

# **Microprocesadores y microcontroladores**

**3**

***Fichas 1 y 2***



**serie/desarrollo de contenidos**  
**colección/fluídica y controladores lógicos programables**

## ***Autoridades***

---

### **Presidente de la Nación**

Néstor C. Kirchner

### **Ministro de Educación, Ciencia y Tecnología**

Daniel Filmus

### **Directora Ejecutiva del Instituto Nacional de Educación Tecnológica**

María Rosa Almandoz

### **Director Nacional del Centro Nacional de Educación Tecnológica**

Juan Manuel Kirschenbaum

## ***Especialista en contenidos***

- Marcelo Estévez

## **serie/desarrollo de contenidos**

---

### **Colecciones**

- Autotrónica
- Comunicación de señales y datos
- Diseño gráfico industrial
- Electrónica y sistemas de control
- Fluidica y controladores lógicos programables
  - 1. Tecnología neumática
  - 2. Controladores lógicos programables –PLC–
  - 3. Microprocesadores y microcontroladores
- Gestión de la calidad
- Gestión de las organizaciones
- Informática
- Invernadero computarizado
- Laboratorio interactivo de idiomas
- Procesos de producción integrada
- Proyecto tecnológico
- Unidades de cultura tecnológica



## Índice

---

El Centro Nacional de Educación Tecnológica	7
¿De qué se ocupa <i>Microprocesadores y microcontroladores</i> ?	
• El problema tecnológico	12
• Las primeras decisiones	13
Ficha 1. Introducción a los sistemas basados en microprocesador	
• Referencia histórica	23
• Sistema mínimo microprocesador	29
• Arquitecturas básicas de microprocesadores y microcontroladores. Harvard versus Von Neumann	39
Ficha 2. Introducción a los microcontroladores	
• Controlador y microcontrolador	48
• Microprocesador y microcontrolador	49
• Aplicaciones de los microcontroladores	50
• ¿Qué microcontrolador emplear?	51
• Almacenamiento y ejecución del programa	55
• Recursos comunes a todos los microcontroladores	56
• Recursos especiales	62
• Herramientas para el desarrollo de aplicaciones	66
• Ejemplos de microcontroladores y aplicaciones	67
Ficha 3. Microcontroladores más utilizados	
• Motorola 68HC908 (68HC908KX8)	79
• Intel 8051 (ATMEL AT89S8252)	83
• Microchip PIC 16F84	89
Ficha 4. Programación de microcontroladores	
• Registros del microcontrolador	109
• Lenguaje assembler	112
• Estructura de un programa en assembler	122
• Desarrollo de un programa en assembler	128
• Archivo de código objeto	134

## Ficha 5. Set de instrucciones

- Modos de direccionamiento 145
- Clasificación de las instrucciones 161
  1. Instrucciones de movimiento de datos 161
  2. Instrucciones aritméticas 164
  3. Instrucciones lógicas 168
  4. Instrucciones de manipulación de bits 169
  5. Instrucciones de manipulación de datos 169
  6. Instrucciones de control del programa 170
  7. Instrucciones de operaciones BCD 173
  8. Instrucciones especiales 173

## Ficha 6. Procesando excepciones

- Reset e interrupciones 177
- Vinculación con el mundo exterior 182
- Volviendo a nuestro problema 184

## Anexos

- Sistemas de numeración 199
- Representación de la información 201
- Set de instrucciones de la familia 68HC08 208
- Set de instrucciones PIC 16xxx 217
- Bibliografía 218

## ***El Centro Nacional de Educación Tecnológica***

**Generar valor con equidad  
en la sociedad del conocimiento.**

La misión del Centro Nacional de Educación Tecnológica –CeNET– comprende el diseño, el desarrollo y la implementación de proyectos innovadores en el área de la educación tecnológica y de la educación técnico profesional, que vinculan la formación con el mundo del trabajo.

Acorde con esta misión, el CeNET tiene como propósitos los de:

- Constituirse en referente nacional del Sistema de Educación Tecnológica, sobre la base de la excelencia de sus prestaciones y de su gestión.
- Ser un ámbito de capacitación, adopción, adaptación y desarrollo de metodología para la generación de capacidades estratégicas en el campo de la Educación Tecnológica.
- Coordinar, mediante una red, un Sistema de Educación Tecnológica.
- Favorecer el desarrollo de las pequeñas y medianas empresas, a través del sistema educativo.
- Capacitar en el uso de tecnologías a docentes, jóvenes, adultos, personas de la tercera edad, profesionales, técnicos y estudiantes.
- Brindar asistencia técnica.
- Articular recursos asociativos, integrando los actores sociales interesados en el desarrollo del Sistema de Educación Tecnológica.

Desde el CeNET venimos trabajando, así, en distintas líneas de acción que convergen en el objetivo de reunir a profesores, a especialistas en Tecnología y a representantes de la industria y de la empresa, en acciones compartidas que permitan que la Educación Tecnológica se desarrolle en la escuela de un modo sistemático, enriquecedor, profundo... auténticamente formativo, tanto para los alumnos como para los docentes.

Una de nuestras líneas de acción es la de **diseñar, implementar y difundir trayectos de capacitación y de actualización**. En CeNET contamos con quince unidades de gestión de aprendizaje en las que se desarrollan cursos, talleres, pasantías, encuentros, destinados a cada educador y a cada miembro de la comunidad que desee integrarse en ellos:

- Autotrónica.
- Centro multimedial de recursos educativos.
- Comunicación de señales y datos.
- Cultura tecnológica.
- Diseño gráfico industrial.
- Electrónica y sistemas de control.
- Flúidica y controladores lógicos programables.
- Gestión de la calidad.
- Gestión de las organizaciones.
- Informática.
- Invernadero computarizado.
- Laboratorio interactivo de idiomas.
- Procesos de producción integrada. CIM.
- Proyecto tecnológico.
- Simulación por computadora.

Otra de nuestras líneas de trabajo asume la responsabilidad de **generar y participar en redes** que integren al Centro con organismos e instituciones educativos ocupados en la Educación Tecnológica, y con organismos, instituciones y empresas dedicados a la tecnología en general. Entre estas redes, se encuentra la que conecta a CeNET con los Centros Regionales de Educación Tecnológica –CeRET– y con las Unidades de Cultura Tecnológica instalados en todo el país.

También nos ocupa la tarea de **producir materiales didácticos**. Desde CeNET hemos desarrollado cinco series de publicaciones:

- *Educación Tecnológica*, que abarca materiales (uni y multimedia) que buscan posibilitar al destinatario una definición curricular del área de la Tecnología en el ámbito escolar y que incluye marcos teóricos generales, de referencia, acerca del área en su conjunto y de sus contenidos, enfoques, procedimientos y estrategias didácticas más generales.
- *Desarrollo de contenidos*, nuestra segunda serie de publicaciones, que nuclea fascículos de capacitación que pueden permitir una profundización en los campos de problemas y de contenidos de las distintas áreas del conocimiento tecnológico (los quince ámbitos que puntualizábamos y otros que se les vayan sumando) y que recopila, también, experiencias de capacitación docente desarrolladas en cada una de estas áreas.
- *Educación con tecnologías*, que propicia el uso de las nuevas tecnologías de la información y de la comunicación como recursos didácticos, en las clases de todas las áreas y espacios curriculares.
- *Educadores en tecnología*, que focaliza el análisis y las propuestas en uno de los constituyentes del proceso didáctico: el profesional que enseña Tecnología, ahondando en los rasgos de formación, de sus prácticas, de sus procesos de capacitación, de su vinculación con los lineamientos curriculares y con las políticas educativas, de interactividad con sus alumnos, y con sus propios saberes y modos de hacer.
- *Documentos de la escuela técnica*, que difunde los marcos normativos y curriculares que desde el CONET –Consejo Nacional de Educación Técnica– delinearon la educación técnica de nuestro país.

A partir de estas líneas de trabajo, el CeNET intenta constituirse en un ámbito en el que las escuelas, los docentes, los representantes de los sistemas técnico y científico, y las empresas puedan desarrollar proyectos innovadores que redunden en mejoras para la enseñanza y el aprendizaje de la Tecnología.

Buenos Aires, febrero de 2004



## ¿De qué se ocupa Microprocesadores y microcontroladores?

A mediados del siglo XX, tras el desarrollo de la bomba atómica y las centrales nucleares, comienza a hablarse de la **Era atómica**. Algo más adelante, cuando el hombre pisa por primera vez la Luna, se habla de la **Era espacial**. Sin ninguna duda, nuestra época está siendo la **Era de las comunicaciones** debido, fundamentalmente, al desarrollo de la informática y las telecomunicaciones, sustentadas por el enorme avance de las tecnologías electrónicas, microelectrónicas y optoelectrónicas.



Para confirmarlo, sólo meditemos sobre el avance de la telefonía celular y satelital, sobre el enorme potencial de información que nos pone al alcance de la mano Internet, así como sobre la posibilidad de comunicarnos con otros países por escrito o videoconferencia, dependiendo cada vez más del omnipotente computador personal, que incide más y más en nuestra sociedad (automatización de plantas industriales, control de tráfico por tierra, mar y aire, gestión de empresas y bancos, diseño de ingeniería, etc.).

La electrónica ha sido –y es– la causante y soporte de esta verdadera revolución de la sociedad. Y, dentro del campo de la electrónica, el microprocesador –o su versión “customizada”, el microcontrolador– es, con frecuencia, el gran protagonista.

### Microprocesador

Componente electrónico que contiene, en una única pastilla, el procesador (CPU –*Central Process Unit* o unidad central de proceso–) de una máquina programable de tratamiento de información.

El desarrollo de los microprocesadores, un invento introducido recientemente –en 1971– en el ámbito comercial, ha revolucionado las áreas más importantes de la electrónica y este avance no le resulta ajeno al individuo medio que trabaja asiduamente con su computador personal (Pentium, Power PC, etc.), aún cuando la mayor parte de los microprocesadores no reside en los computadores, sino oculta en cualquier dispositivo electrónico que el usuario utiliza sin ser consciente de él.

En 1971, la compañía de semiconductores Intel lanza al mercado el primer **microprocesador**, lo que impone un cambio decisivo en las técnicas de diseño de los equipos de instrumentación y control. Este circuito integrado contiene todos los componentes de la unidad central de procesos (CPU) de una computadora dentro de un solo dispositivo. Los fabricantes, conscientes de la importancia de este mercado, crean una amplia gama de estos circuitos integrados, constituyendo familias de microprocesadores.

En el año 1976, gracias al aumento de la densidad de integración de componentes en un circuito integrado, sale a la luz el primer ordenador en un chip: en él se integran, junto con el microprocesador, otros subsistemas que anteriormente formaban unidades independientes (memoria, entradas/salidas, etc.). A este nuevo integrado se lo denomina **microcomputadora monopastilla**.

Cuando los sistemas basados en microprocesadores se especializan en aplicaciones industriales, aparece la versión industrial de la microcomputadora monopastilla, el

**microcontrolador**, que produce un obvio beneficio en aplicaciones pequeñas. Su característica más sobresaliente es que se trata de un sistema integrado.

El microprocesador es un componente electrónico complejo que incorpora las funciones típicas de todo un computador. La clave del éxito de los microprocesadores como componente electrónico reside en que, modificando el programa almacenado en memoria, puede adaptarse a numerosas y diferentes aplicaciones.

Así, en la actualidad, debido a su creciente potencia de cálculo y variedad de funciones integradas, se ha convertido en el estándar para la implementación de soluciones, tanto de cálculo como de control.

De este modo, los microprocesadores no sólo se utilizan para construir potentes computadoras –quizás su aplicación más difundida– sino también para la realización de tareas de control en sistemas de aplicación específica (por ejemplo, el control de un horno a microondas, un teléfono celular o unos frenos ABS para el automóvil), sustituyendo la circuitería convencional.

Las áreas de aplicación de este tipo de productos también exceden el ambiente de los procesos continuos y se expanden en una gran diversidad de rubros: cervecerías, curtiembres, edificios inteligentes, bancos de prueba o ensayo, estaciones de carga, frigoríficos, plantas automotrices, industrias alimenticias, industrias madereras, industrias metalúrgicas, industrias petroleras, laboratorios, medición de energía, plantas petroquímicas, plantas aceiteras, plantas embotelladoras, plantas generadoras de energía, plantas papeleras, plantas pesqueras, plantas potabilizadoras, silos, subestaciones eléctricas y centros de comando, supermercados, sistemas de transporte y muchas más.

En nuestro curso estudiaremos al microprocesador desde dos puntos de vista:

- Como componente electrónico –se trata de un componente especial, ya que tiene un aspecto hardware, y otro software o de programación–.
- Como ladrillo básico en la construcción de sistemas de procesamiento de datos –**microprocesador**, **mP**, **μP** – o de control –**microcontrolador**, **mC**, **μC**– que sustituye a sistemas cableados.

En las aplicaciones sencillas, resultan preferibles las soluciones no programables que no requieren desarrollo de software (Escribir software consume mucho tiempo, por lo que resulta más costoso; por esto, en aplicaciones sencillas y/o de poca tirada, a menudo es más razonable efectuar tareas en hardware.) Sin embargo, conforme aumenta la complejidad del sistema, aumentan las ventajas del uso de sistemas programables.

Una de las principales ventajas de los sistemas programables es su flexibilidad, la que permite actualizar el funcionamiento de un sistema tan sólo mediante el cambio del programa, sin tener que volver a diseñar el hardware. Esta flexibilidad es muy importante, al permitir que los productos se actualicen con facilidad y economía.

Un **μC** típico tiene instrucciones para:

- transferir información entre registros y memoria,
- realizar operaciones aritméticas y lógicas,

### Programa

Es una lista de instrucciones al procesador. Todos los **μP** y **μC** tienen un conjunto de instrucciones que pueden ejecutar. Cada tipo de procesador y controlador tiene su propio conjunto de instrucciones y, por lo general, los programas escritos para una máquina no funcionan en otra.

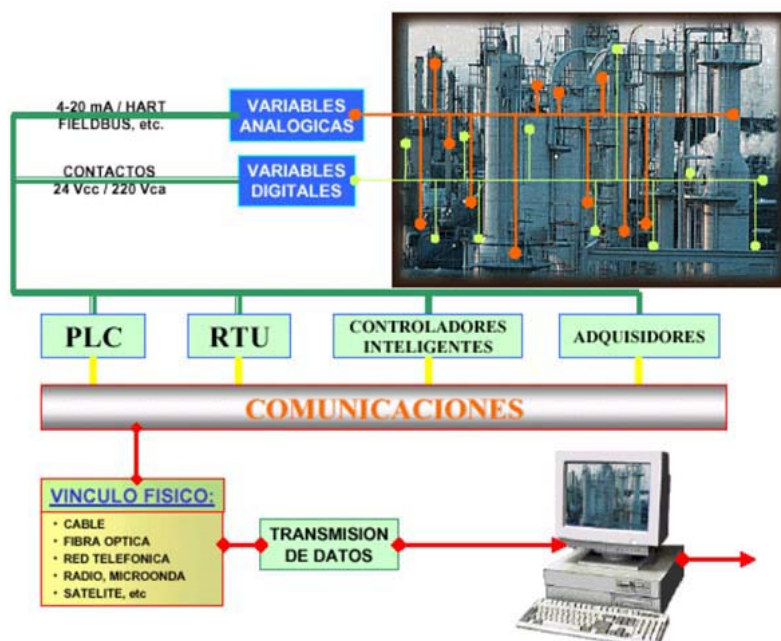
- efectuar comparaciones y pruebas sobre el contenido de sus registros de memoria,
- controlar la secuencia de ejecución de programas,
- controlar entradas y salidas.

La gran oferta de microcontroladores y las muy similares prestaciones que todos ellos ofrecen, han desviado el foco de atención de los potenciales usuarios hacia cuestiones que poco tienen que ver con la funcionalidad misma del producto. Se asume que cualquier microcontrolador que esté disponible en el mercado es bueno y relativamente fácil de usar. En el pasado, un producto debía demostrar que era capaz de hacer determinada tarea; hoy esto se da por descontado y, por lo tanto, lo que pesa en la decisión de utilizar un determinado producto es su precio, el apoyo y soporte local inmediato, los antecedentes de aplicaciones instaladas, la capacidad de conectarse con el equipamiento de campo que se utiliza en el país donde se comercializa, etc.

Podemos decir que los microcontroladores tienden, con el tiempo, a ser todos iguales. Van incorporando prestaciones similares modeladas por un *set* de instrucciones cada vez más parecido, que afecta a casi todos los fabricantes de microprocesadores y microcontroladores.

Otra ventaja interesante que ofrece la utilización de microcontroladores –sobre todo para quien desarrolla aplicaciones basadas en ellos– es que permite obtener mayores ganancias como resultado del valor que agrega al hacer la tarea de integración o de “armado” (En otro ámbito, todos sabemos que es mucho más barato comprar una PC en partes sueltas y luego armarla, que ir a comprarla a un negocio en donde nos la entregan funcionando, con todo instalado y probado. Lo mismo pasa con los mC.). Sin duda, para un usuario final con muchos recursos puede ser más interesante evitarse problemas y comprar un producto “llave en mano”; pero, para el integrador o el usuario que recién comienza, un alto costo inicial puede ser una barrera para decidirse a comenzar.

En la figura vemos equipos de control industrial; en cada uno de ellos existe un microprocesador o microcontrolador para llevar a cabo las tareas de supervisión y control, o para comunicarse entre ellos y dar información a un computador central:



*Los equipos reciben datos de los sensores y los transmiten por algún medio a la PC*

Se trata de sistemas de control clásicos donde se pone de manifiesto el empleo intensivo de los microprocesadores y microcontroladores en la industria.

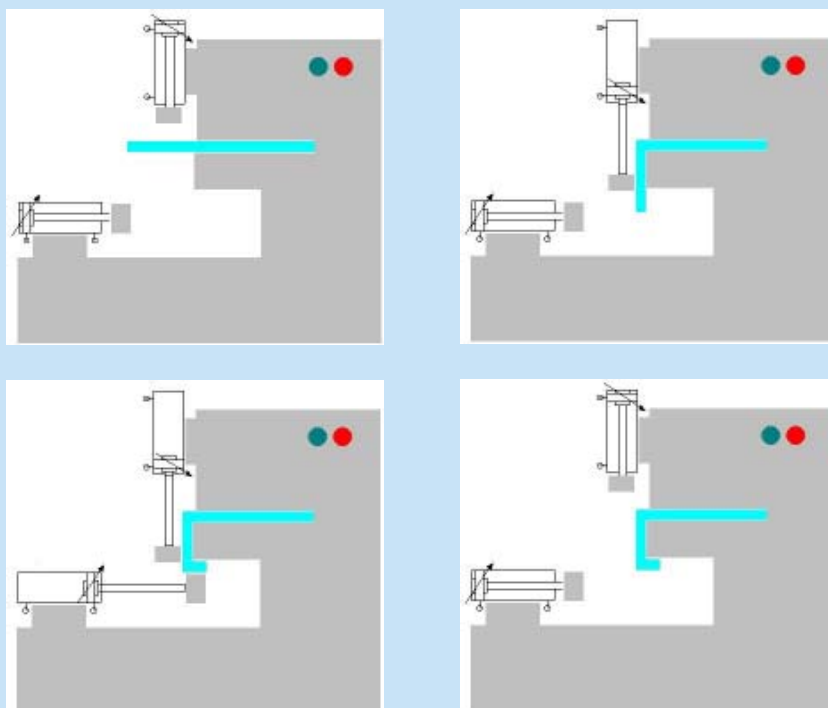
### ***El problema tecnológico***

Este problema nos acompañará a lo largo del cursado de **Microprocesadores y microcontroladores**:

Nos han pedido realizar el estudio de un dispositivo que realice un conformado de un perfil plegándolo; sus dimensiones son 100 mm x 300 mm x 3 mm.

Algunas de las condiciones planteadas son que el ciclo debe ser automático, de fácil manejo, seguro y confiable; y, además, disponemos de los siguientes datos:

- el material de la pieza es aluminio,
- la fuerza necesaria para realizar el plegado es de, aproximadamente, 1000 N.



*Perfil sin plegar y plegado. Fases o etapas de trabajo de nuestra máquina o sistema, para plegar el perfil.*

Naturalmente, existen diferentes soluciones para esta situación.

