CAPITULO 7

ARQUITECTURAS DISTRIBUIDAS

7.1 - MULTIPROCESADORES (MIMD)

Un multiprocesador se define como una computadora que contiene dos o más unidades de procesamiento que trabajan sobre una memoria común bajo un control integrado (recordemos su arquitectura según la clasificación de Flynn).

Si el sistema de multiprocesamiento posee procesadores de aproximadamente igual capacidad, estamos en presencia de multiprocesamiento simétrico, en el otro caso hablamos de multiprocesamiento asimétrico.

Todos los procesadores deben poder acceder y usar la memoria principal. De acuerdo a esta definición se requiere que la memoria principal sea común y solamente existen pequeñas memorias locales a cada procesador.

Si cada procesador posee una gran memoria local se lo puede considerar un sistema de multicomputadoras, el cual puede ser centralizado o distribuido.

Todos los procesadores comparten el acceso a canales de E/S, unidades de control y dispositivos.

Para el sistema de multiprocesamiento debe existir un sistema operativo integrado el cual controla el hardware y el software y debe asegurar la interacción entre los procesadores y sus programas a un nivel elemental de dato, conjunto de datos, tareas y trabajos.

Aún cuando la ejecución de los procesos en arquitecturas MIMD se sincroniza mediante el pasaje de mensajes por medio de una red de interconexión o accediendo a datos en unidades de memoria compartida, las arquitecturas MIMD son computadoras asincrónicas caracterizadas por un hardware de control descentralizado.

La efectividad del costo de n Sistemas Procesadores respecto de n procesadores aislados alienta la experimentación en las MIMD.

7.2 - SISTEMAS DISTRIBUIDOS

Hemos dicho ya que los sistemas MIMD débilmente acoplados reciben también el nombre de Sistemas Distribuidos. Pasamos ahora a tratar tales sistemas.

Existen cuatro grandes razones para construir Sistemas Distribuidos, a saber:

- el compartir los recursos,
- la velocidad del cómputo,
- la confiabilidad y
- la comunicación.

7.2.1 - **Compartir Recursos**

Si una cantidad de nodos (sitios, computadores) están conectados a otros, entonces un usuario en un nodo puede hacer uso de los recursos disponibles en el otro.

Por ejemplo, un usuario en el nodo A desea utilizar una impresora láser que se encuentra en el nodo B, mientras tanto el usuario en el nodo B quiere acceder a un archivo que está en el nodo A.

En general, en un sistema distribuido están provistos los mecanismos para compartir archivos en sitios remotos, o para procesar información en una base de datos distribuida, e imprimir archivos en sitios remotos, utilizar en forma remota dispositivos hardware especializados (como por ejemplo array processor), y otras operaciones.

7.2.2 - **Velocidad de Cómputo**

Si un cálculo particular puede dividirse en una cierta cantidad de subcómputos que puedan ejecutarse concurrentemente, entonces la disponibilidad de un sistema distribuido nos permite justamente distribuir el cálculo entre varios nodos.

Además si un nodo se encuentra sobrecargado de trabajos puede rutearlos hacia otro nodo menos cargado. Este movimiento de trabajos se denomina "compartir la carga".

7.2.3 - **Confiabilidad**

Si un nodo falla en un sistema distribuido, los sitios restantes pueden potencialmente continuar la operatoria.

Si el sistema está compuesto por una gran cantidad de instalaciones autónomas (por. ej. computadoras de propósito general) entonces la falla de una de ellas no afectaría el resto.

Si por otra parte el sistema está compuesto de máquinas pequeñas, cada una de las cuales es responsable de alguna función crucial del sistema, entonces una sola falla puede efectivamente detener la operatoria total del sistema.

En general, si existe la suficiente redundancia en el sistema (tanto de hardware como de software) luego el sistema puede continuar operando aún cuando algunos de sus nodos fallen.

7.2.4 - **Comunicación**

Cuando una cantidad de nodos están conectados a otros mediante una red de comunicación, los usuarios en diferentes nodos tienen la oportunidad de intercambiar información. En un sistema distribuido nos referiremos a esta actividad como "correo electrónico".

Cada usuario de la red tiene asociado una única dirección (en el sentido de domicilio -mailbox-) y puede intercambiar correspondencia con otro usuario en el mismo nodo o en nodos diferentes.

Este correo no es interpretado por el sistema operativo.

7.3 - **TOPOLOGIA DE LA RED**

Los nodos en un sistema pueden estar comunicados de diferentes maneras. Cada configuración tiene sus ventajas y desventajas.

Veremos algunas de las configuraciones que han sido implementadas hasta la fecha y las compararemos respecto de los siguientes criterios :

- **Costo básico** : Cuánto cuesta unir los diferentes nodos en el sistema ? Cuánto cuesta anexar un nodo ?
- **Costo de comunicación** : Cuánto tiempo tarda entregar un mensaje del nodo A al nodo B ?
- **Confiabilidad** : Si una conexión a un nodo falla, pueden comunicarse los otros nodos entre sí ?

