Chapter 10

PARALLEL PROCESSING

Assistant Professor

Er. Shiva Ram Dam

Contents:

1. Parallelism in Uniprocessor System
2. Organization of Multiprocessor System
3. Communication in Multiprocessor Systems
4. Memory Organization in Multiprocessor System

Parallel Processing

- In computers, parallel processing is the processing of program instructions by dividing them among multiple processors with the objective of running a program in less time.
- The amount of hardware increases with parallel processing, and with it, the cost of the system increases.
- When a system processes two or more different instructions simultaneously, it is performing parallel processing.

Parallel Processing

- Parallel Processing (parallelism) is one of the methods to improve system performance.
- Parallel processing means to process more than one (task) operation at a time.
- **Uniprocessor system**: The system with one CPU. Uniprocessor system achieve parallelism both within CPU and Computer system as a whole.
- **Multiprocessor system**: The system with more than one CPU. They achieve parallelism by having more than one processor performing tasks simultaneously.

- Figure below shows one possible way of separating the execution unit into eight functional units operating in parallel

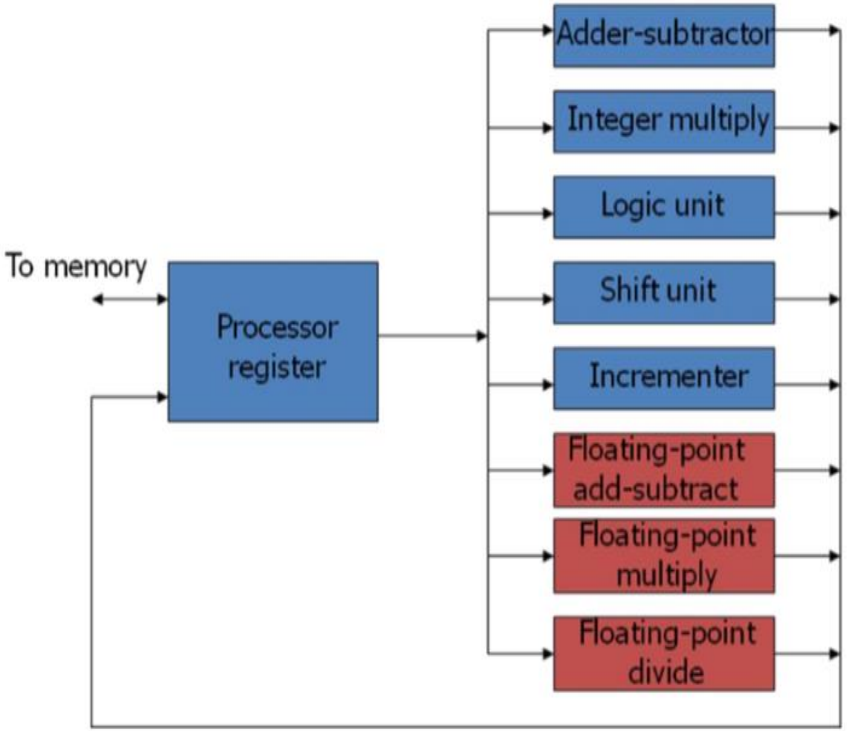


Figure: Processor with multiple functional units

- The operands in the registers are applied to on of the units depending on the operation specifi by the instruction associated with the operand
- The operation performed in each functional un is indicated in each block of the diagram.
- The adder and integer multiplier perform the arithmetic operations with integer numbers.
- The floating-point operations are separated int three circuits operating in parallel. The logic, shift, and increment operations can be performed concurrently on different data.
- All units are independent of each other, so one number can be shifted while another number i being incremented.
- A multifunctional organization is usually associated with a complex control unit to coordinate all the activities among the various components.

