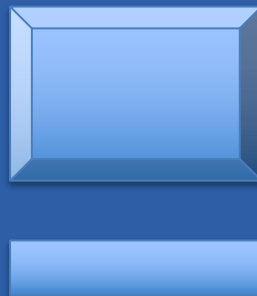


Arquitectura y Organización de Computadores



TEMA 1 Introducción



Objetivos generales



- Repasar los conocimientos previos necesarios para el seguimiento de la misma
 - Representación de información en binario
 - tipos de sistemas informáticos
- Adquirir una visión global de la asignatura

Representación numérica



- Repaso de la asignatura de Fundamentos de la Ingeniería Informática
- Imprescindible manejar:
 - Binario : Naturales y enteros en complemento a dos
 - Hexadecimal
 - Ascii
 - Estas bases de numeración deben resultarnos tan naturales como la base 10

Complemento a la base: Restar sumando



- ¿Podemos hacer restas mediante sumas de números positivos?
- Realizar $45 - 17$ sumando naturales (resultado 28)

45 + 83 = 128

Complemento a la base: Restar sumando



- ¿83?, ¿128?

- Estamos suponiendo que sólo hay dos dígitos (en los computadores el número de dígitos (bits) es limitado).
- El “1” de 128 se pierde por “overflow”, quedando sólo 28, la solución buscada.



- El 83 es el “complemento a 10” (estamos en base 10 y dos dígitos) del 17, puesto que $83 + 17$ es 00 (100), 83 **representa al número -17**
- Si tuviéramos tres dígitos, -17 sería....

Complemento a 2



- Mismo funcionamiento que el Complemento a 10.
- En binario, cualquier número interpretado en C2 cuyo bit más alto sea un 1, es negativo.
- Suponemos 4 dígitos binarios
 - Realizar $7 - 3 = 4 \rightarrow 0111 - 0011 = 0100$
 - Sumando $0111 + 1101 = 10100 \rightarrow 0100$ en 4 dígitos
 - 1101 (13) es $10000 - 00011$
- Varias técnicas para obtener el C2:
 - Complemento a 1 (invertir dígitos) y sumar 1
 - Recorridos de derecha a izquierda con cambios condicionales

Objetivos específicos del tema 1



- **Diferenciar los conceptos de arquitectura, microarquitectura y tecnología de computadores**
 - Definir dichos conceptos
 - Clasificar una característica de un procesador dentro de uno de esos tres conceptos
- **Conocer los principios básicos para la optimización de los computadores**
 - Enunciar los 4 grupos de la taxonomía de Flynn y clasificar computadores dentro de estos grupos
 - Definir localidad y encontrar ejemplos de localidad espacial y temporal en un código dado
- **Conocer la evolución que han sufrido los computadores**
 - Razonar sobre la evolución que han sufrido las arquitecturas y microarquitecturas de computadores de diferentes compañías
 - Explicar los diferentes dominios de aplicación de un computador/procesador

Tipos de sistemas informáticos



Computadores



Tipos de sistemas informáticos



Clasificación según su propósito

- Propósito general
- Propósito específico



Tipos de sistemas informáticos



- Supercomputadores
- Mainframes (servidores de gama alta)
- Servidores de gama media y básica
- Computadores personales (de sobremesa y portátiles)
- Computadores móviles

Servidor: computador conectado a una red de transmisión de datos que da servicio compartido a múltiples usuarios. Pueden ser de diferentes tipos: servidor de aplicaciones, servidor de archivos, servidor de red, ...

Tipos de sistemas informáticos



- **Supercomputadores**

- Cálculo intensivo de tipo científico y técnico (High Performance Computing, HPC)
- Pueden llegar a tener millones de procesadores
- Memoria principal > 1 PB
- Disco > 1PB
- Lista en www.top500.org



Supercomputador Sunway TaihuLight de NRCPC

Tipos de sistemas informáticos



- **Supercomputadores**

Nota de prensa del año 2002

“Ice Age”, el último éxito de taquilla del cine de animación, utiliza la tecnología de supercomputación de [Compag](#)



(...) Para lograr la máxima funcionalidad y rapidez en el proceso de creación de imágenes generadas por ordenador (CGI), Blue Sky Studio eligió los Alphaservers por su gran capacidad y potencia, capaces de soportar y agilizar el proceso de renderización – creación de imágenes por ordenador- que tarda en generarse entre ocho y 30 horas por fotograma, algo asombroso si se tiene en

