

**Fascículo**

**4**



**Trae tus sueños...  
juntos los realizaremos!**

***Facultad de Universidad Abierta y a Distancia  
Fundación Universitaria San Martín***

**Procesadores**

**Semestre 6**



¡Que tus sueños...  
lleguen los realizarse!

Facultad de Universidad Abierta y a Distancia  
Fundación Universitaria San Martín

## Procesadores

## Tabla de contenido

### Página

Introducción	1
Conceptos previos	2
Mapa conceptual Fascículo 4	3
Logros	3
<b>Descripción interna del Microcontrolador Motorola</b>	<b>3</b>
Arquitectura Interna del Microcontrolador	3
Sistemas Embebidos	10
Puertos de entrada y salida digital	12
Volúmenes	13
Entornos de Programación	15
La latencia de interrupción	16
Resumen	23
Bibliografía recomendada	25
Nexo	25
Seguimiento al autoaprendizaje	27

**Créditos: 1**

**Tipo de asignatura: Teórico – Práctica**

## Tabla de convenciones

	Logros
	Actividad
	Observación
	Actividad de trabajo colaborativo
	Resumen
	Bibliografía recomendada
	Nexo
	Seguimiento al autoaprendizaje

Copyright©2008 FUNDACIÓN UNIVERSITARIA SAN MARTÍN  
Facultad de Universidad Abierta y a Distancia,  
“Educación a Través de Escenarios Múltiples”  
Bogotá, D.C.

Prohibida la reproducción total o parcial sin autorización  
por escrito del Presidente de la Fundación.

La actualización de este fascículo estuvo a cargo de  
**ALEJANDRA CASTILLO**  
Docente tutor – Programa de Ingeniería de Sistemas a Distancia.  
Sede Bogotá, D.C.

Corrección de estilo  
**ADRIANA RODRÍGUEZ VALENCIA.**

Diseño gráfico y diagramación a cargo de  
**SANTIAGO BECERRA SÁENZ**  
**ORLANDO DÍAZ CÁRDENAS**

Impreso en: GRÁFICAS SAN MARTÍN  
Calle 61A No. 14-18 - Tels.: 2350298 - 2359825  
Bogotá, D.C., Marzo de 2012

## Procesadores



Facultad de Universidad Abierta y a Distancia  
Fundación Universitaria San Martín

### Introducción

El microcontrolador es quizás el componente electrónico más versátil que existe, sus aplicaciones están limitadas únicamente por la imaginación. Cada día es más frecuente encontrar equipos que los utilicen como elementos de control, de comunicaciones, de señalización, etc. Existen en el mercado una gran cantidad de microcontroladores tales como PICs, Basic Stamp, Xenix, Motorola, siendo estos últimos los microcontroladores de nuestro interés, debido a la gran cantidad de aplicaciones en las que se encuentran, fácil programación, son muy comerciales y se consiguen a muy bajo costo frente a sus competidores, lo cual los ha hecho tan llamativos para los profesionales como para estudiantes, etc.

El curso presentará una forma muy didáctica para la enseñanza de estos dispositivos, utilizando técnicas muy útiles para el aprendizaje y asimilación de la información en forma clara, concisa y muy rica conceptualmente, lo cual brindará al participante herramientas claves para comenzar a desarrollar aplicaciones desde las más sencillas hasta tan complejas como lo desee.

Por ejemplo, el controlador que regula el funcionamiento de un horno dispone de un sensor que mide constantemente su temperatura interna y, cuando 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. Realmente 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.



### Observación

¡Más microcontroladores que personas en el mundo!

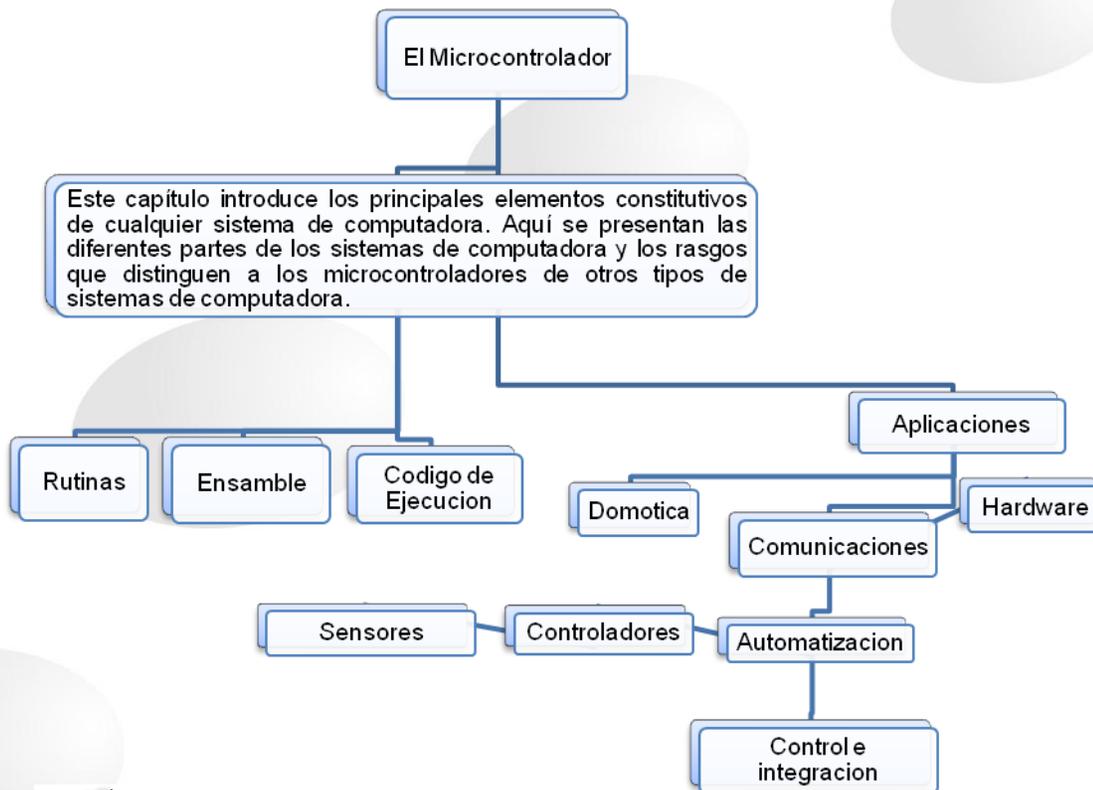
- ¿Cuántos microcontroladores llevas ahora mismo en los bolsillos?
- Teléfono móvil, llaves del coche, iPod, llave del garaje...
- ¿Cuántos tienes en casa? Lavadora, microondas, televisor, teclado, TV
- ¿Y en el coche? Un coche moderno puede tener cerca de 100 microcontroladores (airbags, tracción, ABS) En 2009, Microchip vendió 25 millones de microcontroladores PIC para el control de airbags.

