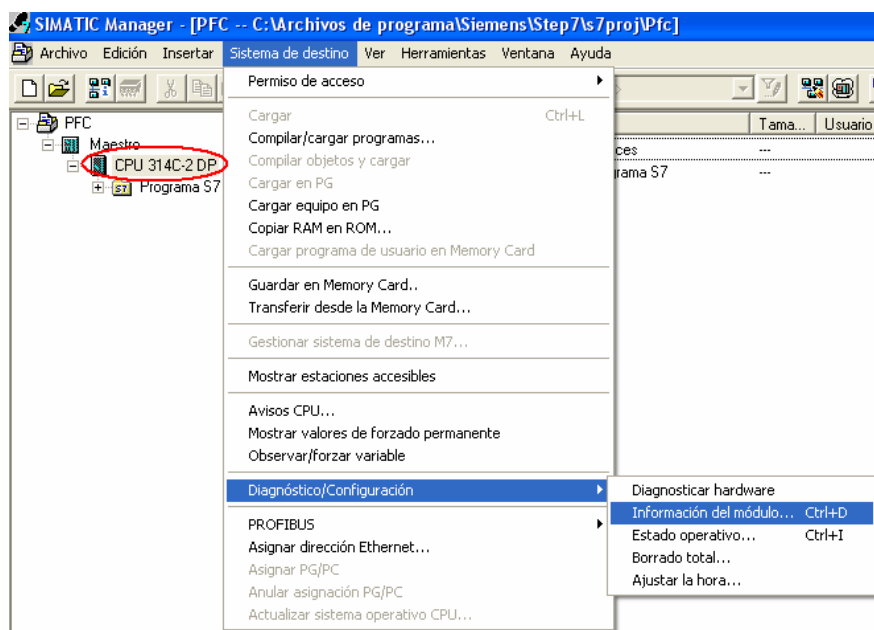


Fig. 5.24. Información CPU314C-2DP

- Los códigos de error asociados a ambas SFCs (registro RET_VAL) están recogidos en el correspondiente manual, *Referencia Bibliográfica [4]*.

DB2 “Datos MPI”

Es el bloque de datos en el cual se almacenan todos los datos que se leen / escriben por MPI. Tiene un tamaño de 3 bytes, de los cuales 2 bytes corresponde a datos

que se leen (DBB0, DBB1) y 1 byte a datos que se escriben (DBB2). En la Fig. 5.25. está representado el DB2, con un pequeño comentario de la función de cada bit. El término **DCS** (*Digital Control System*) hace referencia a señales procedentes del sistema de control y supervisión (SCADA).

Dirección	Nombre	Tipo	Valor inicial	Comentario
0.0		STRUCT		
+0.0	EO_0	BOOL	FALSE	Pulsador Marcha
+0.1	EO_1	BOOL	FALSE	Interruptor Manu/Auto
+0.2	EO_2	BOOL	FALSE	LIBRE
+0.3	EO_3	BOOL	FALSE	Sensor fotoelectrico
+0.4	EO_4	BOOL	FALSE	LIBRE
+0.5	EO_5	BOOL	FALSE	Señal de emergencia
+0.6	EO_6	BOOL	FALSE	Señal encoder
+0.7	EO_7	BOOL	FALSE	LIBRE
+1.0	AO_0	BOOL	FALSE	Lampara verde
+1.1	AO_1	BOOL	FALSE	Lampara blanca
+1.2	AO_2	BOOL	FALSE	Motor derecha
+1.3	AO_3	BOOL	FALSE	Motor izquierda
+1.4	AO_4	BOOL	FALSE	LIBRE
+1.5	AO_5	BOOL	FALSE	LIBRE
+1.6	AO_6	BOOL	FALSE	LIBRE
+1.7	AO_7	BOOL	FALSE	LIBRE
+2.0	DCS0	BOOL	FALSE	Pulsador de Marcha desde DCS
+2.1	DCS1	BOOL	FALSE	Interruptor Manu/Auto desde DCS
+2.2	DCS2	BOOL	FALSE	LIBRE
+2.3	DCS3	BOOL	FALSE	LIBRE
+2.4	DCS4	BOOL	FALSE	LIBRE
+2.5	DCS5	BOOL	FALSE	LIBRE
+2.6	DCS6	BOOL	FALSE	LIBRE
+2.7	DCS7	BOOL	FALSE	LIBRE
=4.0		END_STRUCT		

Fig. 5.25. Bloque de datos DB2 "Datos MPI"

A modo de resumen, se incluye la Fig. 5.26. donde se muestra cómo se realiza el intercambio de datos.

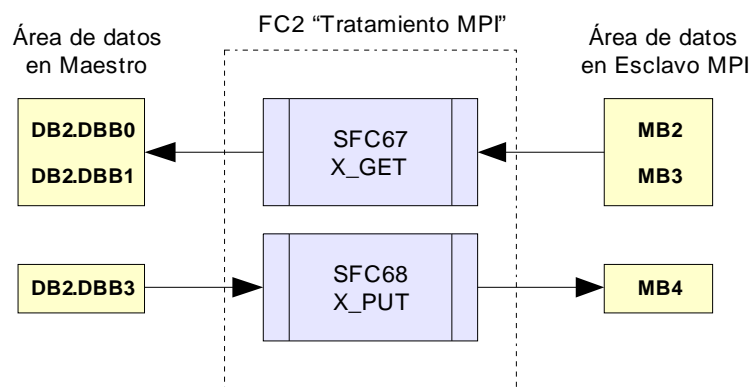


Fig. 5.26. Intercambio de datos vía MPI

5.6 Configuración equipo Esclavo DP

El equipo “Esclavo DP” está formado por una CPU 224 ampliado con un módulo EM 277. El módulo de comunicaciones EM 277 es el que permite a la CPU 224 ser integrada en un bus de comunicación PROFIBUS. Además, a esta CPU se encuentra conectado el MODEM TC35 para comunicación GSM. Por tanto, la configuración del equipo “Esclavo DP” queda dividida en tres partes: por un lado, configurar el módulo de comunicación EM 277 para conexión PROFIBUS; por otro configurar el MODEM TC35 para intercambiar datos con la CPU, y por último configurar la CPU 224 como equipo de control sobre la unidad “*doble alimentador por gravedad*”.

5.6.1 Configuración módulo EM277

El módulo EM 277 no suele estar disponible en el catálogo hardware de la herramienta *HWConfig* (todo depende de la versión de software de la que se disponga; en nuestro caso no estaba disponible). Para insertar dicho módulo en el catálogo es necesario disponer de su **archivo GSD**. Los archivos GSD ofrecen una descripción detallada de las propiedades de un dispositivo en un formato definido exactamente. Estos archivos son preparados para cada tipo de dispositivo por el correspondiente fabricante, poniéndolos a disposición del usuario de equipos PROFIBUS. El archivo GSD permite que el sistema de configuración lea las propiedades de un dispositivo PROFIBUS y utilice dichas informaciones al configurar la red.

Por otra parte, para configurar el módulo con calidad de esclavo DP es preciso ajustar la dirección de estación del puerto DP para que coincida con la dirección fijada en la configuración del “Maestro”. La dirección de estación se ajusta con los interruptores rotativos del módulo. El interruptor superior (serigrafiado con “X10”) marca la decena de la dirección, mientras que el inferior (serigrafiado con “X1”) marca la unidad (Fig. 5.27.)



Fig. 5.27. Ajuste de dirección PROFIBUS

En la presente aplicación se le ha asignado la **dirección PROFIBUS 2**. Tras haber efectuado un cambio con un interruptor es preciso desconectar la CPU y conectarla de nuevo para poder adoptar la nueva dirección del esclavo.

En la aplicación del presente proyecto, el equipo “Maestro” intercambia datos con el equipo “Esclavo DP”. El “Maestro” envía informaciones al búfer de salida del “Esclavo DP” o **buzón receptor**. El “Esclavo DP” responde al mensaje del “Maestro” retornando un búfer de entrada o **buzón emisor** que el “Maestro” almacena en un área de entradas. El tamaño de estos buzones de datos lo determina la configuración del módulo EM 277. La Fig. 5.28. representa de forma general como se lleva a cabo la comunicación.

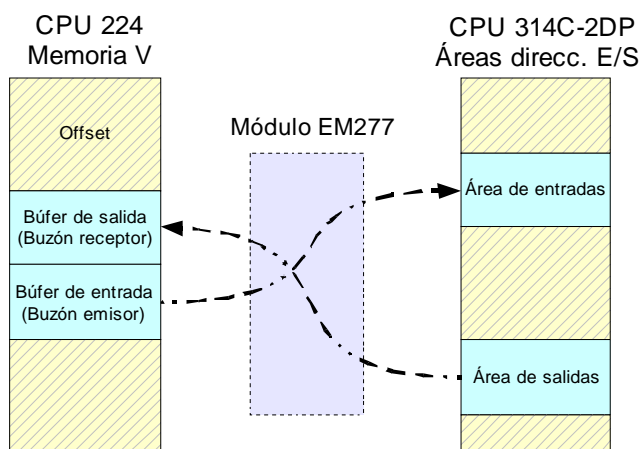


Fig. 5.28. Tarea de comunicación

Importar archivo GSD

El archivo GSD para el EM 277 se llama “**SIEM089D.GSD**”. Este archivo está disponible en Internet, *Referencia Bibliográfica [11]*.

Una vez descargado el archivo GSD hay que incluirlo en el catálogo hardware de la aplicación *HWConfig*. El proceso a seguir es el siguiente:

- 1) De la barra Menú, se elige la opción *Herramientas >> Instalar archivos GSD* (Fig. 5.29.).

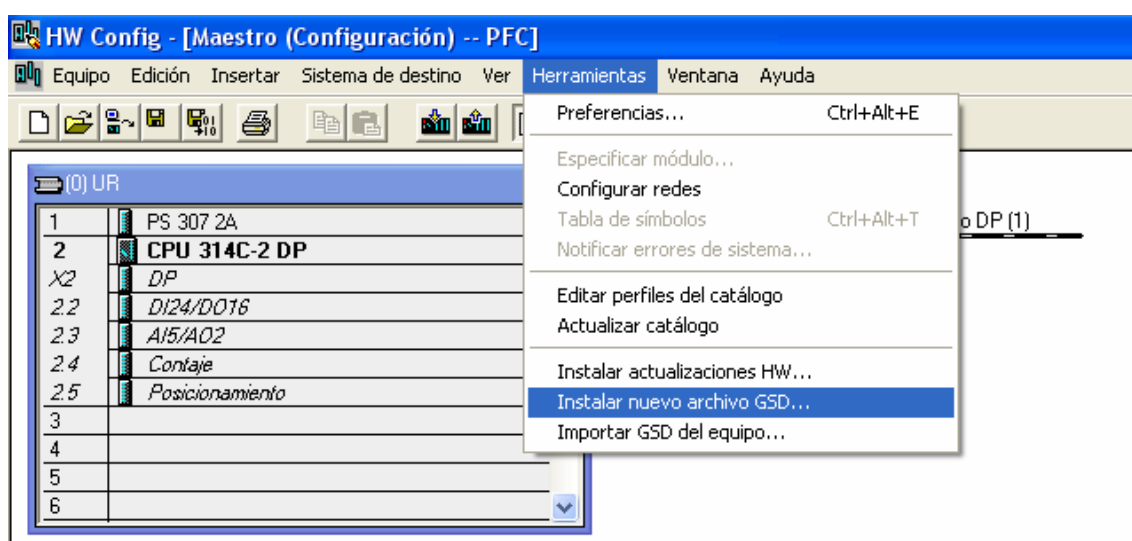


Fig. 5.29. Instalar archivo GSD

- 2) En el siguiente cuadro de diálogo se debe localizar dónde está archivado el archivo para abrirlo.
- 3) A continuación se selecciona la opción *Herramientas >> Actualizar catálogo* (Fig. 5.30.).

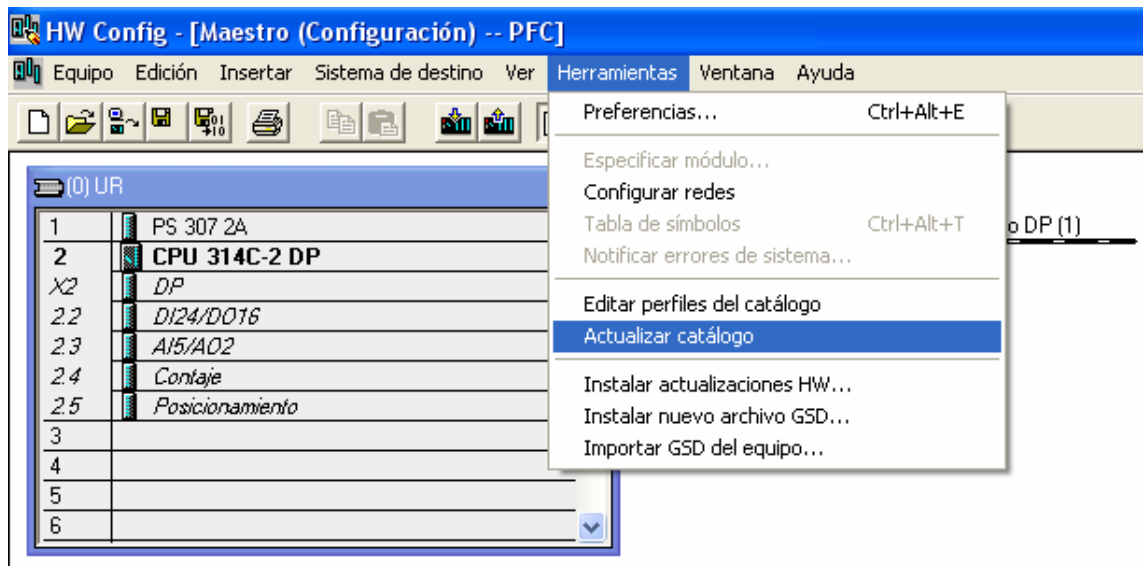


Fig. 5.30. Actualizar catálogo

- 4) Finalmente, el nuevo equipo se puede encontrar en el catálogo bajo *PROFIBUS DP >> Otros aparatos de campo >> PLC >> SIMATIC* (Fig. 5.31).

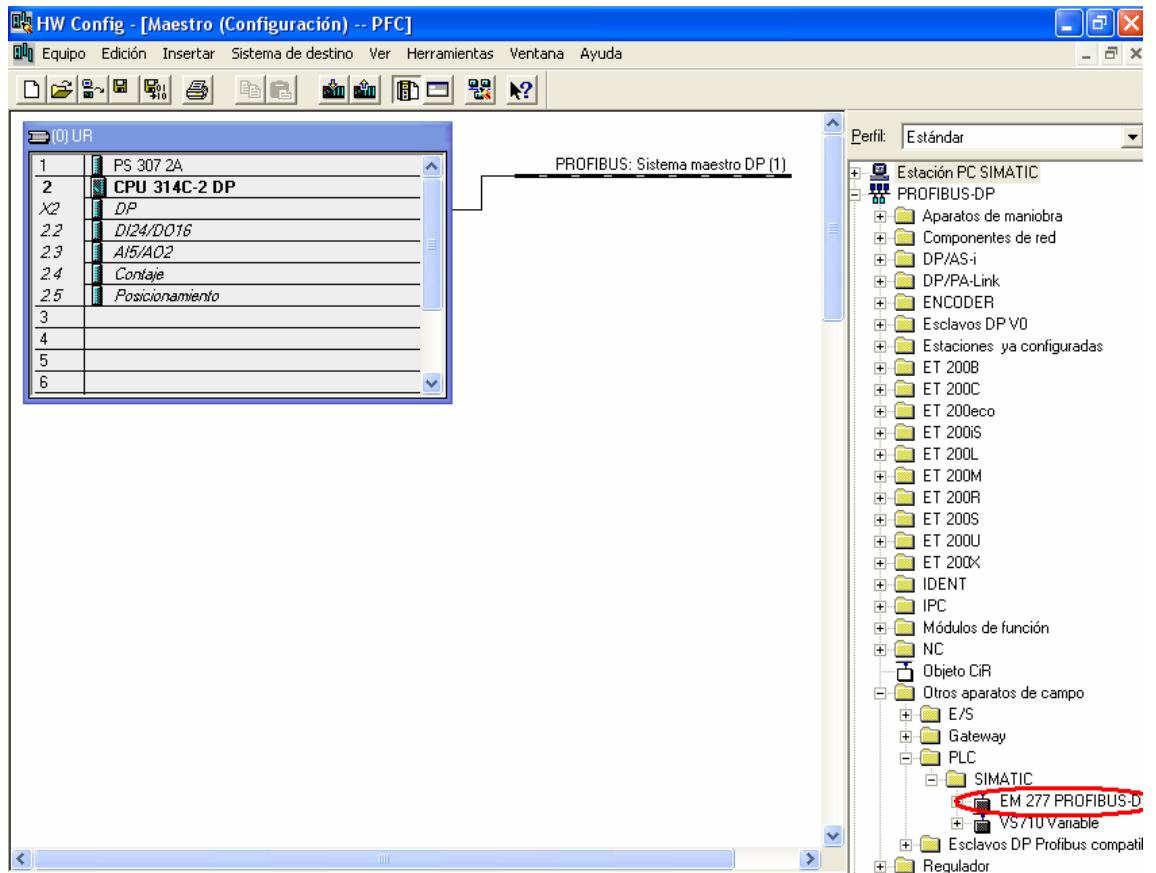


Fig. 5.31. EM 277 insertado en catálogo

Configurar módulo

El siguiente paso es insertar el módulo como esclavo DP en la aplicación y configurar los parámetros de comunicación con el equipo “Maestro”. El proceso a seguir es el que se describe en los siguientes puntos:

- 1) Se inserta el módulo EM 277 en la aplicación: Catálogo hardware >> PROFIBUS DP >> Otros aparatos de campo >> PLC >> SIMATIC >> EM277 PROFIBUS DP. El módulo debe ser arrastrado hacia la línea que marca el bus PROFIBUS. Al arrastrar el equipo se muestra la pantalla de la Fig. 5.32., en la que se pide asignar una dirección PROFIBUS. Para “Esclavo DP” se asigna la **dirección PROFIBUS 2**.

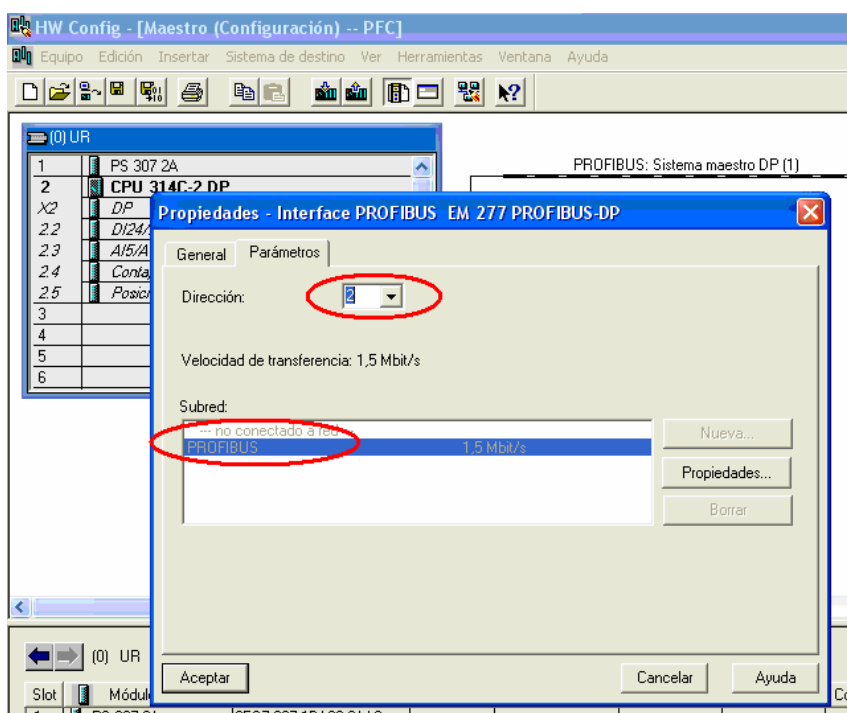


Fig.5.32. Propiedades del interface módulo EM277

Al aceptar, la pantalla resultante es la mostrada en la Fig. 5.33.

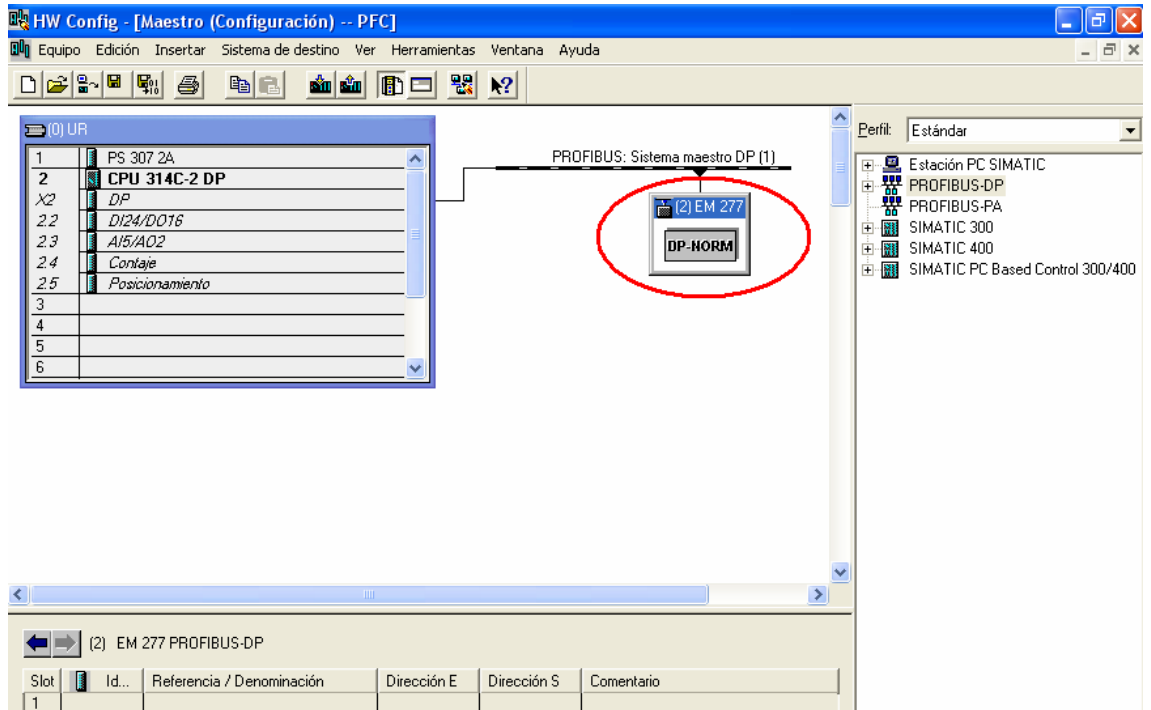


Fig. 5.33. EM 277 insertado en aplicación

- 2) Desplegando el menú para EM 277 se observan los tamaños de búfer disponibles, representados en la Fig. 5.34.

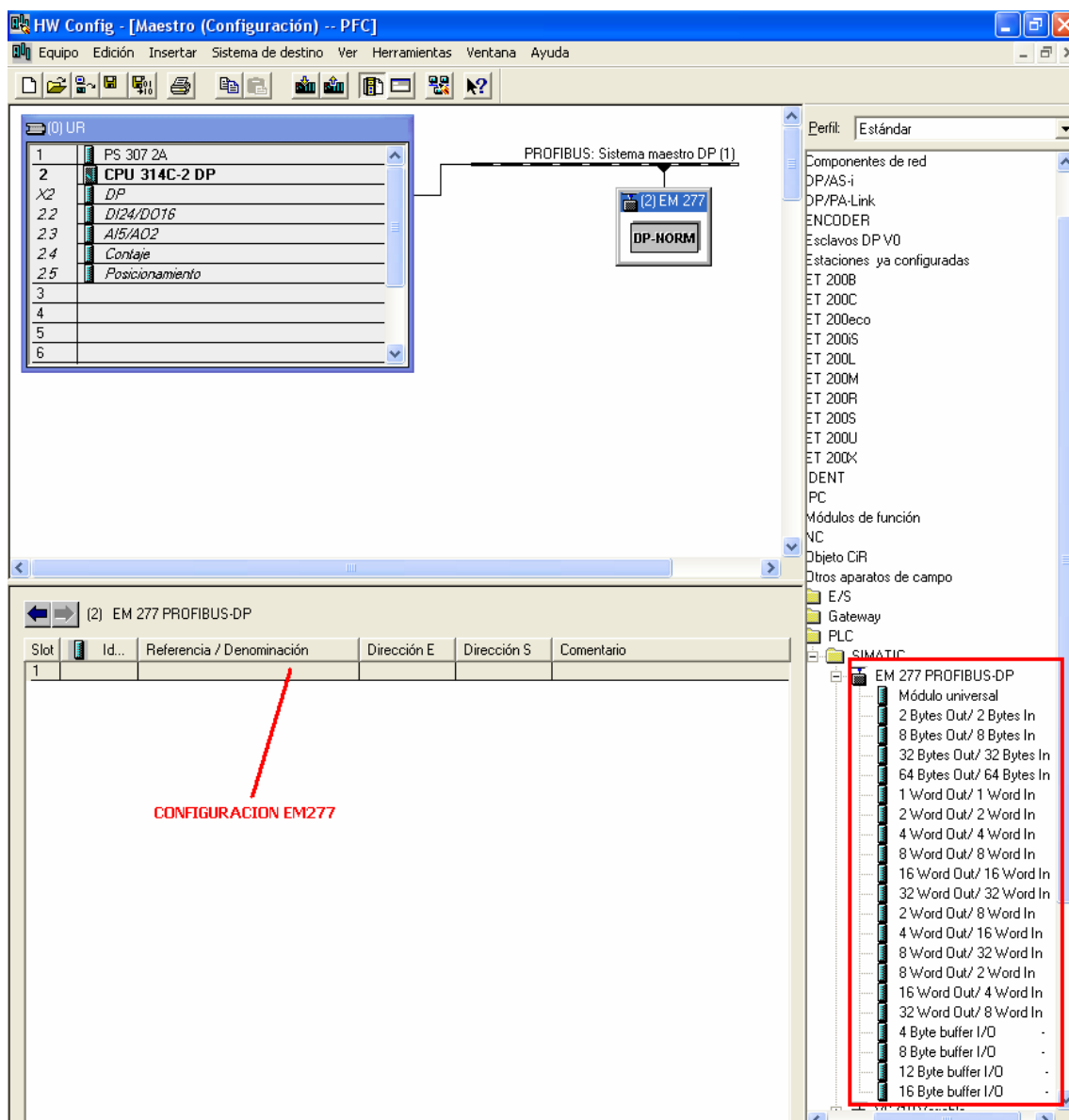


Fig. 5.34. Tamaños de búfer disponibles para EM277.

En el caso de la presente aplicación, se optará por una configuración **8 bytes Out / 8 bytes In**. Aunque para la cantidad de datos que van a tratarse no es necesario un tamaño de búfer de estas características, se ha optado por esta configuración previniendo posibles ampliaciones del sistema.

- 3) Se inserta el módulo deseado: Catálogo hardware >> PROFIBUS DP >> Otros aparatos de campo >> PLC >> SIMATIC >> EM277 PROFIBUS DP >> 8 bytes Out/8 bytes In. Se selecciona el módulo y se arrastra al Slot 1 situado en la ventana inferior del equipo EM 277 (Fig. 5.34.). La Fig. 5.35. muestra la ventana resultante.

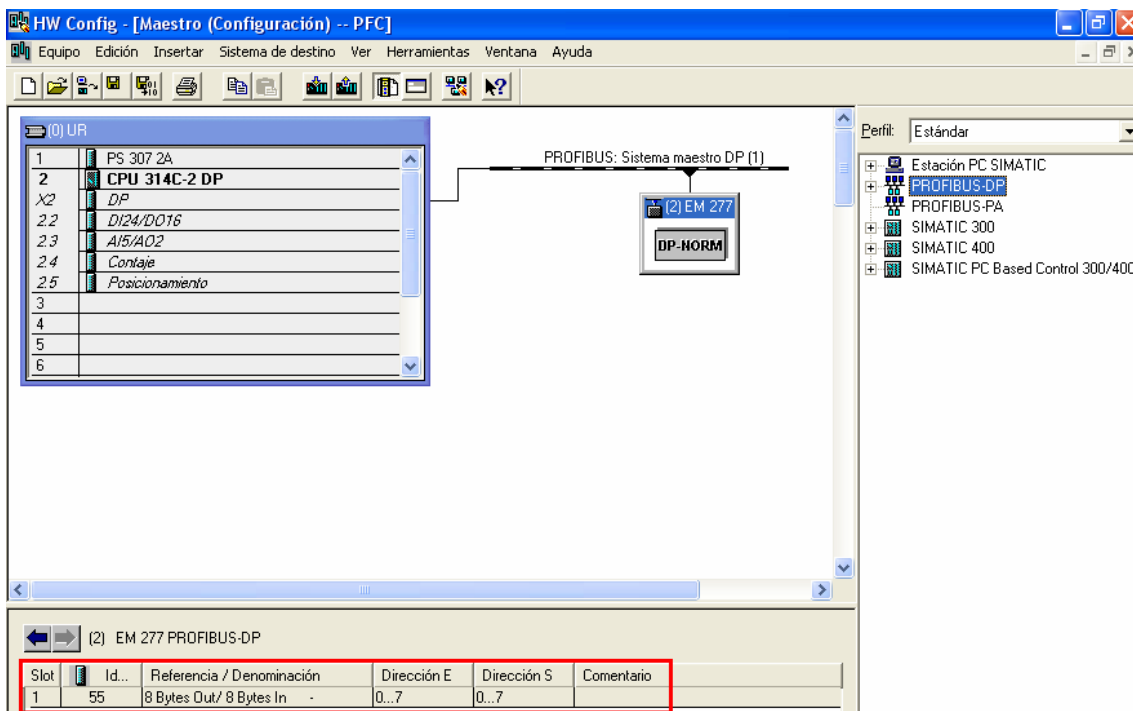


Fig. 5.35. Configuración tamaño de búfer módulo EM277

Como se observa en la Fig. 5.35., el módulo se ha configurado con 8 bytes de datos tanto para el buzón emisor como para el buzón receptor. Las direcciones que se muestran en la figura son asignadas de forma automática por programa y corresponden con las áreas de datos en la CPU 314C-2DP. Es decir, los datos que el módulo EM 277 (“Esclavo DP”) envíe a la CPU 314C-2DP (“Maestro”) quedarán almacenados en los bytes de entradas 0...7 (**PEB0....PEB7**); y los datos que el “Maestro” envíe al “Esclavo DP” serán los contenidos en los bytes de salidas 0...7 (**PAB0....PAB7**).

- 4) Se configura el **Offset** del módulo. Este parámetro determinará el inicio del buzón receptor en la CPU 224, que tendrá un tamaño de 8 bytes. Los siguientes 8 bytes corresponderán al buzón emisor en la CPU 224. El parámetro Offset tendrá un valor de 100. Se opta por el valor 100 como se podría haber elegido otro cualquiera, siempre y cuando se respeten los límites marcados por la memoria de variables de la CPU 224, que es de una extensión de 5119 bytes. Para editar dicho parámetro: Clic botón derecho del ratón sobre icono módulo EM277 >> Propiedades del objeto. En la ventana resultante (Fig. 5.36.) se selecciona la pestaña Parametrizar >> Parámetros del equipo >> Parámetros específicos del aparato >> I/O Offset in the V-memory. Aquí se introduce el valor decimal 100.

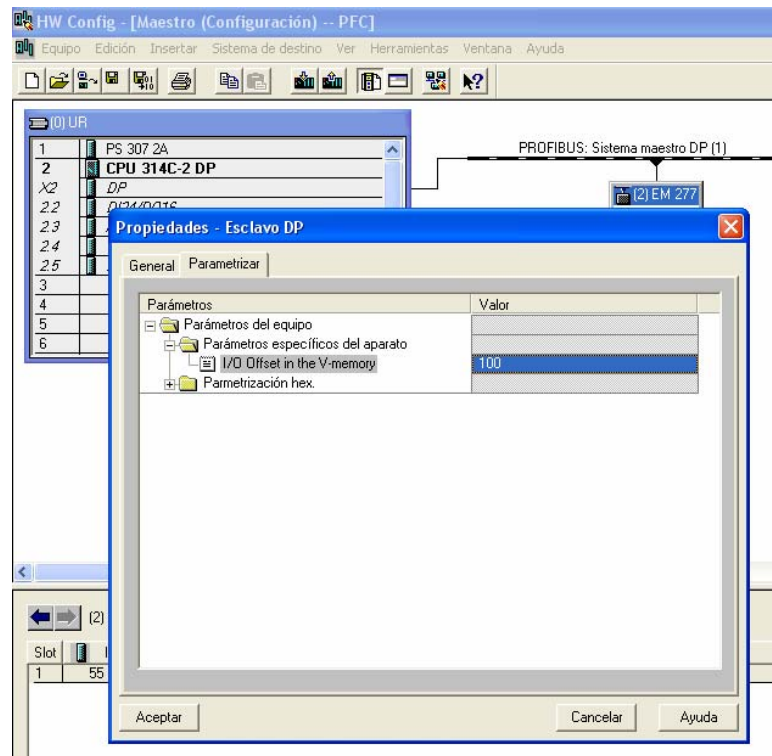


Fig. 5.36. Offset módulo EM277

Finalmente la comunicación entre “Maestro” y “Esclavo DP” queda configurada conforme a la Fig. 5.37. En dicha figura se observa:

- En “Esclavo DP”: un Offset de valor 100 (VB0....VB99), buzón receptor (VB100....VB107), buzón emisor (VB108....VB115).
- En “Maestro”: área de entradas (PEB0....PEB7), área de salidas (PAB0....PAB7).

