



UNIVERSITY OF ENGINEERING AND TECHNOLOGY, TAXILA

FACULTY OF TELECOMMUNICATION AND INFORMATION ENGINEERING

COMPUTER/SOFTWARE ENGINEERING DEPARTMENT

PARALLEL PROCESSING

2K5-CP,SE

INSTRUCTOR: ENGR.WAJAHAT ABBAS

Lecture-4

- SIMD Architecture
- MIMD Architecture
- DSM Architecture

SIMD ARCHITECTURE

The SIMD model of parallel computing consists of two parts: a front-end computer of the usual von Neumann style, and a processor array as shown in Figure 1.4. The processor array is a set of identical synchronized processing elements capable of simultaneously performing the same operation on different data. Each processor in the array has a small amount of local memory where the distributed data resides while it is being processed in parallel.

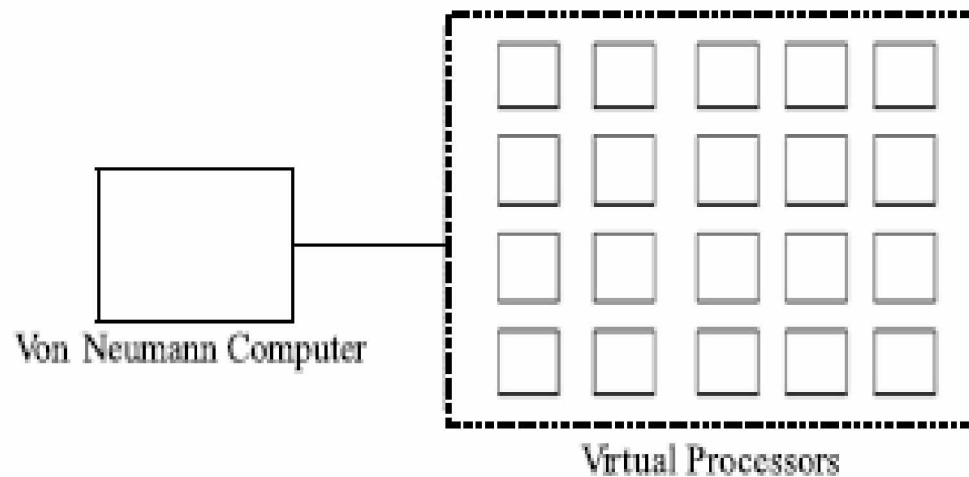


Figure 1.4 SIMD architecture model.

SIMD CONFIGURATIONS

First Scheme

In the first scheme, each processor has its own local memory. Processors can communicate with each other through the interconnection network. If the interconnection network does not provide direct connection between a given pair of processors, then this pair can exchange data via an intermediate processor.

Second Scheme

In the second SIMD scheme, processors and memory modules communicate with each other via the interconnection network. Two processors can transfer data between each other via intermediate memory module(s) or possibly via intermediate processor(s).