La elección del sistema adecuado, en la práctica, no siempre resulta fácil y bien definida. A esto debemos agregarle que, en función de la propia formación, el electricista propondrá una solución *eléctrica*; el especialista en hidráulica, una solución *hidráulica*; y el especialista en neumática pretenderá una solución *neumática*. La solución óptima de un problema exige el conocimiento de todas las alternativas que se ofrecen.

Para esto, le proponemos que...

### Actividad

Analice las posibles soluciones para los elementos de trabajo y de mando más usuales, los criterios de selección, así como sus ventajas y desventajas. Para esto, refiérase al sistema completo, empezando por las señales de entrada (emisores de señal), pasando por la parte de mando (procesadores), llegando hasta los órganos de maniobra y actuadores.

Además, tome en cuenta los siguientes factores:

- Medios de control preferidos.
- Equipos ya instalados.
- Conocimientos técnicos disponibles.
- Sistemas ya instalados: neumática, hidráulica, electricidad, electrónica.

Como criterios de selección:

- Capacidad de regulación.
- Seguridad.
- Influencias ambientales.
- Costo.
- Facilidad de manejo, reprogramabilidad, etc.

Y, como criterios para la elección del sistema:

- Fiabilidad de los elementos.
- Facilidad de conmutación de los elementos (tiempo de conmutación).
- Velocidad de transmisión de la señal.
- Distancias máximas.
- Dimensiones necesarias.
- Tratamiento principal de la señal.

Aquí también puede analizar cómo se comporta el sistema frente a las influencias ambientales, qué formación requiere el personal destinado al servicio y mantenimiento, o bien si ya se dispone de este personal.

Nuestra propuesta es que intente llegar a alguna conclusión usted mismo –aún cuando no le hemos presentado aún información conceptual–, ya sea a través de búsquedas bibliográficas o a través de Internet (si dispone del recurso), de catálogos, del análisis de dispositivos con los que se encuentra en su vida cotidiana, de la integración de conocimientos previos, solicitando información en empresas o industrias que trabajen con alguna de estas tecnologías.

●  
A lo largo de **Microprocesadores y microcontroladores** iremos sugiriéndole que compare sus respuestas iniciales con la información que le acercaremos.

### Las primeras decisiones

Viendo las características del problema planteado surgen, por lo menos, cuatro alternativas de solución con sus ventajas y desventajas:

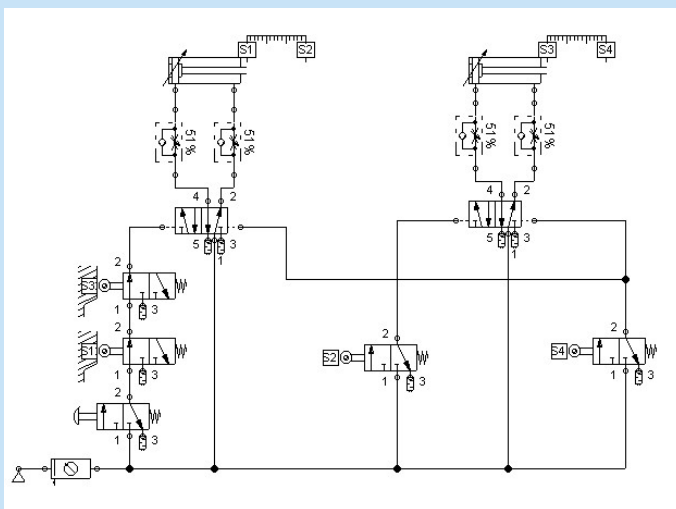
1. Neumática básica
2. Electroneumática
3. PLC –controladores lógicos programables–
4. Microcontroladores

Si algunos conceptos planteados en esta síntesis inicial no le resultan del todo claros, no se preocupe: iremos explicándolos pormenorizadamente a lo largo del texto.

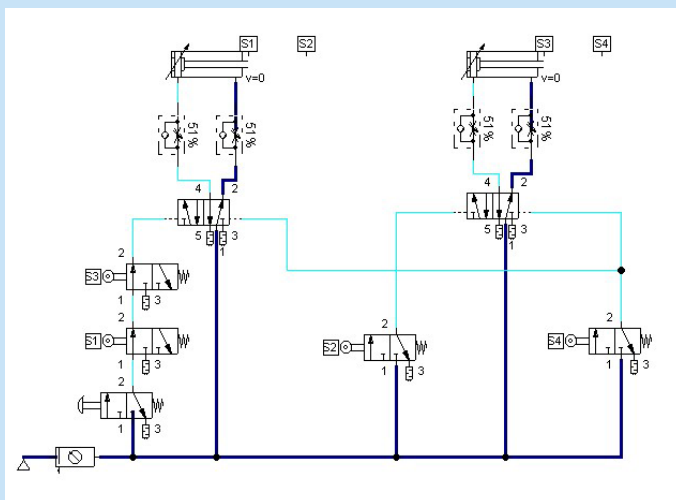
La **alternativa 1, neumática básica**, presenta una solución mecánicamente robusta, con un costo relativamente alto por la cantidad de válvulas neumáticas para el control del sistema y sensado de posiciones. Respecto al mantenimiento, las válvulas y la preparación del aire comprimido lo requieren. Cada cambio en el sistema representa una modificación mecánica de las conexiones.

### Solución neumática

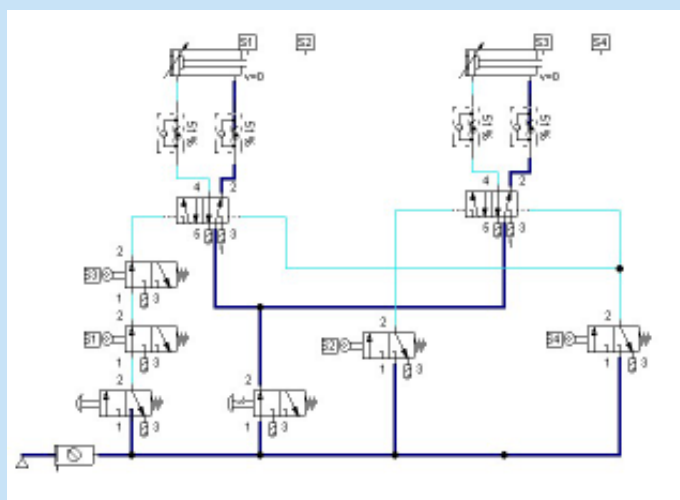
El ciclo es automático, luego de presionar el pulsador de marcha:



Veámoslo presurizado:



En el próximo esquema mostramos el mismo circuito con el agregado de una parada de emergencia. Presionando sobre ella se despresuriza el sistema y queda detenida la máquina en forma instantánea:

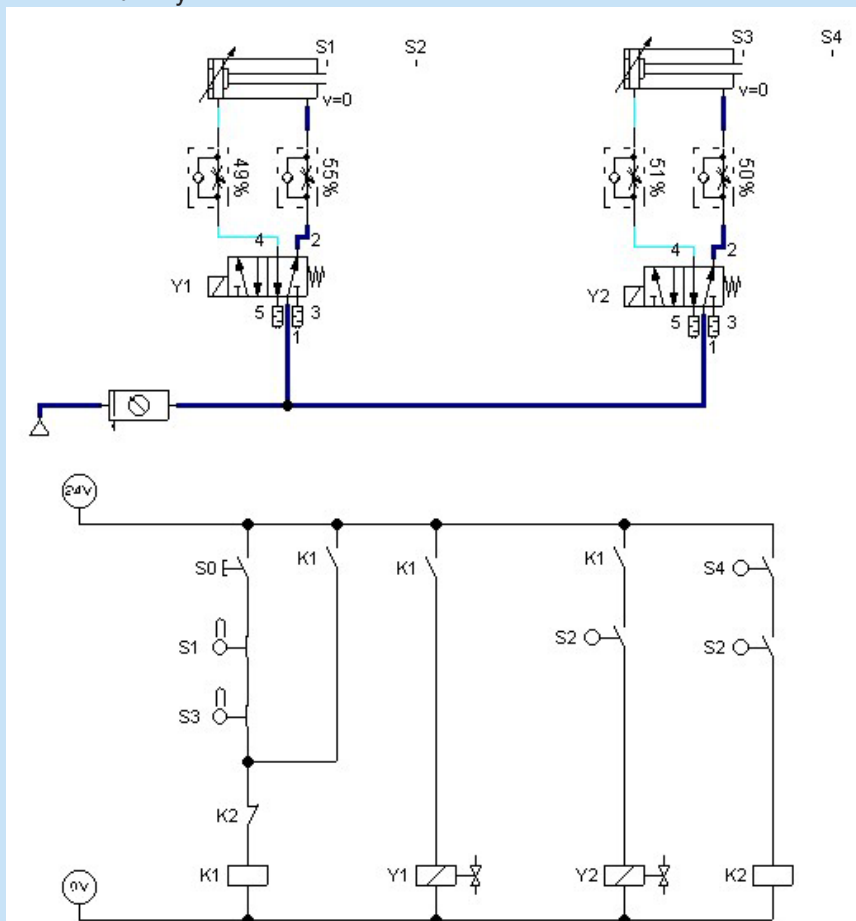


La **alternativa 2, electroneumática**, presenta una solución básicamente eléctrica, reemplazando a las válvulas de lógica por relé. Pero, también en esta solución, el cambio en el sistema representa una modificación mecánica de las conexiones cuyo costo de desarrollo es elevado.

### Solución electroneumática

El ciclo es automático luego de presionar el pulsador de marcha que, en este caso, es eléctrico.

Circuito neumático y eléctrico:





En este circuito se utilizaron dos relés K1 y K2 que sirven de memoria y de reset al sistema.

S0 es el pulsador de marcha y S1, S2, S3, y S4 son los fines de carrera eléctricos de los cilindros.

La **alternativa 3, PLC –controladores lógicos programables–**, presenta una solución muy integrada, reemplazando a las válvulas de lógica o los relé por un dispositivo llamado PLC que permite la actualización del sistema, el mejoramiento de las secuencias y las modificaciones, sin necesidad de cambiar circuitalmente la máquina o sistema. Por otra parte, la necesidad del mantenimiento –en cuanto a desgaste y posibles falsos contactos– es mínima, puesto que es un dispositivo electrónico cuyo costo es competitivo a partir de la utilización de más de 3 o 4 válvulas o relés en una automatización (Dicho PLC representa, en su interior, un sistema con una cantidad importante de relés, memorias, timers, contadores, etc.).

### Solución con PLC

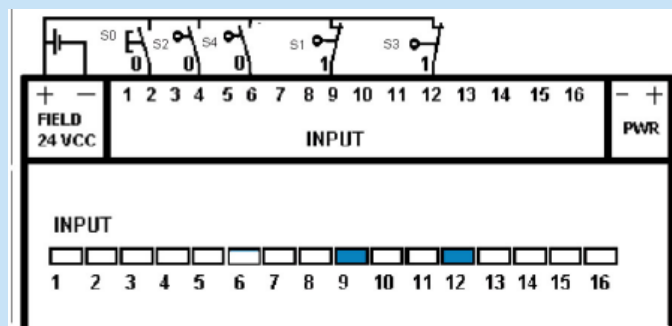
Un PLC no es más que un dispositivo electrónico con capacidad de procesar información que se ingresó en él en forma de un programa, y que puede manejar un número limitado de entradas y salidas –la parte visible, ya que en su estructura podemos diferenciar claramente borneras y conectores de comunicación, separados de acuerdo a sus funciones–.



En el caso de un PLC, éste posee entradas para conectar directamente los finales de carrera y el pulsador de marcha o parada a dichos bornes, de modo que su conexión es simple, vinculándola a una fuente llamada de campo o independiente de la de su propia CPU.

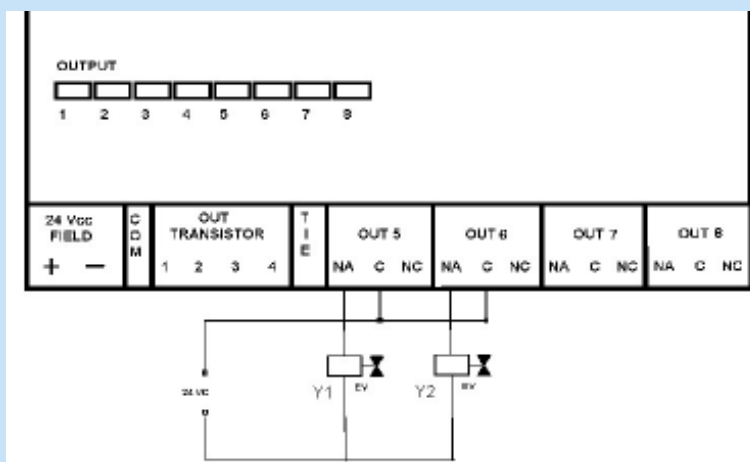
No es necesario conectar contactos NA y NC ya que se pueden leer invertidos en el PLC. Así, es habitual conectar todos los módulos *normal abiertos*.

Veamos un esquema de su conexión eléctrica de entradas:





Ahora, consideremos un esquema de su conexión eléctrica de salidas (En esta oportunidad, utilizamos salidas relé; pero, el PLC de este ejemplo posee también salidas transistorizadas que podrían haberse utilizado sin inconvenientes. Las salidas pueden ser utilizadas con la fuente de campo del PLC o utilizar otra fuente –como se indica en el esquema–, incluso de otra tensión o característica –como es la de corriente alterna–):



Al igual que en el microcontrolador, para vincular las entradas y salidas, y lograr que éstas respondan al problema señalado, el PLC debe ser programado en un lenguaje que interprete su sistema operativo (generalmente, en lista de instrucciones –awl– o diagrama de contactos –ladder–. También es posible en diagrama de funciones –esquemas de compuertas lógicas–)

Veamos cómo es el programa del PLC en lista de instrucciones:

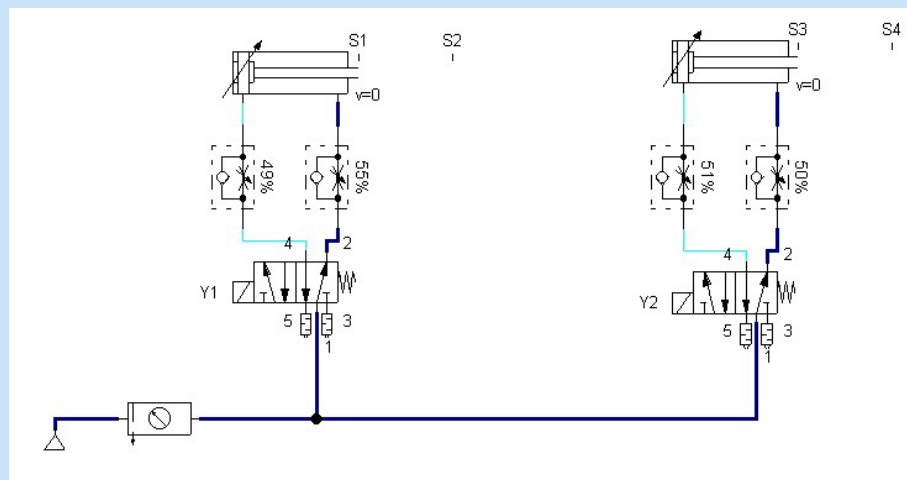
```

LD      IN    02
AND     IN    09
AND     IN    12
ANDN    MEM   01
=SET    OUT   05
=SET    MEM   01
=SET    MEM   02
LD      MEM   02
AND     IN    04
=SET    OUT   06
=SET    MEM   03
=RST    MEM   02
LD      MEM   03
AND     IN    06
=RST    OUT   05
=RST    OUT   06
=RST    MEM   01
=RST    MEM   03
PE

```

Este programa fue desarrollado con la técnica de cascada de flag o de memorias, y está pensado para todas las entradas con módulos normal abiertos –todos los fines de carrera son iguales y normales abiertos, si bien –como los cilindros están en reposo– en posición retraída se encuentran presionados S1 y S3–.

Lógicamente, el circuito neumático es equivalente a la solución electroneumática:

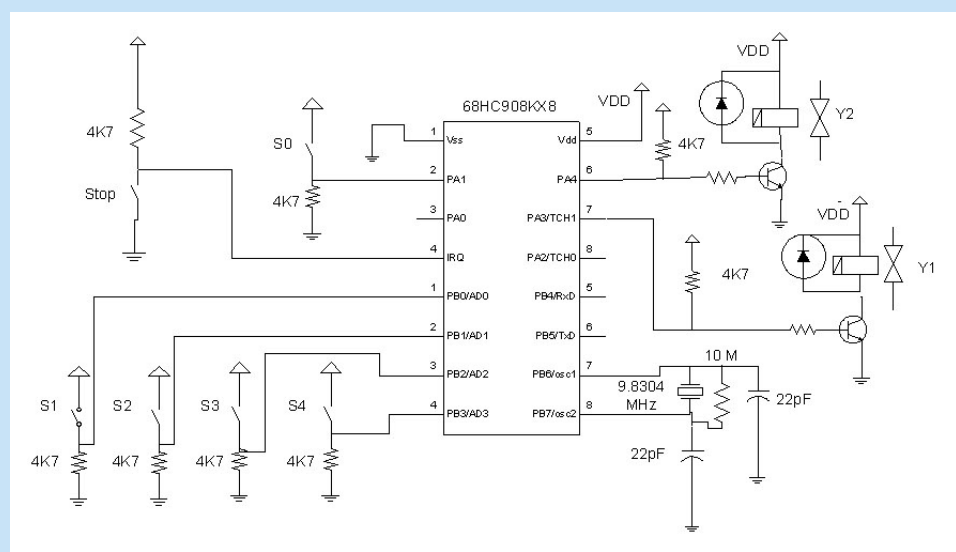


La **alternativa 4, microcontroladores**, presenta una solución completamente integrada electrónicamente, reemplazando a las válvulas de lógica o los relé por un dispositivo llamado microcontrolador que no es ni más ni menos que lo que tienen los PLC –aunque algunos de ellos poseen microprocesadores más potentes, de acuerdo a la aplicación a comandar–, que permite la actualización del sistema, el mejoramiento de las secuencias y todas las modificaciones, sin necesidad de cambiar circuitalmente la máquina o sistema. La necesidad del mantenimiento en cuanto a desgaste y posibles falsos contactos es mínima, ya que es un dispositivo electrónico de costo altamente competitivo: representa íntegramente la solución a múltiples situaciones, teniendo como contrapartida la necesidad de un especialista para su programación y vinculación con sistemas de entrada y salida (Requiere, también, la realización de una placa de circuito impreso para hacer de contenedor de los elementos por los cuales llevará las señales).

### Solución con microcontrolador

El ciclo es automático, luego de presionar el pulsador de marcha –que es un elemento eléctrico S0–.

Circuito electrónico:

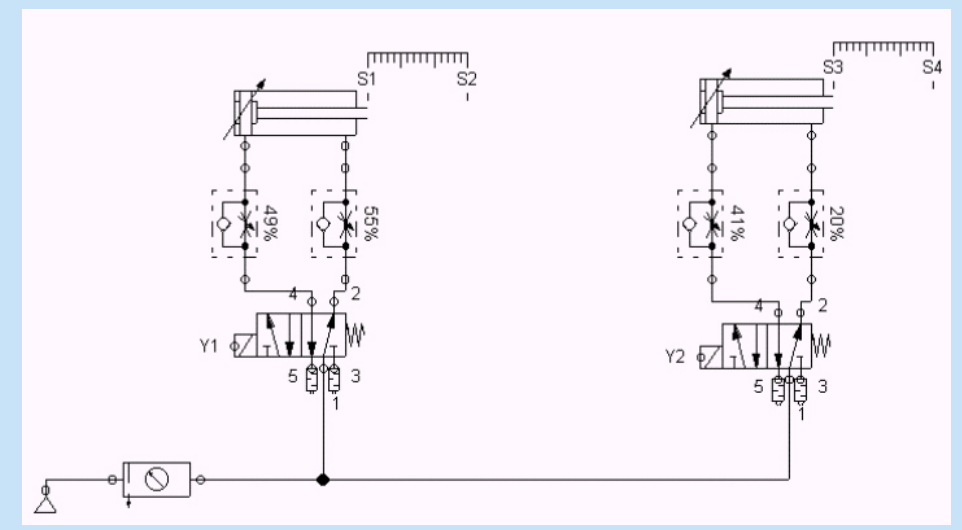


La entrada Stop, de ser programada, como está conectada a una interrupción, permite detener inmediatamente al sistema y esperar a que se restablezcan las condiciones para continuar.