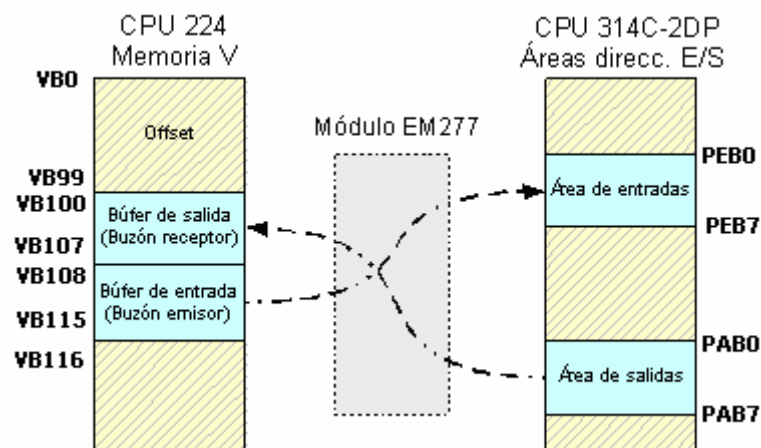


Fig. 5.37. Distribución de datos en la tarea de comunicación

Una vez que todo lo anterior se configura correctamente, ambos equipos pasan al modo intercambio de datos. El equipo “Maestro” escribe los datos de salida en el módulo EM 277 y éste último responde con los datos de entrada de la CPU 224 más recientes. El módulo EM 277 actualiza continuamente sus entradas de la CPU 224 para facilitarle los datos de entrada más recientes al “Maestro”. El módulo transmite entonces los datos de salida a la CPU 224. Los datos de salida del “Maestro” se depositan en la memoria V (buzón receptor) que comienza en la dirección indicada por el “Maestro” durante la inicialización. Los datos de entrada se leen de las direcciones de la memoria V (buzón emisor) que le siguen inmediatamente a los datos del buzón receptor.

Los datos de salida que envíe el “Maestro” se depositan en la memoria V (buzón receptor) inmediatamente después de haberse ejecutado la correspondiente parte del ciclo del programa de usuario. Los datos de entrada (dirigidos al “Maestro”) se copian de la memoria V (buzón emisor) en el EM 277 para transmitirlos simultáneamente al “Maestro”. Los datos de salida sólo se escriben en la memoria V cuando el “Maestro” suministra nuevos datos. Los datos de entrada se transmiten al “Maestro” en el siguiente intercambio de datos con él.

5.6.2 Configuración MODEM TC35

El MODEM TC35 necesita ser configurado con unos determinados parámetros para realizar con éxito el intercambio de datos con la CPU. Este proceso de configuración se lleva a cabo con la aplicación *HyperTerminal* (programa que permite conectar al PC otros equipos a través del puerto serie COM), integrada en el sistema operativo Windows y gracias a los **comandos AT**. Dichos comandos son instrucciones codificadas que conforman un lenguaje de comunicación entre el hombre y un Terminal módem. Los comandos AT que se utilizan para la configuración quedan resumidos en la Tabla 5.5.

Comando	Función
AT+IPR	Ajustar velocidad de comunicación
AT+CNMI	Ajustar propiedades de visualización para mensajes recibidos
AT+CSCA	Número del centro de mensajes cortos del operador (SMSC)
AT+CMGF	Formato del mensaje corto
ATE0	Quitar ECO
AT&W	Almacenar perfil

Tabla 5.5. Comandos AT para configuración MODEM

La aplicación *HyperTerminal* se inicia en la siguiente ruta: Inicio >> Programas >> Accesorios >> Comunicaciones >> HyperTerminal. Los pasos a seguir para la configuración son los siguientes:

- 1) Se conecta el MODEM TC35 a un puerto COM del PC mediante un cable serie RS 232 convencional.
- 2) Se inicia la aplicación HyperTerminal y se selecciona Direct COMx, siendo “x” el puerto al cual se encuentra conectado el MODEM. Los ajustes para esta conexión se muestran en la Fig. 5.38.

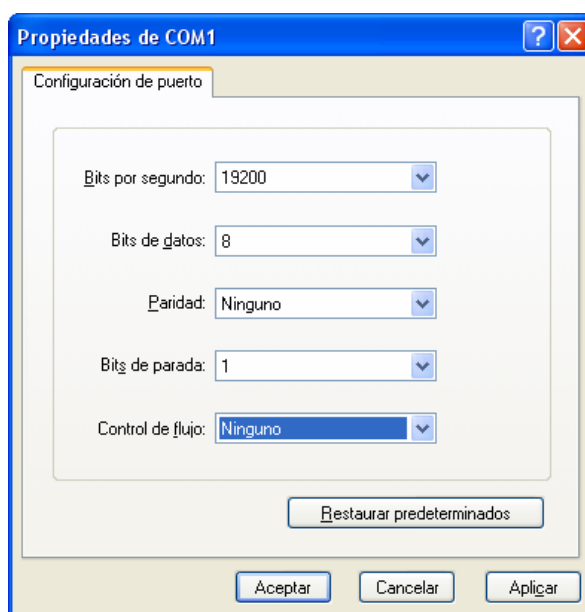


Fig. 5.38. Propiedades de COM para comunicación con MODEM

- 3) Se introduce AT+IPR=9600 (ENTER). Con este comando cambiamos velocidad de transferencia a 9600 bit/s. Debemos ajustar esta velocidad ya que es la máxima velocidad soportada por el cable de comunicación PC/PPI.
- 4) Se modifica la velocidad de transferencia en Propiedades de COM >> 9600 bit/s.
- 5) Se reinicia la conexión (desconectar/conectar).
- 6) Se introduce AT+CNMI=0,0,0,0 (ENTER). Introduciendo estos parámetros se configura el MODEM de la siguiente forma:
 - Enviar mensajes a la CPU sin necesidad de que ésta los solicite.
 - No es necesario recibir confirmación de envío por parte de la CPU.
- 7) Se introduce AT+CSCA="+34656000311" (ENTER). Con esta orden se introduce al MODEM el número del centro de mensajes del operador móvil, en nuestro caso **Orange**.

- 8) Se introduce AT+CMGF=1 (ENTER). Todos los mensajes se encuentran en modo texto.
- 9) Se introduce ATE0 (ENTER). Comando para anular el ECO. A partir de este momento no se mostrará lo que se escribe en pantalla.
- 10) Se introduce AT&W (ENTER). Esta orden grabará todos los parámetros como perfil de usuario.

Importante

- A partir de este momento, el MODEM se encuentra adecuadamente programado, a excepción del número PIN. Al arrancar el MODEM, el led de estado parpadeará a intervalos de 2 ms, indicando que no dispone del número PIN de la tarjeta SIM. La CPU conectada al MODEM será la encargada de proporcionar el número PIN. En el Apartado 5.6.3.3. se describe cómo se realiza esta operación.
- Para que la comunicación se lleve a cabo adecuadamente, la tarjeta SIM debe de estar “*limpia de datos*”, es decir, deberemos borrar todos los números y todos los mensajes almacenados en la memoria de la misma.

5.6.3 Configuración CPU 224

En este apartado se describirá la configuración necesaria para la CPU 224, que junto con el módulo EM 277, conforma el equipo “Esclavo DP”. Además, no hay que olvidar que esta CPU es la encargada de intercambiar datos con el MODEM TC35 para la comunicación GSM. Por tanto, se establecen tres “bloques” de programación: un primer bloque encargado del control de la módulo “*doble alimentador por gravedad*”, un segundo bloque encargado de la comunicación PROFIBUS, y un tercero dedicado a la comunicación GSM.

De acuerdo con todo lo anterior, el programa de usuario de esta CPU se ha estructurado como a continuación se indica:

- Programa Principal (OB1): recoge todas las llamadas a los distintos subprogramas o *subrutinas* y procesa la recepción/envío de mensajes SMS.
- Comunicación PROFIBUS (SBR0): subrutina que procesa el envío/recepción de datos vía PROFIBUS DP.
- Automático (SBR1): controla el funcionamiento del módulo de forma automática (ciclos continuos de trabajo).
- Manual (SBR2): esta subrutina permite forzar cada una de las salidas de forma manual.

- Inicializa MODEM (SBR3): subrutina que inicializa el MODEM TC35 para iniciar comunicación GSM.
- Manda Mensaje (SBR4): subrutina para enviar un SMS.
- Contenido del mensaje (SBR5): determina el contenido del mensaje a enviar.
- Borrar mensaje (SBR6): borra los mensajes recibidos y almacenados en la tarjeta SIM de forma automática.
- Final de RCV (INT0): establece si el mensaje recibido es el adecuado, y en su caso, realiza la acción correspondiente.
- Final de XMT (INT1): cuando el mensaje se ha transmitido se resetean las condiciones de enviar mensaje.

5.6.3.1 Programación para el control del módulo

La programación para el control del módulo “*doble alimentador por gravedad*” está contenida en las subrutinas “Automático” y “Manual”. Este código de programa se ha obtenido a partir de la **Referencia Bibliográfica [1]**, aunque se han realizado una serie de modificaciones sobre esta programación.

Antes de comenzar con la descripción del código de programa correspondiente, es necesario comentar que se ha creado una tabla de símbolos (Fig. 5.39.) en la cual se asignan registros del programa a nombres simbólicos, para así facilitar la comprensión del código.


			Símbolo	Dirección
1			Marcha	E0.2
2			Preparado	E0.3
3			Derecha	E1.0
4			Izquierda	E0.7
5			Valvula_dch	A0.1
6			Valvula_izq	A0.2
7			Detras	E0.5
8			Delante	E0.6
9			Valvula_mono	A0.0
10			Verde	A0.4
11			Amarilla	A0.5
12			Emergencia	E0.0
13			Electromecani	E0.4
14			ManAut	E0.1
15			Blanca	A0.3

Fig. 5.39. Tabla de símbolos módulo doble alimentador

Según se seleccione modo de funcionamiento Automático o modo Manual, la CPU ejecutará la subrutina correspondiente. Las llamadas a estas subrutinas se

encuentran en el Programa Principal. Las condiciones que determinan un modo de funcionamiento u otro se detallan a continuación.

- Para “Automático” (Fig. 5.40.). El interruptor Manual / Automático del panel de control local está en posición Automático (*contacto ManAut*), el interruptor Manual / Automático del panel de control remoto está en Automático (*contacto M4.0*) y no disparo de seta de emergencia (*contacto Emergencia*).

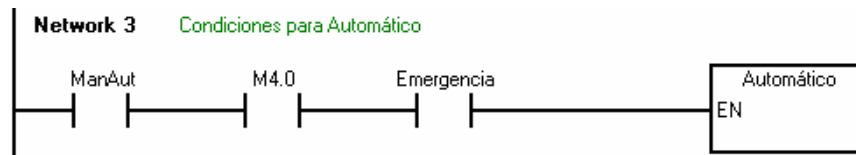


Fig. 5.40. Condiciones para modo Automático

- Para “Manual” (Fig. 5.41.). El interruptor Manual / Automático del panel de control local está en posición Manual (*contacto ManAut*), el interruptor Manual / Automático del panel de control remoto está en Manual (*contacto M4.0*) y no disparo de seta de emergencia (*contacto Emergencia*).

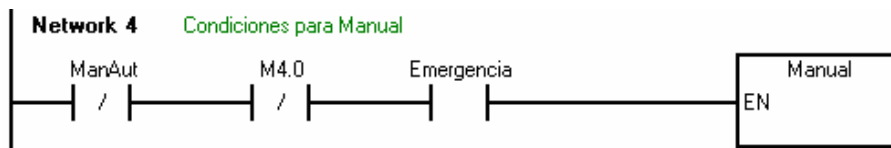
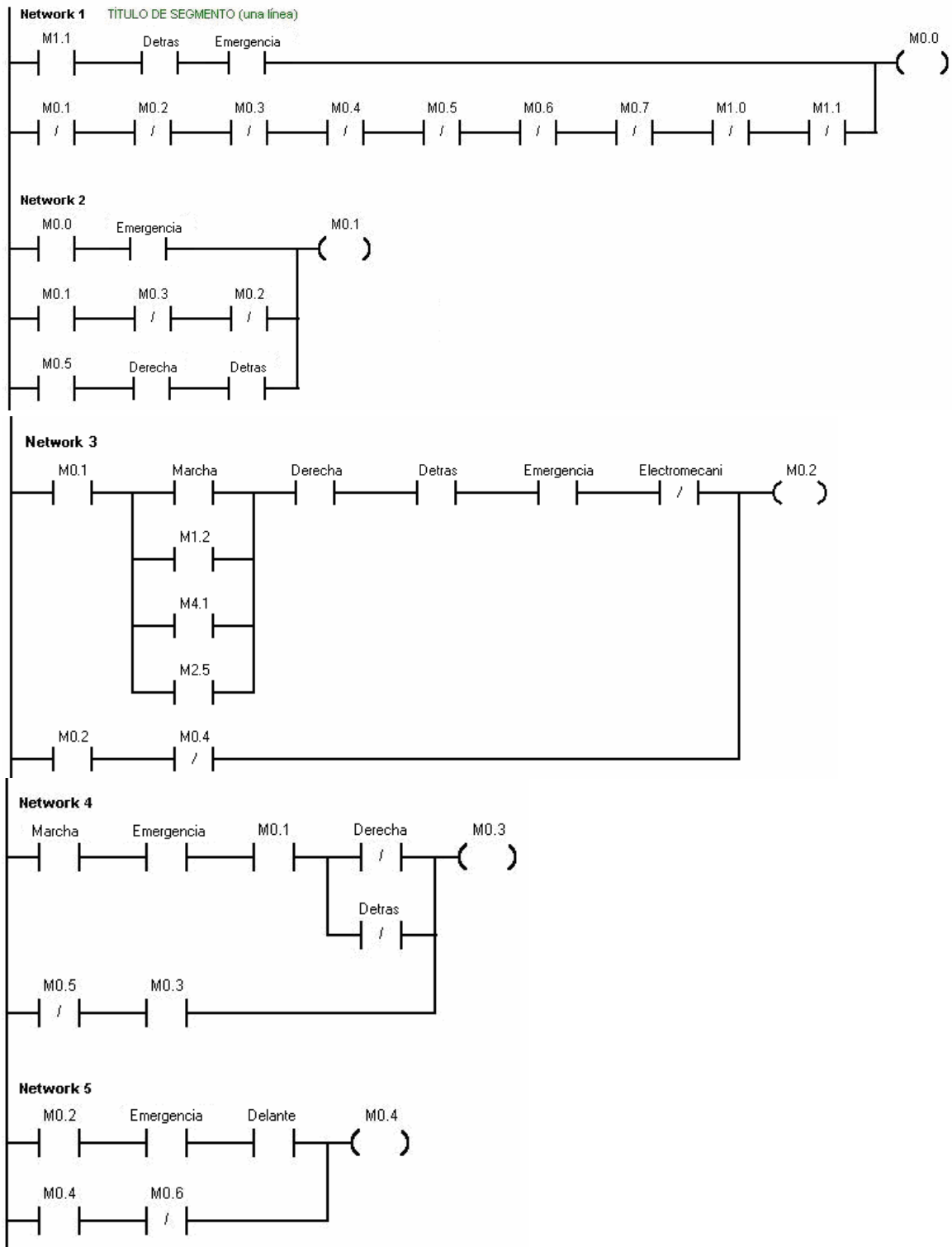
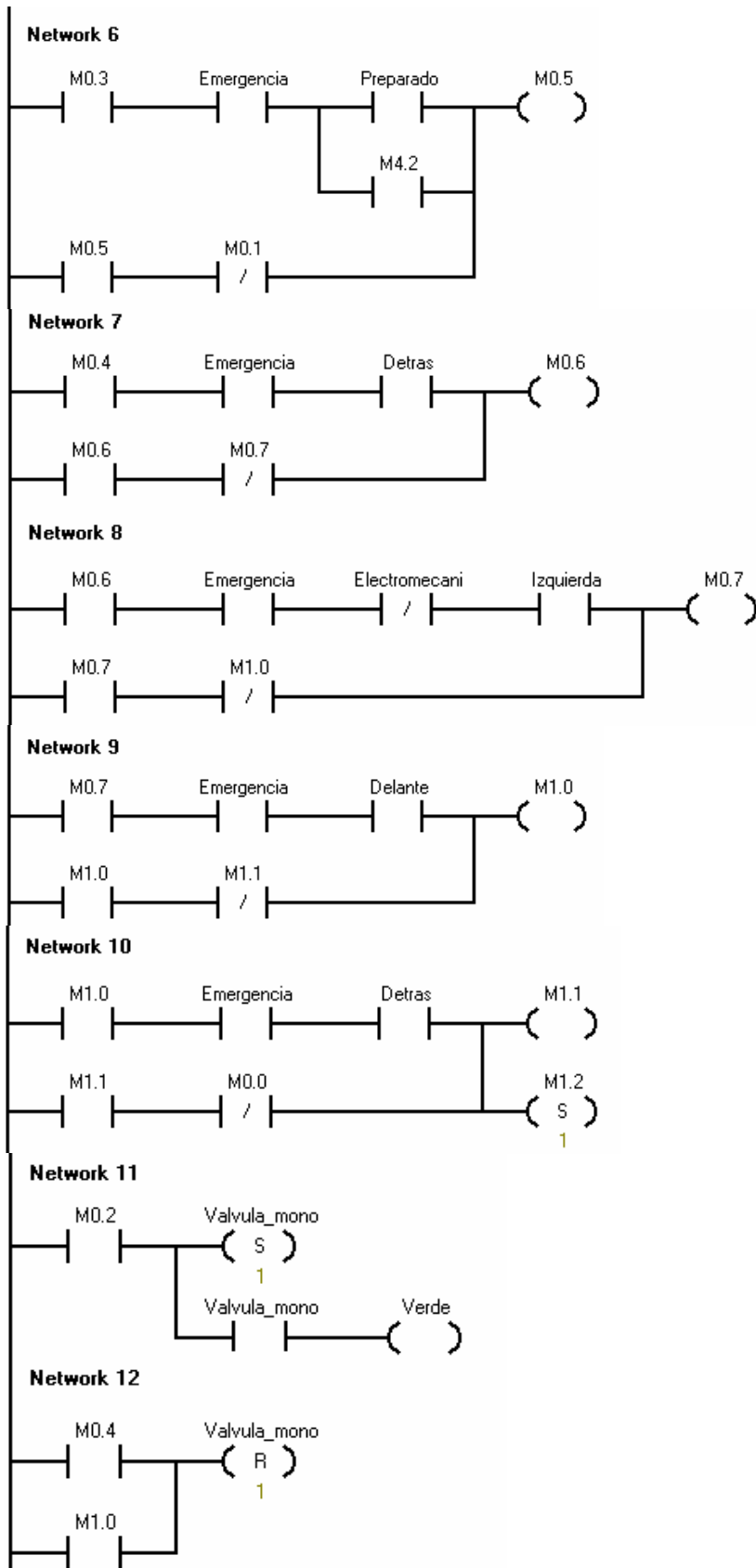


Fig. 5.41. Condiciones para modo Manual

Subrutina Automático

En modo de funcionamiento Automático el módulo realizará ciclos continuos de extracción de piezas. En la Fig. 5.42. se muestra el código de programa correspondiente a esta subrutina.





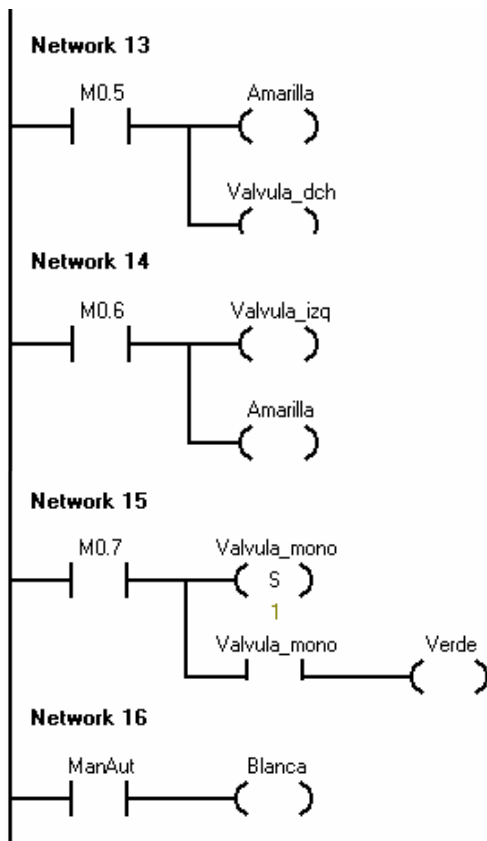


Fig. 5.42. Código de programa para subrutina Automático

Se han utilizado una serie de marcas de sistema que determinan etapas de funcionamiento en el módulo:

- M0.0. Etapa inicial de reposo.
- M0.1. Marca intermedia, no tiene asignada ninguna etapa.
- M0.2. Set cilindro expulsor y activación lámpara verde.
- M0.3. Marca intermedia, no tiene asignada ninguna etapa.
- M0.4. Reset cilindro expulsor.
- M0.5. Orden torres de alimentación derecha y activación lámpara amarilla.
- M0.6. Orden torres de alimentación izquierda y activación lámpara amarilla.
- M0.7. Set cilindro expulsor y activación lámpara verde.
- M1.0. Reset cilindro expulsor.
- M1.1. Marca intermedia, no tiene asignada ninguna etapa.
- M1.2. Señal ciclo terminado.

Junto a las marcas anteriormente comentadas, se utilizan otra serie de registros que completan la funcionalidad del módulo.

- La M4.1 se activa al pulsar Marcha en panel de control SCADA.
- La M4.2 se activa al pulsar Preparado en el panel de control SCADA.
- La M2.5 se activa al enviar el SMS “ONM” desde el terminal móvil vía GSM.
- La M1.2 se activa cuando se completa un ciclo de extracción y sirve para no tener que activar de nuevo la condición de Marcha y realizar ciclos continuos de trabajo.
- Si unidad en Automático, se activa lámpara blanca.

Subrutina Manual

En modo de funcionamiento Manual el módulo podrá controlarse de forma independiente al ciclo de extracción de piezas. En la Fig. 5.43. se muestra el código de programa correspondiente a esta subrutina.

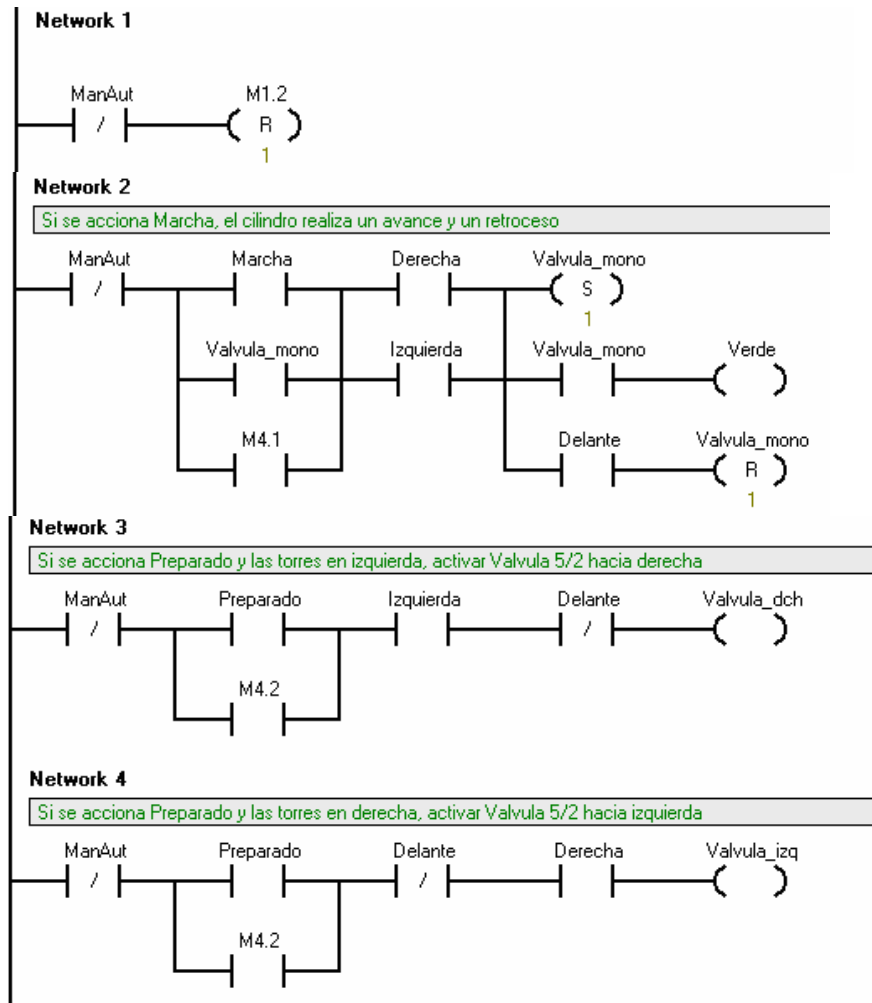




Fig. 5.43. Código de programa para subrutina Manual

La funcionalidad del módulo en modo Manual se ha implementado de la siguiente forma:

- Se anula la M1.2, señal para realizar ciclos continuos de trabajo (Segmento 1).
- Se establecen las condiciones de avance / retroceso del cilindro expulsor. Si se pulsa Marcha en panel de control local ó se pulsa Marcha (*contacto M4.1*) en panel de control SCADA y las torres de alimentación están posicionadas a derecha o izquierda. El avance del cilindro se señala con la lámpara verde (Segmento 2).
- Se establecen las condiciones de movimiento de las torres de alimentación. Si se acciona Preparado panel de control local o en panel de control SCADA (*contacto M4.2*) y las torres en izquierda, activar Válvula 5/2 hacia derecha; y si las torres en derecha, activar Válvula 5/2 hacia izquierda (Segmentos 3 y 4).
- Se establece que si las torres de alimentación están en movimiento, se ilumina la lámpara amarilla (Segmento 5).

5.6.3.2 Programación para comunicación PROFIBUS DP

La programación para el control de la comunicación PROFIBUS DP se procesa en la subrutina “Comunicación PROFIBUS”. Esta subrutina se ha creado a partir de un programa ejemplo incluido en la **Referencia Bibliográfica [3]**.

Previamente a la llamada de esta subrutina, en el “Programa principal”, se realiza una adaptación de los datos que se transmiten por PROFIBUS. Esta adaptación consiste en transferir dichos datos (entradas/salidas) a unos bytes de Marcas. El motivo de esta adaptación reside en que no es conveniente trabajar directamente con entradas y salidas del sistema. La operación puede verse en la Fig. 5.44.

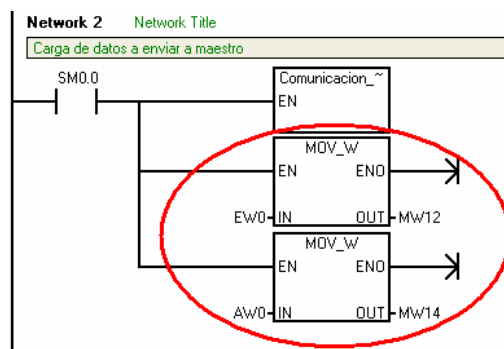


Fig. 5.44. Adaptación de los datos a transmitir por PROFIBUS.

Tras las operaciones “Transferir palabra” (MOV_W) mostradas en la Fig. 5.39., los datos quedan distribuidos de la siguiente forma:

- El byte de entradas EB0 queda almacenado en MB12.
- El byte de entradas EB1 queda almacenado en MB13.
- El byte de salidas AB0 queda almacenado en MB14.
- El byte de salidas AB1 queda almacenado en MB15.

Subrutina Comunicación PROFIBUS

Este programa utiliza la información de configuración del puerto DP almacenada en la memoria de marcas especiales de la CPU. Dichos datos proporcionan la configuración del esclavo DP al programa y están recogidos en la Tabla 5.6.

Área de memoria	Función
SMW220	Estado de error del módulo DP
SMB224	Estado DP (intercambio, fallo...)
SMB225	Dirección del “Maestro”
SMW226	Offset en la memoria de variables
SMB228	Número de bytes de los datos de salida
SMB229	Número de bytes de los datos de entrada
VD500	Puntero de datos de salida
VD504	Puntero de datos de entrada

Tabla 5.6. Datos de configuración DP.

El funcionamiento del programa se resume en los siguientes puntos:

- 1) Cálculo del puntero de datos de salida a la memoria de variables (buzón receptor). Si el sistema está en modo intercambio de datos, se calcula el puntero de datos de salida a partir del Offset previamente configurado, en nuestro caso de valor 100.
- 2) Cálculo del puntero de datos de entrada a la memoria de variables (buzón emisor). Si el sistema está en modo intercambio de datos, se obtiene el número de datos de salida para así obtener el puntero de entrada inicial.
- 3) Determinar la cantidad de datos a copiar. Si en modo intercambio de datos, se obtiene el número de bytes de salida y de entrada a copiar.
- 4) Transferir los datos desde “Maestro” a los registros de la CPU “Esclavo DP”. Copiar los datos desde “Esclavo DP” a los registros de la CPU “Maestro”.

El tratamiento de datos para el buzón emisor queda resumido en la Fig. 5.45. En ella se observa cómo se transfieren los datos al buzón emisor para ser enviados al equipo “Maestro”.

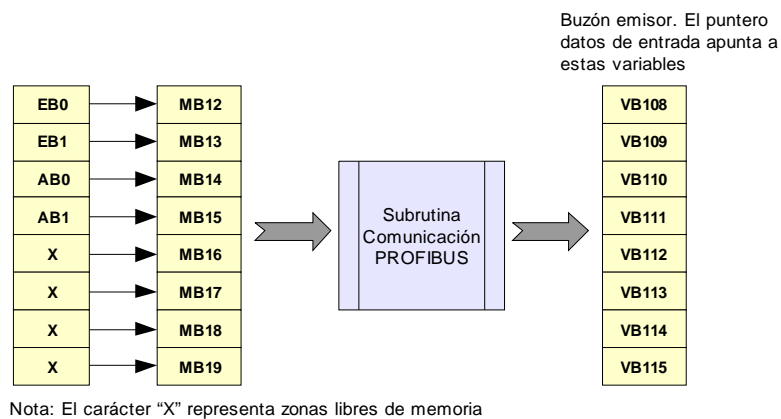


Fig. 5.45. Tratamiento de datos para buzón emisor

El tratamiento de datos para el buzón receptor queda resumido en la Fig. 5.46. En ella se muestra como se recogen los datos desde el buzón receptor para ser procesados en “Esclavo DP”.

Buzón receptor. El puntero datos de salida apunta a estas variables

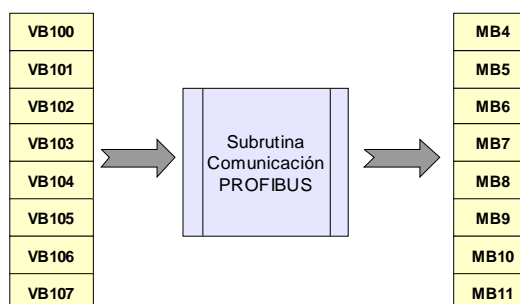


Fig. 5.46. Tratamiento de datos para buzón receptor

Importante

En esta aplicación los datos que se transmiten a través de PROFIBUS poseen *coherencia de bytes*. Esto garantiza que éstos se transfieran en unidades enteras de byte. Esta propiedad está definida por el módulo EM 277.

5.6.3.3 Programación para comunicación GSM

La programación para la comunicación GSM está localizada en varias partes del programa como son “Programa principal”, las subrutinas “Inicializa MODEM”, “Manda mensaje”, “Contenido del mensaje” y “Borrar mensaje”, y las interrupciones “Final RCV” y “Final XTM”.

