Comunicaciones Industriales

GUÍA DIDÁCTICA DEL PROFESOR

Índice

1. Presentación de la guía 3
2. Introducción al módulo 4
3. Capacidades terminales y criterios de evaluación 10
4. Orientaciones metodológicas 18
5. Índice secuencial de las unidades de trabajo: organización de los contenidos 19
6. Estructura de las unidades de trabajo del libro del alumno 20
7. Distribución temporal de las unidades de trabajo 21
8. Elementos curriculares o unidades de trabajo 22
   - Capítulo 1. Principios de comunicación en los entornos industriales..... 22
   - Capítulo 2. Principios de teleinformática: tipos de comunicación y transmisión de datos 23
   - Capítulo 3. Nivel de enlace de datos: transferencia de datos 25
   - Capítulo 4. Redes de área local LAN 27
   - Capítulo 5. Acceso a los puertos de E/S en un PC 28
   - Capítulo 6. El puerto paralelo 30
   - Capítulo 7. El puerto serie RS-232 31
   - Capítulo 8. Buses de campo 32
   - Capítulo 9. Procedimientos de comunicación con dispositivos industriales 34
   - Capítulo 10. Desde C hasta C++ 35
   - Capítulo 11. Clase y Objeto en C++ 36
   - Capítulo 12. C++ en acción 37
   - Capítulo 13. Programación a bajo nivel desde DOS 38
   - Capítulo 14. Programación del puerto paralelo 39
   - Capítulo 15. Programación del puerto serie 39
   - Capítulo 16. C++ Builder 40
   - Capítulo 17. Entrada/salida en C++ Builder 41
   - Capítulo 18. La API del puerto serie 41
   - Anexo La red telefónica 42
9. Actividades, cuestiones, problemas y prácticas propuestas 43
10. Material didáctico (material y equipos didácticos) 59
11. Material pedagógico de apoyo para la impartición del módulo 62

1. Presentación de la guía

La guía didáctica del profesor del módulo Comunicaciones Industriales se ha elaborado con el objetivo de prestar al profesor que imparte la asignatura una propuesta didáctica de apoyo pedagógico para el desarrollo de su función docente.

En la guía se incluyen y se describen los materiales curriculares que presentó el Ministerio de Educación y Ciencia cuando se diseñaron los ciclos formativos y en los que se exponen la definición y el desarrollo de los procesos de enseñanza-aprendizaje de los Ciclos Formativos, tanto de grado superior como de grado medio de la Formación Profesional actual.

Se recogen en esta guía el Real Decreto 619/1995, publicado en el BOE el 08.08.1995, donde se desarrolla el título del módulo, y el Real Decreto 191/1996, publicado en el BOE 06.03.96, donde se desarrolla el currículo del módulo.

La guía sigue las directrices trazadas por el libro publicado por el Ministerio de Educación y Ciencia sobre propuestas didácticas de apoyo al profesor, editado por la Dirección General de Formación Profesional Reglada y Promoción Educativa, en el que se orienta al profesor sobre la programación de los contenidos y las actividades de formación que pueden ser adaptadas y aplicadas por los docentes de forma directa.

La guía está dividida en 10 apartados, que son:

- Introducción al módulo.
- Capacidades terminales y criterios de evaluación.
- Orientaciones metodológicas.
- Índice secuencial de las unidades de trabajo: organización de los contenidos.
- Estructura de las unidades de trabajo del libro del alumno.
- Distribución temporal de las unidades de trabajo.
- Elementos curriculares o unidades de trabajo.
- Actividades, cuestiones, problemas y prácticas propuestas.
- Material didáctico (material y equipos didácticos).
- Material pedagógico de apoyo para la impartición del módulo.

A continuación se desarrollan cada uno de estos 10 puntos.

2. Introducción al módulo

El desarrollo didáctico y la programación del módulo Comunicaciones Industriales se obtienen a partir del perfil del ciclo formativo Sistemas de Regulación y Control Automático.

El ciclo formativo Sistemas de Regulación y Control Automático está dividido en 12 módulos profesionales, como unidades coherentes de formación, necesarios para obtener la titulación de técnico en Sistemas de Regulación y Control Automático. La duración establecida para este ciclo es de 2.000 horas incluida la formación en centros de trabajo. Estas 2.000 horas se dividen en 2 periodos anuales lectivos, cinco trimestres en el centro educativo y el sexto trimestre en el centro de trabajo.

Uno de los módulos incluidos en este ciclo formativo es el de *Comunicaciones Industriales,* que tiene una duración aproximada de 130 horas, a impartir en el 2º curso, con una frecuencia de 6 horas por semana.

La competencia general de este módulo está recogida en la unidad de competencia nº 3 del Real Decreto del título, y que dice:

Desarrollar y mantener sistemas informáticos y de comunicación industrial.

Es importante que las realizaciones que se planteen como básicas tengan como punto de referencia el sistema productivo y en concreto la ocupación o el puesto de trabajo que pueden desempeñar los técnicos que realizan este módulo.

REALIZACIONES PROFESIONALES REFERENCIADAS A LA OCUPACIÓN	CRITERIOS BÁSICOS DE REALIZACIÓN
Configurar, a su nivel, el sistema informático y de comunicación industrial de acuerdo con el correspondiente cuaderno de cargas, determinando la configuración topológica estándar, seleccionando los equipos, dispositivos y el "software" de base y de comunicación, en condiciones óptimas de calidad, fiabilidad y coste.	– Las especificaciones del sistema informático y de comunicaciones recogen con precisión los requerimientos y prestaciones funcionales, técnicas y de coste requeridas. – La arquitectura del sistema informático se configura de acuerdo con los requerimientos de las aplicaciones que se van a utilizar, teniendo en cuenta criterios de estandarización y modularización de las soluciones que permitan un fácil mantenimiento y expansión posterior del sistema. – Las características de la unidad central y de los periféricos del sistema se eligen de acuerdo con las prestaciones requeridas por las aplicaciones y especificadas previamente.

REALIZACIONES PROFESIONALES REFERENCIADAS A LA OCUPACIÓN	CRITERIOS BÁSICOS DE REALIZACIÓN
	– El "software" de base se selecciona de acuerdo con el tipo de arquitectura informática adoptada y con el tipo de aplicaciones que se van a implantar. – La topología del sistema de comunicaciones se elige de acuerdo con los requerimientos del sistema y utilizando un modelo de referencia estándar suficientemente reconocido. – Los equipos, tarjetas y demás elementos auxiliares utilizados en el sistema de comunicación cumplen las especificaciones y las normas establecidas en las especificaciones del sistema. – Los programas de comunicación se eligen de acuerdo con los requerimientos del sistema y con el tipo de información que se va a transmitir. – El sistema de cableado y el tipo de soporte utilizado para el sistema de comunicación se configuran en función de las distancias existentes entre los distintos nodos del sistema, la velocidad necesaria para la transmisión de los datos, las condiciones medioambientales y los costes establecidos. – Los croquis y diagramas de bloque del sistema reflejan con precisión la estructura del sistema y los distintos elementos que lo componen. – Las pruebas y ajustes necesarios que se deben realizar en la puesta en marcha y configuración del sistema están especificados con precisión.
Elaborar la documentación técnica del sistema informático y de comunicación industrial (planos, listas de equipos y dispositivos, "software" de base, programas de comunicación), que permita la construcción y el posterior mantenimiento de dichos sistemas, en el soporte adecuado y con los medios normalizados.	– La memoria descriptiva del sistema explica con precisión las características y ámbito de aplicación del mismo. – La documentación técnica incluye los esquemas y planos de conjunto y de detalle necesarios, utilizando la simbología y presentación normalizadas.

REALIZACIONES PROFESIONALES REFERENCIADAS A LA OCUPACIÓN	CRITERIOS BÁSICOS DE REALIZACIÓN
	– La relación de materiales, equipos y dispositivos se realiza utilizando la codificación normalizada, garantizando su adquisición interna y/o externa. – Los planos constructivos del sistema recogen con suficiente precisión las características de los equipos para su implantación (dimensiones físicas, localización de dispositivos y tarjetas, identificación codificada de E/S y de cableados, etc.). – El "software" de base y los programas de comunicación del sistema están suficientemente documentados y permiten la implantación y el posterior mantenimiento de las funciones de los mismos. – La documentación técnica contiene todos los capítulos necesarios y normalizados internamente para el posterior desarrollo de la ejecución y mantenimiento del sistema, incluyendo, entre otros: • Proceso que hay que seguir en la puesta en servicio. • Pruebas y ajustes que hay que realizar en el proceso de puesta en marcha del sistema. • Parámetros que se deben verificar y ajustar. • Márgenes estables de funcionamiento. • Pautas para la realización del mantenimiento preventivo del sistema.
Realizar, a su nivel, la instalación del "hardware" del sistema informático y de comunicación industrial, configurando los parámetros y realizando las pruebas necesarias para la puesta en servicio de dicho sistema, optimizando las características funcionales y de fiabilidad establecidas.	– La instalación y la configuración física de la unidad central del sistema informático, con sus parámetros correspondientes, se realizan de acuerdo con la documentación del sistema y/o de los equipos que lo conforman. – La instalación y la configuración física de los periféricos básicos del sistema informático, con sus parámetros correspondientes, se realizan de acuerdo con la documentación del sistema y/o de los equipos que lo conforman.

REALIZACIONES PROFESIONALES REFERENCIADAS A LA OCUPACIÓN	CRITERIOS BÁSICOS DE REALIZACIÓN
	– La instalación y la configuración física del sistema de comunicación (tarjetas de comunicación, modems, etc.), con sus parámetros correspondientes, se realizan de acuerdo con la documentación del sistema y/o de los equipos que lo conforman. – Los cableados y conexionados del sistema se realizan asegurando un contacto eléctrico fiable y una adecuada sujeción mecánica de los mismos. – Las condiciones ambientales (temperatura, humedad, etc.) están dentro de los márgenes requeridos por el sistema, tomando, en caso contrario, las medidas oportunas para garantizar su cumplimiento. – La instalación eléctrica para el suministro de energía al sistema reúne los requisitos prescritos en la documentación técnica del mismo. – La instalación de tierra del sistema cumple las características prescritas en la documentación técnica del proyecto y, en todo caso, la normativa electrotécnica vigente. – Las pruebas funcionales de la instalación física del sistema informático y de comunicación industrial aseguran la conformidad del mismo con respecto a los requerimientos establecidos en la documentación de dicho sistema. – Las pruebas del sistema de alimentación ininterrumpida (S.A.I.), en su caso, aseguran una adecuada respuesta ante fallos fortuitos del suministro de energía eléctrica habitual. – El montaje de la instalación del sistema se efectúa de acuerdo con el plan establecido, comunicando y/o resolviendo las incidencias surgidas durante la ejecución del mismo.

REALIZACIONES PROFESIONALES REFERENCIADAS A LA OCUPACIÓN	CRITERIOS BÁSICOS DE REALIZACIÓN
Realizar, a su nivel, la instalación del "software" del sistema informático y de comunicación industrial, configurando los parámetros y realizando las pruebas necesarias para la puesta en servicio de dicho sistema, optimizando las características funcionales y de fiabilidad requeridas.	– La carga del "software" de base se realiza siguiendo el procedimiento establecido en la documentación del mismo. – La introducción de los parámetros requeridos por el "software" de base optimiza el aprovechamiento de los recursos del sistema. – La instalación del "software" de comunicación se efectúa siguiendo el procedimiento establecido en la documentación del mismo. – La introducción de los parámetros requeridos por el "software" de comunicaciones asegura el adecuado proceso de transmisión de información entre los distintos puntos del sistema. – Las pruebas de funcionalidad del "software" de base se realizan de acuerdo con el procedimiento establecido, asegurando la fiabilidad del sistema y el óptimo aprovechamiento de los recursos del mismo. – Las pruebas de funcionalidad del sistema de comunicación se realizan siguiendo el procedimiento establecido, asegurando el intercambio de información con la velocidad adecuada y una transmisión exenta de errores. – Las copias de seguridad de la configuración del sistema se efectúan de acuerdo con los requerimientos de dicho sistema, asegurando el rápido reestablecimiento de su operatividad ante fallos del mismo.
Realizar, a su nivel, el mantenimiento de los sistemas informáticos y de comunicación industrial, identificando, en su caso, las causas de avería, tomando las medidas oportunas para la previsión de fallos y para restablecer, en caso de avería, la capacidad de dichos sistemas en condiciones de tiempo, fiabilidad y calidad requeridas.	– El mantenimiento preventivo del sistema se efectúa aplicando el protocolo normalizado y con la periodicidad establecida. – El sistema de prevención contra virus informáticos se mantiene actualizado. – Las copias de seguridad del "software" de base y de los programas de comunicación se efectúan con la periodicidad y en los casos que se determinen.

REALIZACIONES PROFESIONALES REFERENCIADAS A LA OCUPACIÓN	CRITERIOS BÁSICOS DE REALIZACIÓN
	– Ante una avería en un sistema informático y/o de comunicación industrial: • Las pruebas funcionales iniciales permiten verificar los síntomas recogidos en el parte de avería y, en todo caso, precisar la sintomatología de la disfunción y la naturaleza de la misma (física y/o lógica). • La hipótesis de partida y el plan de actuación elaborado permiten diagnosticar y localizar con precisión el tipo (físico y/o lógico) y el bloque funcional o módulo donde se encuentra la avería. • El diagnóstico y localización de la avería del sistema (del equipo y/o instalación) se realizan utilizando la documentación técnica del mismo, las herramientas y los instrumentos de medida apropiados, aplicando el correspondiente procedimiento en un tiempo adecuado. • El presupuesto recoge con precisión la tipología y coste de la reparación. • Las operaciones de montaje, desmontaje y sustitución de los elementos (del equipo y/o instalación) se realizan utilizando la documentación técnica (planos y procedimientos normalizados) y las herramientas apropiadas, asegurando la integridad de los materiales y medios utilizados y la calidad final de las intervenciones. • Los ajustes de los dispositivos y/o equipos sustituidos se realizan utilizando las herramientas y útiles específicos, con la precisión requerida, siguiendo los procedimientos documentados. • Las pruebas funcionales, ajustes finales y, en caso necesario, las pruebas de fiabilidad recomendadas, se realizan de forma sistemática, siguiendo el procedimiento especificado en la documentación del sistema.

REALIZACIONES PROFESIONALES REFERENCIADAS A LA OCUPACIÓN	CRITERIOS BÁSICOS DE REALIZACIÓN
	• La reparación del equipo y/o instalación se realiza respetando las normas de seguridad personal, de los equipos y materiales recomendadas en la documentación de los mismos y, en todo caso, siguiendo las pautas del buen hacer profesional. • El informe de reparación de averías del sistema se realiza en el formato normalizado, recogiendo la información suficiente para realizar la facturación de la intervención y actualización del histórico de averías del equipo y/o instalación.

3. Capacidades terminales y criterios de evaluación

En este apartado se describen las capacidades terminales y sus correspondientes criterios de evaluación, correspondientes al Real Decreto del título, a partir de las realizaciones planteadas en el apartado anterior.

El título profesional y por tanto las competencias que adquieren los alumnos que realizan este ciclo formativo está basado en la suma de las diferentes capacidades terminales que se adquieren con cada uno de los módulos que forman el ciclo formativo.

Las capacidades terminales del módulo Comunicaciones Industriales, así como sus correspondientes criterios de evaluación, según el Real Decreto del currículo publicado en el BOE, son:

CAPACIDADES TERMINALES	CRITERIOS DE EVALUACIÓN
Analizar los sistemas de comunicación industrial, identificando los distintos elementos que los componen y relacionando la función de cada uno de ellos con el funcionamiento y prestaciones globales del sistema.	– Explicar la función que un sistema de comunicación industrial tiene y las posibilidades que ofrece en cuanto a integración y comparación de recursos y funciones. – Describir la estructura que tiene un sistema de comunicación industrial, indicando los niveles funcionales y operativos que incluye y sus campos de aplicación más característicos.

