

TECNOLOGÍA ELECTRÓNICA DE COMPUTADORES

2º Curso – GRADO EN INGENIERÍA INFORMÁTICA
EN TECNOLOGÍAS DE LA INFORMACIÓN

Tema 9: Circuitos integrados: microcontroladores

Lección 17. Microcontroladores PIC: estructura

Lección 17. Introducción a los microcontroladores PIC

17.1. Familias de microcontroladores PIC de 8 bits

17.2. Módulos internos y diagrama de bloques

17.3. Procedimiento de diseño con microcontroladores

17.4. Herramientas de diseño con microcontroladores PIC

17.1. Familias de Microcontroladores PIC de 8 bits

• Familia PIC10	10 Dispositivos
• Familia PIC12	18 Dispositivos
• Familia PIC14 (Obsoleto)	1 Dispositivo
➔ • Familia PIC16	106 Dispositivos
• Familia PIC18	214 Dispositivos

TOTAL: ¡ 349 Dispositivos !

Versiones de Memoria de Programa:
OTP=One Time Programming, EPROM,
y FLASH-EEPROM

16CXXX = OTP

16CRXXX = EPROM

16FXXX=Flash EEPROM



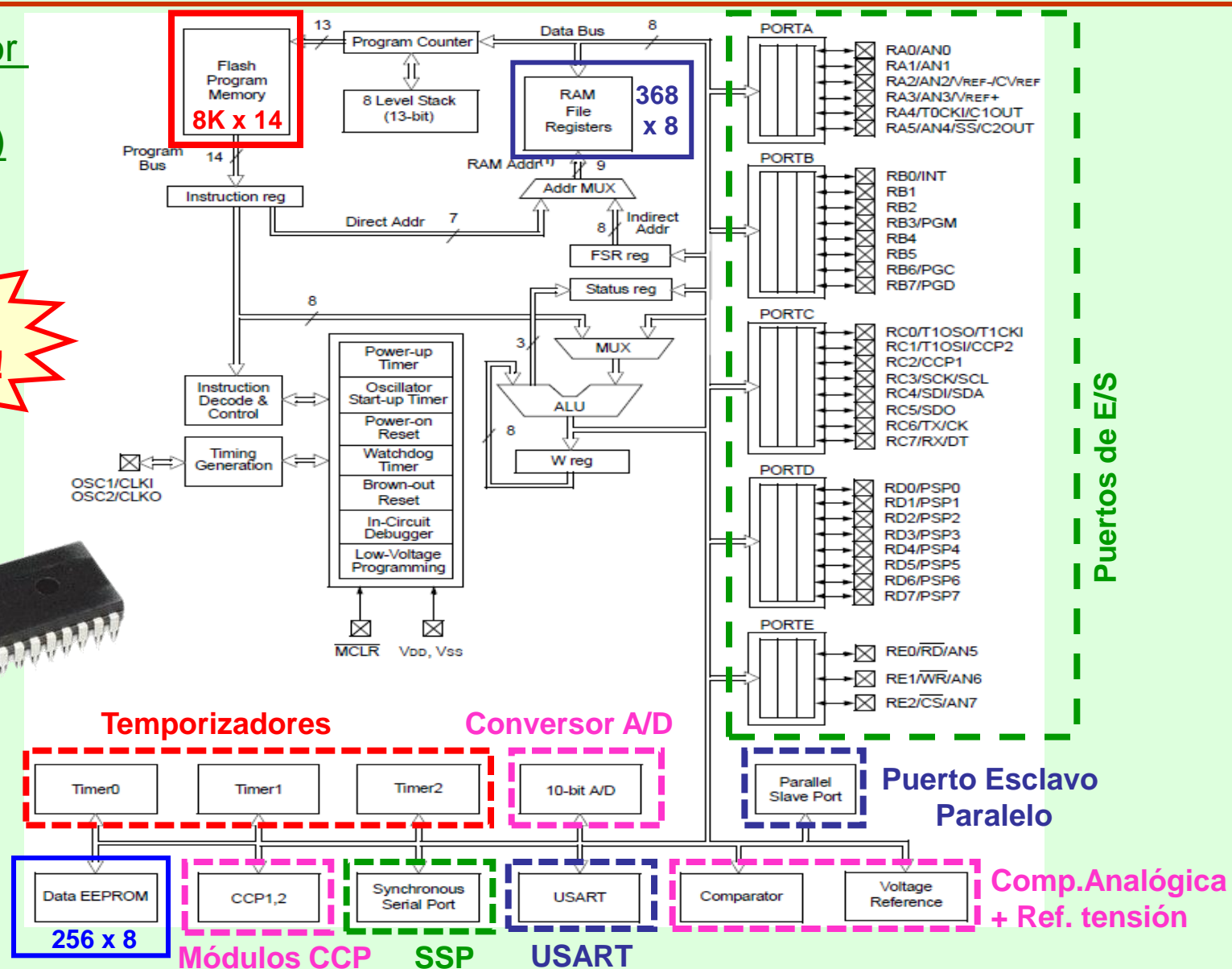
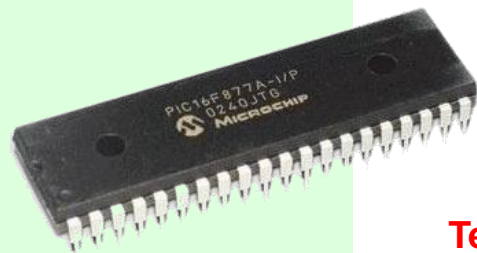
17.2. Módulos internos y diagrama de bloques

