

The background features abstract, overlapping green geometric shapes, primarily triangles and polygons, in various shades of green, creating a modern and dynamic visual effect.

Sistemas Distribuidos

Conceptos de HW

Desarrollo de aplicaciones en red

Febrero 2015

Introducción

- ▶ Los Sistemas Distribuidos constan de varios CPU's
 - ▶ Existen diversas formas de organizar el hardware
- ▶ Con el paso del tiempo se han propuesto diversas clasificaciones
- ▶ La taxonomía más citada es la de Flynn (1972)
 - ▶ Flynn seleccionó dos categorías fundamentales
 - ▶ Número de Instrucciones y número de flujo de datos

Multiprocesador

- ▶ Es cualquier computadora con varios procesadores

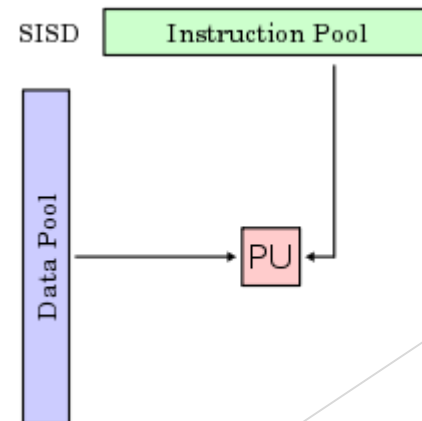
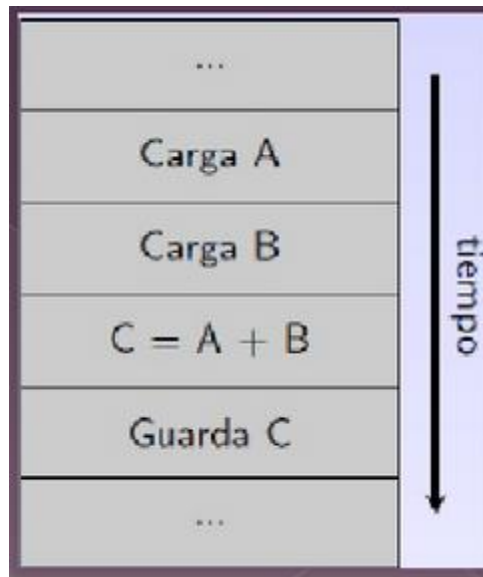


Lemieux cluster,
Pittsburgh
supercomputing
center

- ▶ Un procesador Multi-core es un tipo especial de multiprocesador:
 - ▶ Todos los procesadores están en el mismo chip

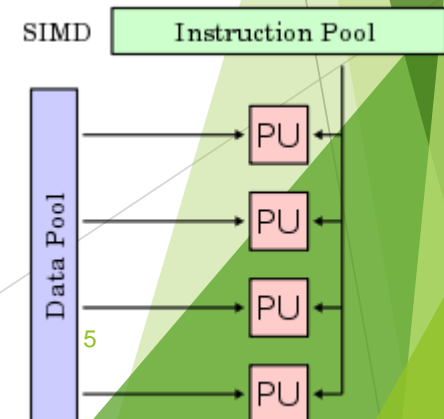
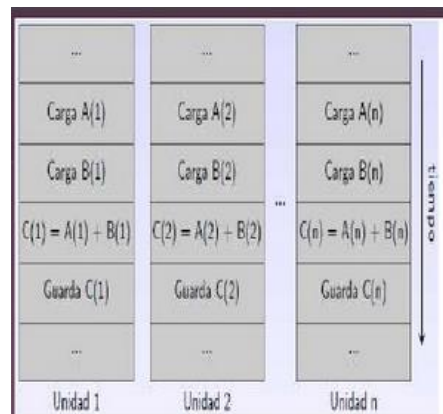
Clasificación Flynn