### Conceptos previos

Para iniciar el estudio del fascículo 4. Es primordial repasar algunos conceptos sobre el lenguaje ensamblador del microcontrolador

1. Usando únicamente las instrucciones de corrimiento y rotación, realice la multiplicación de un número previamente cargado en el acumulador AX por 0Bh, sabiendo que el resultado no será mayor a 16 bits, realice el mismo ejemplo con la función de multiplicación MUL, calcule cuantos ciclos de reloj utiliza cada procedimiento

## Mapa conceptual fascículo 3.



### Logros

Al finalizar el estudio del presente fascículo, el estudiante debe estar en capacidad de:

- Realizara un esquema funcional del Microcontrolador
- Reconocerá la distribución de memoria y puertos del Microcontrolador
- Diseñara tiempos de respuesta y sincronismo del Microcontrolador
- Se lograra realizar el Diseño ciclos de reloj para el procesamiento por parte del micro
- Se realizara el montaje del cristal oscilador externo del Micro
- El estudiante tendrá el esquema de distribución de memoria del Microcontrolador HC08

## Descripción interna del Microcontrolador Motorola

### Arquitectura Interna

Los microcontroladores están compuestos básicamente por cuatro componentes:

El nombre "Motorola" fue adoptado en 1947, pero ha sido utilizado como marca comercial desde que en 1930 desarrolla la primera radio para automóvil. El fundador la instala en su equipo justo a tiempo para participar en una convención comercial de vendedores de autos. Al no lograr un stand para mostrarla, lo hace en uno de los aparcamientos. Inventa la palabra Motorola, uniendo el término "motor" con el sufijo "ola", para sugerir la idea de sonido en movimiento. Incipientes compañías utilizaron el sufijo "-ola" para comercializar sus fonógrafos, radios y otro equipamiento de audio en la década de 1920, la más famosa de ellas fue "Victrola", la empresa RCA lanzó su "radiola", había otra compañía que lanzó una máquina tocadiscos al mercado llamada Rock-Ola, y un editor de películas llamado Moviola.



**MOTOROLA**

Figura 1  
Logo Motorola

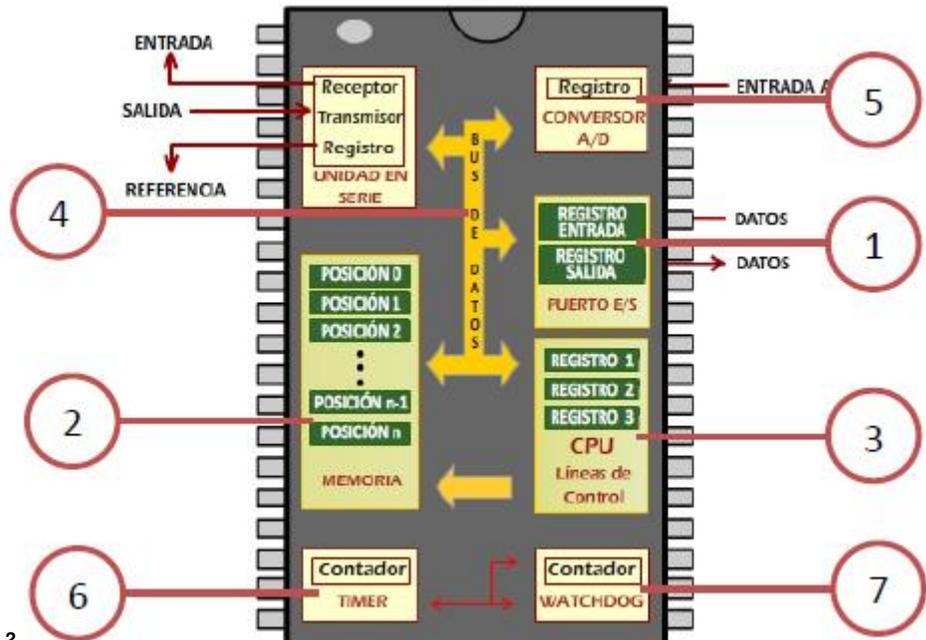


Figura 2  
Microcontrolador Internamente

## Unidad de input – output (1)

Hablamos de entradas y salidas (E-S) cuando nos referimos a las señales de Información que son recibidas y enviadas por unidades funcionales de un sistema de procesamiento de datos informáticos. Esta unidad provee comunicación con el mundo exterior. (Cfr. DEITEL,2010).

## Unidad de memoria (2)

Su función es almacenar datos. Es como un armario grande lleno de cajones. Si supones que tenemos marcados los cajones de tal manera que no se puedan confundir, cualquiera que sea su contenido será fácilmente accesible.