Además de los segmentos de programa, es necesario crear un bloque de datos para almacenar las “consignas” de la comunicación:

- Número PIN de la tarjeta SIM utilizada, en nuestro caso “8136”.
- Número de destino de los mensajes SMS, en nuestro caso “+34669998427”.
- Texto a transmitir en cada uno de los mensajes, en nuestro caso:
 - *Evento parada de emergencia* del sistema: **“PARADA DE EMERGENCIA”**.
 - *Evento microinterruptor activado*: **“MICROINTERRUPTOR ACTIVADO”**.
- Configuración de mensajes a recibir. En nuestro caso el sistema se ha programado para recibir un solo mensaje: **“ONM”**. Este mensaje pondrá en marcha la unidad (mismo efecto que pulsador de marcha en panel de control).

Las “consignas” reflejadas en los puntos anteriores deben escribirse en el MODEM TC35. Para ello volverán a utilizarse comandos AT, en este caso los mostrados en la Tabla 5.7.

Comando	Función
AT+CPIN	Introducir código PIN para tarjeta SIM
AT+CMGS	Enviar un mensaje a determinado destinatario
AT+CMGR	Leer mensaje recibido
AT+CMGD	Borrar mensaje almacenado en memoria tarjeta SIM

Tabla 5.7. Comandos AT para interactuar con el MODEM a través de la CPU

El bloque de datos creado se muestra en la Fig. 5.47. En ella se observan los diferentes registros que se utilizan y los diferentes comandos e instrucciones que se transfieren al MODEM para el envío/recepción de SMS.

```
// Cadena de caracteres inicial para el PIN
VB10 13
VB11 'AT+CPIN=8136'
VB23 16#0D

// Carga destinatario del mensaje en MODEM
VB50 23
VB51 'AT+CMGS="+34669998427"'
VB73 16#0D

// Mensaje con Microinterruptor
VB150 26
VB151 'MICROINTERRUPTOR ACTIVADO'
VB176 16#1A

// Mensaje con Parada de emergencia
VB200 21
VB201 'PARADA DE EMERGENCIA'
VB221 16#1A

// Cadena de caracteres recibidos
VB300 10
VB301 'AT+CMGR='
VB310 16#0D

// Borrar caracteres recibidos
VB350 10
VB351 'AT+CMGD='
VB360 16#0D
```

Fig. 5.47. Bloque de datos para comunicación GSM

Como se observa en la Fig. 5.47., la estructura de los registros utilizados es similar en todos los casos. Cada interacción de la CPU con el MODEM está estructurada de la siguiente forma:

- *Registro inicial* donde se almacena la longitud del mensaje a enviar/recibir. Importante señalar que cada carácter alfanumérico ocupa 1 byte.
- *Bloque de registros* de longitud la definida por registro inicial.

- *Registro final* de mensaje. Solamente se utiliza en la instrucción enviar mensaje, no en la de recibir. Se utilizan dos caracteres de control en formato hexadecimal para marcar el final del mensaje:
 - *Carácter “0D”*. Marca el final de los mensajes que contienen comandos AT.
 - *Carácter “1A”*. Marca el final de los mensajes que no contienen comandos AT, sólo texto.

A continuación se describen los segmentos de programa necesarios para la comunicación GSM.

Programa principal

En el Programa principal se lleva a cabo la inicialización de la comunicación, así como las diferentes llamadas a las subrutinas de comunicación GSM dependiendo del evento que se produzca.

Inicialización de la comunicación.

La inicialización de la comunicación consiste en configurar el puerto de la CPU para poder comunicarse con el MODEM TC35. Este tipo de configuración recibe el nombre de **“Comunicación modo Freeport”**. Freeport es un modo de operación que permite al programa de usuario controlar por completo el puerto de comunicación de la CPU, gracias a la utilización de interrupciones de recepción y de transmisión, así como las operaciones Transmitir mensaje (XMT) y Recibir mensaje (RCV). La configuración del puerto de comunicación en modo Freeport se realiza modificando el valor de determinadas marcas especiales de la CPU (Tabla 5.8.) (Fig. 5.49.).

Área de memoria	Descripción	Valor	Configuración
SMB30	Registro de control modo Freeport	10#9	Protocolo Freeport, 9600 bit/s, 8 bits por carácter, sin paridad
SMB87	Byte de control de recepción de mensajes	16#CC	Finalizar recepción si excede tiempo indicado en SMB92, temporizar tiempo entre mensajes, detectar comienzo del mensaje, recepción mensajes habilitada
SMB88	Carácter de comienzo del mensaje	16#2B	Se establece el carácter “+”
SMB92	Tiempo de vigilancia	10#5	Finalizar la recepción si se excede el tiempo de 5 segundos
SMB94	Número máximo de caracteres a recibir	10#100	Recibir como máximo 100 caracteres

Tabla 5.8. Parámetros de configuración para modo Freeport.

A continuación es necesario configurar las interrupciones de transmisión y de recepción. Estas interrupciones se encuentran asociadas a un evento y cuando éste se

produce, el programa se interrumpe para ejecutar las líneas de código de la interrupción correspondiente. Se configuran dos interrupciones:

- “Final de RCV”: interrupción asociada al evento 23, recepción de mensajes finalizada.
- “Final de XMT”: interrupción asociada al evento 9, transmisión de datos finalizada.

Importante

La interrupción “Final de XMT” tiene prioridad de ejecución sobre la interrupción “Final de RCV” (evento 9 es prioritario). Esto supone que si ambos eventos se produjeran al mismo tiempo, tendría prioridad la transmisión de mensajes sobre la recepción. Esto se ha hecho así porque el recibir un mensaje depende del centro de mensajes cortos del operador de telefonía y éste se asegura, mediante repeticiones sucesivas de envío, que la recepción del mensaje se ha producido.

También es necesario al inicializar la comunicación activar la opción **Recibir mensaje (RCV)**, con lo que se reduce la posibilidad de perder un mensaje recibido. Observando la Fig. 5.44., esta operación queda activada con los siguientes parámetros:

- Puerto de recepción de mensajes (PORT): Puerto 0.
- Búfer de datos (TBL): VB1000. Este búfer tendrá el formato de la Fig. 5.48.

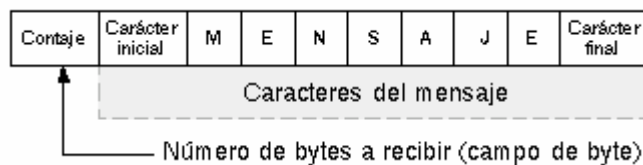


Fig. 5.48. Formato mensaje recibido

Por último y para iniciar el intercambio de datos se inicia el MODEM transfiriéndole el número PIN de la tarjeta SIM (subrutina “Inicializa MODEM”).

La Fig. 5.49. muestra los segmentos de programa necesarios para inicializar la comunicación.

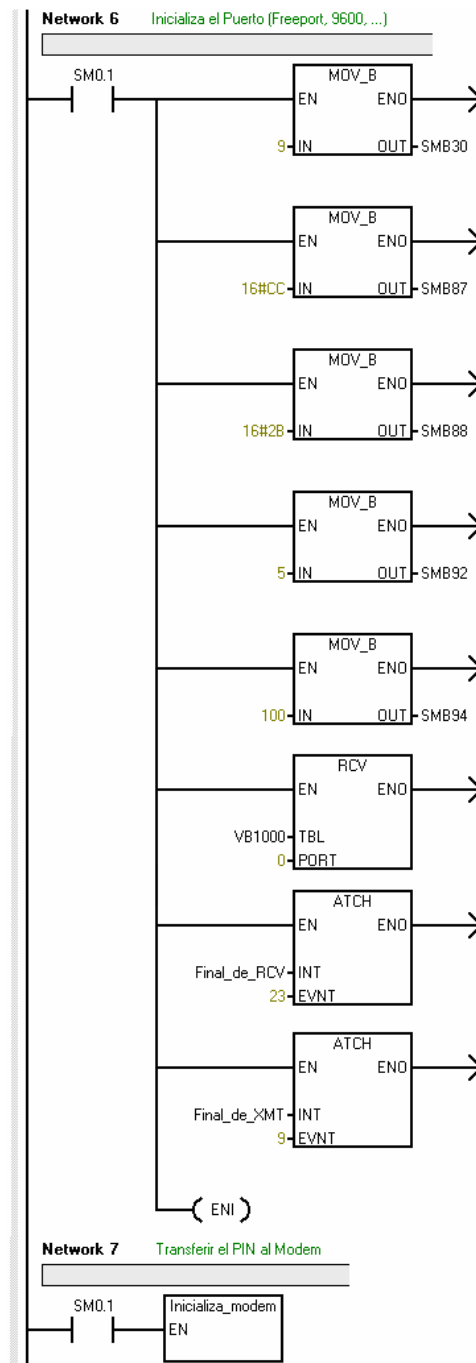


Fig. 5.49. Inicialización de la comunicación GSM

Eventos de comunicación.

La implementación de los eventos de comunicación en el programa de la CPU se observa en la Fig. 5.50.

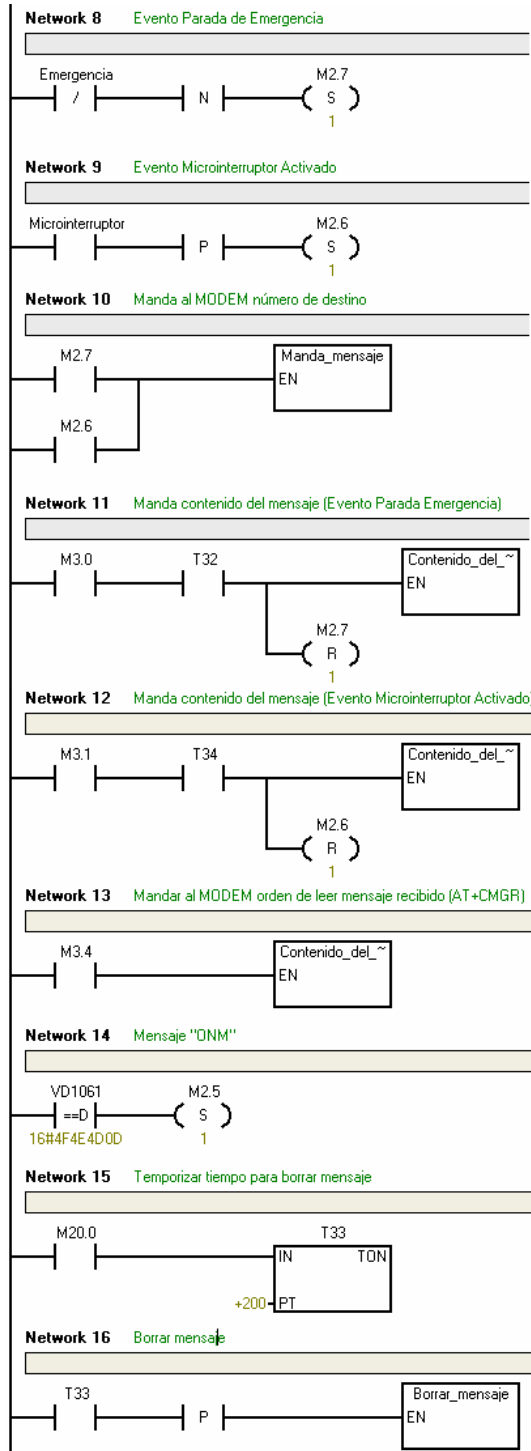


Fig. 5.50. Eventos de comunicación

Como se observa en la Fig. 5.50. los eventos de comunicación se implementan siguiendo la siguiente lógica:

- Transmisión de mensajes: Si se produce en la unidad una “Parada de emergencia” o se activa el “Microinterruptor” se envía al MODEM el número de destino utilizando la subrutina “Manda mensaje”. A continuación y dependiendo del evento, se escribe en el MODEM el texto a transmitir.
- Recepción de mensajes: Si en el MODEM se recibe un mensaje, este notifica a la CPU que tiene mensaje nuevo. Seguidamente ésta manda la orden de leer mensaje al MODEM, el cual proporcionará a la CPU el texto del mensaje recibido, en este caso “ONM”. Automáticamente y tras un periodo de tiempo de 2 s, la CPU dará al MODEM la orden de borrar de la memoria de la tarjeta SIM el mensaje que se acaba de recibir.

Subrutina Inicializa MODEM

En esta subrutina esta implementado el traspaso del número PIN de la tarjeta SIM desde la CPU al MODEM. Esto solo es necesario realizarlo una vez, por lo que la condición de llamada para esta subrutina es la marca especial SM0.1 (solo se activa al pasar a estado RUN la CPU). La Fig. 5.52. muestra la programación correspondiente a esta subrutina, que se resume en los siguientes puntos:

- 1) A través del Byte de control de recepción de mensajes (SMB87) **se inhibe la recepción de mensajes** (Reset SM87.7). Esta operación se realiza para evitar una recepción y un envío simultáneos, hecho que sobrecargaría el puerto de la CPU.
- 2) Se inicializa la operación **Recibir mensaje (RCV)**, con búfer de recepción VB1000 y puerto 0.
- 3) Se inicializa la operación **Transmitir mensaje (XMT)**. Los parámetros para esta operación son:
 - Puerto de envío de mensajes (PORT): Puerto 0.
 - Búfer de datos a enviar (TBL): VB10. Se encuentra representado en la Fig. 5.51.

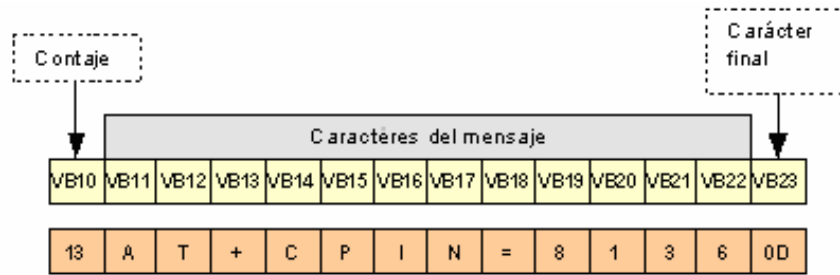


Fig. 5.51. Búfer de datos a enviar para transferir PIN

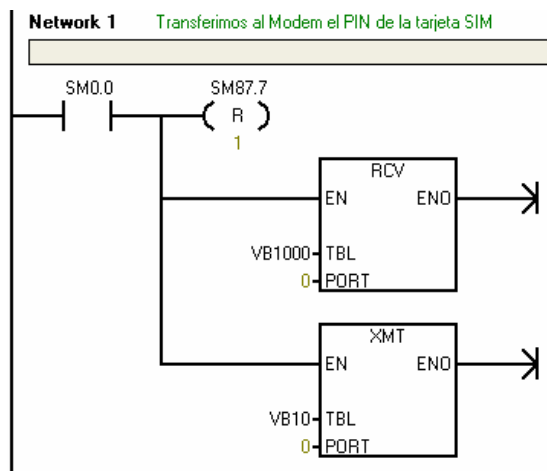


Fig. 5.52. Código de programa para subrutina inicializa MODEM

Subrutina Manda mensaje

La subrutina “Manda mensaje” tiene como objetivo transferir al MODEM el número al cual debe enviar los mensajes SMS. Esta subrutina solo se ejecuta cuando el MODEM debe enviar un SMS, es decir, cuando en la unidad se produce una “Parada de emergencia” o cuando “Microinterruptor activado”. Como se observa en la Fig. 5.54., el código de programa para esta subrutina es muy parecido al código de la subrutina “Inicializa MODEM”. La única diferencia radica en el búfer de datos a enviar, que en este caso tendrá la siguiente estructura (Fig. 5.53.).

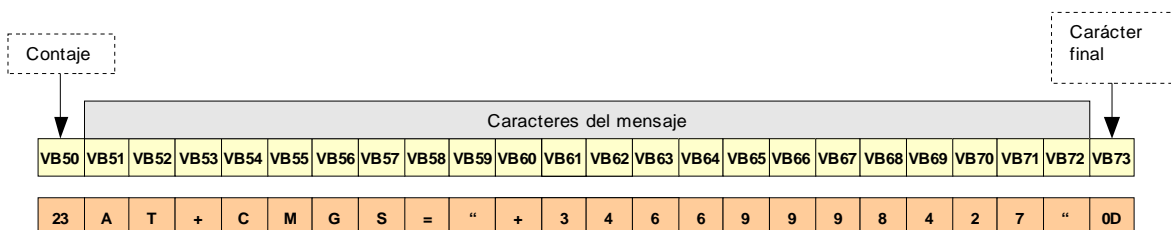


Fig. 5.53. Búfer de datos a enviar para transferir número destinatario

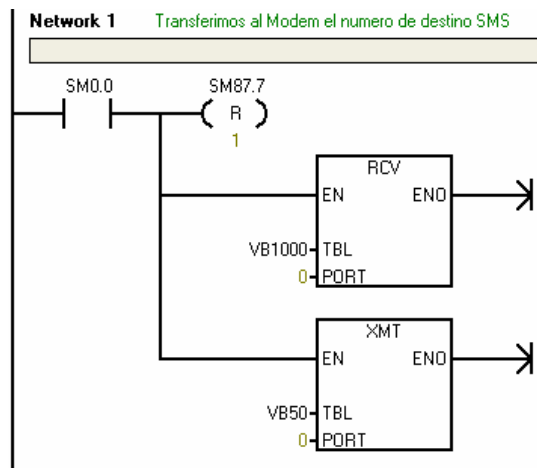


Fig. 5.54. Código de programa para subrutina manda mensaje

Subrutina Contenido del mensaje

En esta subrutina se transfiere al MODEM el texto de los mensajes que éste debe enviar al Terminal móvil. Para el evento parada de emergencia de la unidad se transfiere el texto “PARADA DE EMERGENCIA” y para el evento microinterruptor activado el texto “MICROINTERRUPTOR ACTIVADO”. Además, se envía la orden de leer mensaje recibido que corresponde con el comando “AT+CMGR”. Cada una de estas acciones está condicionada por una marca, la cual se activa dependiendo del mensaje a enviar. El código de programa de esta subrutina (Fig. 5.58.) contiene las siguientes instrucciones:

- 1) Se deshabilita la recepción de mensajes (Reset SM87.7).
- 2) Se inicializa la operación **Transmitir mensaje (XMT)** para el evento parada de emergencia. Los parámetros para esta operación son:
 - Puerto de envío de mensajes (PORT): Puerto 0.
 - Búfer de datos a enviar (TBL): VB200. Se encuentra representado en la Fig. 5.55.

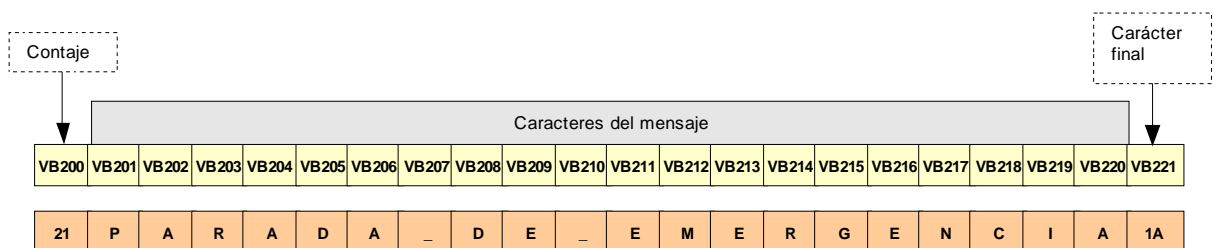


Fig. 5.55. Búfer de datos para parada de emergencia

- 3) Se inicializa la operación **Transmitir mensaje (XMT)** para el evento microinterruptor activado. Los parámetros para esta operación son:
 - Puerto de envío de mensajes (PORT): Puerto 0.
 - Búfer de datos a enviar (TBL): VB150. Se encuentra representado en la Fig. 5.56.

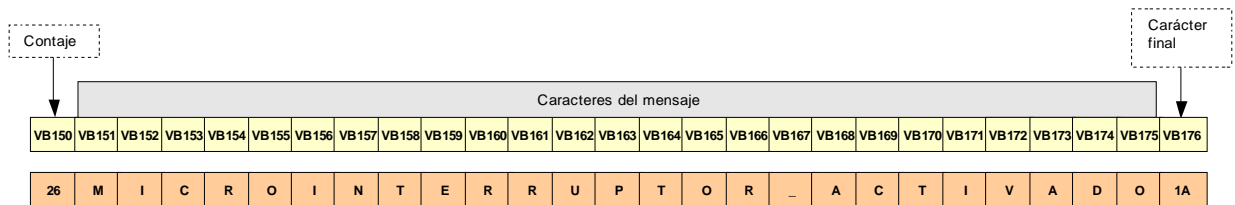


Fig. 5.56. Búfer de datos para microinterruptor activado

- 4) Inicializamos la operación **Transmitir mensaje (XMT)** para la orden leer mensaje recibido. Los parámetros para esta operación son:
 - Puerto de envío de mensajes (PORT): Puerto 0.
 - Búfer de datos a enviar (TBL): VB300. Se encuentra representado en la Fig. 5.57.

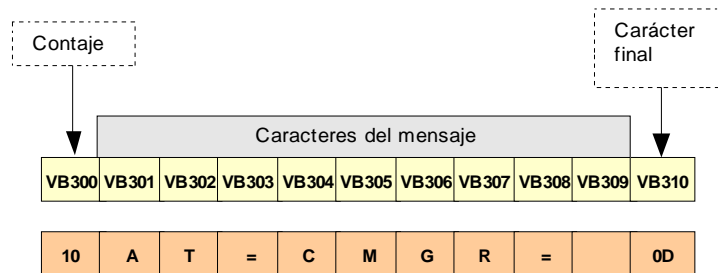


Fig. 5.57. Búfer de datos para orden leer mensaje

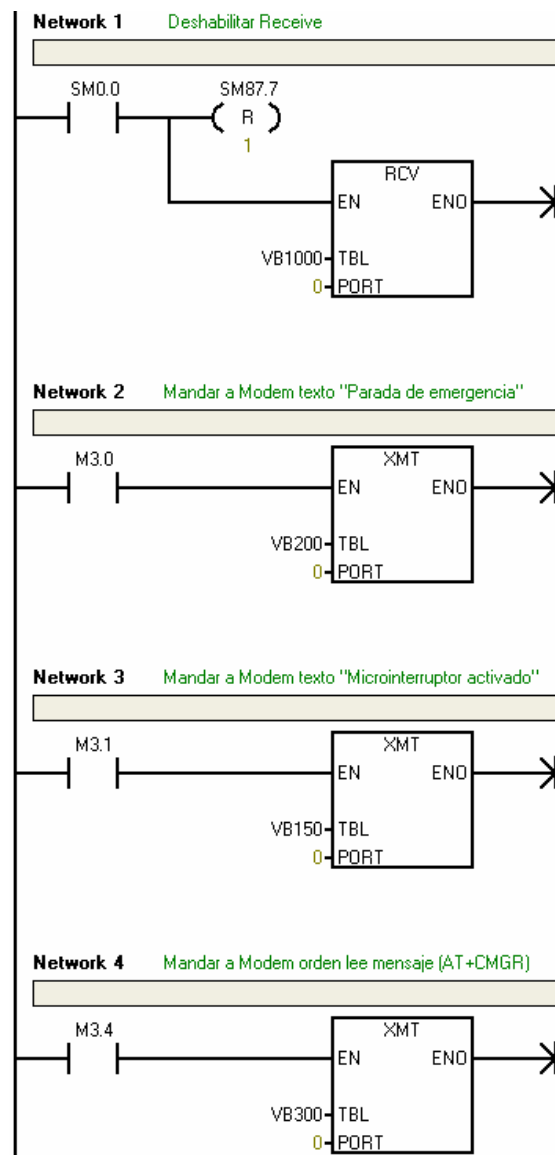


Fig. 5.58. Código de programa para subrutina contenido del mensaje

Subrutina Borrar mensaje

En esta subrutina se envía al MODEM la orden de borrar mensaje (comando AT+CMGD). El código de programa (Fig. 5.60.) es similar al de la subrutina "Inicializa MODEM". Se resume en las siguientes acciones:

- 1) Se deshabilita recepción de mensajes (Reset SM87.7).
- 2) Se inicializa la operación **Recibir mensaje (RCV)**, con búfer de recepción VB1000 y puerto 0.

- 3) Se inicializa la operación **Transmitir mensaje (XMT)** para la orden borrar mensaje. Los parámetros para esta operación son:
 - Puerto de envío de mensajes (PORT): Puerto 0.
 - Búfer de datos a enviar (TBL): VB350. Se encuentra representado en la Fig. 5.59.

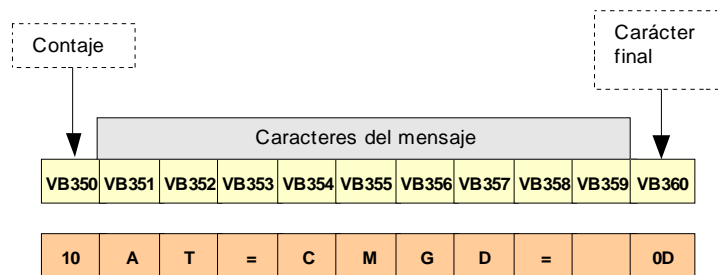


Fig.5.59. Búfer de datos para orden borrar mensaje

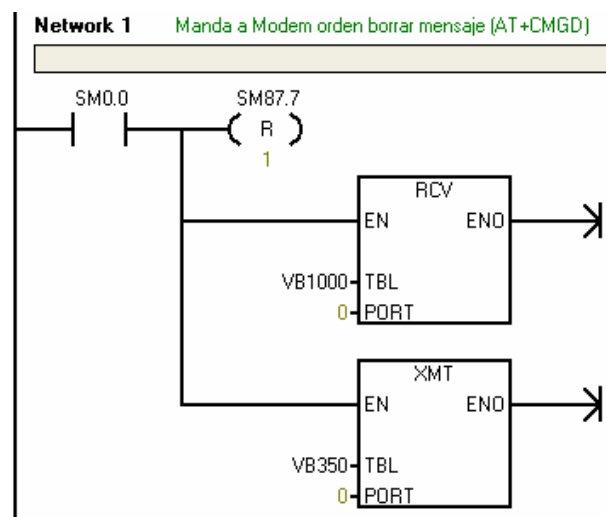


Fig.5.60. Código de programa para orden borrar mensaje

Interrupción Final RCV

Una vez recibido el último carácter del búfer de recepción (*Parámetro TBL de la operación RCV, VB1000*), la CPU genera una interrupción (evento 23) y procesa la rutina “Final RCV” asociada a este evento.

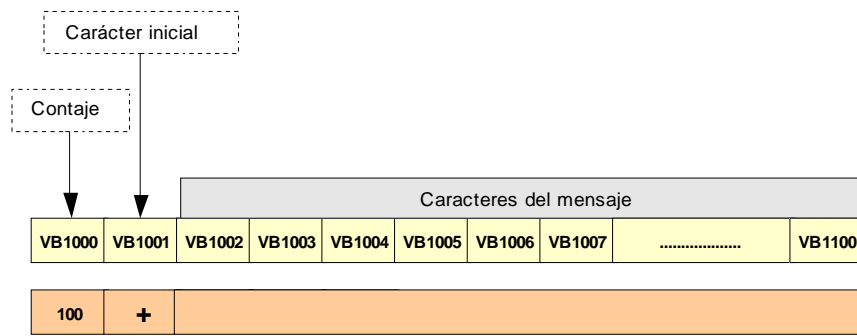


Fig. 5.61. Búfer de recepción

En la Fig. 5.61. se observa el formato del búfer de recepción, con inicio en VB1000 y final en VB1100 (la longitud del búfer es de 100 caracteres como máximo). En la variable VB1001 está depositado el carácter inicial "+", que marca el inicio del mensaje en sí. Los mensajes recibidos pueden ser:

- 1) Indicación de "Nuevo mensaje recibido". El MODEM envía a la CPU el mensaje "+CMTI", indicando que un nuevo mensaje ha sido recibido. Para detectar dicho mensaje se compara el valor de la variable VD1002 (VB1002, VB1003, VB1004, VB1005) con el valor hexadecimal 16#434D5449, que en código ASCII corresponde con la cadena de caracteres "CMTI". Cuando la CPU recibe este mensaje ordena al MODEM leer el nuevo mensaje recibido.
- 2) Indicación de "Mensaje recibido ha sido leído". El MODEM envía a la CPU el mensaje "+CMGR = Caracteres del mensaje". Para detectar que el mensaje ha sido leído se compara el valor de la variable VD1002 (VB1002, VB1003, VB1004, VB1005) con el valor hexadecimal 16#434D4752, que en código ASCII corresponde con la cadena de caracteres "CMGR". Cuando la CPU recibe este mensaje ejecuta la subrutina de "Borrar mensaje".

El código de programa asociado a esta rutina interrupción corresponde con el representado en la Fig. 5.62. En ésta se observa la inicialización de RCV así como las dos comparaciones descritas anteriormente.

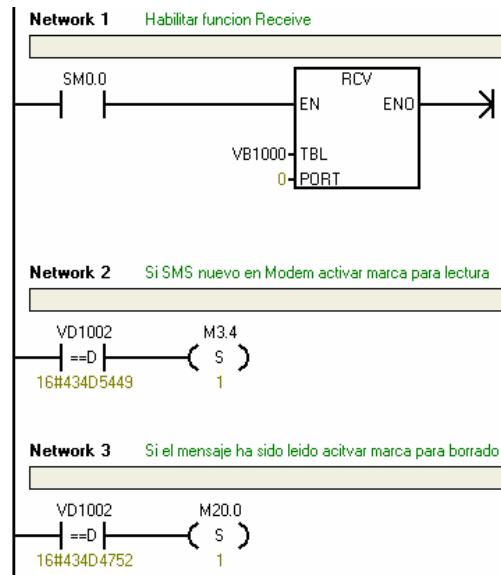


Fig. 5.62. Código de programa para interrupción Final RCV

Interrupción Final XMT

Una vez transmitido el último carácter del búfer de emisión, la CPU genera una interrupción (evento 9) y ejecuta la rutina “Final XMT” asociada a dicha interrupción. En “Final XMT” se procesan todos los tiempos necesarios para que la transmisión se realice correctamente, además de la secuencia para transmitir los mensajes. La Fig. 5.63. muestra el código de programa para “Final XMT”. Cada segmento se encuentra comentado con su función en el programa.

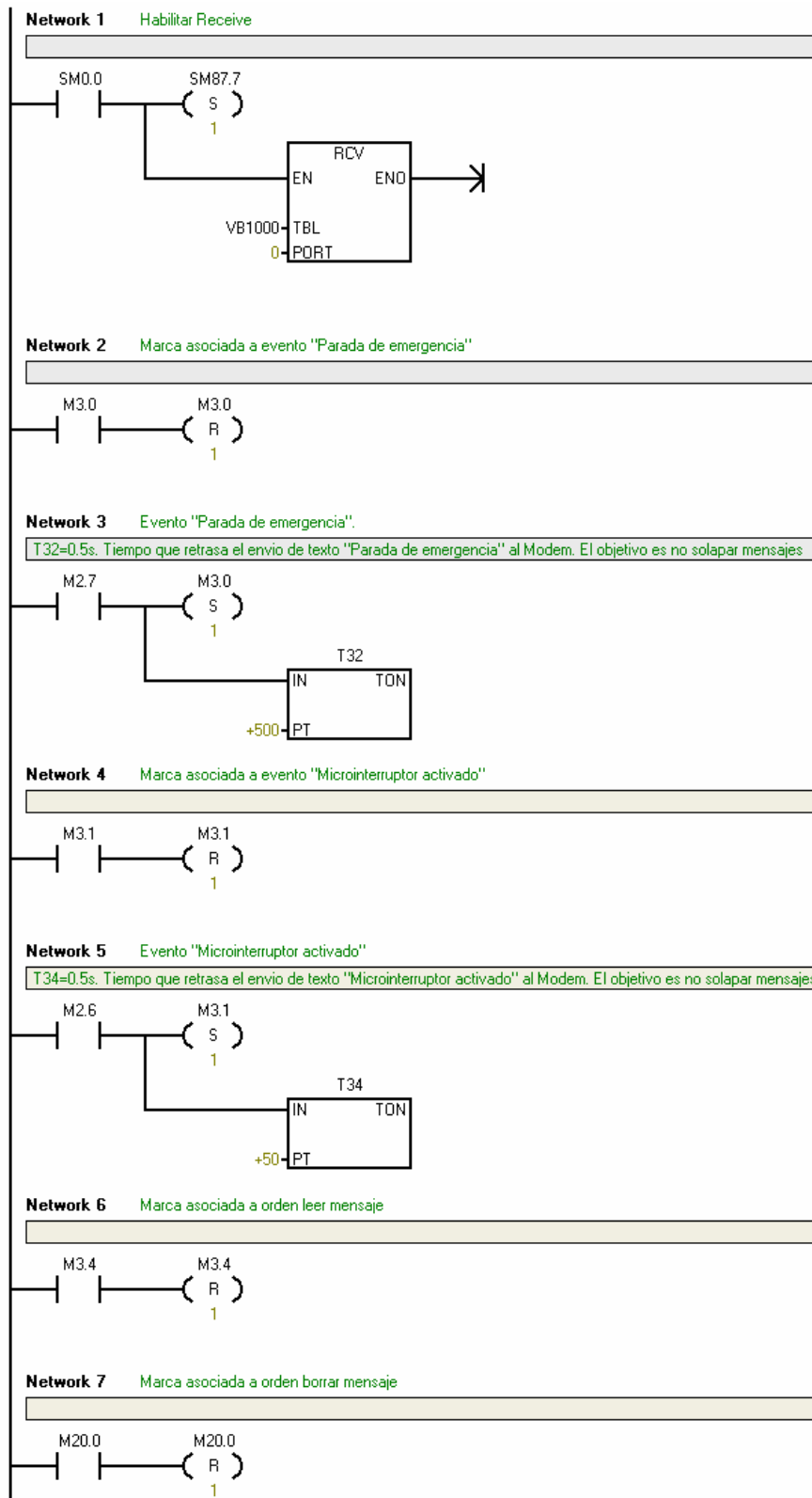


Fig. 5.63. Código de programa para interrupción final XMT

Importante

Al principio de esta interrupción se habilita función RCV. En el resto de las subrutinas, esta operación es deshabilitada al comienzo. De esto se deduce que una transmisión y una recepción simultáneas desde CPU a MODEM es **imposible por programa**, ya que si por ejemplo se produjese una recepción mientras se esta transmitiendo, ésta no sería tenida en cuenta hasta que no se diera el evento final búfer de emisión (evento 9).