CAPACIDADES TERMINALES	CRITERIOS DE EVALUACIÓN
	– Enumerar las características más relevantes que caracterizan un entorno industrial de control de procesos distribuido y de un entorno C.I.M. ("Computer Integrated Manufacturing"). – Explicar el modelo de referencia OSI ("Open System Interconnection") de ISO ("International Standard Organization"), describiendo la función de cada uno de sus niveles y la relación entre ellos. – Explicar la función que desempeña un protocolo de comunicación y su importancia con fines de estandarización, citando los más utilizados en el ámbito industrial. – Clasificar las técnicas de transmisión de datos en función de la tecnología empleada (analógica o digital), el tipo (síncrona o asíncrona) y la modulación utilizada, explicando las características y aplicaciones de cada una de ellas. – Explicar la función que realiza un "módem" en el proceso de comunicación de datos, enumerando las distintas normas que están aceptadas para su estandarización, indicando los parámetros (velocidad de transmisión, tipo de línea de transmisión, tipo de modulación, etc.) y características de cada una de ellas.
Elaborar programas básicos de comunicación entre un ordenador y periféricos externos de aplicación industrial (autómatas, instrumentos de medida, controladores, etc.), en serie y en paralelo, utilizando interfaces y protocolos normalizados.	– Describir el conector estándar correspondiente a la interfase serie RS232-C, indicando la función de cada una de las líneas del mismo. – Describir el conector estándar correspondiente a la interfase paralelo "Centronics", indicando la función de cada una de las líneas del mismo. – En casos prácticos de realización de programas para la comunicación entre un ordenador y un periférico siguiendo las normas RS232-C en un caso y la norma RS-485 en otro:

CAPACIDADES TERMINALES	CRITERIOS DE EVALUACIÓN
	• Identificar con precisión las características del periférico que formará parte de la comunicación y las especificaciones de la comunicación. • Determinar el protocolo de comunicación que se ajusta de forma más adecuada a las características del periférico, asegurando el mínimo de errores en dicha comunicación. • Elaborar el diagrama de flujo correspondiente, utilizando simbología normalizada. • Codificar el programa de comunicación en el lenguaje adecuado. • Verificar la idoneidad del programa con el diagrama de flujo elaborado y con las especificaciones propuestas. • Documentar adecuadamente el programa, aplicando los procedimientos estandarizados y con la suficiente precisión para asegurar su posterior mantenimiento.
Determinar los requisitos necesarios para la implantación y puesta a punto de una red local de comunicación (con las características específicas de un entorno industrial), realizando la configuración física de la misma, cargando los programas e introduciendo los parámetros necesarios del "software" de base de acuerdo con el tipo de aplicaciones que se van a utilizar.	– Explicar las distintas configuraciones topológicas propias de las redes locales, indicando las características diferenciales y de aplicación de cada una de ellas. – Describir la estructura física de una red local de ordenadores, enumerando las tipologías de equipos, de medios físicos, de modos de conexión y estándares empleados y describiendo la función que desempeña cada uno de ellos. – Enumerar y justificar los criterios más usuales utilizados en la selección de una red local (tiempo de respuesta, volumen de datos que se debe transferir, distancias, privacidad y control de accesos, acceso a otras redes, etc.). – Explicar las funciones y posibilidades del sistema operativo de red, su estructura en módulos, describiendo las prestaciones de cada uno de ellos.

CAPACIDADES TERMINALES	CRITERIOS DE EVALUACIÓN
	– Citar los recursos que se pueden compartir en una red local de ordenadores y los modos usuales de utilización de los mismos. – Enumerar los tipos de soporte de transmisión (cables y fibra óptica) utilizados en las redes locales de comunicación, indicando las características y parámetros más representativos de los mismos. – Exponer las características propias y diferenciales de las redes locales de ordenadores y las redes de autómatas programables, indicando las posibilidades de interconexión entre ellas. – En un caso práctico de implantación y puesta en marcha de un sistema informático para trabajar en red local: • Interpretar la documentación de la red (tanto del sistema físico como del sistema operativo), confeccionando los materiales intermedios necesarios para la implantación real de la misma. • Preparar la instalación de suministro de energía eléctrica y, en su caso, el sistema de alimentación ininterrumpida, comprobando la independencia de los circuitos de suministro y las condiciones de seguridad eléctrica y medioambiental requeridas. • Realizar el conexionado físico de las tarjetas, equipos y demás elementos necesarios para la ejecución de la red, siguiendo el procedimiento normalizado y/o documentado. • Efectuar la carga del sistema operativo de la red, siguiendo el procedimiento normalizado e introduciendo los parámetros necesarios para adecuarla al tipo de aplicaciones que se van a utilizar. • Realizar la organización del espacio de almacenamiento del servidor de archivos, asignando el tamaño y los accesos requeridos en función de las prestaciones requeridas por cada usuario.

CAPACIDADES TERMINALES	CRITERIOS DE EVALUACIÓN
	• Preparar el sistema de seguridad y confidencialidad de la información, utilizando los recursos de que dispone el sistema operativo de la red. • Optimizar la configuración que responde a los recursos compartidos por los usuarios de la red. • Efectuar la carga de los programas de utilidades generales y específicos que van a ser utilizados por los usuarios de la red, optimizando sus prestaciones y facilitando su uso. • Documentar el proceso realizado, elaborando los documentos particulares para cada usuario que le facilite la utilización fiable y segura de las aplicaciones que funcionan en la red, así como la integridad de funcionamiento de la misma.
Analizar los buses de campo utilizados en el ámbito industrial, identificando los distintos elementos que los integran y relacionándolos con el resto de elementos que configuran los sistemas automáticos.	– Definir qué es un bus de campo y explicar sus aplicaciones en los procesos de control industrial. – Explicar las características fundamentales de un bus de campo y la capacidad de integración de instrumentación inteligente con sistemas superiores de tratamiento de información. – Enumerar las ventajas que aporta la instrumentación inteligente a los sistemas de control de procesos industriales (introducción de parámetros a distancia, realizar diagnósticos, evaluar datos, etc.). – Describir la estructura que tiene un sistema basado en un bus de campo, integrando los distintos dispositivos que lo pueden configurar y los elementos de base que lo conforman. – Contrastar las ventajas e inconvenientes de utilizar la comunicación analógica estándar de 4-20 miliamperios y la comunicación digital de alta velocidad característica de los buses de campo.

CAPACIDADES TERMINALES	CRITERIOS DE EVALUACIÓN
	– Indicar los buses de campo reconocidos como estándares actuales, FIP ("Factory Information Protocol"), PROFIBUS ("PROcess Field BUS"), indicando sus características específicas y diferenciales. – Explicar cuáles son las características del modelo OSI reducido que utilizan los buses de campo, indicando las características fundamentales en cada uno de los niveles del m odelo. – Describir la estructura de datos que configura una trama de información utilizada en un bus de campo tipo FIP.
Realizar, con precisión y seguridad, medidas en los sistemas de comunicación industrial, utilizando los instrumentos y los elementos auxiliares apropiados y aplicando el procedimiento más adecuado en cada caso.	– Explicar las características más relevantes, la tipología y procedimientos de uso de los instrumentos de medida utilizados en el campo de las comunicaciones industriales, en función de la naturaleza de las magnitudes que se deben medir y del tipo de tecnología empleada (analógica o digital). – En el análisis y estudio de distintos casos prácticos de sistemas de comunicación industrial, donde intervengan variables de distintas tecnologías con sus correspondientes magnitudes físicas: • Seleccionar el instrumento de medida y los elementos auxiliares más adecuados en función del tipo y naturaleza de las magnitudes que se van a medir y de la precisión requerida. • Conexionar adecuadamente los distintos aparatos de medida en función de las características de las magnitudes que se van a medir. • Medir los parámetros propios de los equipos y dispositivos utilizados, operando adecuadamente los instrumentos (monitor de actividad, medidor de tasa de error, analizador de protocolos) y aplicando, con la seguridad requerida, los procedimientos normalizados.

CAPACIDADES TERMINALES	CRITERIOS DE EVALUACIÓN
	• Interpretar la información que corresponde al mensaje y la que es propia del protocolo de comunicación utilizado. • Interpretar las medidas realizadas, relacionando los estados y valores de las magnitudes medidas con las correspondientes de referencia, señalando las diferencias obtenidas y justificando los resultados. • Elaborar un informe-memoria de las actividades desarrolladas y resultados obtenidos, estructurándolo en los apartados necesarios para una adecuada documentación de las mismas (descripción del proceso seguido, medios utilizados, esquemas y planos, explicación funcional, medidas, cálculos, etc.).
Diagnosticar averías en sistemas de comunicación industrial, identificando la naturaleza de la avería, aplicando los procedimientos y técnicas más adecuadas en cada caso.	– Clasificar y explicar la tipología y características de las averías de naturaleza física que se presentan en los sistemas de comunicación industrial. – Clasificar y explicar la tipología y características de las averías de naturaleza lógica que se presentan en los sistemas de comunicación industrial. – Describir las técnicas generales y los medios técnicos específicos necesarios para la localización de averías de naturaleza física en un sistema de comunicación industrial. – Describir las técnicas generales y los medios técnicos específicos necesarios para la localización de averías de naturaleza lógica en un sistema de comunicación industrial. – Describir el proceso general utilizado para el diagnóstico y localización de averías de naturaleza física y/o lógica en un sistema de comunicación industrial. – En varios supuestos y/o casos prácticos de diagnóstico y localización de averías en un sistema de comunicación industrial (red local de autómatas y/u ordenadores):

CAPACIDADES TERMINALES	CRITERIOS DE EVALUACIÓN
	• Interpretar la documentación del sistema de comunicación industrial en cuestión, identificando los distintos bloques funcionales y componentes específicos que lo componen. • Identificar los síntomas de la avería caracterizándola por los efectos que produce. • Realizar al menos una hipótesis de la causa posible que puede producir la avería, relacionándola con los síntomas (físicos y/o lógicos) que presenta el sistema. • Realizar un plan de intervención en el sistema para determinar la causa o causas que producen la avería. • Localizar el elemento (físico o lógico) responsable de la avería y realizar la sustitución (mediante la utilización de componentes similares o equivalentes) o modificación del elemento, configuración y/o programa, aplicando los procedimientos requeridos (comprobación de cableados, monitorizado de actividad, análisis de protocolos, etc.) y en un tiempo adecuado. • Realizar las comprobaciones, modificaciones y ajustes de los parámetros del sistema según las especificaciones de la documentación técnica del mismo, utilizando las herramientas apropiadas, que permitan su puesta a punto en cada caso. • Elaborar un informe-memoria de las actividades desarrolladas y resultados obtenidos, estructurándolo en los apartados necesarios para una adecuada documentación de las mismas (descripción del proceso seguido, medios utilizados, medidas, explicación funcional y esquemas).

4. Orientaciones metodológicas

Se van a exponer una serie de orientaciones metodológicas encaminadas a conseguir que el alumno conozca la importancia del tema Comunicaciones Industriales dentro del proceso productivo de cualquier industria, servicio, residencia, etc., y se interese "profesionalmente" en esta materia técnica.

Los temas deben exponerse en un lenguaje sencillo a la vez que técnico para que el alumno, futuro profesional, vaya conociendo la terminología y el argot que se utiliza en el campo de los sistemas de regulación y de control automático: comunicaciones industriales.

Los diferentes temas que componen el módulo son materias difícilmente transportables al aula, por lo que debemos realizar el trabajo en los talleres y valernos de material gráfico como diapositivas, vídeos, catálogos comerciales, etc.

Utilizar información técnico-comercial, de empresas o distribuidores de la zona, para que los alumnos conozcan los materiales, características, aplicaciones, formas de comercialización, etc.

Inculcar la idea de trabajo en equipo, o de cuadrillas, diseñando los trabajos o actividades por equipos de alumnos (2 o 3 por actividad) que es lo que se van a encontrar después en el mundo del trabajo.

Plantear las prácticas a partir del orden de ejecución de las tareas, la exactitud en los montajes y las conexiones, las verificaciones y comprobaciones de las máquinas y sobre todo guardar y hacer guardar las normas básicas de seguridad.

Como idea de cómo desarrollar este módulo en el aula, se reproduce parte del prólogo del libro que describe una metodología de acción práctica:

Los sistemas automáticos de regulación y control han alcanzado tal nivel de integración en las plantas y sistemas de producción, que no se concibe una explotación de los datos sin el soporte de las comunicaciones industriales. Así pues, los técnicos y técnicas que incorporen a su saber los conceptos y procedimientos en materia de comunicaciones industriales, sobre los sistemas automáticos ya citados, resolverán las tareas y actividades laborales: selección, instalación, elaboración de informes, puesta en marcha, sustitución, mantenimiento, supervisión...

Por otro lado, un campo como el de las *Comunicaciones Industriales* no tiene cabida *ni en un texto oficial ni en un libro de texto, ni tampoco en Internet*. Es por ello que nuestro objetivo ha sido ofrecer un material para cubrir la duración del módulo y, además, ofrecemos unos materiales que permitan realizar el curso y consultar aspectos y conceptos fundamentales. ¿Por qué?, tenemos múltiples respuestas:

- Es evidente que todo el texto no puede ser abordado en el corto espacio de tiempo que se utiliza en este módulo, pero creemos que la inversión en un libro de texto debe ser algo más que para cubrir un periodo académico. Es por ello que hemos incluido la mayor parte de los conceptos fundamentales de las redes de comunicación, ilustrados con ejemplos. Y por otro lado, unas propuestas de programación que creemos que están acordes con las necesidades de comunicación actuales, sobre todo en la segunda parte del texto. Indudablemente no hemos pretendido cubrir el espacio de los grandes textos sobre la materia, ya que los hay de muy variado enfoque y profundidad.
- Si este libro se utiliza en un centro de formación con recursos escasos, pueden aprovecharse los procedimientos y múltiples posibilidades abordadas desde las redes locales de ordenadores y autómatas programables. Así pues, con no muchos dispositivos: ordenadores y autómatas programables, pueden realizarse programas y experimentar procedimientos, imprescindibles para adquirir las competencias del título profesional. Bastará un entorno de programación, ordenador compatible, y un autómata programable, de los que se puede obtener información sobre el protocolo de comunicación del fabricante.
- Si este libro se utiliza en un centro de formación bien dotado de tecnologías de comunicación industrial, además de ordenadores y autómatas programables, con controladores de procesos y diferentes elementos, con comunicaciones, como variadores de velocidad, podrán utilizar todas las técnicas de realización y verificación de procedimientos técnicos que exponemos en el texto.

5. Índice secuencial de las unidades de trabajo: organización de los contenidos

El módulo de Comunicaciones Industriales se estructura a partir de los siguientes contenidos básicos:

- El control distribuido y la integración en los procesos.
- Teleinformática.
- Transmisión de datos.
- Protocolos de comunicación.

- Comunicaciones en serie y en paralelo.
- Redes locales.
- Buses de campo.
- Procedimientos en el área de las comunicaciones industriales.

Desarrollados en las siguientes unidades de trabajo:

1. Principios de comunicación en los entornos industriales.
2. Principios de teleinformática: tipos de comunicación y transmisión de datos.
3. Nivel de enlace de datos: transferencia de datos.
4. Redes de área local LAN.
5. Acceso a los puertos de E/S en un PC.
6. El puerto paralelo.
7. El puerto serie RS-232.
8. Buses de campo.
9. Procedimientos de comunicación con dispositivos industriales.
10. Desde C hasta C++.
11. Clase y Objeto en C++.
12. C++ en acción.
13. Programación a bajo nivel desde DOS.
14. Programación del puerto paralelo.
15. Programación del puerto serie.
16. C++ Builder.
17. Entrada / salida en C++ Builder.
18. La API del puerto serie.

6. Estructura de las unidades de trabajo del libro del alumno

Cada una de las unidades didácticas o capítulos del libro está compuesta por los siguientes apartados:

- Introducción.
- Contenidos.
- Objetivos.
- Desarrollo de los contenidos.
- Actividades, problemas o prácticas propuestas.

7. Distribución temporal de las unidades de trabajo

Según se indicaba en el apartado 2 de esta guía, este módulo se imparte en el 2º curso del ciclo formativo y tiene una duración de 130 horas lectivas, a razón de 6 horas a la semana.

La media de tiempo para el desarrollo de las diferentes unidades o capítulos que forman el módulo es de 7 horas por capítulo.

Las 18 unidades que componen este módulo son:

Capítulo 1. Principios de comunicación en los entornos industriales.

Capítulo 2. Principios de teleinformática: tipos de comunicación y transmisión de datos.

Capítulo 3. Nivel de enlace de datos: transferencia de datos.

Capítulo 4. Redes de área local LAN.

Capítulo 5. Acceso a los puertos de E/S en un PC.

Capítulo 6. El puerto paralelo.

Capítulo 7. El puerto serie RS-232.

Capítulo 8. Buses de campo.

Capítulo 9. Procedimientos de comunicación con dispositivos industriales.

Capítulo 10. Desde C hasta C++.

Capítulo 11. Clase y Objeto en C++.

Capítulo 12. C++ en acción.

Capítulo 13. Programación a bajo nivel desde DOS.

Capítulo 14. Programación del puerto paralelo.

Capítulo 15. Programación del puerto serie.

Capítulo 16. C++ Builder.

Capítulo 17. Entrada / salida en C++ Builder.

Capítulo 18. La API del puerto serie.

8. Elementos curriculares o unidades de trabajo

Los elementos curriculares que definen cada una de las unidades de trabajo o capítulos del libro son:

Capítulo 1. Principios de comunicación en los entornos industriales.

Introducción

Las necesidades de comunicación ya han sido experimentadas por el lector. Sin duda, ha establecido conversaciones telefónicas e incluso habrá participado en alguna conexión a través de Internet. En el campo industrial, las decisiones que exigen tomar las necesidades de la producción precisan ser llevadas a los dispositivos de control a pie de máquina o campo. Es decir, utilizaremos los medios de comunicación para conectar entre sí los diferentes dispositivos que intervienen en los procesos de producción automatizados.