## Unidad de procesamiento central CPU (3)

Tiene la capacidad de ejecutar procesos tales como: multiplicar, dividir, restar, y mover su contenido desde una ubicación de memoria a otra. Estas posiciones de unidad de procesamiento central (CPU) se llaman registros.

## Procesadores

### El bus de datos (4)

Es el medio por el que los datos van de un bloque a otro, representan un grupo de 8, 16, o más cables. Hay dos tipos de buses dirección sirve para transmitir la dirección de la memoria a la CPU, y el bus de datos sirve para conectar los bloques dentro del microcontrolador.

### Conversor A/D (5)

Como las señales periféricas generalmente son diferentes a las que el microcontrolador puede entender (ceros "0" y unos "1"), estas tienen que convertirse en un modelo que pueda ser comprendido por un microcontrolador. Esta tarea la realiza el conversor Análogo/Digital.

### Contador TIMER (6)

Es responsable de generar esperas (Timer) que hacen que el microcontrolador realice tareas por períodos de tiempo determinados.

### Contador WATCHDOG (7)

El llamado watchdog o perro-guardián identifica las interrupciones en el ordenador y simplemente lo reinicia para seguir trabajando.

### Analogía con una planta procesadora

La estructura de un microcontrolador se podría comparar con la estructura básica de una planta procesadora de papas fritas.

**Unidad de input – output** / Zona de carga y descarga (1). En la fábrica sería la zona de descarga y despacho del producto terminado.

**Unidad de memoria** / Zona de almacenamiento (2). En una fábrica es la bodega. Allí los espacios están marcados de manera tal que no se pueden confundir, cualquiera que sea su contenido será fácilmente accesible.

**Unidad de procesamiento central CPU** / Zonas de proceso (3). En una fábrica serían los procedimientos que se le hacen a los productos, la transformación del producto, desde el lavado de las papas hasta el empaque.

**El bus de datos / Zonas de transporte (4).** En una fábrica serían los caminos internos por donde circulan los insumos y el producto.



Figura 3  
Comparación del Micro con una planta procesadora



### Observación

Los microprocesadores son circuitos integrados digitales complejos. Desde su nacimiento, allá por el año 1971, se han tornado cada vez más y más complejos y poderosos. Por ejemplo, los microprocesadores modernos que se usan en las computadoras pueden contener más de 300 millones de transistores.

### Millones de Transistores

La memoria del programa en forma de flash NOR o OTP ROM también incluye a menudo en el chip, así como una cantidad por lo general pequeños de RAM. Los microcontroladores son diseñados para aplicaciones embebidas, en contraste con los microprocesadores utilizados en computadoras personales u otras aplicaciones de propósito general. Los microcontroladores son utilizados en los productos controlados automáticamente y dispositivos, tales como los sistemas de control de motores de automóviles, dispositivos médicos implantables, controles remotos, máquinas de oficina, electrodomésticos, herramientas eléctricas, juguetes y otros sistemas embebidos. Al reducir el tamaño y el costo en comparación con un diseño que utiliza un microprocesador separado, la memoria y entrada/salida, los microcontroladores que sea económico para controlar digital-

## Procesadores

mente aún más dispositivos y procesos. Mezcla la señal de microcontroladores son comunes, la integración de componentes analógicos necesarios para controlar que no -sistemas electrónicos digitales.

Algunos microcontroladores pueden utilizar cuatro bits palabras y operan a la velocidad del reloj frecuencias tan bajas como 4 kHz, de bajo consumo de energía (o milivatios microvatios). Por lo general, tendrá la capacidad para mantener la funcionalidad a la espera de un evento como pulsar un botón o interrumpir otros, el consumo de energía durante el sueño (reloj de la CPU y los periféricos más alejados) puede ser justo nanovatios, por lo que muchos de ellos muy adecuado para la batería de larga duración las aplicaciones. Otros microcontroladores pueden servir de rendimiento crítico roles, donde se tenga que actuar más como un procesador de señal digital (DSP), con velocidades de reloj más altas y el consumo de energía. Para resolver el problema de la complejidad y el alto costo de los sistemas basados en microprocesadores, se crean los microcontroladores, que no es otra cosa que un sistema de microcomputadora completo. Es decir, un microcontrolador contiene en un solo circuito integrado el microprocesador, la memoria de datos, la memoria de programa y las unidades de entrada/salida, lo cual lo hace muy pequeño, barato y fácil de manejar, por lo que es ideal para muchas aplicaciones de propósito específico.

Ahora que hemos discutido sobre las diferentes partes de un sistema de computadora, estamos listos para decir qué es un microcontrolador. Nos muestra un sistema de computadora genérico con una parte encerrada por línea de puntos. La porción enmarcada es un microcontrolador y la mitad inferior de la figura es un diagrama de bloques que presenta su estructura interna con un mayor detalle. El cristal no está en el microcontrolador si bien es una parte esencial del circuito oscilador. En algunos casos, se pueden usar componentes más económicos reemplazando al cristal, como

resonadores cerámicos o bien circuitos con resistor y capacitor (R-C). Un **microcontrolador** puede definirse como un sistema de computadora completo incluyendo un CPU, memoria, reloj oscilador y I/O en el mismo circuito integrado.

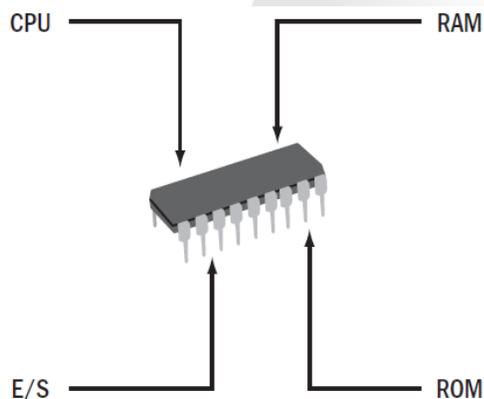


Figura 4  
Un microcontrolador encierra todos los elementos de una microcomputadora en un solo circuito integrado

Cuando carece de alguno de estos elementos, ya sea I/O o memoria, el circuito integrado lleva el nombre de **microprocesador**. En una PC la CPU es un microprocesador. En una computadora mainframe la CPU se conforma por varios circuitos integrados. (Cfr. DEITEL,2010).