7.4. - **ARQUITECTURAS DE MEMORIA DISTRIBUIDA**

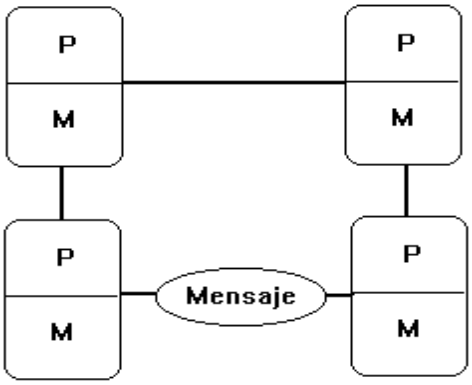


Fig. 7.1. - Estructura MIMD con memoria distribuida.

Las arquitecturas de memorias distribuidas conectan nodos de procesamiento consistentes de un procesador autónomo y su memoria local (ver Fig. 7.1) con una red de interconexión de procesador-a-procesador.

Los nodos comparten datos pasándose mensajes explícitamente a través de la red, debido a que no existe una memoria compartida.

Como un producto de las investigaciones de los años 80 estas arquitecturas se construyeron en un esfuerzo de proveer una arquitectura multiprocesador que pudiera "expandirse" (crecer en cantidad de procesadores) y satisficiera los requerimientos de performance de voluminosas aplicaciones científicas caracterizadas por referencias locales a datos.

Se han propuesto varias topologías de red de interconexión que permiten esta expansibilidad arquitectural y una mayor performance para programas paralelos y que difieren en los patrones de comunicación entre los procesadores. Veremos algunas a continuación.

7.4.1 - **Totalmente conectada (Completely connected)**

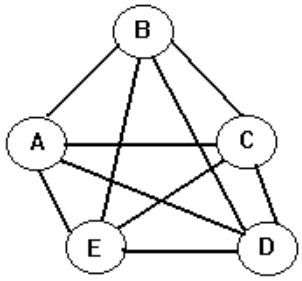


Fig. 7.2. - Totalmente conectada.

En esta topología cualquier nodo en el sistema está conectado a todos los otros nodos de la red. (Fig. 7.2).

El costo básico de esta configuración es alto, ya que debe existir una conexión directa entre cada dos nodos. El costo de anexar un nuevo nodo crece según X, donde X es la cantidad de nodos que contiene la red.

Sin embargo, en este entorno los mensajes entre nodos pueden entregarse muy velozmente, cualquier mensaje utiliza solamente una conexión para viajar entre dos nodos.

Estos sistemas son muy confiables ya que deben fallar muchos nodos para que el sistema completo falle.

Un sistema está particionado si se ha dividido en dos subsistemas que no pueden comunicarse entre sí.

7.4.2 - **Parcialmente conectada (Partially connected)**

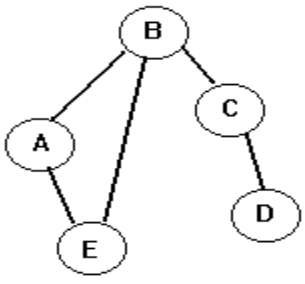


Fig. 7.3. - Parcialmente conectada.

En una red parcialmente conectada existen conexiones directas entre dos nodos pero no para todos los nodos (Fig. 7.3). De allí que el costo básico de la red sea menor.

Un mensaje de un nodo a otro puede requerir viajar entre varios nodos antes de arribar a su destino, lo que resulta en una comunicación más lenta.

No es tan confiable como la totalmente conectada.

Por ejemplo en la figura si falla la conexión entre el nodo B y el nodo C entonces la red está particionada en dos subsistemas. Para minimizar esta posibilidad en general cada nodo está conectado con por lo menos otros dos nodos.

7.4.3 - **Jerárquica o Arbol (Tree)**

En una red jerárquica los nodos están organizados como un árbol (Fig. 7.4). Es una organización común en las redes de computadoras corporativas donde las oficinas individuales están conectadas a una oficina central local, las que a su vez cuelgan de las oficinas regionales y esta finalmente de las oficinas en la central.

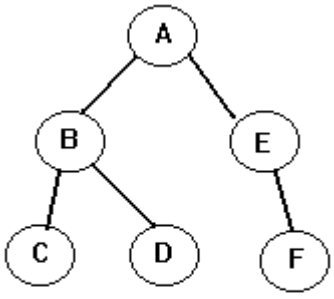


Fig. 7.4. - Jerárquica o Arbol.

Cada nodo (excepto la raíz) tiene un único antecesor y varios hijos. El costo básico de esta configuración es menor que el de la parcialmente conectada.

Un padre y su hijo se comunican directamente. Los hermanos se comunican entre sí solo a través del padre. En forma similar los primos solo pueden comunicarse a través de sus abuelos.

Si un nodo falla todos sus hijos quedan incomunicados con el resto de la red. En general, la falla de cualquier nodo puede particionar la red en varios subárboles disjuntos.

