

03/01 - STEP 7 Micro/WIN 32 V3.1.0.31

Autor: José A. Serantes Romero
jasero@edu.xunta.es

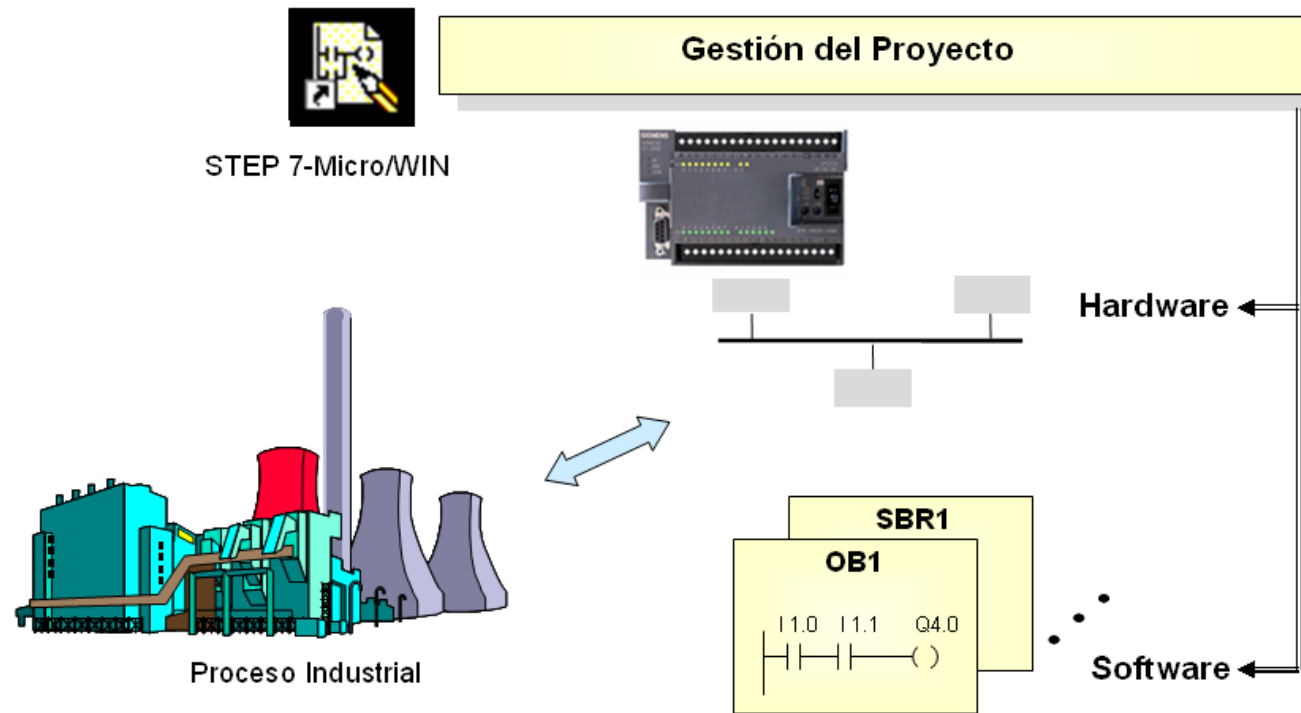
Del proceso industrial al proyecto (I)

- Punto de partida en el estudio de un proceso industrial:
 - Dividir el proceso en un conjunto de secciones y subsecciones.
 - De todas y cada una de ellas han de conocerse:
 - Respecto a la interacción entre ellas
 - Cuáles interactúan con cuáles.
 - Cómo interactúan entre sí.
 - Respecto a la dependencia
 - Cuáles dependen de cuáles.
 - Cómo es esa dependencia.

Del proceso industrial al proyecto (II)

- Proyecto de automatización de un proceso industrial:
 - Para automatizar es necesario:
 - Conocer el estado del proceso.
 - Parámetros de cada una de sus secciones y sub-secciones.
 - Será necesario enumerar:
 - » Cantidad de parámetros digitales a medir en el proceso.
 - » Cantidad de parámetros analógicos a medir en el proceso.
 - » Cantidad y tipo de consignas del operario.
 - » ...
 - Interpretar esos estados y manipular adecuadamente esa información.
 - Almacenamiento.
 - Operaciones combinadas.
 - » Lógicas.
 - » Aritméticas.
 - Lazos de regulación.
 - ...
 - Actuar sobre el estado del proceso.
 - Modificar el estado de cada una de sus secciones y sub-secciones.

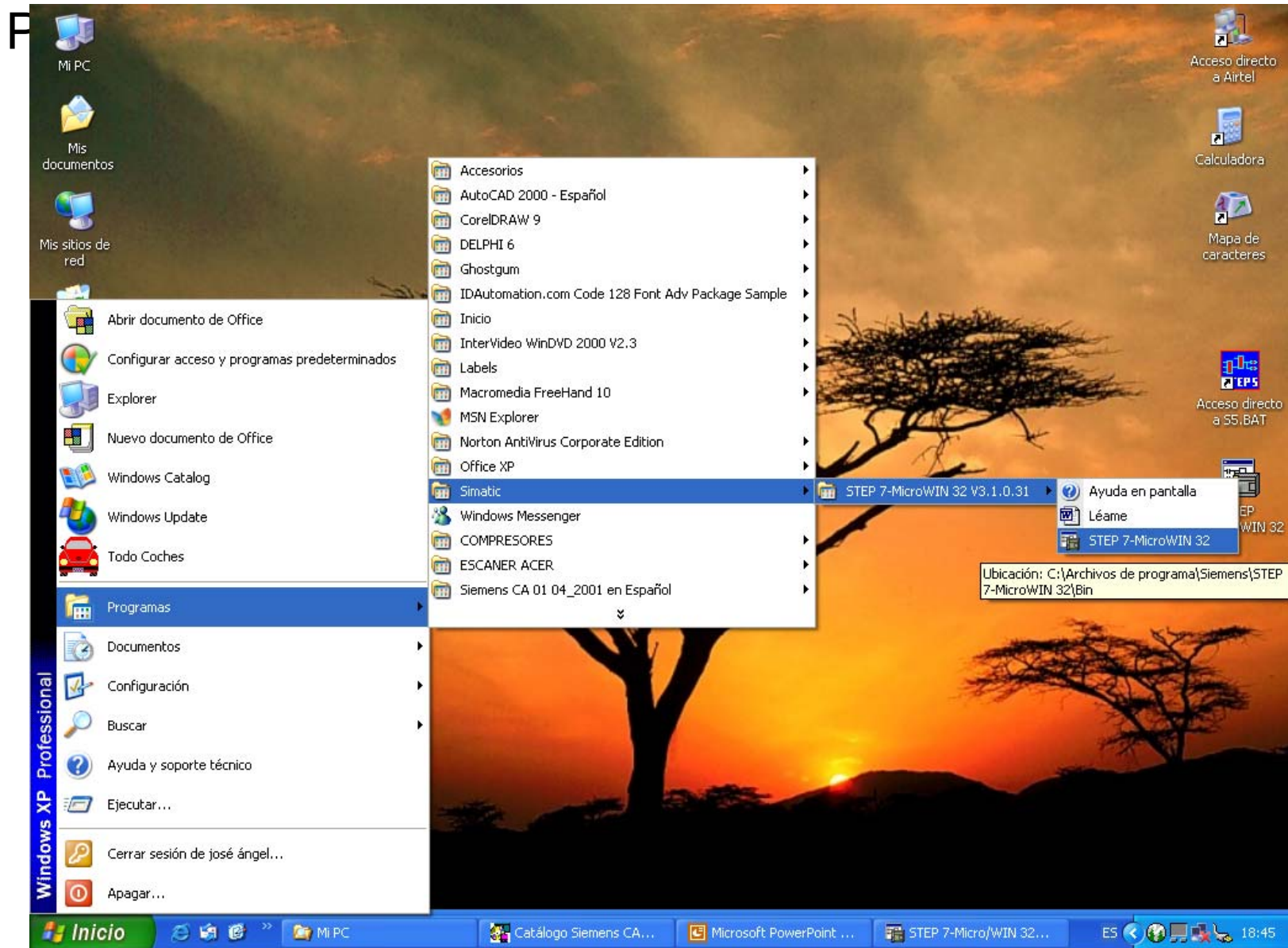
Del proceso industrial al proyecto (III)



- Necesidades para la implantación de un proceso industrial:
 - Hardware
 - Número y tipo de entradas y salidas.
 - Número y tipo de módulos.
 - Capacidad y tipo de CPUs.
 - Sistemas de diálogo hombre – máquina.
 - Redes
 - Software
 - Estructuración del programa.
 - Almacenamiento y gestión interna de información.
 - Datos de configuración.
 - Datos de comunicación.
 - Documentación
 - Documentación del proyecto y el programa.
 - Mantenimiento.
 - Utilización.