### Observación

Las computadoras son sistemas basados en microprocesadores, y son llamados sistemas abiertos, debido a la flexibilidad y posibilidad de elegir o cambiar sus componentes (el propio procesador, la memoria, los periféricos, etcétera), mientras que un microcontrolador es un sistema cerrado.

Los microcontroladores pueden clasificarse según su arquitectura, que puede ser **Von Neumann o Harvard**. La arquitectura Von Neumann se caracteriza por disponer de una sola memoria principal donde se almacenan datos e instrucciones de forma indistinta. A dicha memoria se accede a través de un sistema de buses único. El bus de direcciones es usado para identificar qué posición de memoria está siendo accedida, mientras que el bus de datos es utilizado para trasladar información entre la CPU y alguna

## Procesadores

dirección de memoria o viceversa. Con un único sistema de buses, la arquitectura Von Neumann es usada secuencialmente para acceder a instrucciones de la memoria de programa y ejecutarlas regresando desde/hacia la memoria de datos. Esto significa que el ciclo de instrucción no puede solaparse con ningún acceso a la memoria de datos.

La **Arquitectura Harvard** se caracteriza por disponer de dos memorias independientes, una que contiene sólo instrucciones y otra con 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. Una de las ventajas de esta arquitectura es que la operación del microcontrolador puede ser controlada más fácilmente si se presentara una anomalía en el contador de programa.

Existe otra arquitectura que permite accesos a tablas de datos desde la memoria de programa. Esta arquitectura es la llamada arquitectura **Harvard modificada**. Esta última arquitectura es la dominante en los microcontroladores actuales ya que la memoria de programa es usualmente ROM, OTP, EPROM o

FLASH mientras que la memoria de datos es usualmente RAM, permitiendo que las tablas de datos estén en la memoria de programa para que estas no se pierdan cada vez que el sistema se apaga. Otra ventaja importante en la arquitectura Harvard modificada es que las transferencias de datos pueden ser solapadas con los ciclos de decodificación de instrucciones. Esto quiere decir que la siguiente instrucción puede ser cargada de la memoria de programa mientras se está ejecutando una instrucción que accede a la memoria de datos. La desventaja de la arquitectura Harvard modificada podría ser que se requieren instrucciones especiales para acceder a valores en memoria RAM y ROM haciendo la programación un poco complicada.

En cuanto a las técnicas de fabricación, casi la totalidad de los microcontroladores actuales se fabrican en tecnologías CMOS (Complementary Metal Oxide Semiconductor) que supera a todas las tecnologías existentes por su bajo consumo y su inmunidad frente al ruido. (Cfr. DEITEL,2010).

### Sistemas Embebidos

Sistema empotrado/embebido (Embedded System): sistema informático cuyo hardware y software están concebidos para un uso concreto. Se componen de dos elementos:

- Controlador.
- Sistema a controlar.

En todas partes, pues un sistema embebido es un ordenador más, que a diferencia de un Computador personal, por ejemplo, carece de teclado y pantalla en la mayoría de los casos. Dicho de otra manera, un sistema embebido consiste de un sistema electrónico programable especialmente diseñado para soluciones específicas.

Estas pueden ser:

- Sistemas de telefonía fija o móvil, por ejemplo un terminal telefónico móvil.
- Automatización de procesos de producción
- Equipos e instrumentación industrial
- Sistemas de transporte, desde cintas transportadoras, sistemas robotizados hasta vehículos de transporte de todo tipo.

En este capítulo cabe resaltar el sector de la automoción, en donde un turismo de diseño actual puede incorporar hasta más de 40 sistemas embebidos. Estos controlan áreas tan especiales, como el funcionamiento del motor de gasolina o gasóleo como el control de par MSR, sistema de frenado antibloqueo ABS, bloqueo diferencial electrónico EDS, sistema de control de tracción ASR, control de estabilidad con

## Procesadores

asistencia de frenado ESP, airbags, la cerradura centralizada con su mando a distancia, etc. etc.

- Electrodomésticos de todo tipo, como microondas, lavadoras, frigoríficos, lavavajillas, etc.
- Tiene una gran aplicación en la industria juguetera y de ocio
- Sistemas periféricos de un PC, como los MODEM, router, teclados, ratones de nueva generación, equipos multimedia, etc.

El sistema embebido es por tanto un ordenador especializado para una solución especializada en donde prevalecen las siguientes características:

- Esta especialmente diseñado para la solución óptima de la tarea o tareas a resolver.
- Generalmente es una "pieza especializada" instalada en un sistema anfitrión. A diferencia de un PC, el sistema embebido se dota con los módulos estrictamente necesarios para su función. De ahí su coste óptimo.
- Es una solución única en el mercado, no existe otra igual.

La incorporación de un sistema embebido proporciona a un producto un valor añadido importante que lo distingue claramente de los productos de la competencia. Esto es posible gracias a que el sistema embebido con respecto a la solución anterior proporciona una:

- Solución más precisa y rápida en su especialidad respecto a la solución anterior
- Mayor número de opciones respecto a la solución anterior
- Coste reducido, por lo tanto reducción de costes en el proceso de fabricación
- Aumenta la competitividad del producto por su diferenciación respecto a otros productos similares

**Estado de bajo consumo:** Es un estado del sistema donde se detiene el reloj principal y sus circuitos asociados con el objetivo de ahorrar energ-

ía en periodos de tiempo donde el microcontrolador se mantiene en espera de instrucciones.