- ▶ SISD (Single Instruction, Single Data)
 - ▶ Un flujo de instrucción, un flujo de datos
 - ▶ Todas las computadoras de un procesador (un CPU) caen dentro de esta categoría (computadoras personales anteriores o mainframes)
 - ▶ Se ejecuta una instrucción por cada ciclo reloj



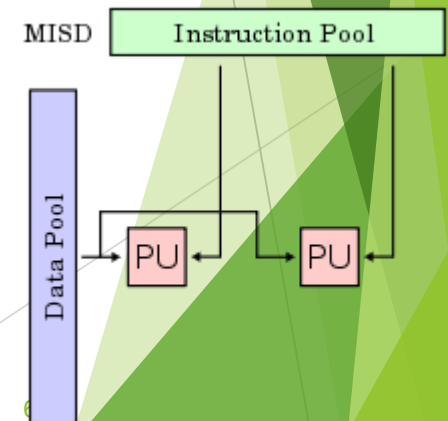
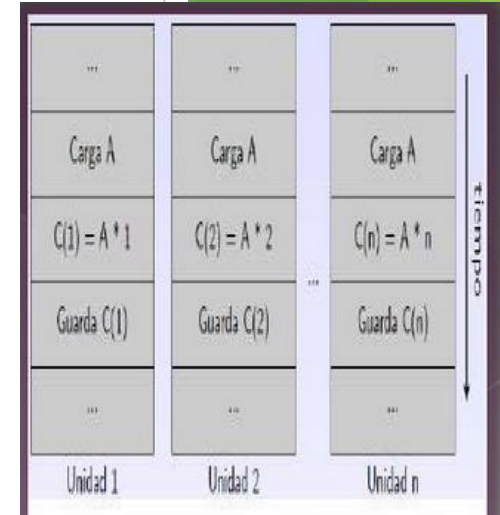
Clasificación Flynn

- ▶ SIMD (Single Instruction, Multiple Data)
 - ▶ Un flujo de instrucción y varios flujos de datos
 - ▶ Busca paralelismo a nivel de datos
 - ▶ Una única unidad de control despacha instrucciones a diferentes unidades de procesamiento
 - ▶ Instrucción que aplica una misma operación sobre un conjunto grande de datos
 - ▶ Las unidades reciben la misma instrucción, pero operan sobre diferentes conjuntos de datos. Ejecutándose de manera sincronizada por todas las unidades de procesamiento



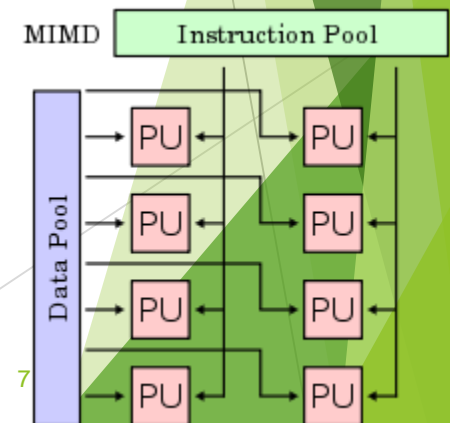
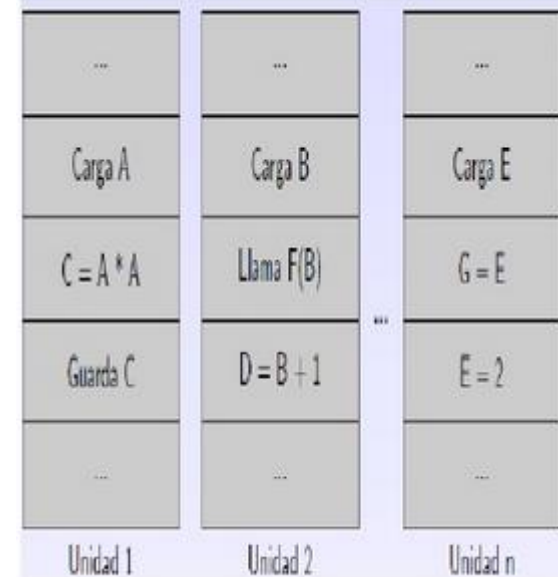
Clasificación Flynn