Entre lo

**1.440 fotogramas/min * 30h/fotograma = 43.200h =
1800 días = 4,9 años por minuto de película!!**

Tipos de sistemas informáticos

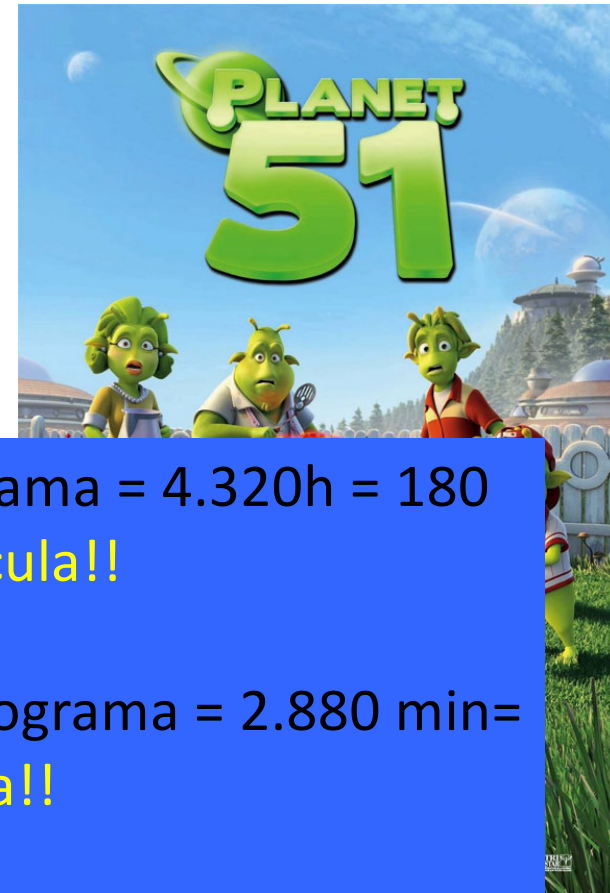


- **Supercomputadores**

Nota de prensa del año 2010

Planet 51 designed by Ilion Animation Studios on a [Bull](#) supercomputer

(...) Each frame which makes up the movie involves a graphics generation process that is managed by intensive image rendering. The process of rendering the scene to be shown on the screen involves a lot of work to overcome the limitations of the computer. The process is to



$1.440 \text{ fotogramas/min} * 3\text{h/fotograma} = 4.320\text{h} = 180 \text{ días} = 6 \text{ meses por minuto de película!!}$

$1.440 \text{ fotogramas/min} * 2 \text{ min/fotograma} = 2.880 \text{ min} = 48\text{h} = 2 \text{ días por minuto de película!!}$

Tipos de sistemas informáticos



- **Mainframes**

- Acceso a grandes bases de datos desde terminales (entidades bancarias, reserva de billetes,...)
- Varios cientos de procesadores
- De cientos a miles de usuarios



Tipos de sistemas informáticos



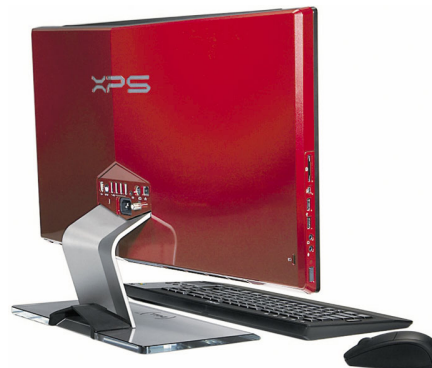
- **Servidores de gama media y básica**
 - Similares a los mainframes, pero en departamentos o empresas pequeñas
 - Se controlan remotamente
 - Suelen disponerse en racks
 - Hasta cientos de usuarios
 - Decenas de procesadores



Tipos de sistemas informáticos



- **Computadores personales**
 - Monousuario
 - Los de gama alta se denominan **estaciones de trabajo** (*workstations*): mayor capacidad de cálculo y gráfica
 - Incluye los PC de sobremesa (fijos) y los portátiles
 - Existe muchos modelos y fabricantes



Tipos de sistemas informáticos



- **Computadores móviles**
 - Pequeño, ligero, alimentación por acumuladores
 - Memoria RAM
 - Memoria Flash