Aún cuando se han propuesto varias topologías basadas en la estructura de árbol, los árboles binarios completos han sido la variante más analizada.

Se han empleado varias estrategias para reducir el diámetro de comunicación de estas topologías (2(n-1) para un árbol binario completo de n niveles y 2^n-1 procesadores). Algunas soluciones incluyen agregar conexiones extra a la red para unir todos los nodos de un mismo nivel.

7.4.4 - **Estrella (Star)**

En una red estrella uno de los nodos está conectado a todos los demás (Fig. 7.5).

El costo básico de este sistema es lineal respecto del número de nodos.

El costo de comunicación es también bajo, ya que un mensaje del proceso A al proceso B solamente requiere de dos transferencias (Desde A a C y de C a B). Sin embargo esta velocidad puede no ser tal ya que el nodo central puede convertirse en un cuello de botella.

Mientras haya pocos mensajes la velocidad se mantendrá alta. En algunos sistemas estrella el nodo central está totalmente dedicado al pasaje de mensajes.

Si el nodo central falla la red queda totalmente particionada.

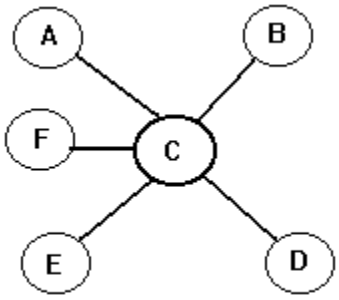


Fig. 7.5. - Estrella

7.4.5 - **Anillo (Ring)**

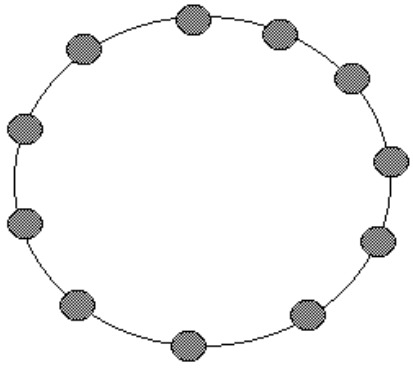


Fig. 7.6. - Anillo

En una red anillo cada nodo está conectado a solo otros dos nodos (Fig. 7.6).

El anillo puede ser unidireccional o bidireccional. En una arquitectura unidireccional un nodo puede transmitir información hacia uno solo de sus vecinos. Todos los nodos entregan información hacia la misma dirección.

Esta topología ha sido muy utilizada por IBM para sus redes denominadas "Token Ring".

En una arquitectura bidireccional un nodo puede transmitir información hacia cualquiera de sus vecinos.

De forma típica, se utilizan paquetes de mensajes de tamaño fijo incluyendo un campo que indica el nodo destino.

El costo básico de un anillo es nuevamente lineal respecto de la cantidad de nodos.

Sin embargo el costo de comunicación puede ser bastante alto ya que un mensaje de un nodo a otro debe viajar alrededor del anillo hasta que llega a destino. En un anillo unidireccional deberá recorrer a lo sumo n-1 nodos, en tanto que en un bidireccional a lo sumo n/2.

En una red bidireccional deben fallar dos conexiones para que la red se particione. En un anillo unidireccional la falla de un solo nodo particionará la red.

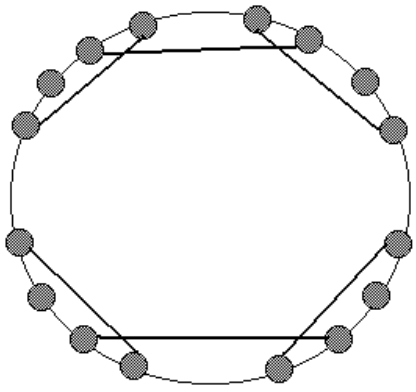


Fig. 7.7. - Anillo cordado.

Un remedio que suele utilizarse es el proveer al anillo de una doble conexión como puede verse en la Fig. 7.7. Esta arquitectura se denomina *anillo cordado (chordal ring)*.

Las topologías de anillo son más apropiadas para un pequeño número de procesadores que ejecutan algoritmos en donde lo predominante no debe ser la comunicación de datos.

En el Capítulo 20 de Sistemas Distribuidos, veremos algo más respecto de las topologías de Anillo.