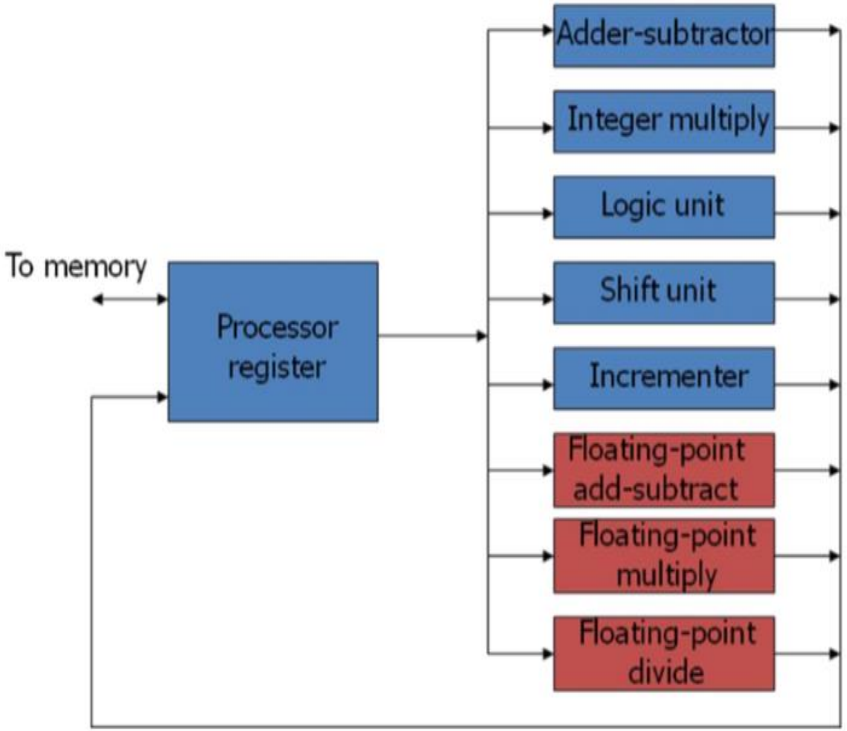


Figure: Processor with multiple functional units

Parallelism in uniprocessor system

- A system that process two different instructions simultaneously, can be consider as parallel processing system.

  FETCH2: DR← M , PC←PC+1.

  - Here, 2 micro-operation occurs during fetch 2 state.
  - But both are used to process same instruction so, it is not considered as parallel processing.
- The intanium microprocessor can fetch 3 instruction simultaneously.
  - Instruction Pipelining => (overlapping fetch, decode and execute)
  - Arithmetic Pipelining => a+b*c (for i= 0 to 100)
  - A system with DMA Controller (in transparent mode)

Instruction pipelining

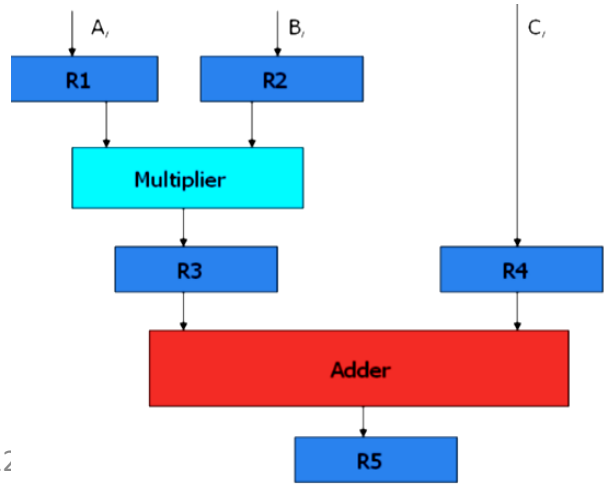
Clock cycle:	1	2	3	4	5	6	7
1. Load *R*1	I	A	E				
2. Load *R*2		I	A	E			
3. No-operation			I	A	E		
4. Add *R*1 + *R*2				I	A	E	
5. Store *R*3					I	A	E

(b) Pipeline timing with delayed load

Arithmetic Pipelining example

- Consider the below example: To perform the combined multiply and add operations with a stream of numbers $A_i * B_i + C_i$ for i = 1, 2, 3, …, 7
- Each sub-operation is to be implemented in a segment within a pipeline.

$R1 \leftarrow A_i$, $R2 \leftarrow B_i$	Input A_i and B_i
$R3 \leftarrow R1 * R2$, $R4 \leftarrow C_i$	Multiply and input C_i
$R5 \leftarrow R3 + R4$	Add C_i to product



Clock Pulse Number	Segment 1		Segment 2		Segment 3
	R1	R2	R3	R4	R5
1	A_1	B_1	--	--	--
2	A_2	B_2	A_1*B_1	C_1	--
3	A_3	B_3	A_2*B_2	C_2	$A_1*B_1+C_1$
4	A_4	B_4	A_3*B_3	C_3	$A_2*B_2+C_2$
5	A_5	B_5	A_4*B_4	C_4	$A_3*B_3+C_3$
6	A_6	B_6	A_5*B_5	C_5	$A_4*B_4+C_4$
7	A_7	B_7	A_6*B_6	C_6	$A_5*B_5+C_5$
8	--	--	A_7*B_7	C_7	$A_6*B_6+C_6$
9	--	--	--	--	$A_7*B_7+C_7$

10.3 Organization of Multiprocessor System

- Characteristics of Multiprocessor
- Flynn’s Classification
- System topologies
- MIMD system architectures

a) Characteristics of multiprocessor:

- A multiprocessor system is controlled by one operating system that provides interaction between processor and all component of system.
- VLSI technology has reduced cost of multiple processor system.
- Multiprocessing improves reliability of the system so that a failure or error in one part has less effect on rest of system.
- Multiprocessor improves performance by decomposing a program into parallel executable tasks.
- Multiprocessor are classified by the way their memory organized. A multiprocessor with common shared memory is shared memory multiprocessor. A multiprocessor that has its own private local memory is distributed memory multiprocessor.

