

Unidades Funcionales del PC

- **CPU:**
 - **Unidad de control del ordenador:** Contiene la instrucción que se está ejecutando en ese instante. Su ejecución está siendo “atendida por el Kernel”.
 - **Unidad Aritmético Lógica:** Opera con los datos procedentes de los registros de entrada almacena los resultados de la operaciones y los envía por el BUS a la MC o UC.
- **RAM (donde se almacenan todas las instrucciones y datos):**
 - **Selector de memoria:** Permite la transferencia de datos entre el RDM y el RIM en ambos sentidos .
 - **RDM (Registro de Dirección de Memoria):** Dirección de la celda a partir de la que se va a leer o escribir datos.
 - **RIM (Registro de Intercambio de Memoria):** Si la operación es de lectura recibe el dato del RDM, y si es de escritura almacena la información que hay que grabar procedente de cualquier unidad funcional.
- **Bus del sistema (conjunto de circuitos integrados que se encargan de la conexión y comunicación entre la CPU y el resto de unidades de la máquina):**
 - **Líneas de datos:** Los “camino” físicos por donde se transmiten los datos.
 - **Líneas de dirección:** Sirven para seleccionar el destino de la información que hay sobre el bus de datos.
 - **Líneas de control:** Gestionan el uso y acceso a los buses de datos y de dirección.

Operaciones:

Están compuestas por: **Código de operación** que indica la operación que se va a realizar, y por los **Operandos** que son los valores que se usarán para realizar la operación.

Tipos:

- **Instrucciones de 3 operandos (I+O+O+O).**
- **Instrucciones de 2 operandos (I+O+O).**
- **Instrucciones de un operando (I+O).**
- **Instrucciones sin operandos (I).**

Métodos de direccionamiento:

- **Direccionamiento inmediato:** La instrucción contienen el dato que hay que emplear, no siendo necesario el acceso a memoria .
- **Direccionamiento directo:** La instrucción contiene la dirección de memoria central donde se encuentra el dato.
- **Direccionamiento indirecto:** La instrucción contiene la posición de memoria que contiene la dirección de ese dato.
- **Direccionamiento relativo:** La dirección es calculada. La dirección se obtiene sumando la dirección contenida en la propia instrucción con una magnitud fija contenida en un

registro especial.

Procesos (7):

Es un espacio de dirección de memoria específico donde se va a ejecutar un fichero.
Se utiliza el **PID (Identificador de procesos)** para identificar procesos.

+ **Estado del proceso:** Conjunto de valores de los registros de la unidad de control y de la Unidad Aritmética Lógica .

- **Estado Nuevo:** El proceso está siendo creado a partir del fichero ejecutable.
- **Estado Listo:** No está en ejecución y está esperando preparado para hacerlo a que el S.O. le asigne un procesador.
- **En ejecución:** Se pasa a este estado cuando se está al comienzo de la cola de procesos preparados, y el kernel atiende la instrucción en curso.
- **Estado Bloqueado:** El proceso está bloqueado esperando a que ocurra algún suceso.
- **Estado Terminado:** El proceso finaliza su ejecución y el S.O libera su imagen de memoria.

+ **Momento del proceso:** Es el valor del **registro contador del programa** y contiene la dirección de memoria de la instrucción que va a ejecutar.

+ **Colas de procesos:** Contienen a todos los procesos existentes.

- **Colas de procesos preparados:** Contiene a todos los procesos que están en estado Listo.
- **Colas de dispositivos:** Contiene a todos los procesos que están en estado Bloqueado.

+ **Planificación de procesos:**

Las Políticas de planificación son procesos que se ejecutan concurrentemente.
Gracias al **cambio de contexto** se puede interrumpir un proceso en ejecución para pasar a otro.

- **Políticas de planificación de procesos a corto plazo:**

- **Planificación sin desalojo o cooperativa:** Sólo se cambia de proceso en ejecución si dicho proceso se bloquea o termina.
- **Planificación con desalojo o apropiativa:** Si aparece en estado listo un proceso de mayor prioridad el proceso que se está ejecutando se interrumpe o desaloja.
- **Tiempo compartido:** Se desaloja el proceso en ejecución porque finaliza su cuanto de ejecución.
 - **Cuanto de ejecución:** Cantidad máxima de tiempo que un proceso puede estar en ejecución.

- **Políticas de planificación de procesos a largo plazo:**

- El planificador crea los nuevos procesos y controla el grado de multiprogramación.

+ **Contexto de un proceso:** El estado del proceso y el espacio de memoria reservada para él.

+ **Árbol de procesos:** Una aplicación multiproceso es un árbol n-ario de procesos. Todos los procesos de ese árbol cooperan en la realización de una tarea común, pero para poder hacerlo deben

comunicarse y sincronizarse, esta comunicación se establece a través de flujos o streams de bytes de información.

El proceso que ejecuta el **método start()** se convierte en el proceso padre del proceso al que va referenciado el método.

En general los procesos de un equipo son independientes y se ejecutan asincrónicamente.

+ Comunicación de procesos:

Buffer (espacio de memoria interna).

Por defecto están definidos 3 buffers de datos:

- **Buffer de entrada estándar (stdin):** Buffer donde el proceso lee los datos de entrada que requiere para su ejecución.
- **Buffer de salida estándar (stdout):** Buffer donde el proceso escribe los datos de salida que obtiene.
- **Buffer de errores (stderr):** Buffer donde el proceso envía los mensajes de error.

Conceptos Varios (8):

- **Código fuente:** Contiene el algoritmo de resolución de una tarea.
- **Código objeto:** Traducción del código fuente a lenguaje máquina interno.
- **Ejecutable:** Fichero binario que contiene la información necesaria para crear un proceso.
- **Palabra de memoria:** Cantidad de información que se puede leer o escribir en la memoria central en una sola aplicación.
- **Desbordamiento de pila:** Cuando las instrucciones o datos almacenados sobrepasan el espacio de la pila.
- **Demonio:** Proceso que está siempre en ejecución, no es interactivo y normalmente presta un servicio a otros procesos.
- **Kernel (núcleo del S.O):** El proceso se comunica con el Kernel, que según el tipo de interrupción ejecuta una determinada rutina en respuesta.
- **Modo dual:** Característica del hardware que permite al sistema protegerse.
 - **Modo usuario:** Cuando la instrucción en ejecución pertenece a cualquiera de los procesos en ejecución.
 - **Modo Kernel:** Cuando ha habido una interrupción y se está ejecutando la rutina correspondiente a dicha interrupción.