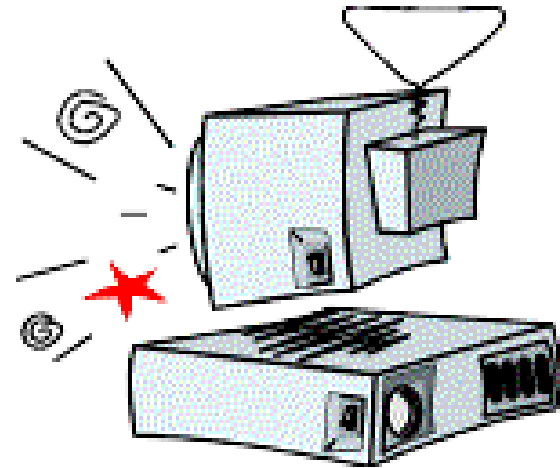
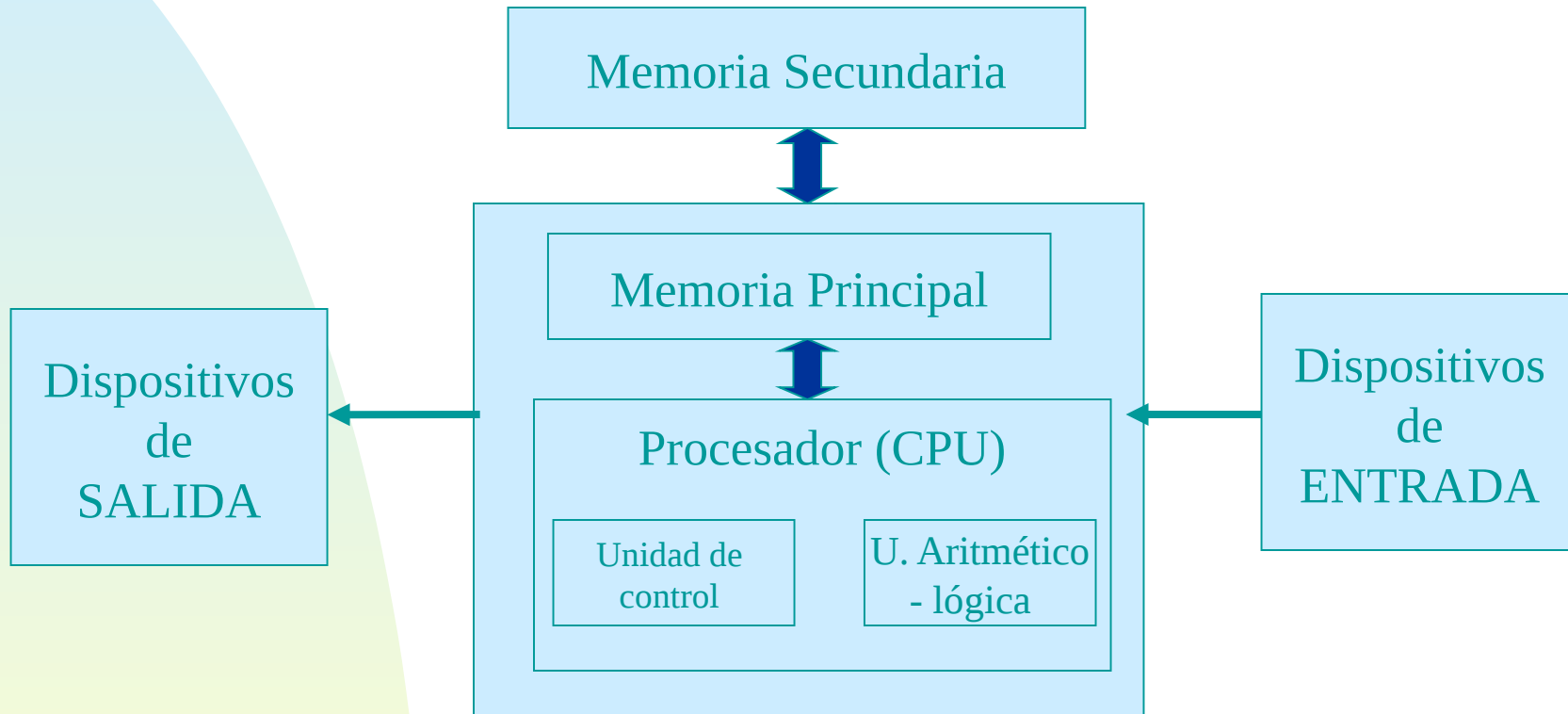


