



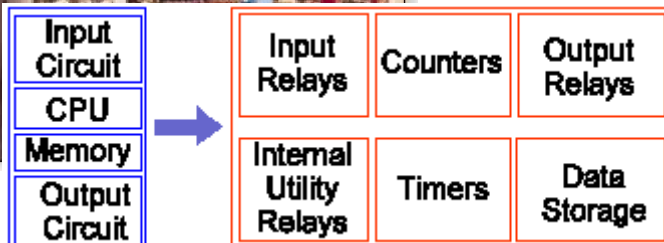
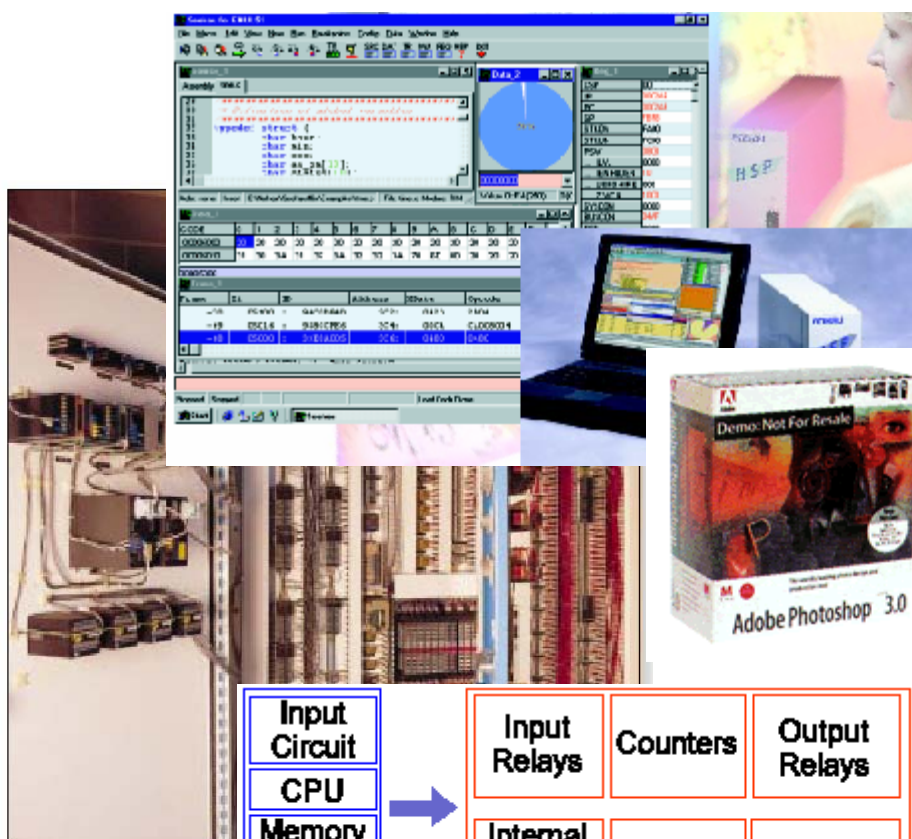
Universidad
de Huelva

DA
iESi

Escuela Politécnica Superior
Universidad de Huelva

Departamento de Ing. Electrónica,
Sistemas Informáticos y Automática

INTRODUCCIÓN A LA INFORMÁTICA INDUSTRIAL



Manuel Sánchez Raya
Versión 0.7
3 de Octubre de 2002

ÍNDICE

1.- Concepto de Informática Industrial.	2
1.2.- Aplicaciones de la Informática Industrial.	4
1.3.- Sistemas de Control Industrial.	5
1.3.1.- Sistemas basados en Autómatas (PLC).....	11
1.3.2.- Sistemas basados en PC.	13
1.3.3.- Sistemas basados en microcontroladores.	13
1.4.- Comunicación entre sistemas informáticos.....	14
1.4.1.- Redes locales.....	16
1.4.2.- Redes de comunicación industriales.	16

BIBLIOGRAFÍA:

Apuntes de Ingeniería Técnica Industrial. Universidad de Málaga.

TEMA 1 – Introducción al control industrial mediante sistemas informáticos.

1.- Concepto de Informática Industrial.

El concepto de Informática Industrial es bastante amplio. En sentido general hace referencia al software necesario para operar un computador en un entorno industrial. Este software estará compuesto de instrucciones del procesador que ejecutará la máquina

Los sistemas computadores, desde el punto de vista de su aplicación, se pueden clasificar en dos grandes grupos:

- [Sistemas de propósito general](#): diseñados para solucionar cualquier problema y siendo por tanto más versátiles.
- [Sistemas de propósito particular](#), generalmente autónomos denominados **sistemas empotrados**.

Estos últimos serán los que estudiaremos a lo largo del curso debido a su mayor profusión en entornos industriales. Como sabemos, no es el hardware el que diferencia a un sistema de otro, si no que el software es el que marca la diferencia entre un sistema apenas útil de otro que realiza perfectamente su cometido. Por tanto, el software constituye una parte muy importante del sistema y será aquí donde enfocaremos nuestro curso.

El software o “programas” corresponde a todo elemento lógico que tiene el computador y que junto al hardware confieren al computador la característica de una máquina virtual. El software empleado en los sistemas de propósito general se puede clasificar en dos grandes grupos:

- [Software básico](#): necesario para funcionamiento de E/S.
- [Software de aplicación](#): Aplicaciones del computador.

Software básico

Este tipo de software se utiliza principalmente como medio de comunicación entre el usuario y la máquina, ya que permite generar una interfaz de comunicación a nivel de hardware. Clasificación del software básico:

- [Assembler o ensamblador](#): es un software que permite programar a muy bajo nivel el microprocesador del sistema así como todos los componentes internos a nivel de hardware, como por ejemplo, chip, tarjetas, etc. Se emplea *lenguaje ensamblador* que se convierte a lenguaje máquina (0 y 1) que es el que entiende directamente el microprocesador.
- [Compiladores](#): permiten traducir de lenguaje de alto nivel (Códigos reconocidos y escritos por el hombre), a lenguaje ensamblador y luego mediante un ensamblador a código máquina (0 y 1), para que pueda ser interpretado por el computador.
- [Traductores](#): traducen de lenguaje de alto nivel a lenguaje de máquina, pero en memoria van traduciendo instrucción por instrucción del programa fuente (Programa base escrito en lenguaje de alto nivel), para convertirlo en el programa objeto (Programa escrito en código binario). Una vez que el programa ha sido depurado es más rápido compilarlo, evitando traducirlo cada vez que se necesite usar.
- [Lenguajes de programación](#): permiten generar programas para ser utilizados por el computador. Los niveles de los lenguajes tienen directa relación a la cercanía del programa en la máquina. De forma que un lenguaje de alto nivel estará más alejado del control directo del microprocesador que uno de bajo nivel. Los niveles que podemos encontrar en lenguajes de programación son:

- Lenguajes de muy bajo nivel, corresponden a lenguajes escritos en código nemotécnico para representar las instrucciones que maneja el procesador. Se utilizan principalmente para programar componentes electrónicos. Por ejemplo: ensamblador.
 - Lenguaje de bajo nivel, son lenguajes de programación intermedio entre el usuario y la máquina. Por ejemplo: compiladores y traductores.
 - Lenguajes de alto nivel, son lenguajes de programación cuyo objetivo es simplificar aún más la programación. Logran cierto grado de independencia entre el programa y el computador. Por cada instrucción escrita en lenguaje de alto nivel se obtiene un conjunto de instrucciones en lenguaje de máquina. Por ejemplo: Cobol, Clipper, Pascal, Fortran, Basic, etc.
 - Lenguajes de muy alto nivel, se emplean para la resolución de problemas específicos y poseen instrucciones propias de la aplicación en que son utilizadas. Por ejemplo: programas para comunicaciones.
- Sistemas operativos: tienen por misión generar una interfaz de comunicación entre el usuario y la máquina, siendo el único software que necesita el computador para que los demás programas puedan ser cargados. Entre los sistemas operativos hay una gran variedad, porque estos programas “conocen” los detalles de las máquinas y pueden administrar sus recursos en forma eficiente. Entre los sistemas operativos más comunes, tenemos el Disk Operating System más conocido como DOS, UNIX, ORACLE, Sistemas 7 u 8, usados por los equipos Macintosh, etc. Los Sistemas Operativos pueden ser:
- Monousuarios, atienden los requerimientos de un solo usuario a la vez.
 - Monoprocesador, procesan sólo los requerimientos de un usuario a la vez.
 - Multiusuario, atienden los requerimientos de varios usuarios.
 - Multiprocesador, pueden realizar varios procesos al mismo tiempo.
- Monitores: Son pequeños programas que se emplean para realizar tareas de mantenimiento en sistemas pequeños. Por ejemplo el empleado en las prácticas de la asignatura o la interfaz básica de E/S que poseen los PCs (BIOS).

Software de aplicación

El software de aplicación incluye Programas de Aplicación y Paquetes de Aplicación.