S1, S2, S3, y S4 son fines de carrera eléctricos de los cilindros.

Y1 e Y2 son las bobinas de las electroválvulas que comandan los cilindros.

Circuito neumático:



Plantear una alternativa con componentes de hidráulica no parece ser una opción válida, por las fuerzas necesarias a desarrollar, ya que son de un bajo rango, fáciles de obtener con elementos de neumática. Asimismo, la implementación de soluciones con hidráulica representa un costo mayor.

Balanceando las ventajas y desventajas, y pensando que el costo final representaría el indicador más decisivo a la hora de volcarnos por una alternativa, es que consideramos la mejor solución desde el punto de vista económico a la **automatización con microcontroladores**, cuyos costos principales son: válvulas finales de mando, cilindros de doble efecto, reguladores de caudal, pulsadores de marcha, parada, FRL (hasta acá son los mismos para cualquiera de las soluciones), diferenciándose en aspectos como: placa de circuito impreso, componentes electrónicos –frente a una gran cantidad de válvulas o relés de costos elevados o frente a un PLC–.



## **FICHA 1**

---

### ***Introducción a los sistemas basados en microprocesador***



**Lógica cableada**

Corresponde a circuitos diseñados exclusivamente para realizar una aplicación concreta.

**Lógica programada**

Corresponde a un concepto más amplio que da lugar a las máquinas universales programadas. Se trata de máquinas potentes y flexibles capaces de realizar la mayoría de las funciones básicas necesarias para cualquier aplicación, que trabajan de forma secuencial, ejecutando progresivamente las operaciones necesarias.

**Referencia histórica**

Decíamos que en 1971, la compañía de semiconductores Intel lanza al mercado el primer microprocesador. Veamos algunos datos que nos muestran cómo va desplegándose el proceso tecnológico que permite llegar a estos complejos sistemas electrónicos:

**3000 A.C.** Ábaco, empleado para realizar cálculos aritméticos.

**Edad Media.** Autómatas de reloj: máquinas con un programa interior fijo.

**Siglo XVII.** Blaise Pascal, matemático francés, construye la primera máquina mecánica capaz de sumar y restar.

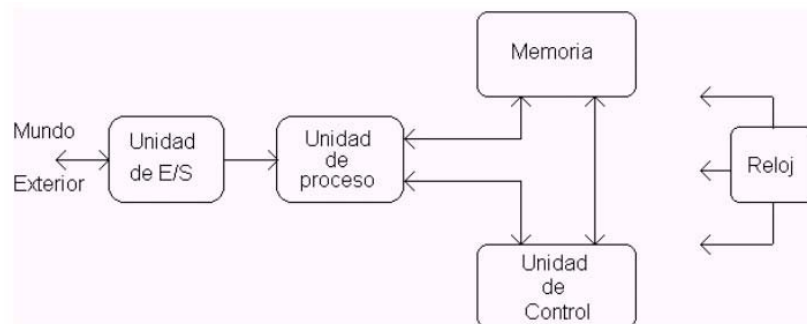
**1674.** G. H. von Leibniz, matemático alemán, incorpora a la máquina de Pascal la posibilidad de multiplicar y dividir.

**1850.** D. D. Parmalee automatiza la máquina Pascal-Leibniz. Aparecen las primeras cajas registradoras.

**Finales del siglo XIX.** Charles Babbage, matemático inglés, idea una máquina que es capaz de realizar cualquier operación. La máquina se instruye desde el exterior por medio de un sistema de tarjetas perforadas.

**1925.** V. Bush, ingeniero norteamericano, construye la primera máquina capaz de resolver ecuaciones diferenciales.

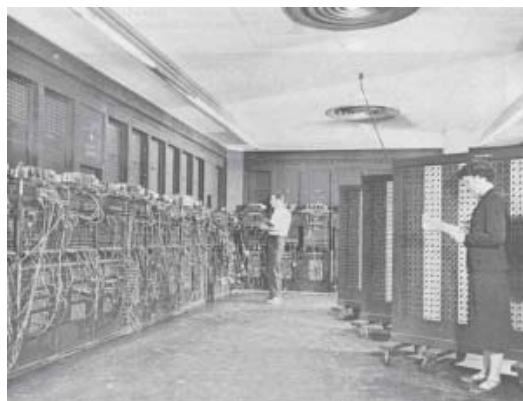
**1940.** Von Neumann acaba configurando la arquitectura básica de los ordenadores modernos, basada en los conceptos de programa almacenado y de ruptura de la secuencia de programa mediante la toma de decisiones. Su diagrama de bloques:



A partir de la concepción de máquina universal de Von Neumann, los ordenadores se desarrollan paralelamente a la tecnología. Aparecen cinco generaciones.

- **Primera generación.** Es la generación de las válvulas de vacío. Los ordenadores son muy voluminosos, con alto consumo y coste muy elevado. Algunos ordenadores famosos de esta generación son:

ENIAC –*Electronic Numerical Integrator and Computer*–, construido por la Universidad de Pennsylvania durante la II Guerra Mundial, estructurado por John Mauchly y John Eckert, estadounidenses, y puesta en funcionamiento en 1945, marca el inicio de nuestra era computacional. Se trata de una máquina programable y universal a la que se podía llamar electrónica; ocupa 160 metros cuadrados, pesa 30 toneladas, con 17.468 válvulas de vacío, 70.000 resistencias, 10.000 condensadores y un consumo de, aproximadamente, 200 KW. Deja de funcionar en 1955 y es desmontada en 1957.



Fotografía del ENIAC. Fuente: <http://ftp.arl.mil/>

UNIVAC –*Universal Automatic Computer*–; construido en 1951, es el primer computador digital con fines comerciales.

IBM604 e IBM650 aparecen en 1953 y se hacen muy populares en las décadas de los '50 y '60.

- **Segunda generación.** Aparece con el transistor bipolar. Se consigue una gran disminución de volumen, mayor fiabilidad y menor consumo. Los ordenadores se empiezan a hacer populares en las grandes empresas.
- **Tercera generación.** Aparecen los primeros circuitos integrados, se agudizan aún más las mejoras de la generación anterior. Paralelamente, empiezan a desarrollarse los lenguajes de alto nivel –como el *Fortran* y el *Cobol*–, así como los primeros sistemas operativos.
- **Cuarta generación.** Aparecen las técnicas de alta escala de integración –LSI–, propiciando, en 1971, la aparición del primer microprocesador 4004 de Intel. Posteriormente, en 1972, el 8008 de Intel y, a partir de éste, todos los fabricantes de circuitos integrados empiezan a lanzar sus microprocesadores: 6800 y 6802 de Motorola, 6502 de Rockwell, Z80 de Zilog, 8088 de Intel, etc. El ordenador empieza a hacerse muy popular. Aparecen también microprocesadores de 16 bits en el bus de datos: 8086, 80286 de Intel, 68000, 68010 de Motorola, Z8000 de Zilog, etc.
- **Quinta generación.** Aumenta la densidad de integración al aparecer la tecnología VLSI<sup>1</sup> de muy alta escala de integración. En 1976, gracias al aumento de la densidad de integración de componentes en un circuito integrado, sale a la luz el primer ordenador en un chip; como decíamos, a este nuevo integrado se le denominó microcomputadora monopastilla. Cuando los sistemas basados en microprocesadores se especializan en aplicaciones industriales, aparece la versión industrial de la microcomputadora monopastilla, el **microcontrolador**; estos dispositivos producen un obvio beneficio en aplicaciones pequeñas; su característica más sobresaliente es que son sistemas integrados, utilizados en la actualidad (en automóviles: inyección, suspensión; control de procesos; control de interruptores de potencia; etc.).

En este marco, resulta interesante reconstruir la historia de cambios tecnológicos desarrollada por *Intel*<sup>2</sup> –una compañía del valle del silicio, especializada en la fabricación de memorias–. Veamos...

<sup>1</sup> VLSI –Very large-scale integration–. Muy alta escala de integración; más de 10000 transistores por chip.

<sup>2</sup> <http://www.intel.es> ; <http://www.intel.com>



La casa *Data Point* saca a concurso un driver para la CRT<sup>3</sup> –hasta aquí, todo se hacía con transistores–. A este concurso sólo se presentan las firmas *Texas Instruments* e *Intel*. Al cabo del tiempo, *Texas* se retira e *Intel* termina su proyecto; pero no gana, porque el dispositivo es considerado poco eficiente como controlador: resulta 10 veces más lento de lo requerido y *Datapoint* no lo compra.

Por la época (1969), la compañía japonesa *Busicom*, fabricante de calculadoras, encarga a *Intel* la realización del circuito de control de un nuevo modelo de calculadora de sobremesa. El primer diseño resulta tan complejo que Ted Hoff, ingeniero de *Intel*, rediseña el sistema basándose en la estructura de los computadores: existiría un chip que integraría una CPU sencilla, que ejecutaría un programa de control que se guardaría en una memoria ROM; emplearía una RAM para almacenamiento de datos y un registro como puerto de entrada/salida. Por lo tanto, el nuevo diseño resulta un conjunto de cuatro chips que implementaban la estructura de un computador de modo que, cambiando el programa, podía dedicarse a realizar tareas muy diferentes.

Desde que se contrata a *Intel* hasta que el diseño resulta operativo pasan muchos meses, durante los cuales los precios caen y el valor del contrato inicial resulta desorbitado. *Busicom* pide, entonces, a *Intel* una rebaja en el precio. *Intel* acepta, a cambio de quedarse con los derechos del conjunto de circuitos integrados construido –la familia 4000– para poder comercializarlos por su cuenta. Finalmente, *Intel* introduce en sus catálogos el chip programable desarrollado –el 4004– en noviembre de 1971, bautizándolo como microprocesador y, así, comienza a comercializarlo.

El 4004 puede direccionar sólo 4096 (4k) localidades de memoria de 4 bits, reconocer 45 instrucciones y ejecutar una instrucción en 20  $\mu$ seg en promedio. *Intel*, entonces, lo saca al mercado para el uso de memorias. Al registrarse una demanda considerable, el equipo de diseño se pone a trabajar para desarrollarlo, consiguiendo el 8080 y el 8088, que son de 8 bits.

Con el microprocesador surgen las primeras computadoras de tipo personal y la introducción del procesador 8080, un dispositivo 10 veces más rápido que el 8008, con capacidad de direccionar 64kB de memoria.

Dos ingenieros del equipo de desarrollo crean la empresa *Zilog*, que comienza a competir con *Intel*, y fabrican el Z-80. *Intel* saca, entonces, el 8085 y, después, el 8086. Más tarde salen algunas versiones posteriores como el 8088 que es de 16 bits pero con buses de 8.

*MIT* introduce, en 1975, un kit que, en la actualidad, es considerado como la primera computadora personal: el modelo *Altair*. Esta pequeña computadora incluye una arquitectura abierta (basada en ranuras o slots) que permite conectar varios aditamentos y periféricos de otras marcas.

Hacia 1980, el mundo de las computadoras personales está dominado, básicamente, por dos tipos de sistemas:

- El *Apple II*, con un gran número de usuarios y una importante base de software, que crece rápidamente.
- Un sistema más sencillo que gira en torno al original *MIT Altair*, que se basa en la filosofía de la compatibilidad, apoyado en slots de expansión y en el empleo del sistema operativo CP/M<sup>4</sup>. Son máquinas construidas por varias compañías

<sup>3</sup> Tubos de rayos catódicos o terminal de video utilizado por televisores y computadoras.

<sup>4</sup> Sistema operativo de gran utilización hasta que aparece el DOS desarrollado por Microsoft.

y se venden con diversos nombres aunque, en esencia, utilizan el mismo software y el mismo hardware interconectable.

En 1981 se presenta la *IBM PC –IBM Personal Computer–*, introduciendo los lenguajes y sistemas operativos. La empresa llega a un acuerdo con *Microsoft* –por entonces una pequeña compañía– para incluir su sistema operativo DOS en sus modelos. La primera *IBM PC* incluye un microprocesador Intel 8088, con 128kB de RAM (expandibles 512kB), una unidad de disco flexible de 5 1/4 pulgadas y 360kB de capacidad.



Luego, aparece el estándar que toma el nombre de *IBM PC-XT –Personal Computer. Extended Technology–*; también aprovecha los microprocesadores de 16 bits y cumple con la principal virtud de la plataforma: su arquitectura abierta.

La plataforma PC no es la única; pero sí es la predominante, por su amplia gama de aplicaciones, a diferencia de otros formatos como *Macintosh, Sun, Amiga, Silicon Graphics*, etc.

La industria de microprocesadores se sigue desarrollando y se crea el 80186 que ofrece periferia integrada. Más tarde, aparece el 80286, adoptado por IBM; se trata de un 8086 pero con mayor capacidad de direccionamiento de memoria, que implementa el modo protegido.

Su evolución es el 80386, el primer micro Intel de 32 bits que puede direccionar hasta 4 GB. Cuenta con modo protegido, modo real y un nuevo modo que es el virtual –que trabaja en modo real, pero en realidad trabaja en protegido–; de aquí surgen el 80386SX y el 80486 con sus variantes:

- 486 SX: Como 386 sin coprocesador matemático, con caché interna y mejor arquitectura.
- 486 DX: Genérico pero a 33 y 50 MHz.
- 486 DX2: Interno a doble frecuencia: 50,66 MHz.
- 486 DX4: Interno a cuádruple frecuencia: 75,100 MHz.

La evolución hasta el momento actual:

**Equipo XT.** Año 1981, con procesador 8086 u 8088 de Intel, máquina pionera de los PC, usaba 128K, 256 o 512 Kilobytes de memoria Ram. Se le instalaba el sistema operativo DOS, disco duro de 5 o 10 megabytes de capacidad. Su gabinete era horizontal. El monitor era monocromático con fondo negro y letras de color verde. Su velocidad de procesamiento estaba entre los 4 y 8 MHz.

**Equipo AT 80286.** Su memoria Ram podía alcanzar los 16 MB, añadiéndole una placa de expansión especial. El disco duro normal para él era de 30 o 40MB. Las unidades de diskettes de 1.2 y 1.44 MB de capacidad. Sistema operativo DOS 2.1. Gabinete horizontal. Los monitores eran a color con tecnología EGA (baja resolución, comparados con los actuales VGA). Su velocidad promediaba los 25MHz.

**Equipos AT 80386.** Usaban en promedio 8 y 16 MB de ram, integrando módulos removibles de memoria, tipo SIMM de 32 pines. El disco duro promediaba los 512 MB, DOS 4.0. Monitores súper VGA. módem interno de 14.400 baudios por segundo. Velocidad promedio: 40MHz. Aparece la unidad CD ROM de simple velocidad.

**Equipos AT 80486.** Ram promedio de 8 y 16 MB. Discos duros de 1 gigabyte promedio. DOS 6.22. Módem de 28.800 bps. Monitores con resolución .28. Unidad de CD ROM de 2, 4 y 8 velocidades.

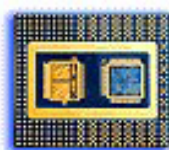
**Equipos Pentium de 75Mz.** Ram promedio de 16 MB expandible a 128 MB, discos duros de 3 gigas o mayor, módem de 33.600 bps, Windows 95, coprocesador matemático y memoria caché interna. Unidad de CD ROM de 16 velocidades. Otros procesadores: el 586 y 686 de AMD.

**Pentium MMX.** Tecnología con prestaciones para multimedia, con velocidades de 200, 233 y 266 MHz. Discos duros de 4 y 6 gigas. RAM promedio de 16 y 32 MB expandible. Unidad de CD ROM de 24 y 36 velocidades. Sistema Operativo Windows 95 A y Windows 95 B.

**Pentium Celeron.** Con velocidades desde 300 MHz a 1.3 GHz, caché de 128 KB, discos duros mayores de 6 gigas como promedio, monitores no entrelazados, RAM promedio de 32 MB expandible, motherboard multifuncional, tanto en equipos genéricos como de marca, Incluye, normalmente, sonido, video y módem fax incorporado en la placa madre. La velocidad de los módem promedio es de 56.600 bps. Las unidades de CD ROM alcanzan velocidades de 40X. De mejora Windows 95 creando Windows 98.

**Las generaciones de Pentium II, Pentium III y Pentium IV.** Básicamente, utilizan los mismos elementos de base: motherboard multifuncional, ram promedio de 128 MB, discos de 15, 30, 40 o más gigas, multimedia de 52x-56x, módem de 56.600. Sus velocidades varían desde los 350MHz a 550MHz los pentium II, de 500, 1 GHz los pentium III y 1.4 a 2GHz los pentium IV. Sistemas Operativos: Windows 98, Windows 2000, Windows NT, Windows Me y Windows XP. Otros procesadores: los K6-2 y Athlon de AMD.

- Pentium<sup>5</sup> P5: Trabaja a 5V con frecuencias de 60 y 66 MHz.
- Pentium P54: Trabaja con 3,34V y con frecuencias de 75 y 90 MHz.
- Pentium P54C/P54C5: Frecuencias de 100,120,135,150,166 MHz.
- Pentium P55C (MMX): Frecuencias de 166,200,233.
- Pentium PRO: Frecuencias de 180 y 200, con mejoras en arquitectura y con la caché integrada, utiliza un zócalo Socket 8.



Pentium® Pro processor

<sup>5</sup> El Pentium es un micro de 32 bits, pero con bus de datos de 64.

- Pentium II: Frecuencias de 233, 266, 300, que es un PPRO sin caché con MMX y caché de segundo nivel; tiene zócalo slot 1 y SEC.



- Utiliza el slot 2, que es una variante del slot 1, pero incompatible con aquel.
  - Está optimizado para aplicaciones de 32 bits.
  - Posee 32 KBytes de caché L1 (de primer nivel) repartidos en 16KB. para datos y los otros 16 para instrucciones.
  - La caché de segundo nivel puede ser de 512 KB o 1 MB.
  - Para comunicarse con el bus utiliza una velocidad de 100 MHz.
  - Incorpora 7,5 millones de transistores.
  - Puede cachear hasta 4 GB de memoria RAM.
- Pentium III: Se le han añadido las llamadas SSE –*Streaming SIMD Extensions*–, que son 70 nuevas instrucciones orientadas hacia tareas multimedia, especialmente en 3D. Estas extensiones son el equivalente a las 3D Now que lleva implementando AMD desde hace tiempo en el K6-2, K6-III y Athlon y que también han incorporado otros fabricantes –como IDT– en sus Winchip2 y 3.



Por supuesto, dicho juego de instrucciones –a pesar de realizar operaciones similares en ambos procesadores– es totalmente incompatible...

Otra de las novedades introducidas y también la más polémica es la incorporación de un número de serie que permite identificar unívocamente a cada una de las unidades, con lo que se obtiene una especie de “carné de identidad”, único para cada PC. Este ID se puede utilizar para realizar transacciones más seguras a través de Internet y para facilitar la vida a los administradores de redes; pero, también ha sido duramente criticado por algunos grupos de presión como una invasión de la privacidad, con lo que Intel se ha visto obligada a ofrecer una utilidad que permite desactivar dicha función:

- Está optimizado para aplicaciones de 32 bits.
- Se comercializa en versiones que van desde los 450 hasta los 600 MHz.
- Posee 32 Kbytes de caché L1 (de primer nivel) repartidos en 16Kb. para datos y los otros 16 para instrucciones.
- La caché L2 (segundo nivel) es de 512 Kb. y trabaja a la mitad de la frecuencia del procesador.
- La velocidad a la que se comunica con el bus (la placa base) es de 100 MHz.
- Incorpora 9,5 millones de transistores.
- Pueden cachear hasta 4 GB.
- Los modelos actuales todavía están fabricados con tecnología de 0,25 micras.

- **Pentium IV:** Ya hay tres nuevos procesadores basados en la arquitectura Pentium 4: 2.53, 2.40 y 2.26 GHz. Estos nuevos chips incluyen un bus de sistema a 533 MHz a diferencia de los anteriores Pentium 4 que interactuaban con un bus de velocidad menor: a 400 MHz. La compañía anunció, asimismo, la aparición del chipset 850E, diseñado para operar a la velocidad del nuevo bus de sistema.
- **Itanium:** La arquitectura EPIC –*Explicitly Parallel Instruction Computing*– paraleliza todas aquellas órdenes que se ejecutan en un ordenador. Si en los procesadores superescalares, como son el Pentium III o el Pentium 4, se ejecutan dos instrucciones por ciclo de reloj, Itanium es capaz de ejecutar hasta seis instrucciones por ciclo. Esto significa que es tres veces más rápido (En Itanium tenemos, por ejemplo, seis unidades de suma, dos de coma flotante y cuatro de enteros).