# Introducción a la programación

Arquitectura de un ordenador y funcionamiento



# Esquema funcional



# Memoria Principal

Memoria interna, central :

Conjunto de circuitos (chips de memoria) donde se almacenan los datos y los programas en ejecución

- ◆ Acceso directo desde el procesador
- ◆ Capacidad limitada
- ◆ Dos tipos básicos:

RAM : permite lectura y escritura,  
no permanente

ROM : permite sólo lectura, permante

# Capacidad de la memoria

- Cantidad de información (bits) que puede almacenar, 1 bit (SI / NO)

VALOR	UNIDAD	EQUIVALENCIA
8 bits	<b>1 byte</b>	1 carácter
1024 byte $10^3$ bytes	<b>1 kbytes</b>	1 página
1024 kbyte $10^6$ bytes	<b>1 Megabyte</b>	1 libro
1024 Megas $10^9$ bytes	<b>1 Gigabytes</b>	1 enciclopedia
1024 Gigas $10^{12}$ bytes	<b>1 Terabyte</b>	1 biblioteca
1024 Teras $10^{15}$ bytes	<b>1 Petabyte</b>	?

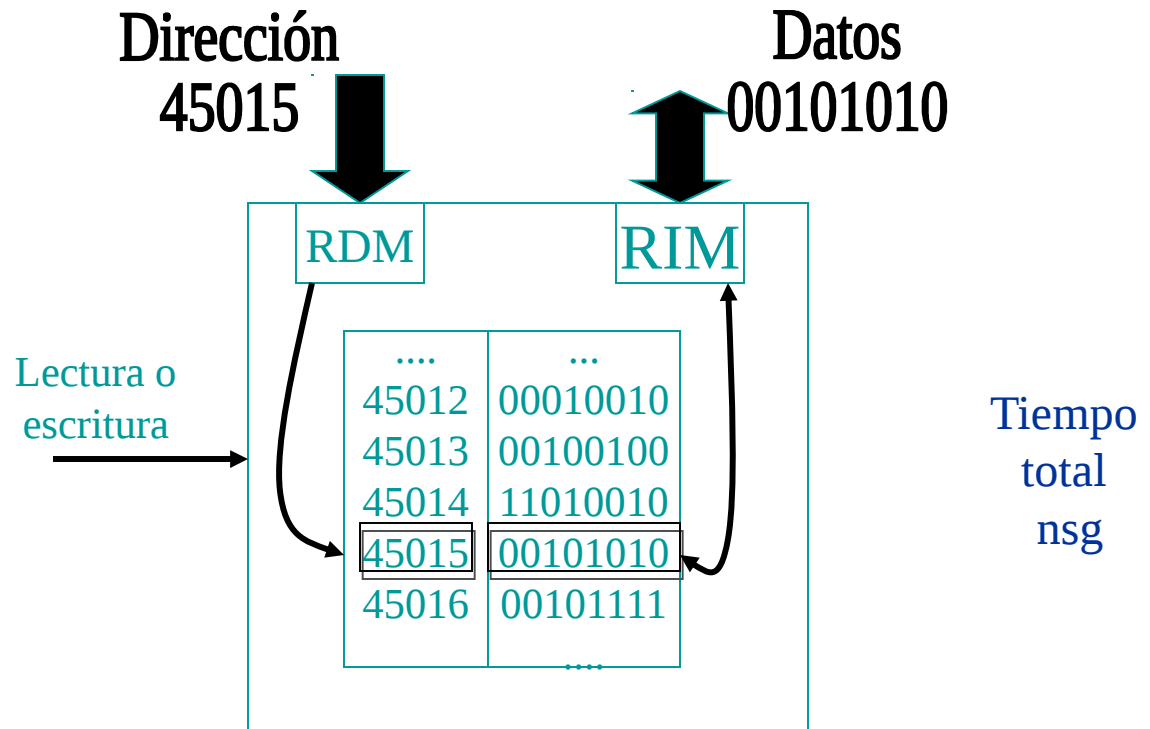
# Organización de la memoria

- Una tabla de celdas o posiciones de memoria con un mismo tamaño (1 byte), donde cada celda tiene un posición o dirección de memoria que la identifica

DIRECCIÓN	CONTENIDO
20012	00100011
20013	11100111
20014	01010000
20015	00100001
.....	...

