

Introducción a la Programación Paralela

J.A. Medina

Ciencias de la Computación

Universidad de Alcalá

¿Qué es la computación paralelo?

- Es una forma de cómputo en la que muchas instrucciones se ejecutan simultáneamente (**Monoprocesador**)
- Varios procesadores trabajando juntos para resolver una tarea común (**Multiprocesador**)
- Inicialmente:
 - procesamiento era en serie a través del paradigma de programación estructurada
 - Su objetivo era descomponerlo en series discretas de instrucciones que se ejecutan una detrás de otra.
- La Programación paralela:
 - Trata de resolver grandes problemas, que a menudo se pueden dividir en unos más pequeños que luego son resueltos simultáneamente (en paralelo)
 - Mayor velocidad de computo
 - nos vemos obligados a dividir un determinado conjunto de operaciones en partes independientes que se puede ejecutar de manera simultanea en un computador paralelo.
 - Rediseñar los algoritmos que se utilizan

¿Por qué realizar la computación paralela?

- **La computación paralela permite:**
 - Resolver problemas que no caben en una sola CPU
 - Resolver problemas que no pueden resolverse en un plazo razonable
- **Límites de la computación con una sola cpu**
 - Velocidad de la luz (30 cm/ns), actualmente estamos hablando de ciclos de reloj de pocos nanosegundos.
 - límite físico. más transistores en menos espacio. => mayor dificultad para disipar calor
Ley de Moore establece que el número de transistores se doblan cada dieciocho meses
 - los procesadores actuales ejecutan instrucciones a un ritmo de 3.6 Ghz (velocidades mayores operaciones aritméticas)
Si crece la frecuencia de reloj también lo hace el consumo

Paralelismo en monoprocesadores

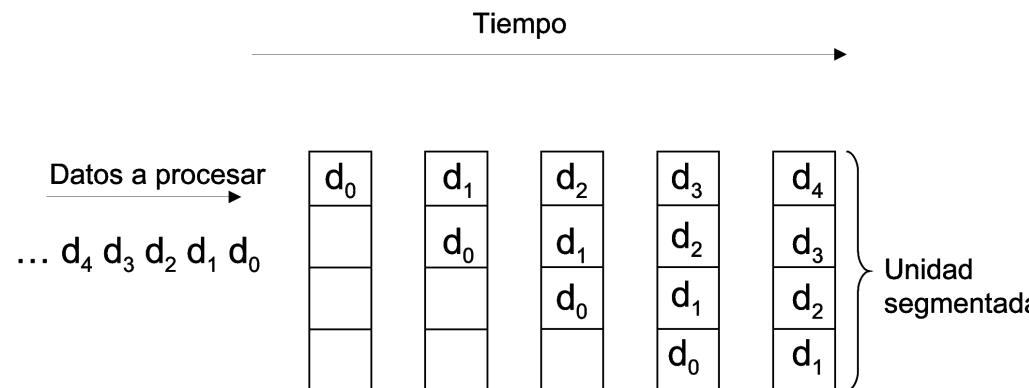
Paralelismo en monoprocesadores

- El tiempo de ejecución es el producto de tres factores:
 - El número de instrucciones del programa (N)
 - El número de ciclos de reloj por instrucción (C)
 - El número de segundos del ciclo de reloj (tc)
- Con el fin de mejorar el tiempo de ejecución se ha recurrido a:
 - Múltiples unidades funcionales
 - Solapamiento entre operaciones de la cpu y entrada/salida (I/O)
 - Uso de los recursos a tiempo compartido
 - Incremento de la longitud de palabra

Formas básicas de paralelismo

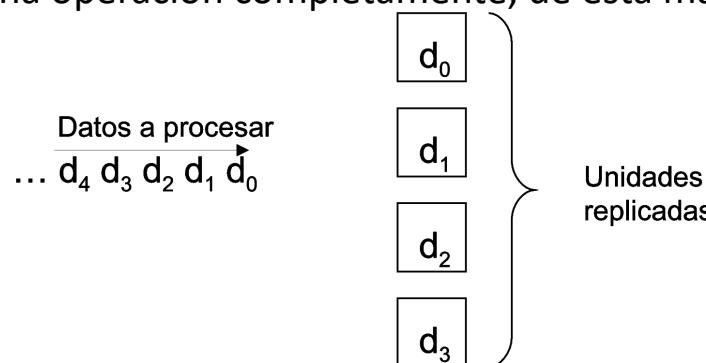
- Segmentación: También conocida como **pipe-line**

- Dividir una unidad funcional en etapas independientes, **intercalando registros** para el almacenamiento de los resultados intermedios
- Los datos pasan de una etapa a otra por los tiempos de reloj



- Replica. Permite aumentar el número de operaciones por unidad de tiempo

- Varias unidades ejecutan una operación completamente, de esta manera se ejecuta tantas instrucciones como unidades funcionales



Procesador Vectorial

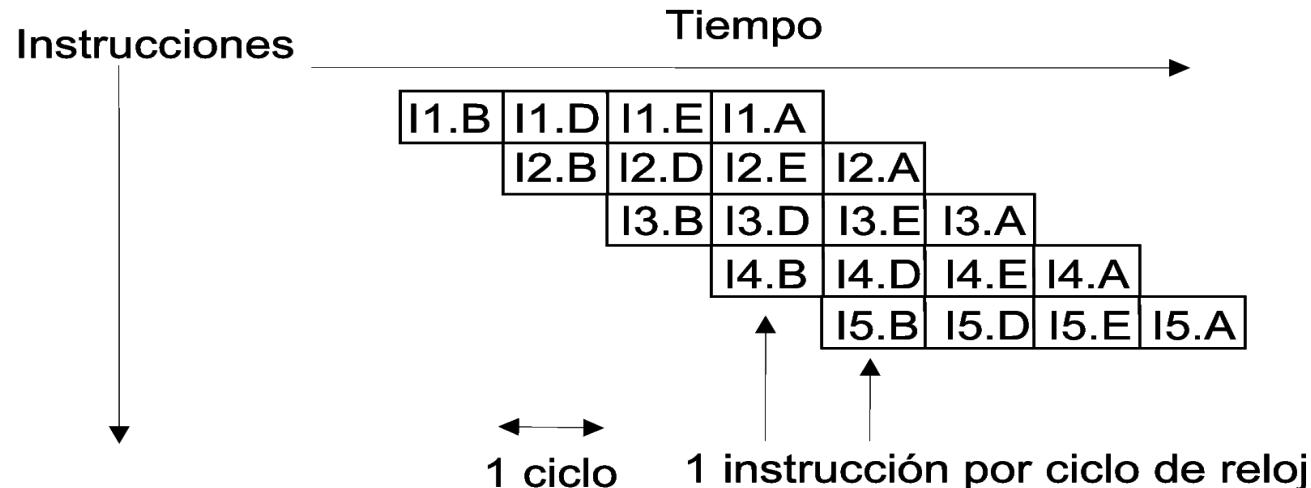
- Trabajan sobre conjuntos de datos homogéneos
- Se define como el procesador que dispone de instrucciones vectoriales en el juego de instrucciones de código maquina
- La implementación real de este tipo de instrucciones podría hacerse mediante replica pero, es inusual, actualmente están implementadas utilizando técnicas de segmentación
- En este tipo de maquinas es importante la organización de la memoria en módulos
- El número de ciclos por instrucción disminuye sustancialmente cuando se usan instrucciones vectoriales
- Para un rendimiento optimo de la unidad vectorial debe suministrar datos a una velocidad suficiente para tenerla llena.
- El uso de pipelining ha proporcionado las maquinas más potentes en la década de los 80.

