**Clasificación general de registros**

**1. Registros visibles al usuario (programador)**

Se pueden usar directamente en instrucciones de programa.

**a) De propósito general**

* Usados libremente en operaciones.
* Ej: AX, BX, CX, DX (en x86).
* Algunas arquitecturas limitan su uso a ciertos contextos (enteros, punto flotante, direcciones).

**b) Registros de datos**

* Específicos para almacenar operandos de datos.
* Se relacionan con el tamaño de la ALU (por ejemplo, 8, 16 o 32 bits).
* No suelen usarse para direcciones.

**c) Registros de direcciones**

* Usados para guardar direcciones de memoria.
* Asisten a modos de direccionamiento como indexado, indirecto, etc.
* Su tamaño depende del tamaño del espacio de direcciones de la CPU.

**d) Registros de estado (flags)**

* Almacenan **bits de estado** del procesador.
* Ejemplos: **Carry (C), Zero (Z), Overflow (V), Negative (N)**.
* Usados para decisiones lógicas, saltos condicionales, interrupciones, etc.
* A veces son modificables por el usuario, otras no.

**2. Registros invisibles al usuario**

Solo los usa la **Unidad de Control (UC)** del procesador.

**Incluyen:**

* **PC (Program Counter):** Dirección de la próxima instrucción.
* **IR (Instruction Register):** Guarda la instrucción actual.
* **MAR (Memory Address Register):** Dirección que se usará para acceder a memoria.
* **MBR (Memory Buffer Register):** Dato que entra o sale de memoria.

También pueden haber **registros temporales internos** que manejan transferencias internas de datos.

**Características clave de los registros**

**➤ Tamaño**

* **Datos:** tamaño adecuado para los tipos de datos comunes (8, 16, 32, 64 bits).
* **Direcciones:** deben cubrir el espacio de direcciones posible.

**➤ Cantidad**

* Cuantos más registros, mejor rendimiento.
* Pero más registros requieren **más bits en las instrucciones** para codificarlos.
* Un número típico: entre **8 y 32 registros**.

**➤ Uso**

* Pueden ser **todos de propósito general**, lo que da flexibilidad.
* O **especializados** (como en arquitecturas tipo acumulador), lo que ahorra bits en las instrucciones.

**Ejemplos en procesadores reales**

**En x86:**

* **Registros de datos:** AX, BX, CX, DX (y sus variantes AL, AH, etc.).
* **Registros de direcciones:** SP (stack pointer), BP (base pointer), SI, DI.
* **Registros de segmentos:** CS (código), DS (datos), SS (stack), ES, FS, GS.
* **Registros de control:** IP/EIP (program counter), FLAGS (estado).

**En Motorola 68000:**

* Datos: D0–D7
* Direcciones: A0–A7
* Control: PC, Estado