Conceptos básicos



- **Arquitectura de un computador:**

- Gene Amdahl, IBM en 1964, durante el diseño del IBM S/360
- Es la primera vez que se distingue *arquitectura* de *implementación*

La ***arquitectura*** de un computador que debe conocer un programador de ***lenguaje máquina*** para poder escribir programas en ese computador

- La *arquitectura* especifica el comportamiento funcional del procesador mientras que la ***implementación*** es la estructura lógica que lleva a cabo dicha funcionalidad.
- La ***realización*** es la estructura física que plasma la implementación



Arquitectura (ISA: Instruction Set Architecture)

Repertorio de instrucciones
Modos de direccionamiento
Formato de instrucciones
Tamaño y tipos de datos

IBM 360, M68k,
IA32, PowerPC

Implementación (Microarquitectura)

Memoria cache
Segmentación
Técnicas
superescalares

IBM 360/91, M68040,
Intel P6, PowerPC 604

Realización (Tecnología)

Frecuencia de reloj
Capacidad de cache
Empaquetamiento
Consumo de potencia
Tecnología de fabricación

P6 => Pentium Pro,
Pentium II y Pentium III

Conceptos básicos

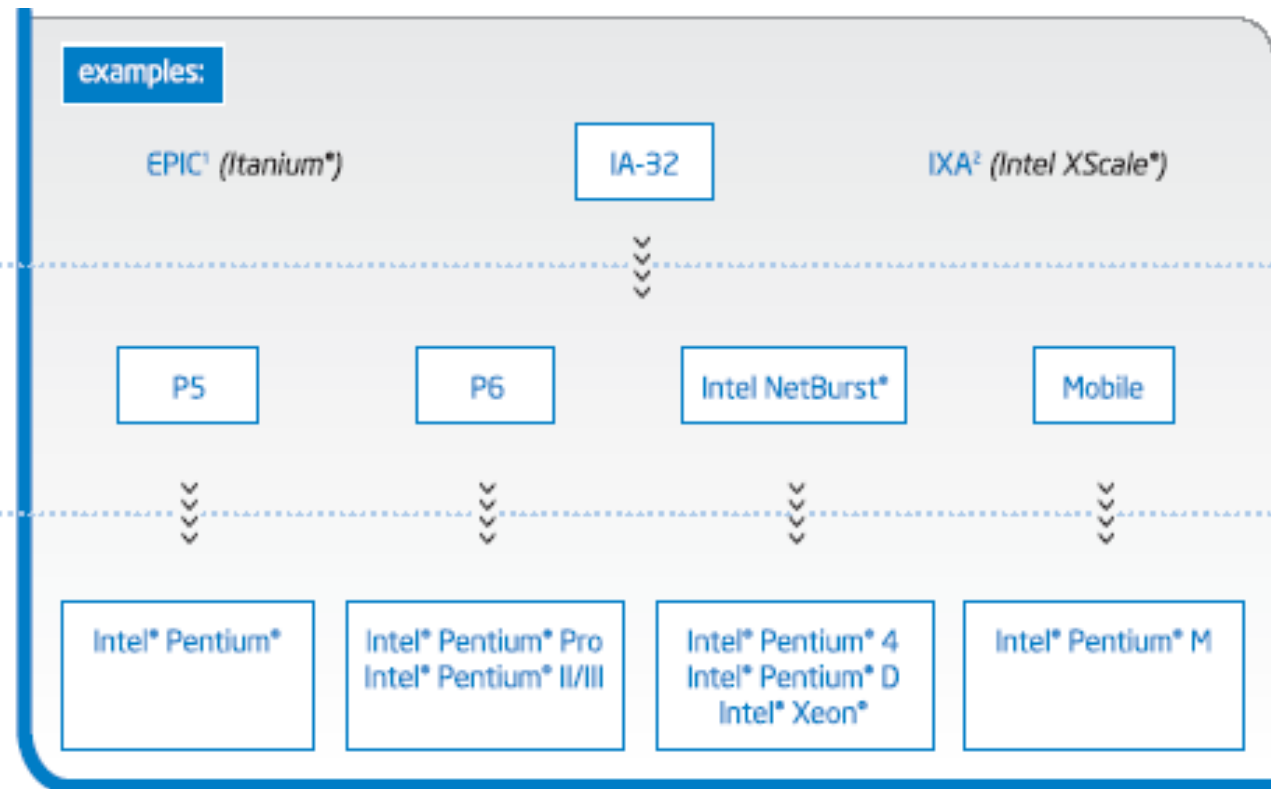


- Arquitectura, Organización, Realización

Architecture
Instruction set definition
and compatibility

Microarchitecture
Hardware implementation
maintaining instruction
set compatibility with
high-level architecture