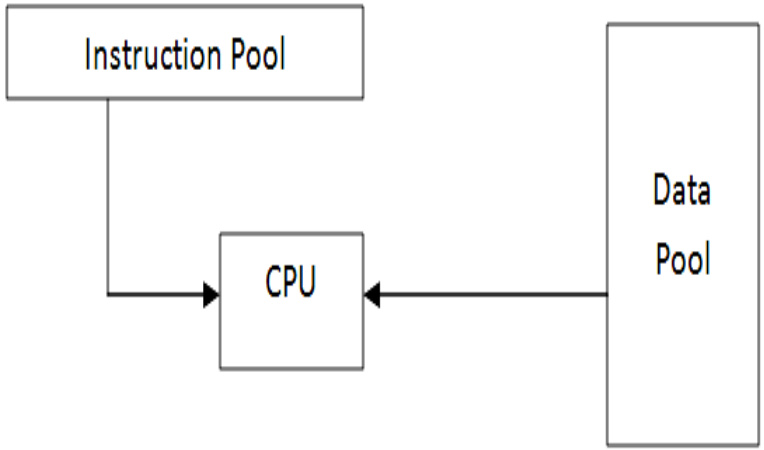
b) Flynn’s Classification

- There are a variety of ways that parallel processing can be classified.
  - Internal organization of the processors
  - Interconnection structure between processors
  - The flow of information through the system
- One classification introduced by M. J. Flynn considers the organization of a computer system by the number of instructions and data items that are manipulated simultaneously.
- The normal operation of a computer is to fetch instructions from memory and execute them in the processor.
- The sequence of instructions read from memory constitutes an instruction stream.
- The operations performed on the data in the processor constitute a data stream. Parallel processing may occur in the instruction stream, in the data stream, or in both.
- Flynn's classification divides computers into four major groups based on instruction and data processing.
- A computer is classified by whether it processes a single instruction at a time or multiple instructions simultaneously, and whether it operates on one or multiple data sets.

i.	SISD	Single Instruction with Single Data
ii.	SIMD	Single Instruction with Multiple Data
iii.	MISD	Multiple Instruction with Single Data
iv.	MIMD	Multiple Instruction with Multiple Data

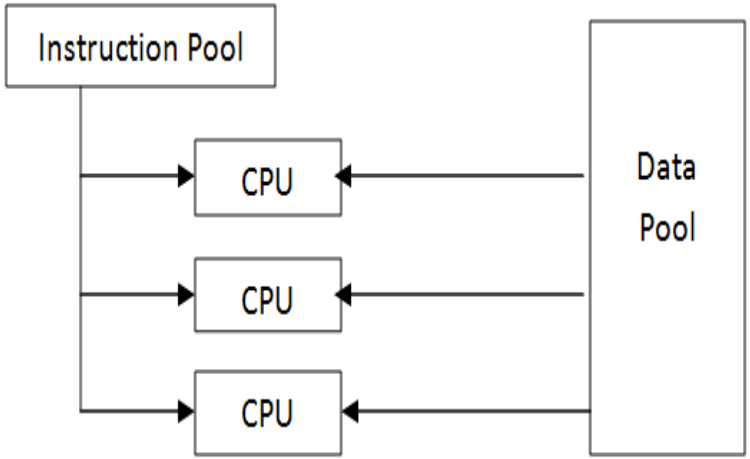
SISD

- Represents the organization of a single computer containing a control unit, a processor unit, and a memory unit.
- Instructions are executed sequentially and the system may or may not have internal parallel processing capabilities.
- Parallel processing may be achieved by means of multiple functional units or by pipeline processing.



SiMD

- Represents an organization that includes many processing units under the supervision of a common control unit.
- All processors receive the same instruction from the control unit but operate on different items of data.
- The shared memory unit must contain multiple modules so that it can communicate with all the processors simultaneously.
- Since there is only one instruction, each processor does not have to fetch and decode each instruction.