Procesador Escalar

- El numero de ciclos por instrucción disminuye sustancialmente cuando se usan instrucciones vectoriales
- La idea básica es explotar el paralelismo entre instrucciones de forma transparentes al usuario utilizando las técnicas de **replica y/o segmentación**
- Orden lógico de las instrucciones debe respectarse aunque internamente se modifique, especialmente R/W.
- Normalmente una instrucción se puede indicar en cuatro etapas:
 - Búsqueda de la instrucción
 - Decodificación y lectura de operandos
 - Ejecución
 - Almacenamiento de resultado
- La duración de las subetapas pueden ser asociada con un cierto submúltiplo del ciclo de reloj

Procesador Segmentado

- Se utiliza la segmentación no sobre la unidad funcional sino sobre todo el proceso de ejecución de la instrucción.
- Cuatro etapas básicas(búsqueda, decodificación, ejecución, almacenamiento).
- Tras etapa inicial de llenado se consigue ejecutar una instrucción por ciclo de reloj



In.B=Búsqueda de la instrucción n

In.D=Decodificación de la instrucción n

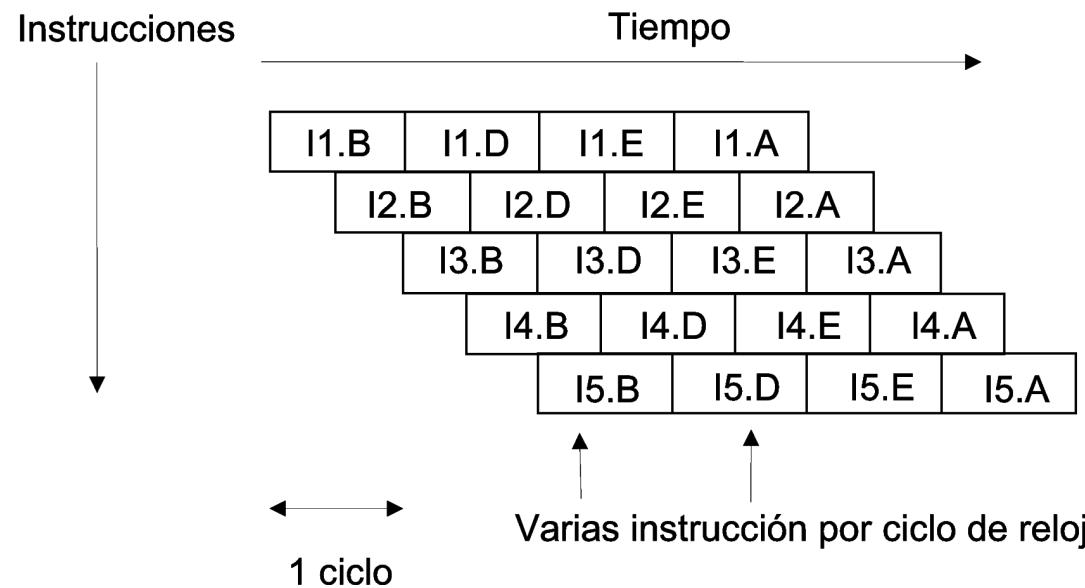
In.E=Ejecución de la instrucción n

In.A=Almacenamiento de resultados de la instrucción n

Procesador Supersegmentado

Cada etapa se divide en subetapas y se lanzan subetapas sin completar el ciclo de reloj

La duración de la subetapa puede ser asociada con un cierto ciclo de reloj (1/2 o 1/4)



$I_n.B$ =Búsqueda de la instrucción n

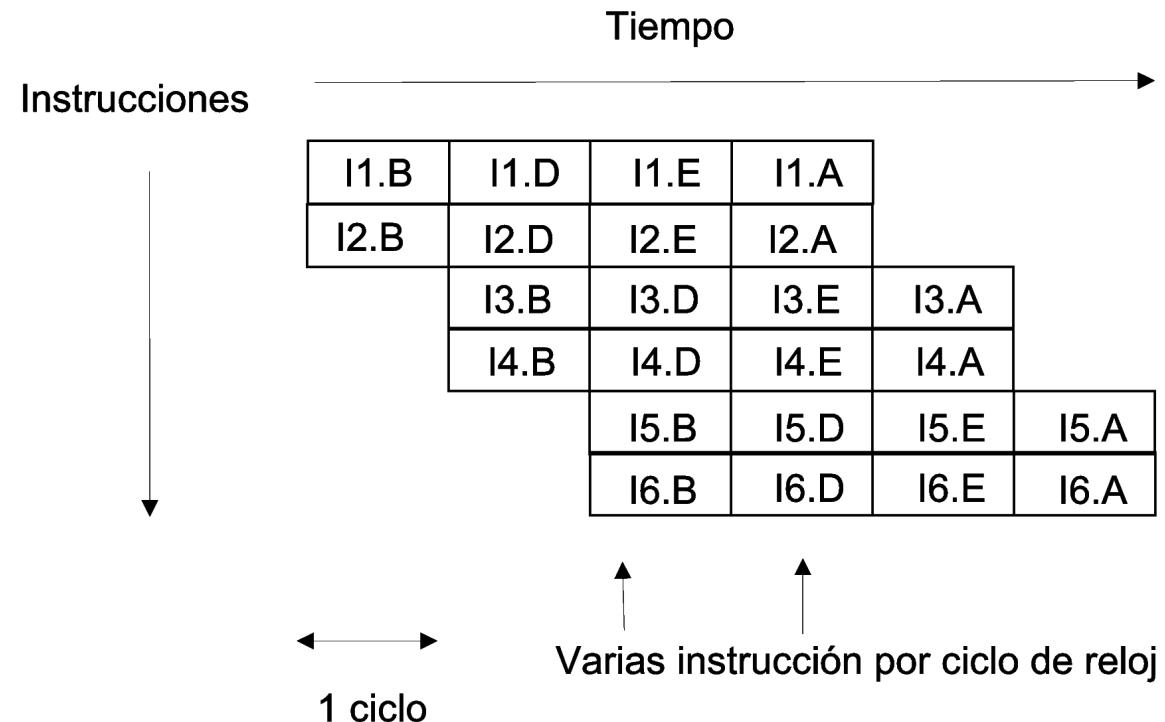
$I_n.D$ =Decodificación de la instrucción n

$I_n.E$ =Ejecución de la instrucción n

$I_n.A$ =Almacenamiento de resultados de la instrucción n

Procesador superescalar

Lanzar varias instrucciones de forma simultaneas
(ejecución fuera de orden => especulación)



$I_n.B$ =Búsqueda de la instrucción n

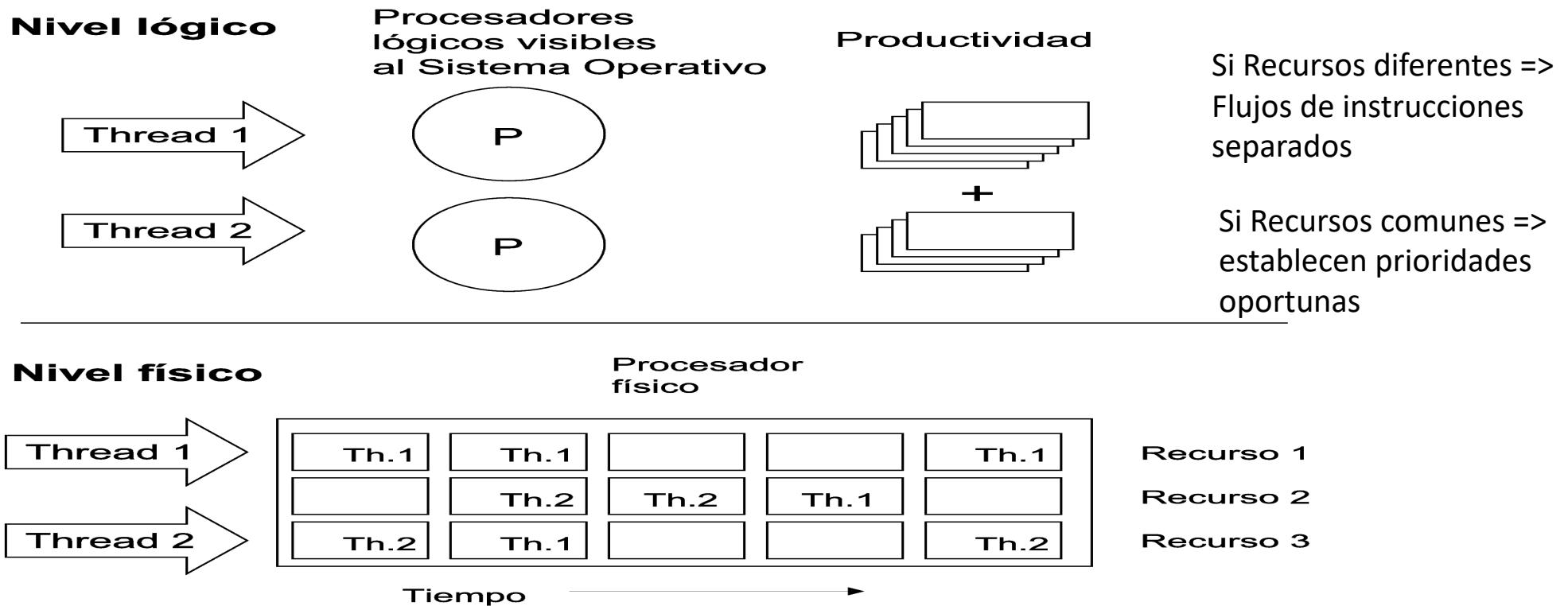
$I_n.D$ =Decodificación de la instrucción n