Módulos Internos Disponibles en la Familia Media (PIC16)

- Puertos de Entrada/Salida
- Puerto Esclavo Paralelo (PSP)
- Temporizadores/contadores (TMR0, TMR1, TMR2)
- Captura / Comparación / PWM (CCP1 y CCP2)
- Conversión Analógica / Digital (A/D)
- Transmisor Receptor Asíncrono Síncrono Universal (USART ó SCI)
- Puerto Serie Síncrono Básico ó Maestro (BSSP ó MSSP)
- Memoria EEPROM de datos
- FLASH EEPROM de programa modificable desde el código
- Comparador analógico
- Referencia de tensión configurable

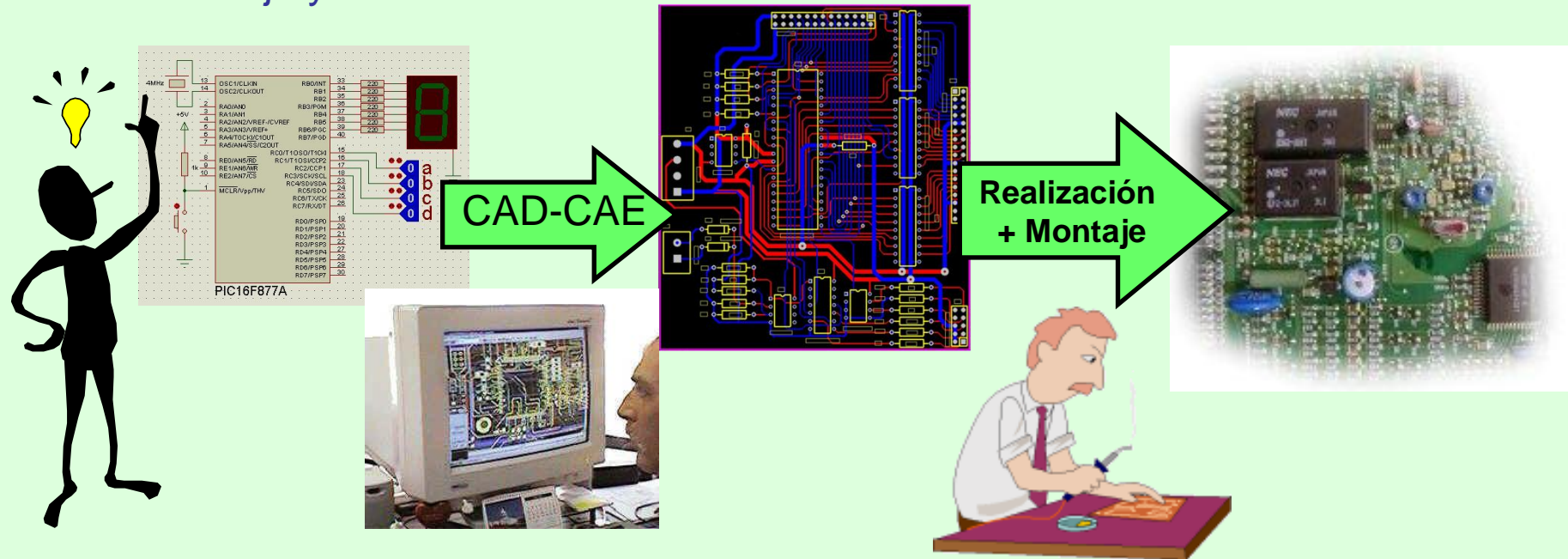
Microcontrolador PIC16F877A (familia media)

¡ Tiene
de todo !!