- Programas de Aplicación: este tipo de software está orientado al personal experto en computación, utilizando para su programación lenguajes de alto nivel. Estos programas tienen por misión resolver un determinado problema dentro de la organización. El costo de estos programas es alto ya que es asumido por la entidad que pide el servicio. Ejemplos: Sistemas de facturación, Control de inventarios, Sistemas de control de gastos, Sistemas de control de planta.
- Paquetes de Aplicación: este software está orientado a personas no expertas en computación. Tienen por misión resolver diversos problemas en las distintas áreas de la organización. Este software es de costo mayor que los Programas de aplicación, porque necesitan una gran cantidad de horas-hombre para su desarrollo. El costo real se prorratea entre los usuarios que adquieren el sistema. Entre los paquetes de aplicación más conocidos figuran: Excel, Word; programas de dibujo como Corel Draw, Autocad, etc. Clasificación de Paquetes de Aplicación :

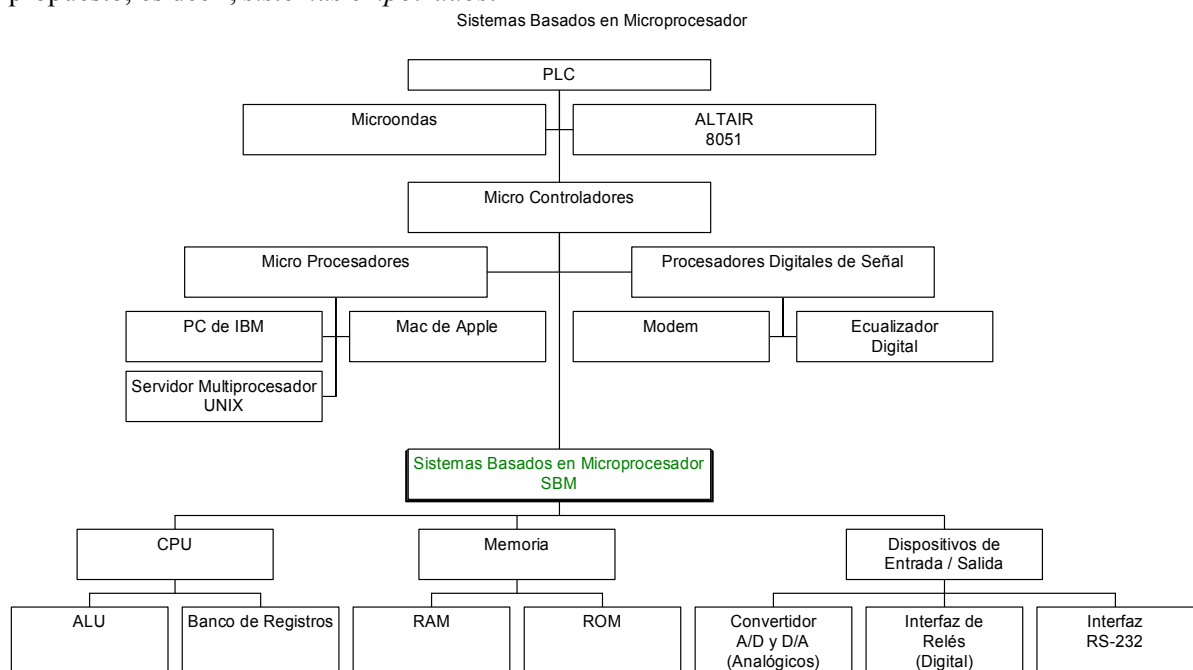
- Procesadores de texto, son programas que permiten automatizar las funciones de la redacción de documentos, con acciones como corrección ortográfica, envío de archivos por correo electrónico. Ejemplo: Word, Wordperfect, Wordstar, etc.
- Plantillas de Cálculo, son programas que permiten generar modelos matemáticos a través de una matriz compuesta por filas y columnas. Ejemplo: Excel, Quattro Pro, Lotus, etc.
- Bases de datos, son programas que permiten administrar una gran cantidad de datos contenidos en una base; realizar acciones como consultas, clasificación, generar reportes, etc. Ejemplo: FoxPro, Access.
- Aplicaciones Gráficas, permiten realizar presentaciones dinámicas para ventas, comercialización, circuitos, planos. Power Point, Corel, Autocad, etc.
- Aplicaciones EDA (Electronic Design Automation), permiten el diseño de circuitos electrónicos analógicos y digitales, así como el diseño de otro tipo de soportes, como placas de circuito impreso. Suelen ser paquetes integrados con todas las herramientas. Ejemplo: Orcad v9, Protel 99, Electronics Workbench, etc.

1.2.- Aplicaciones de la Informática Industrial.

Además del software de propósito general revisado en el epígrafe anterior, en entornos industriales encontramos otro tipo de software muy ligado al hardware y normalmente independiente empleado para controlar las maquina-herramienta que encontraremos en estos entornos. Este software suele ser muy específico y está destinado a:

- Control de motores.
- Control de procesos en general.
- Control de plantas de producción.
- Adquisición y almacenamiento de datos: Data Loggers.
- Control de motores de combustión interna: Automoción.

Cada aplicación tiene su hardware preferido, siendo todas ellas aplicaciones en las que tanto el hardware como el software son altamente especializados y diseñados específicamente para el fin propuesto, es decir, *sistemas empotrados*.



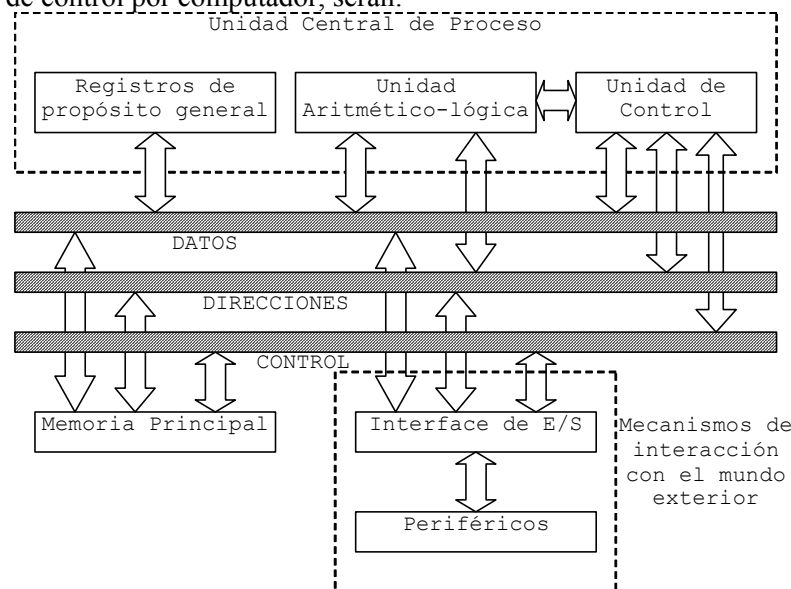
1.3.- Sistemas de Control Industrial.

Los sistemas de control industrial basados en microcontrolador no son más que aplicaciones del paradigma del microprocesador con sus elementos junto con una adaptación específica tanto del Hardware como del Software. Esto no quiere decir que no se pueda emplear un Hardware y Software de propósito general en estas aplicaciones, si no que la forma más común en que nos lo encontraremos será la de una aplicación “empotrada”, o sea, adaptada en gran medida a resolver el problema en cuestión y solo ese.

Los computadores de propósito general son modulares, permitiendo añadir nuevas unidades, en particular unidades específicas de E/S. Las capacidades de la unidad básica del sistema son:

- Potencia de procesamiento, se mide en instrucciones por segundo.
- Capacidad de almacenamiento, se mide en Bytes o Kbytes o Mbytes.
- Ancho de banda de E/S, se mide en transacciones por segundo o muestras por segundo.
- Estructura de interrupciones, se suele medir la latencia, que es el tiempo que se tarda en atender una interrupción en el peor de los casos.

Estas son las capacidades que determinan el comportamiento global del sistema. Los elementos de un sistema de control por computador, serán:



Todo sistema basado en microprocesador está compuesto por una unidad principal denominada **Unidad Central de Proceso** que es la encargada de buscar las instrucciones a ejecutar en la **memoria del sistema**, otro componente del sistema y por último la **Unidad de Entrada/Salida** que es la encargada de conectar el sistema con el mundo real a través de convertidores analógico/digital si el entorno es analógico o directamente a través de relés si la operación a realizar es digital. Todo ello se interconecta a través de los **buses** de datos y direcciones.

La Unidad Central de Proceso esta compuesta de los registros, más la ALU y la unidad de control, sus características determinan la potencia de procesamiento:

- Longitud de palabra.
- Juego de instrucciones.
- Métodos de direccionamiento.
- Número de registros.
- Velocidad de transferencia de datos.
- Estructura de interrupciones.

Almacenamiento.

El sistema de almacenamiento se suele componer de:

- [Memoria principal](#): formada por una memoria de acceso rápido (RAM o ROM) donde se guarda el programa y/o los datos.
- [Memoria secundaria auxiliar](#), normalmente formada por discos o por memorias FLASH emulando discos magnéticos, son dispositivos asíncronos y se emplean para almacenar programas o datos a los cuales no se accede con gran frecuencia.
- [Memoria Cache](#): Suele tener varios niveles jerárquicos y se emplea para aumentar la velocidad de acceso a la memoria principal basándonos en el principio de localidad (la siguiente instrucción a ejecutar casi siempre se encuentra muy cerca de la actual instrucción).