**Conversores A/D:** Procesa señales analógicas convirtiéndolas en señales digitales.

**Modulador de Anchura de Pulso PWM:** Son circuitos que proporcionan en su salida impulsos de anchura variable, que se ofrecen al exterior a través de las patitas del encapsulado.

**Comparadores analógicos:** 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. (Cfr. DEITEL,2010).

### **Puertos de entrada y salida digital**

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 I2C**, interfaz serie de dos hilos desarrollado por Philips.
- **CAN** (Controller Area Network), permite la adaptación con redes de co-

## Procesadores

nexionado multiplexado desarrollado conjuntamente por Bosch e Intel para el cableado de dispositivos en automóviles.

## Temporizadores

- **Timers:** Contadores de 8 a 32 bits que se incrementan automáticamente. Permiten
- temporizar operaciones, interfaces, etc. Puede haber varios por chip (de 1 a 13).
- **Watchdog Timer:** Contador que se incrementa automáticamente, y se pone a 0 desde código. Si llega al final de la cuenta, resetea la MCU.

## Módulo CCP (Capture/Compare/PWM)

- Utiliza un registro dedicado para las tres funciones (CCPR) y un pin de E/S dedicado. Actúa junto con el primer temporizador (TMR1).
- **Capture:** en el momento en el que ocurre un flanco (programable) en el pin, se guarda el valor de TMR1 en CCPR y salta una interrupción.
- **Compare:** se guarda un valor en CCPR, y salta una interrupción cada vez que el valor de TMR coincide con el de CCPR.
- **PWM (Pulse Width Modulation):** utilizado para control de motores. Usa el pin como salida, emitiendo un pulso modulado en anchura.

## Volúmenes

Aproximadamente el 55% de todas las CPUs vendido en el mundo son de 8 bits microcontroladores y microprocesadores. De acuerdo con Semico, cuatro mil millones de microcontroladores de 8 bits se vendieron en 2006. Una casa típica en un país desarrollado es probable que tenga sólo cuatro microprocesadores de propósito general, pero alrededor de tres docenas de microcontroladores. Un típico automóvil de gama media tiene hasta 30 o más microcontroladores. También se pueden encontrar en muchos aparatos eléctricos como lavadoras, hornos microondas y teléfonos.

## Procesadores

Los fabricantes a menudo han producido versiones de sus microcontroladores especiales con el fin de ayudar a los equipos y el desarrollo de software del sistema de destino. Originalmente, estos incluyen EPROM versiones que tienen una "ventana" en la parte superior del dispositivo a través del cual se puede borrar la memoria del programa de ultravioleta de luz, listo para la reprogramación después de programación ("quemar") y el ciclo de prueba. Desde 1998, las versiones de EPROM son raras y han sido sustituidos por EEPROM y Flash, que son más fáciles de usar (se puede borrar electrónicamente) y más barato de fabricar. (Cfr. PATTERSON,2000).



Figura 5  
Microcontrolador 80-pin

Otras versiones están disponibles en la ROM es como un dispositivo externo en lugar de la memoria interna, sin embargo son cada vez menos frecuente debido a la amplia disponibilidad de los programadores de microcontroladores barato. El uso de dispositivos programables en campo en un microcontrolador puede permitir actualizar el campo de la firmware o permitir que se modifiquen finales de fábrica para los productos que se han reunido, pero no enviado todavía. Memoria programable también reduce el tiempo de preparación requerido para la implementación de un nuevo producto. Donde cientos de miles de dispositivos idénticos se requieren, usando las piezas programadas en el momento de fabricación puede ser una opción económica. Estos "máscara programado" partes tiene el programa establecido en la misma forma que la lógica del chip, al mismo tiempo.

## Procesadores



Facultad de Universidad Abierta y a Distancia  
Fundación Universitaria San Martín

### Entornos de programación

Microcontroladores fueron programadas originalmente sólo en lenguaje ensamblador, pero diversos lenguajes de alto nivel de programación son ahora de uso común para los microcontroladores de destino. Estos lenguajes son diseñados especialmente para el propósito, o versiones de los lenguajes de propósito general tales como el lenguaje de programación C. Compiladores para lenguajes de propósito general suele tener algunas restricciones, así como mejoras para apoyar mejor a las características únicas de los microcontroladores. Algunos microcontroladores tienen un ambiente para ayudar a desarrollar ciertos tipos de aplicaciones. Los vendedores de microcontroladores a menudo hacen las herramientas de libre disposición para hacer más fácil la adopción de su hardware. Muchos microcontroladores son tan extravagantes que efectivamente requieren sus propios dialectos no estándar de C, tales como la SDCC para el 8051, lo que impide el uso de herramientas estándar (tales como librerías de código o las herramientas de análisis estático), incluso para el código relacionado con las características de hardware. Los intérpretes se utilizan a menudo para ocultar tales peculiaridades de bajo nivel.

Intérprete de firmware está disponible para algunos microcontroladores. Por ejemplo, BASE en los primeros microcontroladores Intel 8052; BASIC y FORTH en el Z8 de Zilog, así como algunos dispositivos modernos. Normalmente, estos intérpretes de apoyo de programación interactiva. Los simuladores están disponibles para algunos microcontroladores. Estos permiten a un desarrollador para analizar lo que el comportamiento del microcontrolador y su programa debe ser si se utiliza la parte real. Un simulador mostrará el estado del procesador interno y también el de las salidas, así como permitir que las señales de entrada que se generen. Mientras que por un lado, la mayoría de los simuladores se limitará a ser capaces de simular otros equipos mucho más en un sistema, que pueden ejer-

cer las condiciones que de otra manera puede ser difícil de reproducir a voluntad en la ejecución física, y puede ser la manera más rápida para depurar y analizar los problemas.