Processors
Productized implementation
of microarchitecture





- Ejemplo: código ensamblador IA-32

8048374:	55	push	%ebp
8048375:	89 e5	mov	%esp,%ebp
8048377:	83 ec 08	sub	\$0x8,%esp
804837a:	83 e4 f0	and	\$0xfffffffff0,%esp
804837d:	b8 00 00 00 00	mov	\$0x0,%eax
8048382:	29 c4	sub	%eax,%esp
8048384:	c7 45 fc 00 00 00 00	movl	\$0x0,0xffffffffc(%ebp)
804838b:	83 7d fc 09	cmpl	\$0x9,0xffffffffc(%ebp)
804838f:	7e 02	jle	8048393 <main+0x1f>
8048391:	eb 13	jmp	80483a6 <main+0x32>
8048393:	c7 04 24 84 84 04 08	movl	\$0x8048484,(%esp)
804839a:	e8 01 ff ff ff	call	80482a0 <printf@plt>
804839f:	8d 45 fc	lea	0xffffffffc(%ebp),%eax
80483a2:	ff 00	incl	(%eax)
80483a4:	eb e5	jmp	804838b <main+0x17>
80483a6:	c9	leave	
80483a7:	c3	ret	
80483a8:	90	nop	
80483a9:	90	nop	
80483aa:	90	nop	

Organización



- Organización básica de un computador:
 - Procesador (CPU)
 - Sistema de almacenamiento
 - Sistema de entrada/salida
 - Sistemas de interconexión



Arquitectura Von Neumann: computadores que comparten una única memoria para el almacenamiento de datos e instrucciones

Arquitectura Harvard: computadores con memorias separadas para datos e instrucciones

Rendimiento



Compilador

Tecnología

$$T_{CPU} = NI \times CPI \times T_{ciclo} = NI \times CPI / f$$

Repertorio de
Instrucciones

Microarquitectura

Principios básicos de optimización



- ***Paralelismo:***

incrementar el rendimiento del computador mediante la ejecución simultánea de varias instrucciones

- ***Jerarquía de memoria:***

aprovechar el *principio de localidad* para incrementar el rendimiento del sistema de memoria, sustituyendo una memoria única por varias memorias con diferentes prestaciones y tamaños, organizadas de manera jerárquica

Principios básicos de optimización



- Clasificación del paralelismo: Taxonomía de Flynn
 - **SISD** (Single Instruction Single Data)
 - un flujo de instrucciones único trabaja sobre un flujo de datos único (arquitectura clásica, superescalar, VLIW)
 - **SIMD** (Single Instruction Multiple Data):
 - un flujo de instrucciones único trabaja sobre múltiples flujos de datos (computadores vectoriales)
 - **MISD** (Multiple Instruction Single Data):
 - múltiples flujos de instrucciones trabajan sobre un flujo de datos único
 - **MIMD** (Multiple Instruction Multiple Data)
 - múltiples flujos de instrucciones trabajan sobre múltiples flujos de datos (multiprocesadores)

Principios básicos de optimización



- Ejemplo de paralelismo:

```
for i=1 to 4 do
begin
    C[i] = A[i] + B[i];
    D[i] = E[i] - F[i];
    G[i] = K[i] * B[i];
end;
```

Este programa
implica realizar
12 operaciones:
4 sumas, 4 restas
y 4 multiplicaciones
(obviamos las de
control del bucle)

Principios básicos de optimización



- Ejemplo (cont.):

- Ejecución **SISD**: 12 operaciones secuencialmente

```
C[1] = A[1] + B[1];  
D[1] = E[1] - F[1];  
G[1] = K[1] * B[1];  
C[2] = A[2] + B[2];  
D[2] = E[2] - F[2];  
G[2] = K[2] * B[2];
```

```
C[3] = A[3] + B[3];  
D[3] = E[3] - F[3];  
G[3] = K[3] * B[3];  
C[4] = A[4] + B[4];  
D[4] = E[4] - F[4];  
G[4] = K[4] * B[4];
```

Principios básicos de optimización



- Ejemplo (cont.):