El control distribuido se ha convertido en el criterio y concepto que deben seguir todos los sistemas de comunicación industrial. El propio término nos indica la distribución de responsabilidades en todo el proceso de control de una compañía o empresa, ya sea grande o mediana. El entorno CIM utiliza las redes de comunicación y es el origen de que se necesiten técnicos con competencias en comunicaciones industriales.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Explicar la función que un sistema de comunicación industrial tiene y las posibilidades que ofrece en cuanto a integración y compartición de recursos y funciones. – Describir la estructura de un sistema de comunicación industrial, indicando los niveles funcionales y operativos. – Enumerar las características más relevantes que caracterizan un entorno industrial de control de procesos distribuido y de un entorno CIM (Computer Integrated Manufacturing). – Explicar el modelo de referencia OSI (Open System Interconnection) de ISO (International Standard Organization).	l.1. El proceso de la comunicación. 1.1.1 Componentes de la comunicación. 1.1.2. Particularidades de la comunicación. 1.1.3. Una aproximación con aplicación industrial. 1.2. Entorno CIM (Computer Integrated Manufacturing). 1.2.1. Tecnologías en el entorno CIM. 1.2.2. Niveles de integración en un sistema CIM. 1.3. Redes de comunicación. 1.3.1. El modelo cliente/servidor y los sistemas distribuidos.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Explicar la función que desempeña un protocolo de comunicación y su importancia. – Describir las funciones, facilidades y prestaciones que ofrecen las redes de comunicación de datos y relacionarlas con las comunicaciones industriales. – Identificar y aplicar los estándares relacionados con las comunicaciones industriales.	1.3.2. Redes de conmutacion. 1.3.3. Redes de área amplia (WAN) y redes de área local (LAN). 1.4. Funciones de un sistema de comunicacion industrial. 1.5. Protocolos de comunicaciones: modelo OSI. 1.5.1. Funciones de los protocolos. 1.5.2. El modelo OSI. 1.6. Introducción a los niveles de comunicacion industrial. 1.6.1. Mejoras para el mantenimiento y la puesta en marcha. 1.7. Normalización de las comunicaciones.

Capítulo 2. Principios de teleinformática: tipos de comunicación y transmisión de datos.

Introducción

Por teleinformática se entiende el conjunto de aportaciones de dos disciplinas de la ingeniería, como son telecomunicaciones e informática, y la conexión de equipos a distancia que permite el tratamiento automático de la información. En el capítulo primero hemos presentado los niveles de comunicación para los procesos de fabricación integrados. Además, se han ilustrado las redes de comunicación, como el medio o soporte para llevar a cabo la estructura piramidal CIM.

En este capítulo comenzamos a desmenuzar los componentes necesarios del mundo de la teleinformática, para abordar con éxito las comunicaciones industriales. Como elementos de la comunicación nos referimos a todos aquellos conceptos que van a ser decisorios para distinguir unos tipos de red de otra: envío de caracteres, formato de mensajes, cadenas de bits, velocidad de transmisión... También, desde el punto de vista de un tipo de comunicación, se establecen varios criterios que van a distinguir aspectos sobre el tipo de señal que se transmite o si existe sincronización o no entre los mensajes y los dos extremos de la comunicación.

Describiremos las técnicas de transmisión de datos y los conceptos de codificación y de modulación y sus tipos. Fundamentales para el intercambio de información en el plano físico (señales eléctricas, cables y en el conexionado de tarjetas electrónicas). Y de esta forma describir la plataforma sobre la que se ejecutan los programas de comunicación.

Finalmente describiremos los aspectos fundamentales sobre los módem, los cuales representan uno de los dispositivos más antiguos en cuanto a su utilización en las redes de comunicaciones con ordenadores. Además, en la conexión y programación de estos dispositivos podremos establecer muchos de los "trucos" básicos de las comunicaciones.

OBJETIVOS	**CONOCIMIENTOS (CONTENIDOS SOPORTE)**
– Clasificar técnicas de transmisión de datos en función de tecnologías, tipos, modulación y codificación, tamaño de los datos transmitidos. – Explicar las funciones de corrección de errores. – Distinguir los medios de transmisión y los parámetros de selección. – Explicar las funciones de un módem y sus parámetros y características.	2.1. Elementos que intervienen en la comunicación. 2.1.1. Mensajes. 2.1.2. Caracteres y códigos de representación. 2.1.3. Detección de errores. 2.1.4. Cadenas de bits, tramas y velocidad de transmisión. 2.2. Conceptos de señales y transmisión de datos. 2.2.1. Señales. 2.2.2. Espectro de frecuencia y ancho de banda. 2.2.3. Conflictos en la transmisión. 2.2.4. Medios de transmisión. 2.3. Clasificación de las comunicaciones de datos. 2.3.1. Simultaneidad de datos en el medio de transmisión. 2.3.2. Disposición de las estaciones de trabajo. 2.3.3. Simultaneidad de emisión y recepción. 2.3.4. Tipo de sincronismo. 2.3.5. Tipo de señal transmitida: comunicación analógica o digital. 2.4. Técnicas de transmisión de datos. 2.4.1. Multiplexación. 2.4.2. Banda base.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
	2.4.3. Banda ancha. 2.5. Introducción al módem. 2.5.1. Tipos de modulación. 2.5.2. Tipos conexión al DTE. 2.5.3. Velocidad del módem. 2.5.4. Funciones especiales en los módem. 2.5.5. Comandos Hayes. 2.5.6. Estándares para módems.

Capítulo 3. Nivel de enlace de datos: transferencia de datos.

Introducción

Hasta el momento hemos estudiado las técnicas y medios de transmisión. Con un soporte o interfaz físico, podemos realizar la conexión entre dos estaciones. Es decir, podemos preparar a estas últimas para que intercambien "chorros" de bits en serie.

Para que estos bits tengan sentido, en cada extremo se debe acordar un protocolo de enlace de datos. Al recibir una trama de bits, se producen sobre la misma una serie de transformaciones. Esto ya se introdujo en el capítulo 2: "tramas". En este capítulo se describirán las técnicas del control y la gestión de las tramas. Esto corresponde al nivel 2 de enlace de datos del modelo OSI.

Es decir, con las estrategias y protocolos de este capítulo, podemos crear programas –en lenguaje C o C++– que traten las tramas de datos, siempre que se haya resuelto en cada extremo:

- La misma interfaz física: RS-232, por ejemplo.
- El protocolo de intercambio de datos: HDLC, Xmódem.
- Se haya decidido una aplicación origen y destino de los datos o información extraída de las tramas, por medio de la aplicación del protocolo.

Las tramas que viajan entre estaciones son identificadas por los programas que corren con la responsabilidad de la puesta en funcionamiento del protocolo. Cuando se intercam-

bian ficheros entre dos estaciones, los datos se dividen para que quepan en las tramas. Las tramas tienen un tamaño máximo de información. El tamaño viene a ser fijo. Es decir si hay que enviar un fichero de 4.500 bytes, y el tamaño de información o datos de la trama es de 64 bytes, se precisan 71 tramas para transmitir completamente el fichero.

OBJETIVOS	**CONOCIMIENTOS (CONTENIDOS SOPORTE)**
– Establecer las problemáticas de la transmisión en el nivel de enlace. – Explicar las funciones detección y corrección de errores. – Distinguir y aplicar los fenómenos de inundación y las técnicas de corrección. – Explicar las características más importantes de los protocolos de enlace de datos. – Realizar funciones con tramas del protocolo Xmódem.	3.1. Funciones del enlace de datos. 3.2. Delimitadores, sincronización y transparencia de datos. 3.2.1. Delimitadores y sincronización. 3.2.2. Transparencia de datos. 3.3. Detección de errores. 3.3.1. Acciones ante la detección de errores. 3.4. Control de la transmisión: control de flujo. 3.4.1. Coordinación en la comunicación. 3.4.2. Control de desbordamiento de tramas. 3.4.3. Estrategias para el control de la transmisión. 3.5. Control de la transmisión: control de errores. 3.5.1. Parada y espera con errores y control de duplicados. 3.5.2. Control de errores con el método de ventana deslizante. 3.5.3. Tramas especiales. 3.6. Protocolo Xmódem. 3.6.1. Formato de la trama. 3.6.2. Tramas especiales. 3.6.3. Los números de secuencia. 3.6.4. Esquema de funcionamiento. 3.6.5. Paralelismo con Host-Link, de Omron Electronics, S.A. 3.6.6. Resumen.

Capítulo 4. Redes de área local LAN.

Introducción

En el capítulo primero se centraron los contenidos sobre la fabricación integrada y los protocolos de comunicación. En el segundo hemos entrado en la terminología, las técnicas y los medios de transmisión aplicados en las comunicaciones industriales. Después nos hemos dedicado a la realización de enlaces punto a punto por medio de las interfaces serie y paralelo del PC. En este capítulo desarrollaremos las técnicas que nos permitirán, utilizando la documentación adecuada, identificar los requisitos y características de una red local. Abordaremos la caracterización y diferencias entre red de ordenadores, tipo oficina, y la red que incluye equipos de automatización (A.P., controladores, terminales de operario...).

Analizaremos tipos de redes que siguen estándares, así como redes que utilizan sistemas abiertos y propietarios para comunicar los niveles descritos por el modelo CIM.

Para ello abordamos los parámetros fundamentales, los medios necesarios (hardware y software) y los recursos que pueden compartirse cuando se utilizan las redes locales. En las actividades dejaremos el tema a punto para que se intente realizar una red local siguiendo las pautas de un sistema operativo de red como Microsoft Windows 98.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Explicar las distintas configuraciones topológicas propias de las redes locales, indicando las características diferenciales y de aplicación de cada una de ellas. – Describir la estructura física de una red local de ordenadores, enumerando las tipologías de equipos, de medios físicos, de modos de conexión y estándares empleados y describiendo la función que desempeña cada uno de ellos. – Enumerar y justificar los criterios más usuales utilizados en la selección de una red local (tiempo de respuesta, volumen de datos que se debe transferir, distancias, privacidad y control de accesos, acceso a otras redes).	4.1. Caracterización de una red local. 4.1.1. Consideraciones fundamentales. 4.1.2. Estándares en redes de área locales. 4.2. Redes locales: topología. 4.2.1. Topología en bus y en árbol. 4.2.2. Topología en estrella. 4.2.3. Topología en anillo. 4.2.4. Otras topologías. Topología física y topología lógica. 4.3. Técnicas de acceso al medio en redes locales. 4.3.1. Centralizado: sondeo. 4.3.2. Distribuido: colisiones. 4.3.3. Distribuido: paso de testigo. 4.4. Estándar Ethernet: IEEE 802.3.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Citar los recursos que se pueden compartir en una red local de ordenadores y los modos usuales de utilización de los mismos. – Exponer las características propias y diferenciales de las redes locales de ordenadores y las redes de autómatas programables, indicando las posibilidades de interconexión entre ellas.	4.4.1. Acceso al medio MAC. Trama. 4.4.2. Influencia del medio de transmisión: 10BASE y 100BASE. 4.5. Conexión entre redes locales. 4.5.1. Repetidor. 4.5.2. Puente (bridge). 4.5.3. Encaminador (router). 4.5.4. Pasarelas (gateway). 4.6. Software necesario en redes de comunicación. 4.7. Ethernet y TCP/LP. 4.7.1. Introducción. 4.7.2. Descripción de los niveles TCP/IP. 4.7.3. Nombres y direcciones IP. 4.7.4. Encaminamiento y conexiones entre redes. 4.8. Una guía práctica. 4.8.1. Configuración de una red en Windows 98. 4.8.2. Configuración de aplicaciones TCP/IP con autómatas programables.

Capítulo 5. Acceso a los puertos de E/S en un PC.

Introducción

En los últimos años, la revolución de las tecnologías digitales ha provocado la explosión de los sistemas programables frente a los sistemas cableados, ejemplo de esto son los ordenadores personales, los microcontroladores o los autómatas programables.

Los sistemas programados aportan flexibilidad: sólo con cambiar el programa de una máquina programable ésta puede realizar tareas completamente diferentes. Podemos definir un sistema basado en microprocesador como una estructura hardware que puede ser controlada por un software o programa.

En este capítulo se abordan los sistemas programables desde una perspectiva general, haciendo especial hincapié en los aspectos relacionados con la E/S. Se describirá el funcionamiento del hardware del PC que más se relaciona con las comunicaciones serie y paralelo.

OBJETIVOS	**CONOCIMIENTOS (CONTENIDOS SOPORTE)**
– Conocer la estructura básica de un sistema basado en microprocesador. – Conocer la arquitectura interna del PC. – Conocer el acceso a los puertos de comunicación del PC. – Conocer los fundamentos de las interrupciones hardware.	5.1. Sistemas basados en microprocesador. 5.1.1. El microprocesador. 5.1.2. La memoria principal. 5.1.3. Las unidades de entrada/salida. 5.1.4. Los buses. 5.1.5. El juego de instrucciones. Programas. Nemónicos. 5.2. Direcciones de memoria de entrada/salida. 5.2.1. Direcciones de memoria. 5.2.2. Direccionamiento segmentado. 5.3. Diagramas de flujo. 5.4. La programación modular y el STACK. 5.5. Direcciones de entrada/salida. 5.6. Registros de entrada/salida. 5.7. Técnicas de acceso a puertos. 5.7.1. Acceso a E/S mediante sondeo/scan/poolling. 5.7.2. Acceso a E/S mediante interrupciones hardware. 5.8. Interrupciones hardware. 5.8.1. Definición. 5.8.2. Clasificación. 5.8.3. Prioridades. 5.8.4. Rutina de servicio de interrupción (ISR, Interrup Service/Rutine). 5.8.5. Descripción del funcionamiento. 5.9. El controlador de interrupciones PIC 8259A. 5.10. Tabla de vectores de interrupción (TVI).

Capítulo 6. El puerto paralelo.

Introducción

La interfaz paralelo o CENTRONICS se desarrolló para conectar una impresora al PC. Con el tiempo, su uso se ha extendido a otras áreas, como la adquisición de datos, la conexión de periféricos, etc. El CCITT trata de normalizar la interfaz paralelo a través de la norma IEEE 1284 (1994).

El puerto paralelo de los PC es una interfaz muy flexible, que puede ser utilizada para mucho más que para conectar la impresora: comunicación entre PC, control de dispositivos, adquisición de datos, conexión de periféricos, etc. Desde un punto de vista más genérico, el puerto paralelo puede verse como un grupo de 5 entradas y 12 salidas digitales de propósito general.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Conocer la interfaz paralelo del PC. – Conocer los registros de E/S del puerto paralelo. – Conocer los distintos cables disponibles para el puerto paralelo. – Aprender cómo conectar dispositivos externos a través del puerto paralelo.	6.1. Características físicas. 6.2. Protocolo CENTRONICS. 6.3. Registros del puerto paralelo SPP. 6.4. Tipos de cables. 6.5. Circuitos básicos para monitorizar y controlar líneas. 6.6. Circuitos para control de salidas. 6.6.1. Expansión del número de salidas. 6.6.2. Circuitos para atacar distintos tipos de cargas. 6.7. Circuitos para control de entradas. 6.7.1. Lectura de un byte mediante puertos SPP no bidireccionales. 6.7.2. Lectura de 40 entradas. 6.7.3. Detección de niveles. 6.7.4. Lectura de señales analógicas mediante convertidores A/D. 6.7.5. Detección de rebotes. 6.7.6. Interrupciones hardware. 6.8. Modos para transferencia de datos. 6.8.1. Modo Nibble. 6.8.2. Modo Byte.

Capítulo 7. El puerto serie RS-232.

Introducción

La interfaz serie se basa en la norma RS-232C del EIA (Electronic Industries Association) del año 1969. Esta norma fue adaptada a la norma V24 del CCITT (Comité Consultivo Internacional Telegráfico y Telefónico).

El puerto serie de los PC es la forma más fácil de intercambiar información punto a punto entre dispositivos de distinta o igual naturaleza: PC, autómatas programables, sensores inteligentes, etc.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Conocer la norma RS-232. – Conocer la arquitectura de los puertos serie del PC. – Conocer las técnicas de acceso a los puertos. – Conocer los cables y tipos de enlace posibles con esta norma.	7.1. Características físicas. 7.2. Transmisiones síncronas y asíncronas. 7.3. Descripción de las señales. 7.4. La UART: registros de E/S del puerto serie. 7.4.1 Registro del buffer del receptor: RXR. 7.4.2. Registro del buffer del transmisor: TXR. 7.4.3. Registro de activación de interrupción: IER. 7.4.4. Registro de identificación de interrupción: IIR. 7.4.5. Registro de control de línea: LCR. 7.4.6. Registros del divisor de frecuencia: DLH y DLL. 7.4.7. Registros del control del módem: MCR. 7.4.8. Registro de estado de línea: LSR. 7.4.9. Registro de estado del módem: MSR. 7.4.10. Registro de control HIFO: FCR. 7.5. Transmisión y recepción por sondeo. 7.6. Transmisión y recepción por interrupciones.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
	7.7. Técnicas de control de flujo. 7.7.1. Control de flujo hardware. 7.7.2. Control de flujo software. 7.8. Tipos de cables y enlaces. 7.9. Norma RS-485. 7.9.1. Características. 7.9.2. Tipos de enlaces. 7.9.3. Conversores RS-232 a RS-485. 7.9.4. Norma RS-422 frente a RS-485.

Capítulo 8. Buses de campo.

Introducción

Los buses de campo representan el eslabón final entre los dispositivos de campo (sensores y accionadores) y los esquemas de control y gestión de un sistema de fabricación integrado, por medio de comunicaciones industriales.

Cuando se instala una célula de fabricación flexible, o un conjunto de secciones de fabricación, en medianas y grandes industrias, el bus de campo se está haciendo imprescindible. Mediante los mismos se reduce el cableado, se insertan más variables de control, se comunican estados de fabricación y controles de mantenimiento, desde puntos remotos.

Incluso en secciones que constan de unas pocas máquinas, la utilización de un bus de campo reduce el cableado y permite la supervisión y el control remoto.