# Elementos de la Memoria

- RDM: registro de direcciones de Memoria
- RIM: registro de intercambio de Memoria



# Procesador (CPU)

- Elemento central del ordenador: controla y realiza las operaciones del sistema

- ◆ Familia Intel: Distintos modelos Pentium, Celeron

Otros: AMD, Alpha, ARM, Qualcomm

- Características:

- ◆ Tamaño de palabra: 8, 16, 32, 64 bits

- ◆ Núcleos (cores)

- ◆ Tamaño de direcciones : 8, 16, 24, 32 bits

- ◆ Velocidad del reloj: 2 Ghz → Instrucciones por segundo

- ◆ Juego de instrucciones máquina:

- ◆ CISC, RISC, ARM

# Elementos del procesador

- **Unidad de Control:** detecta las señales de estado de los distintos elementos y genera las señales de control a cada uno de ellos
- **Unidad Aritmético-lógica:** formada por circuitos digitales que realizan diversos tipos de operaciones
- Registros específicos y auxiliares
- Memoria Caché



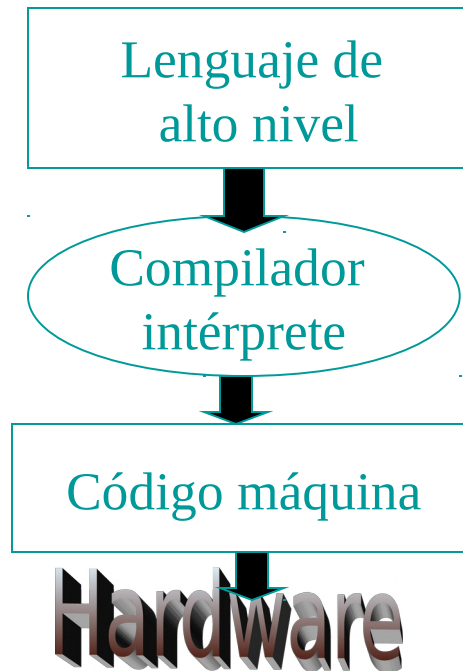
# Ejecución de programas I

- Programa: secuencia de instrucciones que realizan una determinada tarea
- Instrucciones máquina o código máquina: conjunto de instrucciones que es capaz de ejecutar la CPU directamente
- Formato: Ej.
  - ♦ Incrementar contador ( Inc Cont )
  - ♦ 0001010      000100100010

Código de operación	Dirección o dato con que operar
------------------------	------------------------------------

