

Parte I – Conceptos Generales sobre Computadores

1. Arquitectura Von Neumann: el modelo de referencia

En 1946, John Von Neumann propuso un modelo que aún hoy sigue siendo la base de la informática moderna. Su innovación principal fue el concepto de **programa almacenado**: las instrucciones de un programa y los datos se guardan juntos en la memoria, lo que permite que un mismo ordenador ejecute programas diferentes sin necesidad de modificar físicamente el hardware.

La arquitectura Von Neumann consta de **cuatro bloques fundamentales**:

- **Unidad de Control (UC)**: coordina las operaciones del sistema, lee instrucciones de memoria y genera señales de control.
- **Unidad Aritmético-Lógica (ALU)**: realiza operaciones matemáticas y lógicas básicas (sumas, restas, AND, OR, NOT).
- **Memoria Principal**: almacena datos e instrucciones en forma de celdas con direcciones.
- **Dispositivos de Entrada/Salida (E/S)**: permiten la comunicación con el exterior (teclado, pantalla, discos).

La **CPU** agrupa la UC y la ALU, y se comunica con el resto mediante **buses** (de datos, de direcciones y de control).

Este modelo supuso el salto de computadores cableados a computadores **programables**, base de los ordenadores de propósito general. A partir de aquí surgieron todas las generaciones de computadores.

2. Componentes principales del hardware

- **Procesador (microprocesador)**: ejecuta las instrucciones máquina. Parámetros clave: velocidad de reloj (MHz/GHz), memoria caché, núcleos, voltaje y arquitectura (RISC y CISC).
- **Memorias**:
 - **RAM (DDR3, DDR4, DDR5)**: volátil y rápida, permite ejecutar programas.
 - **ROM y sus variantes (PROM, EPROM, EEPROM, Flash)**: no volátiles, usadas en BIOS o dispositivos móviles.
- **Almacenamiento**:
 - **HDD**: discos mecánicos con platos y cabezales.

- **SSD:** unidades electrónicas mucho más rápidas, silenciosas y resistentes.
- **Periféricos:** entrada (ratón, teclado), salida (pantalla, impresora), comunicación (tarjeta de red), almacenamiento (USB, DVD).

3. Tipos de ordenadores

Existen **superordenadores** (usados en ciencia), **mainframes** (banca, empresas), **servidores**, **workstations**, PCs de sobremesa y portátiles.

4. Evolución histórica

De válvulas de vacío (1ª generación) a transistores (2ª), circuitos integrados (3ª), microprocesadores y PCs (4ª), multimedia e interfaces gráficas (5ª) y finalmente multiprocesadores, redes, cloud e inteligencia artificial (6ª).

5. Lenguaje máquina y ensamblador

Los programas son secuencias de instrucciones almacenadas en memoria. Cada instrucción pasa por fases: **fetch (lectura)**, **decodificación**, **ejecución y actualización del contador de programa**.

El **lenguaje ensamblador** traduce estas instrucciones a símbolos legibles por humanos (ADD, MOV, JMP), pero sigue siendo poco portable.

Parte II – Sistemas Operativos

1. Definición y funciones

El sistema operativo (SO) es el software fundamental que controla el hardware, organiza archivos, gestiona usuarios y ofrece una base para ejecutar aplicaciones. Funciones clave:

- Gestión de procesos.
- Gestión de memoria (memoria virtual y paginación).
- Manejo de dispositivos de E/S mediante drivers.
- Gestión de sistemas de archivos.
- Seguridad y control de usuarios.
- Interfaz con el usuario (línea de comandos o GUI).

2. Gestión de memoria: memoria virtual y paginación

Cuando la RAM no es suficiente, el SO utiliza el disco como **memoria virtual**, dando la ilusión de que hay más memoria de la disponible.

La técnica de **paginación** divide programas en páginas y memoria en marcos. El SO mueve páginas de un lado a otro según necesidad, optimizando recursos.

3. Planificación de procesos: algoritmos clave

Cuando varios procesos compiten por la CPU, el sistema operativo debe decidir cuál ejecutar. Aquí entran los **algoritmos de planificación**.

FIFO (First In, First Out)

Es el más simple. Los procesos se atienden en **orden de llegada**:

- El primero que entra en la cola de listos es el primero en ejecutarse.
- El proceso mantiene la CPU hasta que finaliza.

Ejemplo:

Si entran tres procesos (P1: 10 ms, P2: 5 ms, P3: 3 ms):

- Se ejecuta primero P1 (10 ms), luego P2 (5 ms) y finalmente P3 (3 ms).
- Tiempo total = 18 ms.

Ventaja: sencillo y justo.

Inconveniente: si un proceso es muy largo, los demás esperan demasiado (efecto “convoy”).

Round Robin

Se basa en **repartir el tiempo en turnos iguales (quantum)** para cada proceso:

- Cada proceso recibe un intervalo de tiempo fijo (ej. 4 ms).
- Si no termina en su turno, vuelve a la cola y espera el siguiente.

Ejemplo:

Con quantum de 4 ms y procesos P1 (10 ms), P2 (5 ms), P3 (3 ms):

- Turno 1: P1 (4 ms), P2 (4 ms), P3 (3 ms → termina).
- Turno 2: P1 (4 ms), P2 (1 ms → termina).
- Turno 3: P1 (2 ms → termina).

Resultado: todos los procesos avanzan de forma equitativa, reduciendo tiempos de espera percibidos.

Ventaja: da sensación de equidad, ideal para sistemas interactivos.

Inconveniente: demasiados cambios de contexto si el quantum es muy pequeño.

4. Manejo de E/S e interrupciones

El SO gestiona almacenamiento (HDD, SSD), comunicación (redes) e interfaz de usuario (pantalla, teclado). Los drivers convierten órdenes generales en instrucciones específicas para cada dispositivo.

5. Bloqueo de recursos (Deadlock)

Ocurre cuando dos procesos esperan indefinidamente recursos que tiene el otro. Ejemplo clásico: dos procesos que se bloquean entre sí al pedir impresora y memoria.

El SO debe prevenir, evitar o resolver estos bloqueos.

6. Sistemas de archivos

- **NTFS (Windows):** soporta cifrado y compresión.
- **HFS+ (Apple):** jerárquico, usado en macOS.
- **EXT4 (Linux):** rápido y eficiente.
- **Journaling:** los cambios se registran como transacciones, garantizando la integridad tras un fallo.

7. Multiproceso, multitarea, multihilo y multiusuario

- **Multiproceso:** varios procesadores físicos ejecutando procesos en paralelo.
- **Multitarea:** múltiples tareas a la vez en un solo procesador.
- **Multihilo:** varios hilos dentro de un proceso, que comparten memoria.
- **Multiusuario:** varios usuarios simultáneos en un mismo sistema.

8. Virtualización y núcleos virtuales

- La virtualización permite ejecutar varias máquinas virtuales sobre un mismo host físico.
- **Hypervisores:** sistemas mínimos que solo gestionan máquinas virtuales.
- Tipos: virtualización total, paravirtualización y asistida por hardware.
- **Hyper-threading:** un núcleo físico simula dos núcleos virtuales, mejorando la eficiencia aunque nunca alcanza a un núcleo físico real.