7.4.6 - **Red con vecinos cercanos (Mesh connected)**

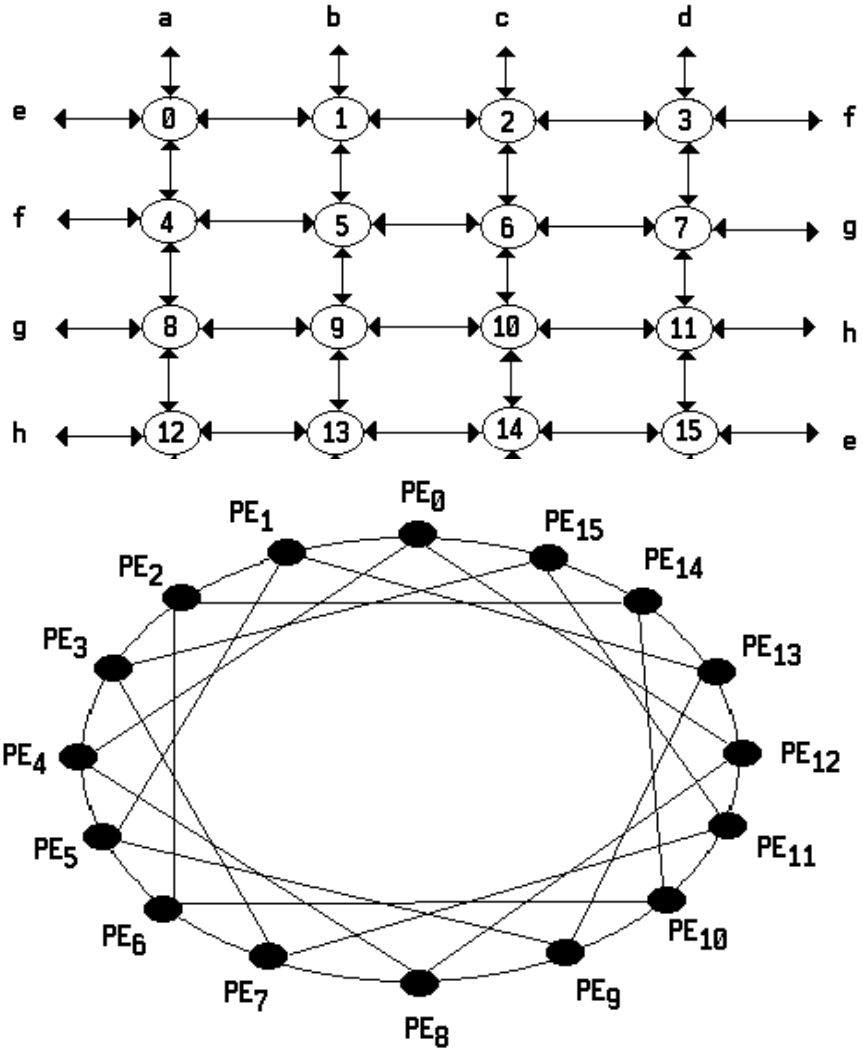


Fig. 7.9. - Red con vecinos cercanos vista de otra forma.

Para simplificar la explicación de este tipo de red también conocida como **Near-neighbor mesh** (mesh: grilla, malla) la aplicaremos a un ejemplo específico.

Sean 16 nodos N numerados del 0 al 15. Cada N(i) puede enviar mensajes a cada N(**i+1**), N(**i-1**), N(**i-r**) y N(**i+r**); siendo r la raíz cuadrada de N (generalmente se elige N como un numero de cuadrado perfecto).

En la Fig. 7.8 puede verse esta estructura y en la Fig. 7.9 puede verse la misma estructura dibujada de otra forma.

Cada N(i) está conectado aquí a sus cuatro vecinos más cercanos en la red. Este tipo de red es una red parcialmente conectada.

Los pasos que insume transferir datos desde un nodo N(i) a un nodo N(j) en una red de tamaño N es un valor cuya cota superior es :

I = SQRT(N) - 1

En una red de por ejemplo 64 nodos se necesitan a lo sumo 7 pasos para rutear un dato de cualquier nodo a otro.

Se puede aumentar la comunicación agregando conexiones adicionales de tipo diagonal o usando buses para conectar nodos por fila o columna.

La correspondencia entre estas topologías y los algoritmos orientados a matrices alientan las investigaciones sobre estas arquitecturas.

7.4.7 - **Cubo**

En la Fig. 7.10 se ilustra un cubo tridimensional.

Las líneas verticales conectan vértices cuyas direcciones difieren en el bit más significativo. Los vértices en ambos extremos de las líneas diagonales difieren en el bit de posición media. Las líneas horizontales difieren en el bit menos significativo.

Este concepto de cubo puede extenderse a un espacio de dimensión n obteniéndose el n-cubo con n bits para cada vértice.

Una red n-cubo se corresponde con N nodos donde $n = \log_2 N$.

En un n-cubo cada nodo está conectado exactamente con n vecinos. Esos vecinos difieren exactamente en un bit.

Un cubo n-dimensional que conecta 2^n nodos con n >> 3 se denomina hipercubo.

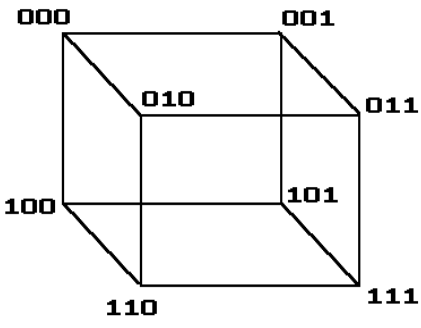


Fig. 7.10. - N-cubo de grado 3.

Las características principales de esta red son :

- La comunicación entre los procesadores es mediante caminos redundantes, luego, pueden ocurrir sobrecargas internas.