- ▶ MISD (Multiple Instruction, Single Data) ¿?
 - ▶ Un flujo de varias instrucciones y un flujo de datos
 - ▶ Arquitectura paralela
 - ▶ Muchas unidades de procesamiento realizan diferentes operaciones en los mismos datos
 - ▶ No existen muchos ejemplos
 - ▶ Facilitan escalamiento y uso de recursos
 - ▶ Paralelismo redundante
 - ▶ Aplicación limitada en el mundo real
 - ▶ Cada unidad ejecuta una distinta instrucción
 - ▶ Cada unidad procesa el mismo dato



Clasificación Flynn

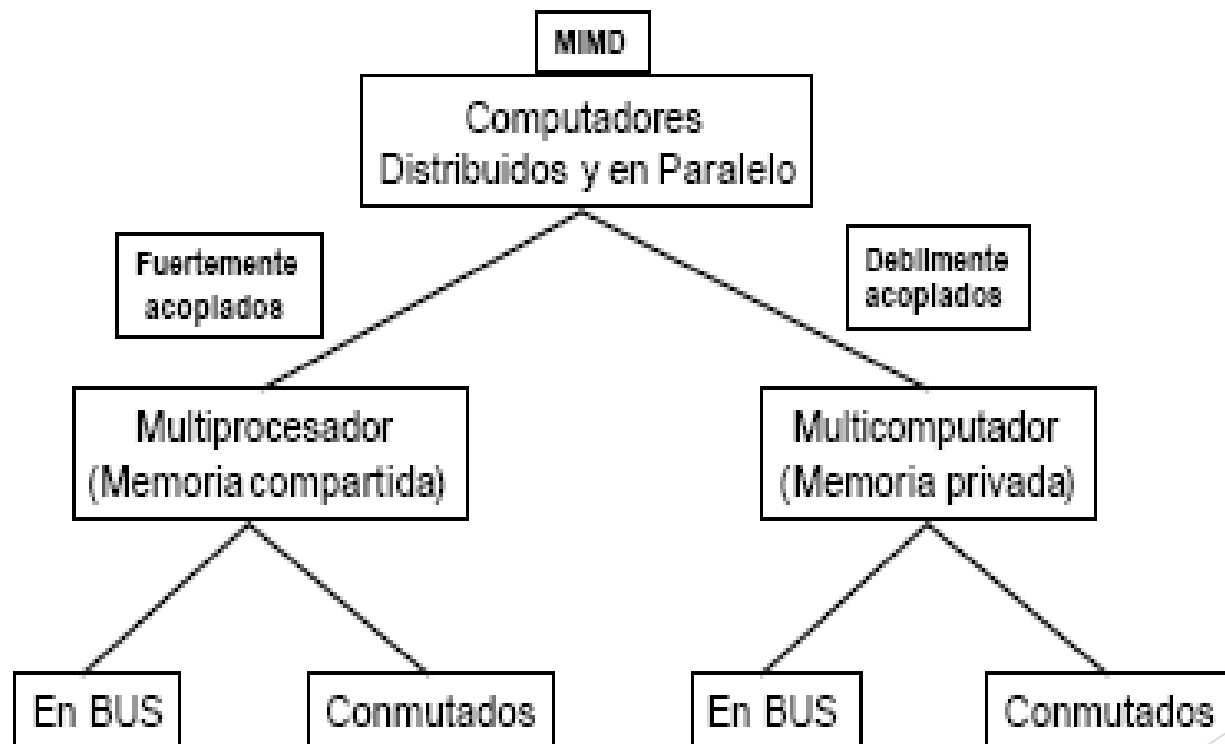
- ▶ MIMD (Multiple Instruction, Multiple Data)
 - ▶ Un súper conjunto de SISD
 - ▶ Pueden tener memoria compartida o distribuida
 - ▶ Cada unidad ejecuta una instrucción distinta
 - ▶ Cada unidad procesa un dato distinto
 - ▶ Casi todos los chip multi-core ejecutan diferentes hilos (MI), operando en diferentes partes de la memoria (MD)
 - ▶ Los multi-core tienen memoria compartida multiprocesador, todos los núcleos comparten la misma memoria
 - ▶ Empleado para lograr paralelismo
 - ▶ Los sistemas distribuidos entran en esta categoría
 - ▶ Ejemplos: Simulación, modelado, servidores web, aplicaciones científicas, multithreading