Entrada/Salida

Las interfaces de E/S representan una de las áreas más complejas en sistemas computadores, debido en principio a la gran variedad de unidades y el amplio rango en la velocidad de transferencia de datos y métodos de comunicación. Por ejemplo, podemos encontrar:

- Impresoras, teclados, ratón, monitor, unidades específicas, etc.
- Puerto serie, paralelo, SCSI, sonido, etc...

Estructura del Bus.

Un bus es un conjunto de hilos o conductores que permiten la transferencia de información entre los distintos elementos del computador y se pueden describir a tres niveles: mecánico, eléctrico y funcional.

- [Nivel Mecánico](#): especifica el número de conductores (pistas o cables).
- [Nivel Eléctrico](#): voltajes de las señales, tipos de puertos de salida (colector abierto, triestado, etc).
- [Nivel Funcional](#): describe el tipo de información que circula en el bus (DATOS, DIRECCIONES, SEÑALES DE CONTROL).

Interfaces.

Realizan el enlace entre el computador y los sensores y actuadores externos en el proceso de control. Las unidades externas (sensores) pueden suministrar tres tipos de información:

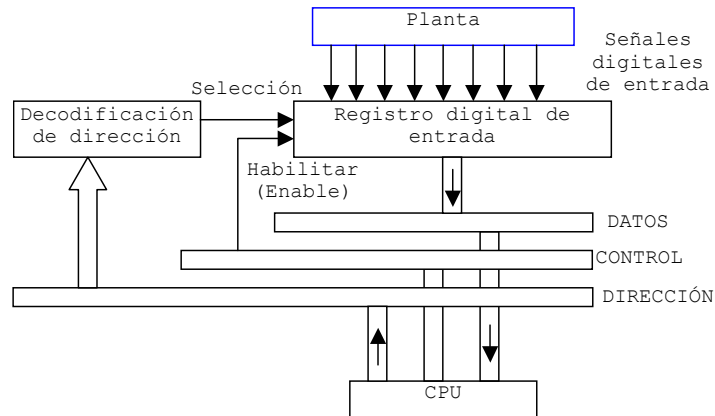
- [Digital](#).
- [Analógica](#), mediante una digitalización previa
- [Pulso](#).

Cada tipo de sensor requiere un tipo específico de interfaz que generalmente necesita incorporar al computador una tarjeta especializada.

Las interfaces también se utilizan para la comunicación entre distintos computadores y operadores remotos, para de esta forma integrar un control distribuido. Estas interfaces se puedan considerar como interfaces digitales ya que la información se emite en binario con un protocolo específico (RS-232, Ethernet, etc).

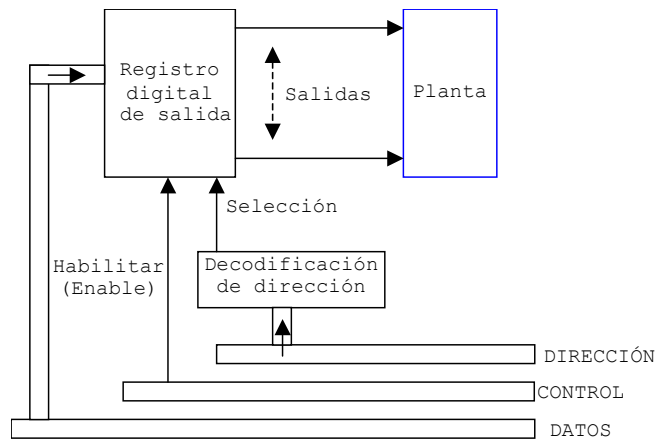
Interfaz Digital.

La interfaz digital de entrada se puede observar en la figura siguiente:



El registro de entrada tiene normalmente el mismo número de bits que la palabra del computador empleado. Los niveles de entrada son 0 y 5V. El decodificador de direcciones detecta una petición de datos de la CPU.

La señal de habilitación es necesaria para volcar el registro en el bus. Asegura que los datos permanecen estables (orden /RD de control). El tiempo de transferencia está gobernado por el reloj del sistema.



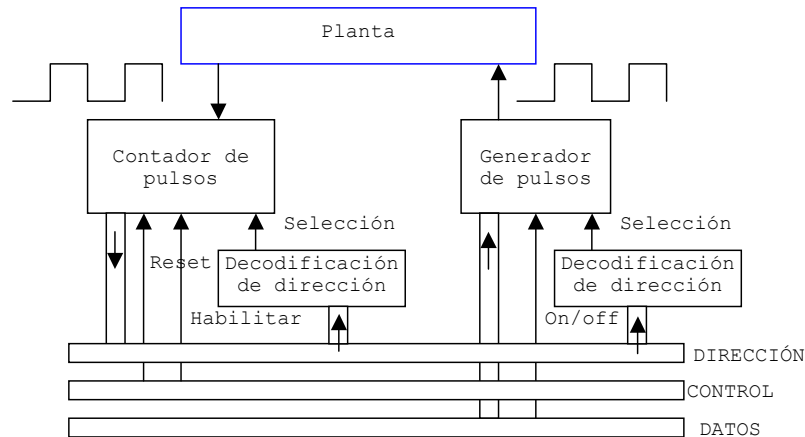
El registro de salida almacena los datos cuando es direccionado y la señal de selección indica que los datos son estables (tiempo típico de respuesta de 1 microsegundo).

Los valores de salida son 0V o +5V. Si estos valores no son adecuados para operar con los actuadores externos se deberá convertir la señal.

Existen circuitos integrados específicos para comunicación digital que son desarrollados para la mayoría de empresas de fabricación de computadores.

Interfaz de Pulso.

Está compuesto por temporizadores/contadores (timers) y toma la siguiente forma:



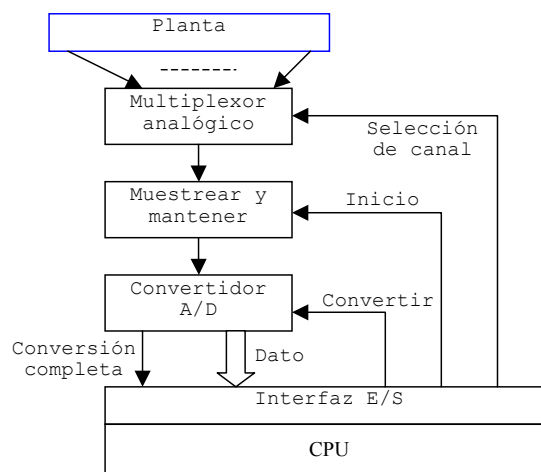
En el esquema anterior podría haberse eliminado el contador de pulsos conectando la entrada directamente a una interrupción de la CPU y realizar la cuenta de pulsos mediante programa.

Los tipos de salida de pulso generalmente suelen ser los siguientes:

- Serie de pulsos de duración fija.
- Pulso único de longitud variable.
- Pulso con modulación (PWM).

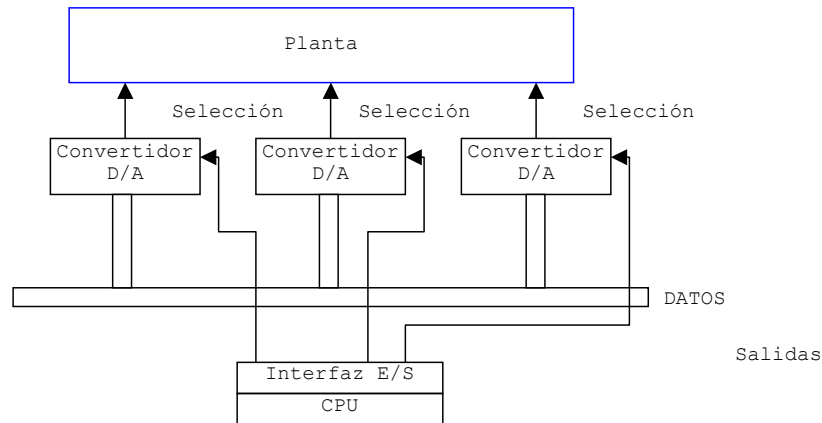
Interfaz analógica.

Podemos dibujarla de forma esquemática como:



En este esquema se realiza muestreo y cuantificación. La señal de conversión completa se puede muestrear de forma periódica (polling) o se puede usar como señal de interrupción. Si las señales analógicas de entrada varían en un amplio rango se suelen usar amplificadores de ganancia programable.

Se pueden simultanear etapas y de esta forma aumentar la velocidad.



Los convertidores D/A son más simples y económicos, se emplea uno por línea. Tienen un tiempo de conversión de 5 a 20 milisegundos y un rango de salida típico de -5 a $5V$ o de -10 a $10V$.

Reloj de tiempo Real.

Es un elemento vital para los sistemas de control por computador. Básicamente se trata de un generador de pulsos (timer) con una frecuencia muy precisa. En el caso más simple se utiliza la tensión alterna de la red eléctrica (50 ó 60Hz) o bien la propia señal de reloj de la CPU.

Los pulsos se emplean para generar interrupciones y contando los pulsos mediante un contador podemos disponer de relojes con distintas precisiones.