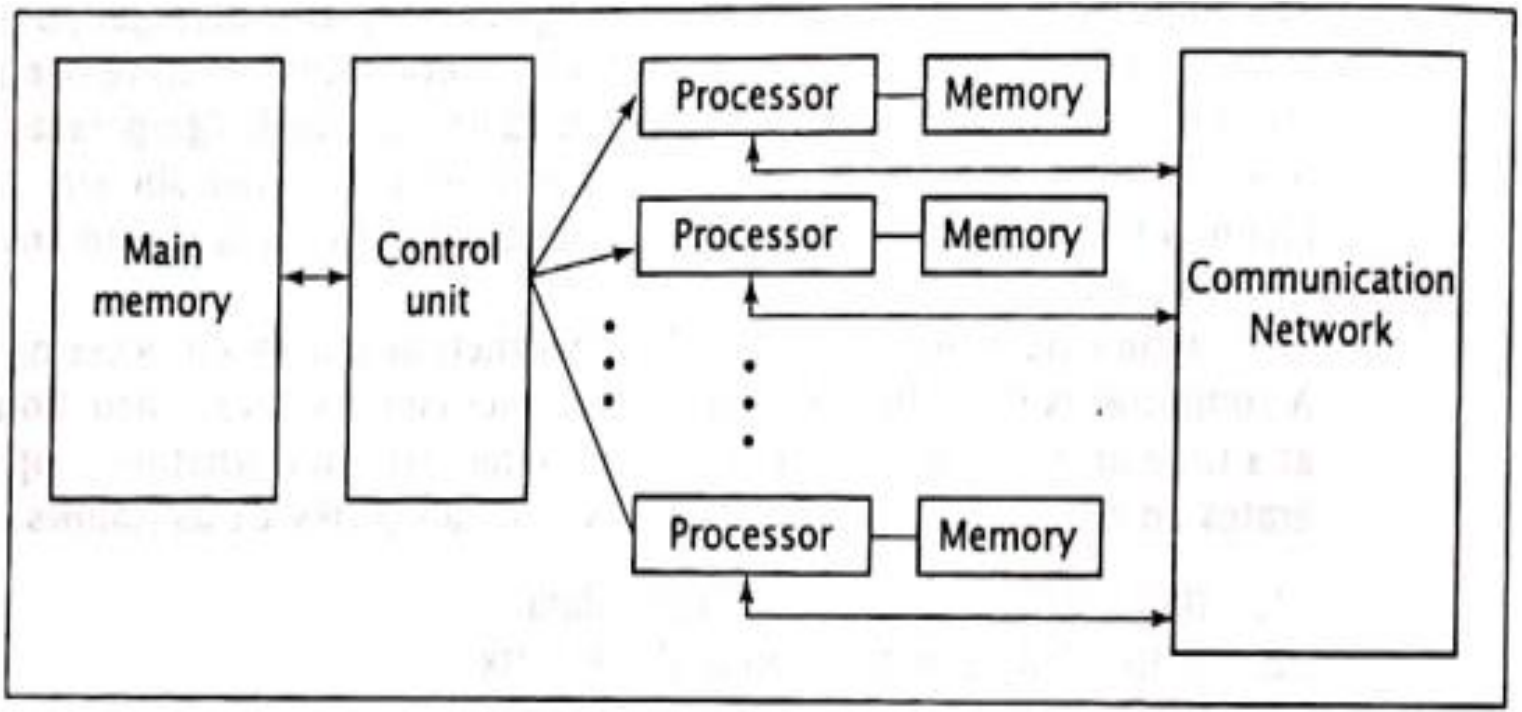
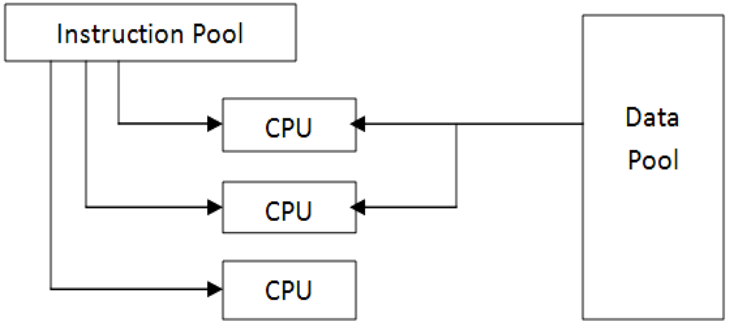


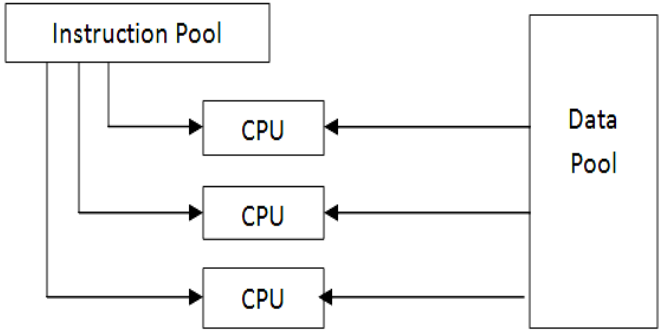
Figure: A generic SIMD organization

MISD and MIMD

- This classification is not practical to implement. So, no significant MISD computers ever been built. It is used for fault tolerance.
- Multiple instructions operate on a single data stream.



- Systems referred to as multiprocessors or multi-computers are usually MIMD. It may execute multiple instructions simultaneously unlike SIMD. So, each processor must include its own control unit. MIMD machines are well suited for general purpose use.

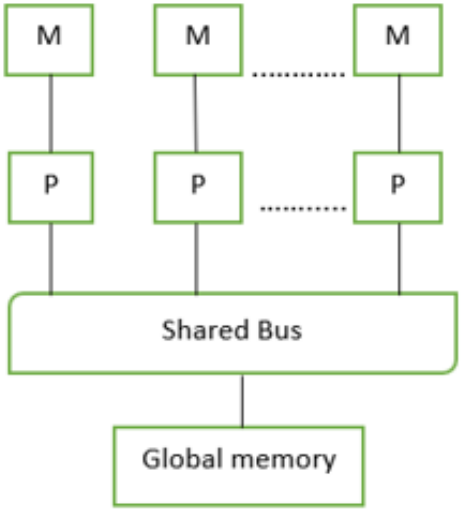


C) System Topologies

- It refers to the various ways of connection of processor. MIMD system topologies
  a) Shared bus Topology
  b) Ring Topology
  c) Tree Topology
  d) Mesh Topology
  e) Hyper cube Topology
  f) Completely connected Topology