Principales fabricantes:



Gama Pentium: Classic, MMX, Pro, PII, Celeron, Xeon, PIII, P 4



K5, K6, K6-2, K6-III, Athlon



6x86, 6x86MX, MII, Media GX



Winchip C6, Winchip2, Winchip3

### Registros internos

Son bloques de biestables que permiten el almacenamiento de los datos básicos con los cuales va a trabajar la CPU durante la ejecución de cada instrucción.

### Unidad de control

Su función es enviar, consecutivamente, instrucciones del programa desde la memoria hasta la unidad de proceso.

### Unidad de proceso u operativa

Las instrucciones son decodificadas y ejecutadas en esta unidad.

### Unidad de memoria

En ella se almacena el programa a ejecutar, en forma de secuencia de instrucciones.

### Unidad de entrada/salida

Realiza la comunicación con el mundo exterior, es decir, con el ser humano. Recibe y representa los datos y resultados de forma asequible para el hombre.

## Sistema mínimo microprocesador

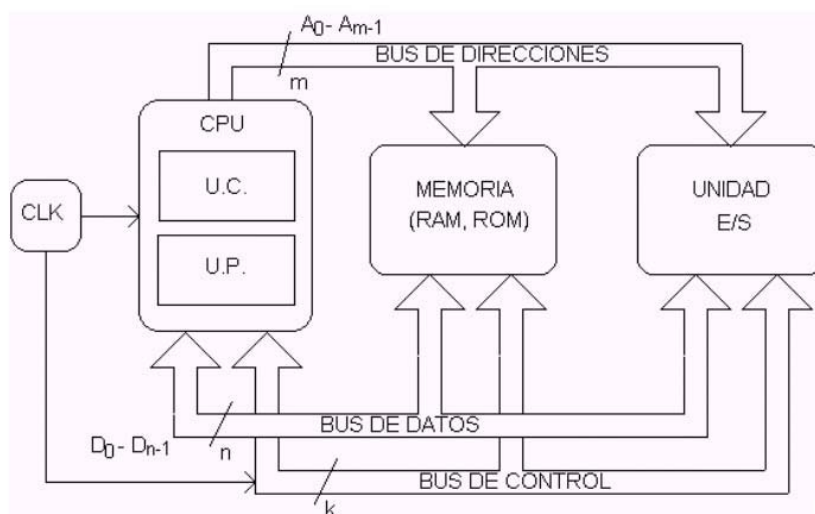
El sistema mínimo está basado en la estructura de Von Neumann. Sus bloques básicos son:

**1. Unidad central de proceso –CPU, Central Process Unit –, formada por los bloques:**

- 1.1. Registros internos.
- 1.2. Unidad de control.
- 1.3. Unidad de proceso u operativa.

**2. Unidad de memoria,** donde residen las instrucciones del programa y sus datos.

**3. Unidad de entrada/salida,** encargada de recibir y entregar información al exterior.



Los tres módulos están conectados entre sí por medio de los **buses de comunicación**.

Nos referiremos a tres clases de buses de comunicación:

- bus de direcciones,
  - bus de datos,
  - bus de control.
- **Bus de direcciones:**  $A_0-A_{m-1}$ . Es el empleado por la CPU para *seleccionar* la dirección de memoria o el dispositivo de E/S con el cual va a intercambiar información. Es, por tanto, *unidireccional*. El tamaño –número de conductores que lo constituyen– determina la **capacidad de direccionamiento** de la CPU, que es el máximo número de posiciones de memoria y dispositivos E/S a los que la CPU puede acceder. Para  $m$  líneas la capacidad de direccionamiento será:  $2^m$ .

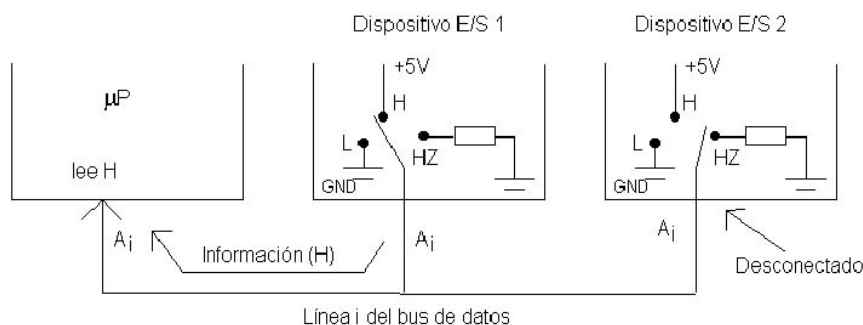
Ejemplos:

- 68HC08, Z80, 8051:  $m=16$   
Capacidad de dir. =  $2^{16} = 65536$  posiciones (64 k Byte)
- 80286:  $m=20$   
Capacidad de dir. =  $2^{20} = 1.048.576$  posiciones (1 Mega Byte)

- **Bus de datos:**  $D_0-D_{N-1}$ . El bus de datos es el conjunto de conductores a través del cual el microprocesador intercambia información con la unidad de memoria o E/S seleccionada mediante el bus de direcciones.

Características:

- Bidireccional: La información puede viajar en los dos sentidos.
- Número de líneas (N): representa la cantidad de bits que se pueden transmitir simultáneamente. Suele denominarse “la palabra del  $\mu P$ ”.
- Triestado: Las líneas del bus de datos deben ser triestado. Las líneas triestado son aquellas capaces de tener tres estados:
  - Estado alto (*High*, H).
  - Estado bajo (*Low*, L).
  - Estado de alta impedancia (*High Impedance*, HZ).



El  $\mu P$  se está comunicando con el dispositivo E/S 1. Sólo estos dos dispositivos pueden disponer del bus de datos. El resto de dispositivos conectados físicamente al bus de datos debe permanecer con sus líneas en alta impedancia (dispositivo E/S 2). La unidad de control de la CPU es la que decide qué elemento envía la información y qué elemento la recibe, así como los elementos que deben desconectarse del bus y ponerse en alta impedancia.

#### Bus de comunicación

Conjunto de conductores por los cuales se transmite la información digital en forma de pulsos eléctricos.



### Reloj

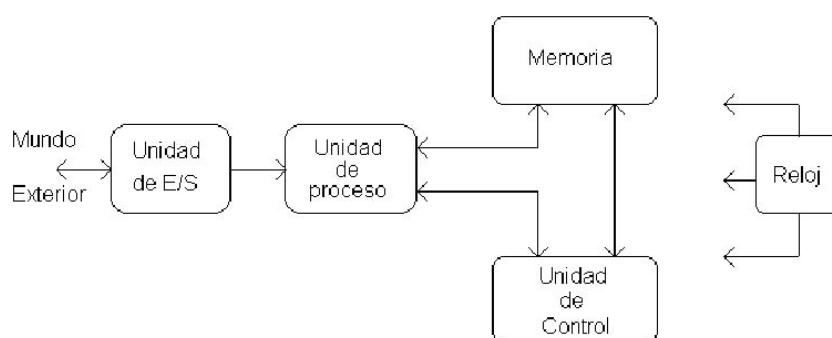
El reloj se encarga de sincronizar todo el sistema, compensando los retardos de los diferentes módulos. El sistema  $\mu P$  es un sistema secuencial; la secuencia que realiza es la ejecución continua de instrucciones.

- **Bus de control.** Está formado por un conjunto de líneas por las que circulan las señales auxiliares de gobierno y sincronización del sistema. Las líneas existentes dependen del fabricante del  $\mu P$  y de las funciones que desee implementarle.

Algunas señales típicas en todos los sistemas son:

- señal de reloj de sincronización,
- señal de RESET o inicialización,
- señal de lectura/escritura en memoria, etc.

Su diagrama de bloques es el siguiente:



Esta triple estructura que se ve en el diagrama **programa-unidad de proceso-unidad de control** surge de la propia estructura de los algoritmos:

### Algoritmo

Secuencia ordenada de acciones mediante la cual se resuelve un problema.

Ejemplo: Comprobar si dos números son iguales

INICIO

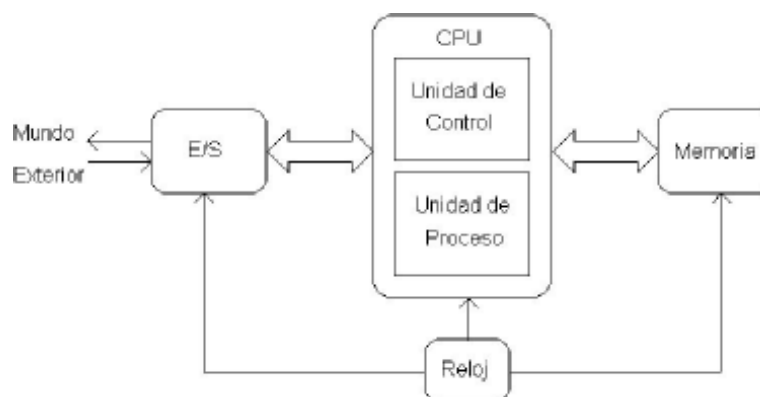
Iguales = 0

Leer A y B

Hacer  $D = A - B$

Si  $D = 0$  hacer Iguales = 1

FIN



En la **parte operativa** se realizan acciones y ensayos:

- leer A y B (acción);
- hacer  $A-B$  y asignarlo a D (acciones);
- comprobar si  $D=0$  (ensayo).

La **parte de control** define el encadenamiento (orden) de las acciones, según el resultado de los ensayos.

Esta estructura de los algoritmos es la que dio lugar a la arquitectura Von Neumann. El elemento básico es el microprocesador o CPU que engloba a la unidad de control y a la unidad de proceso.

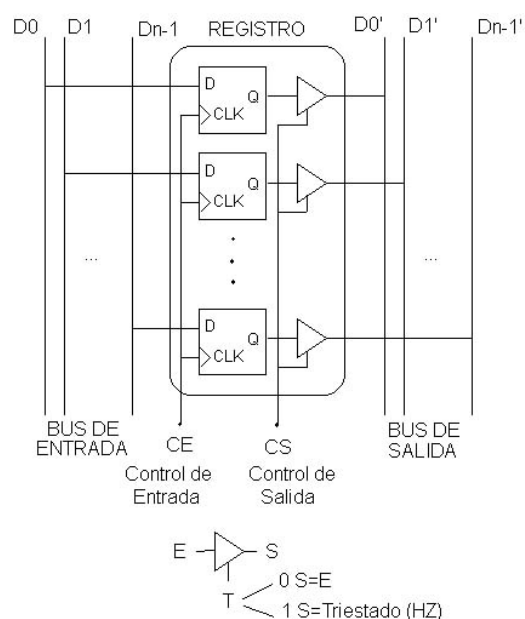
Le hablábamos al comienzo de los tres bloques básicos del sistema microprocesador. Nos referiremos, a continuación a cada uno de ellos.

## 1. Unidad central de proceso –CPU–

Está formada por los siguientes bloques:

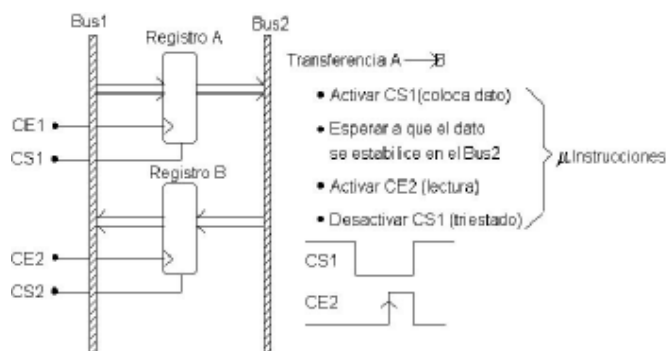
- 1.1. Registros internos
- 1.2. Unidad de control
- 1.3. Unidad de proceso u operativa

### 1.1. Registros internos



Los registros pueden escribirse, leerse e intercambiar información entre ellos por medio de los **buses internos de la CPU**.

Con estructuras adecuadas, es posible intercambiar información entre registros, cargar registros, descargarlos, etc.:





Las señales CS1 y CE2 son gestionadas por la unidad de control de la CPU instruida por el código de la instrucción de transferencia A-B.

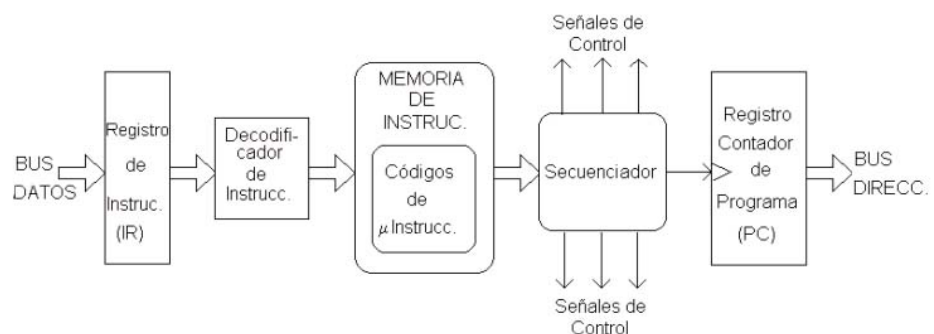
Los datos que están almacenados en los registros de la CPU tienen un **tiempo de acceso muy bajo**, muy inferior del correspondiente a los datos que se encuentran en memoria externa.

Existen diferentes tipos de registros que permiten realizar diferentes funciones:

- **De desplazamiento**, empleando biestables D encadenados. Permiten hacer rotaciones y desplazamientos de bits (x2, %2, E/S serie, etc.)
- **Contadores**, con biestables T encadenados. Permiten medir tiempos, temporizar, etc.
- **Registros índices o de direcciones**. Permiten el acceso estructurado a datos de memoria.
- **Registros de datos**, para almacenamiento de variables del programa.
- **Registros especiales**, contador de programa (*Program Counter –PC–*), puntero de pila (*Stack Pointer –SP–*).

### 1.2. Unidad de control

Su misión principal consiste en **interpretar y ejecutar** las instrucciones que le llegan desde la memoria:

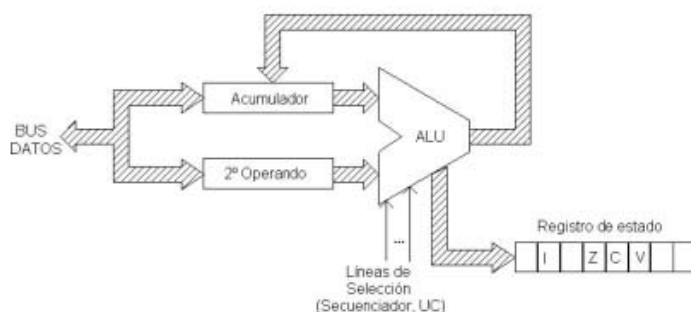


- El código binario de la instrucción en curso se recibe por el bus de datos y se almacena en el **Registro de Instrucciones (IR)**.
- El **decodificador de instrucciones** selecciona las posiciones que corresponden a esa instrucción en una memoria ROM interna de la CPU. En ella se almacenan las diferentes instrucciones elementales o microinstrucciones –*microinstrucciones*– que componen esa instrucción.
- Estas microinstrucciones hacen que el **secuenciador** active las señales correspondientes para ejecutar la instrucción.
- El **contador de programa –PC–** es un registro que contiene la dirección de memoria donde está la siguiente instrucción del programa a ejecutar. Una de las primeras acciones del secuenciador, antes de ejecutar una instrucción, es incrementar el PC para que *apunte* a la instrucción siguiente.
- Existen instrucciones que permiten modificar el PC o incrementarlo en más de una unidad, permitiendo la realización de saltos en la secuencia del programa.

### 1.3. Unidad de proceso u operativa

El bloque principal es la **ALU** –unidad lógico-aritmética– que permite realizar las operaciones aritméticas y lógicas indicadas por las instrucciones del programa.

El **secuenciador** –UC–, instruido por el código de la instrucción en curso, activa las líneas de selección de la ALU para realizar la operación.



Los operandos se suministran por medio de dos registros cargados desde el bus de datos:

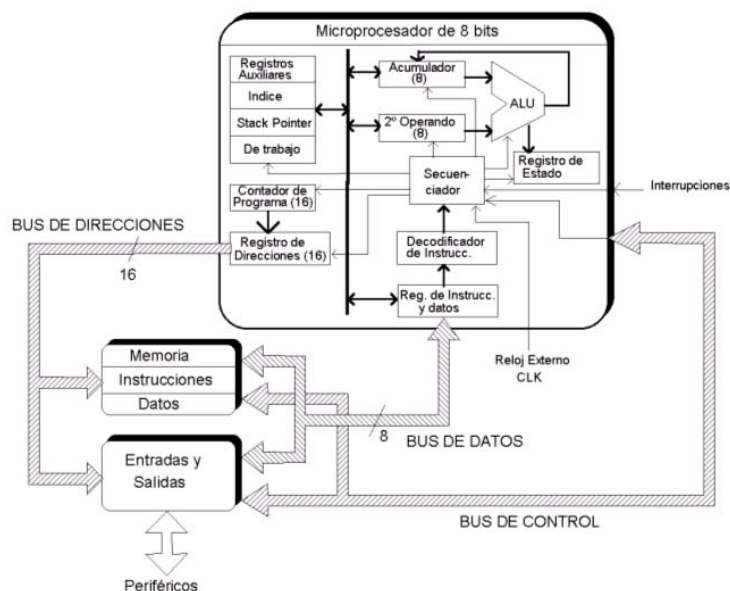
- **Registro acumulador**, contiene siempre el resultado de la última operación realizada en la ALU.
- **Registro segundo operando**, suministra el segundo operando para realizar la instrucción y viene normalmente aportado por el código de operación de la instrucción a ejecutar según los diferentes modos de direccionamiento.

Esta forma de trabajar es típica de los micros de 8 bits y permite simplificar las instrucciones, ya que cada instrucción sólo tiene que suministrar un operando: el otro se encuentra cargado previamente en el acumulador.

El **registro de estado** está formado por bits denominados banderas –*flags*– 1 ó 0, de acuerdo con el resultado obtenido. Algunos bits típicos son:

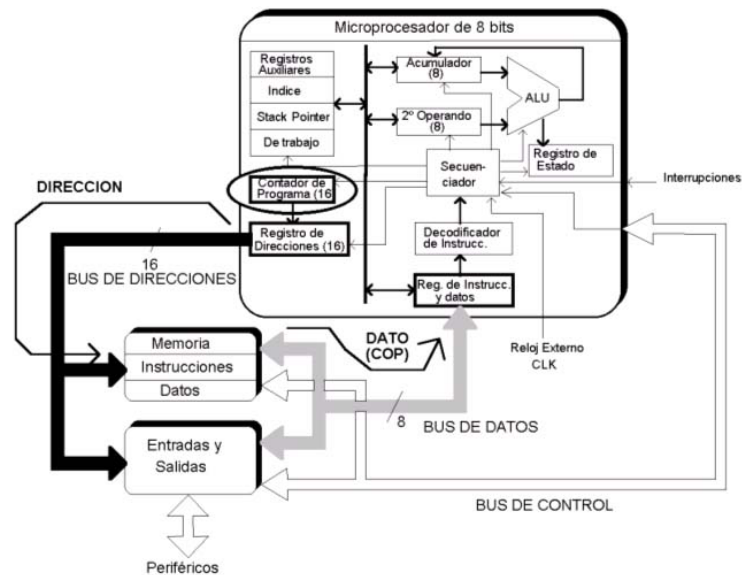
- **Z**, bit zero, se pone a 1 si el resultado fue nulo.
- **C**, bit carry, se pone a 1 si hubo acarreo de orden superior.
- **V**, bit overflow, se pone a 1 si hubo desbordamiento.
- **I**, bit de interrupción. Este bit es independiente del resultado. Escribiendo un 1 en él, por medio de la instrucción correspondiente, se puede inhibir la interrupción exterior enmascarable.