- Debido a las múltiples conexiones entre los nodos, las fallas en una rama no provocan la partición de la red.
- Para un cubo n-dimensional con 2^n nodos, la máxima distancia de comunicación entre dos nodos es **n** = $\log_2$ N.

Por ejemplo un cubo tridimensional requiere solamente 3 inputs por nodo para 2^3 procesadores. La cantidad de inputs crece con la cantidad de nodos, por ejemplo 256 nodos requieren 8 inputs para cada procesador.

En la Fig. 7.11 puede apreciarse la función de ruteo de datos entre los nodos de un n-cubo de grado 3.

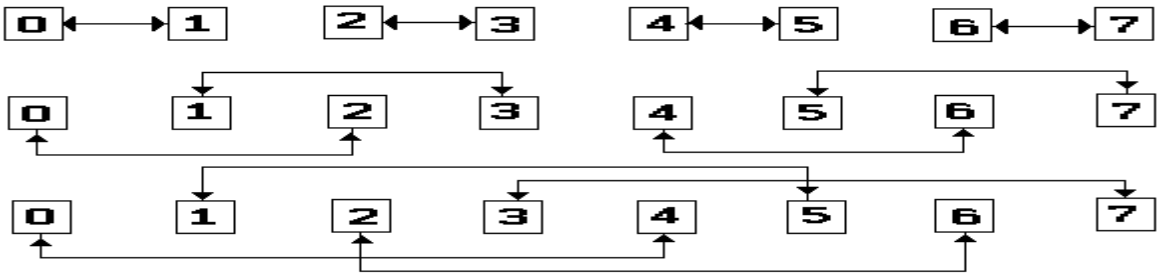


Fig. 7.11. - Función de ruteo de datos en un n-cubo de grado 3.

En la figura 7.12 se grafica un cubo cuatridimensional.

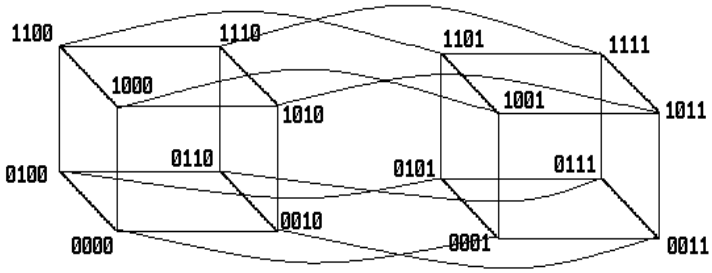


Fig. 7.12. - Hipercubo cuatridimensional.

7.4.8 - **Red Barrel Shifter**

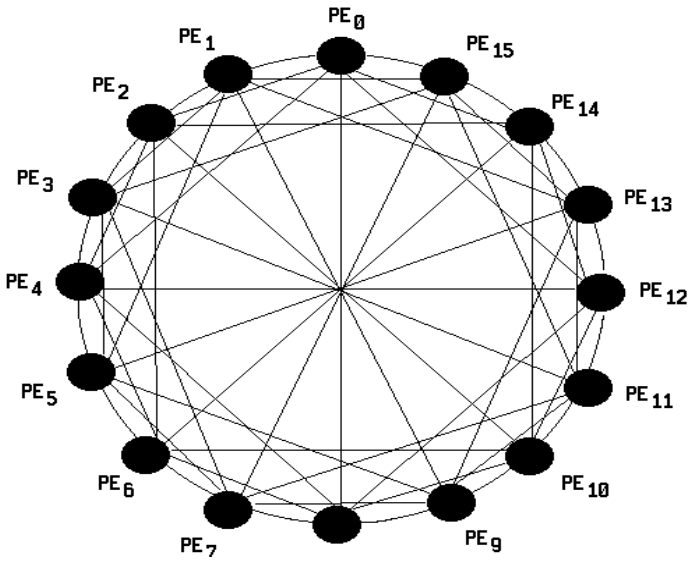
A título gráfico incluimos las redes Barrel Shifter conocidas también como redes PM2I (plus-minus-2^i).

En un Barrel Shifter de tamaño N con N = 2^n, se requieren B pasos para transmitir un mensaje de un nodo a otro, estando B acotado por :

B = $\log_2$ N / 2

La topología de red con Vecinos Cercanos es un subconjunto de las Barrel Shifter.

Por ejemplo, para N = 16 una topología con Vecinos Cercanos cuenta con 32 conexiones en tanto que un Barrel Shifter cuenta con 56. Véanse las figuras 7.13 y 7.14.



7.4.9. - **Arquitecturas con topología reconfigurable**

Si bien las arquitecturas de memoria distribuida conllevan a una topología física subyacente, las arquitecturas reconfigurables proveen conmutadores (switches) programables que permiten que el usuario seleccione la mejor topología lógica que se adecue a los patrones de comunicación que necesite (ver Fig. 7.15).