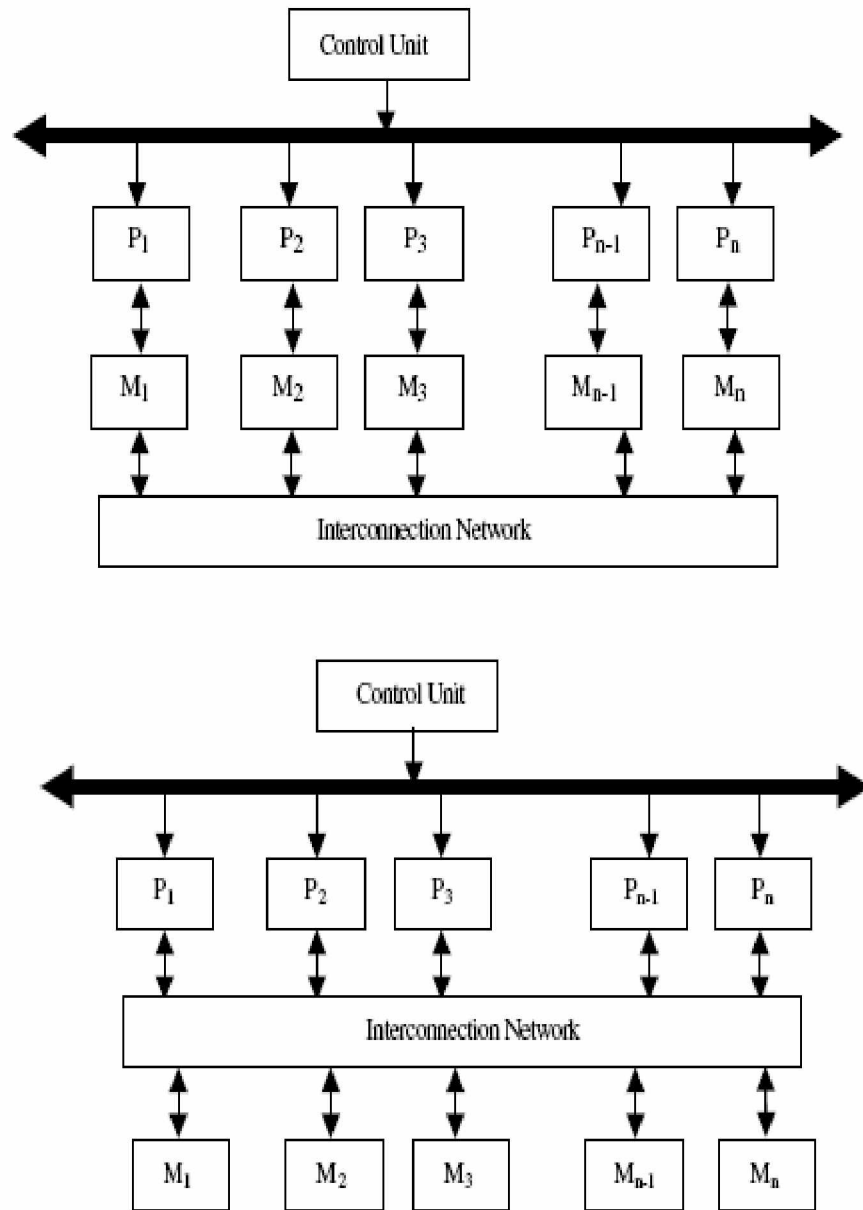


Figure 1.5 Two SIMD schemes.



MIMD ARCHITECTURE

Multiple-instruction multiple-data streams (MIMD) parallel architectures are made of multiple processors and multiple memory modules connected together via some interconnection network. They fall into two broad categories: shared memory or message passing. Figure 1.6 illustrates the general architecture of these two categories.

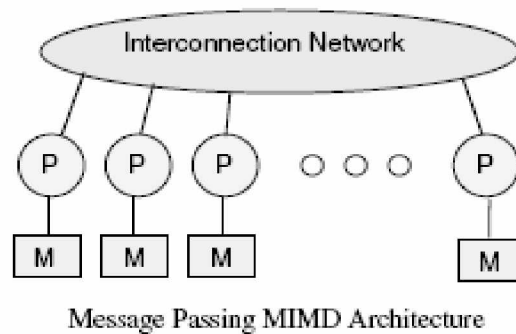
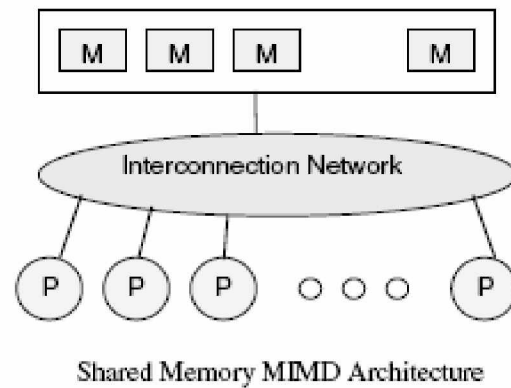


Figure 1.6 Shared memory versus message passing architecture.