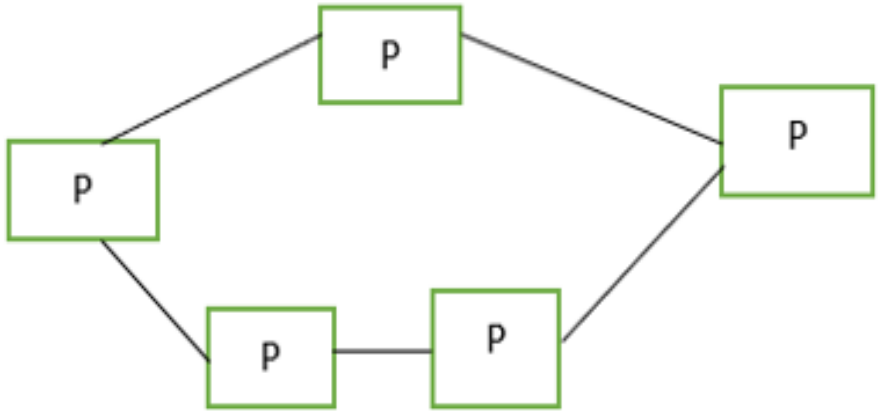
a) **Shared Bus Topology: -**

- Simplest Topology
- A processor communicates with each other through this bus.
- This bus can handle only one data transmission at a time.
- Easy to expand.
- If more processors are added then demand of bus will be high and result in delay.

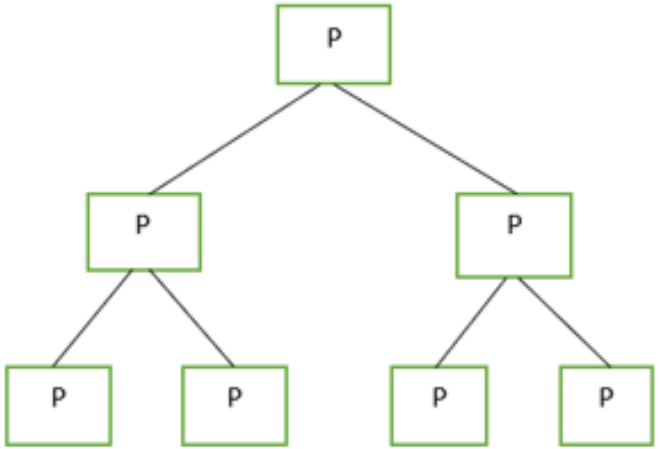


b) Ring Topology: -

It has dedicated connections between processors. A data needs to be travel through several processors to reach from source to final destination.

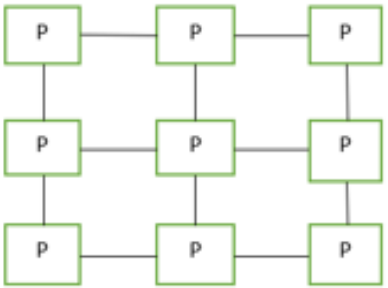


c) **Tree Topology**: - It also uses direct connection between processors. There is only one unique path between any pair of processors.



d) **Mess Topology: -** In this topology every processor is connected to its below and above processor as well as left and right processor.

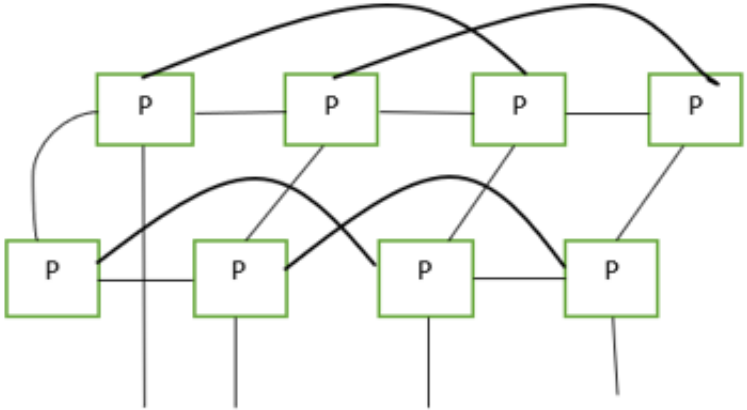
-given mesh topology is 3*3 mesh



Example: - IlliacIV Multiprocessor uses this topology.