Algunas de estas máquinas se caracterizan

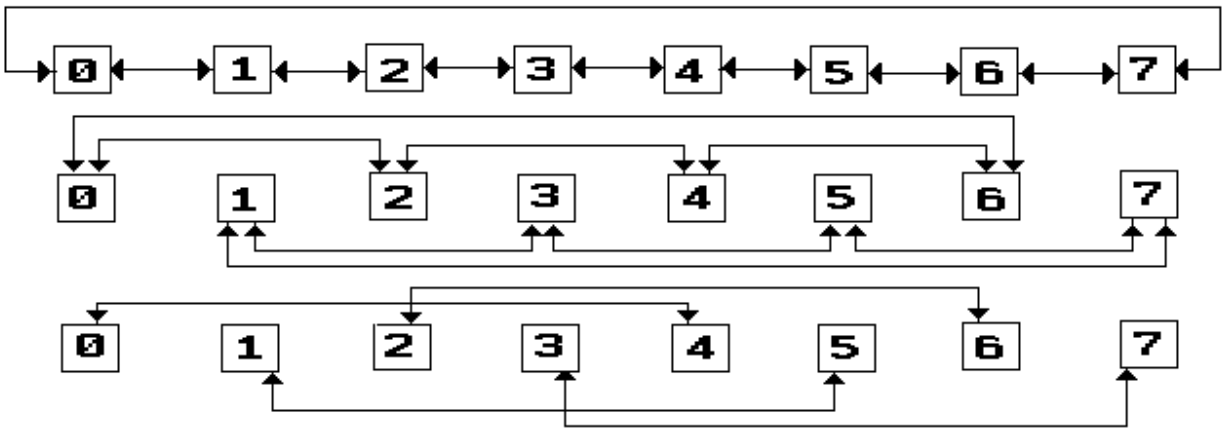


Fig. 7.14. - Función de ruteo de datos en un Barrel Shifter con 8 nodos.

por permitir definir distintas topologías lógicas (como en la Configurable Highly Parallel Computer o Chip de Lawrence Snyder) o por permitir el particionamiento de una topología en múltiples topologías del mismo tipo (como la Partitionable SIMD/MIMD System o Pasm de Howard J. Siegel).

El principal motivo para construir arquitecturas reconfigurables es el hecho de tratar de obtener que una arquitectura pueda actuar como muchas arquitecturas de propósito específico.

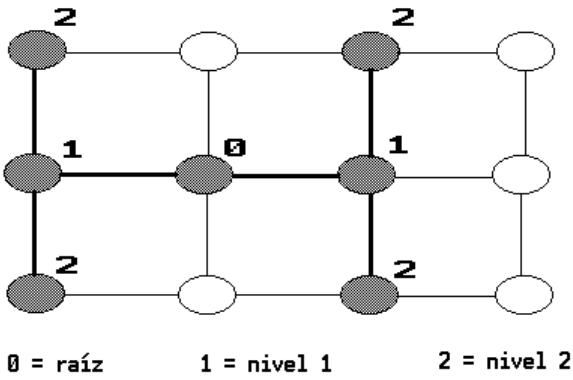


Fig. 7.15. - Una topología jerárquica mapeada sobre una red con vecinos cercanos de tipo reconfigurable.

7.5. - **ARQUITECTURAS DE MEMORIA COMPARTIDA**

Las arquitecturas de memoria compartida logran la coordinación entre los procesadores proveyendo una memoria global y compartida que cada procesador puede direccionar.

Estas arquitecturas tienen múltiples procesadores de propósito general que comparten la memoria, y no son del tipo CPU y procesadores de E/S.

Las computadoras de memoria compartida no tienen los problemas de las arquitecturas basadas en el pasaje de mensajes, como ser la latencia de envío del mensaje así como el encolamiento de datos desde y hacia los nodos.

Sin embargo, deben resolver problemas tales como la sincronización de acceso a los datos y la coherencia de la memoria cache.

La coordinación de los procesadores con variables compartidas requiere de mecanismos de sincronización atómicos para evitar que un proceso acceda a un dato antes de que el otro termine de actualizarlo.

Este mecanismo provee un "semáforo" que está sujeto al dato y que debe testearse antes de accederlo. El mecanismo "test-and-set" es un ejemplo de una operación atómica para controlar el valor del semáforo.

Generalmente cada procesador en una arquitectura de este tipo tiene una memoria local utilizada como cache. Sin embargo, pueden existir varias copias de la misma porción de memoria compartida en las cache de los procesadores en un momento dado.

Mantener una versión consistente de tales datos es el problema de la coherencia de la cache, el cual conlleva a que deben proveerse nuevas versiones a cada procesador cada vez que uno de los procesadores actualiza su copia.

A pesar de que los sistemas con un número pequeño de procesadores utilizan mecanismos hardware que "espían" las caches para determinar si han sido actualizadas, en los sistemas grandes se confía más en mecanismos de software para minimizar el impacto en la performance.