### La latencia de interrupción

A diferencia de los ordenadores de uso general, microcontroladores utilizados en sistemas embebidos a menudo tratan de optimizar la latencia de interrupción más de rendimiento de las instrucciones. Los temas incluyen tanto reducción de la latencia, y por lo que es ser más predecible (con soporte en tiempo real de control).

Cuando un dispositivo electrónico que produce una interrupción, los resultados intermedios (registros) tienen que ser guardados antes de que el software responsable de manejar la interrupción se pueden ejecutar. También debe ser restaurado después de que el software ha finalizado. Si hay más registros, el proceso de guardar y restaurar toma más tiempo, el aumento de la latencia. Formas de reducir ese contexto / restore latencia incluyen tener relativamente pocos registros en sus unidades de procesamiento central (no deseados, ya que ralentiza la mayoría no interrumpir el procesamiento de cantidades), o al menos no con el hardware de salvarlos a todos (esto no funciona si el software se necesita para compensar por el ahorro del resto "a mano"). Otra técnica consiste en pasar las puertas de silicio en la "sombra de registros": Uno o más registros duplicados utilizado sólo por la interrupción de software, tal vez el apoyo a una pila dedicado. (Cfr. PATTERSON,2000).

Otros factores que afectan la latencia de interrupción son:

- Ciclos necesarios para completar las actividades actuales de la CPU. Para reducir al mínimo los costos, los microcontroladores suelen tener tuberías cortas (a menudo tres instrucciones o menos), buffers peque-

ños escribir, y asegurarse de que ya las instrucciones son de continuación o reiniciar. RISC principios de diseño de asegurar que la mayoría de las instrucciones de tomar el mismo número de ciclos, ayudando a evitar la necesidad de que la mayoría de dicha continuación / reiniciar la lógica.

- La longitud de cualquier sección crítica que debe ser interrumpido. La entrada a la sección crítica restringe el acceso concurrente de estructura de datos. Cuando una estructura de datos debe tener acceso a un controlador de interrupciones, la sección crítica debe bloquear esa interrupción. En consecuencia, la latencia de interrupción se incrementa en por mucho tiempo que interrumpir está bloqueado. Cuando hay fuertes restricciones externas sobre la latencia del sistema, los desarrolladores a menudo necesitan de herramientas para medir las latencias de interrupciones y rastrear que las secciones críticas causar ralentizaciones.
  - Una técnica común a pocas cuadras todas las interrupciones durante la duración de la sección crítica. Esto es fácil de implementar, pero a veces las secciones críticas incómodamente largo.
  - Una técnica más compleja a pocas cuadras las interrupciones que puedan provocar el acceso a la estructura de datos. Esto a menudo se basa en las prioridades de interrupción, que tienden a no responden bien a los datos pertinentes del sistema de estructuras. En consecuencia, esta técnica se utiliza sobre todo en ambientes muy restringidos.
  - Los procesadores pueden tener soporte de hardware para algunos sectores críticos. Los ejemplos incluyen el apoyo al acceso a la atómica bits o bytes en una palabra, o de otras primitivas acceso atómico como el LDREX / STREX primitivas de acceso exclusivo introducido en el ARMv6 arquitectura.
    - Interrupción de anidación. Algunos microcontroladores permiten una mayor prioridad interrumpe a interrumpir las inferiores prio-

ridad. Esto permite que el software para gestionar la latencia, dando tiempo crítico interrumpe una prioridad más alta (y por lo tanto la latencia más baja y más previsible) que los menos críticos.

- **Desencadenar tasa.** Cuando las interrupciones se producen back-to-back, los microcontroladores pueden evitar un contexto adicional de guardar / restaurar el ciclo de una forma de llamar a la cola de optimización.

Menor microcontroladores final tienden a apoyar menos controles latencia de interrupción que los de gama alta.

### **La familia de los PIC como elección**

¿Qué es lo que ocurre con los PIC?, ¿Por qué están en boca de todos?. Hemos buscado en multitud de bibliografía y realmente nadie da una respuesta concreta, pero una aproximación a la realidad puede ser esta: Los PIC tienen “ángel”, tienen “algo” que fascina a los diseñadores, puede ser la velocidad, el precio, la facilidad de uso, la información, las herramientas de apoyo.

Quizás un poco de todo eso es lo que produce esa imagen de sencillez y utilidad. Es probable que en un futuro próximo otra familia de microcontroladores le arrebatase ese “algo”. Queremos constatar que para las aplicaciones más habituales (casi un 90%) la elección de una versión adecuada de PIC es la mejor solución; sin embargo, dado su carácter general, otras familias de microcontroladores son más eficaces en aplicaciones específicas, especialmente si en ellas predomina una característica concreta, que puede estar muy desarrollada en otra familia. Los detalles más importantes que vuelven “locos” a los profesionales de la microelectrónica y microinformática y las razones de la excelente acogida que tienen los PIC son los siguientes:

## Procesadores

- Sencillez de manejo: Tienen un juego de instrucciones reducido; 35 en la gama media.
- Buena información, fácil de conseguir y económica.
- Precio: Su coste es comparativamente inferior al de sus competidores.
- Poseen una elevada velocidad de funcionamiento. Buen promedio de
- parámetros: velocidad, consumo, tamaño, alimentación, código compacto, etc.
- Herramientas de desarrollo fáciles y baratas. Muchas herramientas software se pueden recoger libremente a través de Internet desde Microchip
- Existe una gran variedad de herramientas hardware que permiten grabar,
- depurar, borrar y comprobar el comportamiento de los PIC.
- Diseño rápido.
- La gran variedad de modelos de PIC permite elegir el que mejor responde a los requerimientos de la aplicación.

Una de las razones del éxito de los PIC 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.