Hay buses de campo que son controlados por A.P. y dispositivos auxiliares (derivadores, conectores, módulos de entradas y salidas), que tienen una facilidad de conexión que asombra al técnico más experimentado. Una tarjeta para un A.P. específico asume las tareas de elemento principal del bus, el resto de dispositivos, conectados con los accesorios adecuados, emiten y reciben las señales a través del bus de comunicación, actuando como secundarios. Todo muy sencillo, pero es más, en la práctica cotidiana resulta eficiente y fiable, integrando todo tipo de dispositivos, como convertidores de frecuencia, controladores de temperatura, etc.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Definir qué es un bus de campo y explicar sus aplicaciones. – Explicar las características fundamentales de un bus de campo. – Determinar la capacidad de integración de instrumentación inteligente con sistemas superiores de tratamiento de información. – Distinguir entre un bus de campo y una red industrial. – Explicar cuáles son las características del modelo OSI reducido que utilizan los buses de campo. – Describir la estructura que tiene un sistema basado en un bus de campo. – Explicar los distintos dispositivos que lo pueden configurar y los elementos de base que lo conforman.	8.1. Buses de campo: ubicación y diferencias con las redes industriales. 8.1.1. Ubicación y redes industriales. 8.1.2. Buses de campo y buses de dispositivos o entradas y salidas remotas. 8.1.3. Un poco de historia. 8.2. Ventajas e inconvenientes de los buses de campo. 8.2.1. Cableado. 8.2.2. Velocidad. 8.2.3. Respecto al lazo de corriente 4-20 mA. 8.2.4. Ahorro económico. 8.2.5. Estándar europeo para buses de campo. 8.3. Aspectos más relevantes de los buses de campo. Bus Profibus. 8.3.1. Características generales. 8.3.2. Modelo OSI de comunicación en buses de campo. 8.3.3. Medios de transmisión y topología de la red en Profibus-DP y FMS. 8.3.4. Modelos de coordinación de la información. 8.3.5. Resumen de funciones Profibus-DP. 8.3.6. Breve descripción de dispositivos Profibus-DP. 8.3.7. Orientaciones para aplicaciones específicas. 8.3.8. Ficheros de base de datos GSD para configuración de dispositivos Profibus-DP. 8.3.9. Profibus-FMS (Fieldbus Message Specification, descripción de mensajes en el bus de campo). 8.4. Bus de campo WorldFip.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
	8.5. Comparativa de buses de campo. 8.6. Configuración de una aplicación con Profibus-DP. 8.6.1. Instalación. 8.6.2. Configuración de estaciones participantes en bus Profibus-DP con HMS.

Capítulo 9. Procedimientos de comunicación con dispositivos industriales.

Introducción

En este capítulo realizaremos una exposición de orientaciones prácticas sobre comunicaciones con dispositivos industriales. En un libro de texto se pueden dejar muchos detalles sin presentar. Nosotros presentamos características, diagramas y posibles aplicaciones de las diferentes posibilidades de comunicación. Es el profesor, con los medios a su alcance, el que podrá utilizar este material e, incluso, superarlo.

Nuestra intención es recopilar los tipos de comunicación que se realizan con dispositivos industriales: enlaces 1 a 1 y multipunto; y en estos últimos, unas guías prácticas para la instalación y configuración de buses de campo.

Los tres apartados se han podido realizar gracias a la colaboración de la firma comercial Omron Electronics, S.A., que ha prestado su personal, componentes e instalaciones de la delegación Levante, C/ Guillem de Castro, n.º 8, 46001, Valencia. Sin dicha colaboración no habría sido posible esta exposición. Sirva como homenaje a todas las personas que trabajan en la misma, y en especial a nuestro buen amigo Marcos Jiménez, delegado.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Dotar al alumnado de procedimientos de trabajo para la identificación del tipo de comunicación que se precisa para cada aplicación.	9.1. Enlace punto a punto 1 a 1. 9.1.1. Protocolo host-link. 9.1.2. Vía radio. 9.2. Comunicaciones multipunto 1 a N: topología en bus.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Realización de la selección de dispositivos para la configuración de una instalación de bus de dispositivos o de campo. – Establecimiento del diagrama de conexiones de un sistema de comunicación industrial. – Selección de parámetros que hay que utilizar en la configuración del software y de los dispositivos de un bus de campo. – Realización de medidas con instrumentos de comunicaciones industriales.	9.2.1. Host-link 1 a n. 9.2.2. Reducción del cableado. 9.2.3. Bus de campo: Compobus/S. 9.2.4. Bus de campo: Compobus/D.

Capítulo 10. Desde C hasta C++.

Introducción

El lenguaje C es quizás uno de los lenguajes de programación más potentes y flexibles, esto le ha permitido mantenerse vivo durante muchos años sin apenas cambios significativos. El lenguaje C se remonta prácticamente a los inicios de la informática, fue una evolución del lenguaje ensamblador desarrollado sobre máquinas UNIX. Sin embargo, C también ha evolucionado para adaptarse a nuevas técnicas de programación, como la programación orientada a objetos y a los nuevos sistemas operativos (Windows).

El lenguaje de programación C++ es el heredero del lenguaje C y constituye una de las bases más sólidas para el desarrollo de programas bajo Windows.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Introducir el lenguaje C++. – Conocer la asignación dinámica en C++. – Conocer las nuevas características aportadas por C++.	10.1. Clasificación de los lenguajes de programación. 10.2. Notas sobre los ejemplos. 10.3. Notación húngara. 10.4. Ampliaciones en el lenguaje. 10.4.1. Extensiones de los ficheros. 10.4.2. Comentarios.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
	10.4.3. Nuevo tipo bool. 10.4.4. Entrada/salida. 10.4.5. Declaración de variables. 10.4.6. Operador de resolución de ámbito. 10.4.7. Referencias. 10.4.8. Operadores new y delete. 10.4.9. Conversiones de tipo. 10.5. Ampliaciones en las funciones. 10.5.1. Funciones. 10.5.2. Prototipos de funciones. 10.5.3. Clases de almacenamiento. 10.5.4. Funciones insertadas. 10.5.5. Funciones sin prototipo. 10.5.6. Paso de argumentos por valor o por referencia. 10.5.7. Modificadores const y volatile. 10.5.8. Funciones con argumentos por defecto. 10.5.9. Retorno de valores por valor, puntero o referencia. 10.5.10. Sobrecarga de funciones.

Capítulo 11. Clase y Objeto en C++.

Introducción

La diferencia fundamental entre un lenguaje orientado a objetos como C++ y un lenguaje estructurado como C son las clases y los objetos. Las clases permiten desarrollar aplicaciones orientadas al componente y son el fundamento de todas las herramientas de programación visual o RAD (desarrollo rápido de aplicaciones) para Windows, como por ejemplo C++ Builder, Delphi, Visual C++, Visual Basic, etc.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Conocer los fundamentos de la POO. – Conocer los conceptos asociados a las clases y objetos. – Desarrollar pequeños programas basados en la POO.	11.1. Principios de la POO. 11.1.1. Encapsulación. 11.1.2. Polimorfismo. 11.1.3. Herencia. 11.2. Desde las estructuras hacia las clases. 11.3. Clases y Objetos. 11.3.1. Concepto. 11.3.2. Declaración y definición de clases. 11.3.3. Punteros a objeto. 11.3.4. Constructores. 11.3.5. Destructores. 11.3.6. Asignación de objetos. 11.3.7. Paso de objetos a las funciones. 11.3.8. Objetos devueltos por funciones. 11.3.9. Arrays de objetos. 11.3.10. Arrays de punteros a objetos. 11.3.11. Funciones amigas. 11.4. Herencia. 11.5. Polimorfismo.

Capítulo 12. C++ en acción.

Introducción

En este capítulo se abordan una serie de ejemplos desarrollados bajo la perspectiva de la POO y que ayudarán a aplicar los conocimientos de los dos capítulos anteriores. Al mismo tiempo, las clases que se muestran aquí se utilizarán en los capítulos siguientes para la programación de las comunicaciones bajo DOS. No olvide además que la programación Windows que se verá en los temas finales se basa también en este modelo de programación.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Poner en práctica la POO. – Desarrollar clases útiles para próximos capítulos.	12.1. Cómo seguir los ejemplos. 12.2. Clase CMARCO. 12.3. Clase CLEDS. 12.4. Clase CDISPLAY. 12.5. Clase CVENTANA. 12.6. Clase CBIGNUM. 12.7. Clase CLIFO. 12.8. Colas FIFO. 12.9. Listas enlazadas.

Capítulo 13. Programación a bajo nivel desde DOS.

Introducción

En este capítulo se abordan algunos de los temas de programación de sistemas bajo MS-DOS que permiten avanzar en la comprensión de la programación a bajo nivel y el acceso a los puertos en el PC. Aquí se ponen en práctica los conceptos desarrollados en el capítulo "Acceso a los puertos de E/S en un PC". Lea y comprenda ese capítulo antes de empezar con éste.

Los ejercicios se han probado con el compilador Borland C++ 3.1 para DOS.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Revisar el acceso a nivel de bit. – Comprender el acceso a los puertos en un PC. – Programar interrupciones software. – Programar interrupciones hardware.	13.1. Operaciones a nivel de bit. 13.2. Acceso a bajo nivel desde C. 13.2.1. Acceso directo a los puertos. 13.2.2. Acceso mediante la API de la BIOS. 13.2.3. Acceso mediante la API del DOS. 13.3. Acceso a áreas reservadas de memoria. 13.4. Interrupciones software. 13.5. Interrupciones hardware. 13.6. Control de tiempos.

Capítulo 14. Programación del puerto paralelo.

Introducción

En este capítulo se muestran ejemplos de aplicación para la programación del puerto paralelo desde el MS-DOS. También se ponen en práctica aquí los conocimientos teóricos del puerto paralelo expuestos en la primera parte y las técnicas de programación orientada a objetos de los primeros capítulos de esta segunda parte.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Aplicar los conocimientos teóricos sobre el puerto paralelo. – Desarrollar una clase de acceso a E/S. – Desarrollar programas que acceden al puerto paralelo. – Desarrollar programas de comunicaciones sobre el puerto paralelo.	14.1. Cómo seguir los ejemplos. 14.2. Clases para acceso a puertos de E/S. 14.3. Monitorizar y controlar los registros de E/S. 14.4. Juego de luces sobre el puerto paralelo. 14.5. Microautómata. 14.6. Chat entre dos ordenadores.

Capítulo 15. Programación del puerto serie.

Introducción

En este capítulo se muestran ejemplos de aplicación para la programación del puerto serie desde el MS-DOS. Se ponen en práctica los conocimientos teóricos del puerto serie expuestos en la primera parte de este libro y las técnicas de programación orientada a objetos.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Aplicar los conocimientos teóricos sobre el puerto serie. – Desarrollar una clase para acceso al puerto serie desde DOS. – Desarrollar programas que accedan al puerto serie.	15.1. Cómo seguir los ejemplos. 15.2. Una clase para comunicaciones serie. 15.3. Repetidor de señal. 15.4. Chat entre dos ordenadores. 15.5. Dos contadores conectados. 15.6. Recibir datos por interrupciones.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Desarrollar programas de comunicaciones sobre el puerto serie.	15.7. El código de la clase CPortCom. 15.8. Protocolo con autómatas Omron. 15.9. Clase para comunicación con autómatas Omron.

Capítulo 16. C++ Builder.

Introducción

El entorno Windows es ya desde hace tiempo el entorno más habitual para desarrollar aplicaciones. Aunque los programas desarrollados bajo DOS pueden ser ejecutados en una ventana de Windows no pueden aprovechar muchas de las prestaciones y funciones que ofrece Windows.

El presente capítulo trata de introducir al lector en un moderno sistema de desarrollo del programa para Windows: el C++ Builder.

Los capítulos que restan abordan el C++ Builder desde una perspectiva orientada a los programas de comunicaciones, en ningún caso pretende ser una guía de referencia.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Conocer el entorno de desarrollo del C++ Builder. – Realizar pequeños programas basados en Windows.	16.1. Introducción. 16.2. El entorno de desarrollo. 16.3. Gestión de proyectos. 16.4. Manipulación de componentes. 16.5. La primera aplicación. 16.6. Salida de información. 16.7. Entrada de información. 16.8. Medir el tiempo. 16.9. Utilización de cuadros de diálogo estándar. 16.10. Manejo de más de un formulario. 16.11. La cola de mensajes. 16.12. Instalación de nuevos componentes.

Capítulo 17. Entrada/salida en C++ Builder.

Introducción

En este tema se aborda la problemática del acceso a bajo nivel a la E/S desde Windows. Se muestran distintas alternativas para trabajar con la E/S y se desarrollan algunos ejemplos prácticos.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Conocer el acceso a bajo nivel desde Windows. – Desarrollar programas de E/S a nivel de registros desde Windows.	17.1. Acceso a los puertos de E/S bajo Windows. 17.2. Leer y escribir en un puerto. 17.3. Monitorizar el puerto paralelo. 17.4. Acceso a puertos mediante controlador en formato DLL. 17.5. Acceso al sistema de ficheros. 17.6. Acceso a la API de Windows.

Capítulo 18. La API del puerto serie.

Introducción

En este capítulo se hace una introducción a la API WIN 32 para comunicaciones serie y se desarrolla una clase que permite transmitir y recibir a través del puerto serie. Esta clase puede ser utilizada por cualquier programa escrito en C++ sobre un sistema operativo WIN 95/98/NT/2000.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Conocer la API WIN 32 para acceso al puerto serie. – Desarrollar programas de comunicaciones series basados en Windows.	18.1. Funciones de la API WIN32 para comunicaciones serie. 18.2. Líneas de control de flujo. 18.3. Buffer del transmisor y del receptor. 18.4. Leer y escribir en el puerto. 18.5. Clase TWinSerCom para comunicaciones bajo Windows.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
	18.6. Un chat bajo Windows. 18.7. Enviar y recibir tramas. 18.8. Control del puerto mediante eventos. 18.9. Procesos y subprocesos. 18.10. Recepción en segundo plano.

Anexo. La red telefónica.

Introducción

En este anexo se hace una descripción de los diferentes elementos de los que se compone una red de telefonía y que nos permitirá tener una visión genérica sobre el funcionamiento y los componentes de las redes telefónicas.

OBJETIVOS	CONOCIMIENTOS (CONTENIDOS SOPORTE)
– Conocer el proceso de la comunicación telefónica. – Conocer los componentes funcionales de una red telefónica. – Comprender los términos relacionados con el tráfico telefónico. – Conocer los sistemas de telefonía. – Describir los componentes de un teléfono celular. – Conocer y describir las redes inteligentes.	A.1. Introducción. A.2. ¿Qué es la red telefónica conmutada? A.3. El proceso de la comunicación telefónica. A.4. Componentes funcionales de una red telefónica. A.5. Tráfico telefónico. A.6. Estructura de la red telefónica conmutada y jerarquía de centrales. A.7. Sistemas de telefonía. A.8. Equipos de datos y telefónicos. A.9. Red inteligente. A.10. Centralitas privadas PBX.

9. Actividades, cuestiones, problemas y prácticas propuestas

Las actividades, cuestiones, problemas y prácticas propuestas que se plantean en el libro son un modelo indicativo de lo que los profesores pueden plantear o proponer como aplicación o desarrollo de los temas tratados en cada capítulo, siendo el profesor el que mejor conoce las necesidades y los recursos de sus alumnos y por lo tanto el que debe elaborar y proponer las acciones más convenientes.

Capítulo 1

1. ¿Qué sentido tiene hablar de comunicación uno a muchos? Indica al menos una ventaja.
2. Indica las diferencias entre una estación y un medio de transmisión.
3. ¿Recuerdas el concepto de comunicación distribuida?
4. ¿Qué tipo de aplicaciones informáticas se encargan de la supervisión, control y adquisición de datos?
5. ¿Qué diferencias existen entre una estación principal y una secundaria?
6. Las funciones de un sistema de comunicación son...
7. Las diferencias entre los niveles de integración CIM y los niveles de comunicación industrial son:
8. Un datagrama es...
9. Las redes de área local se distinguen de las de área amplia en...
10. Suponiendo que dispones de ordenadores en red, ¿forman una red local? ¿Por qué? Intenta averiguar qué programa gestiona la red y en qué tipo de topología están conectados los dispositivos, y si existen subredes.
11. Solicita al profesor un libro o un texto donde encontrar otros ejemplos del modelo distribuido o descentralizado.
12. Disponemos de un sistema de comunicación industrial, como el ilustrado en el apartado sobre los niveles de comunicación industrial. En qué nivel o niveles de comunicación se realizará el intercambio entre:

    a) El valor de un bit de salida de un autómata programable y la salida correspondiente.

    b) El valor analógico entre un instrumento de control y un autómata programable.

    c) Entre un valor de salida de robot y la activación de una compuerta de una máquina.
13. Puedes solicitar información (por fax o Internet):
    - Normalización: relación de comités de AENOR.
    - ¿Dónde se pueden encontrar normas IEC o EN?

- En Internet existen numerosos documentos y foros sobre todas estas materias.
- Las firmas comerciales más implantadas disponen de páginas web. En ellas puedes encontrar abundante información sobre dispositivos de comunicaciones industriales.