- Ejecución **SIMD**: 3 operaciones secuencialmente, 4 en paralelo

```
C = A + B;  
D = E - F;  
G = K * B;
```

```
C[1]=A[1]+B[1]; C[2]=A[2]+B[2]; C[3]=A[3]+B[3]; C[4]=A[4]+B[4];  
D[1]=E[1]-F[1]; D[2]=E[2]-F[2]; D[3]=E[3]-F[3]; D[4]=E[4]-F[4];  
G[1]=K[1]*B[1]; G[2]=K[2]*B[2]; G[3]=K[3]*B[3]; G[4]=K[4]*B[4];
```

Principios básicos de optimización



- Ejemplo (cont.):

- Ejecución **MIMD**: varias alternativas

- 3 operaciones en secuencia, 4 en paralelo

```
C[1]=A[1]+B[1]; C[2]=A[2]+B[2]; C[3]=A[3]+B[3]; C[4]=A[4]+B[4];  
D[1]=E[1]-F[1]; D[2]=E[2]-F[2]; D[3]=E[3]-F[3]; D[4]=E[4]-F[4];  
G[1]=K[1]*B[1]; G[2]=K[2]*B[2]; G[3]=K[3]*B[3]; G[4]=K[4]*B[4];
```

Principios básicos de optimización



- Principio de localidad
 - **Localidad espacial:** tendencia a referenciar posiciones consecutivas de memoria en un corto espacio de tiempo
 - **Localidad temporal:** tendencia a referenciar repetidas veces en un corto espacio de tiempo la misma posición de memoria

Localidad espacial: se accede a instrucciones que están seguidas en memoria

```
R = 0;  
n = 0;  
while (i<=n) {  
    R = R + m;  
    n++;  
}
```

Localidad temporal: se accede de forma repetida a la variable R



Jerarquía de Memoria

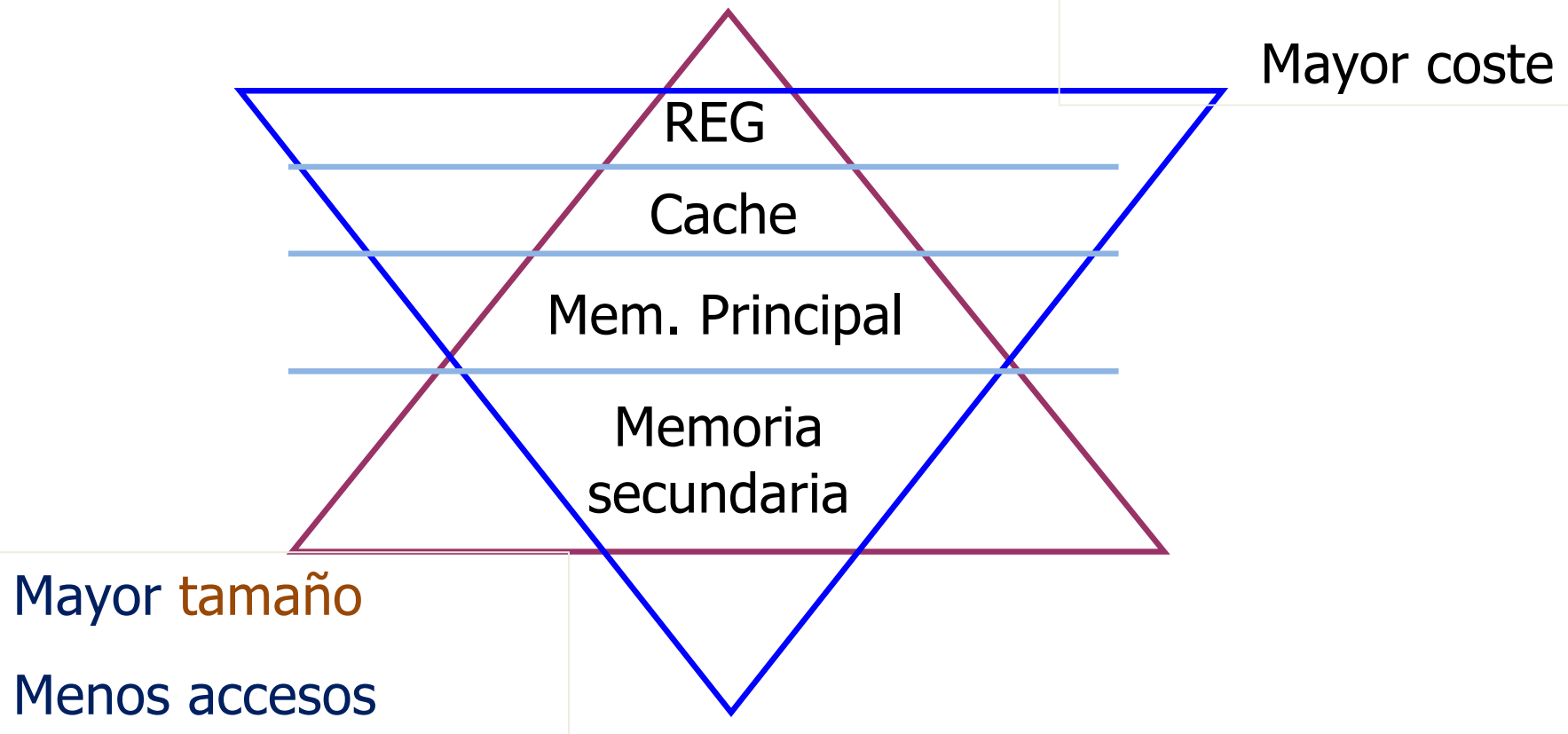
- **Objetivo:** lograr que la velocidad de memoria sea lo más rápida posible, consiguiendo al mismo tiempo un tamaño grande a un coste razonable
- **Principios:** utilizar varios tipos de memoria (de diferentes costes y prestaciones) que se conectan en una jerarquía, aprovechando el principio de localidad