### Las gamas de PIC

Una de las labores más importantes del ingeniero de diseño es la elección del microcontrolador que mejor satisfaga las necesidades del proyecto con el mínimo presupuesto. (Cfr. PATTERSON,2000).

Para resolver aplicaciones sencillas se precisan pocos recursos, en cambio, las aplicaciones grandes requieren numerosos y potentes. Siguiendo esta filosofía Microchip construye diversos modelos de microcontroladores orientados a cubrir, de forma óptima, las necesidades de cada proyecto. Así, hay disponibles Microcontroladores sencillos y baratos para atender las aplicaciones simples y otros complejos y más costosos para las de

mucha envergadura. Microchip dispone de cuatro familias de microcontroladores de 8 bits para adaptarse a las necesidades de la mayoría de los clientes potenciales.

### Ejemplo Programa para PIC

```

; *****
; Programa Display.asm
; Contamos hasta 0x5f.
; El valor del contador se visualizará en 8 diodos LED conectados al
puerto B
; a partir de la patilla 1, sin gestión de punto decimal
; Preparado para PIC16F84
; Velocidad del reloj: 4 MHz
; Ciclo de instrucción: 1 MHz = 1 microsegundo
; Interrupciones: A través de PB.0, para detener y recomenzar la cuenta.
; Perro guardián: Desactivado
; Tipo de Reloj: XT
; Protección del código: Desactivado
; *****

LIST P = 16F84 ;Indicamos el modelo de PIC a utilizar
; Definición de registros
portb EQU 0x06 ;Hemos conectado el teclado al puerto B
;La dirección 0x06 corresponde al registro PORTB (puerto B)
; en el banco1

TRISB EQU 0X06 ; y TRISB en banco 1
estado EQU 0X03 ; La dirección del registro de estado es la 0x03
pc EQU 0x02 ; Contador de Programa, dirección de memoria actual de
programa
intcon EQU 0x0B ; Registro gestor de interrupciones
opcion EQU 0x01 ; Registro OPTION. Recordar que está en el banco 1.

```



## Procesadores

; Definición de bits

banco EQU 0X05 ; Bit del registro de estado correspondiente al banco de datos

Z EQU 0X02 ; Bit indicador de que el registro W está a cero

int EQU 0x00 ; Bit de interrupción externa, es el 0 en el puerto B.

intdeg EQU 0x06 ; Bit 6 de OPTION, que indica si la interrupción PB0 es por nivel

alto.

intf EQU 0x01 ; Bit 1 de INTCON, flag de interrupción por PB0.

inte EQU 0x04 ; Bit 4 de INTCON, habilitador de interrupción por PB0.

GIE EQU 0x07 ; Bit 7 de INTCON, habilitador de interrupciones.

; Definición de constantes

w EQU 0 ; Destino de operación = w

f EQU 1 ; Destino de operación = registro

; Definición de variables

contador EQU 0X0C ; Contador

digito EQU 0X0D ; Para almacenar el dígito

; Comienzo del programa.

ORG 0X00 ; Cubrimos el vector de reset

GOTO inicio ; Saltamos a la primera dirección tras el vector de interrupción

ORG 0x04 ; Vector de interrupción

GOTO RSI

105

; \*\*\*\*\* Inicialización de variables \*\*\*\*\*

ORG 0X05

inicio BSF estado,banco ; Cambiamos a la segunda página de memoria

CLRF TRISB ; Programa la puerta B como de todo salidas

BSF TRISB,int ; Salvo la pata de interrupción PB0, que es de entrada

BSF opcion,intdeg ; Interrupción PB0 cuando esté a nivel alto.

BCF estado,banco ; Volvemos a la página 0.

BCF intcon,intf ; Borramos el flag de interrupción por PB0.

BSF intcon,GIE ; Habilitamos las interrupciones.

BSF intcon,inte ; Habilitamos la interrupción por PB0.

CLRF portb ; Apaga el display, por si había residuos

CLRF contador ; Borra el contador (dirección 0x0C)

CLRW ; Borramos el registro W

; \*\*\*\*\* Cuerpo Principal

\*\*\*\*\*

Reset CLRF digito ; Comienza a contar por el 0

Siguien MOVF digito,w ; Coloca el siguiente dígito a evaluar en W

CALL Tabla ; Llama a la subrutina para coger el dato

; y hacer la conversión decimal-7 segmentos

MOVWF portb

Pausa DECFSZ contador ; Decrementa contador y repite

GOTO Pausa ; hasta que valga 0

INCF digito,f ; Incrementa el valor del dígito al siguiente

MOVF digito,w ; Pone el valor del dígito en W

XORLW 0x0A ; Chekea si el dígito sobrepasa el valor 9

BTFSC estado,Z ; Comprobando con un xor si W vale 0 (Z=1)

GOTO Reset ; Si Z=1 resetea el dígito y comienza de nuevo la cuenta

GOTO Siguien ; En caso contrario, continua la cuenta

; \*\*\*\*\* La tabla queda definida aquí

\*\*\*\*\*

Tabla ADDWF pc,f ; Suma al contador de programa el valor de offset, es decir,

; el valor del dígito. Así se genera un PC distinto

; según su valor,

; asegurando que vaya a la línea correcta de la tabla

RETLW 0x7F ; 0 en código 7 segmentos (desplazado a la izquierda)