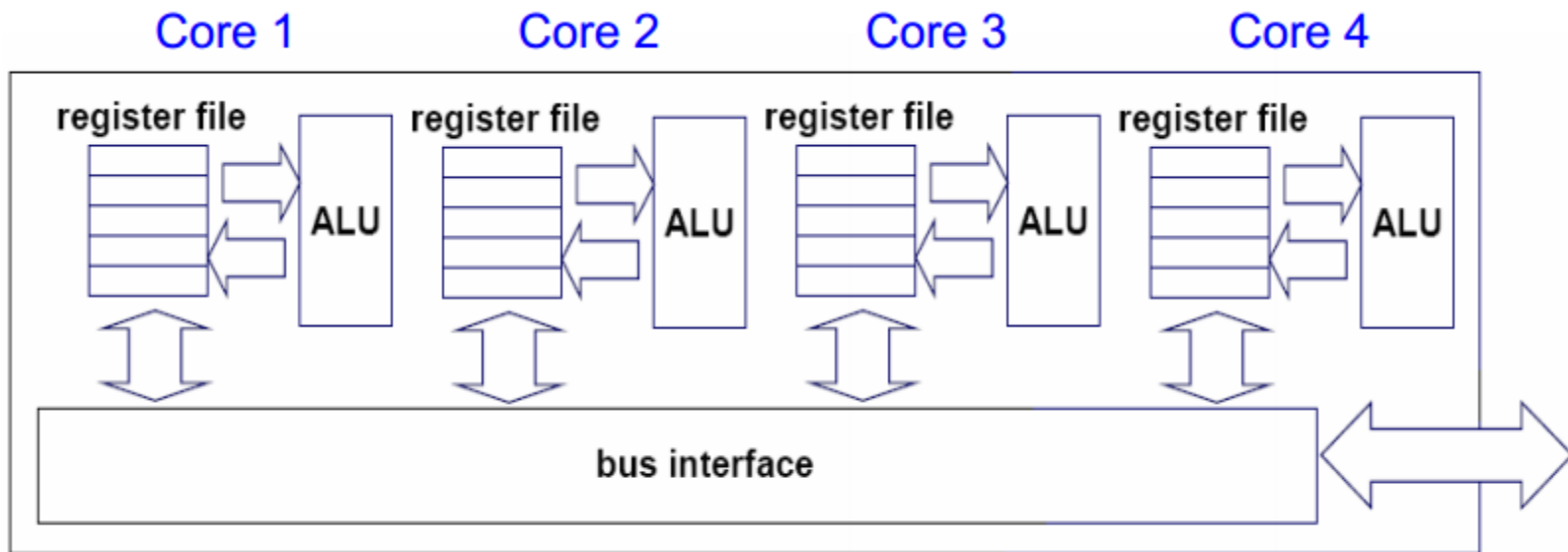
MIMD

- ▶ Divididos en dos grupos
 - ▶ Multiprocesadores
 - ▶ Existe espacio de direcciones virtuales compartido por todos los CPU's
 - ▶ Memoria Compartida
 - ▶ Sistemas paralelos (trabajar sobre un problema)
 - ▶ Fuertemente acoplados HW (el retraso al enviar información es corto)
 - ▶ Multicomputadoras
 - ▶ Cada computadora tiene su propia memoria
 - ▶ Colección de computadoras conectadas por la red
 - ▶ Paso de mensajes
 - ▶ Sistemas distribuidos (para trabajar varios problemas no necesariamente relacionados entre sí)
 - ▶ Débilmente acoplados HW, normalmente la información se envía a través de la red y puede tener un retraso considerable