STEP 7 Micro/WIN 32 V3.1.0.31

- Ejecución de la aplicación después de su instalación en el



Comenzar con STEP 7 Micro - WIN

STEP 7-Micro/WIN 32 - Proyecto1

Archivo Edición Ver CPU Test Herramientas Ventana Ayuda

Ver

- Bloque de programa
- Tabla de símbolos
- Tabla de estado
- Bloque de datos
- Bloque de sistema
- Referencias cruzadas
- Comunicación

Herramientas

Proyecto1 (CPU 221)

- Bloque de programa
 - PRINCIPAL (OB1)
 - SBR_0 (SBR0)
 - INT_0 (INT0)
- Tabla de símbolos
- Tabla de estado
- Bloque de datos
- Bloque de sistema
- Referencias cruzadas
- Comunicación
- Operaciones
 - Aritmética en coma fija
 - Aritmética en coma flotante
 - Comparación
 - Comunicación
 - Contadores
 - Control del programa
 - Conversión
 - Desplazamiento/rotación
 - Interrupción
 - Operaciones lógicas
 - Operaciones lógicas con bits
 - Reloj
 - Tabla
 - Temporizadores
 - Transferencia
 - Subrutinas

KOP (SIMATIC)

Nombre	Tipo var.	Tipo de datos	Comentario
	TEMP		
	TEMP		
	TEMP		
	TEMP		

Network 1 TÍTULO DE SEGMENTO (una línea)

Network 2

Network 3

Network 4

Network 5

PRINCIPAL / SBR_0 / INT_0 /

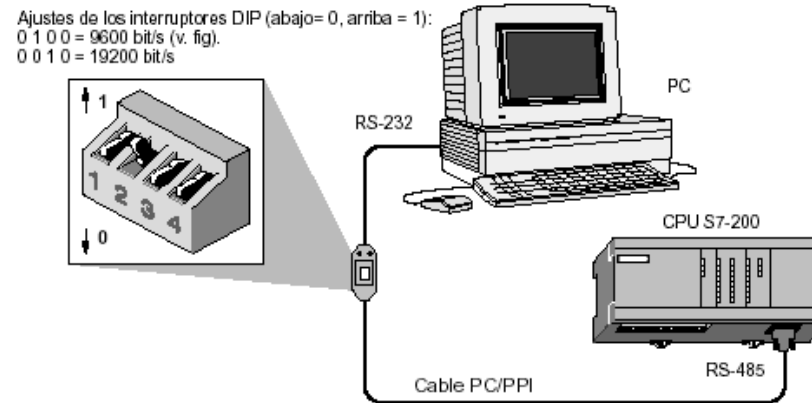
Compilando bloque de datos...
Tamaño del bloque = 0 (bytes), 0 errores

Listo

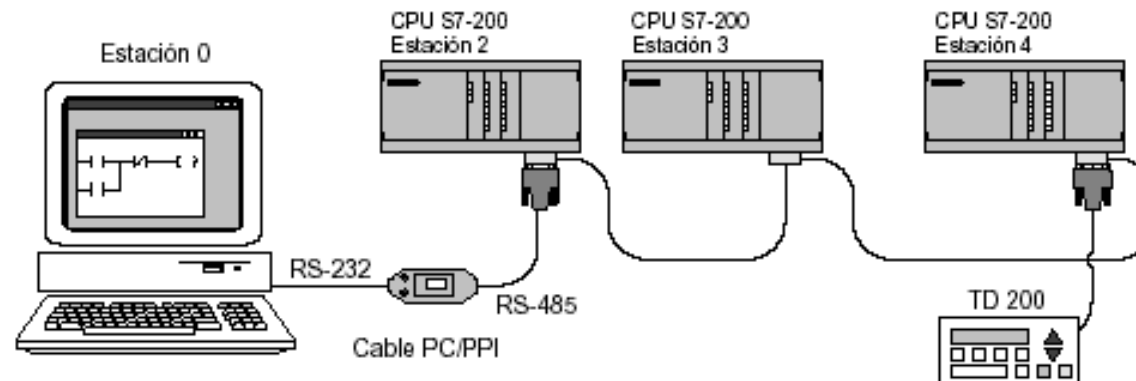
Network 1 Fila 1, Col 1 INS

Conexión de la programadora al PLC (I)

- Establecimiento de un enlace entre los componentes correctamente por PPI
 - Ajuste de los DIP para determinar la velocidad de transferencia deseada (9,6kbit/s)
 - Conexión del cable
 - Extremo RS-232 (canon 9 pines hembra) a COM1 o COM2 del PC
 - Extremo RS-485 (canon 9 pines macho) al interface de la CPU

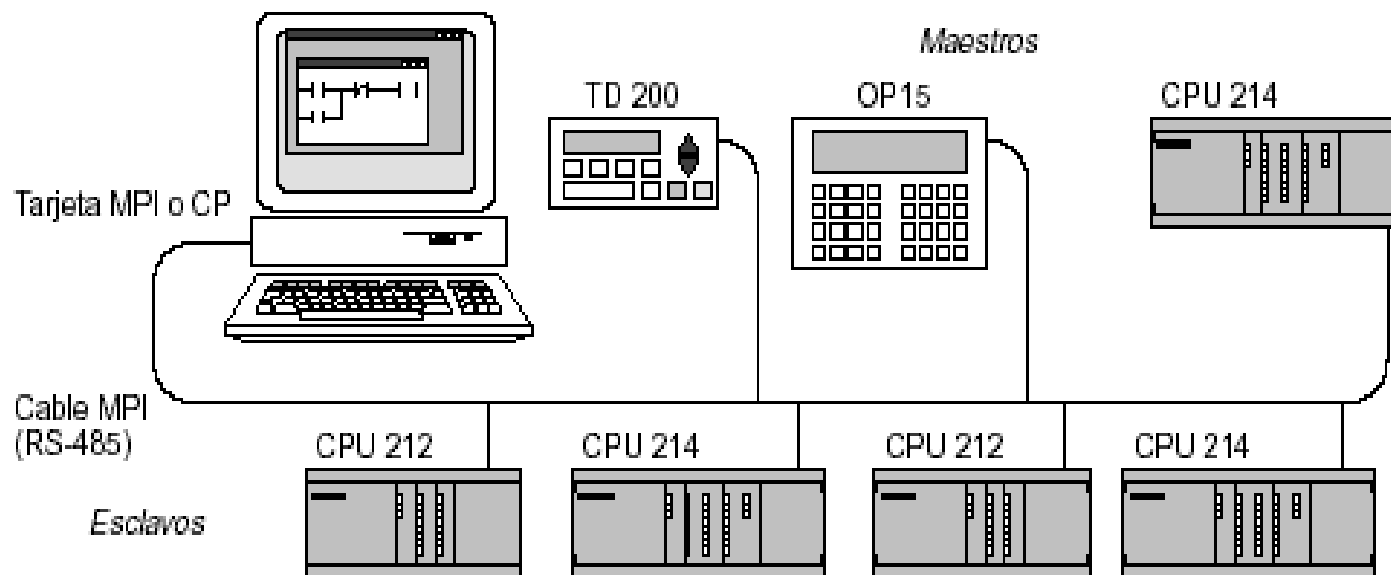


- Establecimiento del enlace con una red de CPUs mediante PPI
 - Los componentes de la red comunicados entre sí .
 - Se puede direccionar cualquiera de ellas, sean esclavas o maestras.



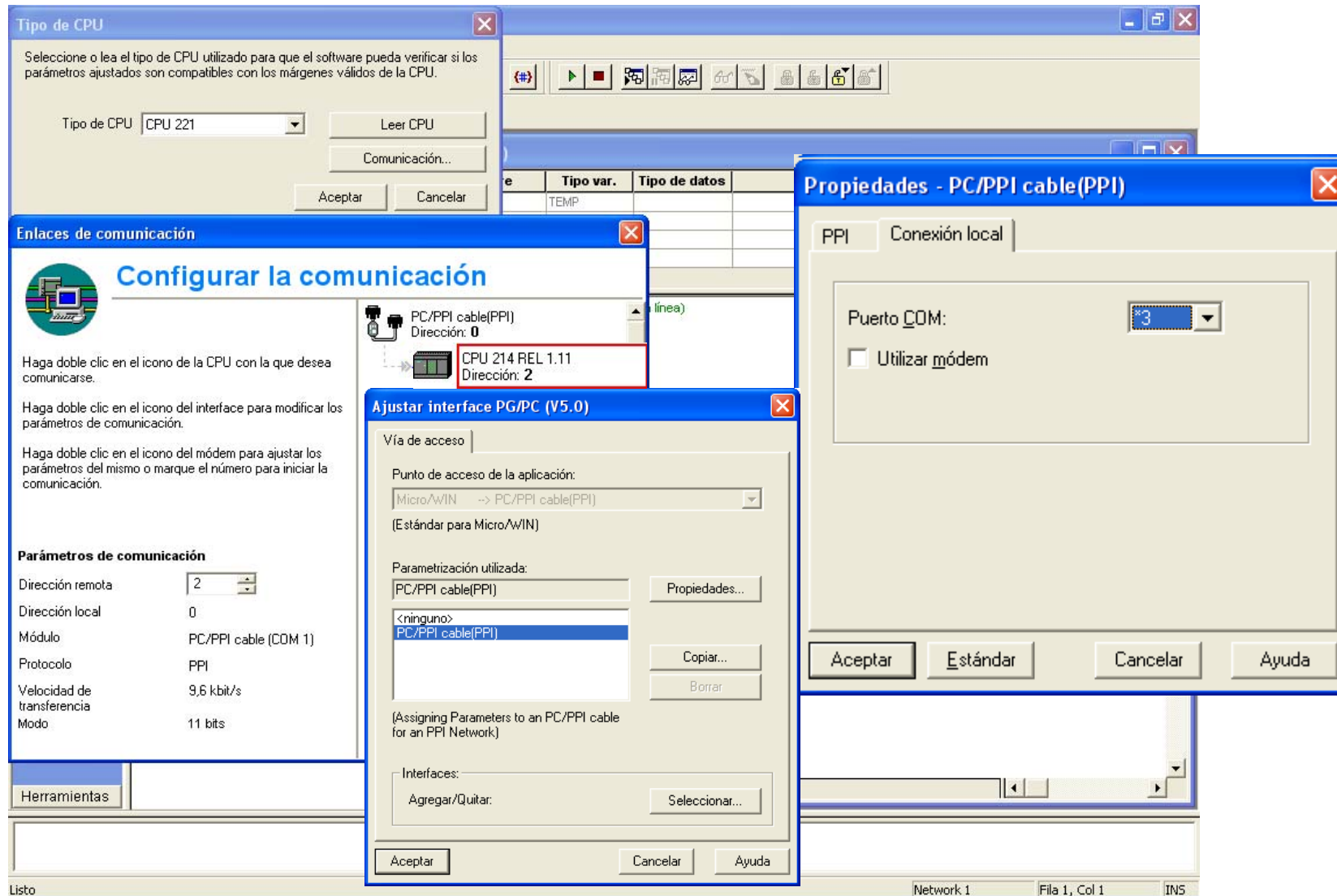
Conexión de la programadora al PLC (II)

- Establecimiento de un enlace mediante MPI o CP
 - Si STEP 7 Micro/WIN está instalado en una programadora con
 - MPI (Interface Multi Punto) y/o
 - CP (Procesador de Comunicaciones)
 - Ambos disponen de un puerto RS-485 sencillo para la conexión a la red de automatización.
 - La red está compuesta por equipos con una dirección única en la red → direccionables.