14. Si miras el manual de uso del autómata programas de gama alta, disponible en el aula-taller, ¿qué posibilidades de comunicación permite: uno a uno o enlace punto a punto, bus, etc.?
15. Diferencia entre las funciones de un protocolo: control de flujo y empaquetado.
16. Un protocolo debe establecer algunas reglas semánticas, ¿por qué?
17. Entre los niveles del protocolo OSI, ¿qué se especifica en el nivel l, 2 y 3?
18. Entre la información técnica que recoge el profesorado del departamento, existen ofertas y publicidad de comunicaciones industriales. Investiga, colecciona recortes. Intenta identificar lo que has visto en este capítulo.
19. Si buscas una norma española debes descubrir, en primer lugar, el comité que tiene asignada su competencia. Este comité se identifica por un número y nombre. Averigua cuáles son los encargados de la automatización industrial, las telecomunicaciones y la compatibilidad electromagnética.

Capítulo 2

1. ¿Para qué se utilizan las tarjetas de interfaz de red? Indica alguna abreviatura.
2. ¿Qué campos de información debe contener un mensaje y por qué?
3. Una trama, ¿qué es?
4. Defina velocidad de transmisión con un ejemplo.
5. Describa el código ASCII. (Seguramente ya le es conocido por el módulo de Informática Industrial.)
6. Indica las diferencias entre una trama comando y una trama respuesta, en el ejemplo Host Link.
7. La diferencia entre frecuencia y fase de una señal es...
8. ¿Cuál es la longitud de onda de una señal de 1.000 Hz? ¿Y de 10 MHz?
9. La norma básica entre ancho de banda y velocidad de transmisión es...
10. ¿Cuál es la unidad adecuada para la medida de potencia en un medio de transmisión?
11. Capacidad de canal, ¿qué es?

12. ¿Qué medio de transmisión utilizaremos para comunicaciones de corta distancia, medio ancho de banda, elevada velocidad de transmisión, y alta inmunidad a las interferencias por los armónicos de la señal de la red eléctrica? ¿Por qué?
13. Inconvenientes de los medios de transmisión no guiados.
14. Principal ventaja de la comunicación serie frente a la paralela.
15. ¿Es cierta la siguiente afirmación: "la comunicación multipunto tiene que ser serie siempre"? Razona la respuesta.
16. ¿Por qué el cable coaxial y la fibra óptica monomodo tienen un valor despreciable en cuanto al parámetro de diafonía?
17. ¿Cuáles son las consideraciones sobre tiempo de propagación y tiempo de transmisión que se deben tener en cuenta en la comunicación en un enlace punto a punto?
18. ¿Qué es modulación? Tipos de modulación.
19. Describe la diferencia entre las siglas PCM y NRZ.
20. ¿Qué persigue el cableado estructurado?
21. ¿Qué técnicas se utilizan para enviar varias comunicaciones por un mismo medio de transmisión? Indícalas y descríbelas.
22. Diferencia fundamental entre banda base y banda ancha.
23. Módem. ¿En qué unidades se mide la velocidad y cuál es su utilidad?
24. ¿Cómo escribiría un código Hayes, en lenguaje C? Por ejemplo: descolgar, colgar, retardo de 2 segundos aproximadamente.

Capítulo 3

1. Describa la trama del protocolo Xmódem.
2. ¿Qué puede motivar la aparición de un error en la transmisión?
3. Diferencia entre CRC-16 y la comprobación de checksum.
4. Indique al menos un método para resolver los duplicados.
5. ¿Para qué se utilizan las tramas de control?
6. ¿Por qué en parada y espera sin errores no hacen falta números de secuencia?
7. ¿Qué diferencia hay entre parada y espera con y sin errores?
8. ¿Qué diferencia existe, desde el punto de vista funcional, al recibir la trama ACK 1 y NAK 1, en un protocolo en genérico?
9. Dando por conocidas las tramas de Xmódem, ¿qué diferencia existe entre una trama especial y una de datos, de las que envía el transmisor? Redacte algunos ejemplos.

10. En lenguaje C se puede componer una trama Xmódem. Un vector o array[x] con x entre 0 y 131, puede ser la trama de datos. Además *trama = 4;* representa a EOT, si trama se declaró *byte trama.* Con la lectura de datos de un fichero podemos obtener los 128 bytes de datos.
    10.1. Realice un programa para realizar el cálculo de checksum de un bloque de 128 bytes de datos de un fichero.
    10.2. Realice un programa para componer una trama Xmódem sobre la variable trama[131].
11. Realice una función en lenguaje C para montar una trama de comando del protocolo Host-Link. El texto serán 122 bytes de un fichero de texto almacenado en disco duro: fich.txt. El número de estación es num_esta.

Capítulo 4

1. ¿Puedes distinguir una red local de información y una red local industrial? ¿Cómo?
2. ¿Qué quiere decir principal/secundario?
3. ¿Qué diferencia hay entre el nivel de red y el nivel físico del sistema OSI?
4. ¿Qué problemas resuelven los métodos de paso de testigo y sondeo?
5. ¿Por qué se dividen o fragmentan los datos de las aplicaciones para ser transmitidos?
6. ¿Qué transformaciones sufren los datos al pasar por los niveles de protocolo en una misma estación?
7. Hágase una tabla en la que se diferencien las características que deben cumplir los repetidores, routers, concentradores y pasarelas.
8. Si tiene acceso a Internet, utilizando uno de los múltiples buscadores, recopile información técnica sobre:
    - Cables.
    - Conectores de red.
    - Tarjetas de interfaz Ethernet II.
    - Concentradores o hub.

Capítulo 5

1. Representar el mapa de la memoria direccionable de un micro que tenga 32 bits en el bus de direcciones y otro que tenga 20 bits. Escribir las direcciones de comienzo y final en hexadecimal.
2. Buscar información sobre los nemónicos de los micros de INTEL, después hacer una tabla resumen donde se clasifiquen por tipos y se vea un breve resumen de su uso.
3. Buscar información sobre la arquitectura interna de un micro de MOTOROLA (u otros fabricantes) y compararla con la arquitectura de INTEL.
4. Representar en un diagrama de flujo el acceso por sondeo a tres dispositivos de E/S. Hacer que uno de ellos tenga mayor prioridad que los otros dos asignándole más tiempo.
5. Decir cuáles son las ventajas que presenta el acceso a E/S por interrupciones frente al acceso por sondeo.
6. Suponiendo un tamaño de pila de 64 Kb, y suponiendo que cada vez que se ejecuta una subrutina se salva la dirección completa SEGMENTO:OFFSET, cuántas llamadas a subrutina pueden anidarse antes de que la pila se agote (STACK OVERFLOW).
7. Identificar en su PC las direcciones de E/S mediante el programa *Inicio-Panel de Control-Sistema* de los siguientes periféricos: teclado, puertos serie, puertos paralelo y PIC.
8. Utilizando el comando dump (D) del programa DEBUG, mostrar la memoria de vídeo en modo texto que empieza en B800:0000.
9. Utilizar el comando salida (O) del programa DEBUG para leer el registro IMR del PIC1 y del PIC2. A partir del dato leído, identificar las interrupciones que se encuentran habilitadas y las que no lo están.
10. Identificar la interrupción asociada al ratón de su PC, identificar la posición que ocupa en la tabla de vectores de interrupción. Utilizando el comando dump del programa DEBUG localizar la dirección de comienzo de la ISR asociada al ratón.
11. Utilizando el comando ensamblar (A) del programa DEBUG escribir un pequeño programa en la dirección CS:0100. Este programa será el siguiente:

    MOV AX, 01

    MOV BX, 03

    ADD AX, BX

    Ejecutar el programa anterior con el comando seguimiento/trace (T) y observar el contenido de los registros después de ejecutar cada instrucción, para indicar, a continuación, lo que hace.

Capítulo 6

1. Identificar la dirección base del puerto paralelo de su ordenador.
2. Con ayuda del comando DEBUG, comprobar si el puerto paralelo de un ordenador es bidireccional. Para ello lea y escriba un uno y un cero en el bit 5 del registro de control y compruebe que se registran ambos estados.
3. Comprobar en la pantalla de SETUP del PC las posibilidades de configuración del puerto paralelo.
4. Construir un cable null-módem paralelo y conectar dos PC utilizando el programa *Interlink* (INTERLNK.EXE e INTERSVR.EXE) del MS-DOS. Encontrarás información sobre estos programas en la ayuda del DOS con el comando HELP.
5. Utilizando el mismo cable, conectar dos PC utilizando el programa *Conexión directa por cable* del grupo *Accesorios.*
6. Construir alguno de los circuitos que se han propuesto a modo de proyecto.

Capítulo 7

1. Identificar el número de puertos serie, su dirección de E/S y el número de interrupción de un PC.
2. Construir un cable null-módem serie y conectar dos PC utilizando el programa INTERLNK.EXE/INTERSVR.EXE del MS-DOS. Encontrará información sobre estos programas en la ayuda del DOS con el comando HELP.
3. Utilizando el mismo cable, conectar dos PC mediante el programa *Conexión directa por cable* del grupo *Accesorios.*
4. Utilice el comando MODE de MS-DOS para configurar el puerto serie. Identifique las opciones de configuración que no pueden ser establecidas con este comando.
5. Conectar dos PC mediante un cable null-módem 2 y comprobar mediante el comando DEBUG que al manipular la RTS (bit MCR.1) de uno de los dos PC se puede ver su estado en el otro PC en la línea CTS (bit MSR.4).

Capítulo 8

1. Diferencie entre redes locales y buses de campo.
2. Ventajas e inconvenientes de los buses de campo.
3. ¿Cuáles son las características fundamentales de los buses de campo?
4. ¿En qué parte de la documentación de EN 50170 buscará lo relacionado con la detección de errores en los datos y la generación de tramas?

5. Indique las partes del modelo OSI que sigue un bus de campo genérico.
6. Describa brevemente los protocolos que soporta Profibus.
7. ¿Qué diferencias existen entre una estación principal y una secundaria?
8. ¿Qué es el modo Sync?
9. ¿Tipos de comunicación entre estaciones principales y entre estas últimas y las estaciones secundarias?
10. Realice un esquema de conexiones de un bus Profibus DP y FMS, diferenciando entre el cableado de bus de dispositivos y bus de campo:

    2 A.P. estaciones principales. 1 tarjeta Profibus para ordenador, tipo configurador.

    4 E/S de 8 entradas y 8 salidas digitales, asignadas dos a cada A.P.

    1 entrada analógica, 1 a 10 voltios, asignada al primer A.P.

    1 variador de velocidad asignado a un A.P.

    1 pasarela de bus de campo a bus de dispositivos. Asignada al PC.

    1 estación principal de bus de dispositivos.

    5 finales de carrera, del bus de dispositivos.

    2 contactores, asignados al bus de dispositivos.

    2 electroválvulas, asignadas al bus de dispositivos.
11. Conecte con las direcciones de Internet de Profibus, WorldFip. Puede descargar archivos y profundizar en las posibilidades de conexión y los dispositivos comerciales disponibles.

Capítulo 9

1. Realizar un procedimiento de comunicación con dispositivos industriales.
2. Identificar para cada aplicación el tipo de comunicación que se precisa.
3. Establecer el diagrama de conexión de un sistema de comunicación industrial.

Capítulo 10

1. Realizar un programa que multiplique dos números enteros creados de forma dinámica mediante el operador new().
2. Realizar una función CadDin() que reciba como argumento un entero con el tamaño de una cadena. La función creará de forma dinámica una cadena con el tamaño indicado y a continuación será leída mediante el teclado.

3. Realizar una función Codead() que reciba como parámetro de entrada un puntero a una cadena; la función se encargará de codificar la cadena cambiando cada carácter por el siguiente dentro de la tabla ASCII. Hacer un programa main() que utilice la función anterior.
4. Repetir el ejercicio anterior sustituyendo el puntero por una referencia.
5. Realizar una función UnCodCad() que reciba una referencia a una cadena codificada mediante Codead() y la decodifique.
6. Realizar una función CalFac() que calcule el factorial de un número entero. Realizar un programa que llame a la función anterior cinco veces; obsérvese el tamaño del ejecutable en función de si se declara inline o no.
7. Realizar una función que reciba como argumentos de entrada dos enteros y retorne su suma y su resta mediante otros dos argumentos de entrada pasados por referencia.
8. Realizar una función RndDoble() que retorne dos números aleatorios mediante el uso de dos argumentos de entrada pasados por referencia.
9. Realizar una función CodCad2() que permita codificar y decodificar una cadena de forma similar a las funciones Codead() y UnCodCad(). La función tendrá dos argumentos de entrada, una referencia a la cadena y un valor lógico indicando si se desea codificar (true) o decodificar (false). El segundo argumento por defecto será true.
10. Realizar una función ByteToBits() que reciba como parámetro de entrada un byte y lo muestre en binario mediante unos y ceros. Cada grupo de cuatro bits se separará por un punto. Por ejemplo ByteToBits(0x41) mostrará 0100.0001.
11. Modificar la función anterior para que además del byte reciba un carácter indicando el tipo de separador que se desea utilizar cada cuatro bits. Hacer que este segundo parámetro sea por defecto un punto (.).
12. Realizar una función sobrecargada ToBits() que permita mostrar el código binario de un byte o de un WORD. Esta función dispondrá de dos versiones, una para mostrar en binario un byte y otra para mostrar en binario un WORD.

Capítulo 11

1. Crear una clase *CNumRO* que implemente una variable de sólo lectura. La clase poseerá un constructor para inicializar el valor y un método que permitirá leerla.
2. Crear una clase *CLibro* que contenga los siguientes datos: título, autor, año de edición, estado (disponible, prestado, reservado) y los métodos *SetTit()/GetTit(), SetAut()/GetAut(),* etc.
3. Crear una clase *CFecha* que contenga tres miembros privados: día, mes y año. Implementar dos constructores, uno que inicialice los tres campos a la fecha del día y

otro que les asigne un valor. Escribir un método que permita incrementar la fecha en un día. Escribir una función miembro que muestre la fecha en el formato DD/MM/AAAA. Crear un programa principal que maneje la clase.

4. Crear una clase *CNumCom* que contenga dos datos miembro: la parte real y la parte imaginaria. Implementar un constructor que inicialice el número complejo. Escribir métodos para leer y escribir la parte real, la parte imaginaria, el módulo y la fase.
5. Crear un programa principal que declare dos números complejos mediante la clase *CNumCom* y pida por teclado sus valores; calcular y mostrar su suma y producto.
6. Crear una clase *CBiestableJK* que encapsule el comportamiento de un biestable J-K. Utilizar una variable lógica para guardar el valor memorizado. Escribir métodos para actualizar las entradas: *SetJ(), SetK(), ResetJ(), ResetK(), SetJK().* Escribir además métodos para leer la salida directa y negada *GetQ(), GetNQ().* En la tabla siguiente se muestra la tabla de verdad de un biestable JK: Qn+ 1 se refiere al estado siguiente y Qn al estado actual.

Tabla de verdad biestable J-K			
J	**K**	**Qn + 1**	**/Qn + 1**
0	**0**	**Qn**	**/Qn**
0	**1**	**0**	**1**
1	**0**	**1**	**0**
1	**1**	**/Qn**	**Qn**

Crear un programa principal que declare un objeto de la clase *CBiestableJK* y muestre en pantalla el valor memorizado. Hacer que funcionen las teclas J y K para manipular las entradas del biestable. Mostrar continuamente el valor memorizado en la pantalla.

7. Crear un programa principal que declare dos objetos de la clase *CBiestableJK.* Uno de los objetos será manipulado mediante las teclas J y K, el otro tendrá en todo momento el valor invertido del primero.
8. Desarrollar una clase *CPublicacion* que contenga el precio y el título. Escribir tres constructores para esta clase: uno sin argumentos, otro con el título y otro con el título y el precio. Escribir además métodos para leer y escribir esas dos propiedades.
9. Desarrollar una clase *CPubLibro* que herede de *CPublicacion* y que añada las propiedades de número de páginas y año de publicación. Escribir un constructor que reciba los cuatro argumentos necesarios para inicializar todas las propiedades del objeto. Escribir métodos para leer y escribir las propiedades añadidas en esta clase.

10. Desarrollar una clase *CPubDisco* que herede de *CPublicacion* y que añada la propiedad de número de minutos de duración. Escribir un constructor que reciba los tres argumentos necesarios para inicializar todas las propiedades del objeto. Escribir métodos para leer y escribir las propiedades añadidas en esta clase.
11. Crear un programa principal para manejar una biblioteca con un máximo de 20 objetos libro organizados en un array de objetos. Desarrollar un menú para añadir nuevos objetos al array y para mostrar el contenido actual del array.
12. Crear una clase *CBiestableT* que herede de la clase *CBiestableJK* y que encapsule el comportamiento de un biestable T. Este biestable se construye uniendo las dos entradas del biestable JK y responde por tanto a la siguiente tabla de verdad.

Tabla de verdad biestable T		
T	**Qn + 1**	**/Qn + 1**
0	**Qn**	**/Qn**
1	**/Qn**	**Qn**

13. Escribir un programa principal que declare un objeto de la clase anterior y se pueda cambiar su contenido mediante la pulsación de la tecla T.