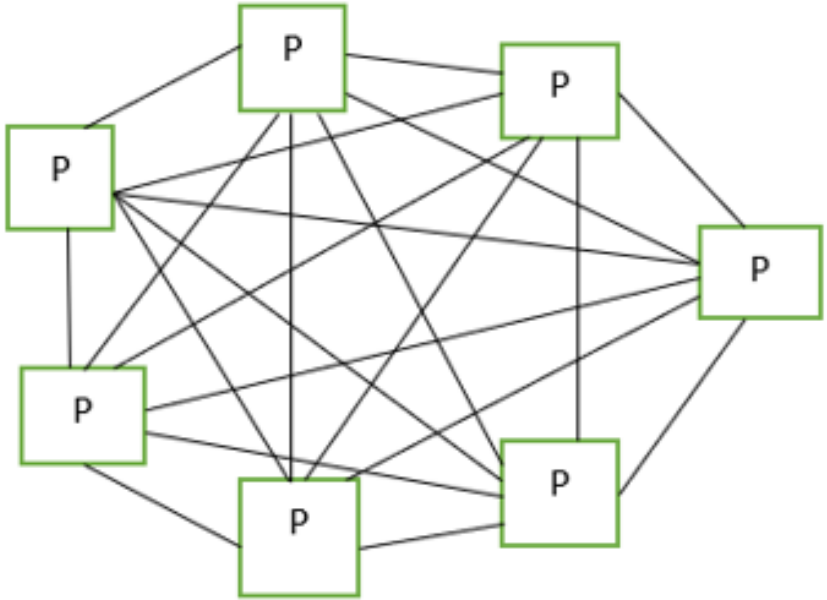
e) Hyper cube:-

- A multidimensional mesh
- Each processor connects to all processor whose binary values differ only by one bit.
  For example: -
  In fig: - processor 0 (0000) connects to processor 1 (0001), 2 (0010), 4 (0100) and 8 (1000)
  e.g: - n CUBE system use hypercube topology



f) **Complete Connected: -**

- Each and every processor is connected to all processor.
- Increases complexity but offers maximum communication.



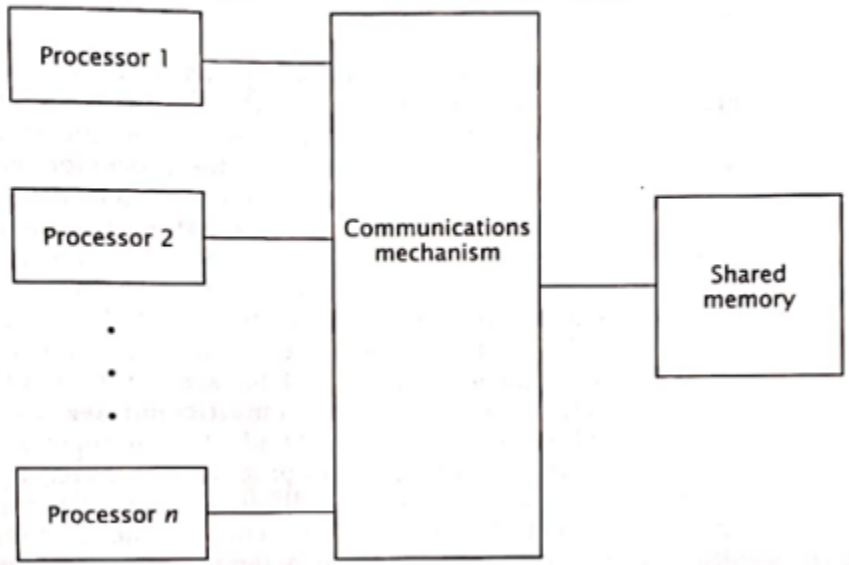
d) MIMD System Architecture:

- It refers to its connections with respect to system memory.
- **Symmetric multiprocessor (SMP)**
- A computer system that has two or more processors with comparable capabilities.
- All processors must be capable of performing the same functions.
- An integrated OS controls entire computer system.
- All processor has access to same I/O devices and memory modules.

- Types of SMD:
  - UMA (Uniform Memory Access)
  - NUMA (Non Uniform Memory Access)

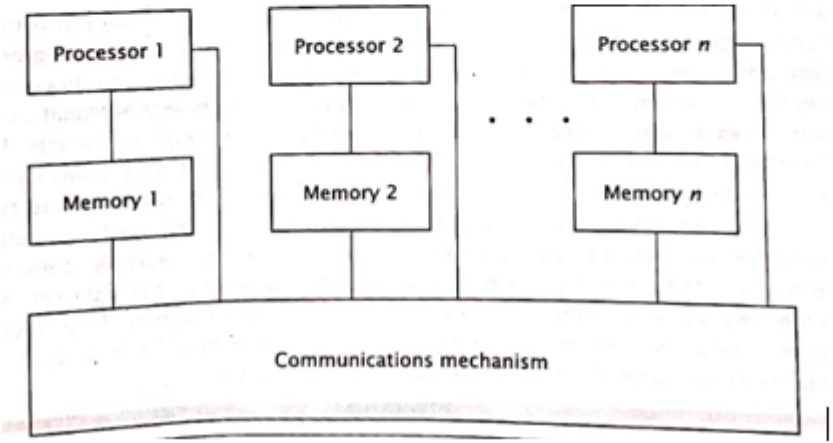
i) **UMA (Uniform Memory Access) architecture**

- It gives all CPU equal access to all locations in shared memory.
- They interact with shared memory through some common mechanism.



ii) **NUMA**

- Non uniform memory access architecture
- It does not allow uniform access to all shared memory locations
- This architecture still allows all processor to memory module closet to it but its local memory more quickly than other. Hence, memory access time are non-uniform.
- NUMA computers be not be SMP.
- NUMA has better performance than UMA
- e.g.:- CRAY T3E



COMA

- Cache only memory access architecture.
- Each processor's local memory is treated as a cache.
- When processor request data that is not in cache (local memory), the system loads that data into local memory as part of memory operation.
- E.g.:- KSR1 and KSR2
  DDM (Data Diffusion machine)

10.3 Communication in multiprocessor system

It is a key factor in determining system overall performance.