Configuración de los parámetros de comunicación en Step 7 Micro/WIN (II)

- Acceder a la barra de menú contextual y seleccionar la opción:
 - CPU → Tipo



La familia S7

- Ver presentación de *familia S7*:
 - CPUs
 - Gama S7 - 200
 - Gama S7 - 300
 - Reconocimiento.
 - Diferencias entre CPUs.
 - Capacidad de expansión.
 - Funciones de programación disponibles.
 - Comunicaciones.
 - Alimentaciones.
 - Protecciones.
 - . . .

El ciclo de la CPU S7-200 (I)

- La CPU ejecuta cíclicamente una serie de tareas, incluido el programa de usuario.
- Modos de operación de la CPU
 - RUN.- Ejecuta todas y cada una de las tareas del ciclo en el orden indicado.
 - STOP.- Ejecuta todas las tareas indicadas excepto el programa de usuario.



El ciclo de la CPU S7-200 (I)

- Lectura de las entradas digitales
 - Al principio de cada ciclo se leen los valores presentes en las entradas y se escriben en la “PAE”.
 - A esta tabla es a la que accede el programa de usuario para consultar el estado de las entradas.
- Lectura de las entradas analógicas
 - Su actualización **NO** forma parte de las tareas del ciclo.
 - A las entradas analógicas se accede directamente desde el programa de usuario.
- Ejecución del programa
 - Consiste en la ejecución desde la primera operación hasta la última.
 - El control total del programa permite:
 - Consulta de entradas.
 - Consulta y modificación del estado de las salidas.
 - Acceso y respuesta a una rutina de interrupción.
- Procesado de las peticiones de comunicación
 - La CPU procesa los mensajes que haya recibido por el interface de comunicación.
- Ejecución del diagnóstico de la CPU
 - Tanto en modo RUN como en STOP

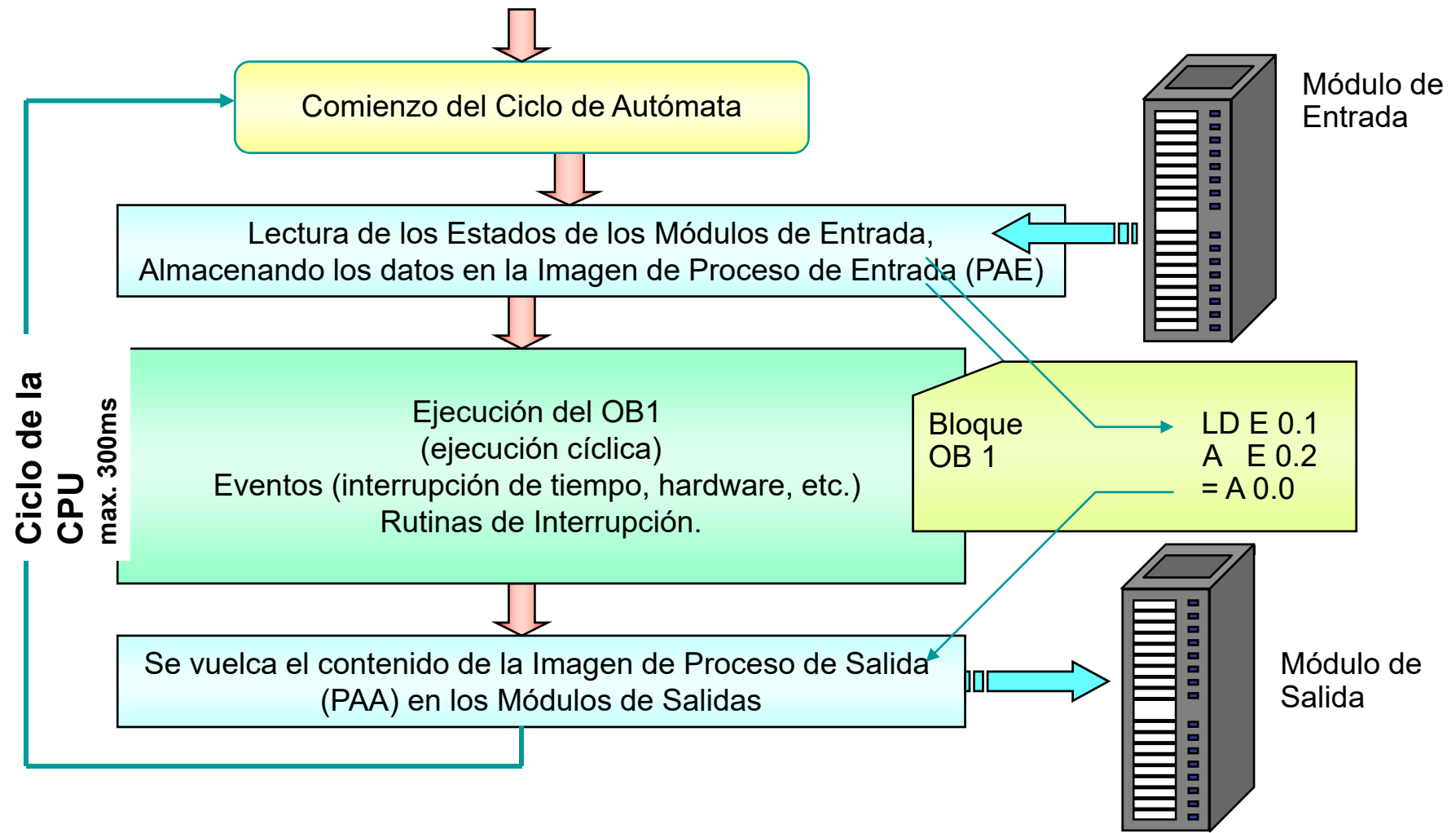
El ciclo de la CPU S7-200 (I)

- Escritura de las salidas digitales
 - Durante la ejecución del programa de usuario, las órdenes de ejecución de las salidas se escriben en la “PAA”.
 - Al final del ciclo, la CPU vuelca el estado de esta tabla físicamente a las salidas.
 - Cuando la CPU pasa de RUN a STOP las salidas digitales pueden:
 - Conservar su estado actual o
 - Adoptar un valor determinado
- Escritura de las salidas analógicas
 - Al igual que con las entradas analógicas, la CPU **NO** actualiza automáticamente las salidas analógicas ni ofrece una imagen del proceso de las mismas.
 - También en este caso, a las salidas analógicas se debe acceder directamente desde el programa de usuario.
- Ejecución de interrupciones
 - Interrupción: Rutina asociada a los eventos de interrupción.
 - Se almacenan como parte del programa de usuario pero **NO** se ejecutan con él.
 - Se ejecutan como respuesta al evento. Justo cuando ocurre el evento.
 - Las interrupciones se ejecutan
 - Según su prioridad.

El ciclo de la CPU S7-200 (II)

- Imagen del proceso de las entradas y salidas
 - En general, es aconsejable utilizar las imágenes del proceso *PAE* y *PAA* que acceder directamente a las entradas o salidas durante la ejecución del programa de usuario.
 - Razones de la existencia de las *PAE* y *PAA*
 - Efecto estabilizador del sistema
 - Todas las entradas se leen en un momento concreto.
 - Todas las salidas se escriben en otro momento concreto.
 - El tiempo de acceso del programa de usuario a las *PAE* y *PAA* es menor que el de acceso directo, con lo cual disminuye el tiempo de ciclo.
 - Flexibilidad adicional
 - Las entradas son unidades de bit → a ellas sólo se puede acceder directamente bit a bit.
 - A las *PAE* y *PAA* se puede acceder en formato de bit, byte, word y dword.
- Control directo de entradas y salidas
 - El acceso directo a una entrada **NO** modifica el contenido de la dirección correspondiente en la *PAE*.
 - El acceso directo a una salida **SI** modifica el contenido de la dirección correspondiente de en la *PAA*.

El ciclo de la CPU S7-200



Estructura de una solución de automatización (I)

- Para automatizar un proceso existen infinitas soluciones.
- Conjunto de reglas generales de común aplicación a numerosos proyectos
 - Estructuración del proceso o la instalación.
 - Especificación de las unidades funcionales.
 - Diseño de los circuitos de seguridad cableados.
 - Definición de las estaciones de operador.
 - Creación de los planos de configuración del PLC.
 - Lista de nombres simbólicos (si se desea utilizar en el direccionamiento)
- Además de lo anterior, también hay que tener en cuenta
 - Hábitos propios de la empresa.
 - Experiencia profesional

Estructura de una solución de automatización (II)

- Estructuración del proceso o la instalación.
 - Definir secciones del proceso independientes entre sí →
 - Determinación de los límites de las distintas partes de la automatización →
 - Para finalmente poder definir y describir con exactitud
 - Las áreas de funciones.
 - Los recursos para cada una de estas áreas.
- Especificación de las funciones de cada sección del proceso o instalación
 - Entradas y salidas necesarias (analógicas y digitales)
 - Descripción exacta del funcionamiento (inicio, desarrollo y fin de cada función)
 - Condiciones de habilitación de cada actuador
 - Estados que se deben alcanzar antes de que se ejecute cada función.
 - Descripción del interface de operador (diálogo hombre-máquina)
 - Interfaces con otras secciones del proceso o de la instalación.