17.3. Procedimiento de Diseño con Microcontroladores

- **Diseño hardware:** conectaremos al micro todo lo que necesita para funcionar y lo que le permite interaccionar con el mundo exterior
 - Definición de las señales de Entrada, Salida y Alimentaciones
 - Etapas de Adaptación/Acondicionamiento Eléctrico de Señales
 - Selección de los módulos del microcontrolador a utilizar y pines
 - Diseño de la Placa de Circuito Impreso (PCB)
 - Montaje y ensamblado



El Procedimiento de Diseño con Microcontroladores (II)

- **Diseño software:** el micro sin un programa cargado en su memoria no hace absolutamente nada, cuando el micro esté grabado con un programa eficiente el conjunto empezará a funcionar
 - Diseño del programa y escritura del código fuente en ensamblador o en C
 - Pruebas, verificación y modificación del código:
 - * Simulación del Programa (puro software)
 - * Emulación dentro del Circuito de Aplicación (software y hardware)
 - Grabación del código máquina en la memoria interna del microcontrolador



17.4. Herramientas de diseño con microcontroladores PIC

- Entorno de Desarrollo: **MPLAB IDE** (*Integrated Development Environment*)

- Herramientas Software (para generar código máquina)

- Ensambladores + Montador de enlaces (Linkers)
- Compiladores de C (varios, de diferentes fabricantes)
- Intérpretes (Basic, ...)
- Bibliotecas de programas

- Herramientas de depuración (para depurar una aplicación)

- Simuladores: **MPLAB-SIM** (incluido en MPLAB-IDE)
- Emuladores en el Circuito (el emulador sustituye al micro en el circuito)
- Depuradores en el Circuito (In-Circuit Debuggers: permiten ejecutar un programa en el micro en modo depuración)
- Programadores/Grabadores de la memoria de los PIC
- Otros: tarjetas de evaluación y demostración

El Programador y los microcontroladores

- **Programador de software de PC** o computador, o similar:
 - Se parte de un “lujoso” hardware ya cableado, instalado y en funcionamiento (microprocesador, memoria y dispositivos de E/S como teclado, ratón y monitor, o pantalla)
 - El desarrollo de la aplicación supone “sólo” **escribir el programa** y depurar errores
- **Programador para microcontroladores:**
 - Primero hay que “**diseñar y crear**” el **interface de E/S** con el mundo exterior
 - Luego se escribe un programa específico para ese “hardware”
 - La mayoría de los programas son a medida de una aplicación y no son portables
- Herramientas de desarrollo para diseños basados en micros
 - Básicamente similares a las que se usan para programas de PC
 - Las librerías gráficas se sustituyen por las de manejo de los módulos del μC

Como todavía no se hacen PCs del tamaño y coste de los microcontroladores

(incluyendo teclado y el monitor) se seguirán utilizando los microcontroladores en los sistemas empotrados

¿Programar en lenguaje C o en ensamblador?

- La ventaja del C frente al ensamblador es la **rapidez en el desarrollo de las aplicaciones y la comodidad** a la hora de utilizar las funciones de manejo de los módulos internos (la diferencia entre **2 ó 3 días** y **1 ó 2 semanas** con aplicaciones para el manejo de un LCD por ejemplo).
- Las ventajas del **ensamblador** sobre el **C** residen en **la eficiencia y lo compacto que resulta el código** (entorno a un 80% menor en tamaño). En el ensamblador de los microcontroladores PIC, una instrucción ocupa una única posición de la memoria de programa. Una simple instrucción en C que nos ocupa una única línea de nuestro código fuente puede traducirse en varias posiciones de memoria de programa (como ejemplo se pueden probar unas cuantas condiciones en un “if...”).
- Además, cuando uno utiliza la programación en ensamblador, se tiene un **control total sobre el tiempo de ejecución de las instrucciones**, lo que puede resultar especialmente importante en ciertas aplicaciones en tiempo real.

¿Qué veremos del PIC?

Hardware

- Organización de la memoria
- Puertos de Entrada/Salida
- Temporizadores/contadores (TMR0, TMR1, TMR2)
- ...

Software

- Juego de instrucciones
- Ejemplos de programas
- Entorno de desarrollo (Prácticas)
- ...