Para obtener frecuencias de reloj altas y precisas se necesitan circuitos basados en osciladores a cristal y contadores. Hay que tener en cuenta que mientras mayor sea la frecuencia del reloj en tiempo real, mayor será la precisión del sistema en discernir eventos pero mayor la carga computacional para la CPU.

Se suelen emplear baterías para mantener el reloj en hora cuando falla la alimentación. La interrupción del reloj de tiempo real resulta básica para los sistemas multitarea en un computador con una sola CPU pues emplean la interrupción generada para distribuir las tareas.

Técnicas de transferencia de datos.

Una de las características de la mayoría de las interfaces es que trabajan de forma síncrona con el computador pero a una velocidad mucho más lenta. El control directo de la transferencia por parte de la CPU, llamada “transferencia por programa”, ofrece la máxima flexibilidad pero también es muy ineficiente, debido a la diferencia de velocidad entre la CPU y la unidad interfaz.

En la actualidad es común el uso de procesadores de entrada/salida especializados, que descargan a la CPU de las operaciones de transferencia. El principal problema es el de la temporización, porque podemos tener:

- Transferencias incondicionales: siempre se pueden realizar.
- Transferencias condicionales: necesitan sincronizarse con la CPU

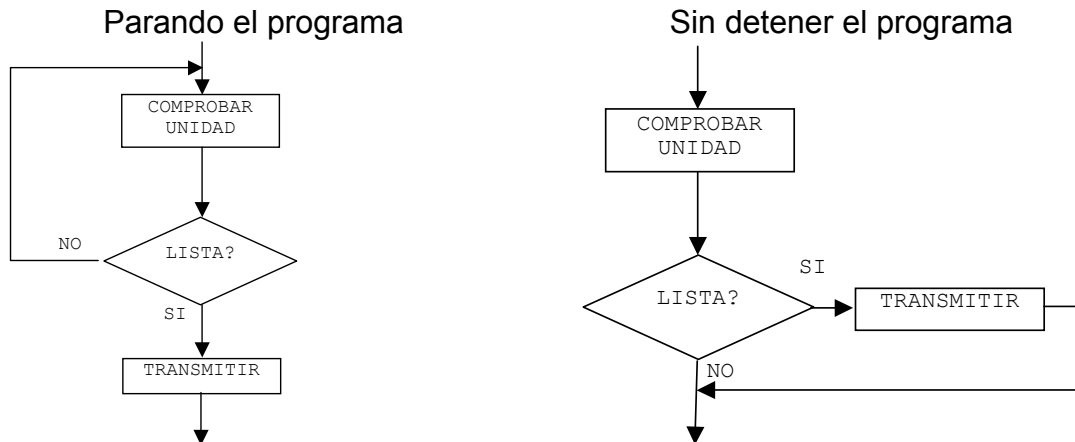
Las técnicas empleadas son:

- [POLLING](#) o encuesta: el procesador está pendiente de la llegada de los datos.
- [INTERRUPCIONES](#): el procesador es avisado cuando esta disponible el dato.

- [DMA](#): otro elemento se encarga de transferir la información de E/S hacia la memoria.

Polling

La técnica de polling consiste en comprobar si el dispositivo está listo para recibir o transmitir datos antes de hacer la transferencia. Podemos hacerlo de dos formas:

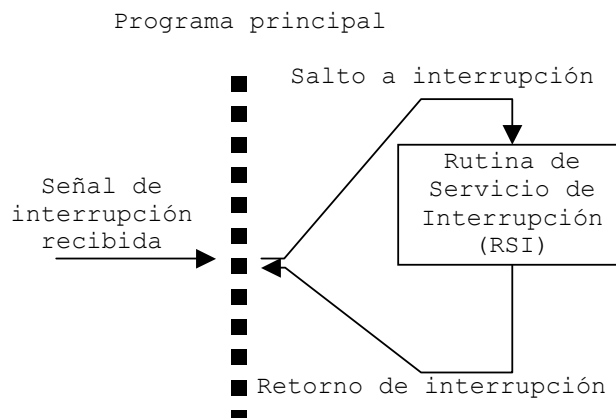


En una primera aproximación, parando el programa, la CPU espera en un bucle activo a que la unidad esté lista, lo que es muy ineficaz desde el punto de vista del uso de la CPU. Se podría implementar sin necesidad de comprobar la unidad utilizando bucles de retardo, pero esto es aún más ineficaz.

Podemos mejorar el anterior sistema evitando detener el programa con el segundo método, si la unidad no está lista entonces la CPU continúa realizando otras operaciones, pero deberá volver a este punto pasado cierto tiempo, esto implica una programación un poco más compleja.

Interrupciones.

Definición: Es un mecanismo mediante el cual el flujo de un programa puede ser temporalmente detenido para permitir la ejecución de un software específico (llamado manejador de interrupción).



Las interrupciones solucionan el problema de la espera activa del “polling” pero también se usan para:

- Mantener el reloj del sistema.
- Entradas de alarma procedentes de un sistema a controlar.
- Indicaciones de fallo hardware.
- Depuración de programas.
- Llamadas al Sistema Operativo.
- Control de fallos de alimentación.

Una vez ha tomado el control la rutina de servicio de interrupción (RSI) es necesario guardar temporalmente los registros que se están usando por el programa interrumpido. Los registros de la CPU representan el estado actual del computador, y por tanto la información que contienen debe ser almacenada antes de atender una interrupción para poder así continuar la ejecución del programa principal una vez atendido dicha interrupción. Hay tres posibilidades:

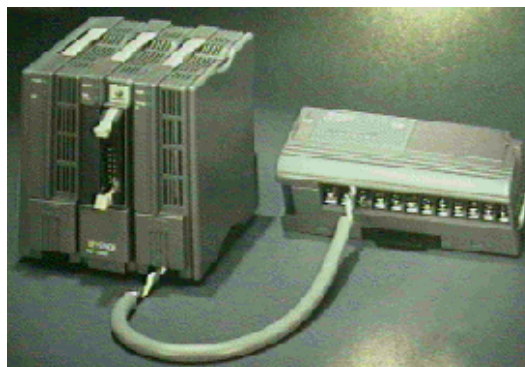
- Almacenar los registros en un [área específica de memoria](#) (no se permite interrupciones múltiples).
- Almacenar los registros [en la pila](#) (hay que evitar el peligro de desbordamiento).
- Utilizar un [conjunto auxiliar de registros](#). Como se hace con el 8051.

Generalmente, los registros se guardan de forma automática, lo que resulta ineficiente si solo van a ser usados parte de ellos por la interrupción. A menos que la velocidad sea crítica es aconsejable salvar todos los registros. Generalmente, los sistemas que salvan automáticamente los registros también los restauran de forma automática.

1.3.1.- Sistemas basados en Autómatas (PLC).

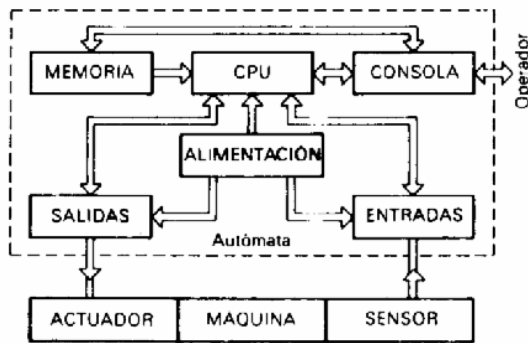
Los sistemas basados en Autómatas Programables tienen una estructura software y hardware particular. Normalmente están compuestos por una unidad principal basada en un microcontrolador que suele tener unos 32K a 128K de RAM ampliables como memoria de trabajo. Esta unidad principal se encuentra protegida del polvo y humedad de forma que se pueda emplear en ambientes un tanto hostiles.

Las características mecánicas y eléctricas de estos equipos son objeto de un estudio de normalización que se ha convertido en la norma Afnor NF-C 63 850. Esta norma tiene en cuenta la noción de integración en medio industrial, definiendo un autómata como “instrumento electrónico, con memoria programable mediante un lenguaje adaptado, para el almacenamiento interno de instrucciones que componen las funciones de automatismo, por ejemplo: lógica secuencial y combinatoria, temporización, contaje, descontaje, comparación, ajuste, mando, regulación, etc., para controlar, medir o supervisar, por medio de módulos de entrada y de salida (lógica digital y analógica), distintas salidas de máquinas o de procesos, en ambiente industrial”



Estructura interna.

Un autómata programable está constituido por un conjunto de tarjetas o de circuitos en los que se han montado componentes electrónicos integrados. Estas tarjetas se alojan en panel o cajas, para ser protegidos mecánicamente. Las conexiones entre tarjetas tienen lugar a través de un circuito impreso denominado “bus” situado en la parte posterior. Entre los principales constituyentes de un autómata programable figuran: la unidad central, la memoria, las tarjetas de entrada y de salida, la consola de programación y la alimentación.



La unidad central es la parte inteligente de la máquina: es ahí donde se interpretan las instrucciones. De su potencia depende el grado de complejidad de los automatismos que puedan ser resueltos y la facilidad con que se efectúa la programación. La tarjeta de memoria permite indicar a la unidad central qué es lo que debe ejecutar: es el contenido de esta memoria lo que se modifica cuando se desea realizar una nueva configuración de automatismo. Se utilizan varias tecnologías para las memorias: FLASH, RAM, ROM.

