

SISD (SINGLE INSTRUCTION, SINGLE DATA – INSTRUCCIÓN UNICA, DATOS UNICOS):

Este tipo de arquitectura es el más sencillo y tradicional. En este modelo, una única unidad de procesamiento ejecuta una instrucción a la vez sobre un solo conjunto de datos. Es similar al modelo secuencial que se utiliza comúnmente en computadoras con un solo núcleo. Un ejemplo de esto serían los procesadores tradicionales, y se pueden usar lenguajes de programación como C, Python o Java para trabajar con ellos.

SIMD (SINGLE INSTRUCTION, MULTIPLE DATA – INSTRUCCIÓN UNICA, MÚLTIPLES DATOS):

En la arquitectura SIMD, un único conjunto de instrucciones se aplica simultáneamente a múltiples datos. Esto es ideal para situaciones donde se necesita realizar la misma operación en un gran volumen de datos al mismo tiempo, como cuando una tarjeta gráfica procesa todos los píxeles de una imagen de una sola vez. Un ejemplo de esto son las GPUs, que procesan muchos datos en paralelo. Los lenguajes utilizados incluyen CUDA y OpenCL para GPUs, o incluso C y C++ cuando se emplean instrucciones especiales para manejar múltiples operaciones simultáneamente.

MISD (MULTIPLE INSTRUCTION, SINGLE DATA – MÚLTIPLES INSTRUCCIONES, DATOS UNICOS):

Este tipo de arquitectura es bastante raro. Aquí, diferentes instrucciones se aplican al mismo conjunto de datos. Aunque no es común en la vida diaria, puede ser útil en sistemas donde la seguridad y la redundancia son prioritarias, como en los sistemas de control de vuelo. No hay un lenguaje específico para esta arquitectura, pero lenguajes como Ada o VHDL podrían ser utilizados en sistemas embebidos que buscan evitar fallos.

MIMD (MULTIPLE INSTRUCTION, MULTIPLE DATA – MÚLTIPLES INSTRUCCIONES, MÚLTIPLES DATOS):

Este modelo es el más avanzado y flexible. En él, diferentes procesadores trabajan de manera independiente, ejecutando distintas instrucciones sobre diferentes conjuntos de datos. Esto permite realizar múltiples tareas al mismo tiempo, como en servidores o supercomputadoras. Ejemplos de esto son los procesadores multinúcleo o las supercomputadoras. Los lenguajes utilizados incluyen OpenMP, MPI, o incluso Python con manejo de hilos o procesos.