Principios básicos de optimización



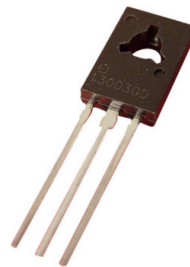
- Jerarquía de memoria



Evolución y tendencias



- Tecnología usada en computadores
 - Válvulas de vacío (años 40 y 50)
 - Primer computador electrónico de propósito general: ENIAC (1946, U. Pensylvania)
 - Primer computador comercial: UNIVAC I (1951)
 - Transistor (años 60)
 - Circuitos integrados (años 70)





Escalas de integración

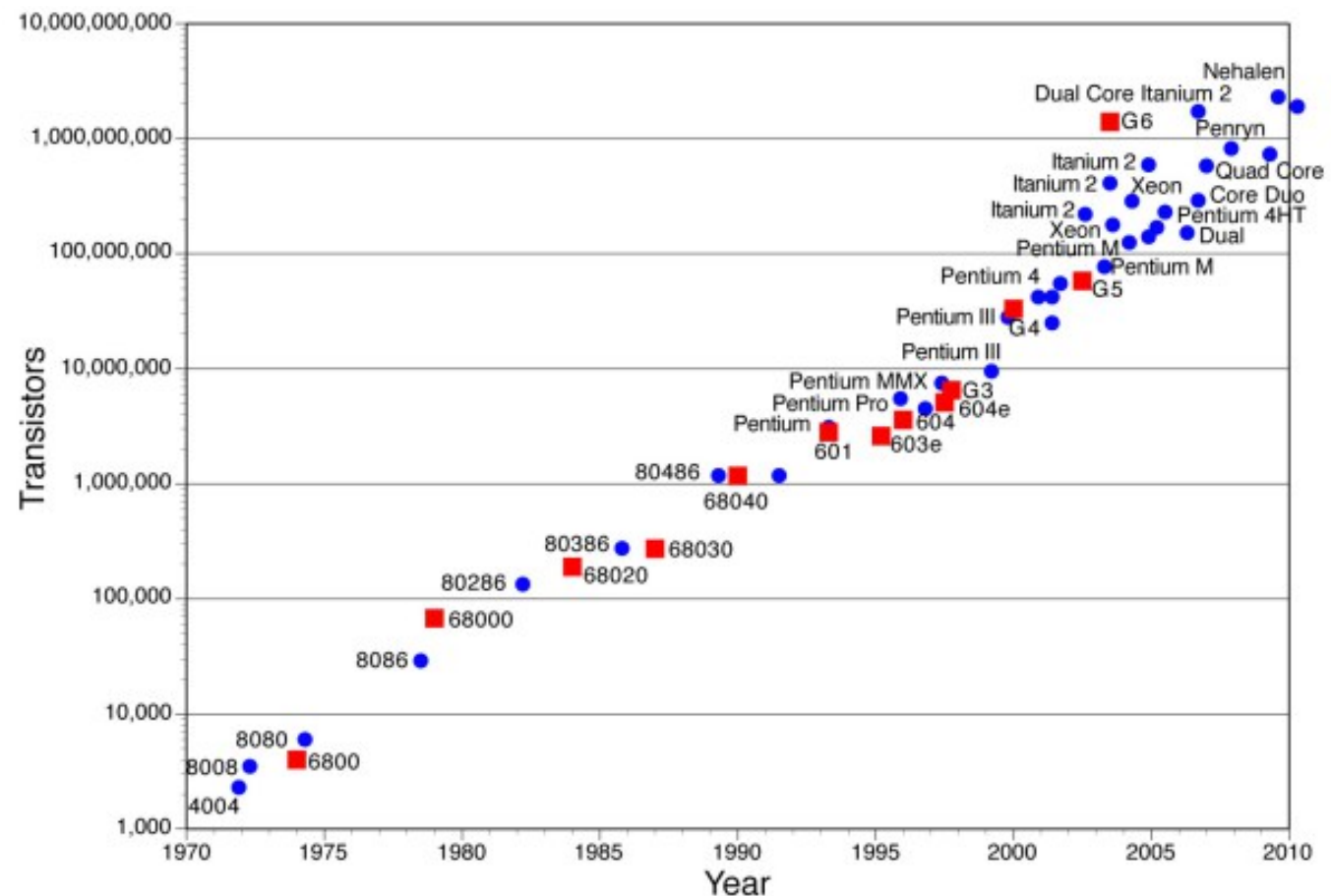
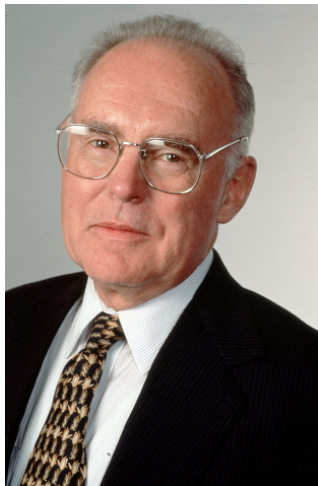
- SSI (Small Scale Integration): 1-100 transistores
- MSI (Medium Scale Integration): 100-1000
- LSI (Large Scale Integration): 10^3 - 10^5
- VLSI (Very Large Scale Integration): 10^5

**¡Procesador
en un chip!**

Evolución y tendencias



- Ley de Moore (1965): “Cada 18 meses se duplica la capacidad de integración”