Diagrama general de un sistema basado en un microprocesador de 8 bits:

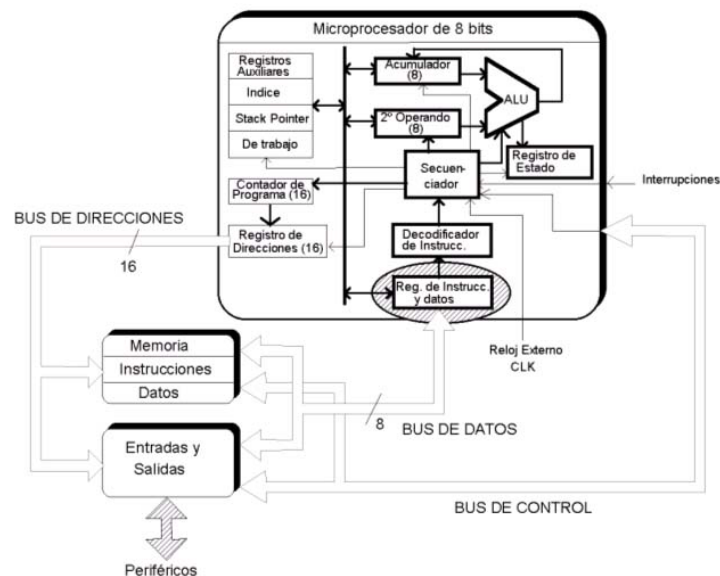


La realización de una instrucción se lleva a cabo en dos fases:

- **Fase de búsqueda.** Es común a todas las instrucciones. Se inicia en el contador de programa, que contiene la dirección de memoria donde se encuentra el código binario de la instrucción. Esta dirección se coloca en el registro de direcciones de la CPU y, de ahí a la memoria, a través del bus de direcciones. Una vez decodificada la dirección en la memoria, su contenido se traslada al bus de datos hacia el registro de Instrucciones de la unidad de control. En este momento finaliza la fase de búsqueda.

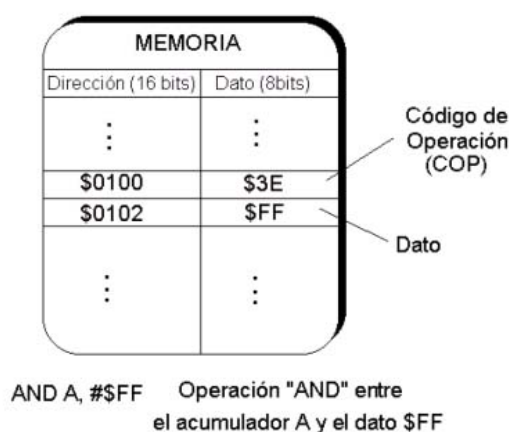


- **Fase de ejecución.** En esta fase se decodifica la instrucción dentro de la unidad de control. Se busca su código de microinstrucciones en la memoria interna de la CPU y se activan las señales correspondientes del secuenciador, para ejecutar la instrucción completa. Durante la fase de ejecución, los buses externos del sistema no son utilizados por la CPU. Pueden ser empleados por otros dispositivos externos para transferir datos.



Las instrucciones que constituyen el programa se almacenan en memoria, en paquetes de 8 bits (Bytes). Normalmente, la forma es la siguiente:

- **Byte 1.** Es el código de operación –**COP** o **OPCODE**– que indica la operación de la que se trata y su función. Por ejemplo: Operación lógica AND.
- **Byte 2 y siguientes.** Normalmente, nos dan la información necesaria para acceder al dato sobre el que va a trabajar la instrucción. Puede ser el propio dato, la dirección de memoria donde se encuentra el dato, etc. las diferentes posibilidades para acceder a ese dato se denominan **modos de direccionamiento** del microprocesador.



Los COP suelen ser del mismo tamaño que el bus de datos del micro.

- En los micros de 8 bits (6805/08, Z80, 8051), los COP son de 8 bits.
- En los micros de 16 bits (68000, 8086, Z8000), los COP son de 16 bits.

En general, el conjunto de instrucciones del microprocesador se puede dividir en los siguientes bloques funcionales:

- **Aritméticas.** Suma, resta, producto, etc.
- **Lógicas.** AND, OR, NOT, EXOR, etc.
- **De transferencia.** Permiten transferir datos entre registros, entre memoria y los registros de la CPU, entre dos posiciones de memoria, etc.
- **De entrada/salida.** Permiten la lectura y escritura en los bloques periféricos de E/S.
- **De ruptura de secuencia del programa.** Permiten realizar saltos condicionales o incondicionales dentro del programa.
- **De control.** Parada (STOP), no operación (NOP), interrupciones, etc.



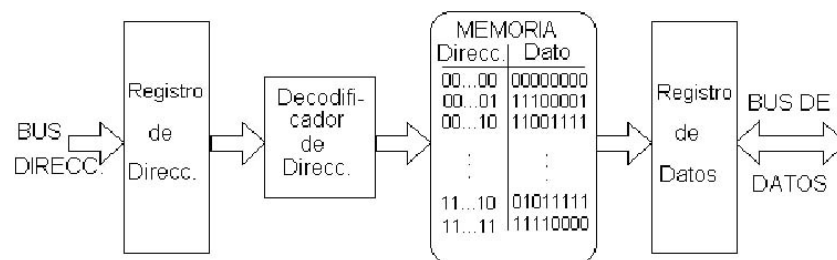
## 2. Unidad de memoria

La memoria se emplea para dos funciones básicas:

- Almacenamiento de las instrucciones que constituyen el programa. Las instrucciones se codifican mediante su **código de operación –COP–**, formado por uno o varios Bytes.
- Almacenamiento de los datos y variables del programa. Éstos están normalmente agrupados en paquetes de 8 bits (Byte), 16 bits (Word) o 32 bits (Long Word).

La velocidad (o el tiempo de respuesta) de la memoria debe ser compatible con el microprocesador empleado. Normalmente, se emplean memorias basadas en semiconductores:

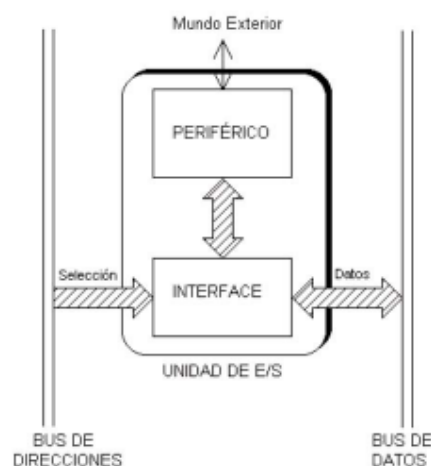
- RAM, memorias de lectura/escritura para almacenamiento de variables y datos del programa. Son volátiles.
- ROM (EPROM, EEPROM), memorias de sólo lectura para almacenamiento de programas fijos (aplicaciones, rutinas básicas de sistemas operativos, etc.) y constantes. Son no volátiles.



## 3. Unidad de entrada/salida

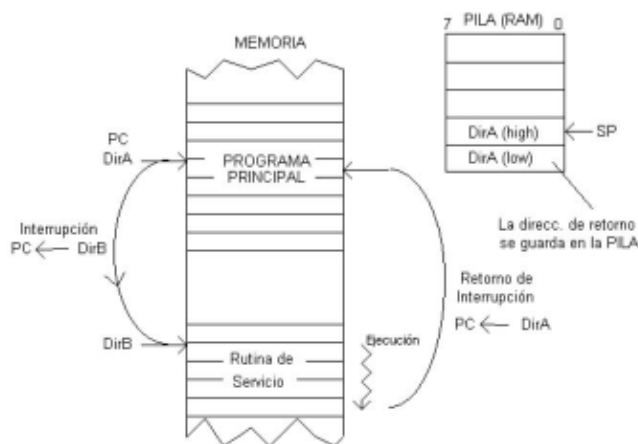
Permiten la comunicación del sistema microprocesador con el mundo exterior. Los dispositivos de E/S se denominan habitualmente **periféricos** (por ejemplo: teclado, pantalla, impresora, unidades de disco, etc.)

Cualquier periférico necesita un módulo adicional que permite realizar su conexión con los buses del sistema microprocesador; este módulo se denomina **interface**:



Existen varios métodos para manejar los dispositivos de E/S:

- Mediante **instrucciones específicas de E/S** que se emplean en el programa de control para acceder al periférico.
- Mediante **acceso directo a memoria –DMA–**. La CPU pone en triestado los buses de direcciones y de datos. Un dispositivo controlador de DMA toma el control de los buses y pasa los datos, directamente, entre el dispositivo E/S y la memoria.
- Mediante **técnicas de interrupción**. El periférico activa las líneas de interrupción de la CPU que detienen el programa en ejecución y trasladan el contador de programa a la dirección de inicio de otro programa creado especialmente para atender al periférico que solicita la interrupción. Suele denominarse **rutina de servicio**.



- Mediante el **tratamiento de E/S como posiciones de memoria**. Permite el empleo de las mismas instrucciones para acceso a memoria y a E/S. Una zona del mapa de memoria es reservada para los dispositivos de E/S. Estas posiciones se llaman puertos de E/S. Escribir o leer en uno de estos puertos equivale a hacerlo en el periférico.

### Arquitecturas básicas de microprocesadores y microcontroladores. Harvard versus Von Neumann

Harvard y Von Neumann son dos arquitecturas que se caracterizan por la forma en la que distribuyen la memoria de datos y de programa dentro de un microcontrolador.

En la arquitectura Harvard, la memoria de datos y la de programa están separadas, y para almacenar las instrucciones se utilizan lo que da en llamarse **palabras**. Palabras las hay de muchos tamaños como, por ejemplo, en los microcontroladores PIC. Los PIC de gama baja usan palabras de 12 bit, los de gama media 14 bit y los de gama alta 16 bit.

Por otro lado, está la arquitectura Von Neumann que se caracteriza por tener la memoria de programa y la de datos implementada en un mismo bloque de memoria, compartiendo datos e instrucciones en un mismo bus. En este tipo de microcontroladores se usan bytes para almacenar datos e instrucciones. Ejemplos de estos microcontroladores son los Zilog, National Semiconductors o los de Motorola.

Ambas arquitecturas tienen ventajas e inconvenientes y, como siempre pasa en el mundo de la electrónica, la selección depende de la aplicación en la que vayamos a usar el microcontrolador, la frecuencia de trabajo, la conexión a otros periféricos, etc.

- **Arquitectura Von Neumann**

Los sistemas basados en arquitectura Von Neumann tienen un solo bus para acceder tanto a datos como a instrucciones. Diremos que un microcontrolador es de 4 bit cuando el bus de éste es de 4 bit; será de 8 bit cuando el bus sea de 8 bit. Esto facilita en gran medida la conexión de memoria externa a través de las líneas de entrada/salida con una mínima implementación extra de hardware.

Por contra, tenemos que una instrucción puede ocupar más de un byte, con lo que para poder leer la instrucción completa tendremos que hacer varias lecturas en la memoria. Será necesario emplear varios ciclos de reloj para extraer una instrucción.

Otra desventaja es que es posible que el contenido del contador del programa se corrompa, con lo que se podría estar leyendo un dato y tratar de interpretarlo como instrucción, con la consecuencia del deterioro y caída del sistema. Normalmente, un microprocesador controla que el contador “no haga cosas raras”.

Una de las características que poseen los microcontroladores basados en este tipo de arquitectura es que suelen tener un repertorio de instrucciones bastante grande. Este tipo de repertorio se llama CISC –*Complex Instruction Set Computer*–. La característica principal de este conjunto es que suele ser bastante elevado, con instrucciones microcodificadas –una instrucción es decodificada por la CPU en varias instrucciones básicas–. Es fácil deducir que esto hace la ejecución del programa un poco más lenta; pero, tiene la ventaja de que ahorramos memoria: para dividir sólo usamos una instrucción (Por supuesto, aún así, tenemos que tener en cuenta que la decodificación se está llevando a cabo en el interior del microcontrolador, lo que va a ralentizar su ejecución; y considerar qué tipo de microcontrolador usar, según la tarea a desarrollar).

De acuerdo con la problemática a resolver, debemos seleccionar el  $\mu$ C según:

- Arquitectura
- Conjunto de instrucciones
- N° de instrucciones

Por ejemplo:

- Pic 16Cxxx Harvard RISC 35
- Pic 17Cxxx Harvard RISC 58
- Motorola HC11 Von Neumann CISC 109
- Intel 8051 Von Neumann CISC 40

Así, pues, podemos decir que la principal ventaja de usar MCU<sup>6</sup> con conjunto de instrucciones CISC es que, para una instrucción compleja, sólo usaremos una posición de memoria (al contrario de lo que ocurre con RISC –*Reduced Instruction Set Computer*, que consideraremos en un momento más–; en este último conjunto de instrucciones, para realizar, por ejemplo, una división debemos de usar varias instrucciones, consumiendo más memoria).

Frente a esta ventaja de los repertorios CISC, se nos presenta una desventaja con respecto a los RISC, y es que el ancho de banda se ve reducido considerablemente, debido a que una instrucción va a consumir varios ciclos de instrucción para ejecutarse. Como ya analizamos, estos microcontroladores son más lentos que los que usan

<sup>6</sup> MCU es la abreviatura de chip o unidad microcontroladora o microprocesador monopañilla.



repertorios RISC y, además, puede ser que el conjunto de instrucciones sea bastante grande (como vamos a considerarlo en nuestro próximo capítulo para Motorola HC11 o INTEL 8051), lo que no es en sí un gran problema pero sí un trabajo extra para aprender a usarse.

- **Arquitectura Harvard**

La arquitectura Harvard se caracteriza por tener la memoria de datos separada de la memoria del programa y éstas, a su vez, unidas a la CPU a través de buses independientes.

El que la información se almacene en palabras tiene una gran ventaja y es que tanto el código de operación como el dato asociado a éste están en la misma posición (por tanto, su lectura es mucho más rápida). Ésta es una gran ventaja ya que dota al microcontrolador de gran agilidad.

La técnica de procesar varias instrucciones al mismo tiempo se conoce como *pipelining* o segmentación. Con esta técnica se dividen las instrucciones en distintas etapas, de modo que el procesador pueda procesar distintas instrucciones en estas etapas.

Veamos un ejemplo de esto.

Una instrucción típica se divide en cinco etapas:

1. Búsqueda de la instrucción.
2. Decodificación de la instrucción.
3. Lectura del dato a través del bus de datos, ya sea de la memoria RAM o de cualquier registro mapeado en cualquier parte de la zona de memoria.
4. Ejecución de la instrucción.
5. Almacenamiento del resultado en memoria.

Una desventaja de este sistema es que la adición de memoria externa es mucho más compleja e, incluso, a veces, imposible.

En contraposición con los microcontroladores con arquitectura von Neumann, los basados en Harvard tienen repertorios de instrucciones RISC –*Reduced Instruction Set Computer*–.

Sus ventajas saltan a la vista: más rapidez de ejecución de código –incluso en algoritmos de división que vienen implementados con más código en los RISC que en los CISC, la velocidad de ejecución es mayor en los primeros– y menor el número de instrucciones –lo que simplifica, también, nuestro aprendizaje cuando queremos programar un MCU para una tarea simple–.

Por el contrario, requiere utilizar más memoria en un PIC<sup>7</sup> que en un HC08 o un 8051 para realizar una función.

<sup>7</sup> PIC es el nombre de los microcontroladores de la línea Microchip, HC08 de la línea Motorola y 8051 de la de Intel.



## ***FICHA 2***

---

### ***Introducción a los microcontroladores***



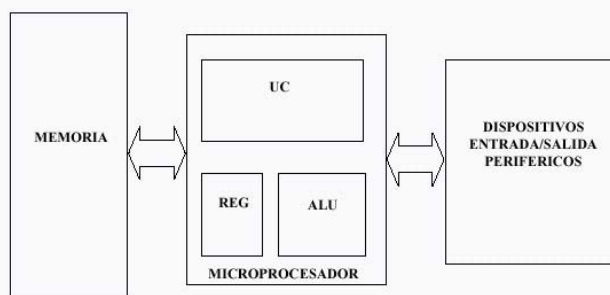
### Microcontrolador

Circuito integrado o chip que incluye en su interior las tres unidades funcionales de un ordenador: CPU –unidad central de proceso–, memoria y unidades de entrada/salida.

Los microcontroladores están conquistando el mundo. Están presentes en nuestro trabajo, en nuestra casa y en nuestra vida, en general. Se pueden encontrar controlando el funcionamiento de los ratones y de los teclados de los computadores, en los teléfonos, en los hornos microondas y en los televisores de nuestro hogar. Pero... la invasión acaba de comenzar y el nacimiento del siglo XXI será testigo de la conquista masiva de estos diminutos computadores que gobernarán la mayor parte de los aparatos que fabricaremos y usaremos los humanos.

Se trata de un computador monopastilla de limitadas prestaciones que, normalmente, se dedica a resolver una tarea específica. Es más pequeño que una caja de fósforos –por lo que suele estar incluido en el dispositivo que controla– y, en ocasiones, cuesta menos que aquélla.

#### MICROPROCESADOR



Un *microcontrolador* es un microprocesador optimizado para controlar equipos electrónicos. Los microcontroladores representan la inmensa mayoría de los chips de ordenador vendidos: un 50% está integrado por controladores “simples” y otro 50% por DSP –Procesador digital de señal– más especializados. Se puede tener uno o dos microprocesadores de propósito general en casa (usted está usando uno para leer este material de capacitación), y entre una y dos docenas de microcontroladores –si sumamos los que contiene cada dispositivo eléctrico de lavadoras, horno microondas, teléfonos, etc.–.

Un microcontrolador difiere de una CPU normal, debido a que es más fácil convertirlo en un ordenador en funcionamiento, con un mínimo de microchips externos de apoyo: el chip se coloca en el dispositivo, conectado a la fuente de energía y de información que necesita... ¡y eso es todo! Por supuesto, un microprocesador tradicional no nos permite hacer esto, ya que espera que todas estas tareas sean manejadas por otros chips.

Consideremos un ejemplo. Un microcontrolador típico tiene un generador de reloj integrado y una pequeña cantidad de memoria RAM y ROM/EPROM/EEPROM; para hacerlo funcionar, sólo se necesitan unos pocos programas de control y un cristal de sincronización.

¿Recuerda el problema que tenemos que resolver? Bien... Dijimos que podíamos solucionarlo por caminos diferentes y que el más adecuado, en cuanto a fuerzas a desarrollar, sería el de los dispositivos de accionamiento neumático; en lugar de controlarlos con dispositivos también neumáticos –que son muy caros– resultaría de máxima eficacia utilizar componentes electrónicos.

Entonces, además del microcontrolador (que posee memoria RAM, ROM, y dispositivos de entrada y salida de información) se necesitarán algunas resistencias, dos transistores, algunos diodos y una placa de circuito impreso.

La ventaja comparativa más importante de este modo de resolución es el costo, que no representa más de una decena de dólares.

Por lo general, los microcontroladores también disponen de una gran variedad de dispositivos de entrada/salida, como convertidores analógico a digital, temporizadores, UART y buses de interfaz serie especializados, como I<sup>2</sup>C y CAN. Frecuentemente, estos dispositivos integrados pueden ser controlados por instrucciones de procesadores especializados. Algunos microcontroladores incluyen un lenguaje de programación integrado, como el lenguaje de programación BASIC.

El controlador embebido se emplea para controlar el funcionamiento de una tarea determinada y, debido a su reducido tamaño, suele ir incorporado –de ahí su denominación– en el propio dispositivo al que gobierna. Se dice que es “la solución en un chip”, porque su reducido tamaño minimiza el número de componentes y el coste.

El microcontrolador es un computador dedicado: En su memoria sólo reside un programa destinado a gobernar una aplicación determinada; sus líneas de entrada/salida soportan el conexionado de los sensores y actuadores de ese dispositivo a controlar. Una vez programado y configurado, el microcontrolador solamente sirve para gobernar la tarea asignada<sup>1</sup>.

Estas características los hacen muy apropiados para desarrollar dispositivos de control o máquinas automáticas, ya que poseen los elementos claves de control encapsulados en un mismo chip, lo que posibilita una inmejorable inmunidad al ruido (Inmunidad al ruido muy envidiada en los sistemas con microprocesador que poseen sus buses al aire, paseándose por el circuito impreso, o por cintas planas de cables de interconexión entre periféricos e interfaces).

Un microcontrolador es la unión de tres tipos de dispositivos en un chip:

- un microprocesador,
- memorias y
- otros dispositivos periféricos.

Evidentemente, el corazón del microcontrolador es un microprocesador; pero, cabe recordar que...

El microcontrolador es para una aplicación concreta y no es universal como el microprocesador.