$I_n.E$ =Ejecución de la instrucción n

$I_n.A$ =Almacenamiento de resultados de la instrucción n

Multithreading

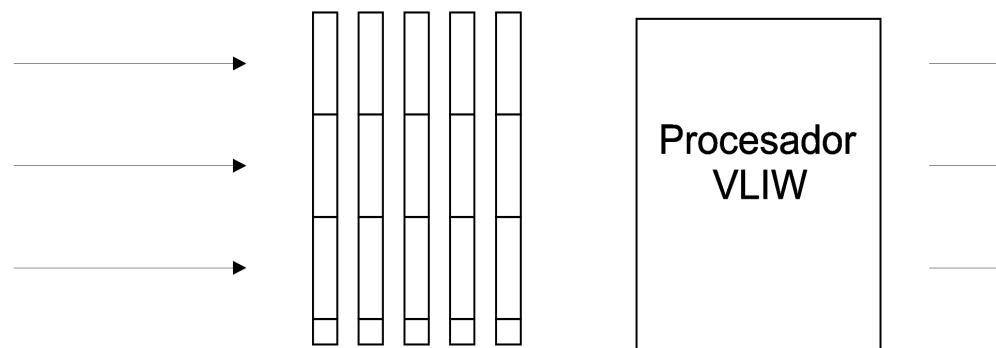
- Los procesadores actuales permiten la ejecución de varios procesos ligeros simultáneamente compartiendo el procesador y recursos.
- El software permite dividir su carga de trabajo en threads



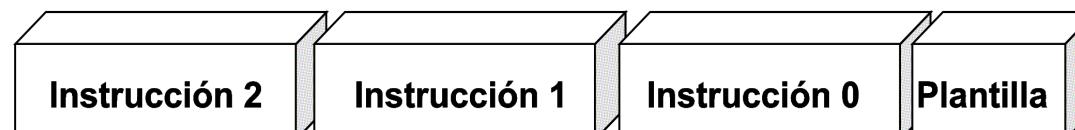
Procesadores VLIW (Very long Instruction Word)

- Palabras con Instrucciones mas largas de lo habitual
- Incorporar varias operaciones en cada instrucción
- Si se añade una multiplicidad de unidades funcionales permite que se ejecuten varias instrucciones => mayor velocidad

Procesamiento de hasta tres instrucciones en paralelo



Estructura de una instrucción de 128 bits



Parallelismo en los computadores multiprocesador

Paralelismo en los computadores multiprocesador

- Uso de varios procesadores trabajando juntos para resolver una tarea común:
 - Cada procesador trabaja en una porción del problema
 - Los procesos pueden intercambiar datos, a través de:
 - la memoria (**Modelo de Memoria Compartida**)
 - por una red de interconexión (**Modelo de Paso de Mensajes**)
- Paralelismo en dos niveles
 - Replica de los elementos de proceso mediante arquitectura multiprocesador
 - También es posible incorporar el paralelismo por segmentación

Clases de Clasificación – Computadores paralelos

Parámetros en los computadores paralelos

- Tipo y número de procesadores
 - Procesador de grano fino (operaciones elementales)
 - Procesador de grano grueso (operaciones complejas)
- Presencia y ausencia de un mecanismo global de control
- Funcionamiento síncrono y asíncrono
- Formas de comunicación de los procesadores
 - Memoria común
 - Mensajes

Parámetros en los computadores paralelos

El paralelismo consigue un aumento de las prestaciones

- **Paralelismo interno:** queda oculto a la arquitectura del computador, aumentando su velocidad sin modificar su funcionamiento.
Segmentación (pipe-line) de funciones.
- **Paralelismo explícito** es aquel que queda visible al usuario.

Computadores paralelos

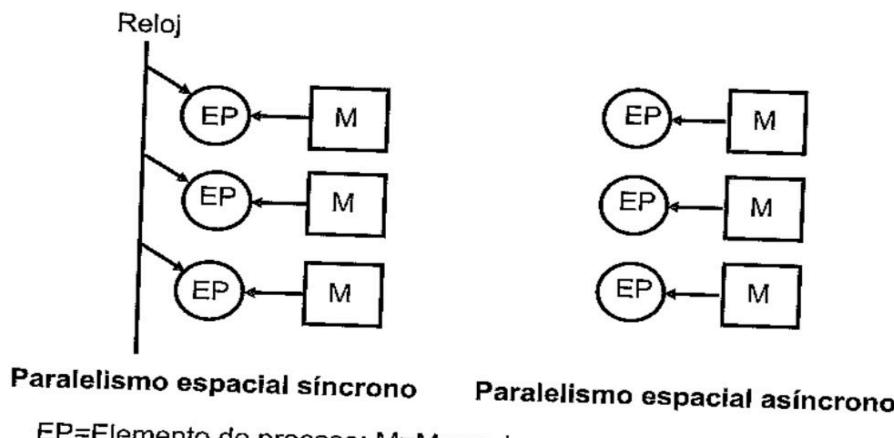
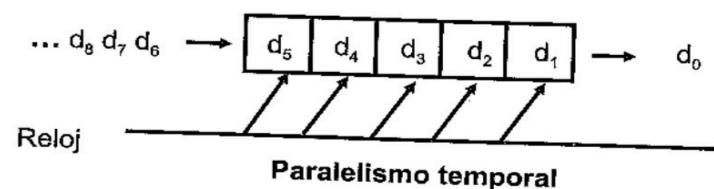
Clasificación de Hwang-Briggs

- Establecen una primera aproximación a las clases de computadores paralelos fijando Tres configuraciones básicas:
 - Computadores pipeline
 - Computadores matriciales
 - Sistemas multiprocesador