A modo de resumen, se incluye una representación en lenguaje GRAFCET de la programación para comunicación GSM (Fig. 5.64. y Fig. 5.65.), en la cual puede verse claramente cómo está estructurado el programa. Las etapas susceptibles de sufrir una interrupción se encuentran marcadas con un óvalo de color rojo. Tras haberse ejecutado la última operación de la rutina de interrupción, el control retorna a la instrucción inmediatamente posterior al punto en el cual ocurrió la interrupción.

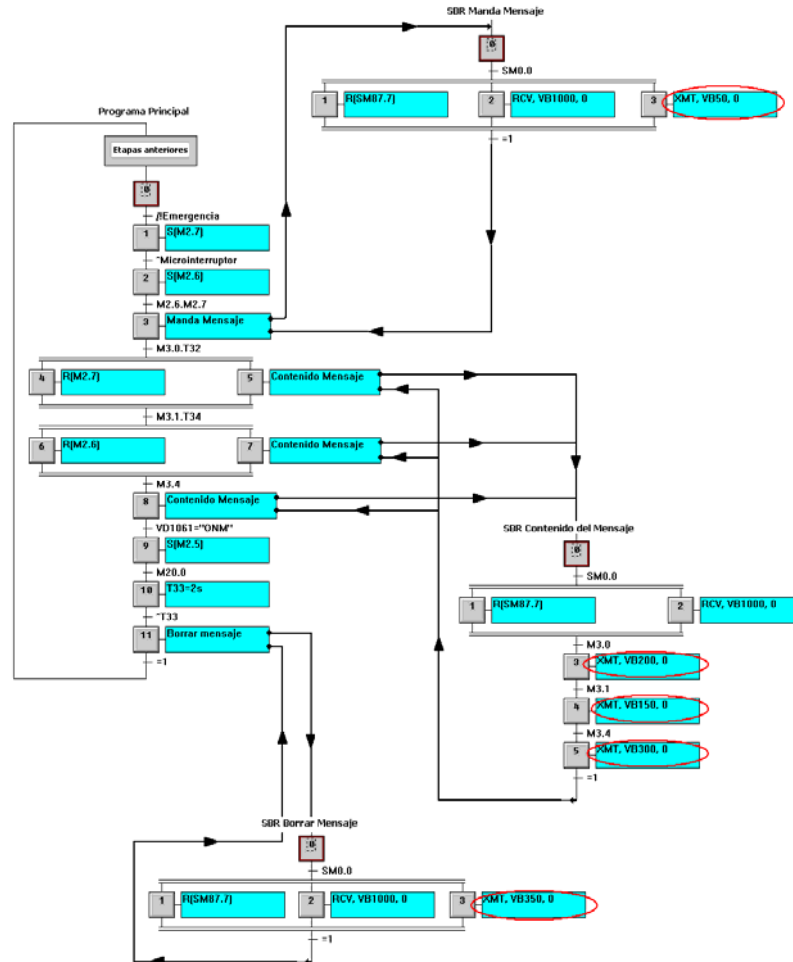


Fig. 5.64. GRAFCET programación para comunicación GSM (I)

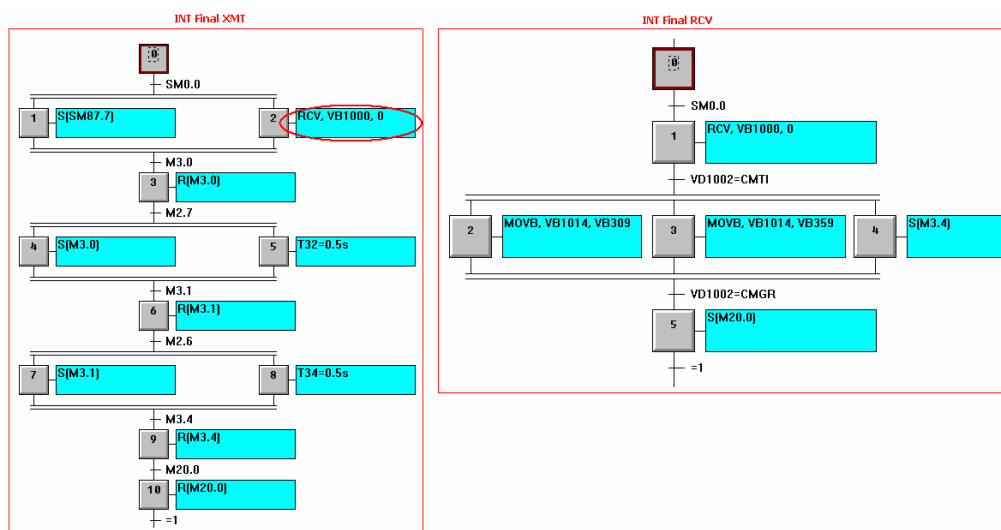


Fig. 5.65. GRAFCET para comunicación GSM (II)

5.7 Configuración equipo Esclavo MPI

El equipo “Esclavo MPI” está formado por una CPU 224 ampliada con un módulo de comunicaciones EM 277 para conexión PROFIBUS. Aunque el “Esclavo MPI” utiliza el interfaz MPI para comunicarse con el equipo “Maestro”, posee un módulo EM 277 que permite integrar este equipo en el bus de comunicación PROFIBUS. El objetivo de esta conexión es simplemente permitir la configuración del equipo a través de PROFIBUS y así evitar el tener que desconectar el interfaz MPI (Port 0 de la CPU) cada vez que queramos modificar la configuración. Además, la velocidad de transferencia a la cual esta ajustado el interfaz de la CPU (187.5 Kbit/s) no es soportada por el cable de configuración PC/PPI.

La configuración del equipo “Esclavo MPI” queda dividida en dos partes: por un lado configurar el módulo EM 277 para la conexión PROFIBUS, y por otro lado configurar la CPU 224 para realizar el intercambio de datos vía MPI y como equipo de control sobre la unidad “cinta transportadora”.

5.7.1 Configuración módulo EM277

Para configurar el módulo EM 277 perteneciente al equipo “Esclavo MPI” se seguirán los mismos pasos que fueron descritos para la configuración en el equipo “Esclavo DP” (*Apartado 5.6.1. “Configuración módulo EM 277”*). Dado que cuando se configuró “Esclavo DP” ya se integro el archivo GSD para EM 277 dentro del catálogo hardware, no será necesario realizar de nuevo esta operación. Por tanto, la

diferencia entre la configuración de ambos módulos radicaré en la parametrización. En los siguientes puntos se indica cómo ha de parametrizarse el módulo EM 277 del “Esclavo MPI”.

- 1) Parámetro **dirección PROFIBUS**. Justo después de insertar el módulo en la aplicación se debe asignar una dirección PROFIBUS. A este módulo se asigna la **dirección PROFIBUS 3** (Fig. 5.66.).

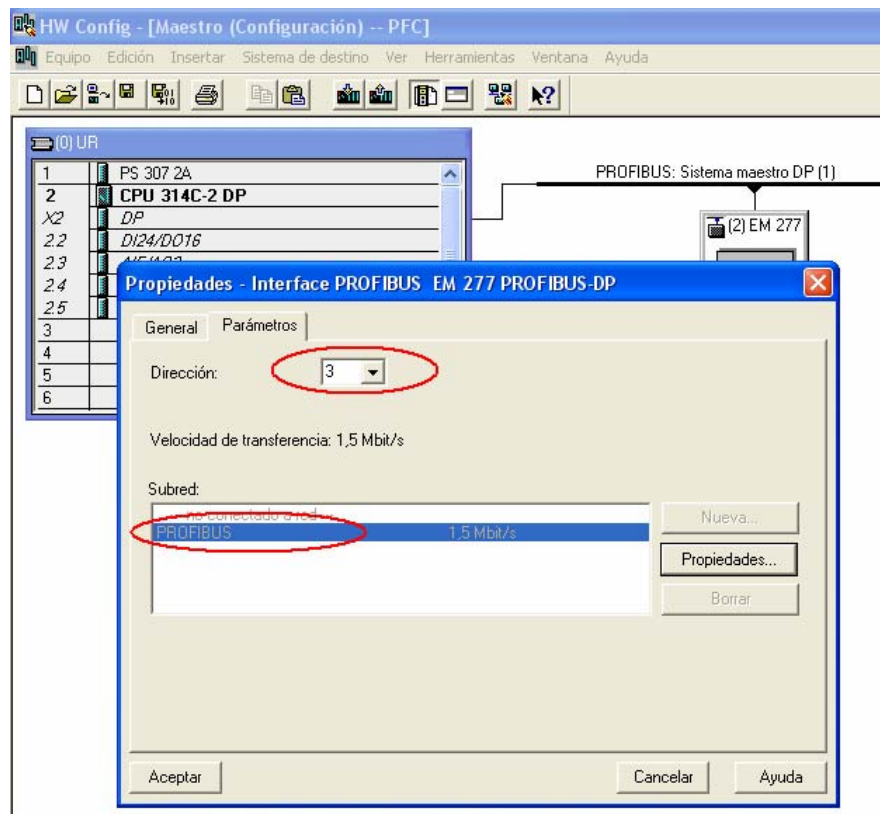


Fig. 5.66. Propiedades interface módulo EM277

- 2) Parámetro **tamaño de búfer de datos**. El tamaño del búfer de datos será igual que en el caso “Esclavo DP” de 8 bytes IN / 8 bytes OUT. El direccionamiento se realiza de forma automática: se asignan los registros inmediatamente posteriores al último registro asignado. Es decir, en este caso los bytes de entradas 8....15 (**PEB8....PEB15**); y los bytes de salidas 8....15 (**PAB8....PAB15**). La Fig. 5.67. muestra esta configuración.

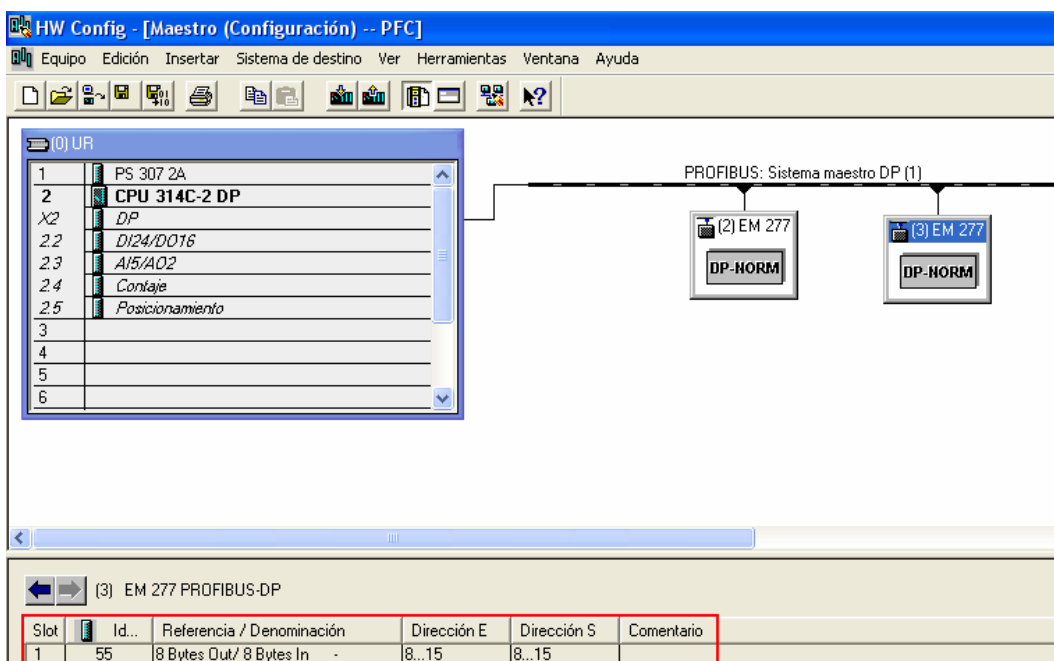


Fig. 5.67. Configuración tamaño de búfer módulo EM277

- 3) Parámetro **Offset**. El parámetro Offset, al igual que en el caso “Esclavo DP” tendrá un valor de 100.

5.7.2 Configuración CPU 224

En este apartado se describirá la programación necesaria para la CPU configurada como “Esclavo MPI”. Al contrario que la CPU “Esclavo DP”, todo el programa de control está contenido en el bloque “Programa principal”. En éste se realiza la adaptación de los datos para la comunicación MPI así como el control de la unidad “*cinta transportadora*”.

5.7.2.1 Programación para el control del módulo

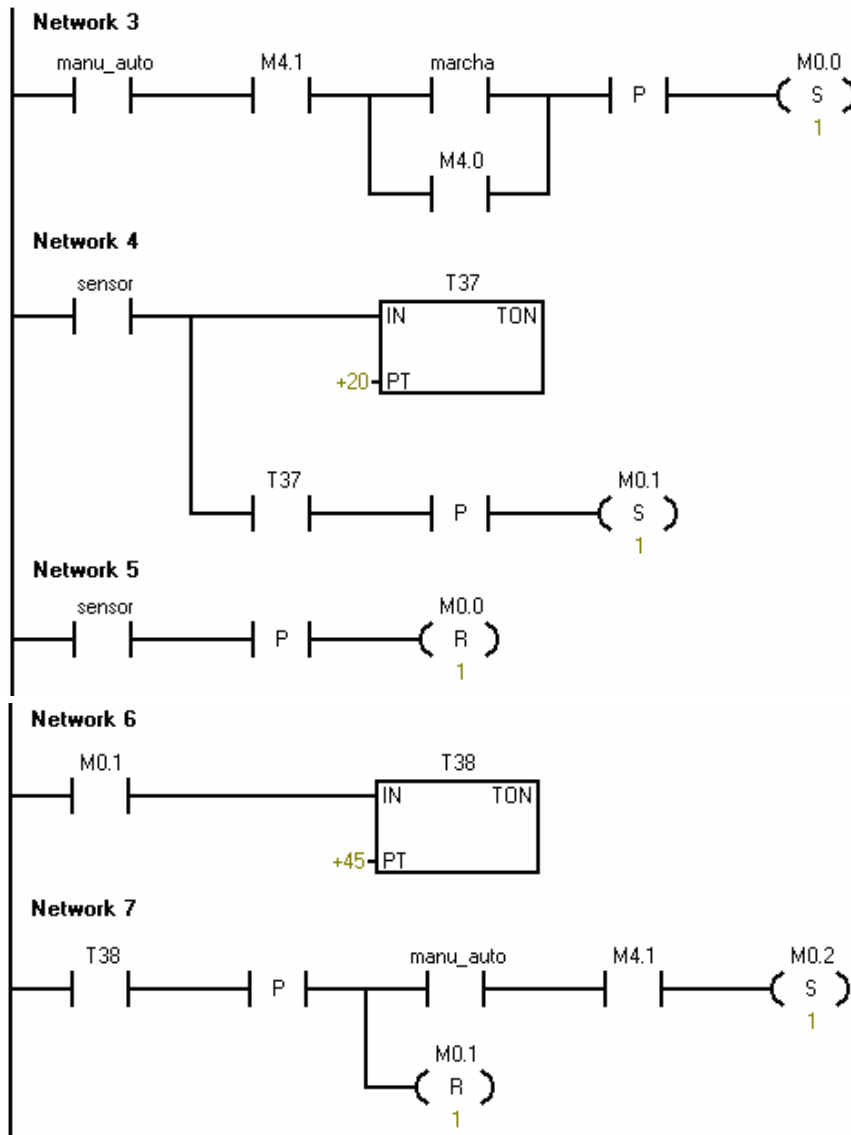
Los algoritmos para el control del módulo “*cinta transportadora*” se han obtenido a partir de la **Referencia Bibliográfica [1]**, aunque se han realizado una serie de modificaciones sobre esta programación.

Antes de comenzar con la descripción del código de programa correspondiente, es necesario comentar que se ha creado una tabla de símbolos (Fig. 5.68.) en la cual se asignan registros del programa a nombres simbólicos, para así facilitar la comprensión del código.

Símbolo	Dirección
manu_auto	E0.1
marcha	E0.0
sensor	E0.3
emergencia	E0.5
verde	A0.0
blanca	A0.1
derecha	A0.2
izquierda	A0.3

Fig. 5.68. Tabla de símbolos módulo cinta transportadora

En la Fig. 5.69. se observa el código de programa para el control del módulo cinta transportadora.



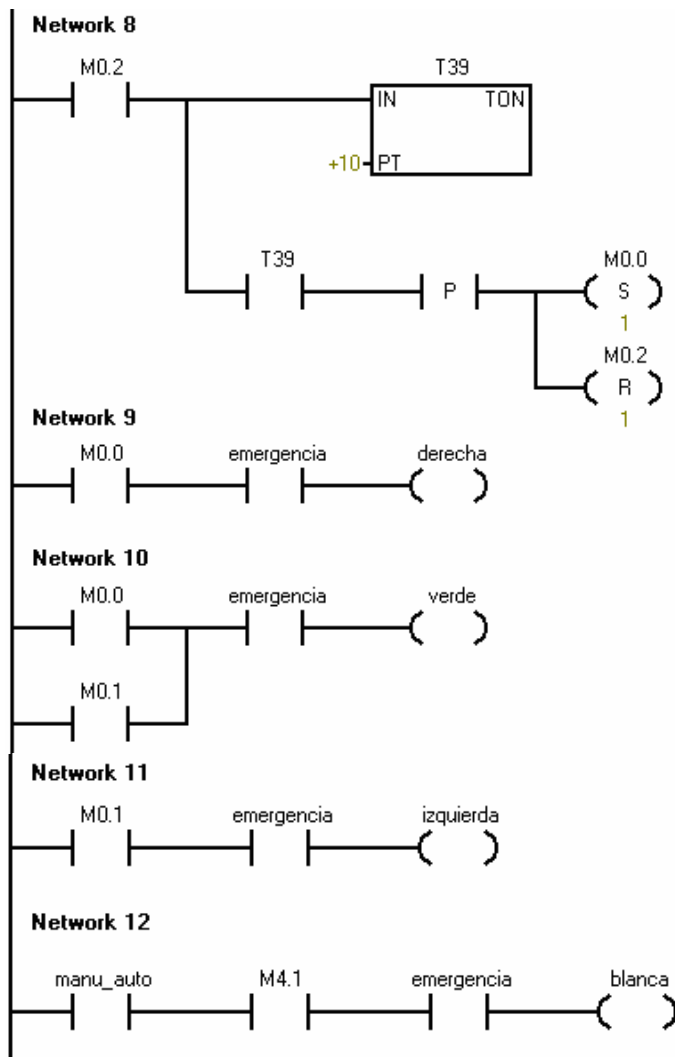


Fig. 5.69. Código de programa control cinta transportadora

Se han utilizado una serie de marcas de sistema que determinan etapas de funcionamiento en el módulo:

- M0.0. Orden de marcha cinta hacia derecha y activación lámpara verde.
- M0.1. Orden de marcha cinta hacia izquierda y activación lámpara verde.
- M0.2. Marca para activar tiempo de espera en posición izquierda.

La funcionalidad del módulo se implementa siguiendo los siguientes puntos:

- Si el interruptor Manual / Automático del panel de control local (*contacto ManAut*) y del panel de control SCADA (*contacto M4.1*) están activados y se activa el botón Marcha en panel de control local (*contacto Marcha*) o en panel de

control remoto (*contacto M4.1*), la cinta comienza a moverse hacia la derecha (Segmento 1).

- Cuando la pieza posicionada sobre la cinta alcanza el sensor ultrasónico (*contacto sensor*) se pone en marcha un temporizador tipo TON (T37) con consigna 2 segundos (Segmento 2).
- Una vez cumplido el tiempo, la cinta se mueve hacia la izquierda durante 4.5 segundos (T38) (Segmentos 2 y 3).
- Pasados los 4.5 segundos, la cinta esperará 1 segundo (T39) y se pondrá en marcha hacia la derecha, repitiéndose el ciclo continuamente (Segmentos 7 y 8).
- Cuando la cinta se encuentre en movimiento se activará la lámpara verde (Segmento 10).
- Cuando el módulo se encuentre realizando ciclos continuos de vaivén, se activará la lámpara blanca (Segmento 12).

5.7.2.2 Programación para comunicación MPI

En primer lugar se configura el interfaz de comunicación de la CPU (Port 0 integrado). Esta configuración se lleva a cabo en el “Bloque de sistema” de la CPU 224 (Fig. 5.70.). Los parámetros a introducir son: **dirección MPI 4** y **velocidad de transferencia 187.5 Kbit/s**, que es la velocidad a la que está ajustado el bus MPI.

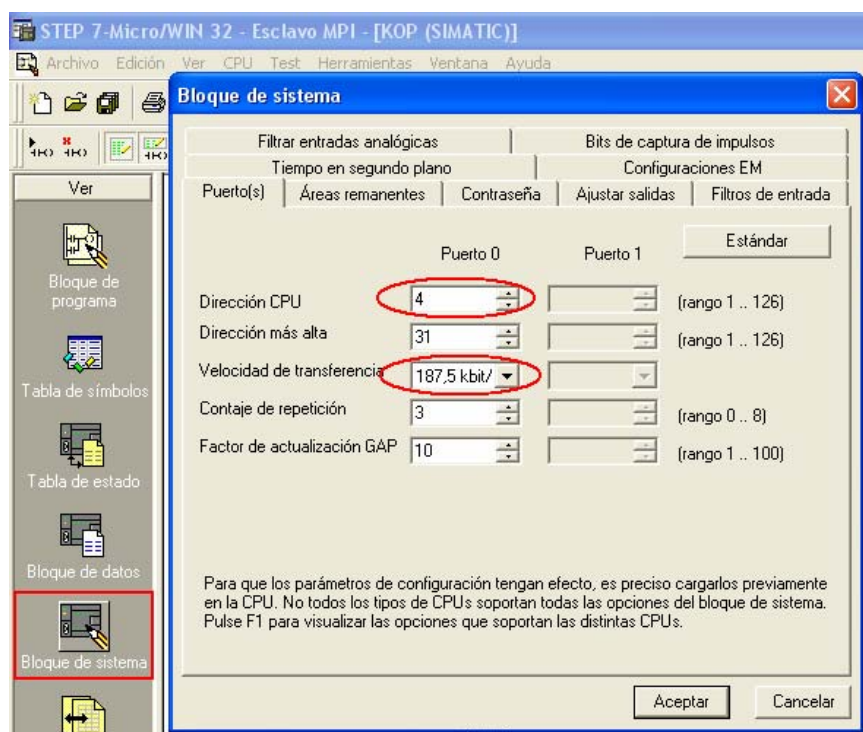


Fig. 5.70. Configuración del puerto de comunicación de la CPU

La programación para la comunicación MPI (Fig. 5.71.) consiste en adaptar los datos que se quieren transmitir al equipo “Maestro”, que es el encargado de leer / escribir estos datos. Puede decirse por tanto que toda la tarea de comunicación recae sobre el equipo “Maestro”.

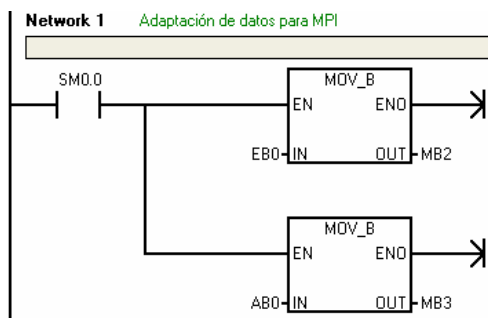


Fig. 5.71. Adaptación de datos para MPI

En la Fig. 5.71. se observa como el byte de Entradas 0 (EBO) se transfiere al byte de Marcas 2 (MB2) y como el byte de Salidas 0 (ABO) se transfiere al byte de Marcas 3 (MB3).

5.8 Configuración buses de comunicación

En este apartado se describe cómo configurar el bus PROFIBUS y el bus MPI de la presente aplicación, utilizando para ello la herramienta *NetPro* (configurar redes), integrada en el software *Administrador SIMATIC*. La Fig. 5.72. muestra la pantalla principal de la aplicación, con todos los elementos integrados hasta ahora.

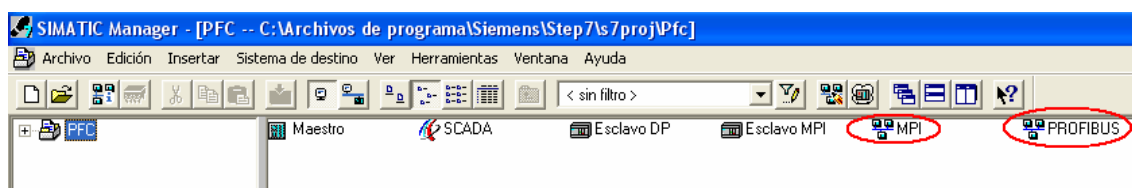


Fig. 5.72. Elementos de la aplicación

Para iniciar la herramienta *NetPro* se selecciona uno de los dos buses (MPI o PROFIBUS) y se hace doble clic sobre él. A continuación, aparece una nueva ventana donde se representa la aplicación configurada hasta ahora pero bajo la herramienta *NetPro* (Fig. 5.73.).

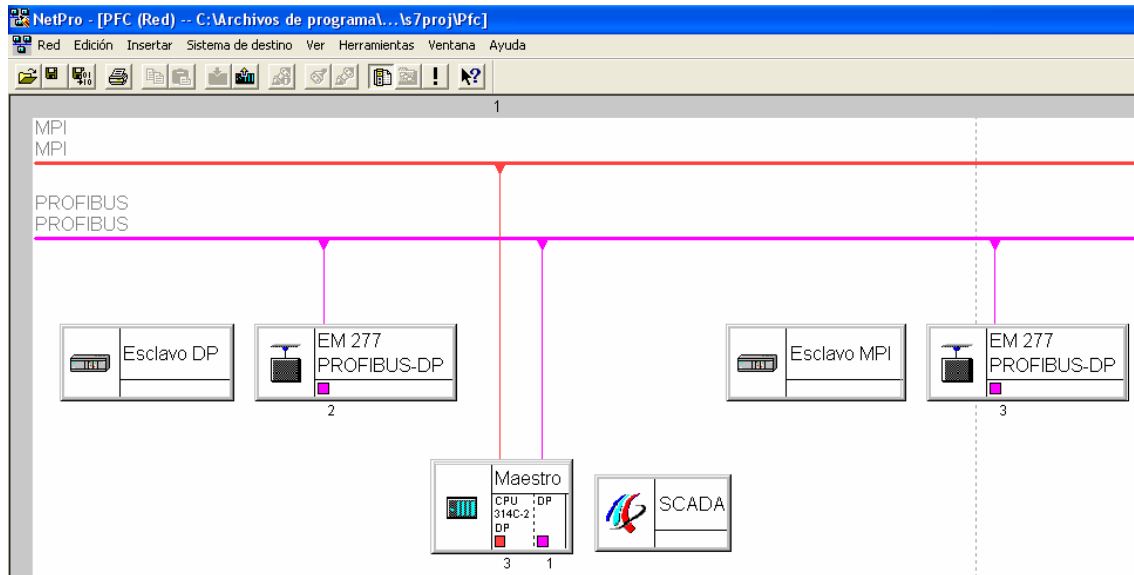


Fig. 5.73. Aplicación representada bajo herramienta NetPro.

5.8.1 Configuración bus PROFIBUS

Configurar el bus PROFIBUS para la presente aplicación consiste en ajustar la velocidad de transferencia del bus y el perfil (DP, FMS....). Además de estos parámetros, hay otros muchos que pueden modificarse para conseguir una comunicación más eficiente para una determinada aplicación (parámetros de tiempo para optimizar el ciclo, sincronismo de esclavos, etc.). En nuestro caso, todos estos parámetros quedarán configurados con los valores “por defecto” y sólo se modificarán velocidad y perfil.

- 1) Se despliega el Menú de configuración para PROFIBUS con la acción Clic botón izquierdo sobre PROFIBUS >> Propiedades del objeto (Fig. 5.74.).

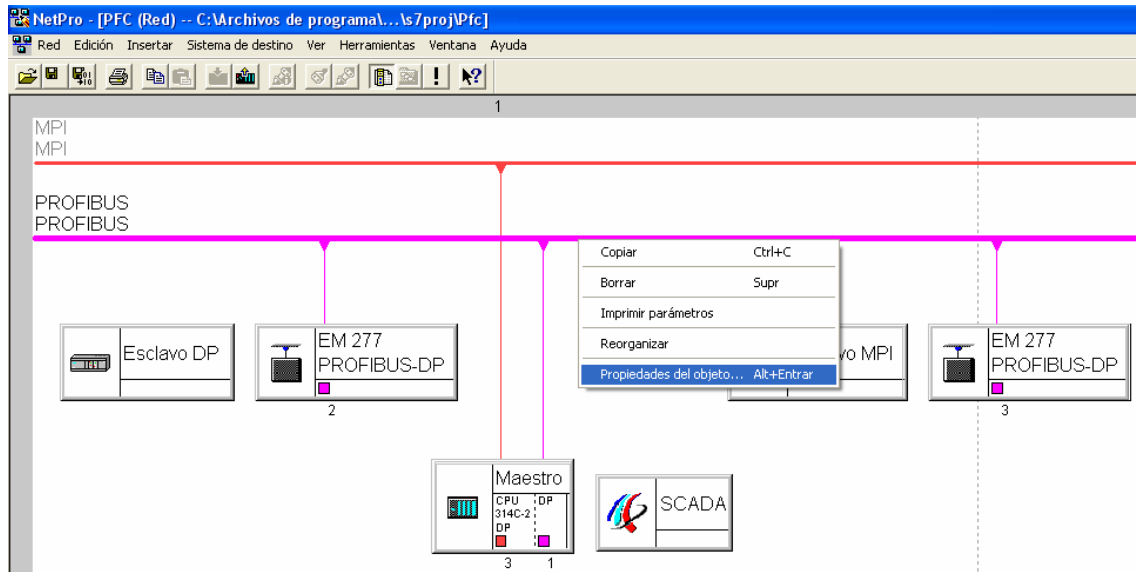


Fig. 5.74. Abrir diálogo propiedades de PROFIBUS.