# Ejecución de programas II

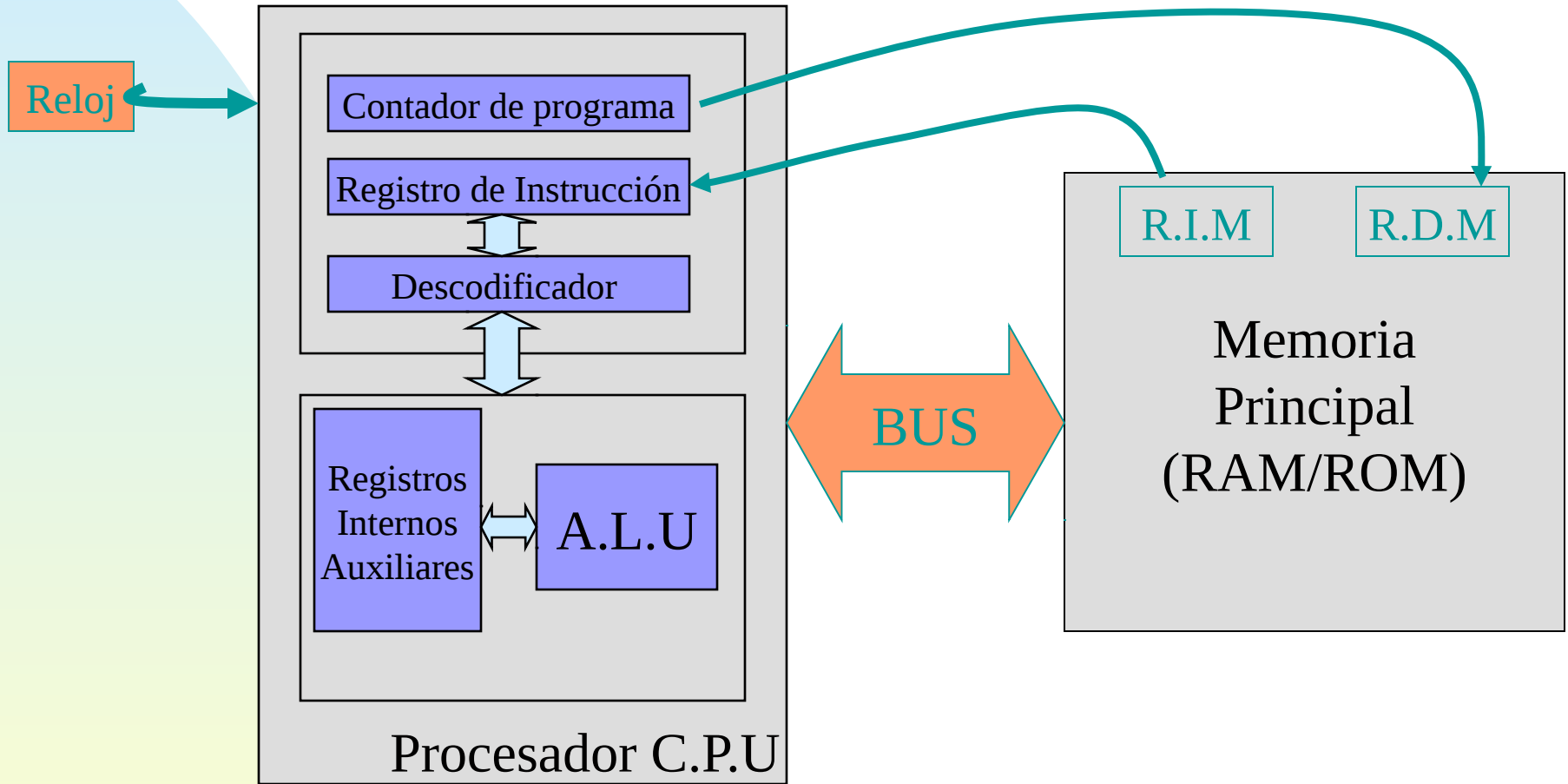
- Lenguaje de alto nivel, cercano al problema: C++, Java, SQL
- Lenguaje máquina, cercano al hardware: cada procesador tiene uno diferente



# Ejecución de programas III

- Tipos de instrucciones de código máquina que puede hacer directamente un ordenados
  - ◆ Aritméticas ( +, - ,\*, %)
  - ◆ Lógicas ( >,< ,=, ≠ )
  - ◆ Control de flujo:
    - ◆ Saltos incondicionales
    - ◆ Saltos condicionales
    - ◆ Rutinas, subprogramas
    - ◆ Interrupciones
  - ◆ Movimiento de datos
    - ◆ Memoria ⇔ procesador
  - ◆ Operaciones de entrada y salida
    - ◆ Lectura o escritura de dispositivos

# Ciclo de ejecución



# Ejemplo programa en código maquina (ensamblador)

```
0x0040104b  31c9      xor ecx, ecx
; JMP XREF from 0x00401061 (entry0)
.--> 0x0040104d  8a8158214000 mov al, byte [ecx + 0x402158] ; [0x402158:1]
| 0x00401053  347d      xor al, 0x7d
| 0x00401055  3a8140214000 cmp al, byte [ecx + 0x402140] ; [0x402140:1]
|,=< 0x0040105b  751e      jne 0x40107b
|| 0x0040105d  41        inc ecx
|| 0x0040105e  83f918    cmp ecx, 0x18 ; "@" @ 0x18
`==< 0x00401061  7cea      jl 0x40104d
| 0x00401063  6a00      push 0
| 0x00401065  8d45fc    lea eax, [ebp-local_1]
| 0x00401068  50        push eax
| 0x00401069  6a12      push 0x12
| 0x0040106b  681c214000 push 0x40211c ; "You are suc
| 0x00401070  ff75f8    push dword [ebp-local_2]
| 0x00401073  ff1564204000 call dword [sym.imp.kernel32.dll_WriteFile] ;
,===< 0x00401079  eb16      jmp 0x401091
```