UNIVERSITY OF ENGINEERING AND TECHNOLOGY, TAXILA

FACULTY OF TELECOMMUNICATION AND INFORMATION ENGINEERING

COMPUTER/SOFTWARE ENGINEERING DEPARTMENT

Shared Memory System

A shared memory system typically accomplishes interprocessor coordination through a global memory shared by all processors. These are typically server systems that communicate through a bus and cache memory controller. The bus/cache architecture alleviates the need for expensive multiported memories and inter- face circuitry as well as the need to adopt a message-passing paradigm when developing application software. Because access to shared memory is balanced, these systems are also called SMP (symmetric multiprocessor) systems.

Shared Memory Organization

A shared memory model is one in which processors communicate by reading and writing locations in a shared memory that is equally accessible by all processors. Each processor may have registers, buffers, caches, and local memory banks as additional memory resources. A number of basic issues in the design of shared memory systems have to be taken into consideration. These include access control, synchronization, protection, and security. Access control determines which process accesses are possible to which resources.

Access control models make the required check for every access request issued by the processors to the shared memory, against the contents of the access control table. The latter contains flags that determine the legality of each access attempt. If there are access attempts to resources, then until the desired access is completed, all disallowed access attempts and illegal processes are blocked. Requests from sharing processes may change the contents of the access control table during execution. The flags of the access control with the synchronization rules determine the system's functionality.

Synchronization constraints limit the time of accesses from sharing processes to shared resources. Appropriate synchronization ensures that the information flows properly and ensures system functionality.

Protection is a system feature that prevents processes from making arbitrary access to resources belonging to other processes. Sharing and protection are incompatible; sharing allows access, whereas protection restricts it.



UNIVERSITY OF ENGINEERING AND TECHNOLOGY, TAXILA

FACULTY OF TELECOMMUNICATION AND INFORMATION ENGINEERING

COMPUTER/SOFTWARE ENGINEERING DEPARTMENT

Message Passing System

A message passing system (also referred to as distributed memory) typically combines the local memory and processor at each node of the interconnection network. There is no global memory, so it is necessary to move data from one local memory to another by means of message passing. This is typically done by a Send/Receive pair of commands, which must be written into the application software by a programmer. Thus, programmers must learn the message-passing paradigm, which involves data copying and dealing with consistency issues.

Message Passing Organization

Message passing systems are a class of multiprocessors in which each processor has access to its own local memory. Unlike shared memory systems, communications in message passing systems are performed via send and receive operations. A node in such a system consists of a processor and its local memory. Nodes are typically able to store messages in buffers (temporary memory locations where messages wait until they can be sent or received), and perform send/receive operations at the same time as processing. Simultaneous message processing and problem calculating are handled by the underlying operating system. Processors do not share a global memory and each processor has access to its own address space. Two important design factors must be considered in designing interconnection networks for message passing systems.

The link bandwidth is defined as the number of bits that can be transmitted per unit time (bits/s).

The network latency is defined as the time to complete a message transfer.

Wormhole routing in message passing was introduced in 1987 as an alternative to the traditional store-and-forward routing in order to reduce the size of the required buffers and to decrease the message latency. In wormhole routing, a packet is divided into smaller units that are called flits (flow control bits) such that flits move in a pipeline fashion with the header flit of the packet leading the way to the destination node. When the header flit is blocked due to network congestion, the remaining flits are blocked as well.



UNIVERSITY OF ENGINEERING AND TECHNOLOGY, TAXILA

FACULTY OF TELECOMMUNICATION AND INFORMATION ENGINEERING

COMPUTER/SOFTWARE ENGINEERING DEPARTMENT

MIMD computers with shared memory are known as multiprocessors or tightly coupled machines. Examples are ENCORE, MULTIMAX, SEQUENT & BALANCE.

MIMD computers with an interconnection network are known as multicomputers or loosely coupled machines. Examples are INTEL iPSC, NCUBE/7 and transputer networks.

Distributed shared memory (DSM) architecture

In such systems, memory is physically distributed; for example, the hardware architecture follows the message passing school of design, but the programming model follows the shared memory school of thought. In effect, software covers up the hardware. As far as a programmer is concerned, the architecture looks and behaves like a shared memory machine, but a message passing architecture lives underneath the software. Thus, the DSM machine is a hybrid that takes advantage of both design schools.

NEXT LECTURE

- Classification of Shared Memory Systems on basis of Interconnection Networks
- Factors for classification of Interconnection Networks