- *Taxonomía de sistemas distribuidos y paralelos (Tanenbaum)*

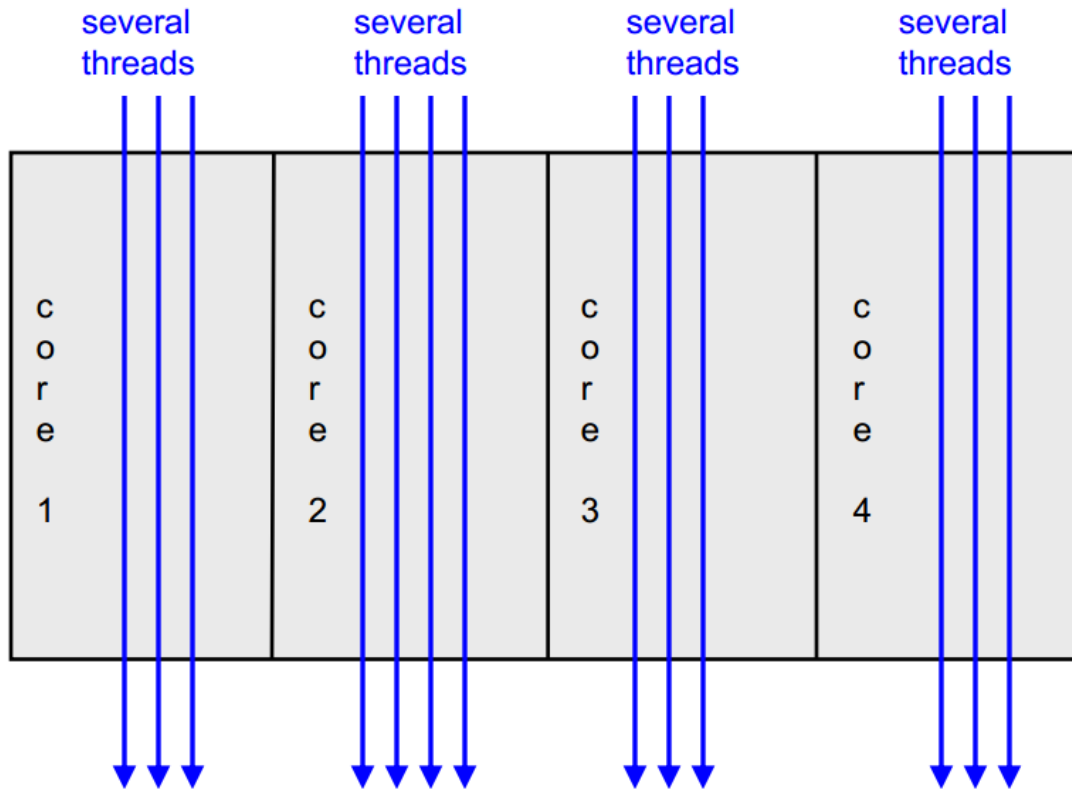