El número de productos que funcionan en base a uno o varios microcontroladores aumenta de forma exponencial. Casi todos los periféricos del computador (ratón, teclado, impresora, etc.) son regulados por el programa de un microcontrolador.

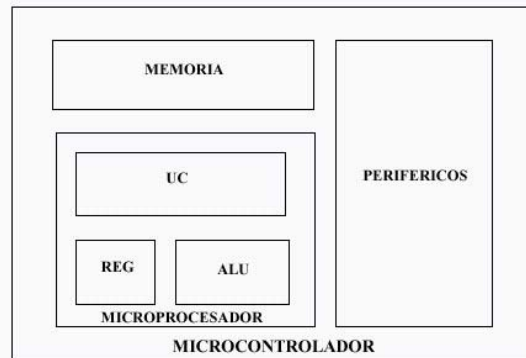
#### **Controlador embebido**

#### **–embedded controller–**

Es un circuito integrado que contiene todos los componentes de un computador.

<sup>1</sup> Aún cuando estamos enfatizando el hecho de que el microcontrolador es un dispositivo dedicado a una tarea en particular, una vez programado, gracias a los avances de la tecnología, este programa puede ser actualizado tantas veces como sea necesario, incluso en el mismo sistema de trabajo; es decir, podemos reprogramarlo para que realice otras tareas o mejorar su funcionamiento incorporando rutinas más apropiadas.

## MICROCONTROLADOR



Los electrodomésticos de línea blanca (lavadoras, hornos, etc.) y de línea marrón (televisores, videos, aparatos de música, etc.) incorporan numerosos microcontroladores. Igualmente, los sistemas de supervisión, vigilancia y alarma en los edificios utilizan estos chips para optimizar el rendimiento de ascensores, calefacción, alarmas de incendio, robo...

Los microcontroladores ofrecen la única solución práctica a muchos problemas de diversos campos:

- Periféricos y dispositivos auxiliares de los computadores.
- Electrodomésticos.
- Aparatos portátiles y de bolsillo (tarjetas, monederos, teléfonos, etc.).
- Máquinas expendedoras y juguetería.
- Instrumentación.
- Industria de automoción.
- Control industrial y robótica.
- Electromedicina.
- Sistema de navegación espacial.
- Sistemas de seguridad y alarma.
- Domótica en general.



*Aplicaciones de microcontroladores en un automóvil*

En las aplicaciones sencillas resultan preferibles las soluciones no programables que no requieren desarrollo de software (escribir software consume mucho tiempo, por lo que resulta más costoso; en aplicaciones sencillas y/o de poca tirada, a menudo es más razonable efectuar tareas en hardware). Sin embargo, conforme aumenta la complejidad del sistema, aumentan las ventajas del uso de sistemas programables.

Una de las principales ventajas de los sistemas programables es su flexibilidad, la que permite actualizar el funcionamiento de un sistema mediante –tan sólo– el cambio del programa, sin tener que volver a diseñar el hardware. Esta flexibilidad es muy importante, al permitir que los productos se actualicen con facilidad y economía.

## Controlador y microcontrolador

Partamos de un ejemplo para caracterizar cómo se vinculan, y en qué se diferencian un controlador y un microcontrolador. El controlador que regula el funcionamiento de un horno dispone de un sensor que mide constantemente su temperatura interna y, cuando ésta traspasa los límites prefijados, genera las señales adecuadas que accionan los efectores que intentan llevar el valor de la temperatura dentro del rango estipulado.

Aunque el concepto de controlador ha permanecido invariable a través del tiempo, su implementación física ha variado frecuentemente. Hace tres décadas, los controladores se construían exclusivamente con componentes de lógica discreta; posteriormente, se emplearon los microprocesadores, que se rodeaban con chips de memoria y E/S sobre una tarjeta de circuito impreso. En la actualidad, todos los elementos del controlador se han podido incluir en un chip, el cual recibe el nombre de **microcontrolador**, que consiste en un sencillo pero completo computador, contenido en el corazón *—chip—* de un circuito integrado.

Un microcontrolador es un circuito integrado de alta escala de integración que incorpora la mayor parte de los elementos que configuran un controlador.

Un microcontrolador dispone normalmente de los siguientes componentes:

- Procesador o CPU.
- Memoria RAM para contener los datos.
- Memoria para el programa tipo ROM/PROM/EPROM.
- Líneas de E/S para comunicarse con el exterior.
- Diversos módulos para el control de periféricos (temporizadores, puertas serie y paralelo, ADC: conversores analógico/digital, DAC: conversores digital/analógico, etc.).
- Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.

Los productos que para su regulación incorpora un microcontrolador, disponen de las siguientes ventajas:

- Aumento de prestaciones. Un mayor control sobre un determinado elemento representa una mejora considerable.
- Aumento de la fiabilidad. El microcontrolador reemplaza a un elevado número de elementos, disminuye el riesgo de averías y requiere menos ajustes.
- Reducción del tamaño en el producto acabado. La integración del microcontrolador en un chip disminuye el volumen, la mano de obra y el *stock*.
- Mayor flexibilidad. Las características de control están programadas, por lo que su modificación sólo necesita cambios en el programa de instrucciones.

En nuestro problema, es bueno recordar que la solución planteada con microcontroladores permite mayor flexibilidad, ya que prevé algunas entradas y salidas extras que posibilitarán controlar otros dispositivos —como son los cargadores automáticos de piezas, la descarga de productos automatizada o trabajar en forma autónoma hasta una cantidad determinada de piezas—. ¿Como se logra esto? Simplemente, actualizando el programa y conectando algunos transistores o relés.

Respecto a la inmunidad al ruido, al encontrarse prácticamente todo en un chip, es difícil interferir en su funcionamiento o toma de decisión.

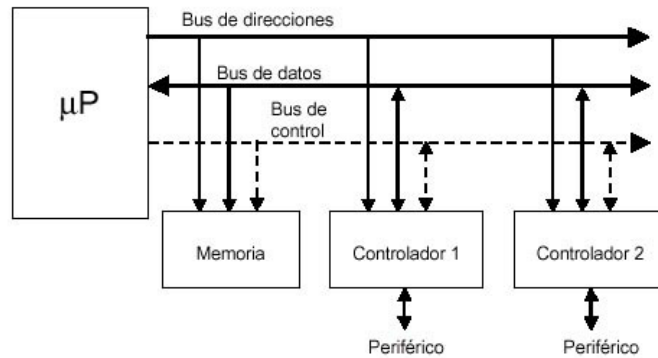
### Controlador

Dispositivo que se emplea para el gobierno de uno o varios procesos.

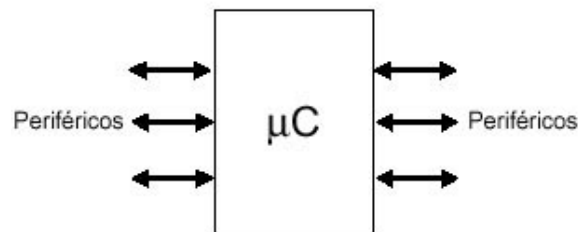
## Microprocesador y microcontrolador

Un microprocesador es un sistema **abierto** (configuración variable) con el que puede construirse un computador con las características que se desee, acoplándole los módulos necesarios.

Un microcontrolador es un sistema **cerrado** que contiene un computador completo y de prestaciones limitadas que no se pueden modificar.



Estructura de un sistema abierto basado en un microprocesador



Microcontrolador

El microcontrolador es un sistema cerrado. Todas las partes del computador están contenidas en su interior y sólo salen al exterior las líneas que gobiernan los periféricos.

Cuando el volumen de información es importante o bien cuando el tratamiento de sus datos es exigente, el sistema con microprocesador es más apropiado, por tener una estructura abierta que permite al diseñador seleccionar los periféricos más adecuados y prever futuras expansiones de memoria.

Decidirse por construir el sistema con el microprocesador o utilizar, directamente, el microcontrolador dependerá de la economía. Si el  $\mu C$  está limitado por su propia CPU, es necesario elegir un  $\mu P$  potente y añadir los buffers, drivers, decodificadores, memorias... pertinentes. Generalmente, salvo que la aplicación exija grandes prestaciones, el  $\mu C$  será una solución válida, con la ventaja de que reduce el espacio y el hardware.

Si sólo dispusiésemos de un modelo de microcontrolador, éste debe tener muy potenciados todos sus recursos para poderse adaptar a las exigencias de las diferentes aplicaciones. Esta potenciación supone, en muchos casos, un despilfarro.

En la práctica, cada fabricante de microcontroladores oferta un elevado número de modelos diferentes, desde los más sencillos hasta los más poderosos. Es posible seleccionar la capacidad de las memorias, el número de líneas de E/S, la cantidad y potencia de los elementos auxiliares, la velocidad de funcionamiento, etc. Por todo ello, un aspecto muy destacado del diseño es la selección del microcontrolador a utilizar.

Para resolver nuestro problema, es necesario determinar la cantidad de entradas y salidas vinculadas con el proceso, así como otros elementos necesarios para la determinación de posiciones, acciones del operario y protecciones.

También es necesario evaluar posibles expansiones, de acuerdo a las tareas a realizar; y considerar la memoria, tanto RAM como ROM.

Todo esto para determinar y caracterizar al microcontrolador más adecuado para nuestra situación problemática y buscar el que se adapte mejor a dicha situación, tanto técnica como económicamente.

### ***Aplicaciones de los microcontroladores***

Cada vez existen más productos que incorporan un microcontrolador con el fin de aumentar sustancialmente sus prestaciones, reducir su tamaño y coste, mejorar su fiabilidad y disminuir el consumo.

Este dato puede dar una idea de la masiva utilización de estos componentes: Algunos fabricantes de microcontroladores superan el millón de unidades de un modelo determinado, producidas en una semana. Aunque en el mercado de la microinformática la mayor atención la acaparan los desarrollos de los microprocesadores, lo cierto es que se venden cientos de microcontroladores por cada uno de aquéllos.

Una aplicación típica podría emplear varios microcontroladores para controlar pequeñas partes del sistema. Estos controladores podrían comunicarse entre ellos y con un procesador central, probablemente más potente, para compartir la información y coordinar sus acciones, como, de hecho, ocurre ya habitualmente en cualquier PC.

Existe una gran diversidad de microcontroladores. Quizá la clasificación más importante sea entre microcontroladores de 4, 8, 16 ó 32 bits. Aunque las prestaciones de los microcontroladores de 16 y 32 bits son superiores a los de 4 y 8 bits, la realidad es que los microcontroladores de 8 bits dominan el mercado y los de 4 bits se resisten a desaparecer. La razón de esta tendencia es que los microcontroladores de 4 y 8 bits son apropiados para la gran mayoría de las aplicaciones, lo que hace absurdo emplear micros más potentes y, consecuentemente, más caros.

Uno de los sectores que más tracciona el mercado del microcontrolador es el mercado automovilístico. De hecho, algunas de las familias de microcontroladores actuales se desarrollaron pensando en este sector, siendo modificadas posteriormente para adaptarse a sistemas más genéricos. El mercado del automóvil es, además, uno de los más exigentes: los componentes electrónicos deben operar con condiciones extremas de vibración, choques, ruido, etc. y seguir siendo fiables, ya que el fallo de cualquier componente en un automóvil puede ser el origen de un accidente.



En cuanto a las técnicas de fabricación, cabe decir que prácticamente la totalidad de los microcontroladores actuales se fabrica con tecnología CMOS 4 –*Complementary Metal Oxide Semiconductor*–. Esta tecnología supera a las técnicas anteriores por su bajo consumo y su alta inmunidad al ruido.

La distribución de las ventas, según su aplicación, es la siguiente:

- Una tercera parte se absorbe en las aplicaciones relacionadas con los computadores y sus periféricos.
- La cuarta parte se utiliza en las aplicaciones de consumo (electrodomésticos, juegos, TV, video, etc.)
- El 16% de las ventas mundiales se destina al área de las comunicaciones.
- Otro 16% es empleado en aplicaciones industriales.
- El resto de los microcontroladores vendidos en el mundo, aproximadamente un 10%, es adquirido por las industrias de automoción.

Los modernos microcontroladores de 32 bits también van afianzando sus posiciones en el mercado. Las áreas más interesadas en ellos son las del procesamiento de imágenes, las comunicaciones, las aplicaciones militares, los procesos industriales y el control de los dispositivos de almacenamiento masivo de datos.

### ¿Qué microcontrolador emplear?

A la hora de escoger el microcontrolador a emplear en un diseño concreto –como el del problema que nos ocupa–, hay que tener en cuenta multitud de factores, como la documentación y herramientas de desarrollo disponibles y su precio, la cantidad de fabricantes que lo producen y, por supuesto, las características del microcontrolador (tipo de memoria de programa, número de temporizadores, interrupciones, etc.).

**Costos.** Como es lógico, los fabricantes de microcontroladores compiten duramente para vender sus productos. Y no les va demasiado mal ya que, sin hacer demasiado ruido, venden 100 veces más microcontroladores que microprocesadores. Para que nos hagamos una idea: para el fabricante que usa el microcontrolador en su producto, una diferencia de precio de algunos dólares en el microcontrolador es importante (el consumidor debe pagar, además, el costo del empaquetado, el de los otros componentes, el diseño del hardware y el desarrollo del software). Si el fabricante desea reducir costos, debe tener en cuenta las herramientas de apoyo con que va a contar: emuladores, simuladores, ensambladores, compiladores, etc. Es habitual que muchos de ellos se decidan por microcontroladores pertenecientes a una única familia, para unificar herramientas de programación, recursos humanos, etc.

**Aplicación.** Antes de seleccionar un microcontrolador es imprescindible analizar los requisitos de la aplicación:

- **Procesamiento de datos.** Puede ser necesario que el microcontrolador realice cálculos críticos en un tiempo limitado. En ese caso, debemos seleccionar un dispositivo suficientemente rápido para ello. Por otro lado, es necesario tener en cuenta la precisión de los datos a manejar: si no es suficiente con un microcontrolador de 8 bits, puede ser necesario acudir a microcontroladores

de 16 ó 32 bits o, incluso, a hardware de coma flotante<sup>2</sup>. Una alternativa más barata y quizá suficiente es usar librerías, para manejar los datos de alta precisión.

- **Entrada/salida.** Para determinar las necesidades de entrada/salida del sistema es conveniente dibujar su diagrama de bloques, de tal forma que sea sencillo identificar la cantidad y el tipo de señales a controlar. Una vez realizado este análisis, puede ser necesario añadir periféricos hardware externos o cambiar a otro microcontrolador más adecuado a ese sistema.
- **Consumo.** Algunos productos que incorporan microcontroladores están alimentados con baterías; su funcionamiento puede ser tan vital como activar una alarma antirrobo. Lo más conveniente, en un caso como éste, puede ser que el microcontrolador esté en estado de bajo consumo pero que despierte ante la activación de una señal (una interrupción) y ejecute el programa adecuado para procesarla.
- **Memoria.** Para detectar las necesidades de memoria de nuestra aplicación debemos separarla en memoria volátil (RAM), memoria no volátil (ROM, EPROM, etc.) y memoria no volátil modificable (EEPROM). Este último tipo de memoria puede ser útil para incluir información específica de la aplicación, como un número de serie o parámetros de calibración. El tipo de memoria a emplear vendrá determinado por el volumen de ventas previsto del producto: de menor a mayor volumen será conveniente emplear EPROM, OTP y ROM. En cuanto a la cantidad de memoria necesaria, puede ser imprescindible realizar una versión preliminar, aunque sea en pseudo-código, de la aplicación, y, a partir de ella, hacer una estimación de cuánta memoria volátil y no volátil es necesaria y si es conveniente disponer de memoria no volátil modificable.
- **Ancho de palabra.** El criterio de diseño debe ser seleccionar el microcontrolador de menor ancho de palabra que satisfaga los requerimientos de la aplicación. Usar un microcontrolador de 4 bits supone una reducción importante en los costes; mientras que uno de 8 bits puede ser el más adecuado si el ancho de los datos es de un byte. Los microcontroladores de 16 y 32 bits, debido a su elevado coste, se reservan para aplicaciones que requieran sus altas prestaciones (Entrada/salida potente o espacio de direccionamiento muy elevado).
- **Diseño de la placa.** La selección de un microcontrolador concreto condiciona el diseño de la placa de circuitos. Debe tenerse en cuenta que, quizás, usar un microcontrolador barato encarezca el resto de componentes del diseño.

<sup>2</sup> Generalmente, los microcontroladores de 8 bits están preparados para trabajar con números enteros y, si necesitamos optimizar esto, tenemos dos posibilidades:

- utilizar un microcontrolador que trabaja con exponente y mantisa (notación científica), llamado de punto flotante, generalmente de 16 o 32 bits; o
- utilizar librerías ya desarrolladas para microcontroladores de 8 bits que trabajan internamente como un microcontrolador de 16 bits. En cuanto al manejo de datos y a las operaciones aritméticas y lógicas, éstas sobrecargan al microcontrolador; pero, logran el objetivo principal: mayor precisión en los resultados, también llamado doble precisión (dos Byte), cuádruple precisión (cuatro Bytes), etc.

### Características de nuestro sistema de control

Es momento de evaluar las características de nuestro sistema de control para satisfacer las necesidades planteadas, consideramos:

Características	Nuestro microcontrolador
Procesamiento de datos	No se requiere gran precisión ni velocidad de procesamiento
Entrada/salida	Siete entradas y tres salidas (Pulsadores, fines de carrera, electroválvulas, etc.).
Consumo	Muy bajo, sin las electroválvulas neumáticas, menor a 150 mA.
Memoria de datos y programa	Menor a 10 bytes de RAM. Menor a 500 bytes de ROM.
Ancho de palabra	Suficiente con 8 bits
Costos del control	Menor a 20 U\$S (Sin las electroválvulas, fines de carrera y pulsadores).
Diseño de la placa	Placa reducida en tamaño y simple faz.
Expansión	Cuatro entradas o tres salidas adicionales.
Adicionales	Canales analógicos, PWM, comunicación RS232, timers, bajo consumo.

La selección del microcontrolador adecuado a nuestro sistema debe responder a los siguientes criterios:

- Los circuitos integrados deben estar disponibles y a precio razonable.
- Las herramientas para desarrollo de hardware deben estar disponibles y a precio razonable.
- Las herramientas para desarrollo de software deben estar disponibles y a precio razonable
- Idealmente, la familia de integrados debe tener una gran variedad de opciones (como entradas/salidas, serie, convertidores analógico/digital, etc.)
- Idealmente, el integrado deberá estar disponible en versiones miniatura y de bajo consumo.
- Idealmente, el integrado deberá tener una arquitectura razonable.

Aunque todas las computadoras poseen los mismos principios y elementos básicos, las hay de diferentes tipos, orientadas a satisfacer diversos propósitos:

- Las **mainframes** son sistemas de computadora muy grandes que se utilizan para trabajos de procesamiento de grandes volúmenes de información.
- Las **PC** son versiones reducidas de las mainframes, aplicables a tareas de menor envergadura aún, tales como, por ejemplo, procesadores de texto o gráficos.
- Los **microcontroladores** son computadoras en un solo circuito integrado, utilizado para controlar una pequeña aplicación.
- Los **pequeños microcontroladores** se usan, por ejemplo, para convertir el movimiento de un mouse de computadora en una salida serie de datos, para ingresarlo a una PC. Muy frecuentemente los microcontroladores se hallan embebidos dentro de un producto y quien lo usa no necesariamente sabe que, en su interior, alberga una computadora.

Al estar todos los microcontroladores integrados en un chip, su estructura fundamental y sus características básicas son muy parecidas. Todos deben disponer de

los bloques esenciales: procesador, memoria de datos y de instrucciones, líneas de E/S, oscilador de reloj y módulos controladores de periféricos. Sin embargo, cada fabricante intenta enfatizar los recursos más idóneos para las aplicaciones a las que se destinan preferentemente.

Una de las razones del éxito de los mC se basa en su utilización. Cuando se aprende a manejar uno de ellos, conociendo su arquitectura y su repertorio de instrucciones, es muy fácil emplear otro modelo.