Computadores paralelos

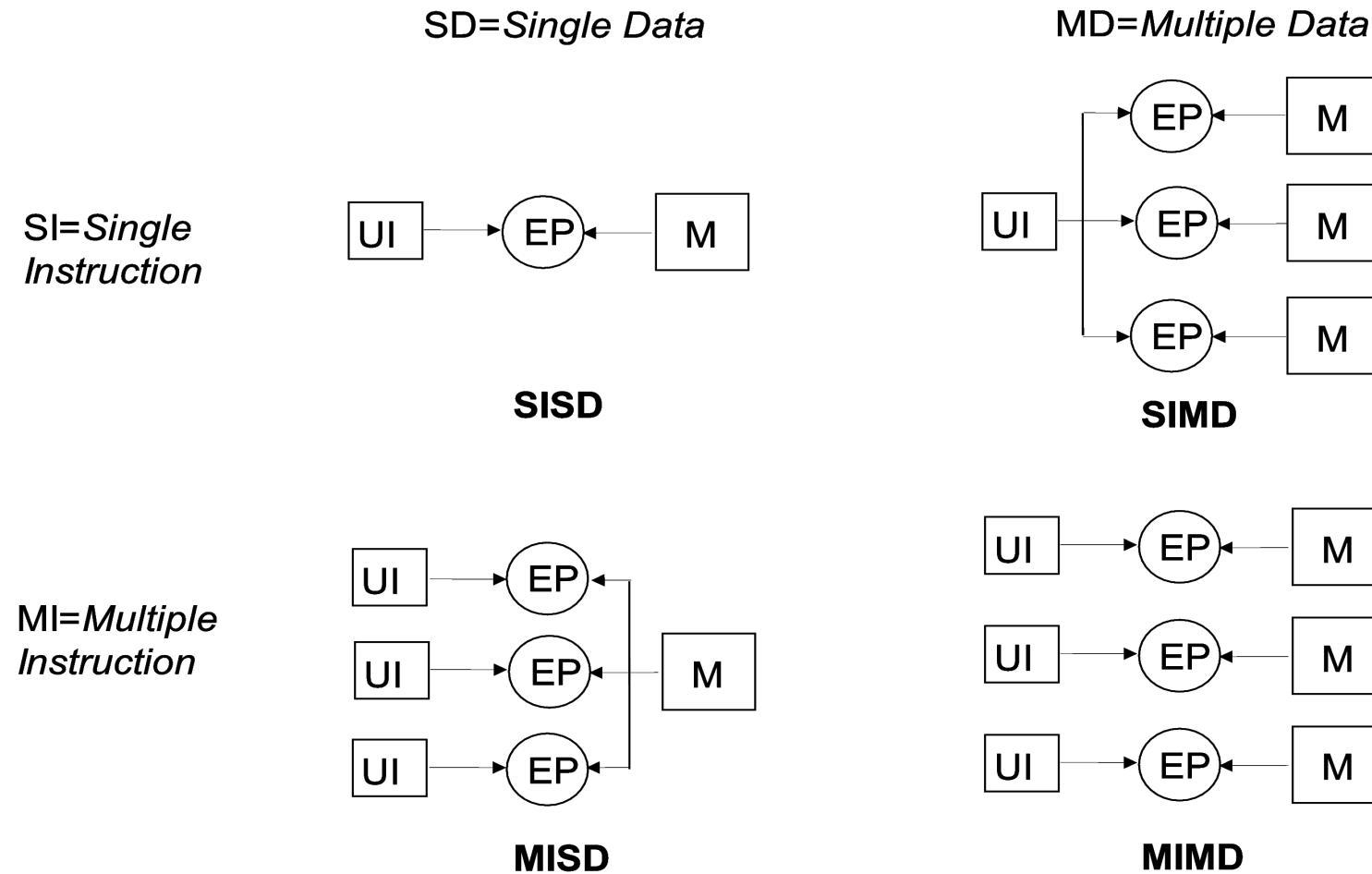
Clasificación de Hwang-Briggs

- De esta forma se puede distinguir:
 - computadores basados en parallelismo temporal (Segmentación)
Son aquellos que se solapan varias operaciones en el mismo instante de tiempo pero misma unidad funcional
 - Parallelismo espacial (existencia de procesos replicados)
 - Síncrono
 - asíncrono



Computadores paralelos

Clasificación de Flynn



EP=Elemento de proceso; M=Memoria; UI=Unidad de instrucción

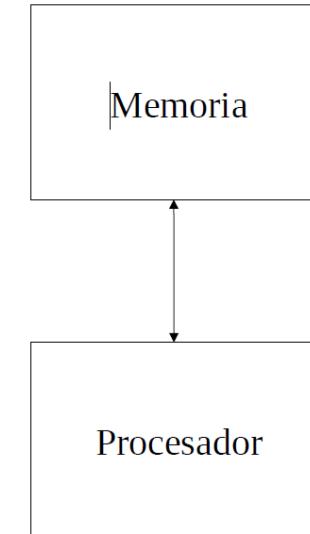
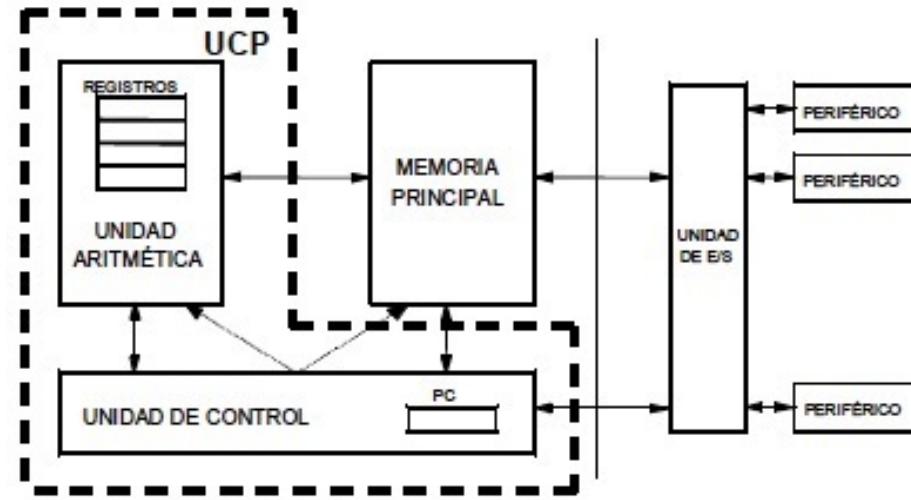
Simple Instrucción sobre Simple Dato (SISD)

