

# ISA y Endianness

Arquitectura de computadoras

Gomez Breschi Emilio

ingeniería en informática

marzo 1, 2026

## 1) ISA: ¿Qué es Instruction Set Architecture?

La **ISA** es como el convenio entre el **software** y el **procesador**. Define qué instrucciones entiende la CPU. Por ejemplo sumar, mover datos, saltar, qué registros existen, cómo se accede a memoria y ciertas reglas de cómo se comporta el procesador. Gracias a eso, programas compilados para una ISA pueden correr en diferentes procesadores que implementen esa misma ISA, aunque por dentro sean distintos.

### Dos ejemplos de ISA

- **x86 / x86-64** (muy común en Computadoras actuales)
- **Arm** (muy común en celulares, tablets, y también algunas laptops o servidores)

## 2) Diferencia: Arquitectura RISC vs CISC (explicación breve)

- **RISC (Reduced Instruction Set Computer)**: suele usar un conjunto de instrucciones más simples y parejas. Muchas veces se apoya en la idea de cargar y guardar desde la memoria con instrucciones muy específicas, y que la mayoría de las operaciones trabajen con registros. Esto facilita optimizar y hacer pipelines más limpios en muchos diseños.
- **CISC (Complex Instruction Set Computer)**: suele tener **más instrucciones** y algunas pueden ser más complejas. Históricamente buscaba que el código fuera más compacto y que ciertas tareas se hicieran con menos instrucciones.

## 3) Uso actual: ¿Qué Endianness usa el procesador de tu computadora?

Mi procesador es de AMD así que es compatible con la arquitectura x86 y x64. Esos procesadores usan **little-endian** para almacenar los datos en memoria.

1 marzo - 2026

## Representación en Memoria del valor $0x12345678$

Big Endian

1000	12
1001	34
1002	56
1003	78

Byte más significativo  
primero

Little Endian

1000	78
1001	56
1002	34
1003	12

Byte menos significativo  
primero