



EDUCACIÓN

SECRETARÍA DE EDUCACIÓN PÚBLICA



TECNOLOGICO
NACIONAL DE MÉXICO

Instituto Tecnológico de Morelia

Arquitectura de Computadoras

Unidad 4: Procesamiento paralelo.

Presentado por:

**Alberto Montoya Arriaga
Michael Aaron Villalón Nieves**

Ingeniería en Sistemas Computacionales

Docente:

Servando González Hernández

Morelia, Michoacán, México. 12 de diciembre de 2025.

Índice

1. Introducción.	3
2. Taxonomía de Flynn	4
2.1. SISD (Single Instruction, Single Data)	5
2.2. SIMD (Single Instruction, Multiple Data)	5
2.3. MISD (Multiple Instruction, Single Data)	6
2.4. MIMD (Multiple Instruction, Multiple Data)	7
2.5. Resumen	8
3. Multiprocesamiento	8
3.1. Simétrico (SMP)	9
3.2. Heterogéneo	10
3.3. Cluster	10
4. Tipos de paralelismo	11
4.1. A nivel de bit	11
4.2. A nivel de instrucción	11
4.3. De datos	11
4.4. De tareas	11
5. Conclusión.	11
6. Bibliografía.	11

1. Introducción.

El procesamiento paralelo es un método del campo de la computación que permite que dos o más procesadores de una computadora se utilicen para trabajar en partes separadas de una tarea. De esta manera, es posible reducir el tiempo dedicado a resolver el problema.

El concepto de computación paralela comenzó a desarrollarse a finales de la década de 1950 por investigadores de IBM. Ellos creían que una sola computadora no satisfaría la creciente demanda de potencia de procesamiento. Una posible solución sería tener dos procesadores (o núcleos) trabajando simultáneamente.

El primer chip comercial con múltiples núcleos fue el IBM Power4, lanzado en 2001. El procesador, basado en la arquitectura PowerPC, era un dual-core con una frecuencia de 1,1 a 1,3 GHz. La CPU, que fue la primera en tener dos núcleos en un solo chip de silicio, se fabricaba en una litografía de 180 nanómetros.

Hay varias formas diferentes de computación paralela: Paralelismo a nivel de bit, paralelismo a nivel de instrucción, paralelismo de datos y paralelismo de tareas. El paralelismo se ha empleado durante muchos años, sobre todo en la computación de altas prestaciones, pero el interés en ella ha crecido últimamente debido a las limitaciones físicas que impiden el aumento de la frecuencia. Como el consumo de energía —y por consiguiente la generación de calor— de las computadoras constituye una preocupación en los últimos años, la computación en paralelo se ha convertido en el paradigma dominante en la arquitectura de computadores, principalmente en forma de procesadores multinúcleo.

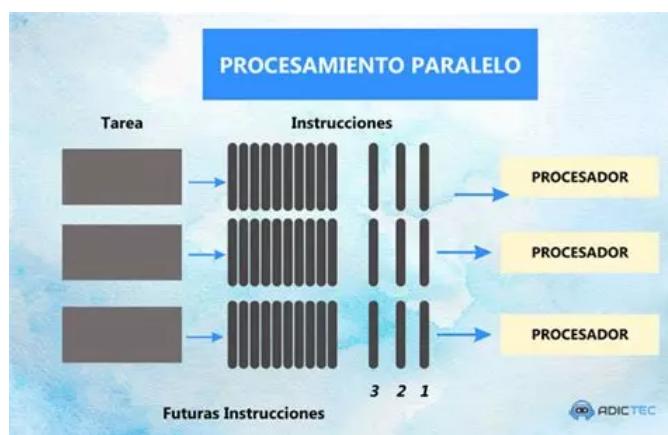


Imagen 1: Imagen de ilustracion.

2. Taxonomía de Flynn

La taxonomía de Flynn es un sistema de clasificación de arquitecturas que se basa en la idea de cuántos flujos de instrucciones y cuántos flujos de datos puede manipular una computadora simultáneamente. Fue creada por el científico de la computación Michael J. Flynn en 1966 y se utiliza hasta el día de hoy en el concepto de computación paralela.

El flujo de instrucciones es una secuencia de instrucciones ejecutadas por el procesador. Una instrucción es una “orden” dada al chip para realizar una operación específica, como una adición o una resta. Cuando tenemos varias órdenes consecutivas, tenemos un flujo de instrucciones.