- 2) En la ventana resultante (Fig. 5.75.) se ajustan velocidad de transferencia y perfil. La velocidad será de **19.2 Kbit/s** y **perfil DP**. El valor 19.2 Kbit/s se ha escogido por dos motivos: es velocidad más que suficiente para la cantidad de datos que se transmiten por el bus y por otro lado, es la velocidad soportada por el cable PC Adapter. El que ambas velocidades sean iguales (cable y bus) permite comunicarse con todos los equipos del bus a través del cable adaptador.
-

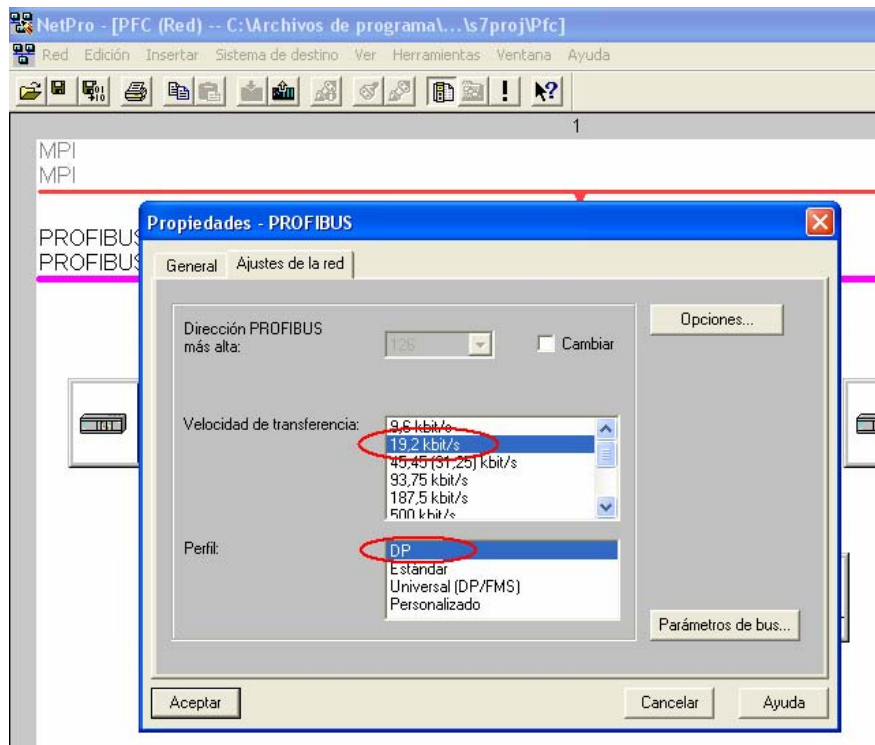


Fig. 5.75. Ajustar velocidad de transferencia y perfil de PROFIBUS

5.8.2 Configuración bus MPI

Configurar el bus MPI en la presente aplicación consiste únicamente en configurar el parámetro velocidad de transferencia en el bus. El resto de parámetros conservarán los valores “por defecto”, ya que para esta aplicación no son importantes.

- 1) Se despliega el Menú de configuración para MPI con la acción Clic botón izquierdo sobre MPI >> Propiedades del objeto (Fig. 5.76.).

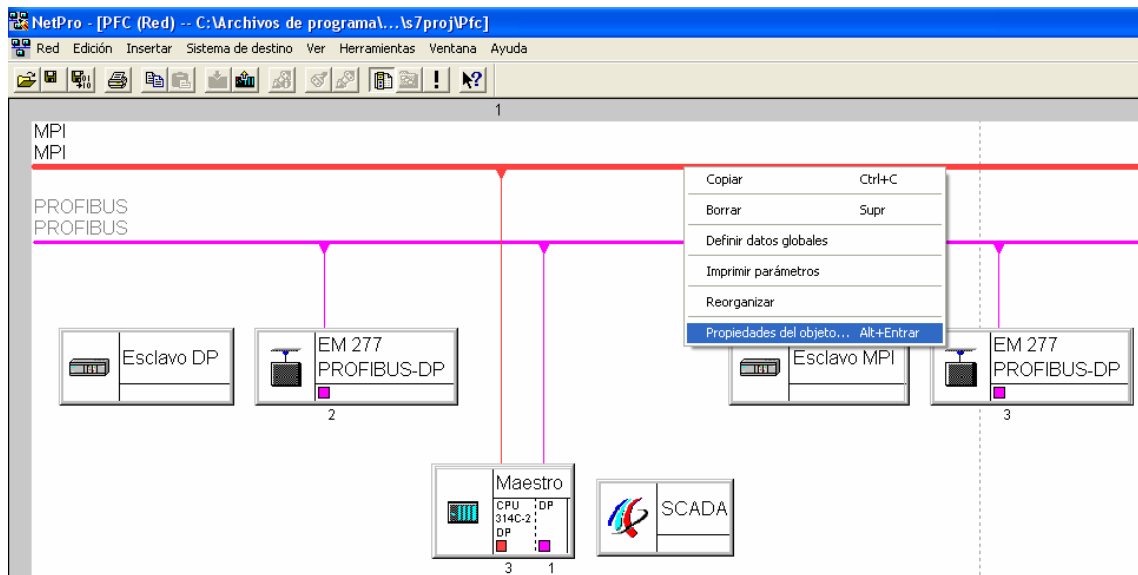


Fig. 5.76. Abrir diálogo propiedades MPI

- 2) En la ventana que emerge (Fig. 5.77.) se ajusta la velocidad de transferencia del bus, que será de **187.5 Kbit/s**. El motivo por el cual se ha escogido un valor tan relativamente alto es que a menores velocidades la transferencia de datos no contempla un tiempo de respuesta útil para una monitorización continua, que es el objetivo de esta aplicación.

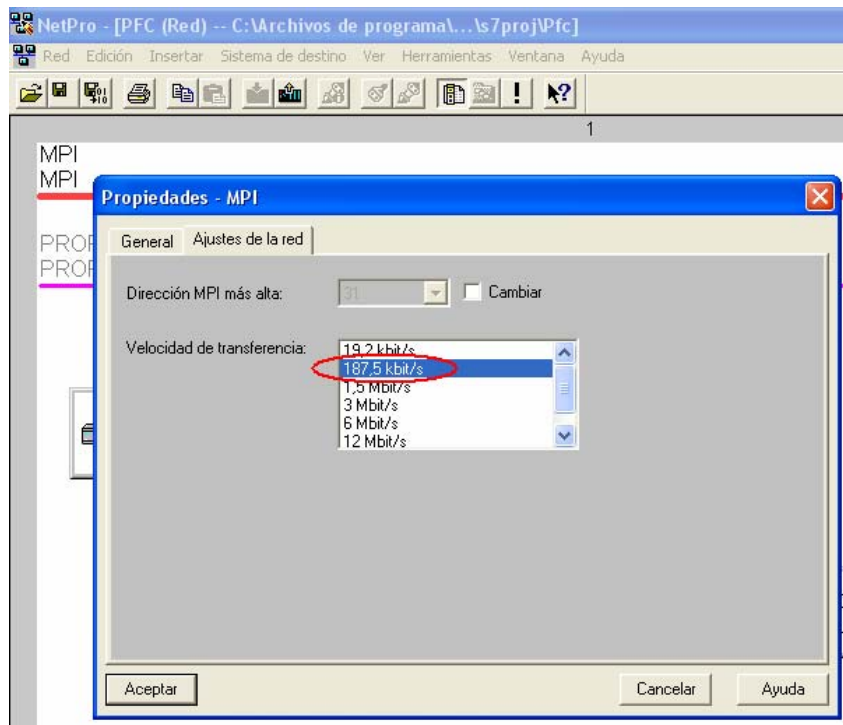


Fig. 5.77. Ajustar velocidad de transferencia bus MPI

5.9 Configuración cables de comunicación

En este apartado se describen las parametrizaciones de los dos cables de comunicación que se utilizan en la aplicación: cable PC/PPI y cable PC Adapter.

5.9.1 Cable PC/PPI

El cable de comunicación PC/PPI tiene como objetivo conectar el equipo “Esclavo DP” al módulo MODEM TC35. El lado RS 485 del cable se encuentra conectado al puerto integrado de la CPU, mientras que el lado RS 232 lo ésta al adaptador macho-macho y éste al MODEM TC35.

Para que la comunicación de la CPU con el MODEM se lleve a cabo con éxito, el cable PC/PPI debe de estar configurado de forma determinada. Para parametrizarlo dispone de 5 pines que pueden adoptar posición “0” ó “1”. Según la combinación creada (“00000”, “01100”.....) el cable adoptará unos parámetros u otros.

En la presente aplicación los parámetros del cable se corresponden con la combinación de los **5 pines “01011”**, que establece las siguientes características de comunicación:

- Velocidad de transferencia: **9.6 Kbit/s**.
- Tipo de MODEM: **MODEM de 10 bits**. Implica que el formato de los datos va ser *8 bits de datos + 1 bit de inicio + 1 bit de parada, sin paridad*.
- **Modo DTE**. La CPU es el equipo Terminal de datos (DTE, del inglés *data Terminal equipment*) y el MODEM es el equipo de terminación de datos (DCE, del inglés *data circuit-terminating equipment*).

5.9.2 Cable PC Adapter

El cable PC Adapter soporta la conexión entre el PC y el equipo “Maestro”. En la parte del PC se encuentra conectado al puerto serie COM1, mientras que el equipo “Maestro” lo está en la interfaz DP. Por tanto, a través de este cable se realiza la comunicación SCADA (implementado en el PC) \leftrightarrow “Maestro” y la comunicación PC \leftrightarrow Bus PROFIBUS. Esta última conexión permite acceder a todos los equipos del bus desde el PC para la configuración de los mismos gracias a las aplicaciones STEP7.

La velocidad de transferencia ajustada para el cable PC Adapter en el lado RS 232 (conectado al PC) es de **38.4 Kbit/s**. La velocidad de transferencia en el lado MPI/DP

(conectado a “Maestro”) será la velocidad ajustada para el bus PROFIBUS (**19.2 Kbit/s**).

5.10 Configuración equipo SCADA

Como se ha ido comentando a lo largo de este capítulo, el equipo SCADA es el encargado de supervisar, controlar y adquirir datos del sistema. Se encuentra implementado en un PC, gracias al software de SIEMENS *ProTool Pro V5.2*. Para iniciar la configuración del SCADA basta con Hacer doble clic sobre icono SCADA en la pantalla principal de la aplicación (Fig. 5.78.).

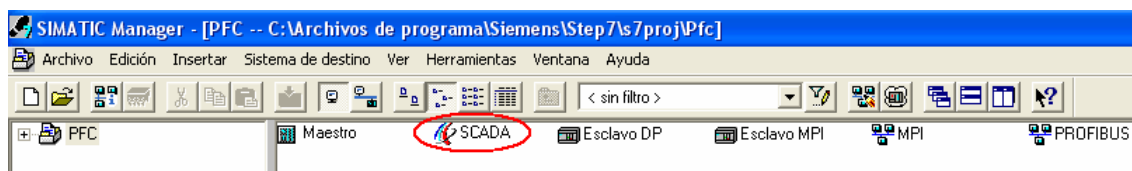


Fig. 5.78. Iniciar configuración SCADA

Seguidamente, el software *Protool Pro V5.2*. se iniciará en su versión diseño (CS), donde se configurarán todos los elementos que compondrán el SCADA. El proceso de configuración sigue los siguientes pasos: en primer lugar, se configura el equipo de destino, en este caso PC; a continuación se indica al SCADA con qué equipo debe interactuar, en nuestro caso el equipo “Maestro”; seguidamente se crean las pantallas de representación o imágenes que va a tener el SCADA; y para finalizar se configuran las alarmas del sistema.

5.10.1 Configurar equipo destino y equipo de control

Al iniciar el software *Protool Pro V5.2*. se inicia de forma automática el asistente para configuración del proyecto. Este asistente permite configurar el equipo de destino, el equipo de control y la conexión a la red PROFIBUS del SCADA.

Selección de equipo de destino

Como se observa en la Fig. 5.79. se presentan varias opciones, de las cuales se elige “**Sistemas basados en Windows**”, que es nuestro caso.

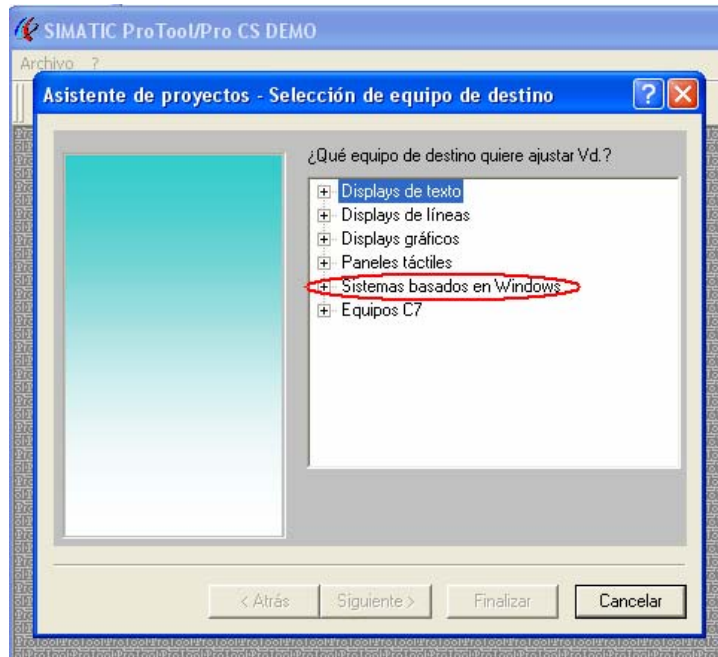


Fig. 5.79. Asistente de configuración equipo destino

Tras esta selección, se desplegará un nuevo Menú donde de nuevo se presentan varias opciones de configuración. Se selecciona la opción **PC**, con una resolución de **1024x768** (Fig. 5.80.), que es la resolución a la cual está ajustado el PC que va a soportar el SCADA.

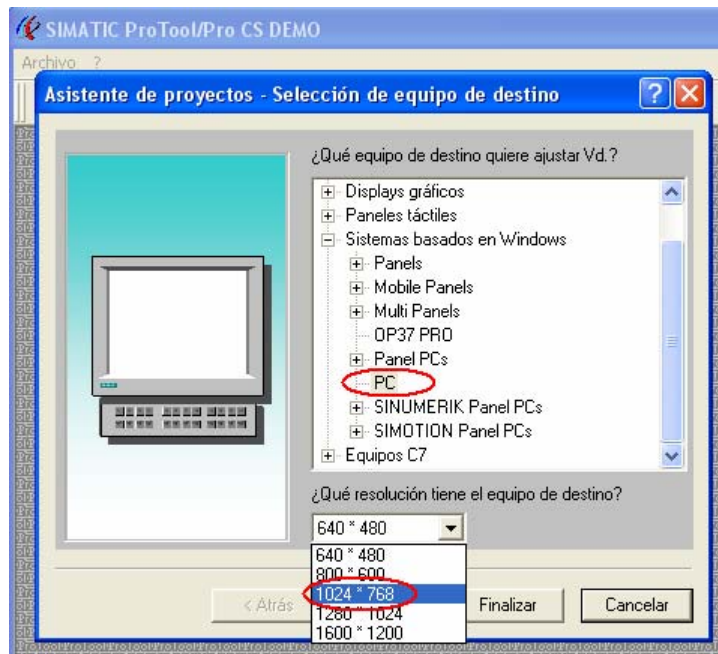


Fig. 5.80. Selección de equipo destino y resolución

Selección del equipo de control

El paso siguiente es configurar el equipo de control. En la Fig. 5.81. se representa la ventana del asistente para esta acción. Se presentan las siguientes opciones:

- Nombre del control: **Maestro**.
- Protocolo de comunicación: **SIMATIC S7 300/400** (equipo “Maestro” es una CPU 314C-2DP).

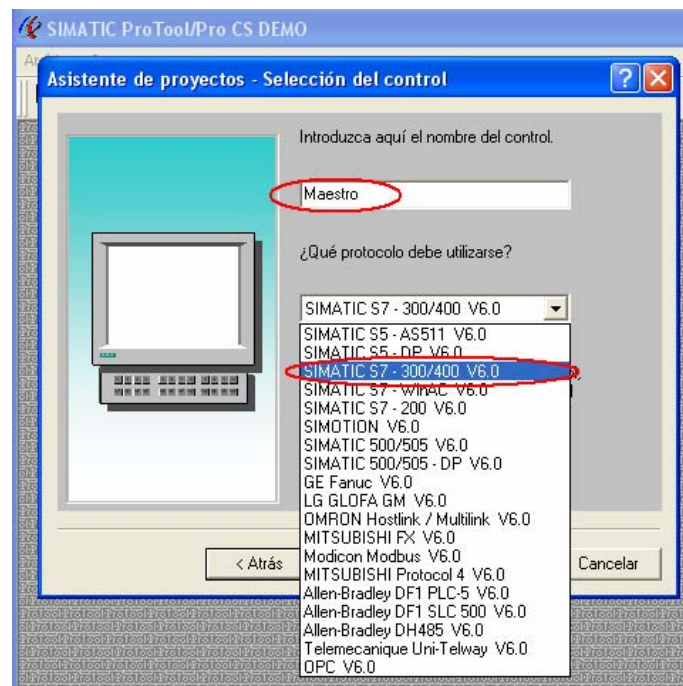


Fig. 5.81. Nombre del control y protocolo de comunicación

En esta misma ventana aparece una pestaña con nombre “Parámetros”. Este apartado permite configurar si el SCADA se encuentra conectado o no a una red, seleccionar el interlocutor de comunicación y asignar una dirección de red al SCADA. En la aplicación se dispone de dos redes creadas (MPI/PROFIBUS) y de tres equipos susceptibles de conexión a través de PROFIBUS (Maestro, Esclavo DP y Esclavo MPI). De forma automática el asistente presenta estas opciones. Se seleccionan los siguientes parámetros:

- Conectar el OP con la red: **PROFIBUS**.
- Seleccionar interlocutor de comunicación: **CPU 314C-2DP**.
- Interface OP: **DP/MPI** (cable PC Adapter).
- Dirección OP: **dirección 4**.

La Fig. 5.82. muestra la ventana “Parámetros” y las selecciones comentadas.

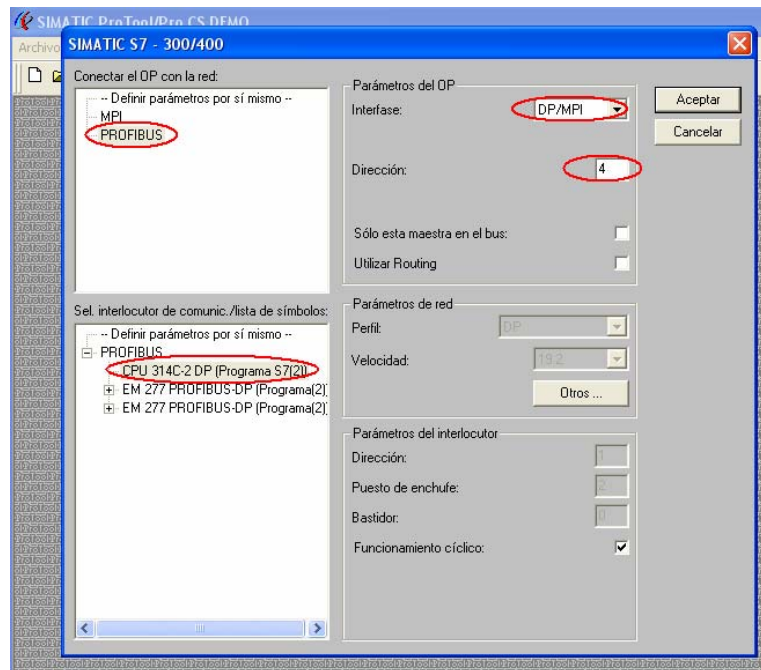


Fig. 5.82. Parametrización conexión SCADA con red PROFIBUS

Importante

En la Fig. 5.82. se observan dos opciones que vienen activadas por defecto. La primera de ellas “*Sólo esta maestra en el bus*” debe desactivarse, ya que en el bus existe otro Maestro. En el segundo caso, “*Funcionamiento cíclico*” se mantendrá activada porque el funcionamiento del interlocutor (“Maestro”) es cíclico en todo momento.

Después de aceptar y finalizar el asistente, se iniciará la ventana de diseño de *ProTool Pro CS* (Fig. 5.83.) en la que aparecerán todos los elementos susceptibles de ser incluidos en un SCADA.

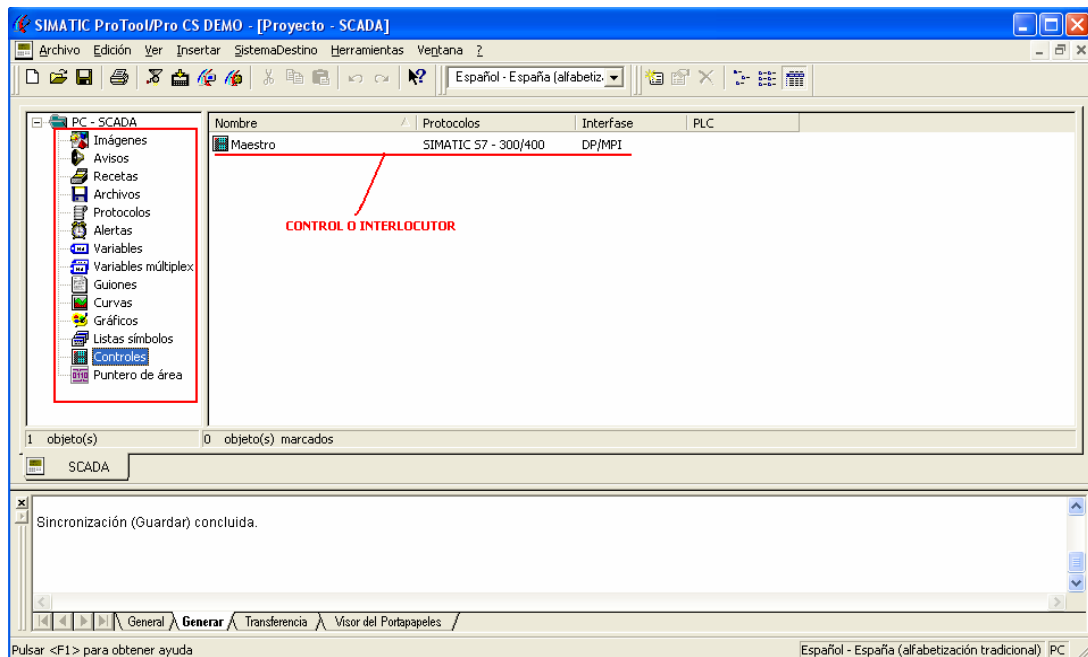


Fig. 5.83. Ventana principal de la aplicación

Configurar área de trabajo

La pantalla de visualización del SCADA se divide en varias áreas que pueden ser configuradas en cuanto a tamaño, visualización, etc. A partir de la **Referencia Bibliográfica [8]** puede obtenerse el uso y opciones de configuración de cada una de estas áreas.

La Fig. 5.84. representa la configuración realizada para la este proyecto.

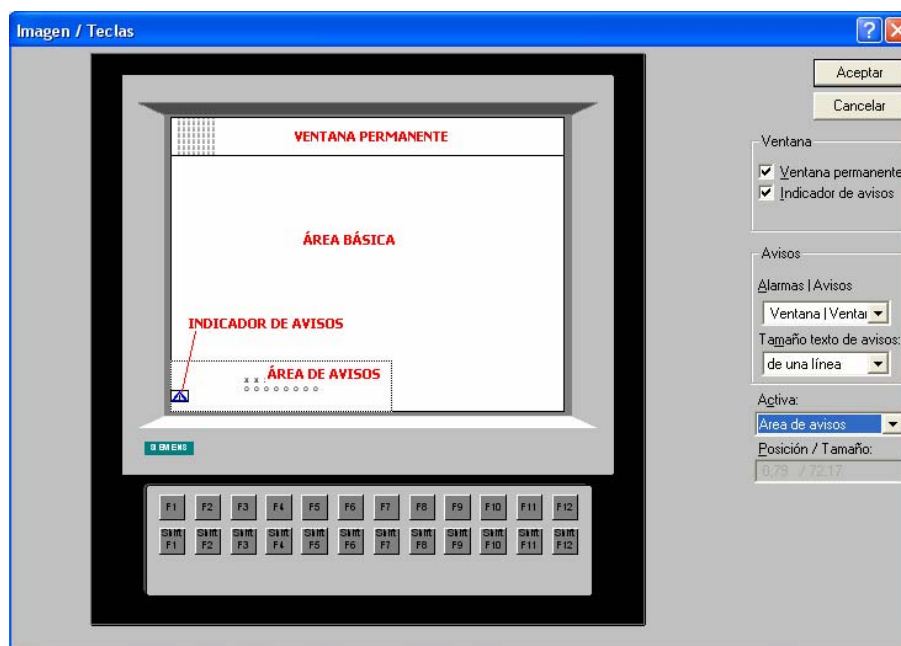


Fig.5.84. Configuración área de trabajo

5.10.2 Configurar imágenes

Las imágenes son una ilustración del proceso a controlar. Una imagen puede constar de partes estáticas y dinámicas. Las partes estáticas, p. ej. texto y gráfico no son actualizadas por el control. Las partes dinámicas están unidas al control y visualizan valores actuales de la memoria del control. La visualización se puede realizar en forma de indicaciones alfanuméricas, curvas y barras. Las partes dinámicas son también entradas en la unidad de operación que se escriben en la memoria del control. El enlace con el control se lleva a cabo a través de variables.

La aplicación práctica de este proyecto consta de tres imágenes de representación:

- 1) Imagen “Principal”: representa un sinóptico de toda la red de comunicación y esta configurada como la imagen inicial.
- 2) Imagen “Alimentador”: representa el módulo “doble alimentador por gravedad”.
- 3) Imagen “Cinta”: representa el módulo “cinta transportadora”.

En la Fig. 5.85. se muestran las tres imágenes incluidas en la ventana principal de la aplicación.

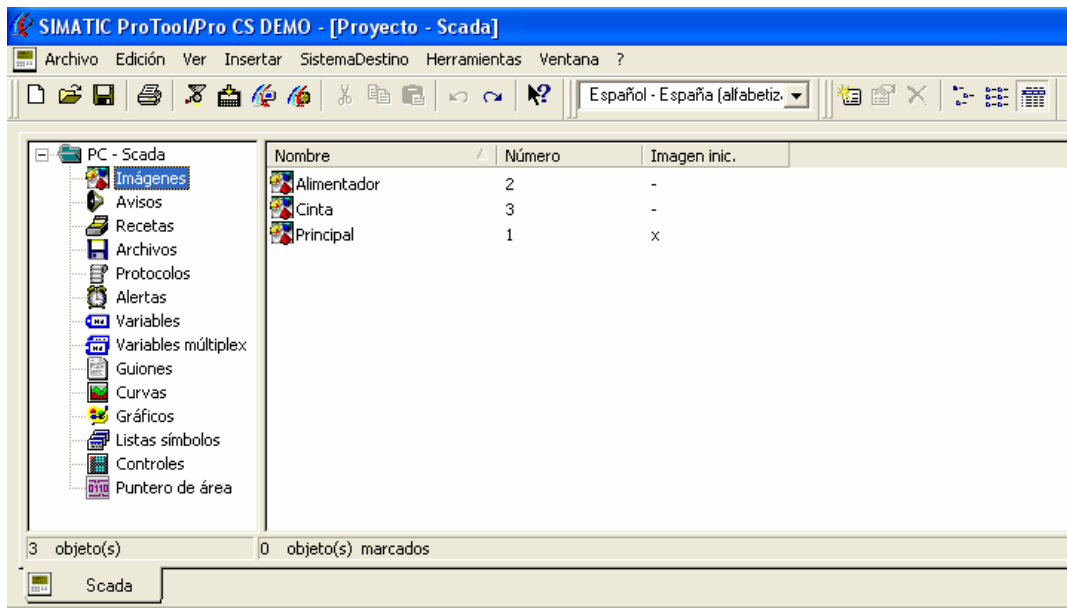


Fig. 5.85. Imágenes de la aplicación

5.10.2.1 Imagen “Principal”

Es la imagen inicial de la aplicación. Representa una visión general de toda la red de comunicación y de los equipos que la componen y desde ella se selecciona el módulo a visualizar/controlar. Se representa en la Fig. 5.86.

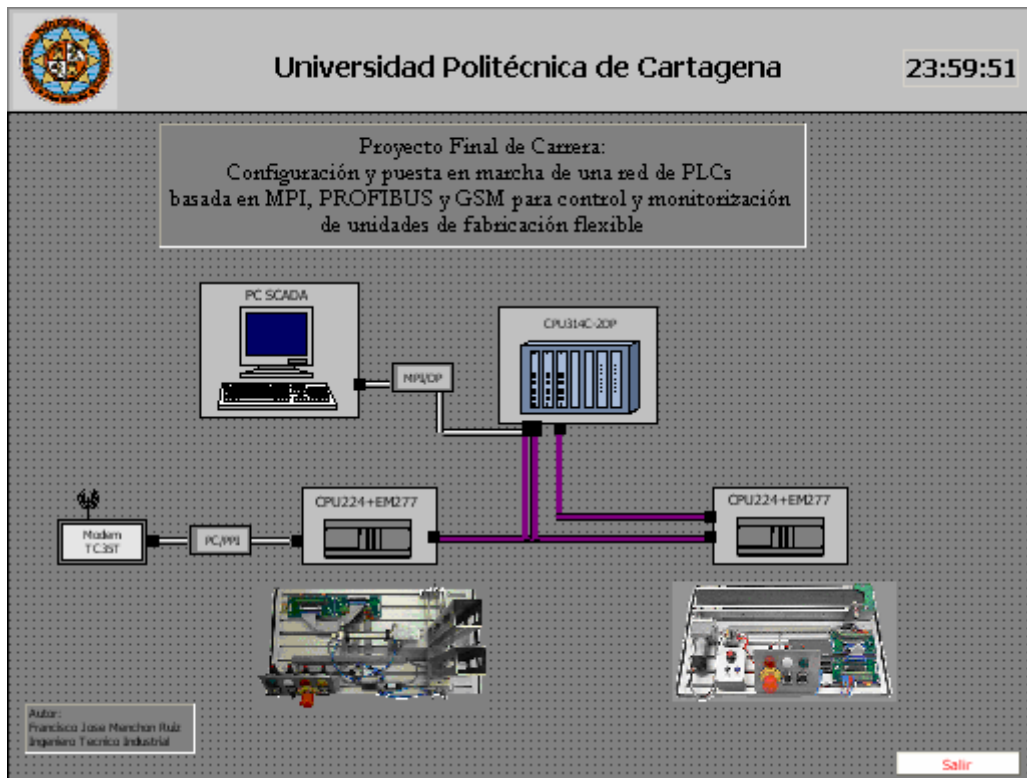


Fig. 5.86. Imagen "Principal"

Se presentan las siguientes opciones:

- Entrar en imagen "Alimentador". Si se selecciona la ilustración "doble alimentador por gravedad", se visualizará en el SCADA este módulo.
- Entrar en imagen "Cinta". Si se selecciona la ilustración "cinta transportadora", se visualizará en el SCADA este módulo.
- Visualizar información equipo. Si se selecciona alguno de los equipos que componen la aplicación, se obtendrá información general del mismo (dirección de red, equipo, conexiones, etc.).
- Salir RUNTIME. Si se selecciona el botón "Salir" finalizará la aplicación SCADA.

El proceso para crear una imagen, insertar gráficos, asignar variables, etc. está recogido en la **Referencia Bibliográfica [8]**.

5.10.2.2 Imagen “Alimentador”

Es la imagen donde se representa y controla la unidad “doble alimentador por gravedad”. Ha sido obtenida a partir de la **Referencia Bibliográfica [1]**. La Fig. 5.87. muestra esta imagen.

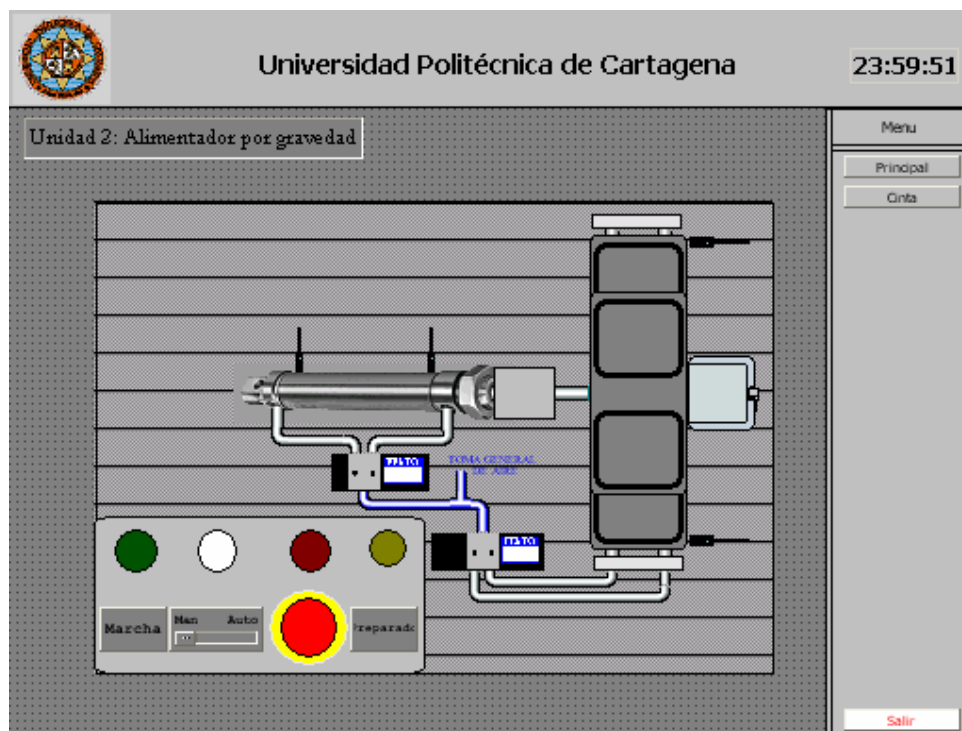








Fig. 5.87. Imagen “Alimentador”

Para integrar la imagen en la aplicación se ha reconfigurado la misma. Esto significa que se han tenido que cambiar todas las variables de control asociadas a los elementos de la imagen por las variables de control de la presente aplicación. La forma de realizar este cambio y de configurar la imagen se recoge en la **Referencia Bibliográfica [8]**. En la Fig. 5.89. se observa el conjunto de variables utilizadas en el SCADA.