Two ways of communication: -

i) Fixed connections
ii) Reconfigurable connections

i) Fixed connection

- The connection that never change
- Inflexible for some system but sufficient for many systems.
- Less costly than reconfigurable connections
- e.g.: - system with shared bus
- clustering is one of the fixed connection topology.

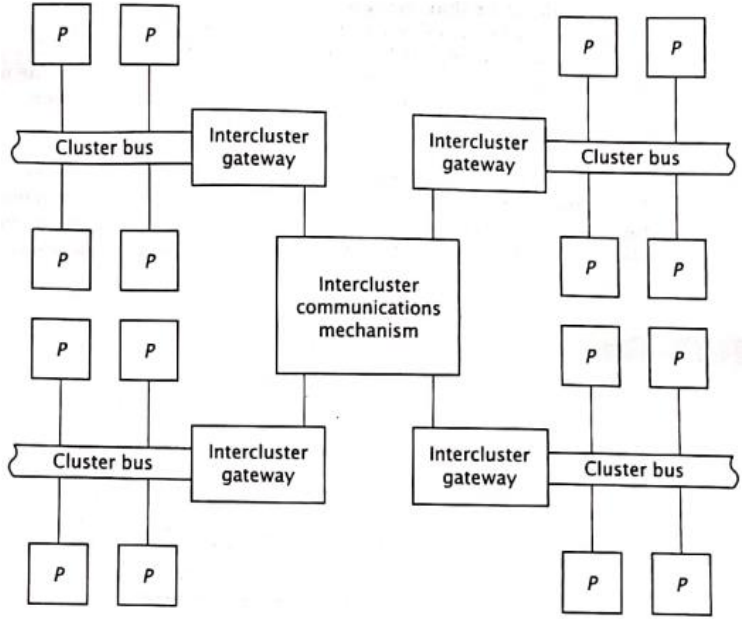


Fig: - A 16 processor multiprocessor that uses clustering

ii) Reconfigurable connection

- In this, there is an ability of reconfiguring (changing) connections between processors and memory, I/O devices and other processor can allow it to meet the needs of individual tasks and maximize system performance.
- Crossbar switch mechanism is used.

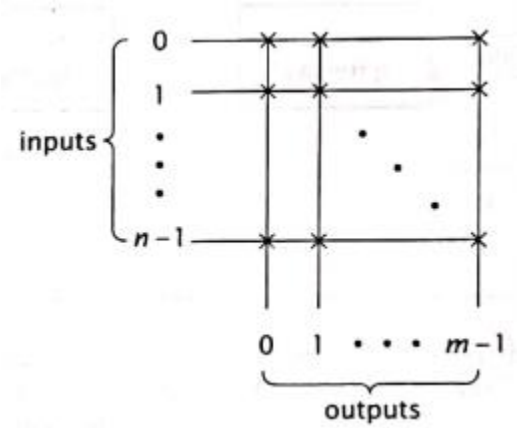
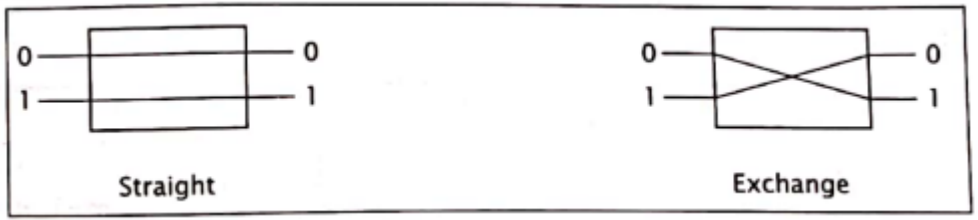


Fig: - (A n*m crossbar switch)

- **Useful when all task does not require same processing resources.**



10.5 Memory organization in Microprocessor system.

- Shared Memory.
- Cache Memory

1) Shared memory:

- Both UMA and NUMA architecture uses shared memory processors can access shared programs and data.
- Processors can also use shared memory to communicate each other through message passing.
- In addition to message passing, the OS uses shared memory to store information about its current state which can be access by processor.
- In this, at first it appears like all processors try to access a single shared memory module and that only one can be successful at given time. In practise, shared memory is partitioned into several modules (M1, M2, M3) all of which can be accessed simultaneously.