Por otro lado, el flujo de datos es el conjunto de datos sobre los cuales se ejecutan las instrucciones. Volviendo al ejemplo matemático anterior, los números que se sumarían o restarían serían el flujo de datos en un procesador.

La taxonomía de Flynn divide los sistemas en cuatro categorías:

- **SISD** (Single Instruction, Single Data): es el modelo de computadora más simple, en el cual una sola instrucción opera en un solo flujo de datos. La mayoría de los primeros chips funcionaban de esta manera.
- **MISD** (Multiple Instruction, Single Data): varias instrucciones se ejecutan sobre los mismos datos. Es un modelo teórico e inusual en el mundo real.
- **SIMD** (Single Instruction, Multiple Data): una sola instrucción se aplica a varios flujos de datos simultáneamente. Puede ser útil en procesadores vectoriales, como los utilizados en NPUs para inteligencia artificial.
- **MIMD** (Multiple Instruction, Multiple Data): varias instrucciones operan en varios flujos de datos. Es el modelo más utilizado en los procesadores modernos con múltiples núcleos y se puede encontrar en todas las categorías de dispositivos electrónicos, desde pequeños teléfonos inteligentes hasta grandes servidores.

2.1. SISD (*Single Instruction, Single Data*)

Flujo único de instrucciones y flujo único de datos. Una computadora secuencial que no aprovecha el paralelismo ni en las instrucciones ni en los flujos de datos. La unidad de control único (CU) obtiene un único flujo de instrucciones (IS) de la memoria. Luego, la CU genera señales de control apropiadas para dirigir un solo elemento de procesamiento (PE) para operar en un solo flujo de datos (DS), es decir, una operación a la vez.

Ejemplos de arquitecturas SISD son las máquinas tradicionales de un solo procesador, como las computadoras personales (PC) más antiguas (en 2010, muchas PC tenían varios núcleos) y las computadoras centrales.

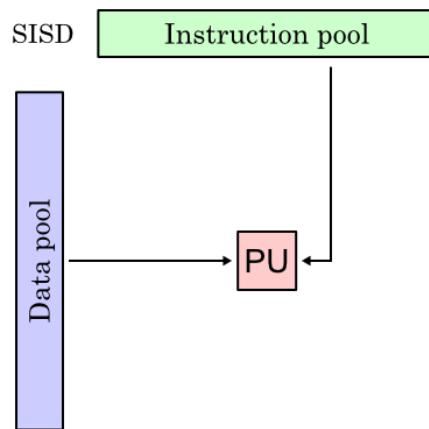


Imagen 2: Diseño de la taxonomía SISD.

2.2. SIMD (*Single Instruction, Multiple Data*)

Flujo de instrucción simple y flujo de datos múltiple. Esto significa que una única instrucción es aplicada sobre diferentes datos al mismo tiempo. En las máquinas de este tipo, varias unidades de procesado diferentes son invocadas por una única unidad de control. Al igual que las MISD, las SIMD soportan procesamiento vectorial (matricial) asignando cada elemento del vector a una unidad funcional diferente para procesamiento concurrente.

El artículo de Flynn de 1972 subdividió SIMD en tres categorías adicionales:

- **Procesador de rayos:** Estos reciben la instrucción (samo) pero cada unidad de procesamiento paralelo tiene su propio archivo de memoria y registro separado y distinto.

- **Procesador de tuberías:** Estos reciben la instrucción (samo) pero luego leen los datos de un recurso central, cada uno procesa fragmentos de esos datos, luego escribe los resultados al mismo recurso central. En la Figura 5 del papel de Flynn 1972 que el recurso es la memoria principal: para las CPU modernas que el recurso es ahora más típicamente el archivo de registro.
- **Procesador asociado:** Estos reciben la instrucción (samo) pero en cada unidad de procesamiento paralelo una independiente se toma la decisión sobre la base de datos local a la unidad, en cuanto a si realizar la ejecución o si saltarla. En terminología moderna esto se conoce como "predicado"(masked) SIMD.

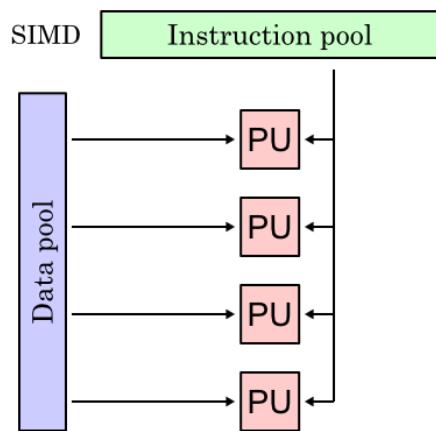


Imagen 3: Diseño de la taxonomía SIMD.