En la imagen “Alimentador” se presentan las siguientes opciones:

Comando	Acción	Nombre simbólico Protocol variable	Dirección Maestro	Dirección Esclavo DP
	Selecciona modo Automático o modo Manual	<i>ManAutoAlimen</i>	DB3.DBX4.0	M4.0
	En modo Automático: pone en marcha el módulo En modo Manual: avance/retroceso del cilindro extractor	<i>MarchaAlimen</i>	DB3.DBX4.1	M4.1

	En modo Automático: posiciona las torres de alimentación a la derecha En modo Manual: posiciona las torres de alimentación en derecha o izquierda según posición actual	<i>PreparAlimen</i>	DB3.DBX4.2	M4.2
	Muestra la imagen "Principal"	(*)		
	Muestra la imagen "Cinta"	(*)		
	Finaliza la Runtime	(*)		

(*) Estos botones no tienen asignada ninguna variable en el control. Son funciones internas de ProTool.

5.10.2.3 Imagen "Cinta"

Es la imagen donde se representa y controla la unidad "cinta transportadora". Ha sido obtenida a partir de la **Referencia Bibliográfica [1]**. La Fig. 5.88. muestra esta imagen.

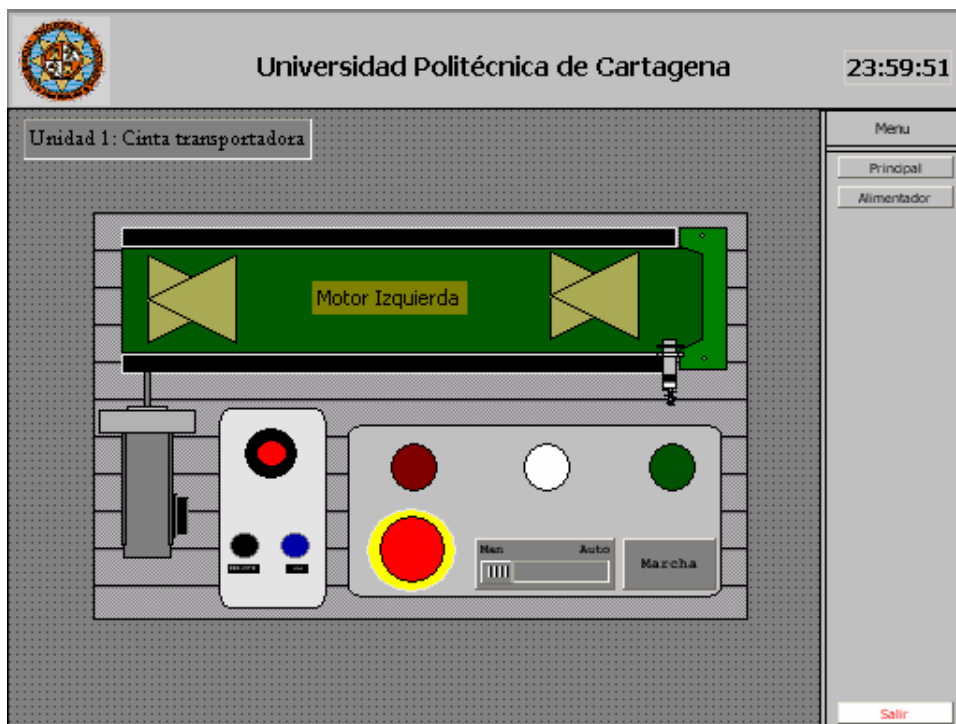







Fig. 5.88. Imagen "Cinta"

Al igual que en el caso anterior, para integrar esta imagen en esta aplicación se ha tenido que modificar toda la configuración referente a variables de control. En la **Referencia Bibliográfica [8]** se detalla cómo se realiza este cambio.

En la imagen "Cinta" se presentan las siguientes opciones:

Comando	Acción	Nombre simbólico Protocol variable	Dirección Maestro	Dirección Esclavo DP
	Selecciona modo Automático o modo Manual, aunque Manual no está implementado	<i>ManAutoCinta</i>	DB2.DBX2.1	M4.1
	En modo Automático: pone en marcha el módulo	<i>MarchaAlimen</i>	DB2.DBX2.0	M4.0
	Muestra la imagen "Principal"		(*)	
	Muestra la imagen "Alimentador"		(*)	
	Finaliza la Runtime		(*)	

(*) Estos botones no tienen asignada ninguna variable en el control. Son funciones internas de ProTool.

5.10.3 Configurar variables

Los elementos que componen las imágenes del SCADA, según sean dinámicos o estáticos, se encuentran asociados a variables o registros pertenecientes al equipo de control. En el caso de la presente aplicación, estos registros están almacenados en dos bloques de datos del equipo "Maestro": DB2 "Datos MPI" y DB3 "Datos PROFIBUS".

Al mismo tiempo, se establecen dos tipos de variables: variables de solo supervisión y variables de supervisión y control. La diferencia entre ambas radica en que las de supervisión y control permiten modificar su valor desde el SCADA, además de su supervisión.

La Fig. 5.89. representa todas las variables de las que consta la aplicación. Se observa nombre simbólico de las variables para ProTool, tipo de variable, control, así como la dirección de la variable dentro del equipo control ("Maestro").

Nombre	Tipo	Control	Dirección	Tiempo polling	Lectura continua	Número de ele...
SensorCinta	BOOL	Maestro	DB 2 DBX 0.3	1.0	-	1
EmergCinta	BOOL	Maestro	DB 2 DBX 0.5	1.0	-	1
VerdeCinta	BOOL	Maestro	DB 2 DBX 1.0	1.0	-	1
BlancaCinta	BOOL	Maestro	DB 2 DBX 1.1	1.0	-	1
DerechaCinta	BOOL	Maestro	DB 2 DBX 1.2	1.0	x	1
IzquierdCint	BOOL	Maestro	DB 2 DBX 1.3	1.0	x	1
MarchaCinta	BOOL	Maestro	DB 2 DBX 2.0	1.0	-	1
ManAutoCinta	BOOL	Maestro	DB 2 DBX 2.1	1.0	-	1
EmergAliment	BOOL	Maestro	DB 3 DBX 0.0	1.0	-	1
MicroAliment	BOOL	Maestro	DB 3 DBX 0.4	1.0	-	1
CiliAtraAlim	BOOL	Maestro	DB 3 DBX 0.5	1.0	-	1
CiliDelaAlim	BOOL	Maestro	DB 3 DBX 0.6	1.0	-	1
DerechAlimen	BOOL	Maestro	DB 3 DBX 0.7	1.0	-	1
IzquiAliment	BOOL	Maestro	DB 3 DBX 1.0	1.0	-	1
VlvExpulAlim	BOOL	Maestro	DB 3 DBX 2.0	1.0	-	1
VlvIzquiAlim	BOOL	Maestro	DB 3 DBX 2.1	1.0	-	1
VlvDerAlimen	BOOL	Maestro	DB 3 DBX 2.2	1.0	-	1
BlancaAlimen	BOOL	Maestro	DB 3 DBX 2.3	1.0	-	1
VerdeAlimen	BOOL	Maestro	DB 3 DBX 2.4	1.0	-	1
AmarilAlime	BOOL	Maestro	DB 3 DBX 2.5	1.0	-	1
ManAutoAlimen	BOOL	Maestro	DB 3 DBX 4.0	1.0	-	1
MarchaAlimen	BOOL	Maestro	DB 3 DBX 4.1	1.0	-	1
PreparAlimen	BOOL	Maestro	DB 3 DBX 4.2	1.0	-	1

Fig. 5.89. Conjunto de variables

5.10.4 Configurar alarmas

ProTool permite la creación de eventos de alarma. Las alarmas informan en la unidad de operación (SCADA) sobre anomalías de servicio del proceso o instalación, que está acoplada al control. Debido a que los avisos de alarma indican estados de servicio excepcionales, es necesario acusarlos (ACK).

En esta aplicación se han configurado tres eventos de alarma cuyo texto de representación será:

- “PARO DE EMERGENCIA ALIMENTADOR”. Aparece cuando el módulo “doble alimentador por gravedad” sufre una parada de emergencia a causa de la activación de la seta de emergencia.
- “PARO DE EMERGENCIA CINTA”. Aparece cuando el módulo “cinta transportadora” sufre una parada de emergencia a causa de la activación de la seta de emergencia.
- “MICROINTERRUPTOR ACTIVADO”. Aparece cuando se activa el microinterruptor de la maqueta “doble alimentador por gravedad”.

El proceso a seguir para la configuración de estas alarmas se resume en los siguientes puntos.

5.10.4.1 Puntero de área

Un puntero de área es un área de memoria en el control, definida por el usuario. El área sirve para el intercambio de datos entre el control y el SCADA. Para su configuración se siguen los siguientes pasos:

- 1) Se inserta el puntero de área. En la pantalla principal de la aplicación debe seleccionarse el objeto “Puntero de Área” y realizar la acción Clic botón izquierdo >> Puntero de área insertar (Fig. 5.90.). El tipo de puntero de área a insertar será “Alarmas”.

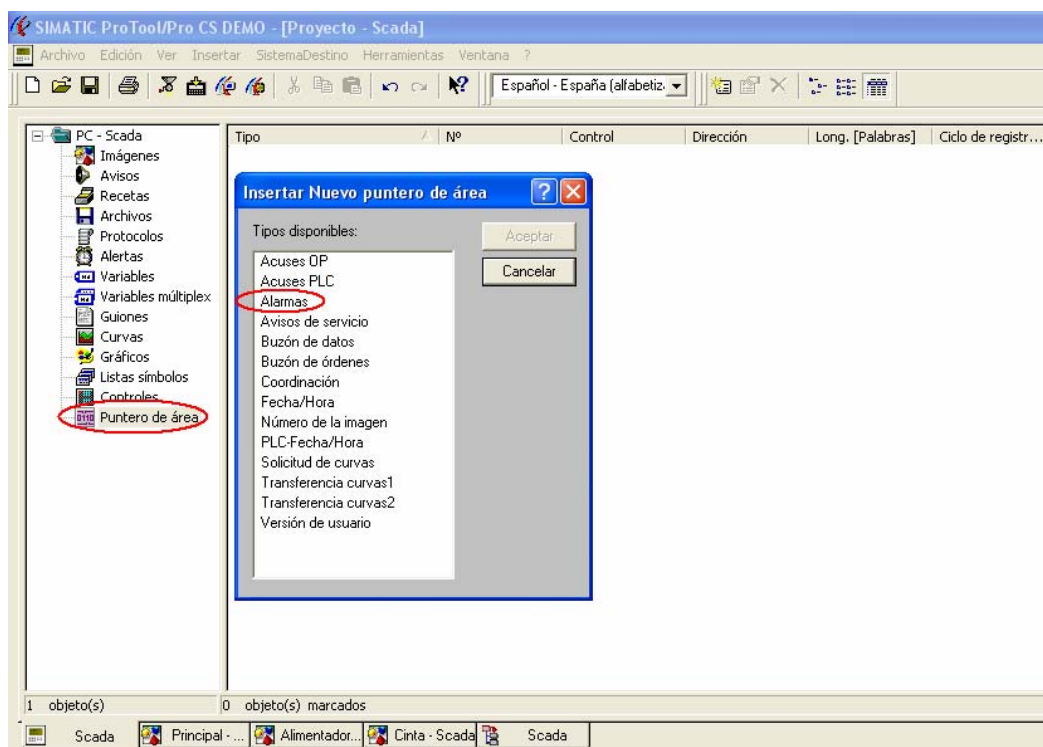


Fig. 5.90. Insertar puntero de área

- 2) Se configura el puntero de área con un control (“**Maestro**”) y con un área de datos en ese control (**DB1.DBW0**). Con esta configuración, la alarma tendrá una longitud de 1 palabra, es decir, 16 bits, lo que permitirá configurar 16 alarmas. El parámetro “Ciclo de registro” permite configurar cada cuántos ciclos de la CPU se actualiza el puntero de área (por defecto, se configura valor 1). La Fig. 5.91. muestra la configuración descrita.

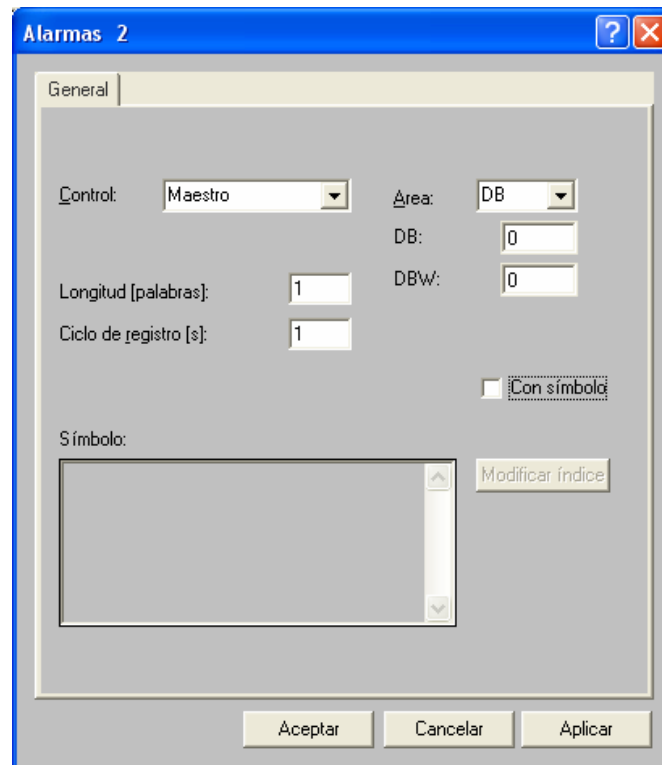


Fig. 5.91. Configurar puntero de área.

5.10.4.2 Alarmas

Una vez creada el área de datos deben de configurarse los textos de cada alarma, así como el registro dentro de dicha área asociado a cada alarma. El proceso a seguir es el siguiente:

- 1) En la pantalla principal de la aplicación se selecciona el objeto “Avisos” y dentro de este “Alarmas” (Fig. 5.92.).

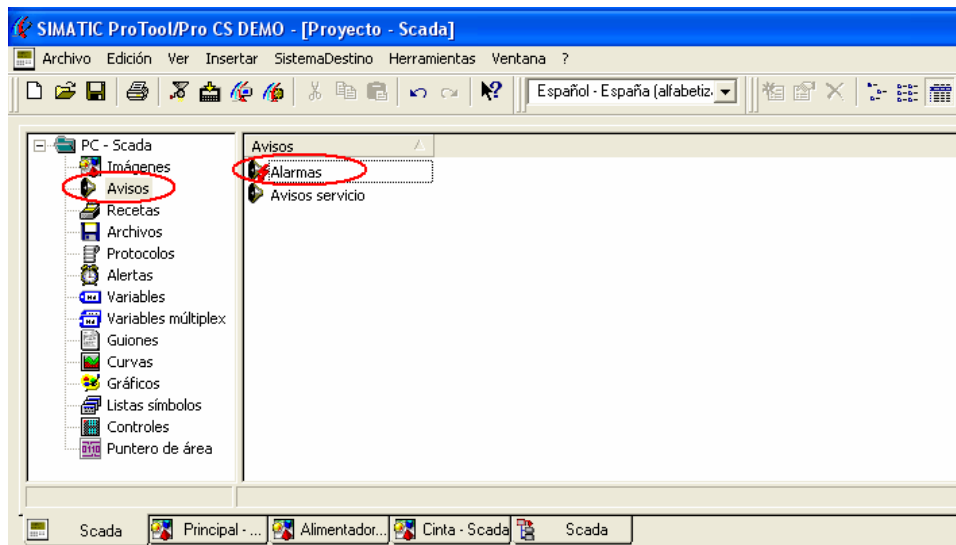


Fig. 5.92. Objeto Alarmas

- 2) La ventana resultante (Fig. 5.93.) muestra los textos asociados a las diferentes alarmas. Es importante destacar que dentro de la palabra DB1.DBW0 definida como área de datos de alarmas, el byte menos significativo del área de datos, es decir, DB1.DBB0 se representa en los registros 09....16; mientras que el byte más significativo (DB1.DBB1) está representado en los registros 01....08.

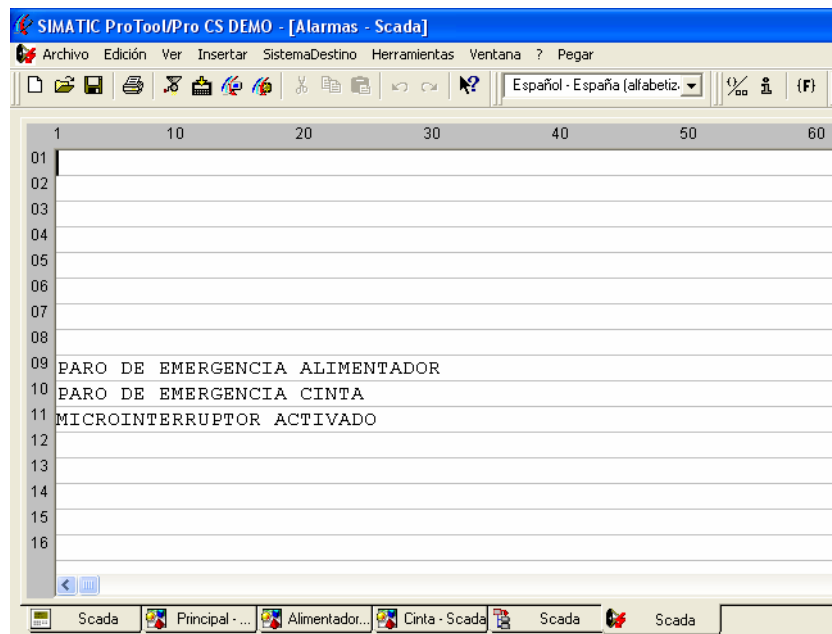


Fig. 5.93. Configuración de alarmas

- 3) Una vez creadas las diferentes alarmas es preciso asociar el evento que activará la alarma al registro correspondiente en el control. Esto se realizará en

la programación del control, es decir, en la programación del equipo “Maestro”. El código de programa encargado de la activación de cada una de las alarmas está incluido en el bloque OB1 y se muestra en la Fig. 5.94.

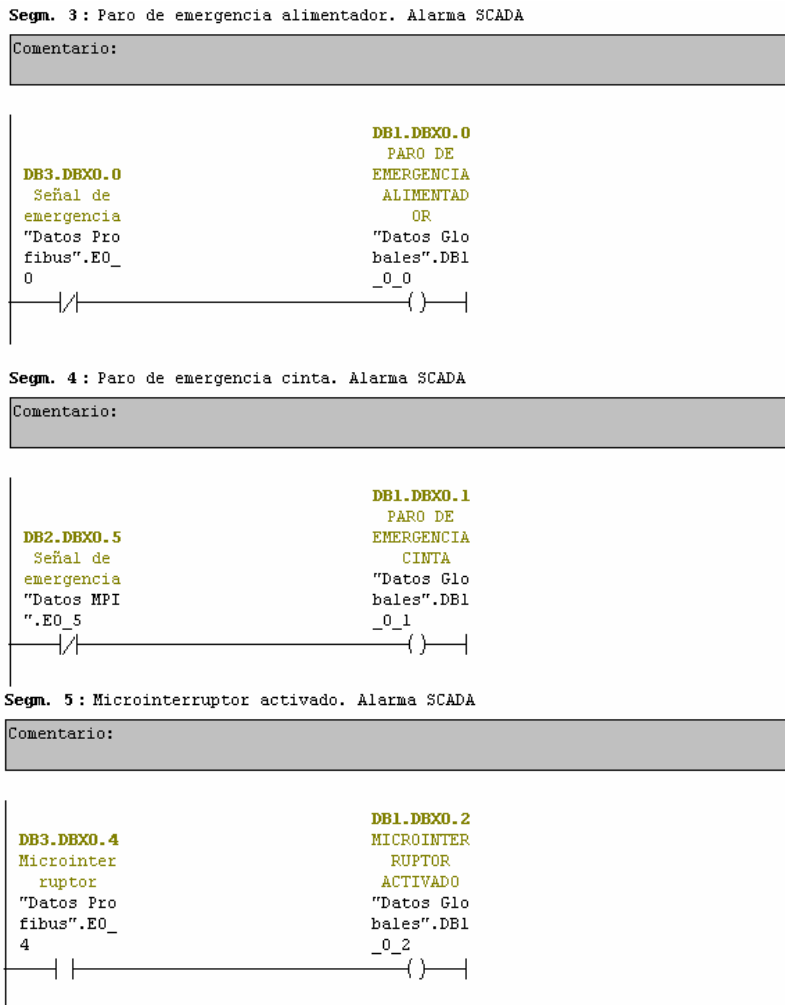


Fig. 5.94. Código de programa en “Maestro” para activación de alarmas

5.11 Transferencia de la aplicación a los equipos

Todo lo realizado hasta ahora se ha hecho a nivel “**offline**”, es decir, se ha configurado una aplicación con el software SIMATIC, pero en ningún momento esta aplicación ha sido transferida a los equipos que la componen. Por tanto, el siguiente paso es introducir en cada uno de los equipos las configuraciones y parámetros que a lo largo de los puntos anteriores se han ido definiendo.

5.11.1 Ajustar interface PG/PC

Los ajustes que se efectúen aquí determinan la comunicación entre el PC y el sistema de automatización. Esta comunicación se realizará a través del cable de comunicación PC Adapter, ajustado conforme se describió en el Apartado 5.9.2., conectado al bus PROFIBUS DP. Por lo tanto, se seguirán los siguientes pasos:

- 1) En el “Panel de control” de Windows, se selecciona el icono “Ajustar interface PG/PC”.



- 2) Se abrirá la ventana “Ajustar interface PG/PC”, donde aparecen los siguientes apartados:
 - Punto de acceso de la aplicación. Aquí se visualizan los puntos de acceso existentes. Las distintas aplicaciones acceden al interface mediante el nombre de un punto de acceso, es decir, dependiendo del software desde el cual se quiera realizar la comunicación se seleccionará un punto u otro.
 - **SIMATIC Manager y ProTool Pro: S7ONLINE (STEP 7).**
 - **STEP 7 Microwin: Micro/Win.**

En la Fig. 5.95. se observan ambas opciones de configuración.

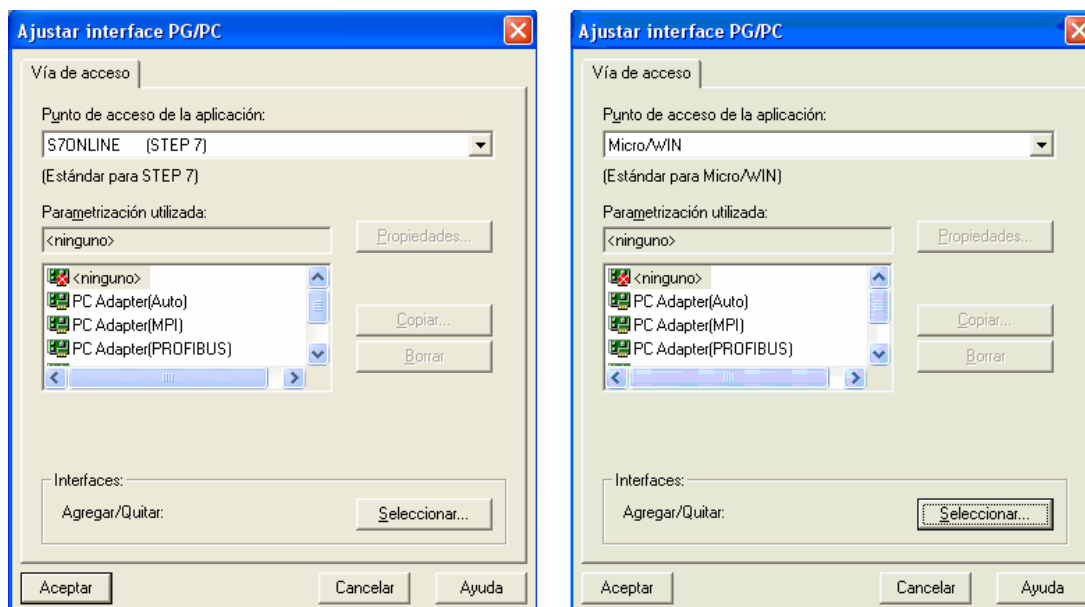


Fig. 5.95. Puntos de acceso a la aplicación

- Parametrización utilizada. Aquí se visualiza la parametrización de interface que se ha asignado al punto de acceso indicado en el campo “Punto de acceso de la aplicación” y que de esta forma determina la vía de acceso de una aplicación. Debajo se muestran todas las parametrizaciones de interface disponibles y una breve descripción de la parametrización seleccionada. En este apartado se seleccionará la opción **PC Adapter (PROFIBUS)**. Los diferentes parámetros de la conexión se configuran seleccionando la ficha “Propiedades”.
 - **Dirección PROFIBUS: 4 (dirección de SCADA).**
 - **Velocidad de transferencia: 19.2 Kbit/s (velocidad de bus PROFIBUS DP).**
 - **Perfil: DP.**
 - **Conexión local: COM1. Velocidad de transferencia: 38400 bit/s.**

Los demás parámetros conservarán sus valores por defecto. La Fig. 5.96. representa esta configuración.

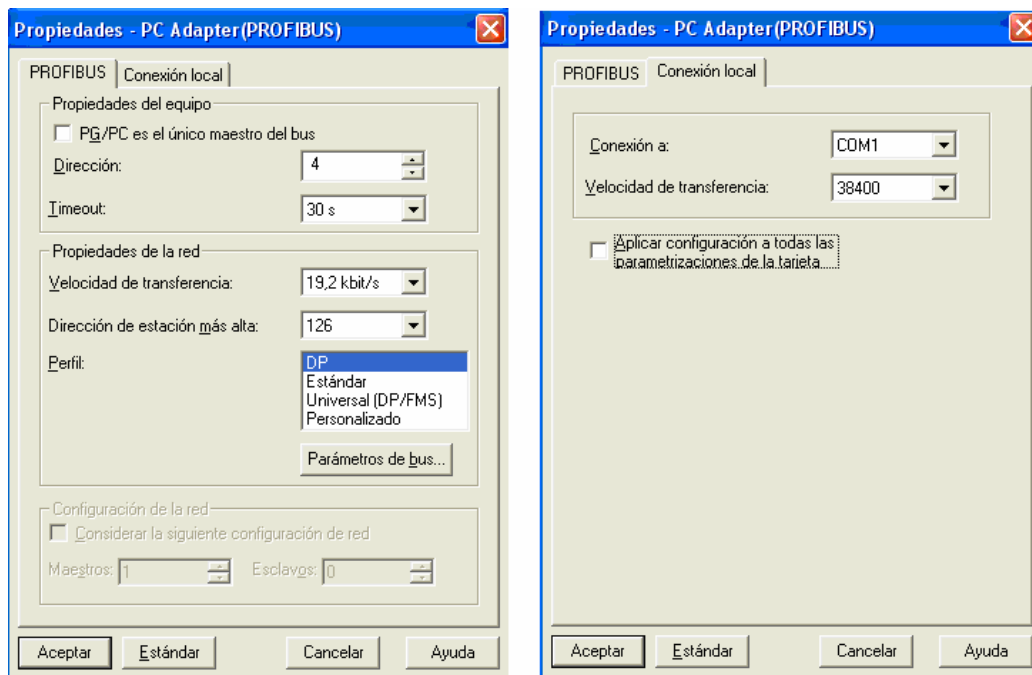


Fig. 5.96. Parametrización del interface PROFIBUS

Finalmente quedarán configurados dos enlaces de comunicación:

- **S7ONLINE (STEP 7) → PC Adapter (PROFIBUS)**. Se seleccionará este enlace cuando se desee comunicar desde *SIMATIC Manager* o *ProTool Pro*.

- **Micro/WIN → PC Adapter (PROFIBUS).** Se seleccionará para comunicación desde *STEP 7 Microwin*.

La Fig. 5.97. muestra ambas configuraciones.

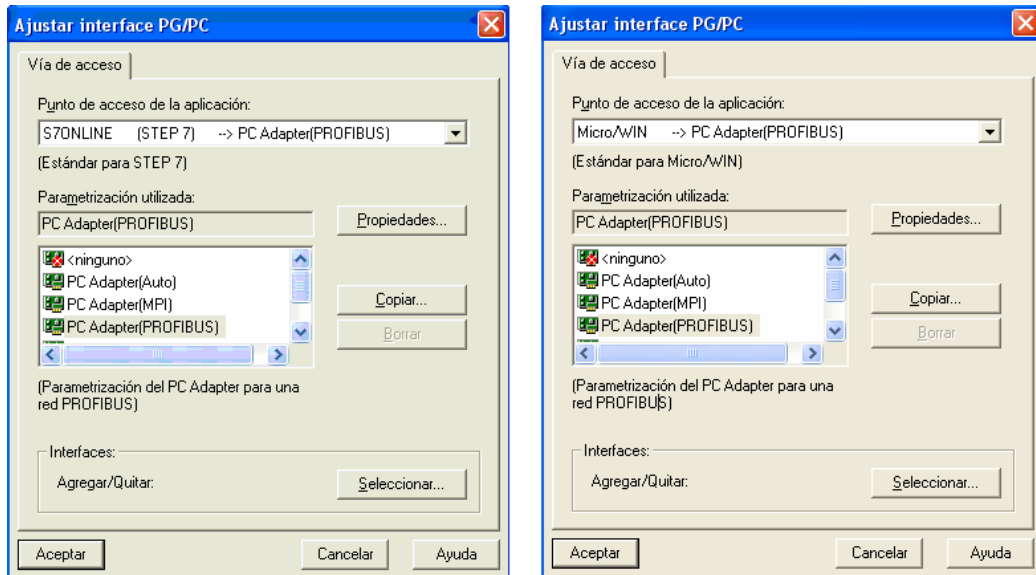


Fig. 5.97. Enlaces configurados para interface PG/PC

5.11.2 Transferir configuración “Maestro”

La transferencia de la configuración al equipo “Maestro” (CPU314C-2DP) se realizará desde el software *SIMATIC Manager*.

- 1) Se seleccionará en “Ajustar interface PG/PC” el enlace configurado como **S7ONLINE (STEP 7) → PC Adapter (PROFIBUS)** (Fig. 5.97.).
- 2) En la pantalla principal de la aplicación, en el software *SIMATIC Manager*, se seleccionará el equipo “Maestro” y a continuación se pulsará el botón “Cargar en CPU”. En la Fig. 5.98. se observa este paso.

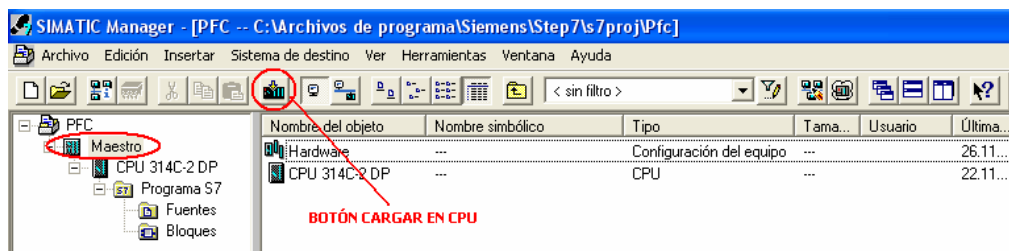


Fig. 5.98. Acción cargar en CPU

- 3) A continuación, como la dirección PROFIBUS DP de la CPU por defecto es 2, y en la presente aplicación se ha configurado como 1, no se cargará directamente la configuración en la CPU, sino que el software mostrará como posibles estaciones accesibles la CPU 314C-2DP, dirección PROFIBUS 2. Al seleccionar esta CPU y cargar la configuración, la dirección PROFIBUS se modificará a valor 1.

Al realizar esta operación se están transfiriendo todos los datos configurados con el software *SIMATIC Manager*, es decir, todo lo configurado con las herramientas *HWConfig*, *NetPro* y *KOP/AWL/FUP Editor*. Por tanto se transfieren las configuraciones hardware y software del equipo “Maestro” y la configuración de las redes de comunicación.