Capítulo 12

1. Utilizar la clase *CVentana* para dibujar en la pantalla 3 ventanas y mostrar caracteres aleatorios y alfanuméricos en todas ellas de forma cíclica.
2. Utilizar la clase *CLeds* para crear un array de objetos de 16 LEDS, distribuir estos LEDS por la pantalla en dos filas. Mediante la tecla 1 se pondrán todos a uno y mediante la tecla 0 se pondrán todos a cero.
3. Utilizar la clase *CDisplay* para construir un efecto luminoso como el que se describe a continuación. Inicialmente el bit 0 se encuentra activado y los demás apagados, en la siguiente fase sólo se activa el bit 1 y así sucesivamente hasta que se active el bit 7. Al llegar al extremo izquierdo el bit se desplaza hacia la derecha de forma similar. Hacer que con la tecla "+" el efecto se ejecute más rápidamente y con la tecla "-" el efecto se ejecute más lentamente.
4. Usar la clase *CVentana* para que los caracteres escritos en un objeto ventana se muestren en formato binario mediante un objeto de la clase *CDisplay*.
5. Escribir un programa que muestre la hora real del ordenador en formato HH:MM:SS mediante seis objetos de la clase *CBigNum*. Decorar el reloj con un objeto de la clase *CMarco*.

6. Desarrollar un programa que maneje tres colas LIFO. Una de las colas se establecerá como cola actual mediante una opción de menú, y será sobre la que se realicen las acciones de añadir y quitar elementos. Se podrán añadir y quitar elementos de la cola actual mediante dos opciones del menú programadas al efecto. Se mostrará el contenido de las colas en tres ventanas de la pantalla.
7. Escribir una función que reciba como parámetros de entrada dos punteros a dos objetos de la clase CLIFO. La función añadirá el contenido de la primera cola a la segunda.
8. Desarrollar un programa que maneje dos colas FIFO que se verán continuamente en dos ventanas de la pantalla. Mediante un menú se permitirá añadir y quitar elementos de la primera cola. Los elementos que sean extraídos de la primera cola se añadirán a la segunda.
9. Crear una nueva clase *CNodoCad* que modifique la clase *CNodo* para asociar en cada nodo una cadena de hasta 80 caracteres.
10. Crear una lista enlazada con la clase *CLista* y la clase *GnodoCad*. Mostrar un menú para añadir nodos a la lista. Mostrar la lista en una ventana.
11. En la clase *CLeds* añadir un método SetTam() que permita fijar el tamaño del LED. Crear un programa principal que muestre 5 LEDS de distinto tamaño.
12. Desarrollar una nueva clase *CPanel* que conste de 98 LEDS distribuidos en 7 filas y 14 columnas. Cada LED tendrá un tamaño de 3 filas y 5 columnas. El tamaño total del panel será de 21 filas y 70 columnas. La clase dispondrá de métodos SetCero() y SetUno() para poner todo el panel apagado o encendido.
13. Desarrollar una clase *CBarra* que implemente una barra de progreso. La clase dispondrá de una propiedad iFil con la fila donde se mostrará, otra propiedad iMax con el máximo correspondiente al 100% y otra propiedad iVal con el valor actual. La clase deberá mostrar en todo momento una barra que represente el valor iVal sobre iMax. Crear un programa que cree dos objetos de la clase anterior.
14. Realizar una nueva clase *CTocken* que encapsule un carácter en movimiento aleatorio. Este carácter se puede mover en cualquiera de las ocho direcciones posibles: horizontal, vertical y diagonal, y en cualquier sentido: arriba o abajo. La clase dispondrá de dos propiedades iFilAct y iColAct que contienen la posición actual del tocken, cDat contiene el carácter que se muestra, cCol contiene el color y otra propiedad, iDir contiene la dirección actual del movimiento del tocken. Cuando el tocken alcanza alguno de los límites de la pantalla toma una nueva dirección de forma aleatoria. En el constructor se especificará el código ASCII que se desea, un método Avanza() hace que se mueva una posición. Por último, otro método SigDir() permite obtener la siguiente dirección que tomará el objeto. Para hacer esto, utilice la función random().

15. Realizar un programa que cree cinco objetos de la clase anterior y se muevan por la pantalla libremente.
16. Realizar una clase *CVentanaTocken* que herede de la clase *CVentana* y que contenga una propiedad que sea un objeto de la clase *CTocken*. Un método Avanza() hace que se mueva el tocken que contiene. Hacer que el tocken no salga de los límites de la ventana; cuando se alcanza uno de los cuatro lados se busca una nueva dirección aleatoria mediante SigDir().
17. Realizar un programa que construya cuatro ventanas de la clase anterior y haga que se muevan los objetos tocken dentro de cada una de las ventanas.
18. Para valorar el esfuerzo que supondría, repetir el ejercicio anterior sin utilizar POO, utilizando programación estructurada.

Capítulo 13

1. Diseñar una función ByteToBits() que reciba como argumento un byte y lo muestre en binario a base de ceros y de unos. Realizar un programa principal que pida un carácter por teclado y lo muestre a continuación en binario.
2. Realizar un programa que pida un número entero de 16 bits e intercambie los 8 bits menos significativos por los 8 bits más significativos.
3. Crear una función sobrecargada ToBits() que permita mostrar en binario cualquiera de los tres tipos básicos: BYTE, WORD o DWORD.
4. Realizar una función que permita mostrar y ocultar el puntero del ratón utilizando la interrupción software 33.
5. Añadir un nuevo método VenScroll() a la clase Ventana que permita hacer el scroll de la ventana. Cuando se llene la ventana, en lugar de borrar su contenido se hará un scroll de los datos contenidos en la ventana. Los nuevos datos serán añadidos en la última fila.
6. Hacer un programa que declare una ventana para salida de mensajes. Comprobar que cuando la ventana se llena se produce el scroll programado en el ejercicio anterior.
7. Programar la interrupción del reloj para que cada 10 segundos se borre la pantalla utilizando acceso directo a memoria de vídeo. Hacer un programa principal que deje escribir en la pantalla hasta que se pulse la tecla Q.
8. Realizar un programa que pida un número entero y calcule su cuadrado. Al mismo tiempo programar la interrupción de reloj para tener en la primera fila el carácter 0x01 desplazándose hacia la derecha cíclicamente. Utilice el acceso directo a la memoria de vídeo para hacer el efecto de desplazamiento.

9. Realizar una función CadToVid() que utilizando el acceso directo a memoria escriba una cadena en la pantalla. La función recibirá tres parámetros: dos enteros que indiquen la fila y columna, y la cadena que se mostrará.
10. Desarrollar una clase *CVidDir* para encapsular el acceso directo a memoria de vídeo. La clase dispondrá de los métodos BorraPan() y MostrarCad().
11. Realizar un programa que al ejecutarse lea la memoria de vídeo y borre la pantalla. Al finalizar el programa se recuperará la pantalla original, pero invirtiendo el orden de las líneas.
12. Investigar en las interrupciones software y hacer una relación de los servicios disponibles a nivel de la BIOS para trabajar con los puertos serie. Hacer lo mismo con los servicios disponibles a nivel del DOS.

Capítulo 14

1. Crear un programa que realice un efecto luminoso sobre el registro de datos del puerto paralelo. Este efecto consistirá en el llenado por la derecha de los ocho bits. Inicialmente todos los bits estarán a cero, un uno llegará por la derecha y se colocará en el bit 7, después llegará otro uno por la derecha y se colocará en la posición 6, etc. Monitorizar el registro de datos en la pantalla y sobre una placa con ocho LEDS.
2. Utilizando un cable nible-módem y un pulsador, realizar un programa que reciba interrupciones por el puerto paralelo y las cuente. La pantalla dispondrá de dos zonas. En la parte superior un reloj digital (ya propuesto como ejercicio en capítulos anteriores) y en la parte inferior un contador con dos dígitos de la clase *CBig-Num.* Por un lado se manejará el reloj, y simultáneamente se atenderán las peticiones de interrupción procedentes del puerto paralelo. Cada vez que se active el pulsador se producirá un pitido e incrementará el contador de interrupciones.
3. Utilizando un cable byte-mode leer mediante un puerto paralelo bidireccional ocho entradas conectadas a una placa con ocho interruptores. Una entrada adicional en S6 se utilizará a modo de STROBE para validar la información presente en los ocho pulsadores. Mostrar los bytes recibidos en una ventana.
4. Basándose en el ejercicio anterior, conectar ocho salidas de un autómata más una de STROBE a un ordenador con un puerto paralelo bidireccional. Realizar un programa en el autómata que cuente de 0x00 a 0xFF. Dicha cuenta se escribirá en las salidas del autómata para que sean transmitidas al ordenador. Además de mostrar la información, se tendrá opción a guardarla en un fichero.
5. Montar un prototipo con 3 entradas por pulsadores y 2 salidas mediante LED. Desarrollar las dos funciones lógicas que se muestran a continuación:

F1 = /A + B * C

F2 = A * (/B) + (/A) * B * C

6. Desarrollar una clase para la transmisión ASCII mediante un cable paralelo byte-mode utilizando técnicas de sondeo. La transmisión se realizará en un solo sentido en modo símplex. El transmisor utiliza D0-D7 para enviar, el receptor utiliza D0-D7 para recibir. Obsérvese que el receptor debe tener un puerto paralelo bidireccional. Las líneas C0 y S6 se utilizarán para controlar la transmisión.
7. Repetir el ejercicio anterior, pero utilizando en el receptor una función ISR para recepción por interrupciones.

Capítulo 15

1. Utilizando la clase de comunicaciones, realizar un programa que transfiera ficheros en modo ASCII (sin realizar entramado). En el transmisor se elegirá un fichero y los datos que se vayan enviando se mostrarán en una ventana. En el receptor los datos leídos del puerto se mostrarán en una ventana.
2. Añadir un nuevo método TxCadena() a la clase *CPortCom*. Este método se encarga de transmitir todos los caracteres de la cadena que recibe como parámetro de entrada.
3. Probar el ejemplo de los contadores sincronizados a distintas velocidades de transmisión. Obsérvese lo que ocurre a velocidades de transmisión elevadas.
4. Realizar un programa que permita monitorizar cualquiera de los registros de E/S de un puerto serie. Los registros se mostrarán como un objeto de la clase *CDisplay*.
5. Utilizando la clase de comunicaciones, desarrollar un programa que envíe la orden ATZ a un módem y muestre su respuesta. Dividir la pantalla en dos ventanas, una para los datos enviados y otra para los datos recibidos.
6. Desarrollar un programa que divida la pantalla en dos partes, en la parte superior se mostrará un reloj y en la parte inferior se mostrará una ventana con los datos que se reciban por el puerto serie. Utilizando la clase de comunicaciones serie hacer que los datos recibidos se lean mediante interrupciones. Utilizar un cable null-módem y un pequeño programa que transmita los caracteres que tecleemos en el ordenador transmisor.
7. Utilizando la interrupción del reloj, mediante una función ISR hacer que se envíe un carácter cada 2 segundos al puerto serie. Esta transmisión se realizará independientemente del programa principal. Mediante un cable null-módem y otro ordenador, comprobar que los datos se reciben correctamente.

8. Realizar un programa que lea bytes del puerto paralelo y los envíe a través del puerto serie. El programa leerá la información presente en el registro de estado y la enviará a través del puerto serie a otro ordenador que la recibirá y la mostrará con un objeto de la clase *CDisplay*. Comprobar el funcionamiento apagando y encendiendo la impresora.
9. Añadir dos métodos a la clase de *CPortCom* para habilitar la interrupción producida por un cambio en la línea CTS. Los métodos se denominarán EnableIRQCTS() y DisableIRQCTS(). Estos métodos programarán el bit correspondiente del registro IER del puerto respetando el resto de bits.
10. Desarrollar un programa principal que muestre un reloj en la parte superior de la pantalla; en la parte inferior y simultáneamente se mostrará un objeto de la clase *CLed* indicando el estado de la línea de entrada CTS. Para probar el funcionamiento del programa usar un cable null-módem y en el otro ordenador activar y desactivar la línea RTS.
11. Añadir dos métodos a la clase de *CPortCom* para habilitar la interrupción producida por la detección de un error. Los métodos se denominarán EnableIRQError() y DisableIRQError(). Estos métodos programarán el bit correspondiente del registro IER del puerto respetando el resto de bits.
12. Desarrollar un programa principal que muestre un reloj en la parte superior de la pantalla, simultáneamente se mostrarán los bytes recibidos en una ventana y los errores que se produzcan en otra. Programar la ISR para que determine la causa de interrupción y opere en consecuencia. Probar el programa con otro ordenador conectado a través de un cable null-módem y que transmita sin parar una información aleatoria.
13. Desarrollar una clave CXModem para construir tramas XMODEM.
14. Enviar un fichero de un PC a otro utilizando la clave anterior.

Capítulo 16

1. Realizar un programa que muestre en la pantalla un objeto *TMemo*. Al pulsar un botón se añadirá un mensaje al objeto. Con otro botón se borrará el texto.
2. Desarrollar un programa que muestre en la pantalla un objeto *TComboBox*. Cada vez que se pulse un botón se añadirá una nueva opción al objeto.
3. Realizar un programa que muestre 3 barras de progreso en la pantalla. La primera se moverá de 0 a 10, la segunda de 0 a 100 y la tercera de 0 a 1000. El incremento de la posición en todas ellas se realizará cada segundo mediante un objeto *TTimer*.
4. Implementar un programa que muestre un objeto *TImage* en la pantalla. Al pulsar un botón se mostrará un fichero BMP en la imagen.

5. Realizar un programa que contenga un objeto *TMainMenu*. Incluir cuatro opciones y hacer que se muestre un mensaje cada vez que se pulsa una de ellas.
6. Desarrollar un programa que muestre en pantalla un formulario con dos objetos *Tlabel*. Al pulsar un botón se intercambiará el texto de ambos.
7. Realizar un programa que muestre un objeto *TShape* en la pantalla. El tamaño de este objeto estará gobernado por otro objeto *TTrackBar*.

Capítulo 17

1. Utilizando acceso directo a los registros, hacer un programa que monitorice los registros que configuran la SDU en un puerto serie.
2. Modificar las *CPort* desarrolladas en la parte de MS-DOS para acceso a los registros del puerto paralelo y del puerto serie para que sean operativas en programas basados en Windows. Sustituir las llamadas inportb() / outportb() del DOS por alguna de las técnicas que se han explicado en este capítulo.
3. Desarrollar una clase *CPortLptWin* basándose en su equivalente en DOS donde se sustituyan las funciones de entrada y salida de puertos por las funciones de la DLPORTIO.DLL
4. Buscar en Internet otras librerías de acceso a puertos para *C++ Builder*, y probarlas con algún ejercicio sencillo.
5. Algunas librerías para *C++ Builder* permiten el control de las interrupciones hardware desde Windows (por ejemplo la librería TVicHW32). Buscar alguna de estas librerías en Internet y probar su funcionamiento con interrupciones hardware.
6. Buscar información en la ayuda de la API de Windows de las funciones *GetComputerName()* y *GetVersionEx(), GetWindowsDirectory(), GetDiskFreeSpace(),* documentarlas y hacer un pequeño programa que las use.

Capítulo 18

Teclear la clase *TWinSerCom* y hacer un pequeño programa que la use para enviar y recibir un carácter.

Los ejercicios planteados en el capítulo 6 con la clase de MS-DOS *CPortCom* reescribirlos en WINDOWS utilizando la clase *TWinSerCom*.

Teclear el ejemplo del MODEM y verificar las respuestas para los comandos HAYES más habituales.

Desarrollar un programa que reciba datos del puerto serie mediante eventos. Los datos recibidos se almacenarán en un fichero. Si se produce un evento de error se mostrará un mensaje y se cancelará la recepción de datos.

Anexo

Describir las bases en las que se fundamenta una red telefónica conmutada.

Describir los componentes de una red de telefonía.

Definir los términos o conceptos relacionados con el tráfico telefónico.

Describir al menos 6 servicios que ofrecen las redes inteligentes.

Buscar 10 direcciones de Internet relacionadas con las redes de telefonía conmutada.

10. Material didáctico (material y equipos didácticos)

En primer lugar debemos considerar el libro de Comunicaciones Industriales como el primer material didáctico con el que cuenta el profesor y el alumno para el aprendizaje, ya que el módulo es ya de por sí complicado para la localización y utilización de material didáctico que nos ayude a desarrollar las clases.

El libro se ha diseñado pensando en ello y se ha procurado ilustrar profusamente incluyéndose en él 274 figuras, esquemas y planos, 135 tablas y cuadros, etc., que facilitan la localización y comprensión de las diferentes partes de las máquinas eléctricas.

Desde el punto de vista práctico el material didáctico de apoyo mas idóneo para impartir las clases es:

- Catálogos de fabricantes.
- Material gráfico o soportes informáticos facilitado por casas comerciales.
- Reglamento eléctrotécnico para baja tensión.
- Vídeos y diapositivas adquiridas a empresas.
- Vídeos y material gráfico realizado por parte de alumnos o profesores, etc.
- Equipo básico del MEC para el desarrollo del módulo.

Los materiales que pueden ser de más utilidad para las necesidades específicas de las prácticas docentes de este módulo en el taller, y que vienen descritas en el libro del Ministerio sobre los Ciclos Formativos de Formación Profesional, en el apartado Dominio Profesional (equipos, materiales, información, procesos, etc.), son los que después intervienen en los sistemas productivos, por lo que los reproducimos a continuación dada su importancia:

Medios de producción o tratamiento de la información

Ordenadores y periféricos. "Software" de base (sistemas operativos) de los sistemas informáticos. "Software" de diseño asistido por ordenador: CAD eléctrico. "Software" para elaboración de documentación técnica. "Software" para sistemas de adquisición de datos. "Software" para diagnóstico para equipos y sistemas informáticos de comunicación industrial. Herramientas manuales para trabajos eléctricos y mecánicos (alicates, destornilladores, pelacables, soldador). Instrumentos de medida y verificación eléctrica (polímetro, osciloscopio). Instrumentos de medida para sistemas de telecomunicación industrial (analizadores de datos, protocolos y de red, reflectómetro para cables).