Las tarjetas de salida conectan el autómata con los actuadores de la máquina y actúan sobre ellos. Los actuadores pueden ser testigos, contactores, electroválvulas, relés, etc. Las tarjetas de entrada conectan al autómata los sensores, proporcionándole la información que éstos captan. Se pueden conectar varios tipos de sensores: finales de carrera, pulsadores, detectores de proximidad, etc., u otros de tipo analógico.

Consola de programación.

La consola de programación es el instrumento con el que el operador introduce datos, efectúa modificaciones del programa, y que le indica el estado de algunas variables, así como los defectos que se pueden producir en la máquina. Finalmente, la alimentación proporciona la tensión y corriente necesaria a los circuitos, así como alimenta las entradas y salidas.

El microprocesador del equipo ejecuta un cierto número de operaciones elementales denominadas “instrucciones”. Lee e interpreta las instrucciones del programa grabado en memoria y deduce de él las operaciones a efectuar. Algunas de estas unidades se pueden atornillar a raíl estándar DIN y se pueden apilar con otras unidades que dotan al conjunto de una E/S mediante relés, convertidores A/D y D/A. El sistema queda por tanto compuesto por varios módulos de E/S y un módulo principal.

La mayoría de Autómatas incorpora de serie RS-232 para comunicarse con un PC o panel de programación y algunos también RS-485 para comunicarse con otros PLCs o accesorios. A través de esta interfaz y de un software que se suele ejecutar en PC se programa la tarea de control que va a realizar el autómata con un lenguaje un tanto peculiar. Esto se debe a que antes que existieran los autómatas programables las tareas de control se realizaban con grandes paneles de relés y temporizadores basados en válvulas de vacío. Estos se programaban con “diagramas de relés”. El lenguaje empleado para los autómatas está basado en estos diagramas de relés.

1.3.2.- Sistemas basados en PC.

El PC de IBM surgió en el año 1982. Desde entonces su gran difusión y bajo coste ha permitido el uso en gran variedad de aplicaciones tanto industriales como no industriales.

Los PC se emplean cuando se desea instalar una interfaz basada en teclado/ratón/pantalla para administración de procesos por ejemplo, o bien en aplicaciones donde se requiera intervención humana. Como ejemplo en la terminal de programación de una máquina herramienta encontraremos casi siempre un PC.

También se emplean cuando el algoritmo de procesamiento es relativamente complejo o hay que procesar gran cantidad de sensores simultáneamente y un microcontrolador común se nos queda corto en velocidad de procesamiento.

Tiene las siguientes ventajas:

- [Bajo coste](#), al fabricarse tanto chipset como procesador central por diferentes casas comerciales, lo que facilita la competencia y la posibilidad de intercambiar partes del sistema, abaratando tanto la puesta en marcha como el posterior mantenimiento.
- Existencia de [gran cantidad de software y de herramientas de desarrollo](#) para poder emplearse en cualquier aplicación. Al ser una plataforma muy conocida, los desarrolladores de software se interesan en preparar programas para ésta.
- Al contar con Sistema Operativo y Herramientas de Desarrollo de alto nivel (compiladores) el [desarrollo del sistema suele ser muy rápido](#).

A pesar de las ventajas citadas, sus principales desventajas son la *rápida obsolescencia*, que puede hacer difícil encontrar repuestos en un momento dado, y el *consumo de energía* de estos procesadores que suele ser bastante grande, lo que hace imposible su uso en aplicaciones móviles y portátiles.

1.3.3.- Sistemas basados en microcontroladores.

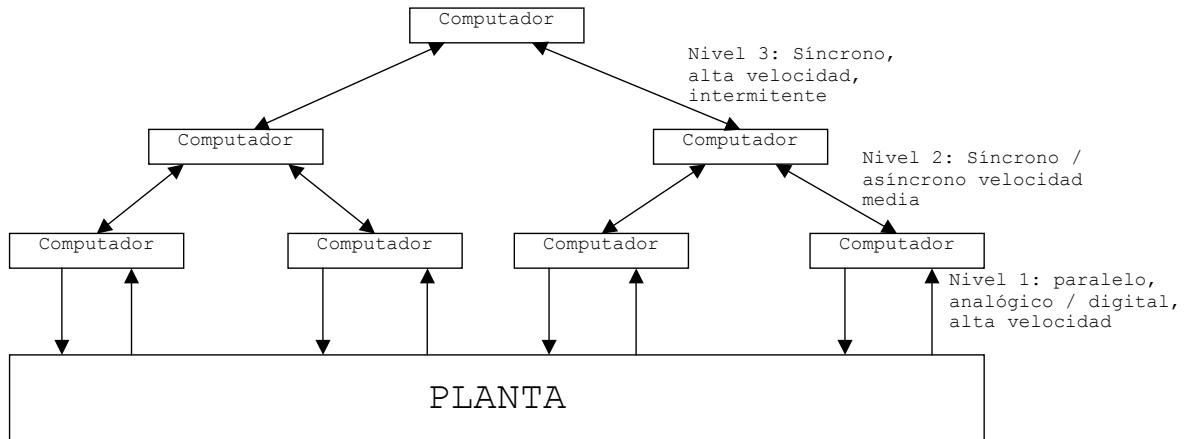
Los sistemas basados en microcontroladores se emplean cuando las tareas de control o supervisión no son muy complejas y/o no requieren una respuesta muy rápida. Esto se debe a que los sistemas microcontroladores al disponer de recursos limitados no pueden desarrollar una potencia de cálculo excesiva. Por tanto, su aplicación principal será en dispositivos portátiles, de bajo consumo de energía y aparatos de bajo costo/gran consumo, como por ejemplo: teléfonos, relojes, televisores, radios, microondas, neveras, etc...

Estos sistemas no poseen generalmente sistema operativo, siendo el software que ejecutan totalmente adaptado a la aplicación que realizan. Estos sistemas suelen tener grandes restricciones tanto en Hardware como en Software. En cuanto al hardware, resulta mínimo el número de componentes empleados, siendo casi siempre solo un circuito integrado (el propio microcontrolador). En cuanto al software, la capacidad de memoria tanto RAM como ROM queda limitada a la contenida en el propio microcontrolador.

Esta adaptación a la aplicación (base de lo que denominamos *sistemas empotrados*) tiene su razón de ser en la premisa de su **bajo coste**. Como debe venderse en grandes cantidades, el coste debe ser mínimo, pasando cualquier funcionalidad que se pueda realizar con componentes analógicos al software para eliminar estos componentes y por tanto reducir su costo. Esta es una de las razones de la profusión de la palabra “digital” en todos los aparatos de consumo, así como de los procesadores digitales de señal: es más barato hacerlo digital y además más fácil de ajustar los parámetros una vez fabricado el aparato.

1.4.- Comunicación entre sistemas informáticos.

Los grandes sistemas utilizan una arquitectura de control distribuida, esto es, varios computadores que se comunican entre sí siguiendo una estructura jerárquica. Esto implica la necesidad de comunicación entre las distintas máquinas y los instrumentos de la planta.



Los niveles 3 y 2 utilizan comunicación digital serie, debido a las distancias que separan los elementos. Las interfaces analógicas tienen problemas de baja relación señal/ruido y las paralelas tienen un coste elevado en grandes distancias.

Con las comunicaciones serie hay que tener en cuenta varias características:

1. Tipo de conductor utilizado que forma el enlace físico real entre equipos:
 - a. Par trenzado.
 - b. Coaxial.
 - c. Fibra óptica.
 - d. Radio.
2. Técnica de transmisión digital:
 - a. Codificación directa.
 - b. Modulación en frecuencia o fase.
3. Modo de transmisión:
 - a. Síncrono.
 - b. Asíncrono.
4. Protocolo de comunicación:
 - a. Orientado a carácter.
 - b. Orientado a bits.
 - c. Orientado a bloques.
5. Interconexión de redes, cuando se emplean varios computadores, LAN (Local Area Network):
 - a. Topología en estrella.
 - b. Topología en anillo.
 - c. Topología en bus.

Tipo de conductor.

Par Trenzado. Está realizado por hilos de cobre aislados y trenzados. Como ventajas tiene un precio muy asequible, un tamaño reducido y son manejables y de fácil instalación. Como desventaja sus características eléctricas quedan por debajo del resto de los conductores, ya que presentan una elevada atenuación y distorsión de la señal transmitida.

Su ancho de banda (capacidad de transmisión) se limita a unos pocos MHz, lo que no permite velocidades de transmisión superiores a unos pocos Mbits hasta distancias de un par de kilómetros. Ejemplo: ADSL, Red Telefónica Conmutada.

Cable coaxial. Tiene una atenuación muy baja y menor distorsión, dispone de un ancho de banda de hasta 500MHz y buena inmunidad al ruido eléctrico. Se emplea con transmisiones moduladas en frecuencia.

Fibra óptica. Representa una alternativa a los conductores eléctricos, En este caso se transmiten señales luminosas a través de un material transparente (silicatos). Su principales ventajas son que presenta una gran facilidad de instalación, debido a su pequeño diámetro, peso y radio de curvatura, tiene pequeña atenuación y distorsión de la señal transmitida permitiendo velocidades de transmisión muy elevadas y resulta insensible a la interferencias electromagnéticas.