SECUENCIAL (SISD)
Modelo Von Neuman

Instrucciones:
De memoria a procesador

Datos:
Entre memoria y procesador

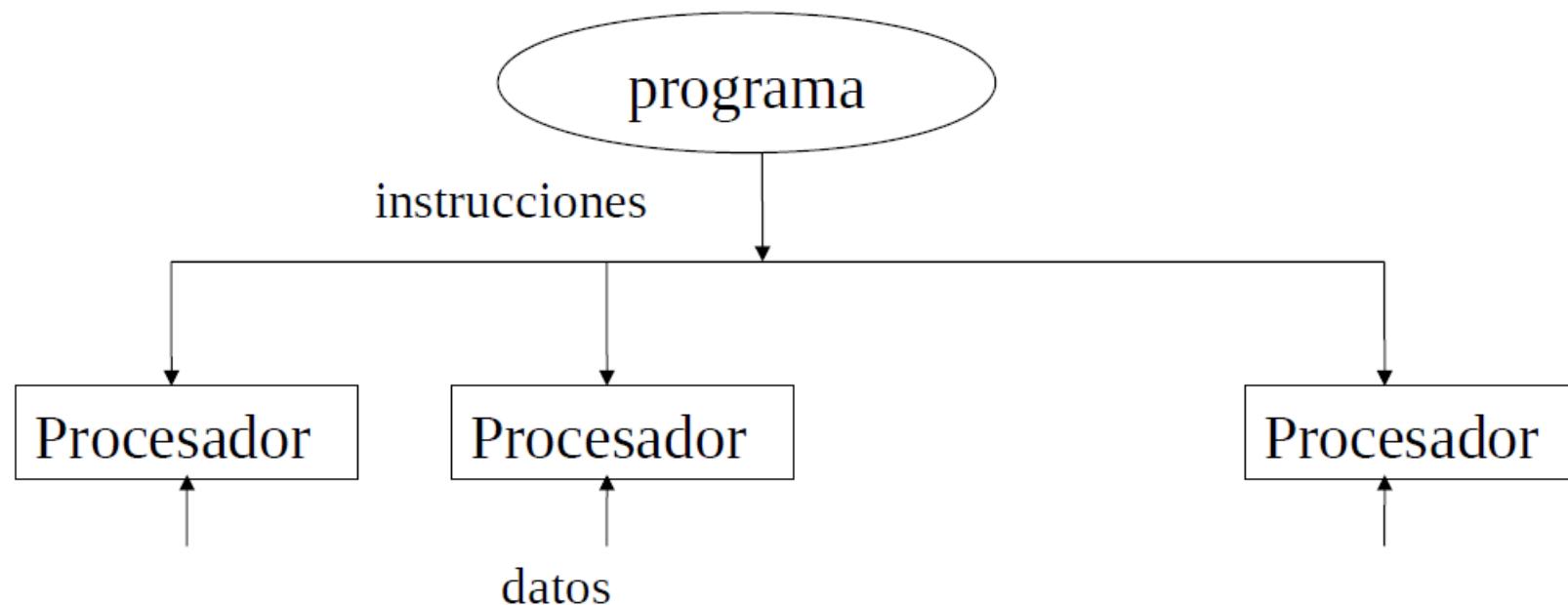
Ej: computador secuencial



Simple Instrucción sobre múltiple Dato (SIMD)

Una única Unidad de Control.

La misma instrucción se ejecuta síncronamente por todas las unidades de procesamiento



Ej: computador vectorial

Múltiple Instrucción sobre Simple Dato (MISD)

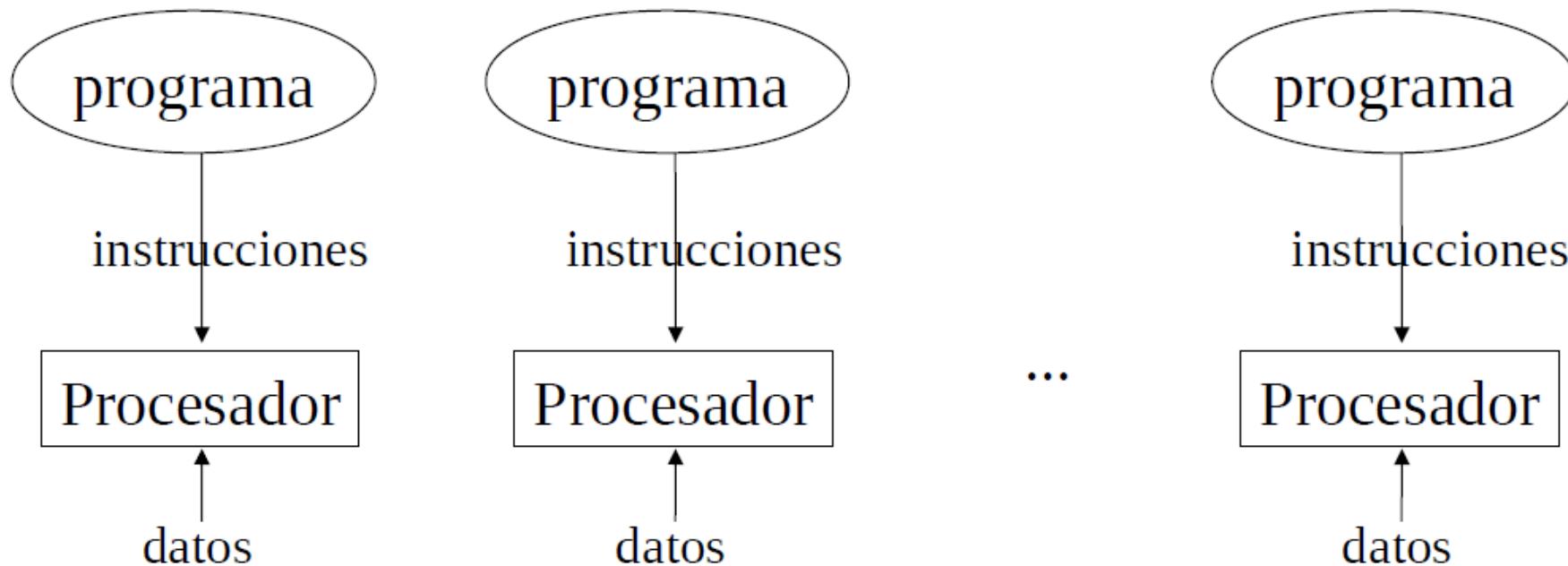
Diferentes flujos de instrucción se ejecuta sobre un mismo conjunto de datos

Muchos autores la consideran una categoría vacía

Difícil de encontrar maquinas reales

Múltiple Instrucción sobre múltiple Datos (MIMD)

Cada procesador ejecuta un programa diferente independientemente de otros procesadores



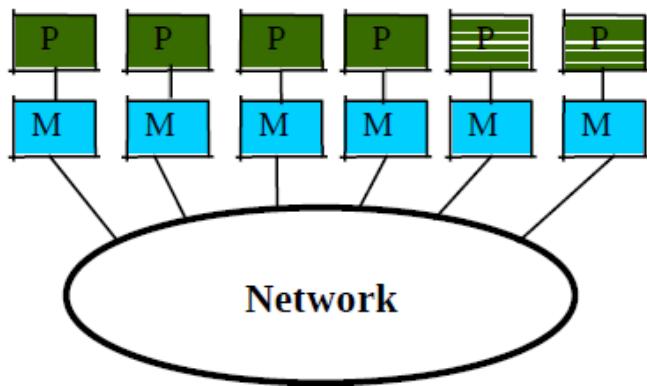
Ej: computador paralelo

Modelos de computadores paralelos

Modelos de computadores paralelos

Memoria distribuida

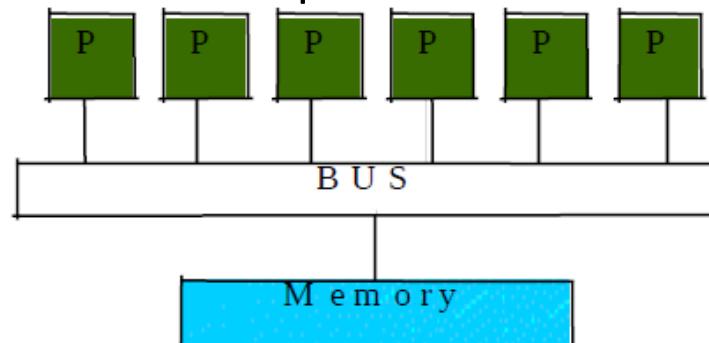
Cada procesador tiene su propia memoria principal local. Se utiliza paso de mensajes para intercambiar datos.



Memoria compartida

Un único espacio de memoria. Todos los procesadores tienen acceso a la memoria a través de una red de conexión:

- Bus
- Red de barras cruzadas
- Red multietapa



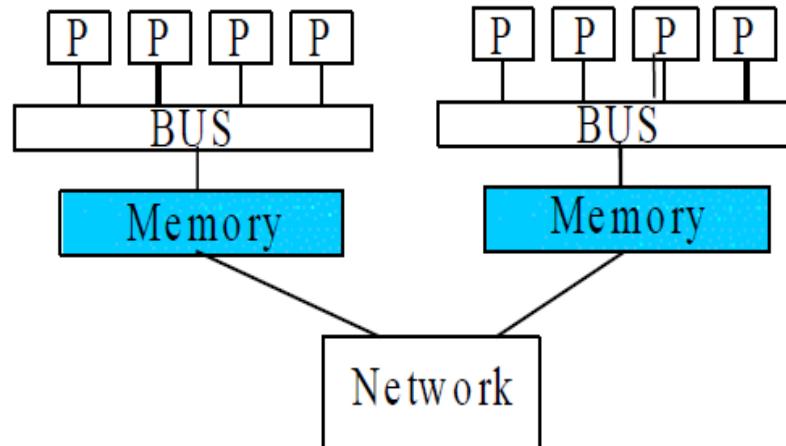
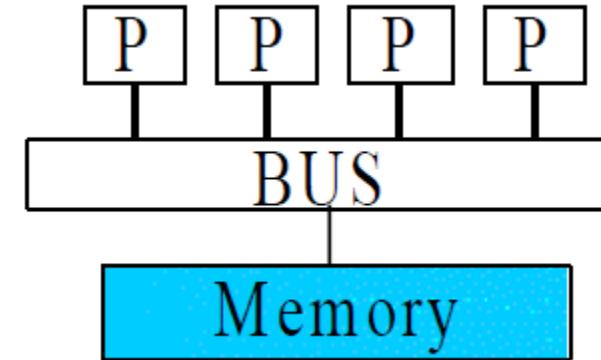
Sistemas de Memoria Compartida

Uniform memory access (UMA)

Cada procesador tiene acceso uniforme a memoria.