7.5.1 - **Bus Multiacceso (Shared Bus)**

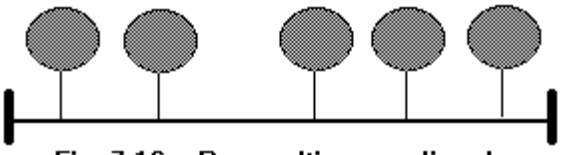


Fig. 7.16. - Bus multiacceso lineal.

En una red bus multiacceso existe una única conexión compartida : el bus. Todos los nodos están conectados a él, el cual puede estar organizado en forma lineal (Fig. 7.16) o en forma de anillo (Fig. 7.17). Se la conoce también como topología de Barra.

Los nodos pueden comunicarse a través del bus.

El costo básico de esta red es lineal respecto del número de nodos.

Efectivamente, un único bus de tiempo compartido se adecua bien para una cierta cantidad de procesadores (de 4 a 20), ya que uno solo de los procesadores accede al bus por vez.

Algunas arquitecturas basadas en bus como la arquitectura experimental Cm***** utilizan dos tipos de buses: uno local que une un conjunto-cluster de procesadores y uno de sistema de más alto nivel que une los procesadores dedicados a servicios que se asocian a cada conjunto-cluster.

El costo de comunicación es bastante bajo aunque la conexión puede convertirse en un cuello de botella.

Nótese que esta configuración es similar a la estrella salvo que aquí es un bus el que hace las veces de nodo central.

La falla de un nodo no afecta al resto. Sin embargo, si el que falla es el bus la red queda totalmente particionada.

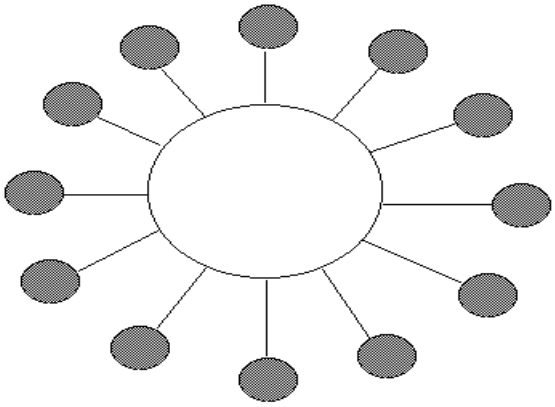


Fig. 7.17. - Bus multiacceso anillo.

7.5.2. - **Crossbar Switch**

El Crossbar Switch es un intento de superar el reducido troughput del sistema de bus multiacceso. El ejemplo más conocido es el crossbar switch del C.mmp.

En el C.mmp el crossbar switch conecta N procesadores con M módulos de memoria, pero también puede utilizarse para interconectar nodos en una red.

En cuanto a costo básico esta configuración es relativamente cara. Cuando la cantidad de procesadores es aproximadamente igual a la cantidad de memoria el costo es proporcional a N^2.

El crossbar switch provee una total conectividad con todos los módulos de memoria debido a que existe un bus separado para cada uno de ellos. Sin embargo, el máximo número de transferencias que pueden tener lugar simultáneamente está limitado a la cantidad de tales módulos.

Los conflictos se producen cuando se requieren dos o más solicitudes sobre el mismo módulo.

Los procesadores pueden competir para acceder a una ubicación en memoria, pero el crossbar impide la contención de las líneas de comunicación proveyendo un camino dedicado entre cada posible par de procesador/memoria.

Para proveer de más flexibilidad requerida en el acceso a los dispositivos de E/S, una extensión natural del crossbar switch es utilizar un switch similar en la parte de los dispositivos de E/S.

En la Fig. 7.18 puede verse este esquema conjuntamente con el esquema clásico para la interconexión de memorias-procesadores.

El crossbar switch es muy poderoso cuando existe un alto bandwidth.

La confiabilidad del switch es problemática; sin embargo puede mejorarse mediante segmentación y redundancia en el switch.

En general, es normalmente bastante fácil particionar el sistema para aislar las unidades lógicas que funcionan mal.

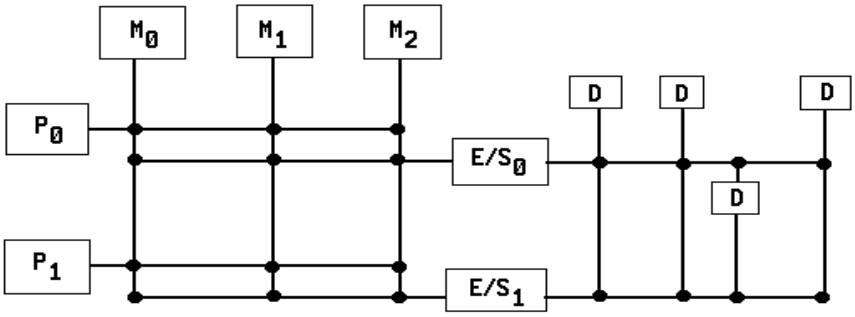


Fig. 7.18. - Crossbar Switch.