5.11.3 Transferir configuración “Esclavo DP”

La transferencia de la configuración del equipo “Esclavo DP” (CPU 224) se realizará desde el software *STEP 7 Microwin32*.

- 1) Se seleccionará en “Ajustar interface PG/PC” el enlace configurado como **Micro/WIN → PC Adapter (PROFIBUS)** (Fig. 5.97.).
- 2) En la pantalla de configuración del equipo “Esclavo DP” se selecciona la pestaña “Comunicación”. En la ventana resultante (Fig. 5.99.) se mostrarán los parámetros del enlace de comunicación, así como la opción “Actualizar”.

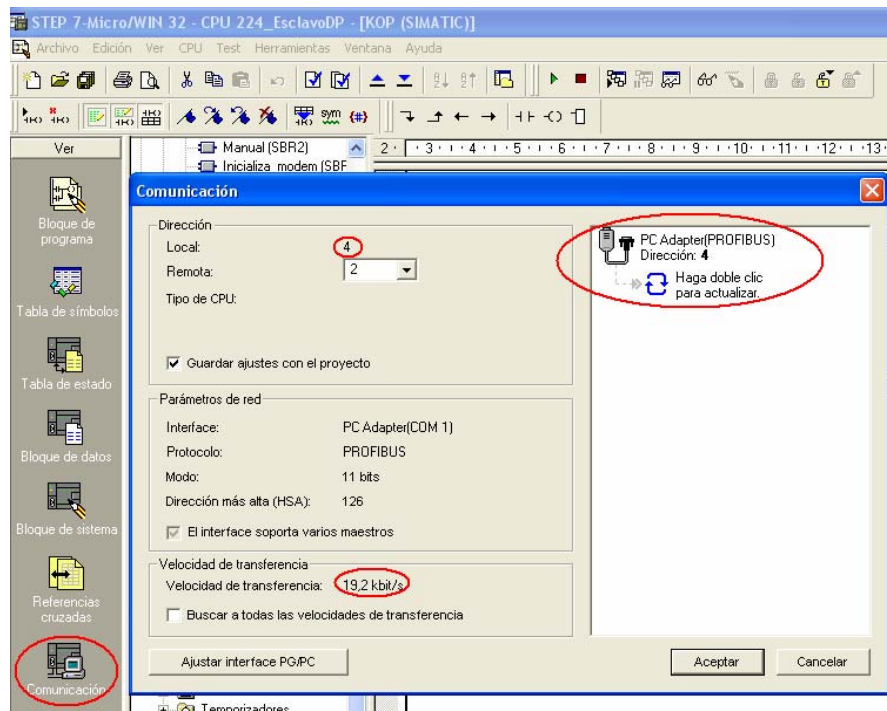


Fig. 5.99. Parámetros del enlace de comunicación

- 3) A continuación debe seleccionarse la opción “Actualizar”. Con esta acción, el software buscará en la red PROFIBUS DP las estaciones susceptibles de ser configuradas, en nuestro caso, dirección PROFIBUS 2 (“Esclavo DP”) y dirección PROFIBUS 3 (“Esclavo MPI”). Las estaciones que no son configuradas con este software aparecerán bajo el nombre “Desconocido”. En la Fig. 5.100. se observa el resultado de la acción “Actualizar”.

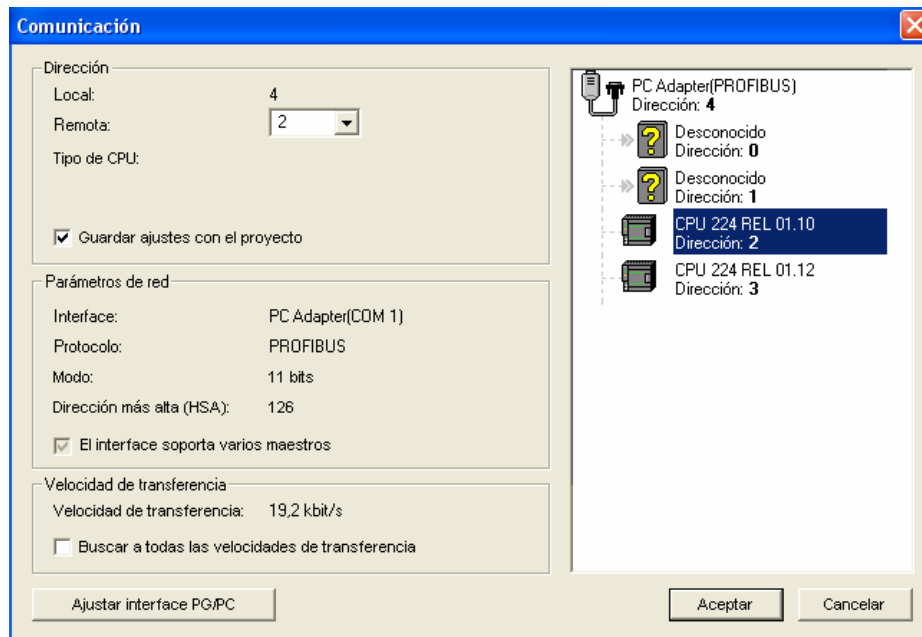


Fig. 5.100. Estaciones accesibles una vez configurada la comunicación

- 4) Se selecciona la estación **CPU 224 Dirección 2**. Esta CPU será configurada como “Esclavo DP”.
- 5) Tras aceptar se establece el enlace entre MicroWin y la CPU citada. A continuación se selecciona el icono “Cargar en CPU” (📁) para transferir la configuración.

5.11.4 Transferir configuración “Esclavo MPI”

La transferencia de la configuración del equipo “Esclavo MPI” (CPU 224) se realizará desde el software *STEP 7 Microwin*. Se seguirán los pasos descritos en el *Apartado 5.11.3. “Transferir configuración Esclavo DP”*, con la diferencia de que se seleccionará la **CPU 224 Dirección 3**.

Capítulo 6

Experimentos realizados

6.1 Introducción

En este capítulo se describirán una serie de experimentos que se han realizado con el fin de validar las configuraciones que anteriormente se han descrito (velocidades de transferencia, coherencia de datos, etc.).

6.2 Experimento 1: Comunicación PROFIBUS

El objetivo de este experimento es comprobar si el intercambio de datos entre equipos vía PROFIBUS se realiza correctamente. Revelará si se ha montado bien el cable y si los parámetros de configuración escogidos para el bus PROFIBUS son los adecuados.

6.2.1 Configuración

Se establecen los siguientes puntos de configuración:

- 1) Se configurará una CPU 314C-2DP como “Maestro” de un bus PROFIBUS DP.
- 2) Se configurará una CPU 224 como “Esclavo DP” del equipo anterior.
- 3) Los datos a enviar / recibir en ambos equipos se tomarán directamente de las entradas / salidas de las CPUs.

Para llevar a cabo los puntos 1) y 2) se seguirán los pasos comentados en **Apartado 5.5 “Configuración equipo Maestro”** y **Apartado 5.6 “Configuración equipo Esclavo DP”**.

La realización del punto 3) supone programar ambas CPUs para que el movimiento de datos cumpla las condiciones descritas.

- Para el equipo “Maestro” el código de programa se muestra en la Fig. 6.1. Para este experimento, sólo es necesario incluir en el OB1 “Programa principal” los segmentos de la Fig. 6.1.
 - En el segmento 1, el byte de entradas EB124 de la CPU se transfiere al buzón emisor PROFIBUS (PAB0).
 - En el segmento 2, el byte PEB0 perteneciente al buzón receptor PROFIBUS se transfiere al byte de salidas de la CPU AB125.

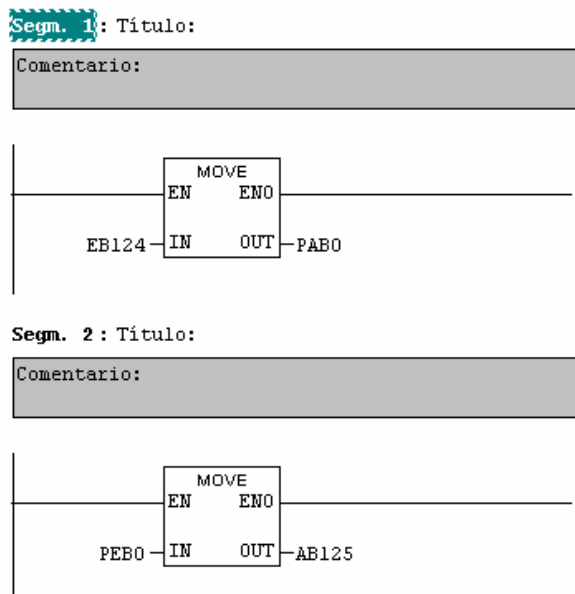


Fig. 6.1. Programación en “Maestro” para experimento 1

- Para el equipo “Esclavo DP” el código de programa se muestra en la Fig. 6.2. El código de programa pertenece a la Subrutina “Comunicación PROFIBUS”, que será el único bloque necesario para este experimento. Los datos que se reciben

desde “Maestro” se transfieren al byte AB0 del “Esclavo DP”. Los datos a enviar a “Maestro” son el byte EB0.

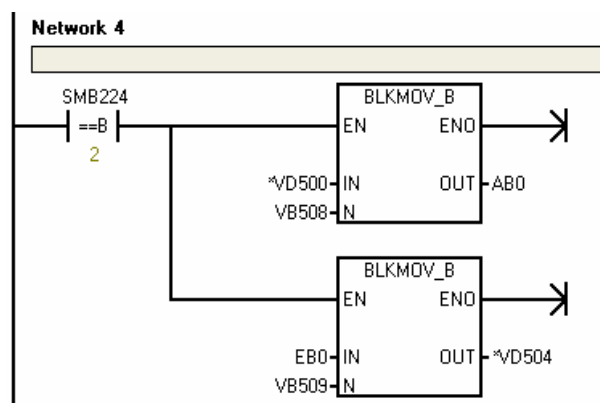


Fig. 6.2. Modificación de programa en “Esclavo DP” para experimento 1

6.2.2 Pruebas a realizar

Se realizarán dos pruebas que determinarán si el intercambio de datos se realiza de forma correcta y a la velocidad deseada.

- Prueba 1: Activar en equipo “Maestro” la entrada E124.0 (EB124) y comprobar si en el equipo “Esclavo DP” se activa la salida A0.0 (AB0).
- Prueba 2: Activar en equipo “Esclavo DP” la entrada E0.0 (EB0) y comprobar si en el equipo “Maestro” se activa la salida A125.0 (AB125).

Si una vez realizadas ambas pruebas se cumplen las condiciones descritas y se observa un tiempo de respuesta menor de 0.5 s, se determina que la conexión configurada es **adecuada**. Por tanto, se determina que una velocidad de transferencia PROFIBUS de **19.2 Kbit/s** es totalmente válida.

6.3 Experimento 2: Comunicación MPI

El objetivo de este experimento es comprobar si el intercambio de datos entre equipos vía MPI se realiza correctamente. Revelará, al igual que en el caso anterior, si se ha montado bien el cable y si los parámetros de configuración escogidos para el bus MPI son los adecuados.

6.3.1 Configuración

Se establecen los siguientes puntos de configuración:

- 1) Se configurará una CPU 314C-2DP como “Maestro” de un bus MPI.
- 2) Se configurará una CPU 224 como “Esclavo MPI” del equipo anterior.
- 3) Los datos a enviar / recibir se tomarán directamente de las entradas / salidas de las CPUs.

Para llevar a cabo los puntos 1) y 2) se seguirán los pasos comentados en *Apartado 5.5 “Configuración equipo Maestro”* y *Apartado 5.7 “Configuración equipo Esclavo MPI”*.

La realización del punto 3) supone programar ambas CPUs para que el movimiento de datos cumpla las condiciones descritas.

- Para el equipo “Maestro” el código de programa se muestra en la Fig. 6.3. Para este experimento, sólo es necesario incluir en el OB1 “Programa principal” los segmentos de la Fig. 6.3.
 - En el segmento 1, la función SFC67 “Leer datos de esclavo” lee del “Esclavo MPI” el byte de entradas EB0 y lo escribe directamente en el byte de salidas AB125 de la CPU “Maestro”.
 - En el segmento 2, la función SFC68 “Escribir datos en esclavo” escribe el byte de entradas EB124 del equipo “Maestro” directamente en el byte de salidas AB0 de la CPU “Esclavo MPI”.

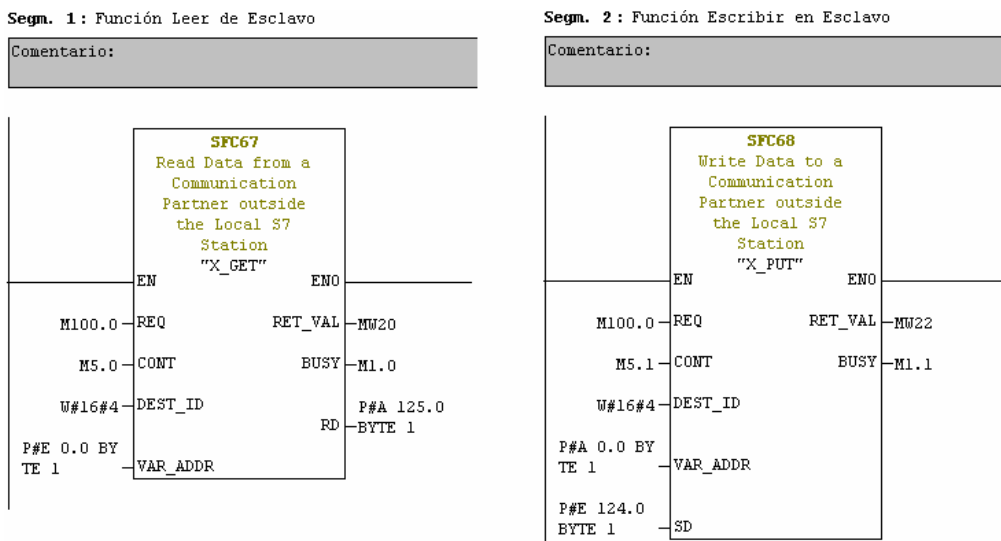


Fig. 6.3. Programación en “Maestro” para experimento 2

- Para el equipo “Esclavo MPI” no es necesario incluir ninguna programación, ya que la función de comunicación se implementa en el equipo “Maestro”. Este es el encargado de leer / escribir los datos, actuando el “Esclavo MPI” como una CPU “pasiva”.

6.3.2 Pruebas a realizar

Las pruebas a realizar en este experimento son las mismas que se describieron para el Experimento 1.

Importante

Tras realizar este experimento se determina que la velocidad de transferencia adecuada para la comunicación MPI es **187.5 Kbit/s**. Si se selecciona una velocidad de transferencia inferior (por ejemplo 19.2 Kbit/s) el tiempo de respuesta que se obtiene supera los 0.5 s, tiempo que no proporciona una lectura “continua” de datos, lo que puede acarrear que no se detecten algunos cambios de estado en los registros. Esto se debe a que el enlace de comunicación es “no configurado”, es decir, no existe un enlace directo entre ambos equipos, sino que sólo el equipo “Maestro” es el encargado de coger / poner datos en el “Esclavo MPI”.

6.4 Experimento 3: Comunicación GSM

El objetivo de este experimento es comprobar si el intercambio de datos entre el MODEM TC35 y el equipo Esclavo DP se realiza correctamente. Revelará si se han realizado bien las conexiones y si los parámetros de configuración de ambos equipos son los adecuados.

6.4.1 Configuración

Se establecen los siguientes parámetros de configuración:

- Para el equipo Esclavo DP (CPU 224): se configurará como se expuso en el Apartado 5.6.3.3 “Programación para comunicación GSM”. En resumen, se configura el puerto de comunicación integrado en la CPU 224 en modo Freeport a 9600 bit/s y se configuran tres eventos de comunicación:
 - Evento Parada de emergencia: al activar la seta de emergencia del módulo doble alimentador se envía el SMS “Parada de emergencia”.

- Evento Microinterruptor activado: al activar el microinterruptor del módulo doble alimentador se envía el SMS “Microinterruptor activado”.
- Evento “ONM”: al enviar al número de la tarjeta SIM insertada en el MODEM TC35 el mensaje “ONM”, el módulo se pondrá en marcha si se encuentra en modo Automático.
- Para el equipo MODEM TC35: se configurará como se expuso en el Apartado 5.6.2 “Configuración MODEM TC35”. En resumen, se configura el puerto de comunicación del módulo a 9600 bit/s, el número del centro de mensajes, mensajes en modo texto y se guarda el perfil configurado.

6.4.2 Pruebas a realizar

Se realizarán las siguientes pruebas para comprobar que la comunicación funciona correctamente:

- Prueba 1: se aplicará tensión al MODEM TC35. Se observará que el indicador parpadea en modo “Fallo”. Seguidamente, se pasará a estado RUN la CPU 224. Tras 5 segundos aproximadamente, se observará que el indicador del MODEM parpadea en modo “Intercambio de datos”. Con esta prueba se confirma que hay comunicación entre ambos dispositivos, ya que supone la transferencia del número de PIN de la tarjeta SIM desde la CPU al MODEM.
- Prueba 2: se activará en el módulo doble alimentador la seta de emergencia y se comprobará en el Terminal móvil de destino que se recibe el SMS “PARADA DE EMERGENCIA”.
- Prueba 3: se activará en el módulo doble alimentador el microinterruptor y se comprobará en el Terminal móvil de destino que se recibe el SMS “MICROINTERRUPTOR ACTIVADO”.
- Prueba 4: se enviará al número de la tarjeta SIM insertada en el MODEM TC35 el mensaje “ONM” y, estando el módulo en modo Automático, se comprobará si éste se pone en Marcha (ciclos continuos de extracción de piezas).
- Prueba 5: se extraerá la tarjeta SIM insertada en el MODEM TC35 y se insertará en un Terminal móvil compatible (teléfono móvil) y se comprobará si los mensajes que se han enviado / recibido se han eliminado de la memoria.

Importante

Si el enlace de datos entre MODEM y CPU no se realiza, debe de reiniciarse el MODEM y reconfigurarse. Para ello, se utilizará el comando *AT&Z* en la aplicación *Hyperterminal*, con lo que el MODEM se configurará con sus parámetros por defecto. A continuación, se debe de volver a configurar el MODEM conforme se describe en el Apartado 5.6.2.

6.5 Experimento 4: Prueba funcionamiento módulo “doble alimentador por gravedad”

Con este experimento se pretende probar la funcionalidad del módulo “doble alimentador por gravedad”, así como el control y la monitorización desde SCADA y vía GSM.

Una vez configurada la aplicación tal y como se ha indicado a lo largo del presente capítulo, se seguirán los siguientes puntos.

6.5.1 Modo automático

En este módulo, el modo de funcionamiento automático se refiere a la realización continúa de ciclos completos de extracción de piezas. Esto implica que las torres de alimentación de piezas estarán en un continuo “izquierda / derecha”, mientras que el cilindro extractor realizará un avance y un retroceso (extracción de una pieza) en las posiciones izquierda y derecha de las torres de alimentación. Una parada de emergencia producirá la caída del sistema, una notificación de alarma en SCADA y el envío del mensaje “PARADA DE EMERGENCIA” al terminal móvil. Una activación del microinterruptor producirá el envío del mensaje “MICROINTERRUPTOR ACTIVADO” al terminal móvil.

Pasos a seguir

- Se seleccionará en el panel de control local del módulo, interruptor Manual / Automático, la posición Automático.
- Se seleccionará en el panel de control remoto (SCADA), interruptor Manual / Automático, la posición Automático (Fig. 6.4.).

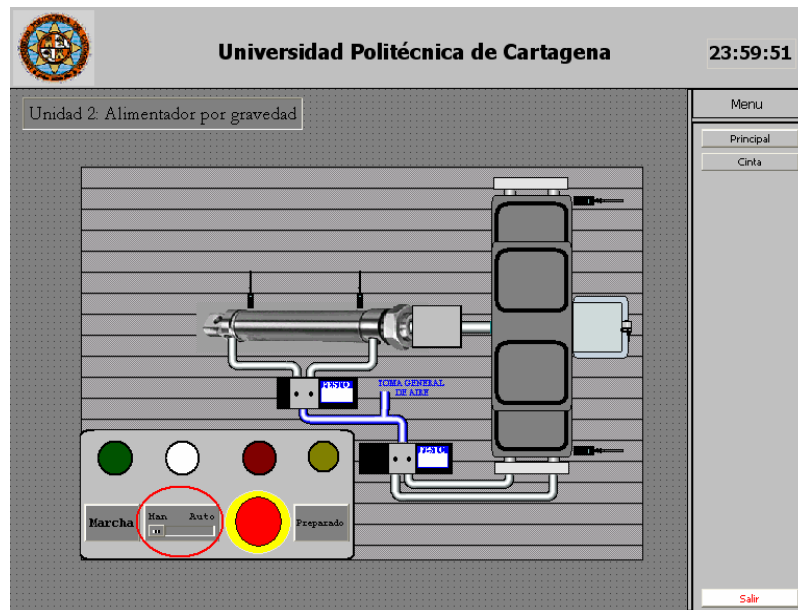


Fig. 6.4. Activar modo Automático en SCADA

- Se activará Marcha. Tres opciones, combinadas en OR:
 - Iniciar Marcha desde panel de control local. Se activará el pulsador Marcha en panel de control local del módulo.
 - Iniciar Marcha desde terminal móvil. Se enviará el mensaje “ONM” al número correspondiente a la tarjeta SIM introducida en el MODEM TC35.
 - Iniciar Marcha desde panel de control remoto (SCADA). Se activará el pulsador Marcha en SCADA (Fig. 6.5.).

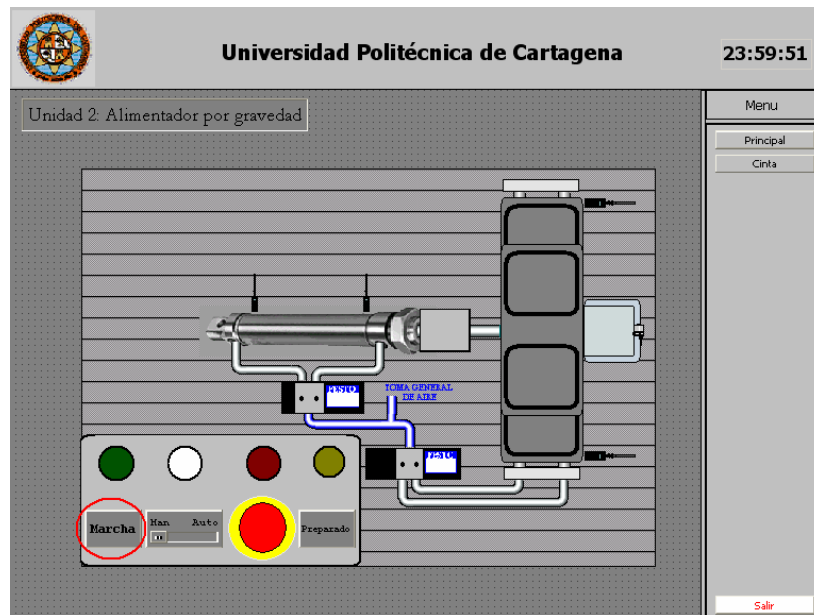


Fig. 6.5. Activar Macha en SCADA

- Una vez el módulo entra en modo Automático, las lámparas se iluminarán con el siguiente criterio:
 - Modo automático seleccionado: ON lámpara blanca.
 - Avance / retroceso del cilindro de extracción: ON lámpara verde.
 - Torres de alimentación en movimiento: ON lámpara amarilla.
 - Parada de emergencia: ON lámpara roja.
- Prueba Paro de Emergencia. En el panel de control local se activará la seta de emergencia y el módulo detendrá su funcionamiento, produciéndose las notificaciones correspondientes.
- Tras el rearme de la seta de emergencia, se activará de nuevo Marcha para continuar con el ciclo de extracción de piezas.
- Prueba Microinterruptor. Se activa el microinterruptor del módulo y se comprueba que el mensaje correspondiente es recibido.

6.5.2 Modo manual

El modo de funcionamiento manual se refiere a controlar los elementos del módulo de forma independiente. El cilindro de extracción de piezas realizará un avance / retroceso al activar el pulsador de marcha en panel de control local o en panel de control remoto. Las torres de alimentación realizarán el movimiento “izquierda / derecha”, dependiendo de la posición actual, al accionar el pulsador preparado en panel de control local o en panel de control remoto.

Pasos a seguir

- Se seleccionará en el panel de control local del módulo, interruptor Manual / Automático, la posición Manual.
- Se seleccionará en el panel de control remoto (SCADA), interruptor Manual / Automático, la posición Manual (Fig. 6.4.).
- Se activará Marcha en panel de control local o en panel de control remoto para realizar un avance / retroceso del cilindro de extracción.
- Se activará Preparado en panel de control local o en panel de control remoto para realizar un movimiento a izquierda o derecha, dependiendo de posición actual, de las torres de alimentación. (Fig. 6.6.).

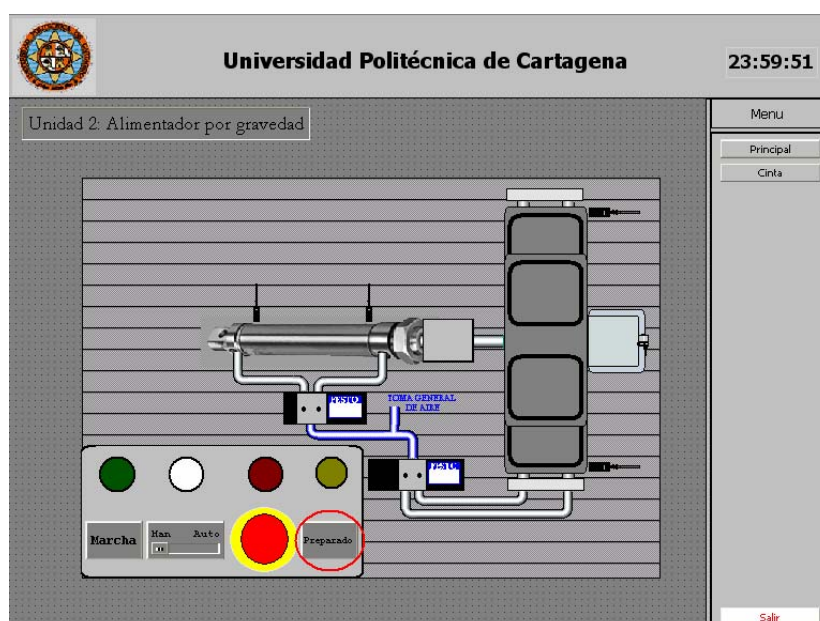


Fig. 6.6. Activar Preparado en SCADA

- Una vez el módulo en modo Manual, las lámparas se iluminarán con el siguiente criterio:
 - Modo manual seleccionado: OFF lámpara blanca.
 - Avance / retroceso del cilindro de extracción: ON lámpara verde.
 - Torres de alimentación en movimiento: ON lámpara amarilla.
 - Parada de emergencia: ON lámpara roja.
- Prueba Paro de Emergencia. En el panel de control local se activará la seta de emergencia y el módulo detendrá su funcionamiento, produciéndose las notificaciones correspondientes.
- Prueba Microinterruptor. Se activa el microinterruptor del módulo y se comprueba que el mensaje correspondiente es recibido.

6.6 Experimento 5: Prueba de funcionamiento módulo “cinta transportadora”

Con este experimento se pretende probar la funcionalidad del módulo “cinta transportadora”, así como el control y la monitorización desde SCADA. Este experimento se realizará una vez configurada la aplicación descrita a lo largo del capítulo.

El funcionamiento del módulo es el siguiente: si el interruptor Manual / Automático del panel de control local y del panel de control SCADA en Automático y accionamos Marcha en panel de control local o panel de control SCADA, la unidad realiza ciclos completos de movimiento, es decir, vaivén de izquierda a derecha y viceversa controlado por tiempo y posición de pieza. Cuando la cinta se mueve hacia la izquierda y el sensor fotoeléctrico detecta la presencia de pieza, tras 5 segundos, se activa marcha hacia derecha. Cuando la cinta se mueve hacia la derecha durante 4.5 segundos, se activa marcha hacia la izquierda. La función Manual no ha sido implementada. La parada de emergencia del sistema provocada por activación de la seta de emergencia detendrá el funcionamiento de toda la unidad y estará señalizada como una alarma en el sistema SCADA.

Pasos a seguir

- Se seleccionará en el panel de control local del módulo, interruptor Manual / Automático, la posición Automático.
- Se seleccionará en el panel de control remoto (SCADA), interruptor Manual / Automático, la posición Automático (Fig. 6.7.).



Fig. 6.7. Activar modo Automático en SCADA

- Se activará Marcha. Dos opciones, combinadas en OR:
 - Iniciar Marcha desde panel de control local. Se activará el pulsador Marcha en panel de control local del módulo.
 - Iniciar Marcha desde panel de control remoto (SCADA). Se activará el pulsador Marcha en SCADA (Fig. 6.8.).

El sistema comenzará así a realizar ciclos completos de movimiento controlados por tiempo y posición de pieza.

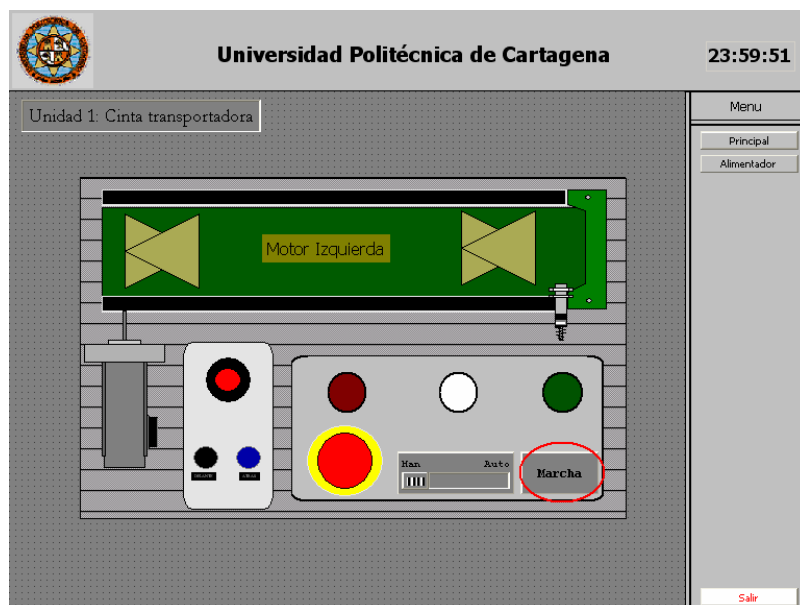


Fig. 6.8. Activar Marcha en SCADA

- Prueba Paro de Emergencia. En el panel de control local se activará la seta de emergencia y el módulo detendrá su funcionamiento, produciéndose la notificación correspondiente en SCADA.
- Tras el rearme de la seta de emergencia, se activará de nuevo Marcha para continuar con el ciclo completo de movimiento.
- Una vez el módulo en modo Automático, las lámparas se iluminarán con el siguiente criterio:
 - Modo automático seleccionado: ON lámpara blanca.
 - Cinta en movimiento: ON lámpara verde.
 - Parada de emergencia: ON lámpara roja.

Capítulo 7

Conclusiones y líneas futuras

7.1 Conclusiones

En este proyecto se ha llevado a cabo la configuración y puesta en marcha de una red de comunicación industrial basada en las tecnologías de comunicación PROFIBUS, MPI y GSM para el control y la monitorización de dos módulos de fabricación flexible.

Cabe destacar que la comunicación mediante el bus PROFIBUS DP entre autómatas programables es una tecnología fácil y sencilla de implementar, que permite la descentralización del sistema de control, dotando de esta forma al sistema de más versatilidad, más robustez y más flexibilidad. Además ofrece la ventaja de ser un bus estandarizado, por lo que la mayoría de las marcas fabricantes en el campo de la automatización comercializan productos para PROFIBUS DP.

Otro aspecto a tener en cuenta es la facilidad de ampliación y modificación de las instalaciones existentes, únicamente con la incorporación de nuevos módulos al bus

PROFIBUS y la gran flexibilidad que ofrece: es un bus de comunicación que puede trabajar tanto a nivel de campo donde se requieren altas velocidades de comunicación como a nivel de célula, donde las velocidades ya no son tan importantes y sí lo es la fiabilidad del bus.

Por otra parte, y en lo que a comunicación vía MPI se refiere, se destaca que el bus MPI no presenta tantas bazas como el bus PROFIBUS, pero en determinadas aplicaciones su implementación puede proporcionar un resultado satisfactorio, con el consiguiente ahorro económico que supone MPI frente a PROFIBUS. La gran desventaja del bus MPI es que el protocolo de transmisión que utiliza es “estándar Siemens”, lo que significa que ningún producto que no sea Siemens puede ser integrado en este tipo de bus.