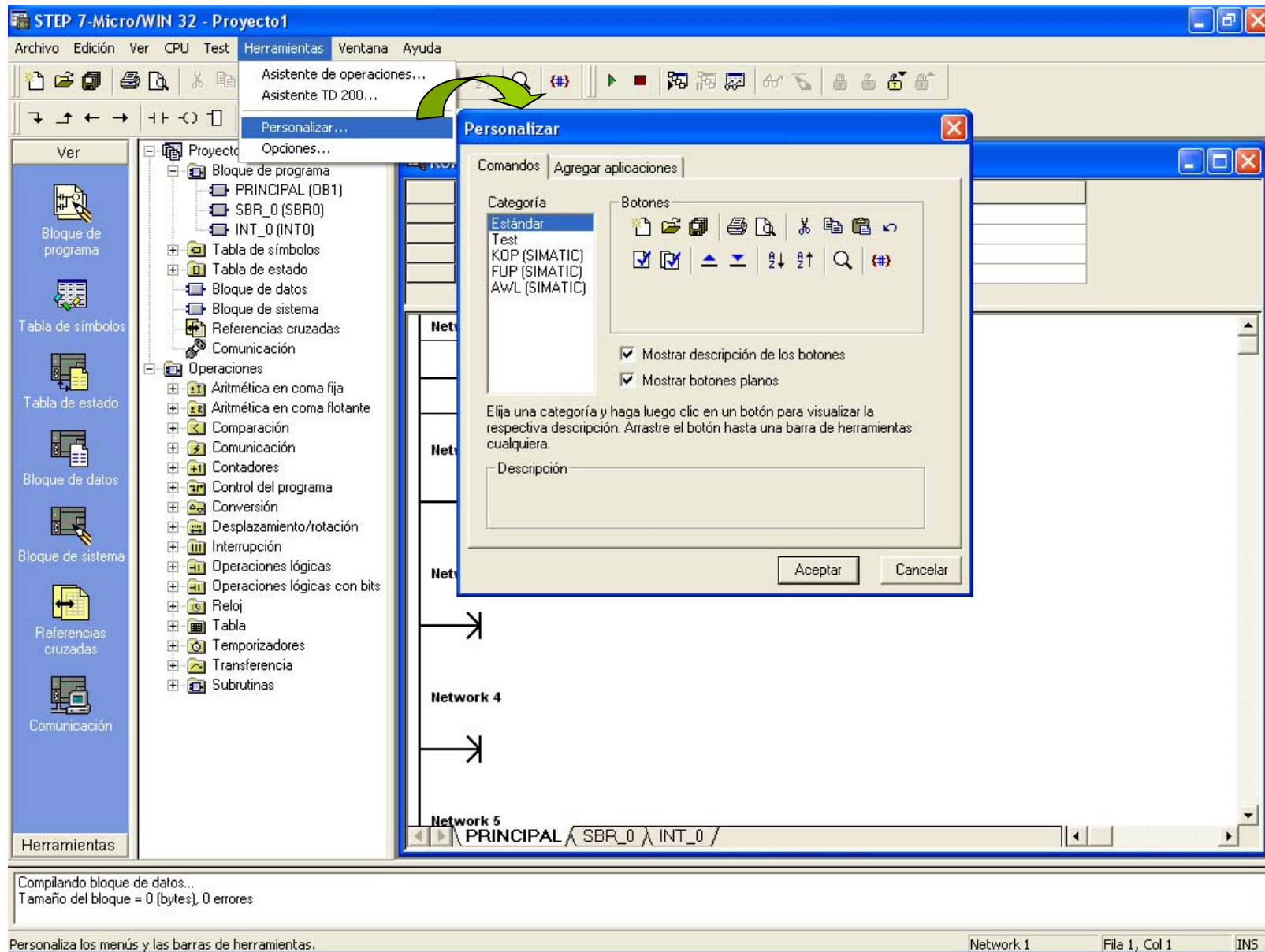
Estructura de una solución de automatización (III)

- Diseño de los circuitos de seguridad cableados
 - Determinación de
 - Posibilidades de que un arranque inesperado o fallo en el sistema de automatización y daños humanos y/o materiales que puedan provocar.
 - Aparatos que requieren de cableado permanente por motivos de seguridad.
 - Esto hace necesario utilizar dispositivos de protección independientes del PLC que eviten riesgos.
 - Criterios de diseño de circuitos de seguridad cableados
 - Definición de los posibles malos funcionamientos que puedan causar peligro.
 - Definición de las condiciones que garanticen la seguridad.
 - Definición de las operaciones a realizar por el PLC
 - Importante: funcionamiento normal + funcionamiento anormal $\Rightarrow$ funcionamiento seguro
 - Prever dispositivos independientes de la CPU
 - Transmitir información clara y concisa a los interfaces de operador.
 - Definir todos los criterios de seguridad a mayores para que el proceso sea seguro y fiable.

Estructura de una solución de automatización (IV)

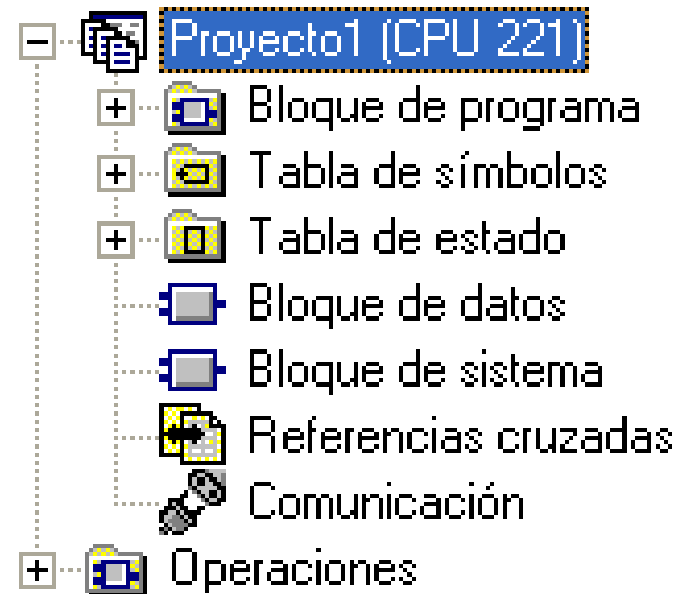
- Definición de las estaciones de operador
 - Panorámica de la ubicación de todas las estaciones de operador con respecto al proceso.
 - Disposición mecánica de los aparatos
 - Pantallas, paneles operativos, interruptores, lámparas, ...
 - Esquemas eléctricos completos para su consulta.
- Planos de configuración del sistema de automatización incorporando los siguientes puntos
 - Ubicación de
 - Las CPUs y los módulos de ampliación.
 - Disposición mecánica de las CPUs y los módulos de ampliación.
 - Esquemas eléctricos de las CPUs y los módulos de ampliación.

Gestión de las barras de herrameintas



Estructura del proyecto

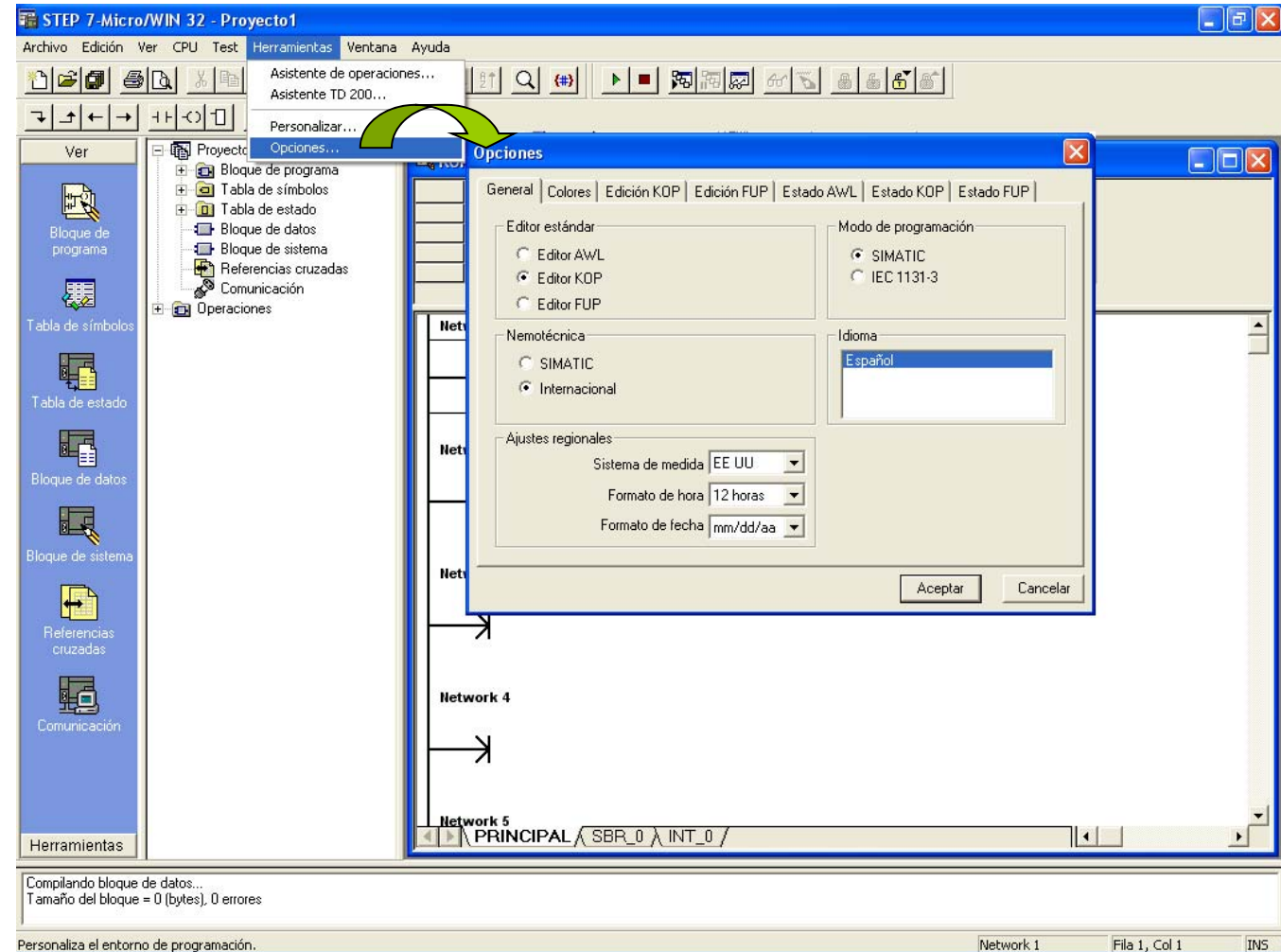
- Cinco componentes principales:



Estructura del proyecto

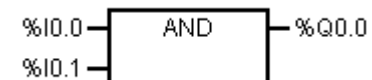
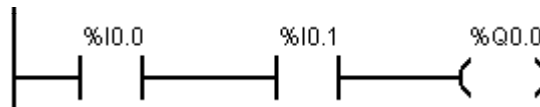
- El Bloque de programa contiene
 - El código ejecutable
 - Programa principal (OB1)
 - Subrutinas o rutinas de interrupción.
 - Comentarios
- El Bloque de datos contiene
 - Datos
 - Valores iniciales,
 - Resultados de operaciones,
 - Constantes,
 - Comentarios
- El Bloque de sistema contiene la información de la configuración del PLC
 - Parámetros de comunicaciones,
 - Áreas de datos remanentes,
 - Filtros de entradas analógicas y digitales,
 - Nivel de acceso al PLC,
- La tabla de símbolos
 - Permite utilizar direccionamiento simbólico para que la programación sea más simple
 - MicroWin traduce todos los símbolos en direcciones absolutas antes de enviar el programa al PLC.
- La tabla de estados
 - Contiene las direcciones:
 - De las que se desea conocer su estado.
 - De las que se desea modificar su valor (forzar)

Opciones

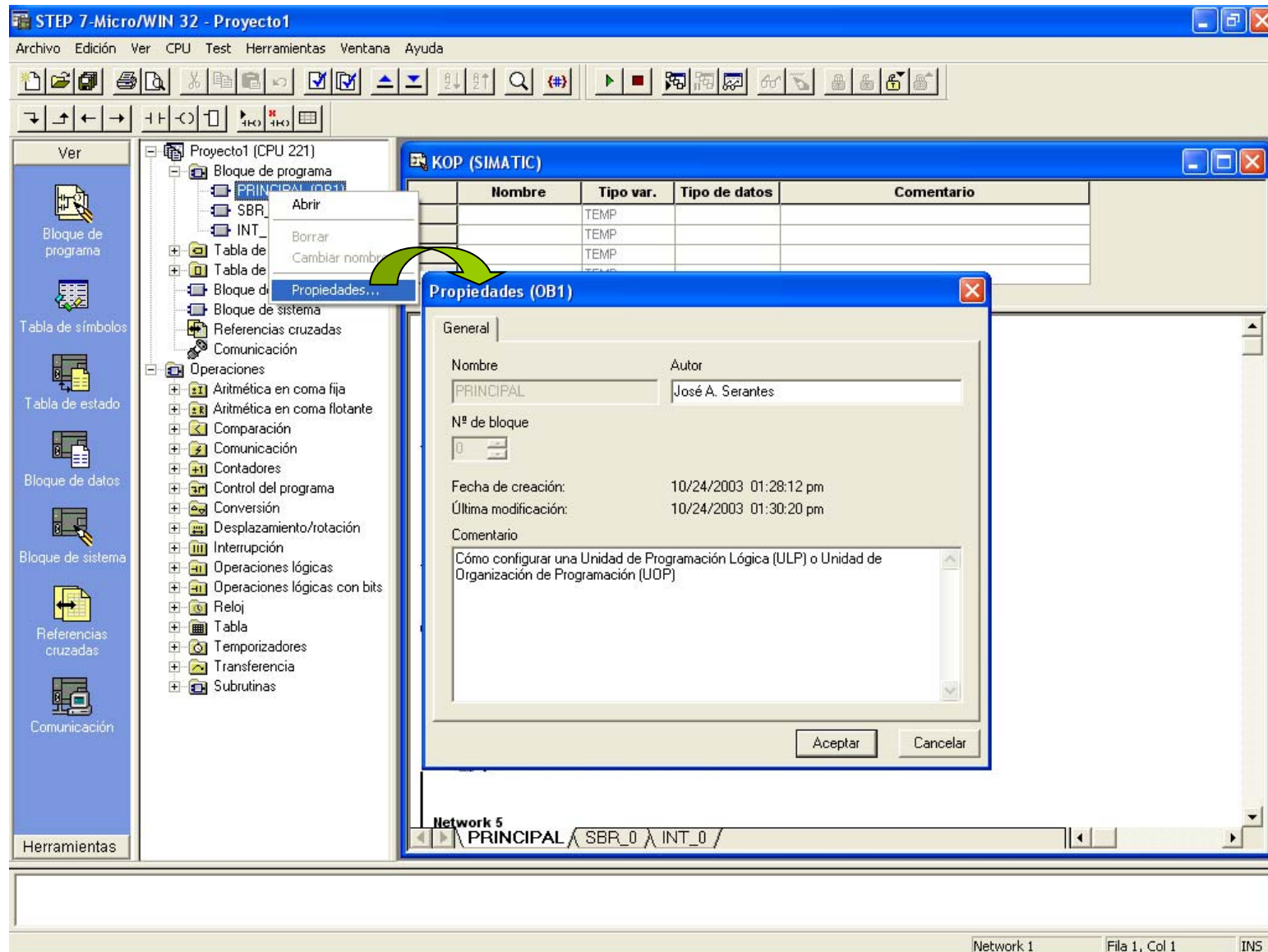


LD
U
=

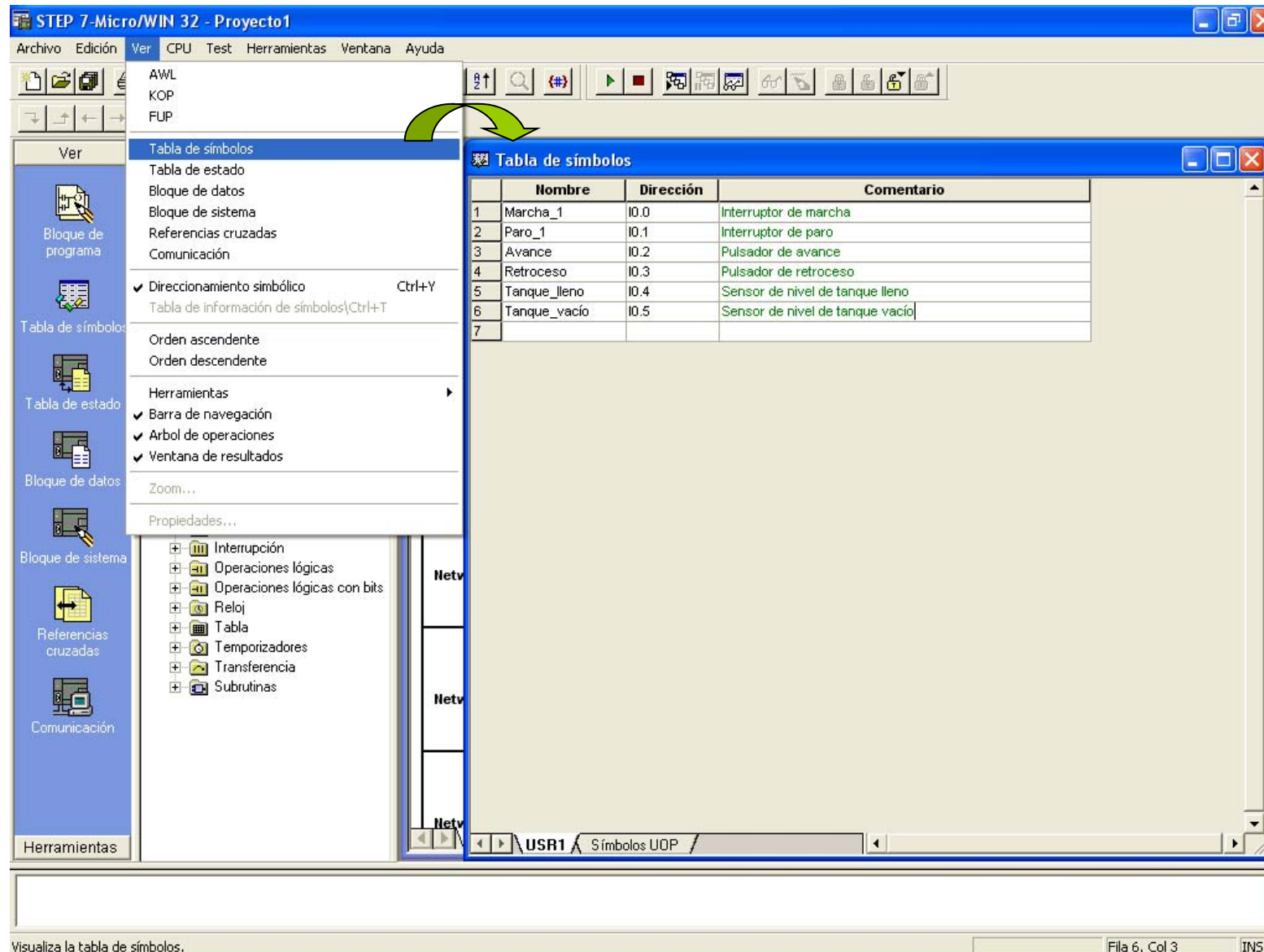
EO.0
EO.1
AO.0



Creación de un proyecto

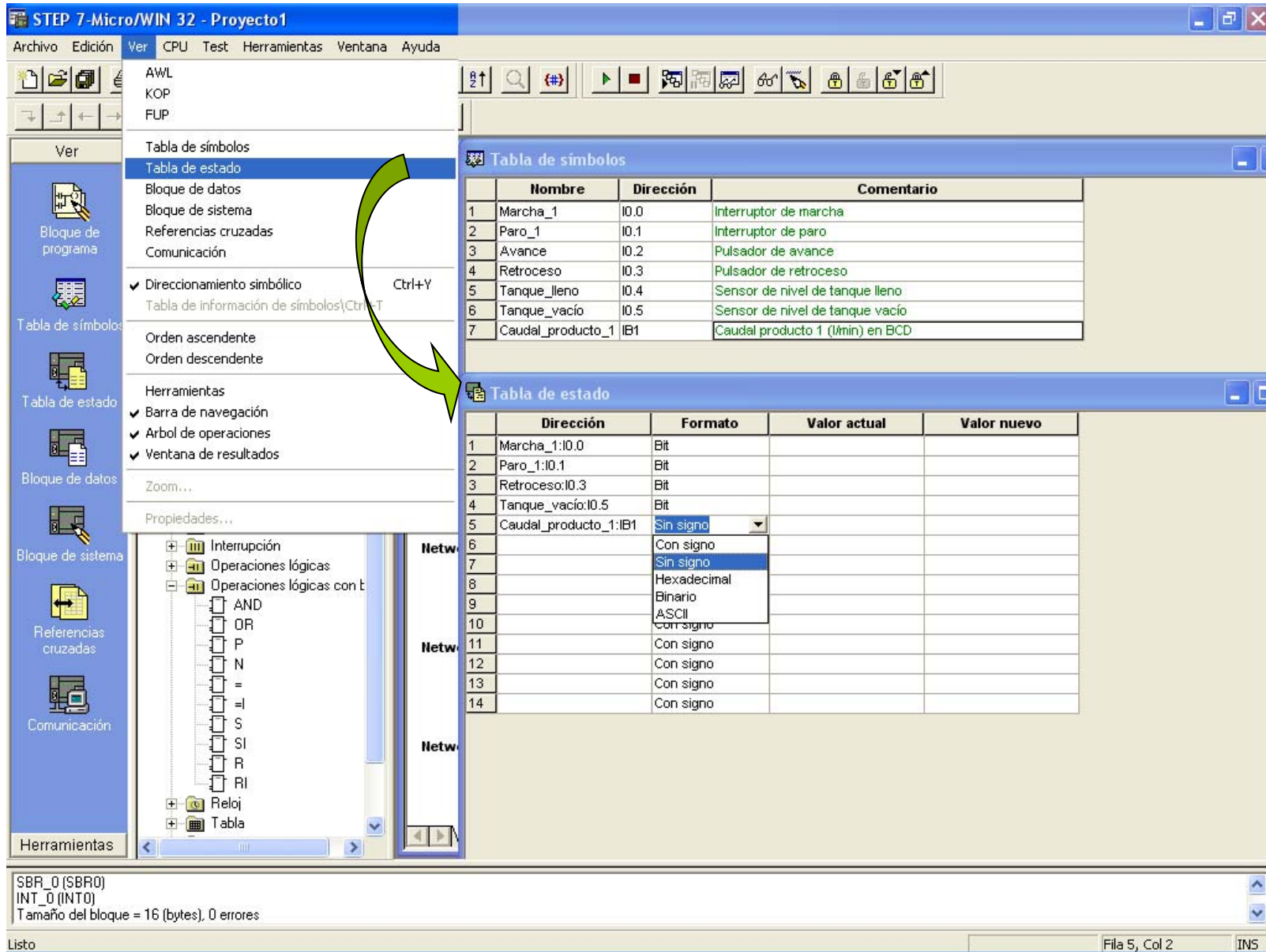


La tabla de símbolos



- En cualquier momento del desarrollo del proyecto es posible:
 - Consultarla.
 - Ampliarla (añadir nuevos símbolos)
 - Reducirla (eliminar símbolos existentes)

Tabla de estado



STEP 7-Micro/WIN 32 - Proyecto1

Archivo Edición Ver CPU Test Herramientas Ventana Ayuda

Ver

- Tabla de símbolos
- Tabla de estado**
- Bloque de datos
- Bloque de sistema
- Referencias cruzadas
- Comunicación
- ✓ Direccionamiento simbólico
- Tabla de información de símbolos (Ctrl+T)
- Orden ascendente
- Orden descendente
- Herramientas
- ✓ Barra de navegación
- ✓ Árbol de operaciones
- ✓ Ventana de resultados
- Zoom...
- Propiedades...

Ctrl+Y

Interrupción

Operaciones lógicas

Operaciones lógicas con t

AND

OR

P

N

=

=I

S

SI

R

RI

Reloj

Tabla

Netw