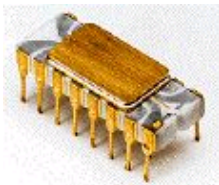


Evolución y tendencias

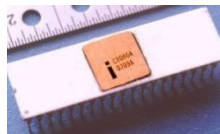


Evolución de la escala de integración

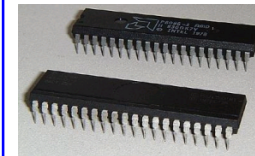
- Ejemplo: microprocesadores de Intel



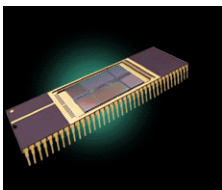
1971: **4004**,
2300 transist.
10 micras



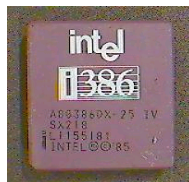
1974: **8080**,
4500 transist.
6 micras



1978: **8086**,
29000 trans.
3 micras



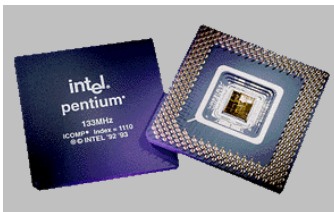
1982: **80286**,
134000 trans.
1,5 micras



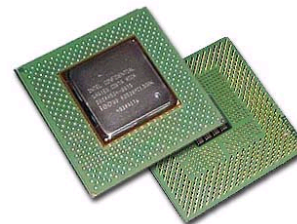
1985: **80386**,
275000 trans.
1,5 micras



1989: **80486**,
1.2 millones
1 micra



1993: **Pentium**,
3.1 millones
0,8 micras



2000: **Pentium 4**,
42 millones
0,18 micras