Multicore

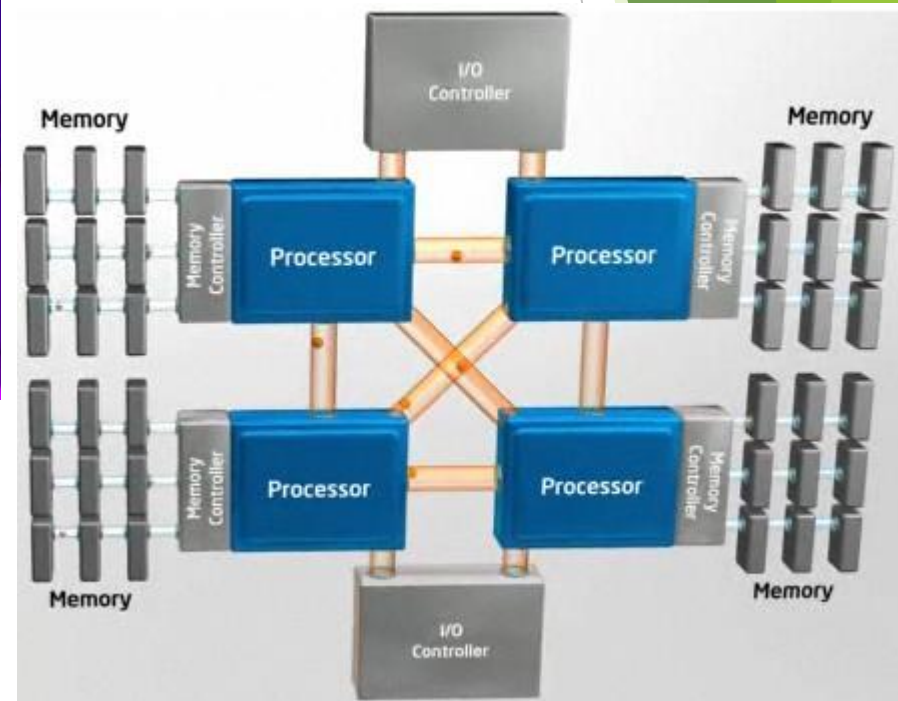
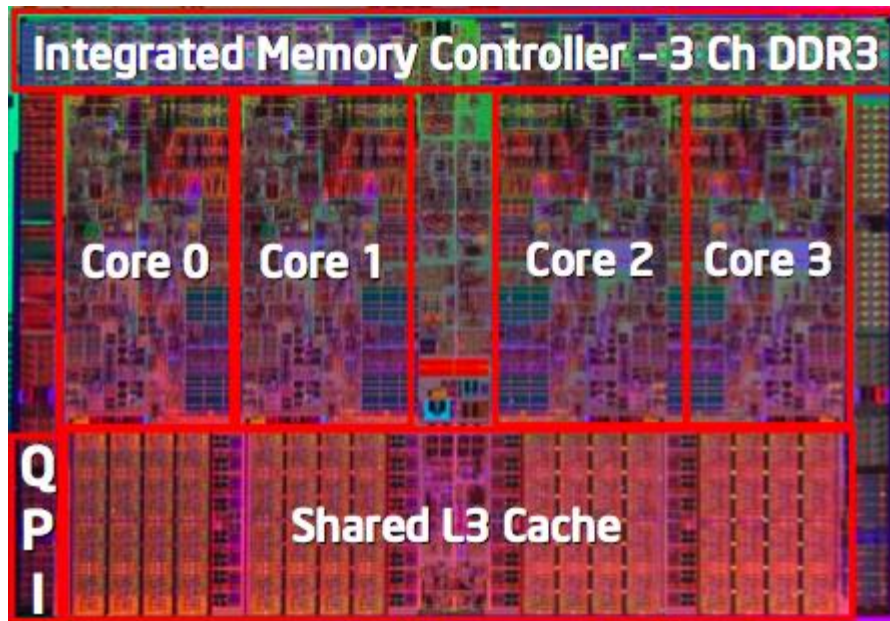