Cuesta trabajo justificar el uso del crossbar switch para grandes sistemas multiprocesadores, debido a la ausencia de un switch a un costo razonable y de buena performance.

El consumo de energía, la cantidad de patas de las conexiones (pinout) y ciertas consideraciones de tamaño han limitado las arquitecturas crossbar a un número pequeño de procesadores (de 4 a 16).

7.5.3. - **Redes de interconexión de múltiples etapas**

Las redes de interconexión de múltiples etapas (MIN- Multistage Interconnection Network) logran un compromiso entre las alternativas de precio/performance ofrecidas por los crossbar y los buses.

Una MIN de N*N conecta N procesadores a N memorias utilizando múltiples etapas o bancos de switches en el camino de la red de interconexión.

Cuando N es una potencia de 2, una forma es utilizar $\log_2 N$ etapas de N/2 switches, usando 2*2 switches

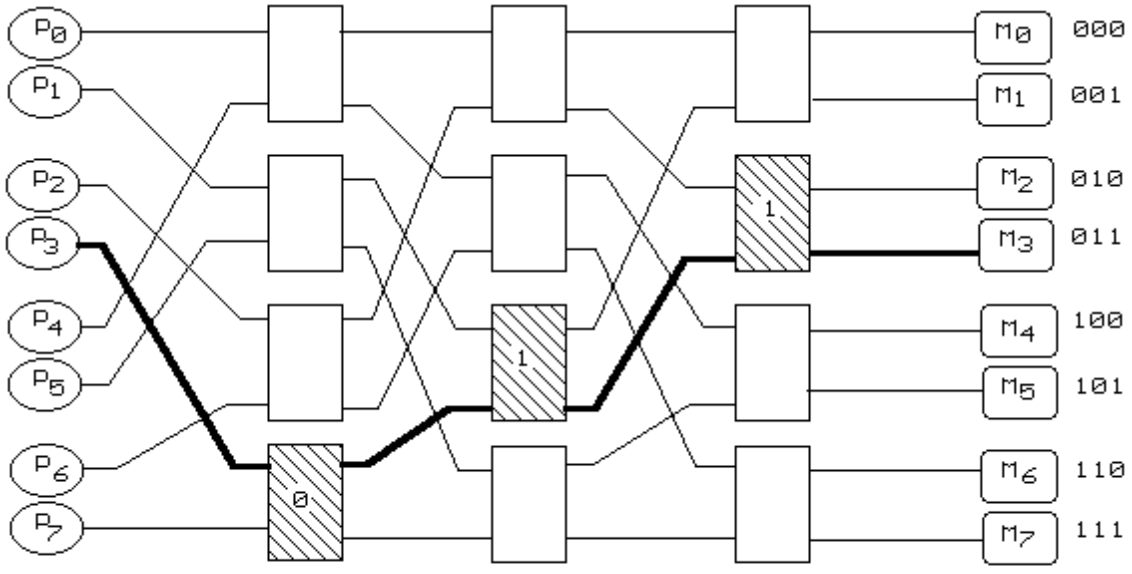


Fig. 7.19. - Ruteo de un requerimiento de P3 a M3 en una red Omega MIN de 8 * 8.

(ver Fig. 7.19). Un procesador que desea acceder a memoria hace un requerimiento especificando la dirección de destino (y el camino) mediante un vector de bits en donde cada uno es un bit de control para cada etapa.

El switch en la etapa i-ésima examina el i-ésimo bit para determinar si el pedido se rutea al output mayor o menor.

En la figura se puede ver una red omega que conecta 8 procesadores y memorias, cuando el bit de control está en cero indica que el requerimiento debe dirigirse al mayor output.

Una característica muy significativa de las redes MIN es que se pueden expandir, debido a que el diámetro de comunicación es proporcional a $\log_2 N$.

El Butterfly BBN (Blot, Neranek, y Newman) puede configurarse hasta abarcar 256 procesadores.

EJERCICIOS

1) Cuáles son las grandes razones para construir Sistemas Distribuidos ?
2) Qué es una topología de red ? Cuáles son los criterios utilizados en la materia para compararlas ?
3) Qué es el particionamiento de la red y cuándo sucede ?
4) Grafique e indique las características de las siguientes topologías de red :
- Totalmente conectada
- Parcialmente conectada
- Jerárquica o árbol
- Estrella
- Anillo
- Topologías reconfigurables
- Red con vecinos cercanos
- Hipercubo (grado 4)
- Barrel Shifter
- Bus multiacceso (shared bus)
- Crossbar switch
- MIN

5) Construya una topología de malla reconfigurable que mapee una estructura de anillo.
6) Construya la función de ruteo de datos en un Hipercubo de grado 4.
7) Cuanto pasos máximos se requieren para rutear un mensaje de un nodo a otro en un Barrel Shifter de 256 nodos ?
8) En qué consiste la similitud de una topología Estrella con una topología de Bus Multiacceso ?
9) Cuál es la característica de las direcciones de los nodos adyacentes en un hipercubo de grado **s** ? Cuántos nodos tiene este hipercubo ?