Netw

Netw

SBR_0 (SBR0)

INT_0 (INT0)

Tamaño del bloque = 16 (bytes), 0 errores

Lista

Fila 5, Col 2

INS

Tabla de símbolos

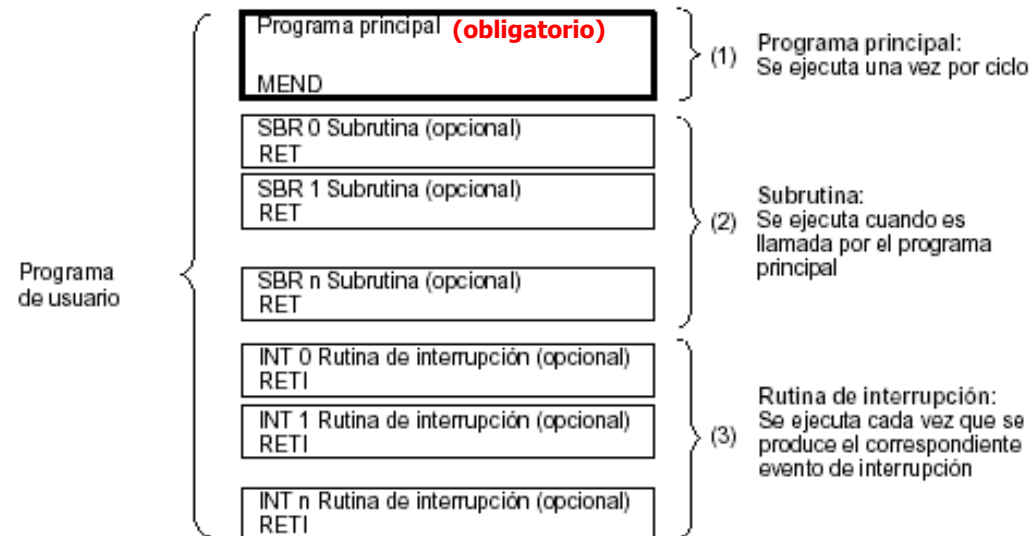
	Nombre	Dirección	Comentario
1	Marcha_1	IO.0	Interrupción de marcha
2	Paro_1	IO.1	Interrupción de paro
3	Avance	IO.2	Pulsador de avance
4	Retroceso	IO.3	Pulsador de retroceso
5	Tanque_lleno	IO.4	Sensor de nivel de tanque lleno
6	Tanque_vacio	IO.5	Sensor de nivel de tanque vacío
7	Caudal_producto_1	IB1	Caudal producto 1 (l/min) en BCD

Tabla de estado

	Dirección	Formato	Valor actual	Valor nuevo
1	Marcha_1:IO.0	Bit		
2	Paro_1:IO.1	Bit		
3	Retroceso:IO.3	Bit		
4	Tanque_vacio:IO.5	Bit		
5	Caudal_producto_1:IB1	Sin signo		
6		Con signo		
7		Sin signo		
8		Hexadecimal		
9		Binario		
10		ASCII		
11		Con signo		
12		Con signo		
13		Con signo		
14		Con signo		

- Permite observar la evolución de los valores asociados a las direcciones seleccionadas durante la ejecución del programa del PLC

Elementos básicos en la estructuración de programas de PLC



- Proceso de puesta en funcionamiento de un programa de PLC
 - El programa de PLC se crea con Micro/WIN y se transfiere al PLC.
 - El PLC ejecuta el programa cíclicamente.
- Un programa de PLC contiene tres partes básicas:
 - Programa principal
 - Operaciones que controlan la aplicación.
 - Ejecutadas de forma secuencial en cada ciclo de la CPU.
 - Si se desea realizar un fin condicional, ordenar ejecución de instrucción “END”
 - Subrutinas
 - Elementos opcionales del programa.
 - Sólo se ejecutan si se llaman desde el programa principal.
 - Si se desea realizar un fin condicional, ordenar ejecución de instrucción “RET”
 - Rutinas de interrupción
 - Elementos opcionales del programa.
 - Se ejecutan como respuesta inmediata al evento de interrupción al que estén asociadas.
 - Si se desea realizar un fin condicional, ordenar ejecución de instrucción “RET”