2.3. MISD (*Multiple Instruction, Single Data*)

Flujo múltiple de instrucciones y único flujo de datos. Esto significa que varias instrucciones actúan sobre el mismo y único trozo de datos. Este tipo de máquinas se pueden interpretar de dos maneras. Una es considerar la clase de máquinas que requerirían que unidades de procesamiento diferentes recibieran instrucciones distintas operando sobre los mismos datos. Esta clase de arquitectura ha sido clasificada por numerosos arquitectos de computadores como impracticable o imposible, y en estos momentos no existen ejemplos que funcionen siguiendo este modelo.

Las arquitecturas segmentadas, o encauzadas, realizan el procesamiento vectorial a través de una serie de etapas, cada una ejecutando una función particular produciendo un resultado intermedio. La razón por la cual dichas arquitecturas son clasificadas como MISD es que los elementos de un vector pueden ser

considerados como pertenecientes al mismo dato, y todas las etapas del cauce representan múltiples instrucciones que son aplicadas sobre ese vector.

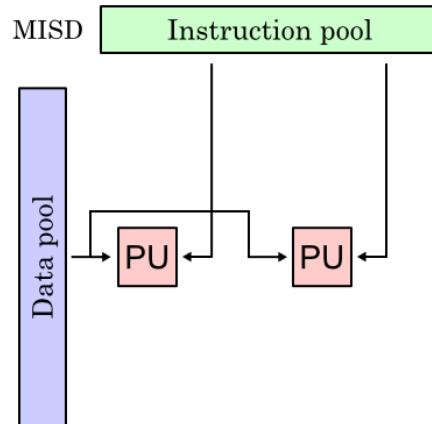


Imagen 4: Diseño de la taxonomía MISD.

2.4. MIMD (*Multiple Instruction, Multiple Data*)

Flujo de instrucciones múltiple y flujo de datos múltiple. Son máquinas que poseen varias unidades procesadoras en las cuales se pueden realizar múltiples instrucciones sobre datos diferentes de forma simultánea. Las MIMD son las más complejas, pero son también las que potencialmente ofrecen una mayor eficiencia en la ejecución concurrente o paralela. Aquí la concurrencia implica que no sólo hay varios procesadores operando simultáneamente, sino que además hay varios programas (procesos) ejecutándose también al mismo tiempo.

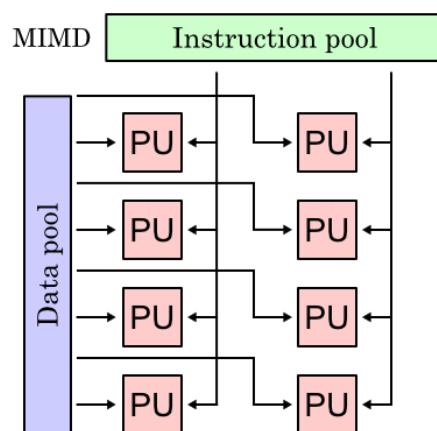


Imagen 5: Diseño de la taxonomía MIMD.