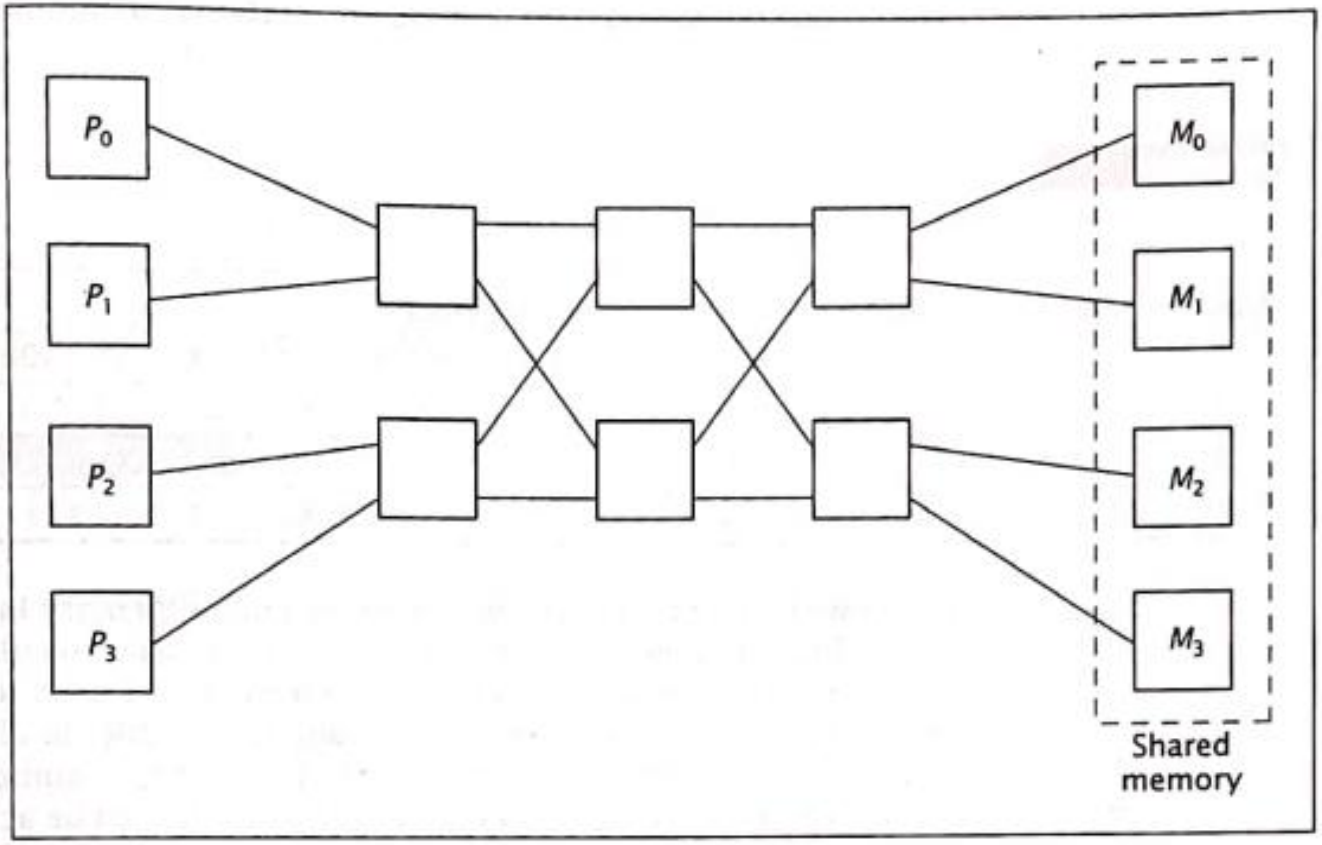


Figure: UMA system with shared memory

ii) Cache Coherence

- It is a problem in multiprocessor system.
- Multiprocessor have individual cache for each processor.
- This can lead to problem when two or more caches hold the value of same memory location simultaneously.
- As one processor stores a value to that location in its cache, the other cache will have an invalid value in its location.
- The extra writes to main memory is needed which decreases system performance.
- This is cache coherence.

Action	Cache 0	Cache 1	Cache 2	Cache 3
Initial	1234: 56	1234: 56	1234: 56	1234: 56
Processor 0 updates 1234H	1234: 78	1234: 56	1234: 56	1234: 56
Processor 3 reads 1234H				Reads 56H instead of 78H

- A system with 4 processor, each of which has write-back cache.
- Assume all 4 cache have loaded content of shared memory location1234H which is 56H.
- Then one of the processors, processor 0 writes value 78H to this location in its cache.
- But caches 1,2and 3 do not have correct value.
- If one of the other processors reads then it will read the old incorrect value 56H instead of correct value 78H.

Assignment:

1. What is cache coherence? Describe how cache coherence can be dealt?
2. explain in brief about MESI protocol.

Exam questions:

1. What is cache coherence? Describe how cache coherence can be dealt? List them. Also explain in brief about MESI protocol. [PU 2017 Fall]
2. How are memory organized in multiprocessor system? Illustrate with suitable diagrams. [PU 2018 Spring]
3. What is Flynn's Taxonomy? Describe in brief. Also explain in brief about cache coherence. [PU 2017 Spring]
4. Describe in detail about different types of memory organization used in multiprocessor systems. [PU 2017 Spring]
5. How are memory organized in multiprocessor system? Illustrate with suitable diagrams. [PU 2018 Spring]

End of chapter 10