Errores y violación de las reglas de compilación en la CPU

- Errores fatales y mensajes asociados
 - Son errores graves que provocan
 - Que la CPU no pueda ejecutar todas las funciones.
 - O incluso ninguna de ellas.
 - Su tratamiento es conducir a la CPU a un estado seguro en el que
 - Se puedan analizar las condiciones que originaron el error.
 - Y se pueda operar para eliminarlo.
 - Cómo saber que se trata de un error fatal
 - La CPU cambia a modo STOP.
 - Además se enciende el led SF (System Failure)
 - Se desactivan las salidas
 - **Nota:** Se cargan los valores predeterminados ajustados con anterioridad en la tabla de salidas.
- Errores de programación en tiempo de ejecución
 - Durante la ejecución del programa de usuario pueden aparecer errores **NO** fatales (p. ej. errores de direccionamiento)
 - Entonces, la CPU genera un código de error **NO** fatal en tiempo de ejecución
 - Ejemplo: Hacer un bucle cerrado en el que se rebase el tiempo de ciclo
- Violación de reglas de compilación
 - Antes de cargar un programa en la CPU, ésta lo compila.
 - Si durante la compilación se detecta una violación de las reglas (p. ej. una operación no válida), la CPU detiene el proceso de carga del programa.
 - Además, genera un código de error **NO** fatal que se puede observar en la ventana de resultados.

Códigos de error (I)

- Códigos de errores fatales y mensajes

Código de error	Descripción
0000	No hay errores fatales
0001	Error de suma de verificación en el programa de usuario
0002	Error de suma de verificación en el programa KOP compilado
0003	Error de tiempo en la vigilancia del tiempo de ciclo (watchdog)
0004	Error EEPROM interno
0005	Error EEPROM interno de suma de verificación en el programa de usuario
0006	Error EEPROM interno de suma de verificación en los parámetros de configuración
0007	Error EEPROM interno de suma de verificación en los datos forzados
0008	Error EEPROM interno de suma de verificación en los valores predeterminados de la imagen de proceso de las salidas
0009	Error EEPROM interno de suma de verificación en los datos de usuario, DB1
000A	Error en el cartucho de memoria
000 B	Error de suma de verificación del cartucho de memoria en el programa de usuario
000C	Error de suma de verificación del cartucho de memoria en los parámetros de configuración
000D	Error de suma de verificación del cartucho de memoria en los datos forzados
000E	Error de suma de verificación del cartucho de memoria en los valores predeterminados de la imagen de proceso de las salidas
000 F	Error de suma de verificación del cartucho de memoria en los datos de usuario, DB1
0010	Error interno de software
0011	Error en el direccionamiento indirecto del contacto de comparación
0012	Error: valor no válido del contacto de comparación
0013	Cartucho de memoria vacío o programa no apto para esta CPU

Eliminación de errores fatales

- Error fatal → la CPU detiene la ejecución del programa
 - Error fatal muy grave → La CPU no puede ejecutar ninguna función.
 - Error fatal leve → La CPU aún puede ejecutar algunas funciones.
 - En ambos casos,
 - la CPU indica el error fatal mediante:
 - Led de STOP encendido.
 - Led de SF (System Failure) encendido
 - Se desactivan las salidas
 - **Nota:** Se cargan los valores predeterminados ajustados con anterioridad en la tabla de salidas.
- Tratamiento de los errores fatales
 - Conducir a la CPU a un “estado seguro”
 - Para a continuación
 - Detectar las causas que han provocado el error accediendo a la opción de menú:
 - “CPU → Información”.
 - Eliminar las causas que han provocado el error.
 - Una vez eliminado el error, para pasar al funcionamiento normal
 - Pasar el selector de modos a STOP y luego a RUN o TERM
 - Al hacer esto,
 - la CPU resetea la condición de error fatal y
 - ejecuta un diagnóstico de arranque para verificar que el error se ha corregido
- **Caso particular:** Incapacidad de establecer comunicación con la CPU
 - En este caso es imposible visualizar el código de error.
 - En este caso se trata de un error de *hardware* que debe reparar el servicio técnico

Códigos de error (II)

- Errores de programación en tiempo de ejecución

Código de error	Error de programación del tiempo de ejecución (no fatal)
0000	No se presentó ningún error
0001	Cuadro HSC habilitado antes de ejecutar el cuadro HDEF
0002	Interrupción de entrada asignada a una entrada que ya está asociada a un HSC (conflicto)
0003	Entradas asignadas a un HSC que ya está asociado a una interrupción de entrada (conflicto)
0004	Se ha intentado ejecutar una operación ENI, DISI o HDEF en una rutina de interrupción
0005	Antes de finalizar el primer HSC se ha intentado ejecutar un segundo HSC con el mismo número (HSC de la rutina de interrupción en conflicto con HSC del programa principal)
0006	Error de direccionamiento indirecto
0007	Error en datos para operación TODW (Escribir en reloj de tiempo real)
0008	Excedida la profundidad máxima de anidado para subrutina de usuario
0009	Ejecución de una operación XMT o RCV mientras se está ejecutando otra operación XMT o RCV
O00A	Se ha intentado redefinir un HSC ejecutando otra operación HDEF para el mismo HSC
0091	Error de margen (con información sobre direcciones): verificar las áreas de operandos
0092	Error en el campo de conteo de una operación (con información sobre el conteo): verificar el valor máximo de conteo
0094	Error de margen al escribir en la memoria no volátil (con información sobre direcciones)

Códigos de error (III)

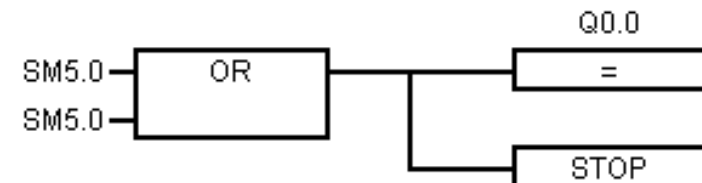
- Violación de reglas de compilación

Código de error	Error de compilación (no fatal)
0080	Programa demasiado extenso para la compilación: reducir el tamaño del programa
0081	Rebase negativo de la pila: dividir el segmento en varios segmentos
0082	Operación no válida: comprobar la nemotécnica de la operación
0083	Falta MEND u operación no admisible en el programa principal: agregar la operación MEND o borrar la operación incorrecta
0084	Reservado
0085	Falta FOR: agregar la operación FOR o borrar la operación NEXT
0086	Falta NEXT: agregar la operación NEXT o borrar la operación FOR
0087	Falta meta (LBL, INT, SBR): agregar la meta apropiada
0088	Falta RET u operación no admisible en una subrutina: agregar RET al final de la subrutina o borrar la operación incorrecta
0089	Falta RETÍ u operación no admisible en una rutina de interrupción: agregar RETÍ al final de la rutina de interrupción o borrar la operación incorrecta
008A	Reservado
008B	Reservado
008C	Meta doble (LBL, INT, SBR): cambiar el nombre de una de las metas
008D	Meta no válida (LBL, INT, SBR): asegurarse de que el número admisible de metas no se haya excedido
0090	Parámetro no válido: comprobar los parámetros admisibles para la operación
0091	Error de margen (con información sobre direcciones): verificar las áreas de operandos
0092	Error en el campo de contaje de una operación (con información sobre el contaje): verificar el valor máximo de contaje
0093	Excedida la profundidad de anidado FOR/NEXT
0095	Falta la operación LSCR (cargar SCR)
0096	Falta la operación SCRE (fin de SCR) u operación no admisible antes de la operación SCRE