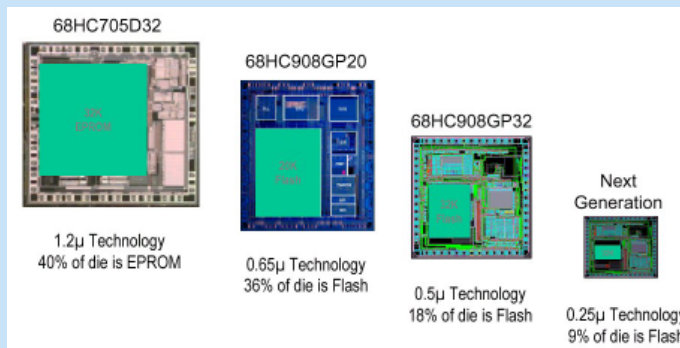
Al igual que la mayoría de las computadoras, los microcontroladores son simples ejecutores de instrucciones de propósito general. La verdadera estrella de un sistema de computadora es el programa de instrucciones que son provistas por un programador humano. Este programa instruye a la computadora a realizar largas secuencias de muy simples acciones para efectuar tareas útiles tales como las que se propuso el programador.

Para la decisión final a nuestro problema se plantearon tres posibilidades:

- los microcontroladores 8051 de INTEL o ATMEL,
- los PIC de MICROCHIPS
- la familia HC08 de Motorola.

Por trayectoria, presencia en el mercado nacional y soporte técnico hemos seleccionado a la familia de microcontroladores de MOTOROLA que está en continua evolución.

Consideremos la evolución del tamaño del “die” o pastilla semiconductora de Motorola:



Estos microcontroladores se aplican en elementos cotidianos:



Redes industriales: protocolo de bajo costo CAN. USB sensor-actuator interface. Periféricos para seguridad: IR infrarrojo detección y comunicación. Balastro regulador de luz: protocolo para Domótica.



Volvamos a nuestro problema...

La elección recae en un microcontrolador que cumple muy bien dichos criterios y posee una presencia en el mercado importante, como son los microcontroladores de Motorola, la familia 68HC08 con una gran variedad de modelos.

De entre estos modelos seleccionamos el 68HC908KX8, versión que posee entradas/salidas digitales y analógicas, así como comunicación RS232, PWM y temporización.

Los microcontroladores más populares son<sup>3</sup>:

- **8048 (Intel).** Es el padre de los microcontroladores actuales, el primero de todos. Su precio, disponibilidad y herramientas de desarrollo hacen que todavía sea muy popular.
- **8051 (Intel, Atmel y otros).** Es, sin duda, el microcontrolador más popular. Fácil de programar y potente. Está bien documentado y posee cientos de variantes e incontables herramientas de desarrollo.
- **80186, 80188 y 80386 EX (Intel).** Versiones en microcontrolador de los populares microprocesadores 8086 y 8088. Su principal ventaja es que permiten aprovechar las herramientas de desarrollo para PC.
- **68HC11/HC08 (Motorola y Toshiba).** Es un microcontrolador de 8 bits potente y popular con gran cantidad de variantes.
- **683xx (Motorola).** Surgido a partir de la popular familia 68000, a la que se incorporan algunos periféricos. Son microcontroladores de altísimas prestaciones.
- **PIC (MicroChip).** Familia de microcontroladores que gana popularidad día a día. Fueron los primeros microcontroladores RISC.

### Almacenamiento y ejecución del programa

Un programa es una lista de instrucciones al procesador. Todos los  $\mu P$  y  $\mu C$  cuentan con un conjunto de instrucciones que pueden ejecutar. Cada tipo de procesador y controlador tiene su propio conjunto de instrucciones y, por lo general, los programas escritos para un dispositivo no funcionan en otro de distinta marca.

Un  $\mu C$  típico tiene instrucciones para:

- transferir información entre registros y memoria;
- realizar operaciones aritméticas y lógicas;
- efectuar comparaciones y pruebas sobre el contenido de sus registros de memoria;
- controlar la secuencia de ejecución de programas.

Por lo general, la operación que una instrucción ha de ejecutar está definida por un **código de operación** (en  $\mu C$  sencillos, generalmente de un solo byte) conocido también como **opcode**.

<sup>3</sup> Es preciso resaltar en este punto que existen innumerables familias de microcontroladores, cada una de las cuales posee un gran número de variantes.

Algunas instrucciones requieren, además del opcode, información extra (**operandos**). Por ejemplo, una instrucción para almacenar el contenido de un registro en una posición de memoria, necesita incluir la dirección de memoria de destino.

La unidad de control y decodificación de instrucciones es el corazón del procesador. Es la encargada de extraer de forma secuencial las instrucciones de la memoria y, luego, ejecutarlas.

Unido a la unidad de control se encuentra un **generador de reloj**, que utiliza un oscilador para producir una señal de reloj muy precisa. El tiempo se divide en un cierto número de ciclos de reloj.

El funcionamiento de la unidad de control y decodificación de instrucciones se puede dividir en dos partes (ambas duran varios ciclos de reloj):

- **Ciclo de búsqueda de instrucciones.** En él se transfiere la dirección de la siguiente instrucción al bus de direcciones, se envía una orden de lectura a los dispositivos de memoria a través del bus de control, se lee la información del bus de datos y, si se trata del primer byte de una instrucción, se transfiere este byte al registro de instrucciones.
- **Ciclo de ejecución de instrucciones.** En él se ejecuta la instrucción. El registro de instrucciones está conectado a un decodificador que determina cuántos bytes de información adicionales se requieren. Éstos se cargan mediante ciclos de búsqueda, como ya indicamos.

Cuando la ejecución está completa, la máquina comienza, automáticamente, el ciclo de búsqueda de la siguiente instrucción del programa.

La ejecución es, por tanto, una secuencia continua de ciclos de búsqueda y ejecución.

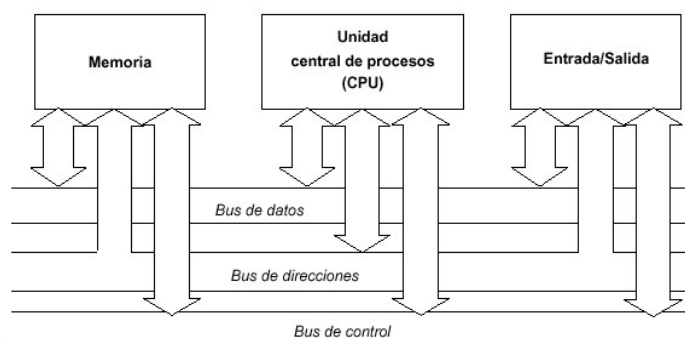
## Recursos comunes a todos los microcontroladores

Nos referiremos en este punto a decisiones respecto de:

- arquitectura básica,
- procesador o CPU,
- memoria,
- puertas de entrada y salida
- reloj principal.

### 1. Arquitectura básica

Aunque, inicialmente, todos los microcontroladores adoptan la arquitectura clásica de Von Neumann, hoy día hay mayor tendencia hacia la arquitectura Harvard.



### Arquitectura

#### de Von Neumann

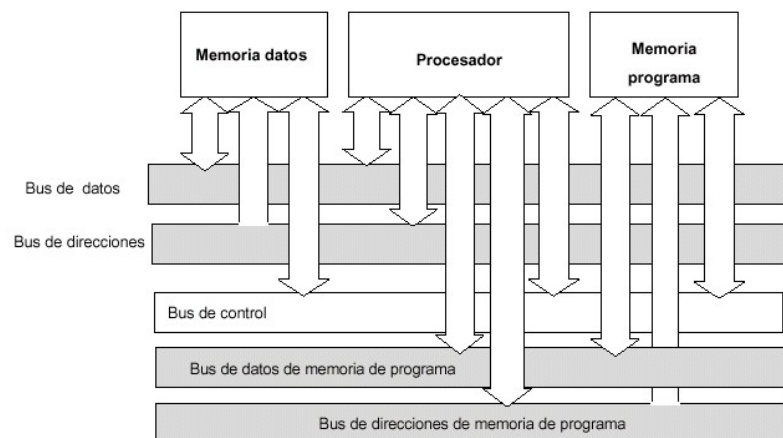
Se caracteriza por disponer de una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. Se accede a esta memoria a través de un sistema de buses único (direcciones, datos y control).

- **Bus de datos.** Se emplea para transferir datos. El número de líneas de este bus es igual a la longitud de palabra del dispositivo.
- **Bus de direcciones.** Permite transferir información de direcciones. El número de líneas en el bus de direcciones determina el número de posiciones de memoria que el procesador puede especificar. Un bus de direcciones de 8 líneas es capaz de posicionar sólo  $2^8$  (256) direcciones.
- **Bus de control.** El procesador utiliza las líneas del bus de control para sincronizar operaciones con componentes externos.

Por ejemplo:

Si el procesador o microcontrolador desea almacenar una palabra de datos en una ubicación de memoria particular, coloca los datos en el bus de datos, la dirección en la que se van a almacenar los datos en el bus de direcciones y diversas señales de control para sincronizar la operación de almacenamiento en el bus de control.

La necesidad de conseguir rendimientos elevados en el procesamiento de las instrucciones ha desembocado en el empleo cada vez mayor de procesadores de arquitectura **Harvard**, frente a los tradicionales que siguen la arquitectura de **Von Neumann**. Mientras esta última –como le planteábamos en la primera parte del curso– se caracteriza porque la CPU se conecta a una memoria única donde coexisten datos e instrucciones a través de un sistema de buses, en la arquitectura Harvard son independientes la memoria de instrucciones y la memoria de datos, y cada una dispone de su propio sistema de buses para el acceso, lo que propicia el paralelismo:



La arquitectura Harvard dispone de dos memorias independientes:

- una que contiene sólo instrucciones y
- otra, sólo datos.

Ambas disponen de sus respectivos sistemas de buses de acceso y es posible realizar operaciones de acceso (lectura o escritura) simultáneamente en ambas memorias.

## 2. Procesador o CPU

Es el elemento más importante del microcontrolador y determina sus principales características, tanto a nivel hardware como software.



Se encarga de direccionar la memoria de instrucciones, recibir el código OP de la instrucción en curso, su decodificación y la ejecución de la operación que implica la instrucción, así como la búsqueda de los operandos y el almacenamiento del resultado.

Existen tres orientaciones en cuanto a la arquitectura y funcionalidad de los procesadores actuales.

- **CISC.** Un gran número de procesadores usados en los microcontroladores está basado en la filosofía CISC –computadores de juego de instrucciones complejo–. Disponen de más de 80 instrucciones de máquina en su repertorio, algunas de las cuales son muy sofisticadas y potentes, requiriendo muchos ciclos para su ejecución. Una ventaja de los procesadores CISC es que ofrecen al programador instrucciones complejas que actúan como macros.
- **RISC.** Tanto la industria de los computadores comerciales como la de los microcontroladores están decantándose hacia la filosofía RISC –computadores de juego de instrucciones reducido–. En estos procesadores, el repertorio de instrucciones máquina es muy reducido y las instrucciones son simples y, generalmente, se ejecutan en un ciclo. La sencillez y rapidez de las instrucciones permiten optimizar el hardware y el software del procesador.
- **SISC.** En los microcontroladores destinados a aplicaciones muy concretas, el juego de instrucciones, además de ser reducido, es específico: las instrucciones se adaptan a las necesidades de la aplicación prevista –computadores de juego de instrucciones específico–.

En nuestro caso, hemos seleccionado un microcontrolador CISC con un juego de 119 instrucciones, muchas de ellas con varios modos de direccionamiento, lo que permite tener una gran flexibilidad en cuanto a la tarea a realizar.

Permite ejecutar acciones directas de entradas y salidas (como las necesarias para nuestro sistema de control), así como desarrollar complejos algoritmos, linealizaciones y parametrización de variables (multiplicar y dividir de forma sencilla), complejos protocolos de comunicación con otros dispositivos de control de mayores prestaciones, etc.

En la implementación hecha por Motorola de esta arquitectura, hay algunos registros o latch especiales (denominados registros del CPU) en el interior del CPU, que actúan como pequeños anotadores de borrador y tablero de control del CPU.

Los registros del CPU son similares a las memorias en que la información puede ser escrita y retenida en su interior. Es importante recordar que ellos están cableados en el CPU (**registros internos**) y no forman parte de la memoria direccionable por la CPU (**registros externos**).

Toda la información (no la de los registros de la CPU) accesible a la CPU es vista (por la CPU) como una única fila de miles de casillas o más. A veces esta organización es llamada sistema de **I/O mapeado en memoria**, dado que la CPU trata a toda posición de memoria por igual, aunque ella contenga instrucciones de programa, variables de datos o entradas / salidas (I/O) de control.

La información del estado y control de I/O (input/output o entrada/salida) es un tipo de posición de memoria que permite al sistema del computador intercambiar información desde o hacia su mundo exterior. Este tipo de memoria es poco usual, ya que la información puede ser “monitoreada” y/o alterada por algún otro dispo-



sitivo, además de la CPU. Los tipos más simples de posiciones de memoria de I/O son un simple “port” (puerto) de entrada y un simple port de salida. En un MCU de 8 bits, un simple port de entrada (input) o salida (output) contiene 8 pines (por lo general un port típico tiene 8 pines; pero, como veremos más adelante, en algunos MCU la cantidad de pines de entrada o salida puede ser inferior a 8 en algún port en particular). En la práctica, la implementación de un port de salida es un óctuple *data latch* con realimentaciones que permitan leerlo a la CPU.

Los microcontroladores M68HC08 incluyen pines de I/O de ports paralelos de propósito general. El sentido de cada pin se puede programar por un bit de control accesible por programa:

- Un pin de port se configura como salida, si en su correspondiente DDR bit es escrito un uno lógico.
- Un pin de port se configura como entrada si en su correspondiente DDR bit es cero.

Luego de un reset, todos los DDR bits son borrados, lo que configura a todos los pines como entrada. Los DDR pueden ser tanto escritos como leídos por el procesador.

### 3. Memoria

En los microcontroladores, la memoria de instrucciones y datos está integrada en el propio chip:

- Una parte es no volátil, tipo ROM, y se destina a contener el programa de instrucciones que gobierna la aplicación.
- Otra parte de memoria es tipo RAM, volátil, y se destina a guardar las variables y los datos.

Hay dos peculiaridades que diferencian a los microcontroladores de los computadores personales:

- En los  $\mu C$  no existen sistemas de almacenamiento masivo como disco duro o disquetes.
- Como los  $\mu C$  sólo se destinan a una tarea, en la memoria ROM sólo hay que almacenar un único programa de trabajo.

La RAM en estos dispositivos es de poca capacidad, pues sólo debe contener las variables y los cambios de información que se produzcan en el transcurso del programa.

Por otra parte, como sólo existe un programa activo, no se requiere guardar su copia en la RAM pues se ejecuta directamente desde la ROM.

Los usuarios de computadores personales están habituados a manejar Megabytes de memoria; pero, los diseñadores con microcontroladores trabajan con capacidades:

- de ROM comprendidas entre 2 k bytes y 64 k bytes, y
- de RAM comprendidas entre 20 bytes y 2 K bytes.

Según el tipo de memoria ROM que dispongan los microcontroladores, su aplicación y utilización son diferentes. Las cinco versiones de memoria no volátil que se pueden

encontrar en los microcontroladores del mercado entre los cuales hemos seleccionado el mejor para nuestro problema, son:

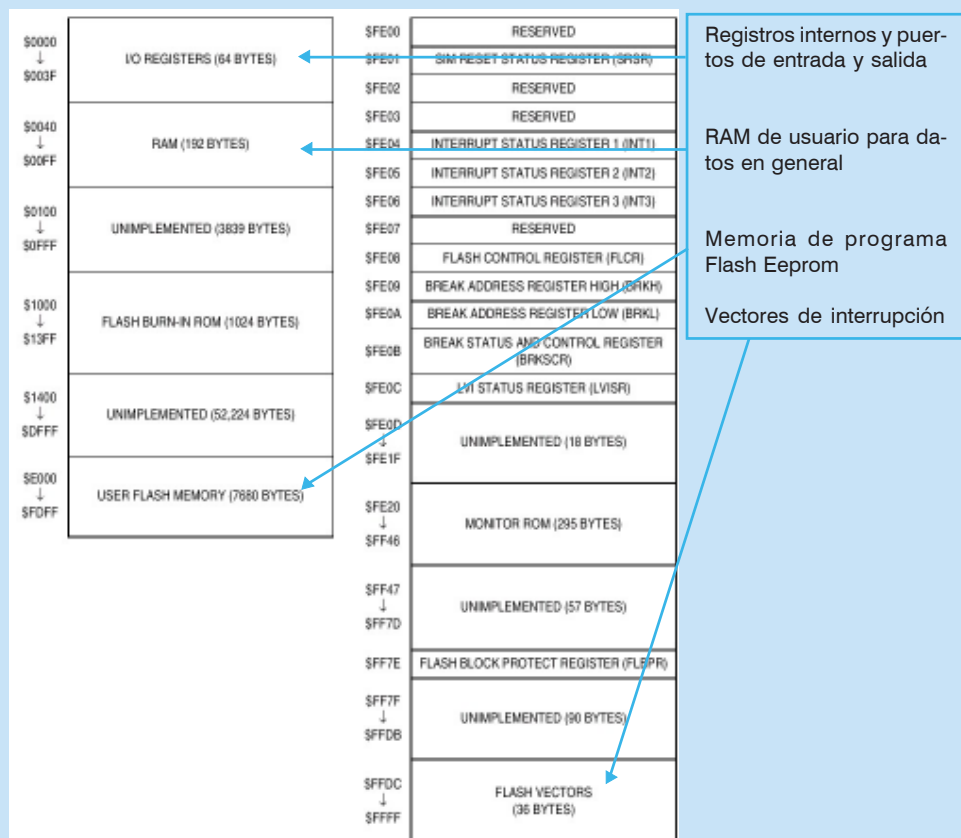
- **ROM con máscara.** Es una memoria no volátil de sólo lectura, cuyo contenido se graba durante la fabricación del chip. El elevado coste del diseño de la máscara sólo hace aconsejable el empleo de los microcontroladores con este tipo de memoria cuando se precisan cantidades superiores a varios miles de unidades.
- **OTP –One Time Programmable–.** El microcontrolador contiene una memoria no volátil de sólo lectura, “programable una sola vez” por el usuario. Es el usuario quien puede escribir el programa en el chip mediante un sencillo grabador controlado por un programa desde un PC. La versión OTP es recomendable cuando es muy corto el ciclo de diseño del producto, o bien en la construcción de prototipos y series muy pequeñas. Tanto en este tipo de memoria como en la EPROM, se suele usar la encriptación mediante fusibles para proteger el código contenido.
- **EPROM –Erasable Programmable Read Only Memory–.** Los microcontroladores que disponen de memoria EPROM pueden borrarse y grabarse muchas veces. La grabación se realiza, como en el caso de los OTP, con un grabador gobernado desde un PC. Si, posteriormente, se desea borrar el contenido, disponen de una ventana de cristal de cuarzo en su superficie por la que se somete a la EPROM a rayos ultravioleta durante varios minutos. Las cápsulas son de material cerámico, por lo que son más caros que los microcontroladores con memoria OTP, los que están hechos con material plástico.
- **EEPROM –Electrical Erasable Programmable Read Only Memory–.** Se trata de memorias de sólo lectura, programables y borrables eléctricamente. Tanto la programación como el borrado se realizan eléctricamente desde el propio grabador y bajo el control programado de un PC. Las operaciones de grabado y de borrado son muy cómodas y rápidas. No disponen de ventana de cristal de cuarzo en la superficie. Los microcontroladores dotados de memoria EEPROM, una vez instalados en el circuito, pueden grabarse y borrarse cuantas veces se quiera, sin ser retirados de dicho circuito. Para ello se usan “grabadores en circuito” que confieren una gran flexibilidad y rapidez a la hora de realizar modificaciones en el programa de trabajo. El número de veces que puede grabarse y borrarse una memoria EEPROM es finito, por lo que no es recomendable una reprogramación continua. Son muy idóneas para la enseñanza y la ingeniería de diseño. Por todas estas ventajas, se va extendiendo en los fabricantes la tendencia de incluir una pequeña zona de memoria EEPROM en los circuitos programables para guardar y modificar cómodamente una serie de parámetros que adecuan el dispositivo a las condiciones del entorno. Este tipo de memoria es relativamente lento.
- **FLASH.** Se trata de una memoria no volátil, de bajo consumo, que se puede escribir y borrar. Funciona como una ROM y una RAM; pero, consume menos y es más pequeña. A diferencia de la ROM, la memoria FLASH es programable en el circuito. Es más rápida y de mayor densidad que la EEPROM. La alternativa FLASH está recomendada frente a la EEPROM, cuando se precisa gran cantidad de memoria de programa no volátil. Es más veloz y tolera más ciclos de escritura/borrado. Las memorias EEPROM y FLASH son muy útiles al permitir que los microcontroladores que las incorporan puedan ser reprogramados “en circuito”, es decir, sin tener que sacar el circuito integrado de la tarjeta. Así, un dispositivo

con este tipo de memoria incorporado al control del motor de un automóvil permite que pueda modificarse el programa durante la rutina de mantenimiento periódico, compensando los desgastes y otros factores tales como la compresión, la instalación de nuevas piezas, etc. La reprogramación del microcontrolador puede convertirse en una labor rutinaria dentro de la puesta a punto.