Multi-core CPU chip

Chip Multi-core CPU

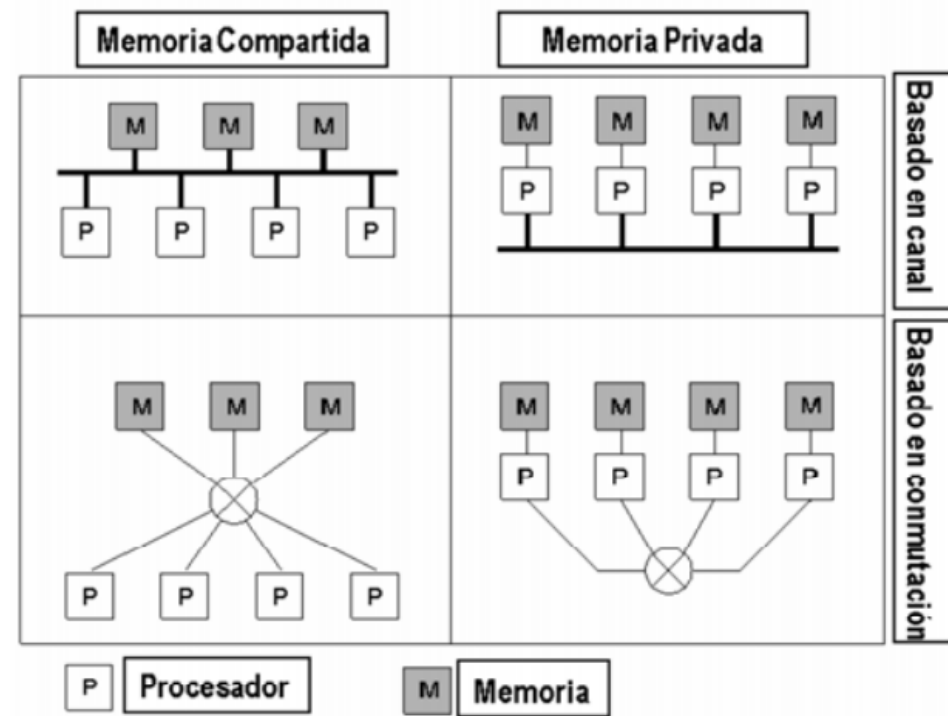


Arquitectura Nehalem (Intel Core)



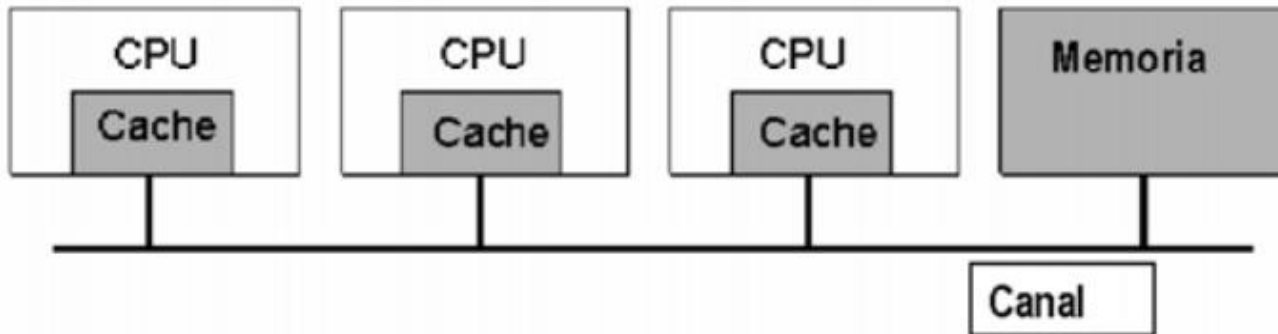
Organización Básica y Memoria

- ▶ Multi-procesadores (fuertemente acoplados)
 - ▶ Suelen ser sistemas paralelos
- ▶ Multi-computadoras (débilmente acoplados)
 - ▶ Suelen ser sistemas distribuidos



Multi-procesador Basado en Canal

- ▶ Con base en canales (buses)
 - ▶ Típicamente con 32 o 64 líneas de direcciones, 32,64 de datos y 32 o más líneas de control
- ▶ Conectados en un canal común
- ▶ Una configuración sencilla consta de un plano base de alta velocidad
- ▶ Se pueden insertar memoria y CPU's
- ▶ Ideal para 4,5 CPUs, si existe un número mayor el canal estará sobrecargado y el rendimiento disminuye (una solución es añadir memoria cache)
- ▶ Limitados para 64 CPU's