En cuanto a la supervisión y control de procesos mediante comunicación vía GSM, gracias a SMS, hay que mencionar que es una alternativa a la supervisión “en planta” pero en ningún caso sería aconsejable eliminar esta última, ya que una comunicación vía GSM depende no sólo de los terminales de recepción / envío, sino también de toda la red GSM, con las posibilidades de fallo que esto conlleva.

Finalmente y en lo que a sistema SCADA se refiere, comentar que con la integración de procesos en una red de comunicación se consigue poder controlar y supervisar desde un solo SCADA todos los procesos, consiguiendo así un sistema centralizado de control.

7.2 Líneas futuras

Como posibles futuras vías de actuación se proponen las siguientes:

- Añadir varios Maestros a la red PROFIBUS DP, de forma que desde cualquier equipo Maestro sea posible acceder a los datos de cualquier Esclavo. De esta forma podrían integrarse en la red más módulos de fabricación, para así implementar un sistema de fabricación flexible integrado en una red PROFIBUS DP.
- Implementar la comunicación vía GSM en el equipo Maestro de la red, de forma que permita el control y supervisión de ambos módulos de fabricación. Para ello es necesario acoplar un módulo de comunicación RS 232 al equipo Maestro.
- Añadir al sistema de supervisión y control SCADA apartados como gráficos, curvas o históricos, que permitan controlar parámetros como número de paros de emergencia producidos, ciclos realizados, etc.

- Implementar un sistema SCADA en un terminal móvil, de forma que el mismo control y supervisión que se tiene en planta esté disponible en este terminal.

Anexo A

Datos técnicos de los equipos

Este Anexo contiene información técnica acerca de los equipos utilizados en este proyecto. Se recogen tablas de características, esquemas de conexión, croquis de dimensiones, etc. que servirán como ampliación a lo mencionado en el Capítulo 3.

A1. CPU 224 AC/DC/Relé

A1.1. Dimensiones técnicas

La Fig. A1. muestra un croquis acotado de una CPU 224.

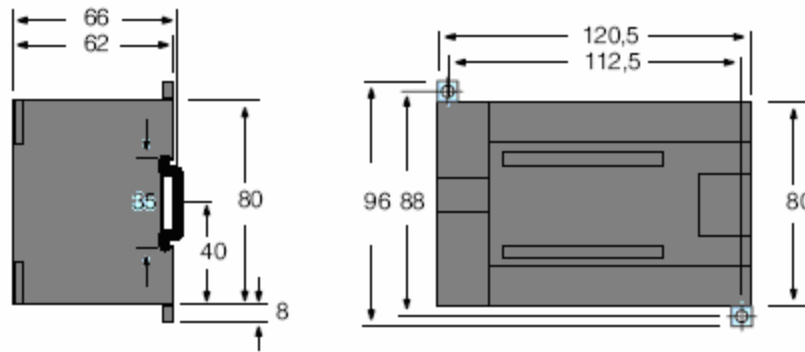


Fig. A1. Dimensiones técnicas de una CPU224.

A1.2. Disposición de terminales

Los terminales del autómata se reflejan en la Fig. A2., donde se observan los bornes de alimentación L1 y N; los bornes de alimentación para salidas 1L, 2L, 3L; los bornes de alimentación para entradas 1M, 2M; las salidas de la fuente de alimentación L+ y M; las entradas que van desde E0.0 hasta E1.5; las salidas que van desde A0.0 hasta A1.1.

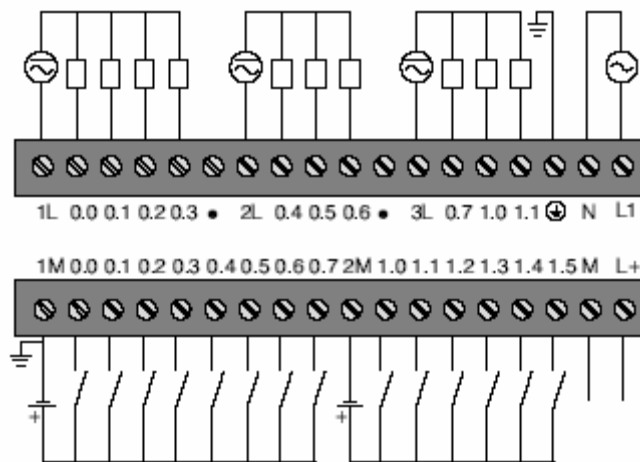


Fig. A2. Disposición de terminales de la CPU 224 AC/DC/Relé.

A1.3. Datos técnicos

En la Tabla A1. se recogen los datos técnicos de la CPU 224 AC/DC/Relé.

En la Tabla A2. se muestra el área de memoria “Marcas especiales”, con una descripción funcional de cada byte.

Datos técnicos de la CPU 224 AC/DC/Relé	
Dimensiones (l x a x p)	120.5 x 80 x 62 mm
Peso	410 g
Disipación	10 W
Características de la CPU	
Entradas digitales integradas	14 entradas
Salidas digitales integradas	10 salidas
Contadores rápidos (valor de 32 bits)	6 contadores rápidos
Salidas de impulsos	2 a 20 Khz.
Potenciómetros analógicos	2 con resolución de 8 bits
Tamaño del programa (almacenado permanentemente)	4096 palabras
Tamaño del bloque de datos (almacenado permanentemente)	2580 palabras
Nº de módulos de ampliación	7 módulos
E/S de ampliación digitales (máx.)	256 E/S
E/S analógicas (máx.)	32 entradas y 32 salidas
Marcas internas	256 bits
Temporizadores	256 temporizadores
Contadores	256 contadores
Velocidad de ejecución	Desde 0.37 µs a 400 µs por operación
Tiempo de respaldo por el condensador de alto rendimiento	Condiciones normales: 190 h Condiciones adversas: 120 h a 40°C
Comunicación integrada	
Nº puertos	1 puerto
Puerto eléctrico	RS-485
Velocidades de transferencia PPI/MPI	9.6, 19.2 y 187.5 Kbits/s
Velocidades de transferencia Freeport	0.3, 0.6, 1.2, 2.4, 4.8, 9.6, 19.2 y 38.4 Kbits/s
Longitud máx. del cable por segmento	
Hasta 38.4 Kbits/s	1200 m
187.5 Kbits/s	1000 m
Nº máx. de estaciones	
Por segmento	32 estaciones
Por red	126 estaciones
Nº máx. de maestros	32 maestros
Modo maestro PPI	Sí
Enlaces MPI	4 en total; 2 reservados: 1 para PG y 1 OP
Alimentación	
Tensión de línea (margen admisible)	AC 85 a 264 V 47 a 63 Hz
Corriente de entrada (sólo CPU)	30/100 mA a AC 240 V 60/200 mA a AC 120 V

Aislamiento	AC 1500 V
Fusible interno	2 A, 250 V, de acción lenta
Características de las entradas	
Tipo de entrada	Sumidero de corriente/fuente
Tensión de entrada máx.	DC 30 V; DC 35 V, 0.5 s
Valor nominal	DC 24 V a 4 mA
Señal 1 lógica (mín.)	DC 15 V a 2.5 mA
Señal 0 lógica (máx.)	DC 5 V a 1 mA
Características de las salidas	
Tipo de salida	Relé, contacto de baja potencia
Tensión de salida máx.	DC 5 a 30 V ó AC 5 a 250 V
Corriente de salida a señal 1	2.00 A
Corriente máx. por grupo (4/3/3)	8 A
Frecuencia de conmutación	Máx. 1 Hz
Retardo de conmutación	Máx. 10 ms
Vida útil a carga nominal	100.000 ciclos abiertos/cerrados

Tabla A1. Datos técnicos CPU 224 AC/DC/Relé.

Marcas especiales para CPU 224	
SMB0	Bits de estado. Contiene 8 bits de estado que la CPU 224 actualiza al final de cada ciclo
SMB1	Bits de estado. Contiene varios indicadores de los posibles errores
SMB2 y SMB3	Control comunicación en modo Freeport
SMB4	Desbordamiento de la cola de espera
SMB5	Estado de las entradas salidas. Condiciones de error
SMB6	Identificador de la CPU
SMB7	Reservado
SMB8 a SMB21	Registro de errores e identificadores de los módulos de ampliación
SMW22 a SMW26	Información tiempos de ciclo
SMB28 y SMB29	Potenciómetros analógicos 0 y 1
SMB30 y SMB130	Registros de control modo Freeport
SMB31 y SMW32	Control de escritura en la memoria no volátil (EEPROM)
SMB34 y SMB35	Duración de las interrupciones temporizadas
SMB36 a SMB65	Bytes de programación de los contadores rápidos HSC0, HSC1 y HSC2
SMB66 a SMB85	Control y supervisión de funciones de modulación
SMB86 a SMB94 y SMB186 a SMB194	Control de recepción de mensajes
SMB98 y SMB99	Número de errores en el bus de ampliación

SMB130 a SMB165	Bytes de programación de los contadores rápidos HSC3, HSC4 y HSC5
SMB166 a SMB194	Tabla de definición de perfiles
SMB200 a SMB299	Estado del módulo inteligente, por ejemplo EM277

Tabla A2. Marcas especiales CPU 224 AC/DC/Relé.

A2. CPU 314C-2DP

A2.1. Dimensiones técnicas

Las dimensiones de la CPU 314C-2DP se observan en la Fig. A3., expresadas en milímetros.

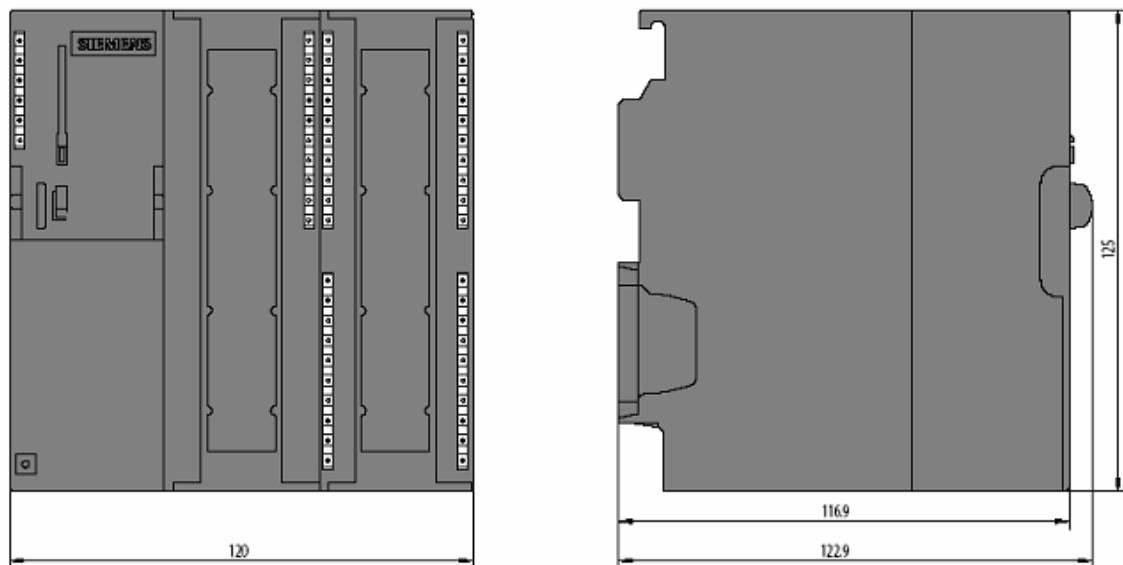


Fig. A3. Dimensiones técnicas de la CPU 314C-2DP.

A2.2. Disposición de terminales

Los terminales de la CPU 314C-2 DP quedan reflejados en la Fig. A4. Se observan las diferentes entradas/salidas digitales/analógicas integradas en la CPU. Los terminales 1L, 2L, 3L corresponden a tensión de alimentación 24VDC; los terminales 1M, 2M, 3M, 4M son terminales de masa; el terminal M_{ana} se identifica con masa analógica. Hay que destacar que cada canal analógico, excepto el correspondiente a la entrada termorresistencia, posee conexión para tensión y para intensidad.

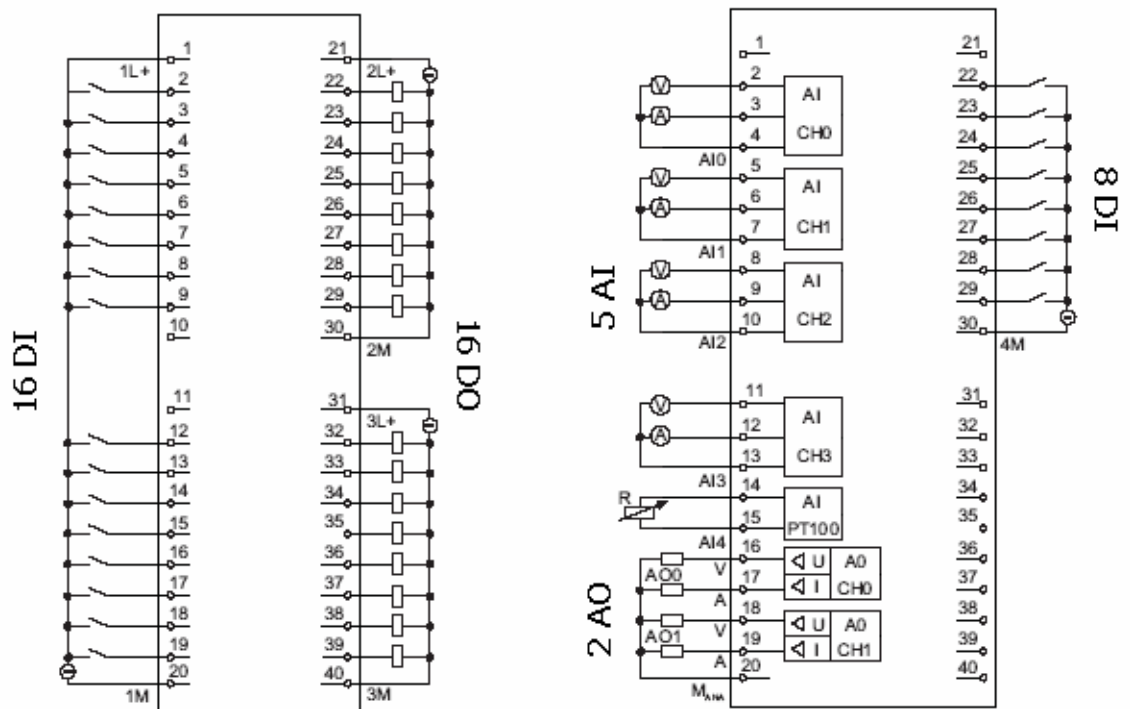


Fig. A4. Disposición de terminales CPU 314C-2DP.

A2.3. Datos técnicos

En la Tabla A3. se recoge información técnica de la CPU 314C-2DP.

Datos técnicos de la CPU 314C-2DP	
Dimensiones (l x a x p)	120 x 125 x 122.9 mm
Peso	676 g
Disipación	14 W
Características de la CPU	
Entradas digitales integradas	24 entradas
Salidas digitales integradas	16 salidas
Entradas analógicas integradas	5 entradas
Salidas analógicas integradas	2 salidas
Salidas de impulso	4 canales max. 2.5 KHz
Posicionamiento controlado	1 canal
Frecuencímetro	4 canales máx. 60 KHz
Regulación integrada	Regulación PID
Memoria de trabajo integrada	48 Kbytes.

Ampliable	NO
Memoria de carga	Enchufable (MMC)
Respaldo	Garantizado por la MMC (no necesario)
Contadores	256 contadores
Contadores IEC	Sí. Tipo: SFB
Temporizadores	256 temporizadores
Temporizadores IEC	Sí. Tipo: SFB
Marcas	256 bytes
Marcas de ciclo	1 byte de marcas
Datos locales	Máx. 510 bytes
Bloques de datos (DBs)	Máx. 127 (16 Kbytes)
Bloques OB	Máx. 16 Kbytes
Bloques FB	Máx. 128 (16 Kbytes)
Bloques FC	Máx. 128 (16 Kbytes)
Bastidores	Máx. 4 bastidores
Módulos por cada bastidor	Máx. 8; en el bastidor 3 máx. 7
Módulos de comunicación	Máx. 8
FM	Máx. 8
CP (punto a punto)	Máx. 10
CP (LAN)	
Duración de respaldo por tampón	(típ.) 6 semanas (a 40°C)
Velocidad de ejecución	Desde 0.1 μ s a 20 μ s por operación
Comunicación integrada	
Nº puertos	2 puertos
1ª Interface	
Tipo de enlace	RS 485
Funcionalidad	MPI: Sí PROFIBUS-DP: No PPI: No
Cantidad de enlaces	Máx. 12
Velocidad de transferencia	Máx. 187.5 Kbit/s
2ª Interface	
Tipo de enlace	RS 485
Funcionalidad	MPI: No PROFIBUS-DP: Sí PPI: No
Cantidad de enlaces	Máx. 12
Velocidad de transferencia	Máx. 15 Mbit/s
Alimentación	
Tensión de alimentación	24VDC nominal 20.4 a 28.8 V admisible

Consumo de corriente (en vacío)	150 mA
Potencia disipada	14 W
Características de las entradas digitales	
Tipo de entrada	Sumidero de corriente/fuente
Tensión de entrada máx.	DC 30 V; DC 35 V, 0.5 s
Valor nominal	DC 24 V
Señal 1 lógica	DC 15 V a 30 V
Señal 0 lógica	DC -3 V a 5 V
Intensidad de entrada	(típ.) 9 mA
Características de las salidas digitales	
Tipo de salida	Salida a tensión
Tensión nominal	24VDC
Corriente de salida a señal "1"	0.5 A nominal
Corriente máx. por grupo	3 A
Margen de resistencia de carga	48 Ω a 4K Ω
Carga de lámparas	Máx. 5 W
Frecuencia de conmutación	Carga ohmica: máx. 100 Hz Carga inductiva: máx. 0.5 Hz Carga de lámparas: máx. 100 Hz
Protección contra cortocircuitos de la Salida	Sí, electrónica
Características de las entradas analógicas	
Cantidad de entradas	4 canales de entrada de tensión e intensidad 1 canal de entrada de resistencia
Entrada de resistencia	
Tensión en vacío	(típ.) 2.5 V
Intensidad media	(típ.) 1.8 mA a 3.3 mA
Principio de medida	Codificación momentánea
Frecuencia máx.	Máx. 400 Hz
Resolución	11 bits + signo
Filtro de entrada	Sí. Red RC para canales 0 a 3
Márgenes de entrada	
Tensión	± 10 V/100 K Ω
Intensidad	± 20 mA/50 Ω
Resistencia	0 Ω a 600 Ω /10 M Ω
Termorresistencia	Pt100/10M Ω
Tensión máx. admisible	50 V para la entrada tensión 2.5 V para la entrada intensidad
Intensidad máx. admisible	0.5 mA para la entrada tensión 50 mA para la entrada intensidad
Características de las salidas analógicas	
Cantidad de salidas	2

Tensión nominal de carga	24 VDC
Resolución	11 bits + signo
Tiempos de estabilización	
Carga óhmica	0.6 ms
Carga capacitiva	1.0 ms
Carga inductiva	0.5 ms
Margen de salida	
Tensión	± 10 V
Intensidad	± 20 mA
Resistencia de carga	
Salida de tensión	Mín. 1 K Ω . Carga capacitiva: máx. 0.1 μ F
Salida de intensidad	Máx. 300 Ω . Carga inductiva: 0.1 mH

Tabla A3. Datos técnicos CPU 314C-2DP.

A3. Fuente de alimentación PS307-2A

A3.1. Dimensiones técnicas

Las dimensiones de la PS307-2A se observan en la Fig. A5.

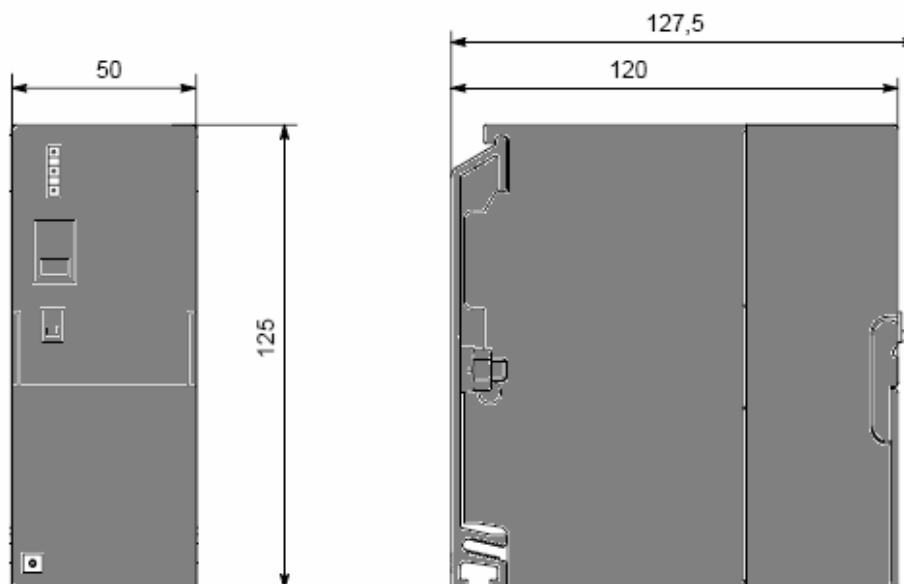


Fig. A5. Dimensiones técnicas PS307-2A.

A3.2. Datos técnicos

Los datos técnicos para esta fuente de alimentación quedan resumidos en la Tabla A4.

Datos técnicos de la fuente de alimentación PS307-2A	
Tamaño físico	
Dimensiones (a x a x p)	50 x 125 x 120 mm
Peso	420 g
Magnitudes de entrada	
Tensión de entrada	120/230 V CA
Frecuencia de red	50 Hz o 60 Hz. Rango admisible: 47 Hz a 63 Hz
Intensidad de entrada	0.5 A a 230 V 0.8 A a 120 V
Intensidad al conectar	20 A
Magnitudes de salida	
Tensión de salida	24 Vcc \pm 5 %
Intensidad de salida	2 A
Protección contra cortocircuitos	Electrónica, de 1.1 a 1.3 x In
Rizado residual	Máx. 150mVpp
Magnitudes características	
Tensión nominal del aislamiento	250 Vca
Separación eléctrica segura	Circuito SELV
Rendimiento	83 %
Potencia absorbida	58 W
Potencia disipada	Típ. 10 W

Tabla A4. Datos técnicos fuente de alimentación PS307-2A.

A4. Módulo de comunicaciones EM 277

En este apartado se describirán las propiedades del conector del módulo EM 277 y se darán una serie de datos técnicos de dicho módulo.

En la Fig. A6. se aprecian las funciones de cada uno de los pines del puerto de comunicación DB9 del módulo EM 277.

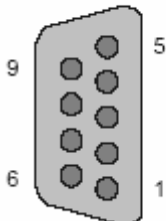
9-PIN D CONECTOR HEMBRA	PIN	DESCRIPCION
	1	Tierra del conector
	2	Retorno 24VDC, Masa (M)
	3	Señal B (Rx/D/TxD+)
	4	Petición de enviar (Nivel TTL)
	5	Retorno 5VDC
	6	Alimentación 5VDC a 90 mA
	7	Alimentación 24VDC a 120 mA máx.
	8	Señal A (Rx/D/TxD-)
	9	Sin conectar

Fig. A6. Propiedades conector modulo EM 277.

Los datos técnicos del módulo EM 277 están expuestos en la Tabla A5.

Datos técnicos del módulo de comunicaciones EM277	
Características generales	
Dimensiones (a x a x p)	71 x 80 x 62 mm
Peso	175 g
Cantidad de salidas	1, tipo RS485
Alimentación y consumo	5 VDC, 150 mA 24 VDC, 30 a 180 mA
Alimentación vía interface de comunicación	5 VDC, intensidad máx.: 90 mA 24 VDC, intensidad máx.: 120 mA
Disipación	2.5 W
Aislamiento	500 VAC (separación galvánica)
Comunicación	
Protocolos	MPI (esclavo) PROFIBUS DP (esclavo)
Velocidad PROFIBUS DP	Máx. 12 Mbit/s
Dirección estación DP	0 a 99, ajustable por hardware
Estaciones máx. por segmento	32 estaciones
Estaciones máx. en la red	126, de ellas máx. 99 EM277
Velocidad MPI	Máx. 12 Mbit/s
Conexiones MPI	6 total, 2 reservadas (1 para PG, 1 para OP)

Tabla A5. Datos técnicos modulo EM277.

A5. Módulo de comunicaciones TC35.

En este apartado se describirán las propiedades del conector del módulo TC35 y se darán una serie de datos técnicos de dicho módulo.

En la Fig. A7. se aprecian la funciones de cada uno de los pines del puerto de comunicación DB9 del módulo TC35.

9-PIN D CONECTOR HEMBRA		PIN	SEÑAL	I/O	DESCRIPCION
	1	DCD	O	Detección de portadora	
	2	RXD	O	Recibir datos	
	3	TXD	I	Transmitir datos	
	4	DTR	I	Terminal de datos listo	
	5	GND	-	Masa	
	6	DSR	O	Equipo de datos listo	
	7	RTS	I	Solicitud de envío	
	8	CTS	O	Libre para envío	
	9	RI	O	Indicador de llamada	

Fig. A7. Propiedades conector modulo TC35.

A continuación, en la Tabla A6. se indican las características técnicas del módulo TC35.

Datos técnicos del módulo de comunicaciones TC35	
Características generales	
Dimensiones (a x a x p)	65 x 74 x 33 mm
Peso	130 g
Cantidad de salidas	1, tipo RS232
Alimentación	+8V.....+30V DC
Funcionamiento de la salida	Clase 4 (2W) para GSM900 Clase 1 (1W) para GSM1800
Control y programación	Mediante comandos AT(*)
Comunicación	
Red de comunicación	GSM900 / GSM1800
Soportes	Datos, voz, SMS y fax
Método envío SMS	Punto a punto MT y MO Modo texto o PDU

Tabla A6. Datos técnicos modulo TC35.

A6. Cable PC/PPI.

Los datos técnicos para el cable de comunicación PC/PPI quedan expuestos en la Tabla A7.

Datos técnicos del cable PC/PPI	
Características generales	
Tensión de alimentación	DC 14.4 a 28.8 V
Corriente de alimentación	Máx. 50 mA RMS
Retardo al cambio de sentido: bit de arranque recibido en RS232 a bit de arranque enviado a RS485	Máx. 1.2 μ s
Retardo del cambio de sentido: bit de parada recibido en RS232 a transmisión inhibida en RS485	1.4 caracteres por máx. Según parámetros transmisión cálculo de tiempo
Retardo de propagación	Máx. 4 μ s, RS485 a RS232 Máx. 1.2 μ s, RS232 a RS485
Aislamiento	DC500V
Características eléctricas del enlace RS485	
Margen de tensión en modo común	-7V a +12V, 1 s 3V RMS continuo
Impedancia de entrada del receptor	Mín. 5.4 K Ω , incluyendo cierre
Cierre/polarización	10 K Ω a +5V en B, pin 3 PROFIBUS 10 K Ω a GND en A, pin 8 PROFIBUS
Umbral/sensibilidad del receptor	+/- 0.2V, 60mV (histéresis típ.)
Tensión de salida diferencial del transmisor	Mín. 2V, RL = 100 Ω Mín. 1.5V, RL = 54 Ω
Características eléctricas del enlace RS232	
Impedancia de entrada del receptor	Mín. 3 K Ω
Umbral/sensibilidad del receptor	+/- 0.8V bajo, máx. 2.4V alto, Histéresis típica 0.5V
Tensión de salida diferencial del transmisor	Mín. +/- 5V, RL = 3 K Ω

Tabla A7. Datos técnicos cable PC/PPI.

A7. Cable PC Adapter.

Los datos técnicos para el cable de comunicación PC Adapter quedan expuestos en la Tabla A8.

Datos técnicos cable PC Adapter	
Características generales	
Tensión de alimentación	24 V ó 5 V DC, alimentación a través de interface MPI/DP
Corriente de alimentación	Máx. 90 mA, típ. 50 mA
Sobreintensida al conectar	Máx. 0.5 A; 14 ms
Grado de protección	IP20

Nivel sonoro	<45 dB (sin ventilador)
Velocidad de transferencia enlace RS232	19.2 Kbit/s / 38.4 Kbit/s, ajustable con selector
Velocidad de transferencia enlace MPI/DP	Máx. 1.5 Mbit/s

Tabla A8. Datos técnicos cable PC Adapter

A8. Cable PROFIBUS.

Los datos técnicos para este tipo de cable se encuentran recogidos en la Tabla A9.

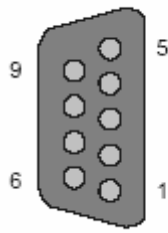
Datos técnicos del cable PROFIBUS FC Standard	
Campos de aplicación	Uso universal
Atenuación	
Con 16 MHz	≤ 42 dB/Km
Con 4 MHz	≤ 22 dB/Km
Con 9.6 KHz	≤ 2.5 dB/Km
Impedancia característica	
Con 9.6 KHz	270 ± 27 Ω
Con 38.4 KHz	185 ± 18.5 Ω
Con 3 a 20 MHz	150 ± 15 Ω
Valor nominal	150 Ω
Resistencia de bucle	≤ 110 Ω/Km
Resistencia de pantalla	≤ 9.5 Ω/Km
Capacidad con 1 KHz	Aprox. 28.5 nF/Km
Tensión de empleo (valor eficaz)	≤ 100 V
Cubierta	
Material	PVC
Diámetro	8.0 ± 0.4 mm
Color	Violeta
Temperatura de empleo	+40°C.....+60°C
Radio de curvatura	
Única	≥ 75 mm
Múltiple	≥ 150 mm
Fuerza de tracción admisible	≤ 100 N
Peso	76 Kg/Km

Tabla A9. Datos técnicos cable PROFIBUS FC Standard.

A9. Conector PROFIBUS.

La Fig. A8. representa la asignación de pines para este conector.

9-PIN D CONECTOR MACHO



PIN	SEÑAL	DESCRIPCION
1	*	No ocupada
2	*	No ocupada
3	B(RxD/TxD-P)	Cable de datos B (Receive/Transmit Data P)
4	*	No ocupada
5	M5V2 (DGND)	Potencial de referencia de datos (Data Ground)
6	P5V2 (VP)	Tensión de alimentación de +5V (Voltage Plus)
7	*	No ocupada
8	A(RxD/TxD-N)	Cable de datos A (Receive/Transmit Data N)
9	*	No ocupada

Fig. A8. Asignación de pines para conector PROFIBUS.

En la Tabla A10. quedan representados los datos técnicos del conector PROFIBUS.

Datos técnicos del conector PROFIBUS bus RS485	
Datos mecánicos	
Dimensiones (a x a x p)	15.8 x 54 x 34 mm
Peso	40 g
Conector hembra PG	Depende de modelo
Grado de protección	IP20
Salida del cable	A 90º
Datos funcionales	
Velocidad de transmisión	9.6 Kbits/s.....12Mbits/s
Resistencia terminadora	Combinación de resistencia integrada y función de seccionamiento activable
Puertos	
Estación PROFIBUS	Conector Sub-D de 9 polos
Cable de bus PROFIBUS	4 bornes en fila para hilos hasta 1.5 mm ²
Tensión de alimentación	4.75 a 5.25 VDC
Consumo	Máx. 5 mA
Temperatura de empleo	0°C a +60°C

Tabla A10. Datos técnicos conector PROFIBUS.

Capítulo 8

Bibliografía y referencias

Bibliografía y referencias

- [1] PFC Automatización e implementación de sistemas SCADA de unidades funcionales de sistemas de fabricación flexible
Autor: Juan José Ruiz Puche, Año 2002.

- [2] Documentación asignatura Automatización Industrial
Asignatura 2º ITI Electrónica, UPCT. Año 2004.

- [3] Sistema de automatización S7-200
Autor: Siemens. Referencia: 6ES7298-8FA21-8DH0.

- [4] Manuales de formación SITRAIN S7 300/400
Autor: Siemens. CD SITRAIN 300/400.

- [5] Manuales de formación SITRAIN – NET Comunicación Industrial
Autor: Siemens. CD SITRAIN-NET Comunicación Industrial Siemens.

- [6] Manual de formación para soluciones generales en automatización Totally Integrated Automation (TIA)
Autor: Siemens. Edición 02/2002.
- [7] Manual de programación con Step 7 MicroWin 32
Autor: Siemens.
- [8] Manual de software ProTool Pro CS
Autor: Siemens.
- [9] Manual de software Administrador SIMATIC S7
Autor: Siemens.
- [10] Control electrónico mediante telefonía móvil digital basada en la red GSM
Revista Tecnología y Desarrollo. Volumen II, Año 2004.
- [11] <http://support.automation.siemens.com>
- [12] <http://www.infopl.net>