2.5. Resumen

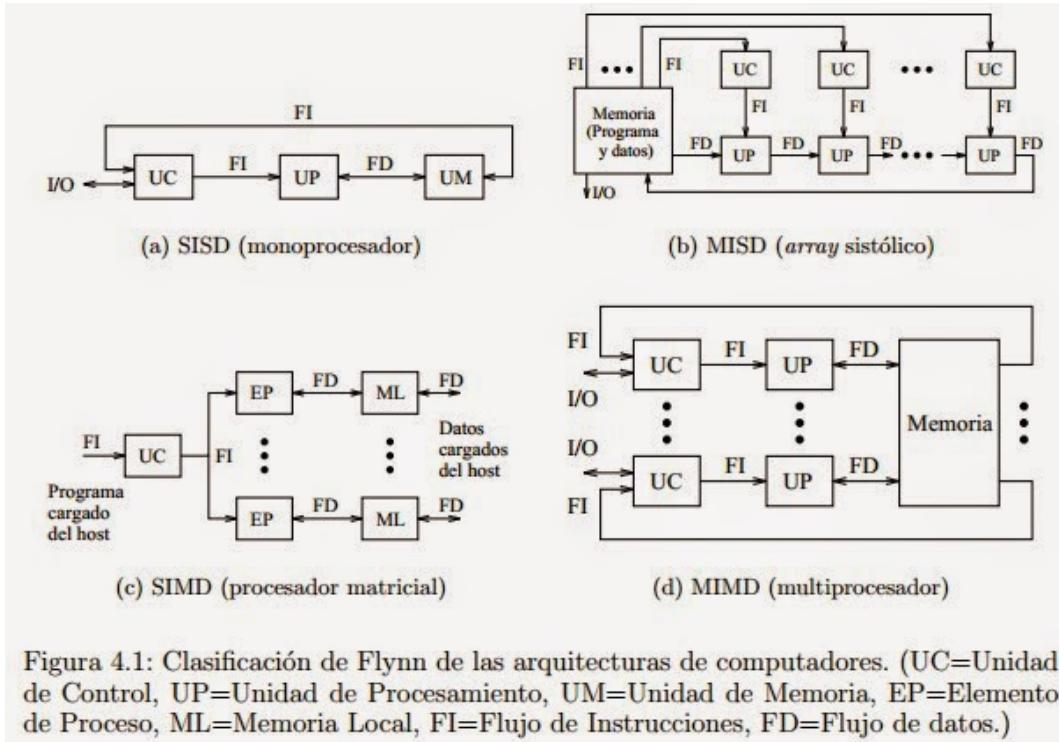


Imagen 6: Resumen de los diseños de cada taxonomía

3. Multiprocesamiento

Multiprocesamiento o multiproceso es el uso de dos o más procesadores (CPU) en una computadora para la ejecución de uno o varios procesos (programas corriendo). Algunas personas, en el idioma español hacen sinónimo este término con el de multitareas (del inglés multitasking) el cual consiste en la ejecución de uno o más procesos concurrentes en un sistema. Así como la multitarea permite a múltiples procesos compartir una única CPU, múltiples CPU pueden ser utilizados para ejecutar múltiples procesos o múltiples hilos (threads) dentro de un único proceso.

El multiprocesamiento se ha empleado desde los años 60 en los entornos de cómputo de alto rendimiento; a pesar de esto, durante muchos años no muchos tomaban esta área de especialización, una computadora que contara con más de un procesador era cara y debido a esto muchos decidían no hacer uso de más de un procesador. Hasta que en 2005 y después de cumplirse 40 años del modelo conocido como Ley de Moore, se empezaron a exceder los 3 GHz de velocidad de esta manera creando problemas de calentamiento motivando el uso de múltiples procesadores.

3.1. Simétrico (SMP)

El multiprocesamiento simétrico o multiprocesamiento de memoria compartida (SMP) involucra una arquitectura de hardware y software de computadora multiprocesador donde dos o más procesadores idénticos están conectados a una sola memoria principal compartida, tienen acceso total a todos los dispositivos de entrada y salida, y están controlados por una sola instancia del sistema operativo que trata a todos los procesadores por igual, sin reservar ninguno para propósitos especiales. La mayoría de los sistemas multiprocesador actuales utilizan una arquitectura SMP. En el caso de los procesadores multinúcleo, la arquitectura SMP se aplica a los núcleos y los trata como procesadores independientes.

Los sistemas SMP permiten que cualquier procesador trabaje en cualquier tarea sin importar su localización en memoria; con un propicio soporte del sistema operativo, estos sistemas pueden mover fácilmente tareas entre los procesadores para garantizar eficientemente el trabajo.