Hay miles de posiciones de memoria o más en un sistema de MCU.

Es importante, entonces, contar con un medio conveniente para no perder de vista su contenido. Un mapa de memoria es una representación gráfica de la totalidad de la memoria del MCU.

Veamos el mapa de memoria de nuestro microcontrolador:



Las memorias pueden ser una forma de periféricos. Ya hemos discutido el uso de diferentes tipos de memoria, pero sin considerar la lógica requerida para soportarlas.

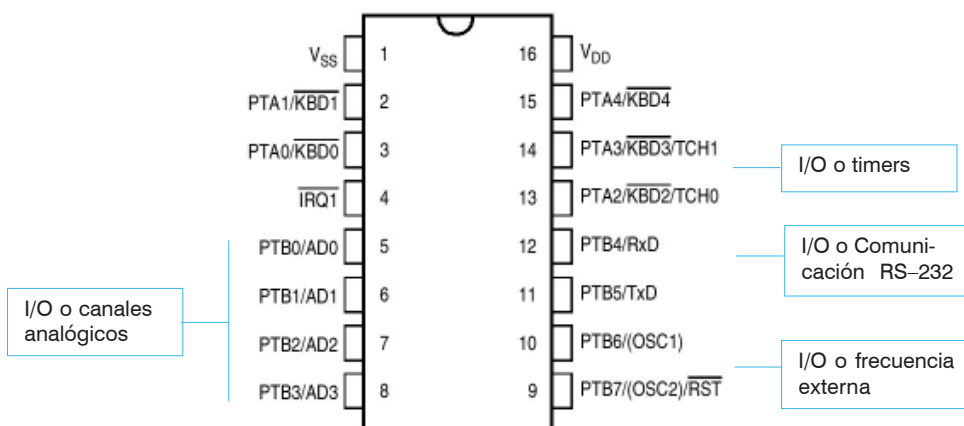
- Las ROM y RAM son muy íntegras y no requieren lógica de soporte más que la lógica de selección de direcciones para distinguir una posición de otra. Esta lógica es provista en el mismo chip que aloja a la memoria.
- Las memorias EPROM (PROM borrable) y EEPROM (PROM borrable eléctricamente) requieren una lógica de soporte para grabarlas (y borrarlas en las EPROM).

La lógica periférica de soporte para el MC68HC908KX8 es similar a un grabador de PROM, contenido en la misma MCU. Un registro de control incluye bits de control para seleccionar entre los modos de grabación y lectura, y para habilitar la fuente de alimentación de programación de alta tensión.

#### 4. Puertas de entrada y salida

La principal utilidad de las patitas que posee la cápsula que contiene un microcontrolador es soportar las líneas de E/S que comunican al computador interno con los periféricos exteriores.

Según los controladores de periféricos que posea cada modelo de microcontrolador, las líneas de E/S se destinan a proporcionar el soporte a las señales de entrada, salida y control.



Pinout del microcontrolador que utilizaremos MC68HC908KX8

Se pueden observar al menos trece líneas que pueden trabajar como entradas o salidas, sin contar la de interrupción (IRQ1)

#### 5. Reloj principal

Todos los microcontroladores disponen de un circuito oscilador que genera una onda cuadrada de alta frecuencia, que configura los impulsos de reloj usados en la sincronización de todas las operaciones del sistema.

Generalmente, el circuito de reloj está incorporado en el microcontrolador y sólo se necesitan unos pocos componentes exteriores para seleccionar y estabilizar la frecuencia de trabajo. Dichos componentes suelen consistir en un cristal de cuarzo junto a elementos pasivos o bien un resonador cerámico o una red R-C. Aumentar la frecuencia de reloj supone disminuir el tiempo en que se ejecutan las instrucciones, pero lleva aparejado un incremento del consumo de energía.

En nuestro problema, encontramos un sistema de clock completo integrado que puede funcionar sin un cristal exterior y puede trimearse para acercarlo a los valores buscados con tolerancias aceptables.

#### Recursos especiales

Cada fabricante oferta numerosas versiones respecto de la arquitectura básica de microcontrolador que acabamos de presentarle. En algunas amplía las capacidades

de las memorias, en otras incorpora nuevos recursos, en otras reduce las prestaciones al mínimo para aplicaciones muy simples, etc. La labor del diseñador es encontrar el modelo que satisfaga todos los requerimientos de su aplicación, minimizando el coste, el hardware y el software.

Los principales recursos específicos que incorporan los microcontroladores son:

1. Temporizadores –*Timers*–
2. Perro guardián –*Watchdog* o COP–
3. Protección ante fallo de alimentación –*Brownout* o LVI–
4. Estado de reposo o de bajo consumo
5. Conversor A/D –ADC–
6. Conversor D/A –DAC–
7. Comparador analógico
8. Modulador de anchura de impulsos –PWM–
9. Puertas de E/S digitales
10. Puertas de comunicación

### 1. Temporizadores –*Timers*–

Se emplean para controlar periodos de tiempo (temporizadores) y para llevar la cuenta de acontecimientos que suceden en el exterior (contadores).

Para la medida de tiempos, se carga un registro con el valor adecuado y, a continuación, dicho valor se va incrementando o decrementando al ritmo de los impulsos de reloj o algún múltiplo, hasta que se desborde y llegue a 0, momento en el que se produce un aviso.

Cuando se desea contar acontecimientos que se materializan por cambios de nivel o flancos en alguna de las patitas del microcontrolador, el mencionado registro se va incrementando o decrementando al ritmo de dichos impulsos.

### 2. Perro guardián –*Watchdog* o COP–

Cuando el computador personal se bloquea por un fallo del software u otra causa, se pulsa el botón del reset y se reinicia el sistema. Pero, un microcontrolador funciona sin el control de un supervisor y de forma continuada las 24 horas del día, por lo que requiere este recurso especial.

El *perro guardián* consiste en un temporizador que, cuando se desborda y pasa por 0, provoca un reset –automáticamente– en el sistema.

Se diseña el programa de trabajo que controla la tarea, de forma tal que refresca o inicializa al perro guardián antes de que provoque el reset. Si falla el programa o se bloquea, no se refresca al *perro guardián* y, al completar su temporización, “ladrará y ladrará” hasta provocar el reset.

### 3. Protección ante fallo de alimentación –*Brownout* o LVI–

Se trata de un circuito que resetea al microcontrolador cuando el voltaje de alimentación (VDD) es inferior a un voltaje mínimo –*brownout*–. Mientras el voltaje de alimentación es inferior al de *brownout*, el dispositivo se mantiene reseteado, comenzando a funcionar normalmente cuando sobrepasa dicho valor.

Su denominación LVI corresponde a *Interrupción por bajo nivel de tensión*.

#### 4. Estado de reposo o de bajo consumo

Son abundantes las situaciones reales de trabajo en las que el microcontrolador debe esperar, sin hacer nada, a que se produzca algún acontecimiento externo que lo ponga de nuevo en funcionamiento.

Para ahorrar energía –factor clave en los aparatos portátiles–, los microcontroladores disponen de una instrucción especial (*Sleep* en los PIC; *Stop wait, Sleep* en Motorola) que los pasa al estado de reposo o de bajo consumo, en el cual los requerimientos de potencia son mínimos. En dicho estado se detiene el reloj principal y se “congelan” sus circuitos asociados, quedando sumido el mC en un profundo “sueño”. Al activarse una interrupción ocasionada por el acontecimiento esperado, el microcontrolador se despierta y reanuda su trabajo (En Motorola –dependiendo de cuál se utiliza–, puede quedar algún circuito interno funcionando).

#### 5. Conversor A/D –ADC–

Los microcontroladores que incorporan un conversor A/D (analógico/digital) pueden procesar señales analógicas, tan abundantes en las aplicaciones.

Un multiplexor permite aplicar a la entrada del ADC diversas señales analógicas, desde las patitas del circuito integrado.

#### 6. Conversor D/A –DAC–

Transforma los datos digitales obtenidos del procesamiento del computador en su correspondiente señal analógica, que saca al exterior por una de las patitas de la cápsula. Existen muchos sistemas de control que trabajan con señales analógicas.

En algunos microcontroladores, dicha señal analógica puede generarse con la salida PWM debidamente filtrada.

#### 7. Comparador analógico

Algunos modelos de microcontroladores disponen internamente de un **amplificador operacional** que actúa como comparador entre una señal fija de referencia y otra variable que se aplica por una de las patitas de la cápsula.

La salida del comparador proporciona un nivel lógico 1 ó 0, según una señal sea mayor o menor que la otra.

También hay modelos de microcontroladores con un módulo de tensión de referencia que proporciona diversas tensiones de referencia que se pueden aplicar en los comparadores (por ejemplo: AT80251 de Atmel).

#### 8. Modulador de ancho de pulso o PWM

Son circuitos que proporcionan en su salida impulsos de anchura variable, que se ofrecen al exterior a través de las patitas del encapsulado.

Permiten implementar salidas analógicas si son filtradas con un capacitor y una resistencia.

Además, se logra comandar servos de aeromodelismo –que son muy aplicados en robótica– y generar control de ángulo para señales de alterna de 220 Vac con triacs.

## 9. Puertas de E/S digitales

Todos los microcontroladores destinan algunas de sus patitas a soportar líneas de E/S digitales. Por lo general, estas líneas se agrupan de ocho en ocho, formando **puertas**.

Las líneas digitales de las puertas pueden configurarse como Entrada o como Salida, cargando un 1 ó un 0 en el bit correspondiente de un registro destinado a su configuración.

## 10. Puertas de comunicación

Con objeto de dotar al microcontrolador de la posibilidad de comunicarse con otros dispositivos externos, otros buses de microprocesadores, buses de sistemas, buses de redes y poder adaptarlos con otros elementos bajo otras normas y protocolos, algunos modelos disponen de recursos que permiten directamente esta tarea, entre los que destacan:

- UART, adaptador de comunicación serie asíncrona.
- USART, adaptador de comunicación serie síncrona y asíncrona.
- Puerta paralela esclava para poder conectarse con los buses de otros microprocesadores.
- USB –*Universal Serial Bus*–, moderno bus serie para los PC.
- Bus I<sup>2</sup>C, que es un interfaz serie de dos hilos desarrollado por Philips.
- CAN –*Controller Area Network*– para permitir la adaptación con redes de conexión multiplexado desarrollado conjuntamente por Bosch e Intel para el cableado de dispositivos en automóviles (En EE.UU. se usa el J1850).

En nuestro microcontrolador encontramos características especiales que, en su mayoría, no serán utilizadas para resolver nuestra problemática, pero que es importante conocer para dimensionar el potencial de este pequeño componente electrónico.

La nueva línea HC908, basa su funcionamiento en la poderosa arquitectura del procesador central CPU08 (común a toda la familia MC68HC08).

Este procesador es del tipo arquitectura Von Neumann, con mapa lineal de memoria sin diferenciación entre memoria de datos (RAM) y programa (FLASH), con lo cual no existen limitaciones en el uso de instrucciones, tanto para la memoria de datos, como para la de programa (No existen instrucciones distintas para Datos –RAM– o para Programa –FLASH–).

Características especiales:

- Selección de oscilador a XTAL clásico, Pin oscilador externo (OSC1) u oscilador interno ajustable por soft.
- Conversor A/D de 8 bits, 4 canales multiplexados (tipo radiométrico).
- Comunicación asíncrona para implementar comunicación del tipo RS232.
- Timer de 16 bits y 2 canales c/u con selección de modos *Input Capture*, *Output Compare*, y funciones PWM o salidas analógicas.
- 13 pines I/O de propósitos generales.
- COP watchdog reset.



- LVI –*Low Voltage Inhibit*– con puntos de disparo seleccionables según tensión de operación (3V o 5V).
- Modos de bajo consumo *Stop* y *Wait*.
- Puertos con características para implementar interrupciones por teclado.

## Herramientas para el desarrollo de aplicaciones

Entre todos los demás, uno de los factores que más importancia tiene a la hora de seleccionar un microcontrolador es el soporte –tanto software como hardware– de que dispone. Un buen conjunto de herramientas de desarrollo puede ser decisivo en la elección, ya que supone una ayuda inestimable en el desarrollo del proyecto.

Las principales herramientas de ayuda al desarrollo de sistemas basados en microcontroladores son:

- **Ensamblador.** La programación en lenguaje ensamblador puede resultar un tanto ardua para el principiante; pero, permite desarrollar programas muy eficientes, ya que otorga al programador el dominio absoluto del sistema. Los fabricantes suelen proporcionar el programa ensamblador de forma gratuita y, en cualquier caso, siempre se puede encontrar una versión gratuita para los microcontroladores más populares.
- **Compilador.** La programación en un lenguaje de alto nivel (como el C); permite disminuir el tiempo de desarrollo de un producto. No obstante, si no se programa con cuidado, el código resultante puede ser mucho más ineficiente que el programado en ensamblador. Las versiones más potentes suelen ser muy caras; aunque, para los microcontroladores más populares pueden encontrarse versiones *demo* limitadas e, incluso, compiladores gratuitos.
- **Depuración.** Debido a que los microcontroladores van a controlar dispositivos físicos, los desarrolladores necesitan herramientas que les permitan comprobar el buen funcionamiento del microcontrolador, cuando es conectado al resto de circuitos.
- **Simulador.** Son capaces de ejecutar en un PC programas realizados para el microcontrolador. Los simuladores permiten tener un control absoluto sobre la ejecución de un programa, siendo ideales para su depuración. Su gran inconveniente es que es difícil simular la entrada y salida de datos del microcontrolador. Tampoco cuentan con los posibles ruidos en las entradas; pero, al menos, permiten el paso físico de la implementación de un modo más seguro y menos costoso, puesto que ahorramos en grabaciones de chips para la prueba *in-situ*.
- **Placas de evaluación.** Se trata de pequeños sistemas con un microcontrolador ya montado, que suelen conectarse a un PC desde el que se cargan los programas que se ejecutan en el microcontrolador. Las placas suelen incluir visualizadores LCD, teclados, LED, fácil acceso a los pines de E/S, etc. El sistema operativo de la placa recibe el nombre de **programa monitor**. El programa monitor de algunas placas de evaluación, aparte de permitir cargar programas y datos en la memoria del microcontrolador, posibilita realizar, en cualquier momento, ejecuciones paso a paso, monitorizar el estado del microcontrolador o modificar los valores almacenados en los registros o en la memoria.



- **Emulador en circuito.** Se trata de un instrumento que se coloca entre el PC anfitrión y el zócalo de la tarjeta de circuito impreso donde se alojará el microcontrolador definitivo. El programa es ejecutado desde el PC; pero, para la tarjeta de aplicación es como si lo hiciese el mismo microcontrolador que luego irá en el zócalo. Presenta en pantalla toda la información tal y como luego sucederá cuando se coloque la cápsula.

### ***Ejemplos de microcontroladores, usos y aplicaciones***

Gran parte de los fabricantes de circuitos integrados dispone de su propia línea de microcontroladores.

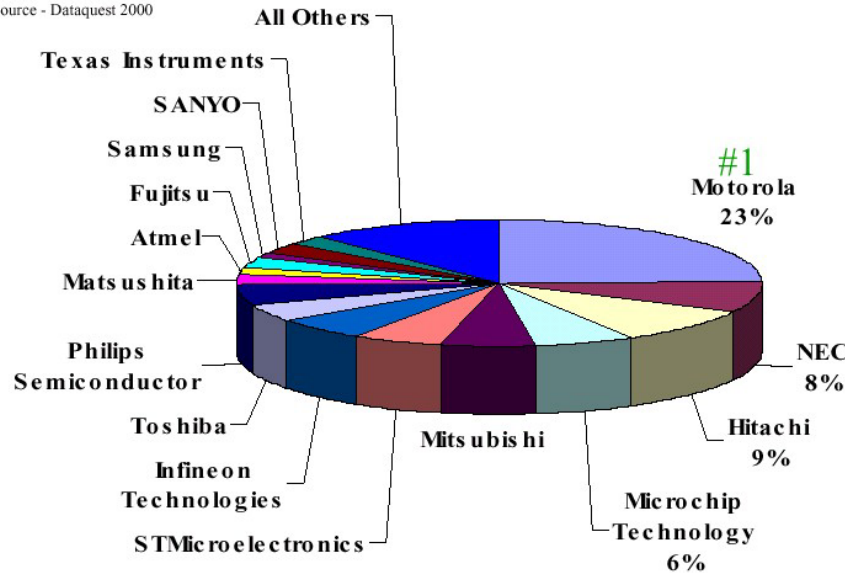
En la tabla reseñamos los fabricantes más destacados:

Fabricante	Modelo de $\mu C$
Intel	8048, 8051, 80C196, 80186, 80188, 80386EX
Motorola	6805, 68HC11, 68HC08, 68HC16, 683XX
Hitachi	HD64180
Philips	Gama completa de clónicos del 8051 / 52
SGS-Thomson (ST)	ST-62XX
Microchip	PICs (Familias 12, 14, 16, 17 y 18)
National Semiconductor	COP8, derivados de 8051/ 8052
Zilog	Z8, Z80
Texas Instruments	TMS370, derivados de 8051/ 8052
Toshiba	TLCS-870
Infineon	C500
Dallas	DS5000
Nec	78K
Atmel	AT91, AT89S8252, derivados de 8051/ 8052

Cada fabricante de microcontroladores oferta un elevado número de modelos diferentes, desde los más sencillos hasta los más poderosos, de forma que es posible seleccionar la capacidad de la memoria, el número de líneas de E/S, la cantidad y potencia de elementos auxiliares, la velocidad de funcionamiento, etc.

En la figura se muestra el ranking de ventas en Europa de microcontroladores de 8 bits:

Source - Dataquest 2000



Se considera a **Intel** como el padre de los microcontroladores y al 8048 como el primer microcontrolador de 8 bits (fabricado por Intel en la década de los '70). Otra de las principales empresas del mundo de dispositivos programables es **Motorola**, que dispone de potentes microcontroladores como son las familias 68HC11 y 68HC08.

Los microcontroladores PIC de la empresa americana **Microchip** se emplean en la actualidad cada vez más, debido a su bajo coste, facilidad de uso, y la abundancia de información y herramientas de apoyo.

#### Microprocesador

- Computadores (ordenadores personales).
- Controladores buses industriales.

#### Microcontrolador:

- Electrodomésticos.
- Tarjetas de control de acceso.
- Juguetes.
- Automación.
- TY, audio y video.
- Máquinas y equipos industriales

#### DSP –procesador digital de señal–:

- Reconocimiento de voz, procesamiento de señales acústicas.
- Módem.

#### Usos y aplicaciones de microprocesadores:



Usos y aplicaciones de microcontroladores:



Usos y aplicaciones del DSP:



Los microcontroladores en uso más comunes, por fabricante, son:

- **Motorola. 8-bit:** 68HC05, 68HC08, 68HC11. **16 bit:** 68HC12, 68HC16. **32-bit:** 683xx
- **Microchip. 12-bit:** Instruction PIC. **14-bit:** Instruction PIC y HT8 PIC16F84. **16-bit:** Instruction PIC
- **Atmel. AVR**
- **Hitachi. Hitachi H8**
- **Holtek. HT8**
- **Intel. 8-bit:** 8XC42, MCS51, 8xC251. **16-bit:** MCS96, MXS296
- **National Semiconductor. COP8**
- **NEC. 78K**
- **ST. ST 62, ST 7**
- **Texas Instruments. TMS370**
- **Zilog. Z8, Z86E02**
- **Genérico.** Algunas arquitecturas de microcontrolador están disponibles por tal cantidad de vendedores y en tantas variedades, que podrían tener, con total corrección, su propia categoría. Entre ellos encontramos, principalmente, las variantes de Intel 8051 y Z80.