Materiales y productos intermedios

Cuadernos de carga de procesos automáticos que se van a informatizar y comunicar. Croquis, esquemas y planos de las configuraciones de las soluciones técnicas concebidas. Cálculos. Listas de materiales. Tarjetas, equipos y "software" de comunicación y redes locales industriales. Equipos y dispositivos de medida industriales. Tarjetas, equipos y "software" para sistemas de adquisición de datos. Elementos soporte de transmisión (cables y fibra óptica) y de interconexión (terminales y conectores).

Principales resultados del trabajo

Productos y/o servicios: Documentación de anteproyectos técnico-económicos de equipos y sistemas informáticos y de comunicación industrial. Documentación de proyectos de equipos y sistemas informáticos y de comunicación industrial. Informes de verificación, puesta en servicio y mantenimiento de equipos y sistemas informáticos y de comunicación industrial.

Procesos, métodos y procedimientos

Procedimientos de dibujo asistido por ordenador. Técnicas de programación informática en lenguajes de alto nivel. Procedimientos de medida y análisis de protocolos y datos en

redes de área local. Procedimientos de diagnóstico de averías en sistemas informáticos y de comunicaciones mediante la utilización de herramientas informáticas.

Información (naturaleza, tipo y soportes)

Especificaciones de proyectos de equipos y sistemas informáticos y de comunicaciones industriales. Diagramas topológicos de redes locales de ordenadores y de autómatas. Planos y esquemas de conexionados de equipos y dispositivos informáticos y de comunicación industrial. Normativa sobre protocolos y estándares de comunicación industrial. Catálogos especializados de materiales y equipos informáticos y de comunicación industrial (en papel o en soporte informático –bases de datos específicas–). Bases de datos de ingeniería (histórico de soluciones). Normas de seguridad de personas y equipos.

11. Material pedagógico de apoyo para la impartición del módulo

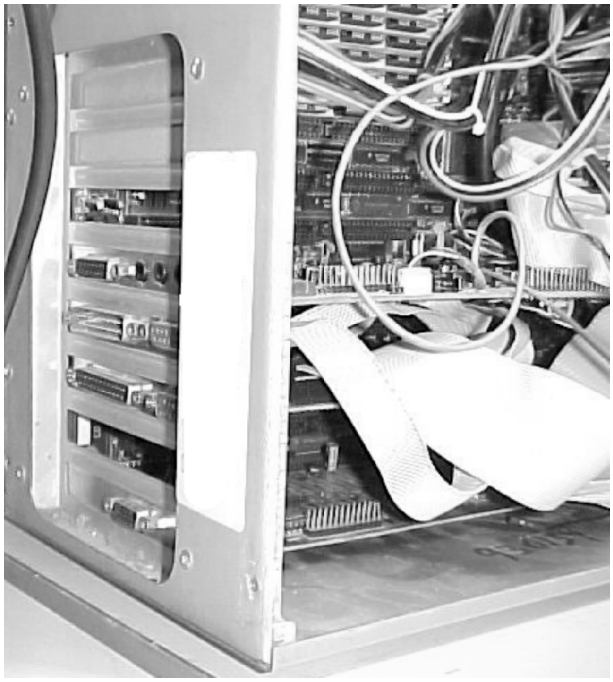

Figura 4.32. Ordenador, parte posterior.

Figura 4.33. Ordenador, puertos de comunicaciones y red.

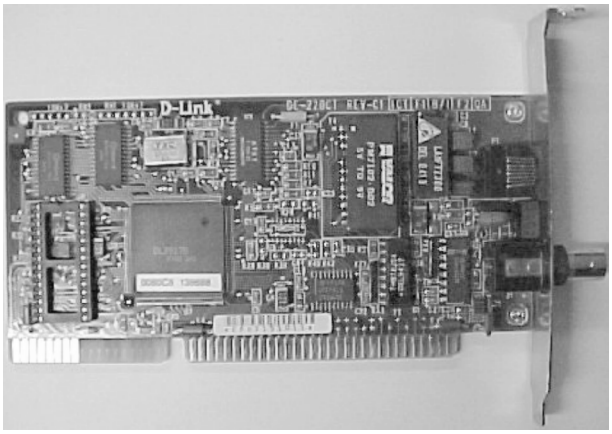

Figura 4.34. Tarjeta de red Ethernet II.

Figura 4.35. Cable de pares trenzados con terminales RJ-45.

Figura 4.36. Trasera de un hub, concentrador. Permite el aislamiento de colisiones al resto de la red y la difusión entre las estaciones conectadas.

Figura 4.37. Parte delantera de un hub, concentrador.

Figura 4.38. T para cable coaxial y T con terminador de línea, que se utiliza en redes Ethernet 10 Mbps, tienden a caer en desuso.

Figura 4.39. **Módem externo. Vistas anterior y posterior.**

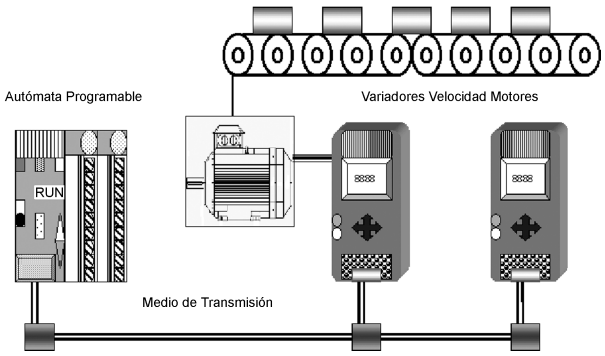

Figura 1.6. **Autómata programable comunicado con dos variadores de velocidad.**

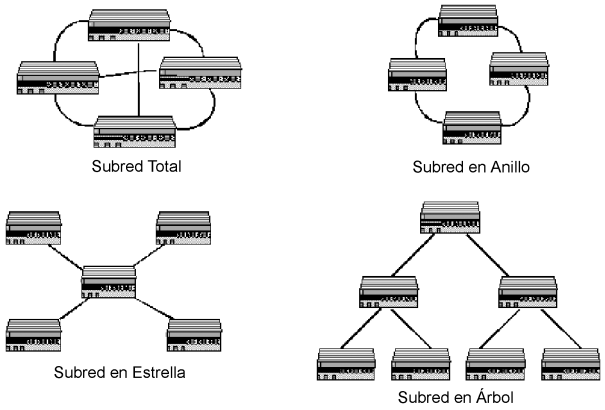

Figura 1.12. **Topologías tipo de subredes punto a punto.**

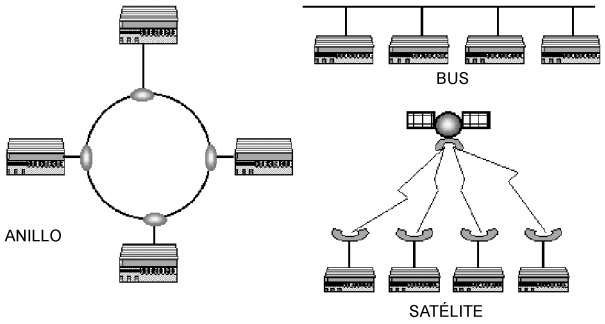

Figura 1.13. **Topologías tipo de subredes de difusión.**

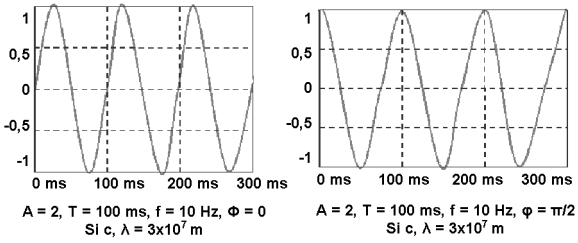

Figura 2.9. **Características de una señal analógica sinusoidal. La única diferencia se encuentra en la fase.**

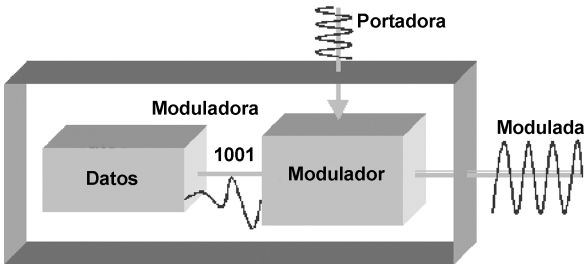

Figura 2.25. **Obtención de la señal modulada, que se transmite al medio, a partir de la señal portadora y las señales de los datos a transmitir: señal moduladora (sean analógicos o digitales). En la modulación se trasladan las frecuencias.**

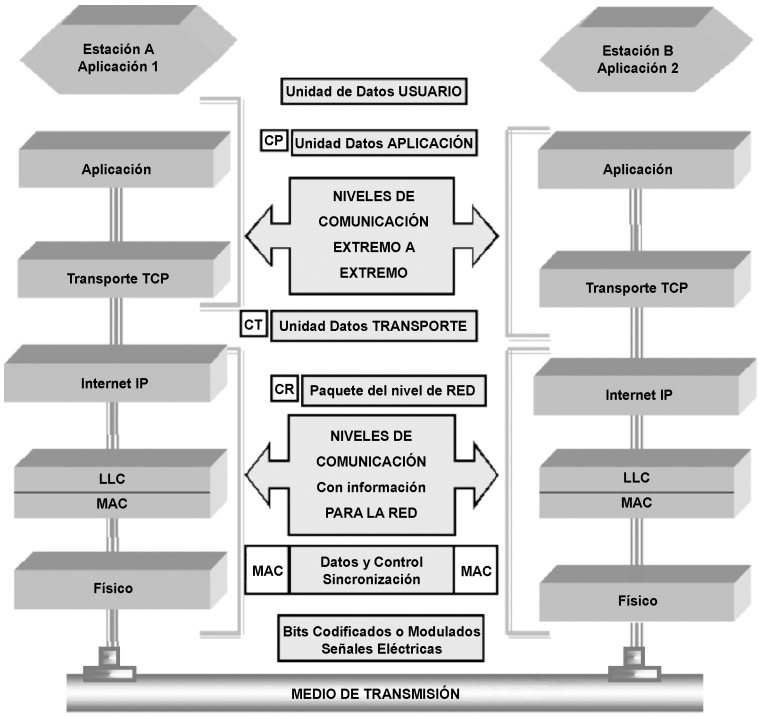

Figura 4.3. Modelo OSI en las redes locales.

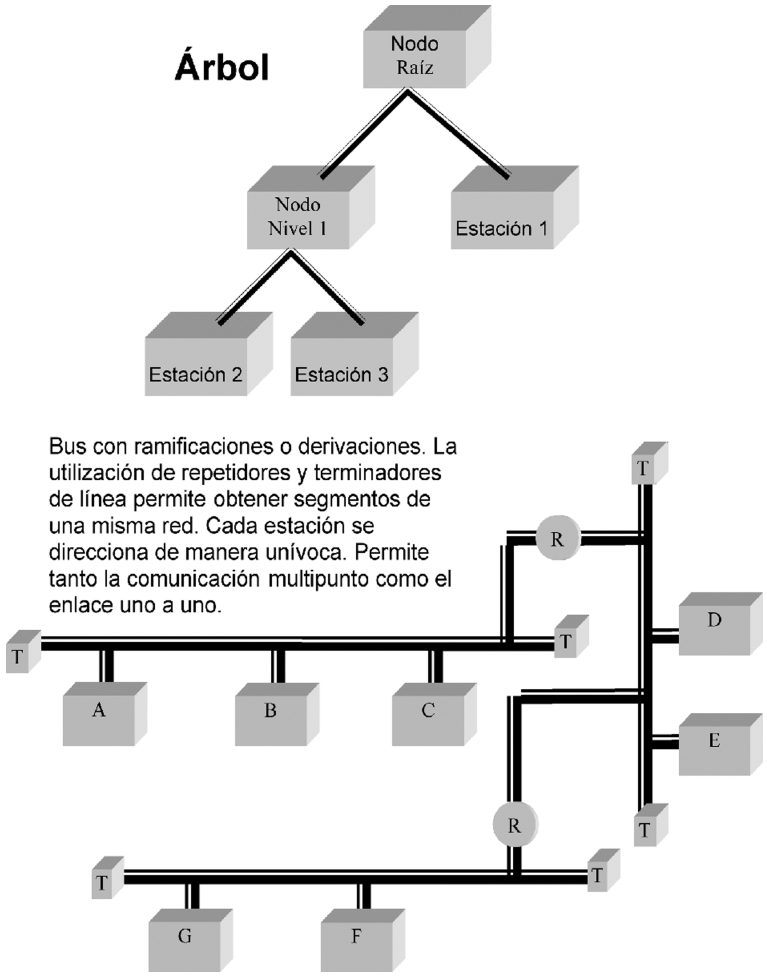

Figura 4.5. Superior: Topología en árbol, generalización de la topología bus. La parte inferior representa un bus lineal, con distintos segmentos.

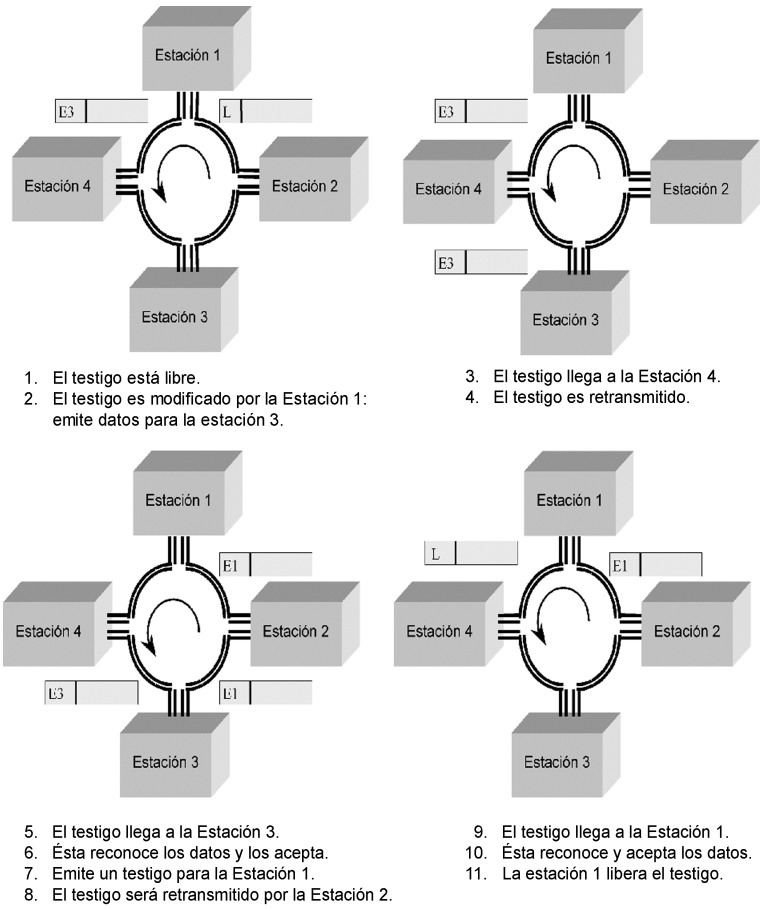

Figura 4.9. Token ring. El testigo circula y vuelve a la estación de la que salió.

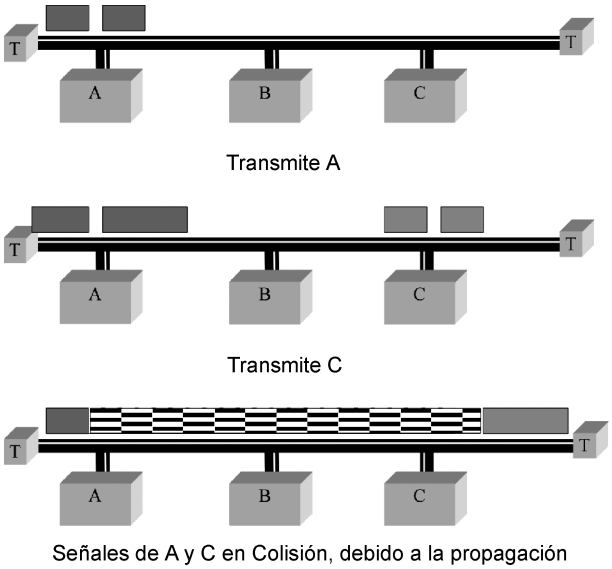

Figura 4.10. Colisiones en una red con topología en bus.

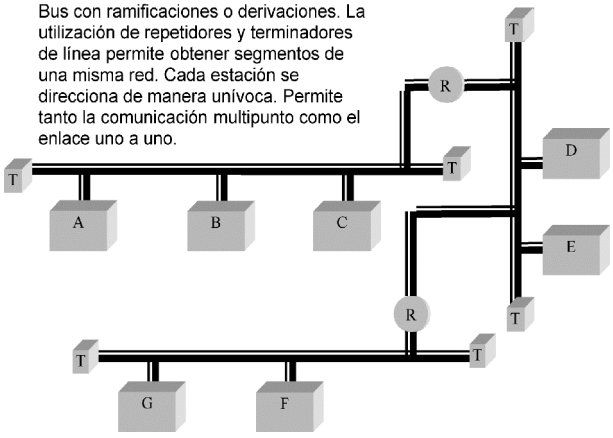

Figura 4.11. Bus lineal con distintos segmentos.