También se llaman symmetric multiprocessors (SMPs)

Memoria cache local en cada procesador



Non-uniform memory access (NUMA)

El tiempo de acceso depende de dónde están los datos.

El acceso local es más rápido.

Más fácil y barato de escalar que SMPs

Memoria compartida distribuida pero no global

Sistema Memoria distribuida – Topologías de red

- Cada sistema contiene su propio procesador y su propia memoria, comunicandose entre si.
- Comunicacion mediante paso de mensajes

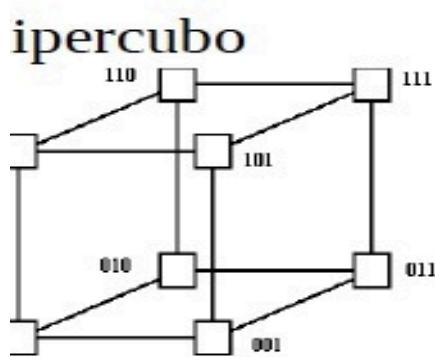
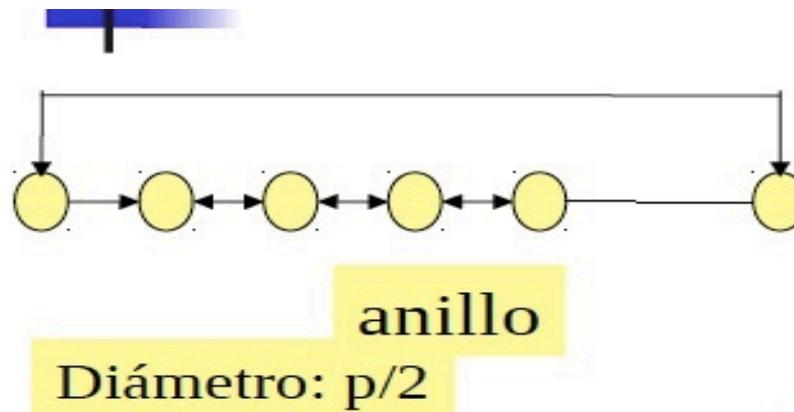


Figure 1.10 Three-dimensional hypercube

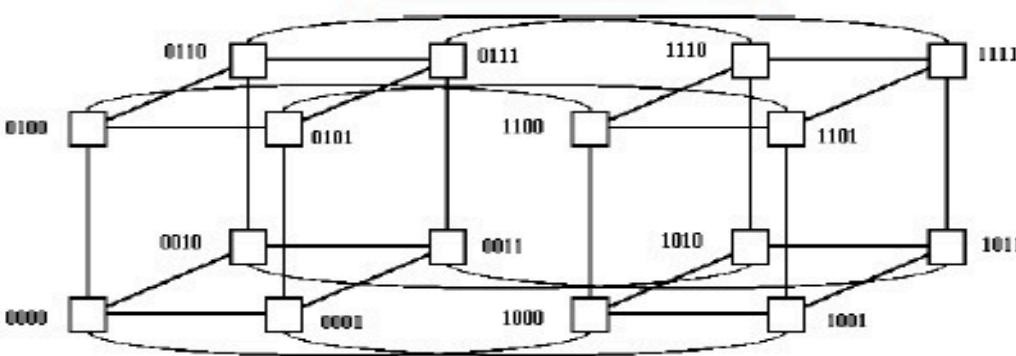


Figure 1.11 Four-dimensional hypercube

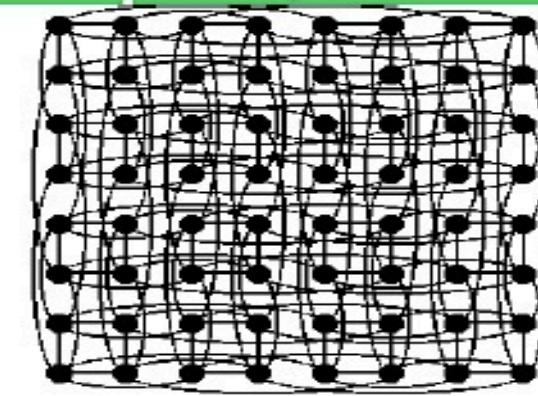
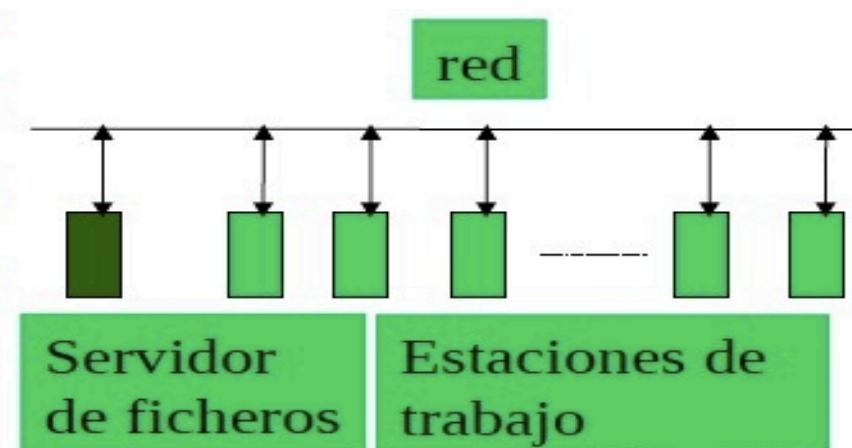
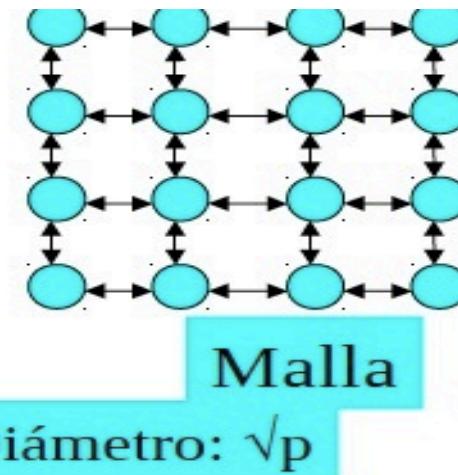
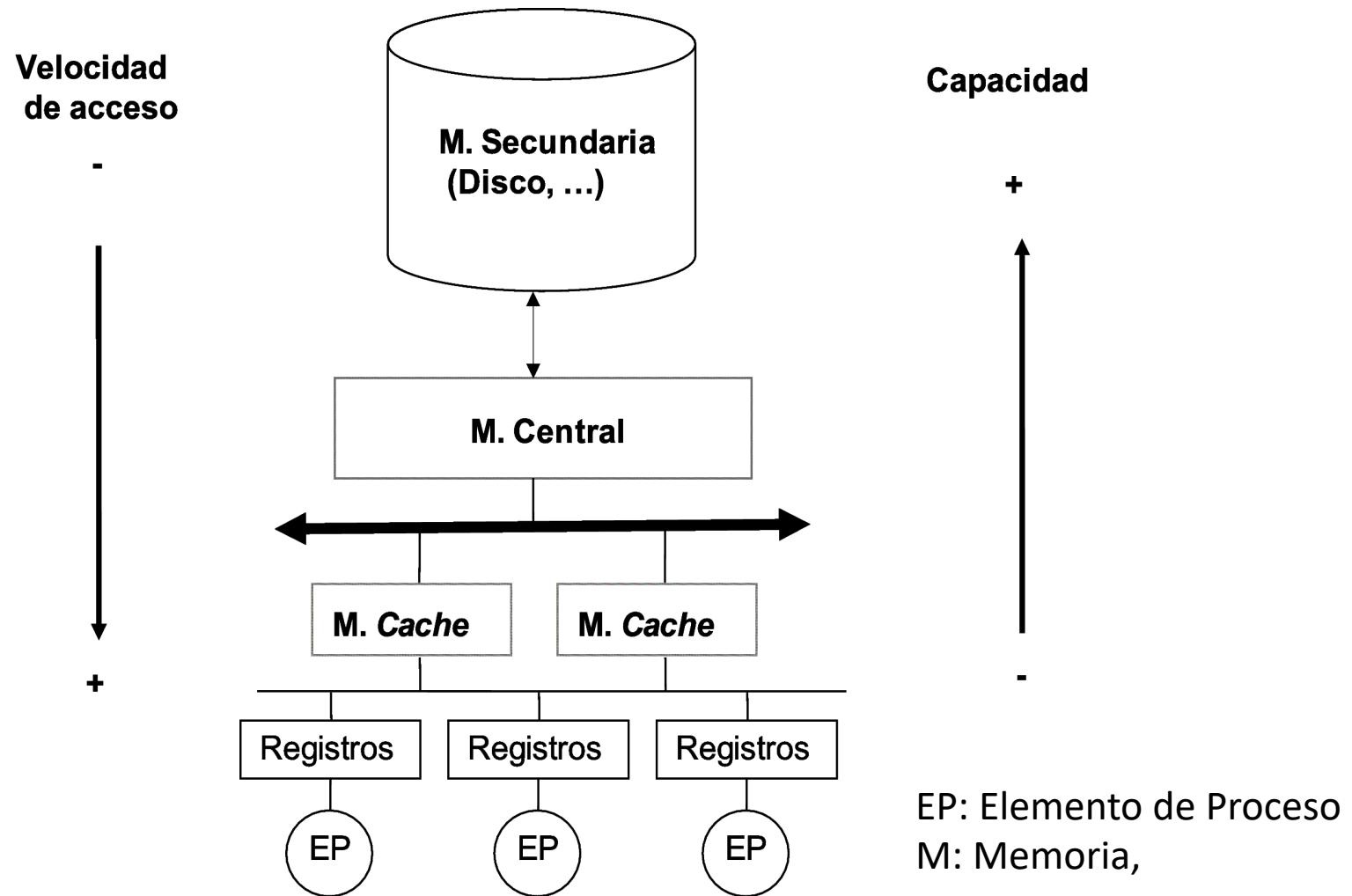