Como principal desventaja tenemos su coste elevado y que la realización de conexiones es muy difícil, por lo que hay que emplear herramientas especiales y costosas.

Sin embargo, son cada vez más empleadas y presentan grandes posibilidades de expansión en la transmisión de datos al emplear haces de luz de diferentes colores. Su aplicación principal son las plantas de procesamiento donde puede haber peligro de explosión, por ejemplo en las refinerías.

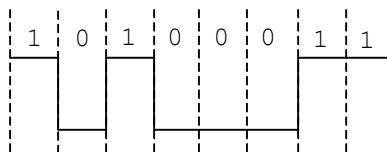
Técnicas de transmisión digital.

Tenemos dos posibilidades para la transmisión de una secuencia de bits:

- Codificación directa: existen varias técnicas que varían la sensibilidad al ruido y distorsiones y a la facilidad de sincronización.
- Modulando en amplitud, frecuencia o fase una señal que hace de portadora y siempre tiene una frecuencia más alta.

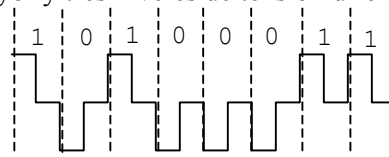
Codificación directa. Cada bit se representa por un valor digital o un cambio en la señal que transporta la información:

- a) Binario directo o NRZ (Non return to zero): es el método más directo, la tensión de 0V representa el cero lógico y la de +5V representa el uno lógico. Es muy sensible al ruido y la distorsión, para compensarlo se usan umbrales ($<1V \rightarrow 0$ lógico, $>3V \rightarrow 1$ lógico). Con esta codificación no es posible detectar cuando comienza un bit o cuando termina, ni la velocidad de transferencia si esta puede variar. Ej: RS-232 (con otros niveles de tensión).

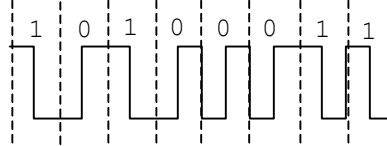


- b) RZ (Return to zero): Soluciona varios problemas del anterior utilizando dos niveles de polaridad opuestos con respecto a cero. Cada bit comienza a cero o uno (potencial negativo o positivo) y se va a potencial cero en la mitad del pulso (retorno a cero), esto

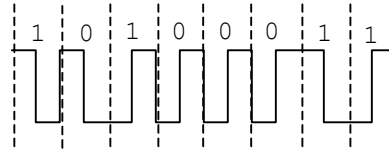
permite utilizar esta transición como señal de sincronismo. Como desventaja requiere un ancho de banda mayor y tres niveles de tensión diferentes.



- c) **Código Manchester**: también denominado bifase, utiliza sólo dos niveles para la codificación de un bit al igual que NRZ y una transición obligatoria en la mitad de cada pulso, lo que se emplea para sincronización. Ejemplo: IEEE 802.3



- d) **Código Manchester diferencial**: es un esquema similar al anterior en el que un bit cero es representado como una transición al comienzo de cada nuevo periodo y un bit uno como la ausencia de dicha transición, existiendo una transición obligatoria en mitad del pulso.



Aunque c) y d) emplean un mayor ancho de banda que los anteriores, debido a sus ventajas son muy utilizados.

Modulación en frecuencia o fase.

Se suele utilizar para comunicaciones a largas distancias como es el caso de los “modems” que utilizan las líneas telefónicas analógicas para efectuar la conexión física. Un MODEM de velocidad media utiliza frecuencias portadoras de 800 a 2500 Hz.

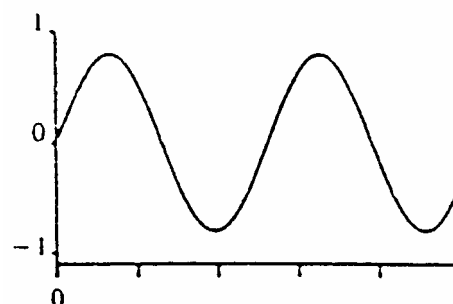
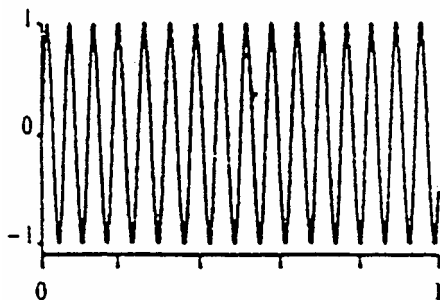
La modulación no tiene porque corresponder directamente a la señal de entrada si determinados cambios en parámetros de la señal portadora pueden asociarse con una determinada secuencia de bits, en lugar de un solo bit.

La técnica QAM (Quadrature Amplitud Modulation) combina la modulación de fase y de amplitud para transportar varios bits por cada cambio en la señal portadora.

Señal que modulada y portadora.

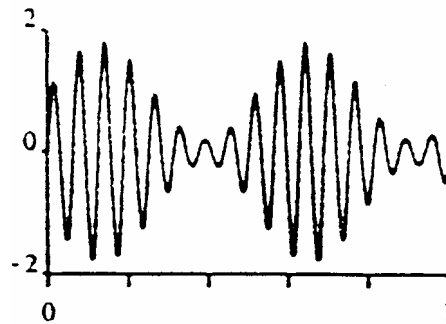
Señal portadora: $u_C(t) = u_0 \sin(\omega_C t + \phi)$

Señal moduladora: $u(t)$



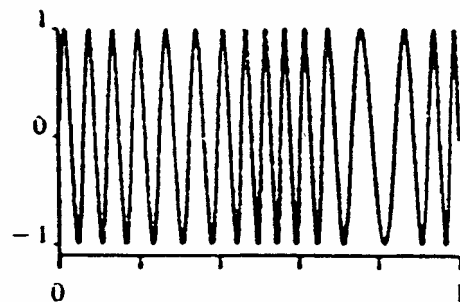
AM. Modulación de la amplitud.

Ecuación: $u_{AM}(t) = (u_0 + k u(t)) \sin(\omega_c t + \phi)$



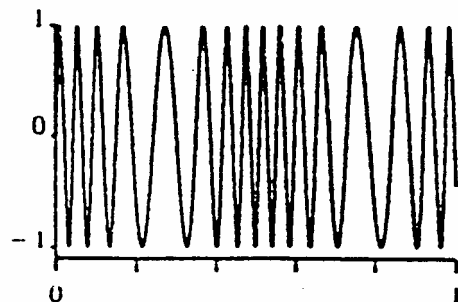
FM. Modulación de la frecuencia.

Ecuación: $u_{FM}(t) = u_0 \sin((\omega_c + k u(t))t + \phi)$



PM. Modulación de la fase.

Ecuación: $u_{PM}(t) = u_0 \sin(\omega_c t + k u(t) + \phi)$



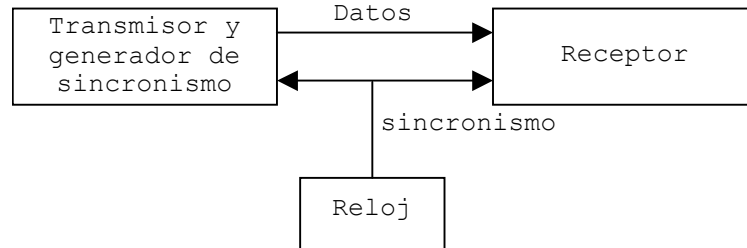
Modos de transmisión.

Es necesario que el transmisor y el receptor compartan una referencia de tiempo común, es decir que estén sincronizados. Esta referencia consiste en una señal en forma de pulso con una frecuencia definida, generada por alguna de las unidades de comunicación o por una unidad interna. Existen tres métodos de sincronización :

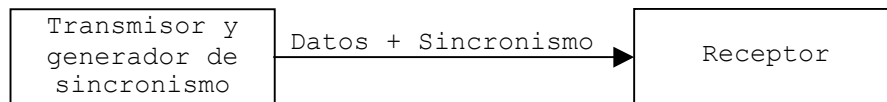
- El transmisor proporciona los datos y la señal de sincronismo.



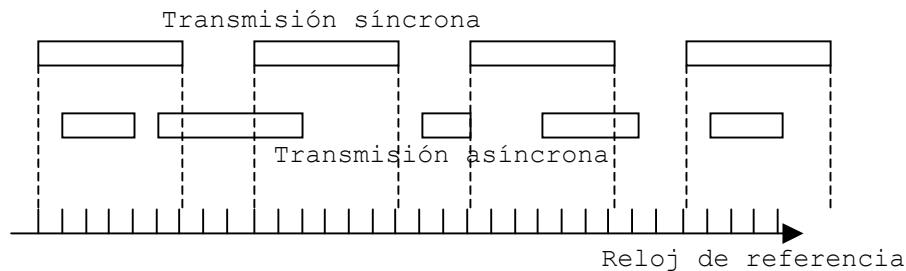
- Transmisor y receptor comparten la señal de sincronismo.



- Solo se manda una sola señal, donde se mezcla la señal de datos con la de sincronismo.