Multi-procesador con Conmutador

- ▶ Ideal para tener un multi-procesador con más de 64 procesadores, dividiendo la memoria en módulos y conectarlos a los CPU's con un conmutador de cruceta
- ▶ En cada intersección está un conmutador de punto de cruce (que puede abrir o cerrar)
- ▶ Si un CPU desea acceder a la memoria el conmutador cierra el cruce que los conecta para permitir el acceso
- ▶ CPU's pueden tener acceso a distintas unidades de memoria al mismo tiempo, si intentan acceder a la misma memoria es necesario que uno de ellos espere
- ▶ Pueden ser más caros y lentos

Multi-procesador con Conmutador

- ▶ Conmutador Cruceta
 - ▶ n CPU's y n Memorias
 - ▶ Una de las desventajas es que se necesitan n^2 conmutadores
- ▶ Red Omega
 - ▶ Diseñador para disminuir el número de conmutadores
 - ▶ Cada conmutador tiene dos entradas y dos salidas
 - ▶ Los conmutadores se activan dependiendo de los estados de las entradas
- ▶ Ejemplo con 4 CPUs y 4 Memorias
 - ▶ Etapas conmutación: $\log_2 n = \log_2 4 = 2$
 - ▶ No. conmutador/etapa: $n/2 = 4/2 = 2$
 - ▶ Total conmutadores: $(n \log_2 n)/2 = (2 \cdot 2)/2 = 2$

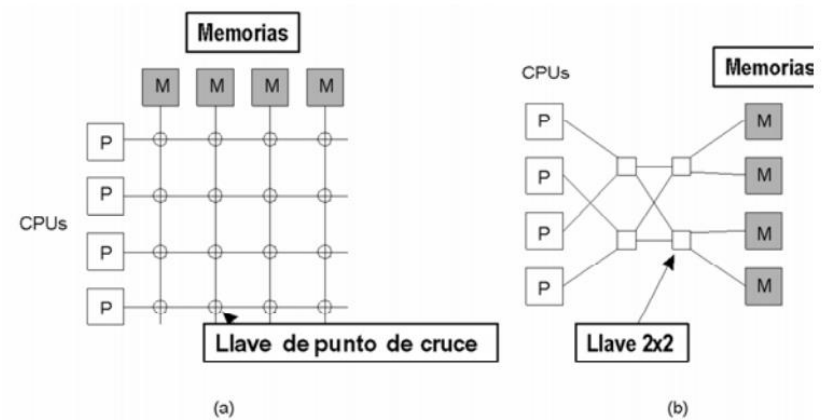


Figura 1.9: (a) Conmutador de cruce (b) Red omega

Multi-computadoras con Base en Canal

- ▶ Más sencilla que un multiprocesador con memoria compartida
- ▶ Cada memoria tiene su propia memoria local
- ▶ El problema principal es la comunicación entre los CPU's
- ▶ Topológicamente un multiprocesador basado en canal es similar a un multi-computador basado en canal, sin embargo, se genera menor tráfico
- ▶ La comunicación puede ser por una LAN

Multi-computadoras con Conmutador

- ▶ Se construyen varias redes de comunicación
- ▶ Cada CPU tiene acceso directo y exclusivo a su propia memoria
- ▶ Entre las topologías populares:
 - ▶ Retícula: fáciles de comprender, basadas en circuitos impresos
 - ▶ Hipercubo: se adecuan mejor a problemas con naturaleza bidimensional, cada conexión vértice es un CPU y cada arista sería una conexión entre 2 CPU's

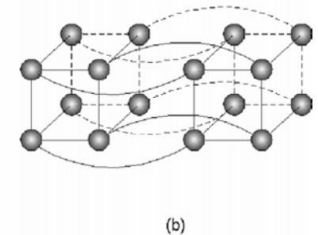
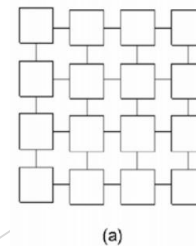


Figura 1.10: (a) Retícula (b) Hipercubo