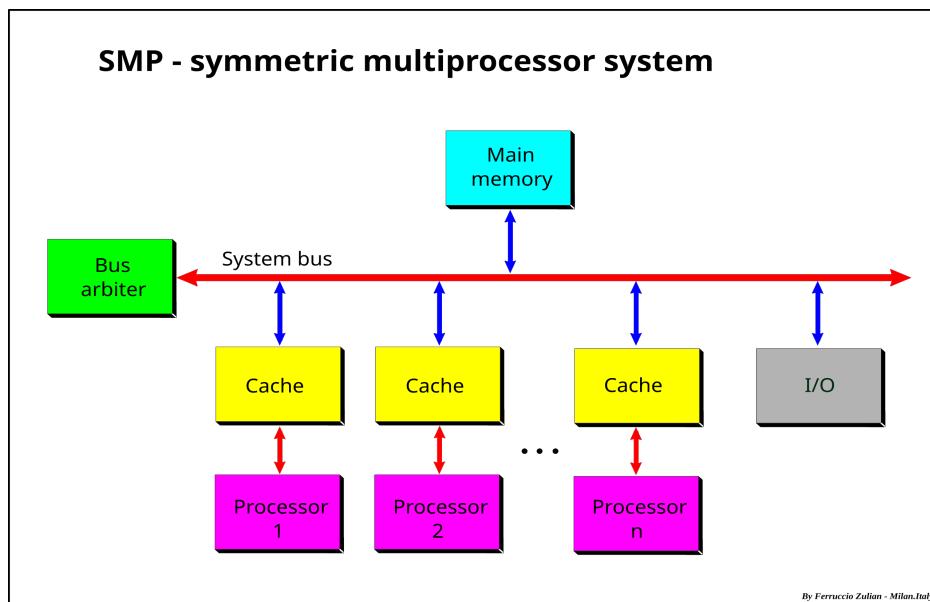


Imagen 7

3.2. Heterogéneo

Se refiere a un enfoque de computación que utiliza diferentes tipos de procesadores o arquitecturas para llevar a cabo tareas específicas. Este método permite aprovechar las fortalezas de cada tipo de procesador, como CPUs, GPUs y FPGAs, optimizando así el rendimiento y la eficiencia energética. En el contexto de sistemas distribuidos y procesamiento de datos, el procesamiento heterogéneo se vuelve crucial, ya que permite la ejecución de cargas de trabajo diversas en entornos que requieren escalabilidad y flexibilidad.

3.3. Cluster

Un clúster es un tipo de arquitectura paralela distribuida que consiste de un conjunto de computadores independientes interconectados operando de forma conjunta como único recurso computacional sin embargo, cada computador puede utilizarse de forma independiente o separada.

En esta arquitectura, el computador paralelo es esencialmente una colección de procesadores secuenciales, cada uno con su propia memoria local, que pueden trabajar conjuntamente.

- Cada nodo tiene rápido acceso a su propia memoria y acceso a la memoria de otros nodos mediante una red de comunicaciones, habitualmente una red de comunicaciones de alta velocidad.
- Los datos son intercambiados entre los nodos como mensajes a través de la red.
- Una red de ordenadores, especialmente si disponen de una interconexión de alta velocidad, puede ser vista como un multicomputador de memoria distribuida y como tal ser utilizada para resolver problemas mediante computación paralela.

4. Tipos de paralelismo

4.1. A nivel de bit

4.2. A nivel de instrucción

4.3. De datos

4.4. De tareas

5. Conclusión.

6. Bibliografía.

1. Adictech. (2024, April 9). ¿Qué es la Computación Paralela?: Beneficios y Aplicaciones. Adictec - Adicción Por La Tecnología. <https://adictec.com/que-es-computacion-paralela/>
2. de, B. (2025). Curso - 4.1 Aspectos Básicos de la Computación Paralela. Google.com. <https://sites.google.com/itmexicali.edu.mx/arquitectura-de-computadoras/4-procesamiento-paralelo/4-1-aspectos-b%C3%A1sicos-de-la-computaci%C3%B3n-paralela>
3. Taxonomía de Flynn AcademiaLab. (2024). Academia-Lab.com. <https://academia-lab.com/enciclopedia/taxonomia-de-flynn/>
4. de, T. (2025). Curso - 4.2 Tipos de Computación Paralela. Google.com. <https://sites.google.com/itmexicali.edu.mx/arquitectura-de-computadoras/4-procesamiento-paralelo/4-2-tipos-de-computaci%C3%B3n-paralela?authuser=0>
5. de, C. (2005, February 13). uso de dos o más procesadores (CPU) en una computadora para la ejecución de uno o varios procesos. Wikipedia.org; Wikimedia Foundation, Inc. <https://es.wikipedia.org/wiki/Multiprocesamiento>
6. Multiprocesamiento simétrico AcademiaLab. (2024). Academia-Lab.com. <https://academia-lab.com/enciclopedia/multiprocesamiento-simetrico/>
7. Procesamiento Heterogéneo. (2025). Glosarix. <https://glosarix.com/glossary/procesamiento-heterogeneo/>