Si bien la transferencia a nivel de bits es siempre síncrona, el sistema de transmisión se considera síncrono si el instante de comienzo y parada de la transmisión es determinado por una referencia externa temporal de carácter fijo, es decir, la transmisión siempre se produce en instantes de tiempo específicos y periódicos. En caso contrario la transmisión se considera asíncrona.



Si la transmisión es síncrona y no hay mensajes a transmitir siempre se envía un mensaje vacío en los periodos de transferencia para asegurar el sincronismo.

Protocolos de comunicación.

Los protocolos de comunicación se emplean para:

- [Iniciar y terminar](#) una transmisión.
- [Verificar la validez](#) de los datos recibidos e incluso en algunos sistemas corregir los errores de transmisión.
- [Enviar más cantidad de datos](#) de los que puede soportar el canal, utilizando la división del canal entre varios usuarios. Esto se conoce con el nombre de multiplexación y permiten que varios usuarios usen el mismo canal de forma “simultánea” (canal virtual). El canal virtual representa una fracción de la capacidad del canal físico original.

La multiplexación puede ser:

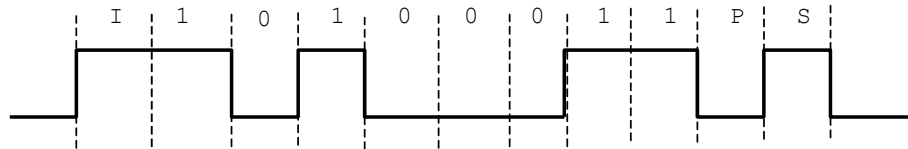
- [Multiplexación en el tiempo \(TDM\)](#): a cada usuario se le asigna un tiempo de canal fijo.
- [Multiplexación en frecuencia \(FDM\)](#): el ancho de banda se divide para diversos canales virtuales.
- [Multiplexación estadística](#): si el canal está ocupado se realiza una nueva petición tras un tiempo de espera aleatorio.

Protocolos orientados a carácter.

Se suelen utilizar en la transmisión asíncrona, en la que se envía cada byte (carácter) de forma independiente, por ejemplo en la interfaz serie RS-232.

El carácter va precedido de un bit de comienzo (I) que permite al receptor sincronizarse. El byte termina con un bit de paridad (P) que permite detectar errores y un pulso de 1, 1.5 o 2 bits de longitud para indicar el final de la emisión del carácter (S).

Este método también se denomina “start-stop”. Si se emplea paridad par, la suma de bits da un número par, si paridad impar la suma se adapta para dar un número impar.



Esta comunicación puede variar según la combinación de número de bits de datos, tipo de paridad y longitud de parada. El más común es el protocolo 8N1 (8 bits, no paridad, 1 bit de parada).

Protocolos orientados a bits.

El más usado es el HDLC (High Level Data Link Control) que es una norma establecida por ISO (International Organization for Standardization) y que ha sido utilizado como referencia para otros protocolos como Ethernet.

La estructura de datos básica del HDLC es la siguiente:

01111110	8 bits	8 bits	Cualquier long	16 bits	01111110
Comienzo	Dirección	Control	Datos	Cheksum	Fin

Los campos son predefinidos y de longitud fija excepto el de datos. La dirección es necesaria cuando se comporte el canal (varios posibles destinos), ya que indica la máquina destino. El byte de control indica el tipo de mensaje. El campo checksum permite verificar los datos (suma de control).

Protocolos orientados a bloques.

Los bytes se agrupan en bloques delimitados por caracteres especiales:

- SOH (cabecera, comienzo)
- SYN (sincronismo)
- STX (comienzo de texto)
- ETX (fin de texto)
- EOT (fin de transmisión)

01010101	SYN SYN...STX	Texto	ETX	BCC	SYN SYN ... EOT
SOH	Bloque de verificación				

Se utiliza para comunicaciones síncronas, por lo que la línea nunca está ociosa, siempre se intercambian mensajes de sincronización si no hay otra cosa que comunicar. La señal de sincronismo se transmite junto con los datos.

1.4.1.- Redes locales

Las redes locales (LAN's, Local Area Network) se utilizan para conectar varias localizaciones llamadas nodos, para que todos puedan conectarse entre sí. Para implantar un sistema de fabricación se pasa necesariamente por la introducción de una o varias LAN, lo que constituye un sistema de comunicaciones distribuido.

De todas las posibles disposiciones con que se relacionen los elementos de la red, surgirán diferentes topologías, es decir, configuraciones geográficas. Las topologías más comunes son:

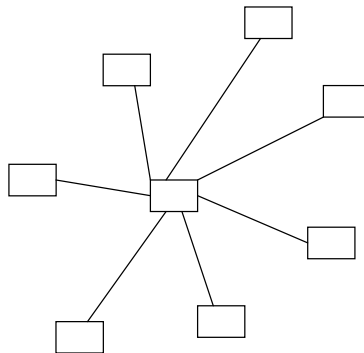
- [En estrella.](#)
- [En anillo.](#)
- [En Bus.](#)

Existen tres estándares principales de redes:

- CSMA/CD, conocida familiarmente como Ethernet.
- Token Bus
- Token Ring

Topología en estrella.

Tienen la siguiente estructura:

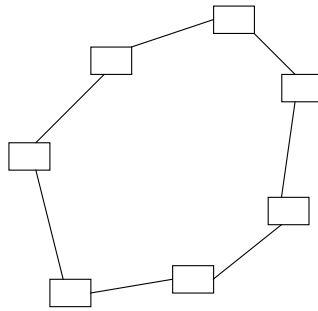


El nodo central se encarga de realizar todo el control, aunque se puede optar por que el control lo realice alguno de los nodos periféricos, quedando el nodo central con la única tarea de conmutación de información.

Tiene un repercusión baja en el resto de la red al producir algún fallo en algún elemento que no sea el central. Tiene una alta flexibilidad. No permite grandes flujos de tráfico por congestión del nodo. El coste de la instalación suele ser elevado, puesto que hay que instalar una línea directa desde cada nodo al nodo central.

Topología en anillo.

Tienen la siguiente estructura:

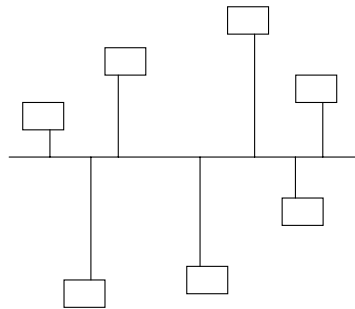


La información circula en un solo sentido hasta regresar a la estación de origen. Mediante esta disposición, cada estación podrá reconocer los mensajes a ella destinados tras comprobar la dirección de destino o bien actuar como transmisor de los mensajes destinados al resto de las estaciones. Por eso se dice que las estaciones juegan un papel activo en el control de la red.

Posee buena flexibilidad a la hora de introducir o eliminar estaciones en la red. Sin embargo, un fallo en el módulo de comunicaciones de cualquiera de las estaciones de la red dejará a esta bloqueada en su totalidad.

Topología en Bus.

Tienen la siguiente estructura:

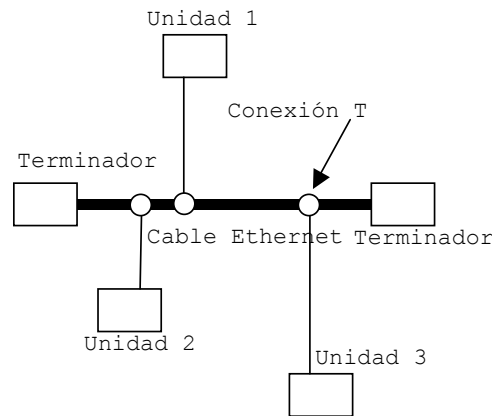


Todos los nodos de la red están conectados a un único canal de comunicación denominado bus. La misión de cada nodo ya no es la de actuar como repetidor, sino que únicamente debe reconocer los mensajes que van por el bus que estén destinados a su estación.

Entre sus ventajas está la sencillez de la instalación, además de presentar una gran flexibilidad al aumentar o disminuir el número de estaciones de la red, se dice que es escalable. Los nodos actúan de forma pasiva.

La red IEEE 802.3 (Ethernet).

Esta en el mercado desde 1980. Es muy usada tanto en aplicaciones industriales como de oficina. Es una topología de bus con conexiones a las ramas.



En el nivel físico consistes en un cable coaxial apantallado, al que los periféricos se conectan mediante transceivers. El cable puede ser grueso (unos 2.5 cm de diámetro) o delgado (0.5 cm), en cuyo caso se conoce como ThinWire Ethernet o Cheapernet.

ThinWire Ethernet utiliza cable coaxial de radiofrecuencia RG-58 y conectores BNC, no estando recomendadas distancias superiores a 500m. La velocidad máxima de transferencia es 10 Mbits/s.

Ethernet es un medio de banda base (no emplea modulación) y utiliza codificación Manchester. Para los niveles de enlace y acceso al medio, los paquetes de datos son similares al HDLC. Cada dispositivo decide cuando accede a la red y no decodifica mensajes que no vayan dirigidos a él.