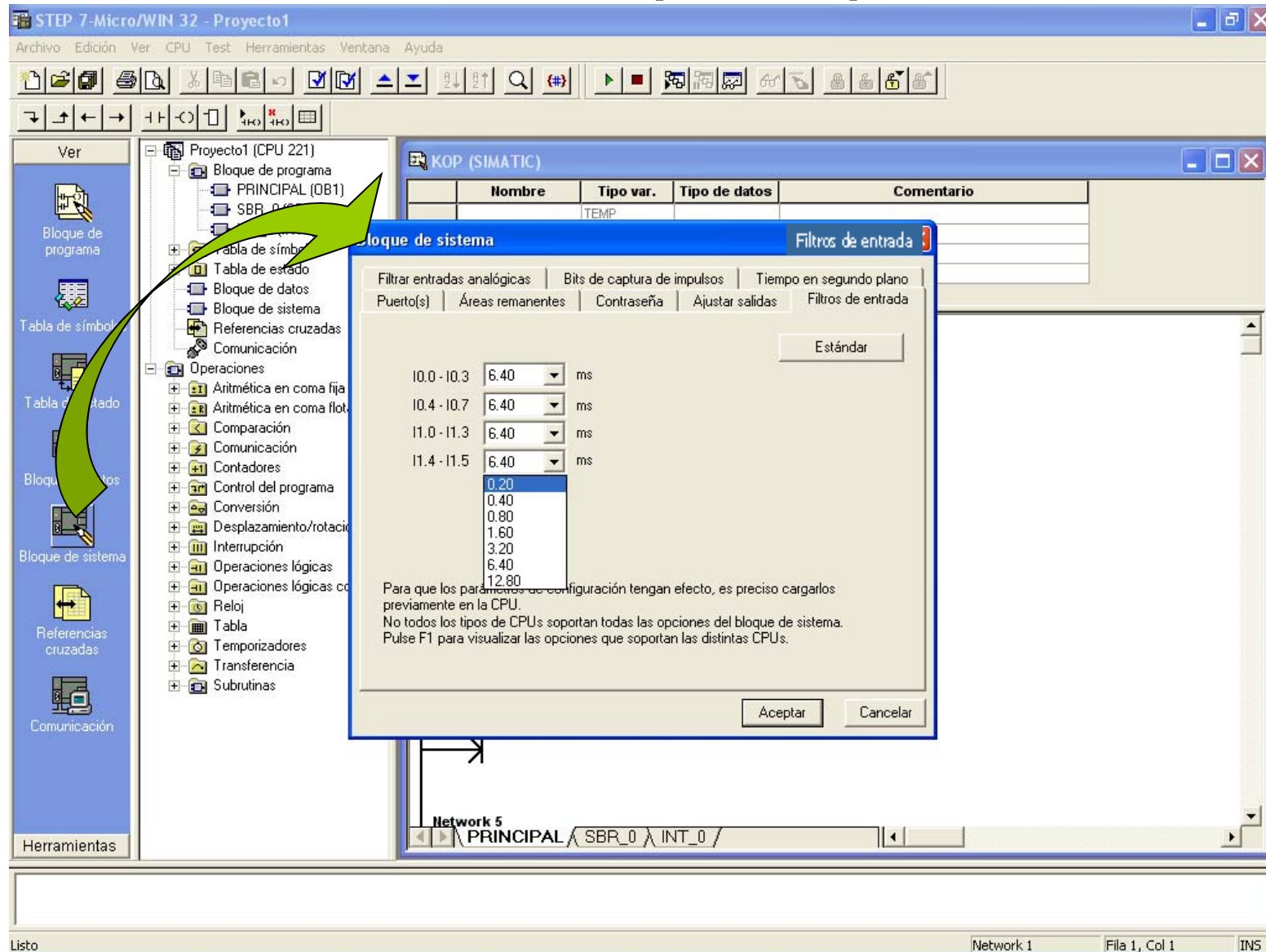
Eliminación de errores NO fatales

- Pueden
 - mermer parcialmente el funcionamiento de la CPU,
- Pero no impiden
 - Ejecutar el programa.
 - Leer entradas.
 - Actualizar salidas.
- Categorías de errores
 - Errores durante el tiempo de ejecución
 - Ejemplo:
 - La CPU al pasar a modo RUN (entre otras cosas) lee la configuración de las E y S y almacena esta información.
 - Esta información se relee y compara cada ciclo con la del momento del arranque.
 - Si durante el funcionamiento normal detecta un cambio, se activa el bit de error correspondiente.
 - Son errores **NO** fatales que
 - Se detectan en modo RUN y
 - Se almacenan en *Marcas especiales (SM)*
 - El propio programa puede (y debe)
 - Consultar esas *marcas especiales* y
 - Tomar decisiones al respecto
 - Errores de compilación del programa
 - La CPU compila siempre el programa antes de que éste sea cargado.
 - Si durante la compilación se detecta una violación de las reglas
 - Se genera un código de error asociado (que se puede observar en la ventana de resultados, y
 - Se suspende el proceso de carga del programa en la CPU.
 - Errores de programación durante el tiempo de ejecución
 - Ejemplo:
 - Desbordar la capacidad de conteo de un contador.
 - También son errores **NO** fatales que
 - Se detectan en modo RUN y
 - Se almacenan en *Marcas especiales (SM)*
 - También el propio programa puede (y debe)
 - Consultar esas *marcas especiales* y
 - Tomar decisiones al respecto

Network 1 TÍTULO DE SEGMENTO (una línea)

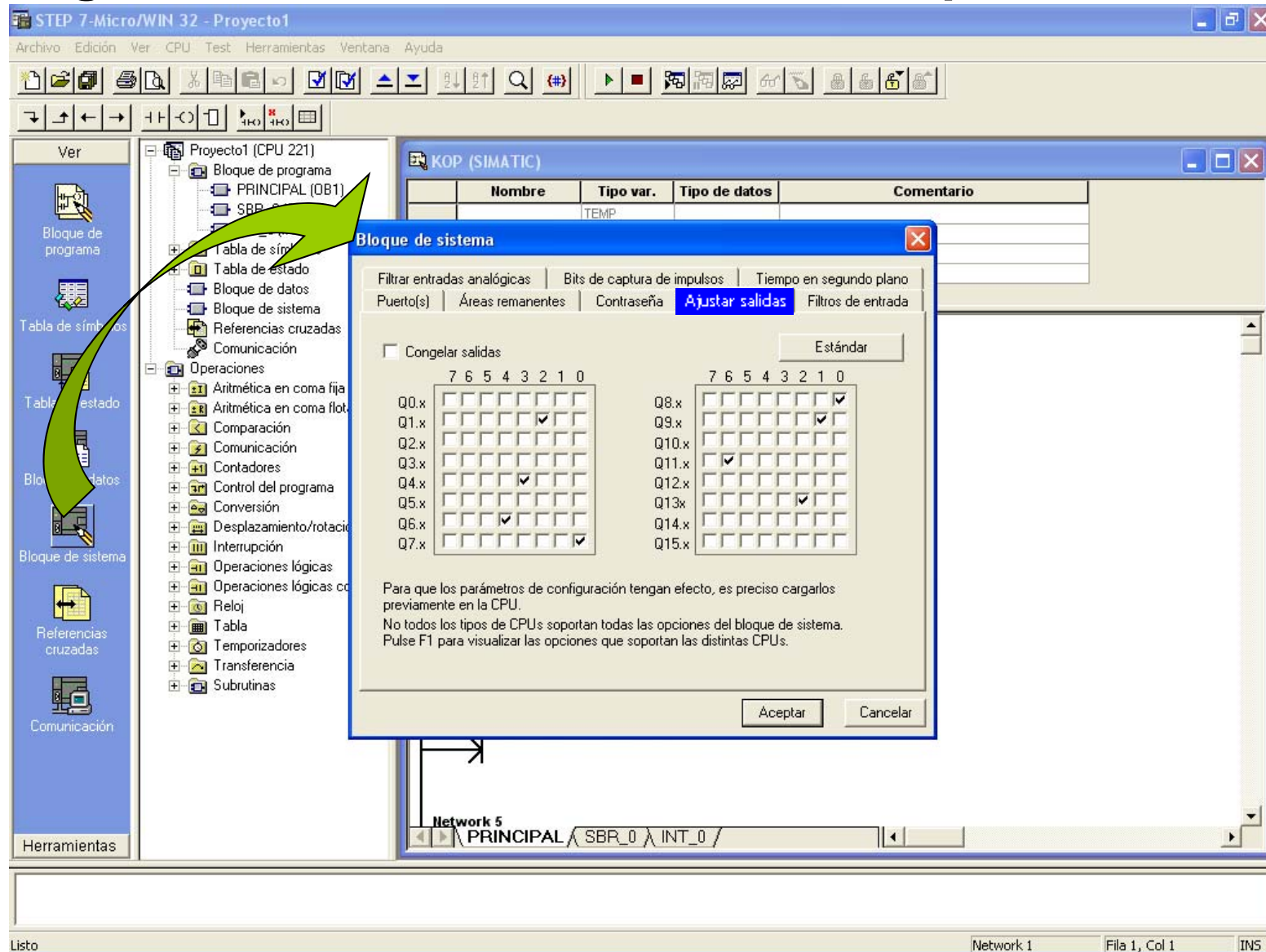


Uso de los filtros de entrada para suprimir interferencias



- Filtros de entrada
 - Utilidad: Filtrado de ED para eliminar las interferencias procedentes del cableado conectado a ellas:
 - La CPU no dispone de una *tabla de imagen de EA* → Las EA no se filtran automáticamente

Configuración de los estados de señal para las salidas



- Ajustar salidas
 - Utilidades:
 - Hacer que las SD adopten valores conocidos cuando la CPU pasa al estado STOP o
 - Hacer que el estado de las SD al pasar la CPU a STOP sea justo el que tenían en el último ciclo antes de pasar a STOP (*Congelar salidas*)

Entradas rápidas

- Son ED especiales capaces de contar más deprisa de lo que son capaces las ED convencionales, cuyas velocidades de contaje dependen del tiempo de ciclo de la CPU.
- Se utilizan para lectura de trenes de impulsos de frecuencias altas
 - Encoders
- Según el modelo de CPU, se dispone de:
 - Un contador software: HSC0
 - Contaje ascendente y descendente mediante un bit que marca el sentido de contaje.
 - Asistido por una entrada de reloj única.
 - Frecuencia de contaje máxima 2kHz.
 - Hasta dos contadores hardware: HSC1 y HSC2
 - Configurables en 12 modos de operación distintos.
 - La frecuencia máxima de contaje cambia de una CPU a otra.
 - Son independientes de la CPU → no la cargan de trabajo.
 - Trabajan independientemente sin interferirse entre sí.
 - Los contadores A/B permiten elegir velocidad simple o cuádruple para el contaje.

Salidas de impulsos rápidos

- Son SD capaces de
 - Generar Trenes de Impulsos de Salida rápidos (PTO)
 - Salida en cuadratura para
 - Una cantidad de pulsos determinada (1 .. 429.4967.295).
 - Durante un tiempo de ciclo determinado.
 - » 250 μ s .. 65535 μ s o
 - » 2ms .. 65535ms
 - » Se aconseja no ajustar a un número impar de tiempo. Causa distorsión en el tren.
 - Controla la modulación del ancho de impulso (PWM)
 - Permite definir un tiempo de ciclo fijo con una salida de ancho de impulso variable.
 - Límites de la modulación

Tiempo de ciclo	ancho de impulsos
250 μ s .. 6535 μ s	0 μ s .. 6535 μ s
2ms .. 65535ms	0ms .. 65535ms