## Procesadores

RETLW 0x0C ; 1 en código 7 segmentos (desplazado a la izquierda)  
 RETLW 0xB6 ; 2 en código 7 segmentos (desplazado a la izquierda)  
 RETLW 0x9F ; 3 en código 7 segmentos (desplazado a la izquierda)  
 RETLW 0xCC ; 4 en código 7 segmentos (desplazado a la izquierda)  
 RETLW 0xDA ; 5 en código 7 segmentos (desplazado a la izquierda)  
 RETLW 0xFA ; 6 en código 7 segmentos (desplazado a la izquierda)  
 RETLW 0x0F ; 7 en código 7 segmentos (desplazado a la izquierda)  
 RETLW 0xFF ; 8 en código 7 segmentos (desplazado a la izquierda)  
 RETLW 0xDF ; 9 en código 7 segmentos (desplazado a la izquierda)  
 RSI BTFSS intcon,intf ; Si no es interrumpido por PB0, volver al programa  
 RETFIE  
 pulsado BTFSC portb,0 ; Retenemos hasta que se suelte el pulsador  
 GOTO pulsado  
 MOVLW 0xFF ; Puesto que se habrá de incrementar  
 MOVWF digito ; Ponemos el marcador a FF  
 BCF intcon,intf ; Borramos la bandera de interrupción  
 BSF intcon,inte ; Y rehabilitamos la interrupción por PB0  
 RETFIE  
 END



### Resumen

Los microcontroladores son muy utilizados hoy en día para múltiples aplicaciones, desde las más sencillas hasta las más complejas y poderosas, tanto a nivel aficionado como profesional e industrial. En este fascículo se realizó el estudio inicial sobre qué es un microcontrolador y se dieron los primeros pasos para abordar el tema.

Por supuesto que para construir nuestros propios proyectos se requiere componentes como el microcontrolador, recomendable comprar al me-

nos uno de ellos y, si se puede otro más ya que nunca se sabe cuándo se puedan presentar errores. Se deben adquirir también otros componentes que se estudiarán a lo largo de los fascículos, como por ejemplo, displays, teclados, leds, algunos circuitos integrados, etcétera, que se detallarán en su momento y seleccionar el microcontrolador requerido para un diseño concreto.

Se deben 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.):

**Costes.** 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 10 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 en el microcontrolador de algunas pesetas es importante (el consumidor deberá pagar además el coste del empaquetado, el de los otros componentes, el diseño del hardware y el desarrollo del software). Si el fabricante desea reducir costes debe tener en cuenta las herramientas de apoyo con que va a contar: emuladores, simuladores, ensambladores, compiladores, etc. Es habitual que muchos de ellos siempre se decanten por microcontroladores pertenecientes a una única familia.

**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 asegurarnos de seleccionar un dispositivo suficientemente rápido para ello.

## Procesadores



Facultad de Universidad Abierta y a Distancia  
Fundación Universitaria San Martín

Por otro lado, habrá que 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. Una alternativa más barata y quizá suficiente es usar librerías para manejar los datos de alta precisión. (Cfr. PATTERSON, 2000).



### Bibliografía recomendada

Deitel P. J. Deitel H. M., Cómo programar en C++, Prentice may, 2010  
David A. Patterson, Estructura y diseño de Computadores, John L. Hennessy  
Ed. Reverté S.A. 2000  
<http://www.technologyca.com/curso-virtual-de-microcontroladores-12-videos/>  
[http://www.ucontrol.com.ar/wiki/index.php/El\\_microcontrolador](http://www.ucontrol.com.ar/wiki/index.php/El_microcontrolador)



### Nexo

Los planteamientos explicados en este fascículo deben ser tenidos en cuenta, dado que se programará microcontrolador Motorola freescale hc8 en el próximo fascículo.

El 15 de agosto de 2011, Google anunció la compra de Motorola Mobility, una división de la compañía, por 12.500 millones de dólares, 3 como parte de su estrategia de potenciar el uso de Android.



Facultad de Universidad Abierta y a Distancia  
Fundación Universitaria San Martín

# Procesadores





### Seguimiento al autoaprendizaje

## Procesadores - Fascículo No. 4

Nombre \_\_\_\_\_

Apellidos \_\_\_\_\_ Fecha \_\_\_\_\_

Ciudad \_\_\_\_\_ Semestre \_\_\_\_\_

Resolver:

1. Cita las diferencias entre un sistema basado en microcontrolador y uno basado en microprocesador. Cita 5 ejemplos de aplicaciones de tu entorno que utilizan microcontroladores.
2. ¿Qué tipo de sensores podrías conectar a un microcontrolador? Inventa una aplicación en la que se use un micro, un pulsador y dos sensores.
3. Busca 2 fabricantes de microcontroladores y detalla para cada uno, las características de un micro de 8 bits con puertos USB y que pueda transmitir datos vía RF (Microchip, Silabs, Emmicroelectronics, etc).
4. Describe para los modelos elegidos, las características principales: velocidad, cantidad de memoria, número de puertos y dispositivos integrados.
5. Si alguna característica es desconocida para ti, apúntala aquí.
6. ¿Para qué sirve el entorno de desarrollo MPLAB IDE de Microchip?
7. Imagina una aplicación (creativa) con un micro que integre Wifi o Bluetooth.
8. En un micro, ¿para qué sirve el registro Watchdog?
9. ¿Qué es un ciclo de máquina?
10. ¿Qué ventajas tiene programar en ensamblador? En un microcontrolador ¿en qué lenguaje programarías? Explica tus razones.
11. Representa de forma detallada, en base a un diagrama de bloques funcionales, la estructura genérica de un microcontrolador. Describe la funcionalidad de cada uno de los bloques.
12. ¿Qué criterios seguirías a la hora de seleccionar un microprocesador y/o un microcontrolador?. Justifica la respuesta.

13. Enumera las principales características del microcontrolador 8051 de Intel. Dicho microcontrolador junto con el 6811 de Motorola se ha convertido, en lo que respecta a arquitecturas de 8 bits, en prácticamente un estándar a nivel industrial. Explica las razones por las que crees que se está extendiendo el uso de este microcontrolador.
14. ¿Por qué la arquitectura predominante en el diseño de microprocesadores y microcontroladores ha pasado a ser la RISC en lugar de la CISC?