Para la coordinación del acceso a la red, se utiliza un protocolo llamado CSMA/CD (Carrier-sensing Múltiple Access/Collision Detection), que detecta y maneja las colisiones. Una colisión ocurre cuando dos o más estaciones comienzan a transmitir datos al mismo tiempo sobre el mismo canal o cuando sus tiempos de transmisión se solapan.

Cada estación de transmisión comprueba primero si el canal está libre. Si dos estaciones quieren transmitir a la vez, los datos colisionan y ambas abortan el acceso al canal, poniendo además un aviso en la línea. Tras un corto periodo de tiempo particular para cada estación se vuelve a intentar el acceso. La más rápida podrá transmitir, y las demás esperarán.

La unidad que transmite, controla constantemente la calidad de la señal en la línea, escuchando lo que ella misma está enviando. La probabilidad de las colisiones se incrementa con el número de unidades y la longitud de la línea, y con ello el tiempo perdido en los intentos de transmisión. En la práctica no hay una cota superior del tiempo que puede llevar la transmisión de un mensaje, lo cual es una desventaja para aplicaciones industriales de tiempo real.

Esta es la razón que se hayan desarrollado redes Ethernet deterministas, es decir, capaces de gestionar prioridades en el caso de que colisionen dos mensajes y asegurar su transmisión en un plazo dado que sea compatible con el tiempo de respuesta de los equipos conectados. (Factor, de la empresa francesa Aptor).

1.4.2.- Redes de comunicación industriales.

A continuación se da un breve repaso a algunas de las redes industriales más importantes que el alumno debe conocer.

Estándares de interfaces serie.

Algunos dispositivos empleados en aplicaciones industriales cumplen estándares del EIA como RS-232, RS-422 o RS-485 para conectarse a computadores y unos con otros. Estos estándares definen solo las características eléctricas, no el protocolo software. La tabla siguiente resume las principales características de estas tres interfaces serie como se define en sus documentos estándares respectivos.

RS-232 es la conexión serie disponible en PCs. Se emplea para múltiples usos, como conectar un ratón, impresora, modem así como instrumentación industrial. A causa de diversas mejoras de cables y drivers, las aplicaciones mejoran el rendimiento más allá de la distancia y velocidad recogida en el protocolo.

RS-422 es la conexión serie empleada en el Apple Macintosh. Usa una señal eléctrica diferencial, opuesto a señales balanceadas respecto a masa como el RS-232. La transmisión diferencial, que necesita dos líneas para las señales de transmisión y otras dos para las de recepción, permiten una mayor inmunidad al ruido y mayores distancias comparado con RS-232, que son grandes ventajas en entorno industrial.

RS-485 es una mejora sobre RS-422 porque incrementa el número de dispositivos de 10 a 32, y define las características eléctricas necesarias para asegurar unos voltajes de señal adecuados bajo carga máxima. Con esta capacidad multipunto, se pueden crear redes de dispositivos conectados a un simple puerto serie RS-485. La inmunidad al ruido y capacidad multipunto hacen que RS-485 sea la conexión serie más empleada en aplicaciones industriales que tienen varios dispositivos distribuidos a un PC u otro controlador para la recolección de datos y otras operaciones.

	RS-232	RS-422	RS-485
Tipo de líneas de transmisión	No balanceada	Diferencial	Diferencial
Máximo número de emisores	1	1	32
Máximo número de receptores	1	10	32
Máxima longitud de cable en metros	15.2	1200	1200
Máxima velocidad	20 kb/s	10Mb/s	10 Mb/s

Protocolos Software

Los protocolos RS-XXX no especifican nada sobre el software empleado para comunicar dispositivos. Una cuestión sobre dispositivos RS-485 en particular es ¿cómo se direccionan los dispositivos?. La respuesta es que eso depende del dispositivo. RS-485 solo especifica que es capaz de conectar múltiples cargas eléctricas a un juego de hilos. No dice nada de cómo direccionar a estas cargas. Es necesario un protocolo que describa como direccionar un determinado dispositivo. Estos protocolos se implementan en su mayoría como protocolos asíncronos. Estas redes generalmente son maestro/esclavo, donde un dispositivo es el maestro y los demás esclavos. Normalmente, todos los dispositivos se encuentran recibiendo a la espera de un mensaje dirigido a ellos. Cuando reciben uno lo procesan, respondiendo con datos en la interfaz serie si es necesario.

Existen varios protocolos serie industriales que son más complejos que simples comandos en código ASCII, compatibles con RS-232 o RS-485. Estos protocolos se emplean principalmente con PLCs. Ejemplos de estos son ModBus y OptoMux.

Redes Industriales Abiertas.

Se conoce como red abierta la que no pertenece a un solo fabricante, si no que se desarrolla por un elemento normalizador (EIA, ISO) con el fin de hacer interoperativos dispositivos de diversos fabricantes. Son versiones sofisticadas de protocolos establecidos y tienen la ventaja de que diversos fabricantes ofrecen productos para estas redes. En la tabla siguiente se da un resumen de estas redes.

Red	Área de aplicación
CANopen, DeviceNet, SDS	Redes basadas en CAN para dispositivos de fabricación
Foundation Fieldbus	Red de control de procesos
Interbus-S	Red de bajo nivel para interconexión de dispositivos de E/S
LonWorks	Red de propósito general ampliamente usada en sistemas empotrados y automatización de edificios
PROFIBUS DP/FMS/PA	Familia de redes para fabricación y procesos

DeviceNet

Originalmente desarrollada por Allen-Bradley. DeviceNet es gestionada por Open DeviceNet Vendors Association (ODVA). Es una red de bajo nivel diseñada para conectar dispositivos industriales (sensores, actuadores) a dispositivos de mayor nivel (controladores). Se enfoca de forma específica en ser intercambiable a muy bajo costo como muchos dispositivos simples empleados en aplicaciones de fabricación, sensores, lectores de código de barras, etc.

Las características clave son: Buses de par trenzado separados para señal y alimentación. Inserción de dispositivos en caliente sin cortar la alimentación. Diseño optoaislado opcional. La tabla siguiente muestra las características eléctricas de la red.

Velocidad	Distancia	Longitud de entronque	
		Máxima	Acumulada
125 kb/s	500 m	6 m	156 m
250 kb/s	250 m	6 m	78 m
500 kb/s	100 m	6 m	39 m

FOUNDATION Fieldbus

Es una red industrial diseñada específicamente para aplicaciones de control de procesos de forma distribuida. Esta red fue creada por la fundación FIELDBUS, una organización con más de 100 compañías que forman más del 80 % de los fabricantes de sistemas de automatización.

La tabla siguiente muestra las características de la capa física (cableado, señales, etc.).

Velocidad de transmisión	31.25 kb/s	1.0 o 2.5 Mb/s
Topología	Bus / Árbol	Bus
Alimentación en el bus	Si	No (excepto 1 Mb/s)
Segura	Si	No (excepto 1 Mb/s)
Número de dispositivos	2-32	2-32
Longitud de cable	1900 m	750 m
Longitud de entroncado	120m	Ninguna

Profibus

Constituye el sistema más avanzado utilizado en Europa, empleado principalmente en procesos de fabricación y automatización de edificios. Se encuentra estandarizado en DIN 19245 y EN 50170. Esta tecnología se administra por la organización de usuarios PROFIBUS con más de 600 fabricantes miembros.

Fue diseñada para cumplir varios requerimientos. Puede usarse para transmisiones de datos críticos en tiempo real de alta velocidad entre controladores y E/S y para comunicaciones complejas entre controladores programables. La red consiste en tres versiones compatibles DP, FMS y PA.

PROFIBUS-DP se diseñó para comunicaciones de alta velocidad entre controladores industriales y E/S distribuida. En una red PROFIBUS, los controladores centrales, PLCs o PCs se comunican con dispositivos distribuidos (E/S, válvulas) a través de un enlace serie de alta velocidad. La mayoría de las comunicaciones tiene lugar de una manera cíclica.

PROFIBUS-FMS se diseñó como comunicación de propósito general entre controladores programables, como PLCs y PCs. FMS contiene una capa de aplicación con servicios de comunicaciones que hace posible acceder variables, transmitir programas y controlar la ejecución del programa así como transmitir eventos. Esto hace posible la ejecución de forma distribuida de procesos complejos. La capa física implementa seguridad en la información y control de acceso y se suele realizar con RS-485 o fibra óptica.

PROFIBUS-PA se diseñó específicamente para la automatización de procesos, usando la capa física proporcionada por FIELDBUS para sensores tele-alimentados (por el bus de comunicación) para que operarán en áreas de seguridad. PROFIBUS-PA usa el protocolo extendido de PROFIBUS-DP para transmisión de datos.

RS-485 se emplea como capa física para estas aplicaciones, permitiendo una tasa de transmisión de 9,6 kb/s a 12 Mb/s. Una sola velocidad de transmisión se selecciona para toda la red cuando el sistema se pone en marcha. Se pueden conectar hasta 32 estaciones a cada segmento sin repetidores y hasta 127 con repetidores. La longitud máxima de cable depende de la velocidad de transmisión como se muestra en la siguiente tabla.

Tasa de transmisión (kb/s)	9.6	19.2	93.75	187.5	500	1,500	12,000
Distancia/segmento (m)	1200	1200	1200	1000	400	200	100