Figure 1.12 Six-dimensional hypercube laid out in one pi

Jerarquía de memorias



Sistemas Actuales y futuros

- Multicore
 - en portátiles: Bipro y Quad
 - en sobremesa también hexa, octa...
- Procesadores específicos
 - Gráficos GPU
 - De tratamiento de señal DSP
 - FPGA y heterogéneos embebidos
 - De juegos, PS4, xbox
- Computadores heterogéneos
 - CPU+GPU
 - Futuro: Plataformas híbridas Itanium2+Xeon con MC
- Distribuidos
 - Redes, Grid, Web, Cloud
 - P2P, móviles

https://www.cpu-monkey.com/es/cpu_family-intel_xeon_e3-13

https://www.cpu-monkey.com/es/cpu_family-intel_xeon_e7-27

Figuras tomadas directamente del libro

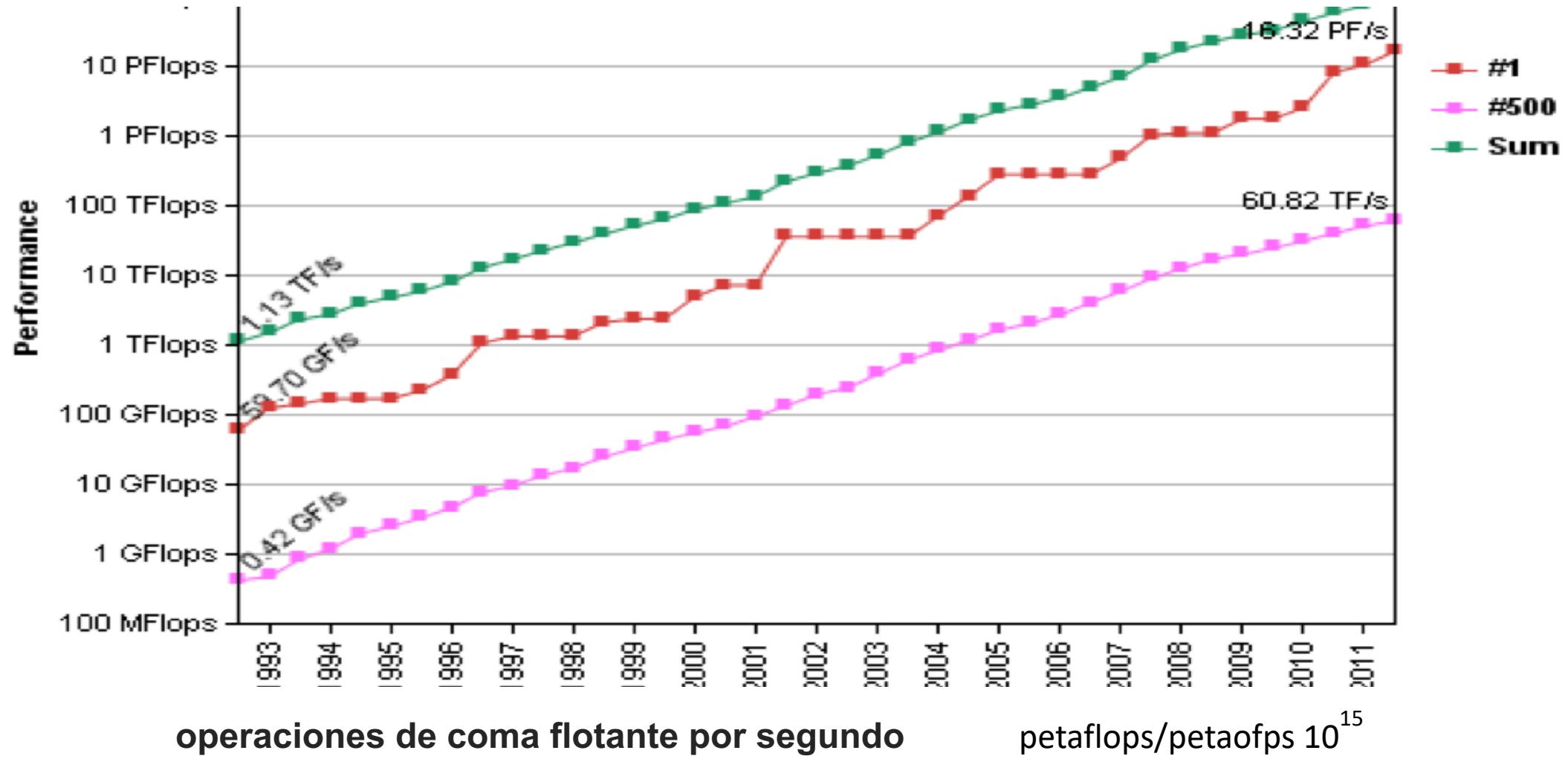
INTODUCCION DE LA PROGRAMACION PARALELA

Paraninfo Cengage Learning, 2008

Autores. Francisco Almeida, Domingo Giménez, José Miguel Mantas, Antonio M. Vidal
y de la conferencia de Casiano Rodríguez sobre OpenMP

Computación de Alto Rendimiento

SuperComputación



Evolución Histórica

Junio de 2008:

- Petaflop supercomputer (Peta = $10^{15} = 1000000000000000$).
- **Roadrunner (LANL)**, 1.026 petaflop/s.
 - BladeCenter QS22 Cluster.
 - PowerXCell 8i 3.2 Ghz / Opteron DC 1.8 GHz.
 - Híbrido: 6,562 dual-core AMD Opteron® y 12,240 Cell chips.
 - 98 terabytes de memoria.
 - 278 IBM BladeCenter® racks (560 m²).
 - 10,000 conexiones (Voltaire Infiniband y Gigabit Ethernet), 90 km de fibra óptica.
- Otros equipos del Top 5
 - IBM BlueGene/L (ANL), 478.2 teraflop/s.
 - IBM BlueGene/P (ANL), 450.3 teraflop/s.
 - **Ranger SunBlade x6420 (U. of Texas)**, 326 teraflop/s.
 - **Jaguar Cray XT4 (ORNL)**, 205 teraflop/s.