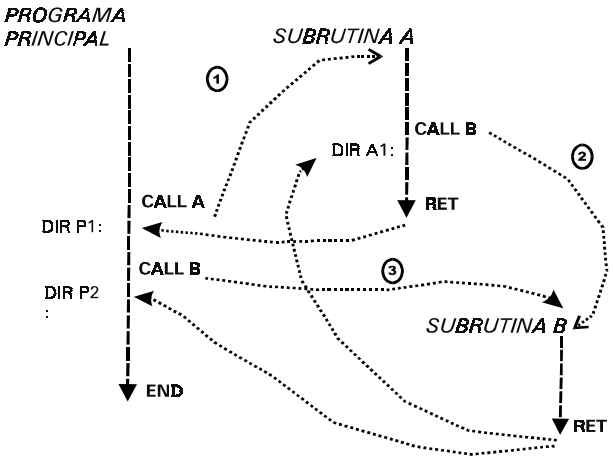

Figura 5.7. Llamada a subrutinas.

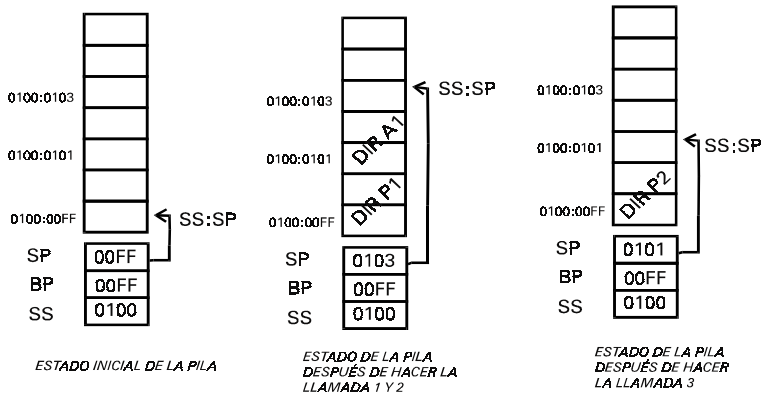

Figura 5.8. Ejemplo de pila.

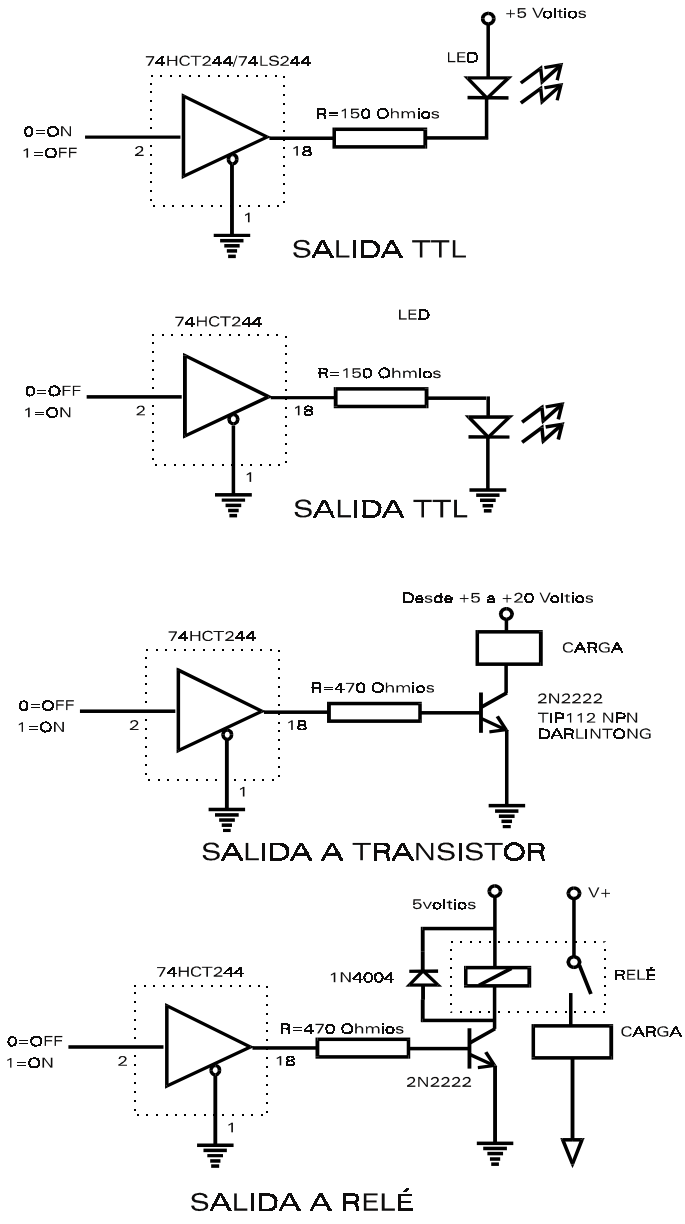

Figura 6.16. **Diferentes tipos de salidas.**

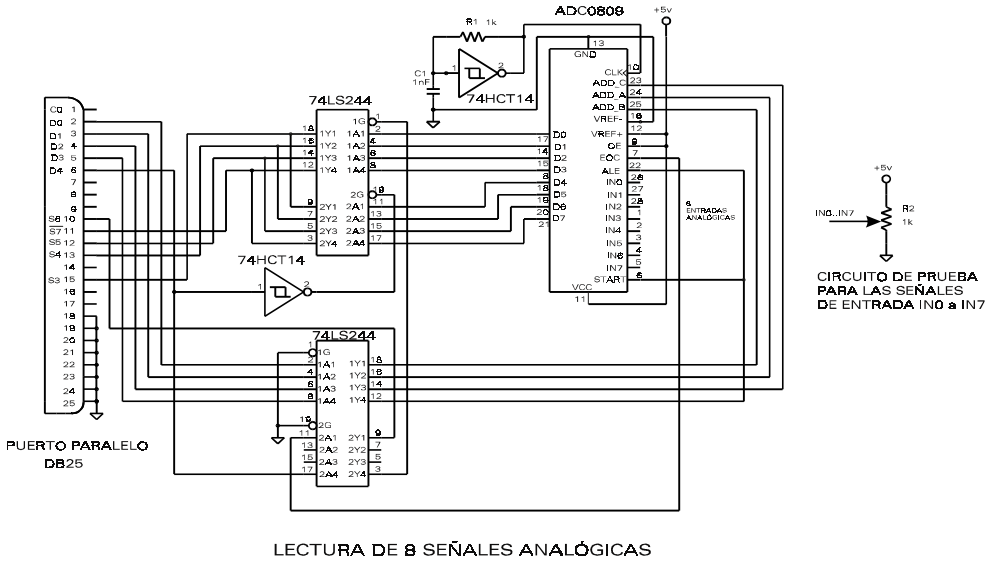

Figura 6.20. Lectura de 8 señales analógicas.

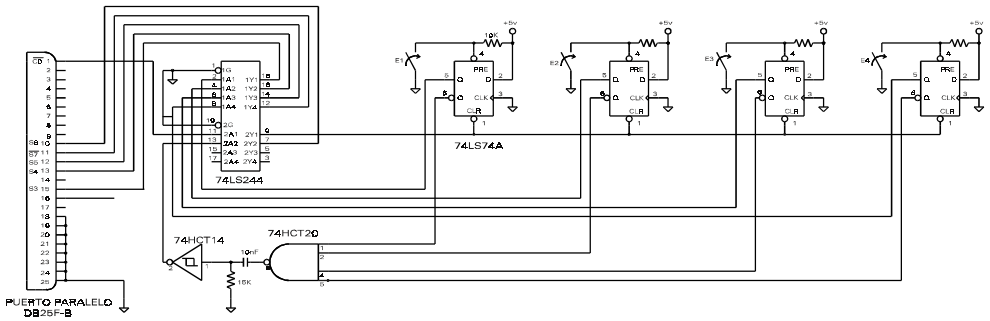

Figura 6.22. Detección de 4 interrupciones hardware.

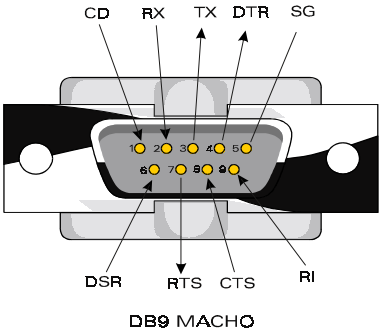

Figura 7.6. Conector DB9 macho con las señales RS-232.

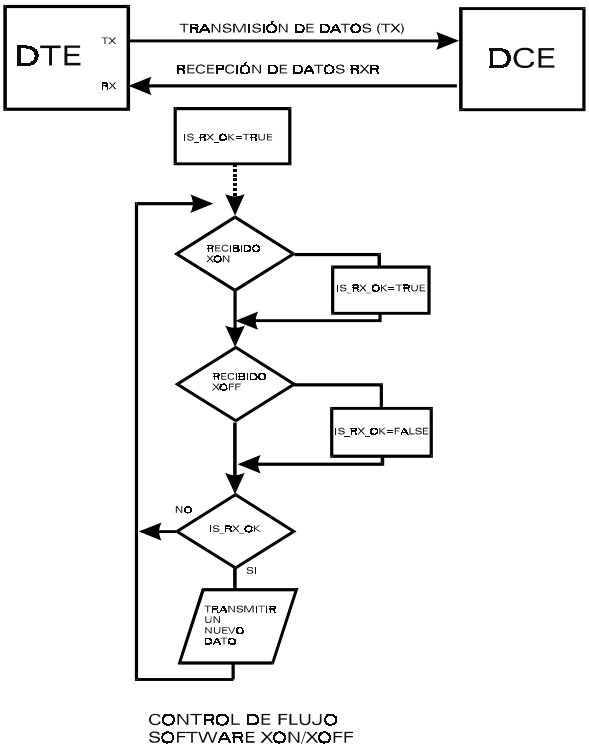

Figura 7.13. Control de flujo software XON/XOFF.

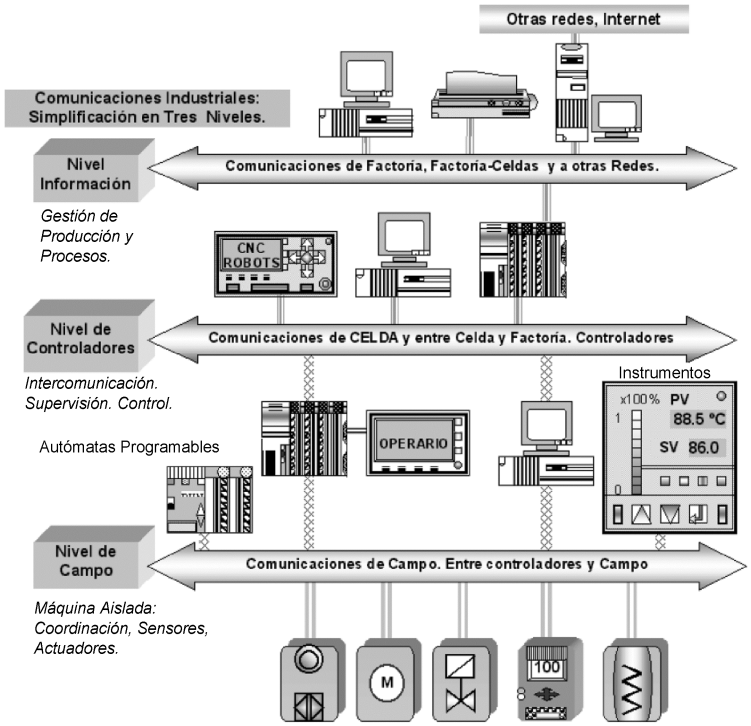

Figura 8.2. Estructura de los niveles de información. Algunos dispositivos realizan la función de gateway o pasarela: intercambio de datos entre redes de comunicación distintas. Generalmente un bus de campo soporta los niveles de controlador y dispositivos de campo.

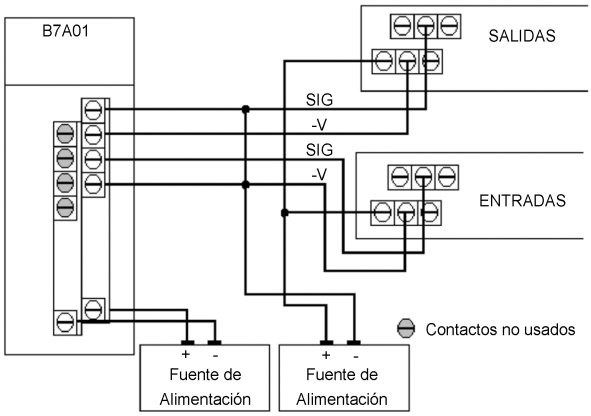

Figura 9.11. Conexión de la unidad remota B7A con dos fuentes de alimentación. De esta forma se consiguen distancias de hasta 500 m.

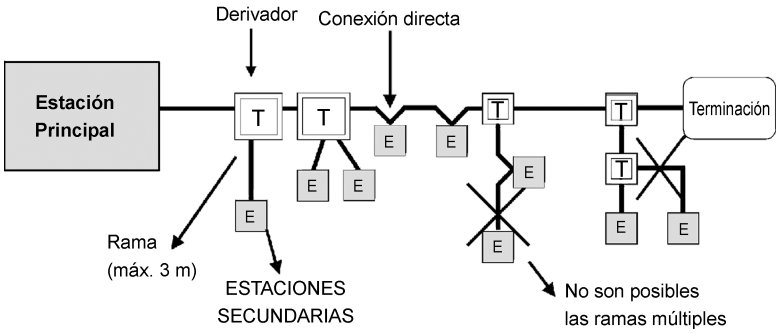

Figura 9.12. En el diagrama se muestran los diferentes componentes de un sistema Compobus/S.

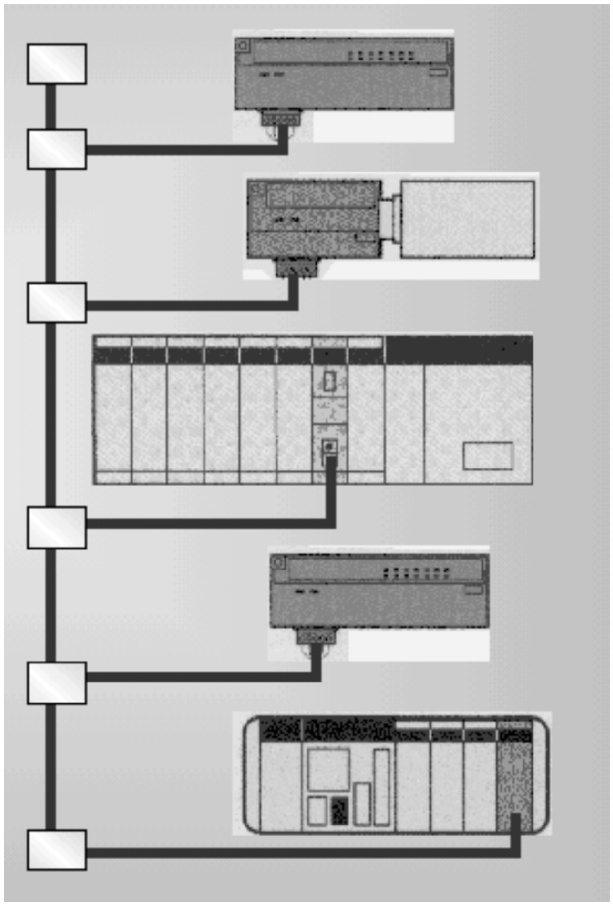

Figura 9.20. Conexiones del bus de campo. Cortesía de Omron Electronics, S.A.

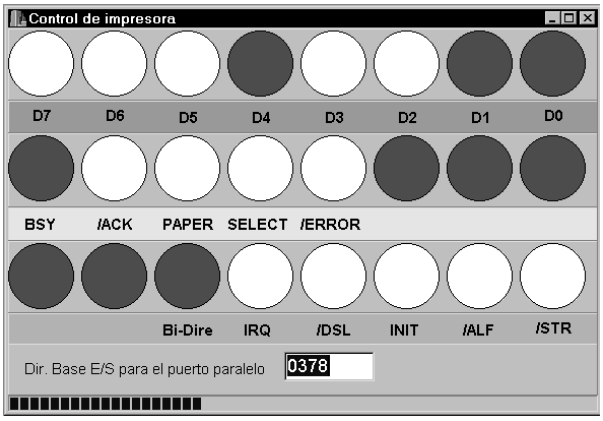

Figura 17.4. **Control del puerto paralelo.**

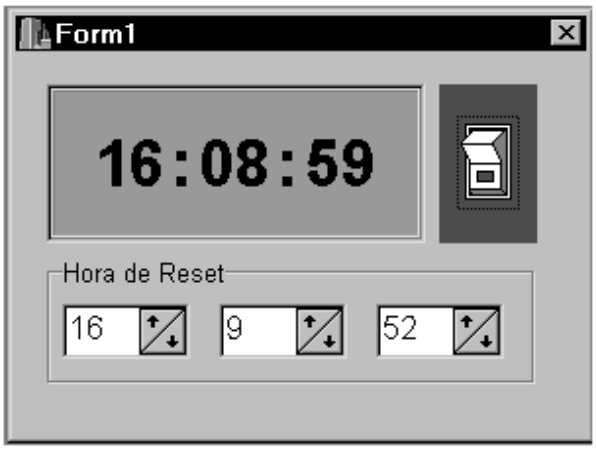

Figura 17.8. **Reset del ordenador.**