Evolución Histórica

Junio de 2010:

- **Jaguar (Oak Ridge National Laboratory, USA), 1.75 petaflop/s.**
 - Pico teórico: 2.7 petaflop/s.
 - Cray XT5-HE Cluster.
 - 37.376 AMD x86, 64 bits, Opteron Six Core 2.6 GHz.
 - 299 terabytes de memoria.
 - **224.162 núcleos.**
 - 10.000 TB de disco, red de 240 Gb/s.
- **Nebulae (National Supercomputing Centre, China), 1.27 petaflop/s.**
 - Pico teórico: 2.98 petaflop/s.
 - Dawning TC3600 Blade.
 - Híbrido: cuad-core Intel X5650 y 4.640 NVidia Tesla C2050 GPU .
 - **120.640 núcleos.**

Evolución Histórica

Junio de 2012:

- **Sequoia (DOE/NNSA/LLNL, EUA)**
- **Pico de desempeño real (LINPACK): 16.3 petaflops.**
 - Pico teórico: 20.1 petaflop/s.
 - IBM cluster.
 - 1572864 cores.
 - 1572 terabytes de memoria.
 - Red personalizada.
 - Sistema operativo basado en Linux.
 - Uno de los sistemas con mayor eficiencia energética.

TOP 10 DE SUPERCOMPUTADORES:

1º Sunway TaihuLight (National Supercomputing Center em Wuxi) – China (93.014,6 Teraflop/s)

2º Tianhe-2 (National Super Computer Center em Guangzhou) – China (33.862,7 Teraflop/s)

3º Titan (DOE/SC/Oak Ridge National Laboratory) – EUA (17.590 Teraflop/s)

4º Sequoia (DOE/NNSA/LLNL) – EUA (17.173,2 Teraflop/s)

5º K Computer (Riken Advanced Institute for Computational Science) – Japão (10.510 Teraflop/s)

6º Mira (DOE/SC/Argonne National Laboratory) – EUA (8.586,6 Teraflop/s)

7º Trinity (DOE/NNSA/LANL/SNL) – EUA (8.100,9 Teraflop/s)

8º Piz Daint (Swiss National Supercomputing Centre) – Suiça (6.271 Teraflop/s)

9º Hazel Hen (HLRS - Höchstleistungsrechenzentrum Stuttgart) – Alemanha (5.640,2 Teraflop/s)

10º Shaheen II (King Abdullah University os Science and Technology) – Arábia Saudita (5.537 Teraflop/s)

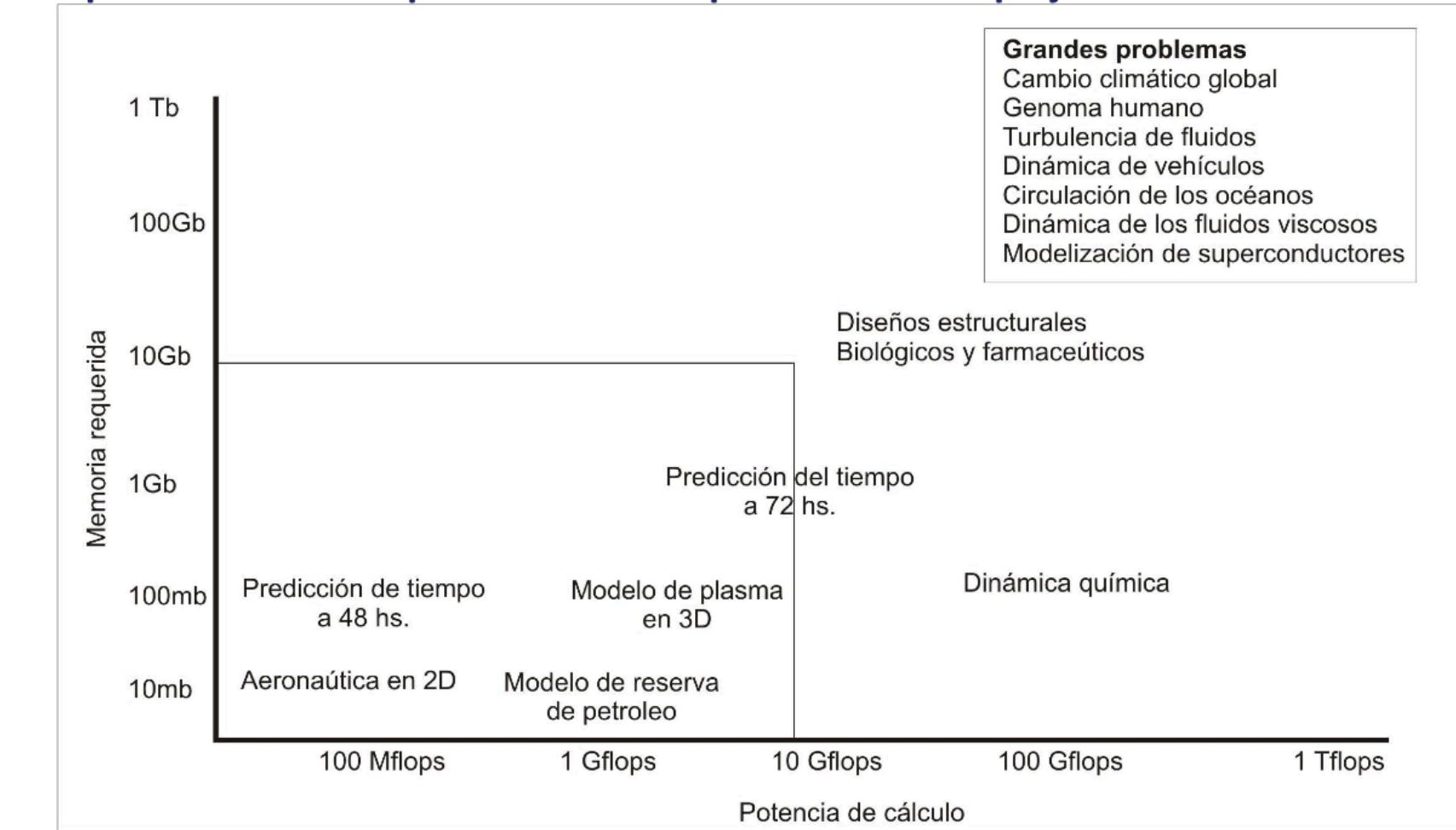
Evolución Histórica



<https://www.xataka.com/ordenadores/china-ya-tiene-202-de-los-500-supercomputadores-mas-potentes-del-mundo-mientras-ee-uu-pierde-liderazgo>

Evolución Histórica

- Requerimientos computacionales de problemas complejos.



Información tomadas directamente del libro
PROGRAMACIÓN MULTITHREADING
Autores. Sergio Nesmachnow y Gerardo Ares
ECAR 2012