II yr–CSE
IV SEM.

UNIT – IV COMPUTER ARCHITECTURE

Parallelism.

① What is parallel processing? Explain its challenges..

Parallel Processing :-

⇒ It is a process of executing operations on multiple processors concurrently.

⇒ In this larger problems are divided into smaller problems, which are then executed simultaneously.

⇒ If speed of execution is faster, then a pgm is said to be efficient.

⇒ A efficient program executed in parallel order takes less time.

Parallel processing challenges :-

The two main challenges which are faced by parallel processors are,

(i) Limited parallelism

(ii) High communication cost (Long-latency remote communication).

(i) Limited Parallelism :-

→ If the level of parallelism is very less in programs, then it is very difficult to attain higher speedups.

i.e the speed of ||lel processors is reduced.

→ To acheive greater speedup using Amdahl's law,

$$S = \frac{1}{\frac{Fe}{Se} + (1-Fe)}$$

S ⇒ speedup

Fe ⇒ fraction of enhanced mode (i.e time spent in enhanced mode)

Se ⇒ Speedup in enhanced mode (No of processor)

Ex :-

Suppose if a speedup of 80 is to be acheived by using 100 processors, then 0.25% of original processing can be made sequential, which is computed in following manner.

$$S = \frac{1}{\frac{Fe}{se} + (1-Fe)}$$

$$80 = \frac{1}{\frac{Fe}{100} + (1-Fe)}$$

$$\Rightarrow 80\left(\frac{Fe}{100} + (1-Fe)\right) = 1$$

$$\frac{80Fe}{100} + 80(1-Fe) = 1$$

$$\Rightarrow 80 - 79.2\ Fe = 1$$

$$-79.2\ Fe = 1 - 80.$$

$$-79.2\ Fe = -79$$

$$79.2\ Fe = 79.$$

$$Fe = \frac{79}{79.2}$$

$$\Rightarrow Fe = 0.9975 = 99.75\%.$$

$$\therefore Fe = 100 - 99.75 = 0.25.$$

→ Linear speedup can be acheived by executing the entire pgm parallely.

↳ However, practically such execution is not possible instead less than the entire processor's complement is used, while executing pgm in ||el or enhanced mode.

→ Limited parallelism issue can be solved by s/w. that consists new alg, which enhance the performance.

(ii) ⇒ The second main hurdle in parallel processing is larger latency to access remote data.

As parallel processors continuously need data, the latency to access data should be small.

But accessing remote data in a shared memory system costs around 100 to 1000 clock cycles.

The access latency is depend on following factors:

(1) Inter connection Network:

⇒ If the N/w is simple, it is easy to refer the memory location, access & retrieve.

⇒ If it is complex, more time needs to be spent for access

(2) Scale of Multiprocessor.

⇒ If No of Multiprocessor increased, then the size of N/w increase making it diffcult to retrieve data.

(3) Type of communication :- It also plays an important role, which in turn is depend on the size of N/w.

=> The problem of long latency, for remote communication can be solved by adjusting architecture & s/w.

↳ we can decrease the frequency of remote memory access by employing either h/w approach or (by caching shared data) s/w approach (by reconstructing pgm).

---

② What is ILP? Explain Limitation in ILP (Instruction Level Parallelism)

→ ILP is a form of parallelism that is identified and exploited by processor hardware i.e., Compiler.

→ It is constructed using the technique which is used for executing the parallel instr.

→ It uses compile technique like s/w pipeline, loop unrolling & trace scheduling.

→ These techniques can be used only when behaviour of branches can be predicted.

=> The ILP does not support redundancy.

↳ It used the techniques like register renaming, alias analysis to detect & exploit the dependencies.

=> The ILP supports functional units, reservation stations and pipeline stages in order to execute the ||lel instr.

⇒ The ILP is a technique that supports overlapping of operations that are being processed.

↳ The operation can either be addition, multi, load or store.

⇒ This technique is designed for boosting the speed of the system.

⇒ In ILP multiple operation will execute simultaneously, resulting higher execution rate.

Limitation Of ILP :-

① Hardware Terminology Data Hazards.

→ These hazards include Write - After - Read (WAR), Write - After - Write (WAW) and Read - After - Write (~~WAR~~ RAW) hazards which are resultant of ||el execution principle employed in ILP technique.

→ These hazards can be eliminated by register renaming they still exist in memory utilization.

→ The WAW & WAR hazards raised because, stack allocation takes place & ~~thes~~ procedure may reuse the mem loc of previously executed procedure on the same stack.

② Window size :

⇒ In ILP size of window is to be maximum, because in ILP scenario, the functional units are pipelined.

→ The window must contain all the memory references that are waiting on a cache miss.

→ If window size is reduced then the parallelism employed will significantly degrade.

③ Dependences :-

→ The dependences among the instr must be removed for successful ILP.

→ These dependences can be either name dependency, data true dependency, control dependency or resource dependency.

⇒ (4) Data Flow unit :-

⇒ The ILP can also be affected by implementation of value prediction scheme because, there exists a possibility of predicting the values wrongly.

⇒ The value prediction must be accurate because, inaccuracy in prediction will ultimately result in an inappropriate speculation & recovery.

(3) Explain with diagrammatic illustration Flynn's classification.

Ans:-

⇒ Michael J. Flynn proposed four different computer organizations based on the instr and data manipulations in order to accomplish parallel processing.

⇒ These four organizations are

(1) SISD - Single Instr stream, single data stream.
(2) SIMD - single instruction stream, multiple data "
(3) MISD - Multiple " " , Single " "
(4) MIMD = " " " , Multiple " "

In general terms the word stream refer to array of entities.

↔ The word instr stream specifies an array of large NO of instr

↳ data stream refers to those resources of data, which are essential while executing the given instr stream.

(1) SISD:-

⇒ An SISD computing system is a uniprocessor machine which is capable of executing single instr, operating on a single data stream.

⇒ In SISD machine, Instr are processed in a sequential manner.

↳ and computers adopting this model are popularly called sequential computers.

→ Most conventional computers have SISD archite

⇒ All instr and data to be processed have to be stored in primary memory.

→ The speed of processing element in SISD model is limited (dependent)

↳ by the rate at which the computer can transfer infor internally.

eg: IBM PC, workstation

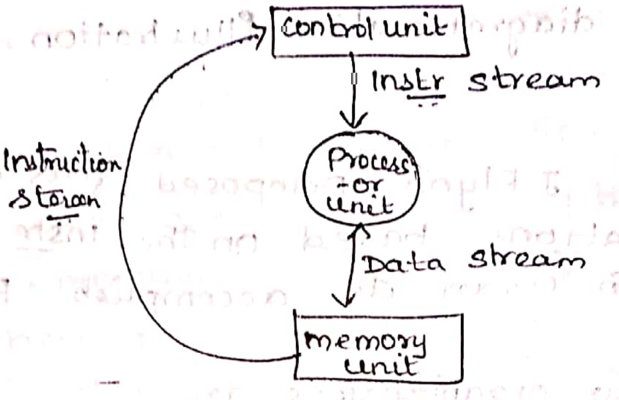


Fig: SISD Organization

(2) SIMD Systems :-

⇒ It consists of single control unit that governs an array of processors which are directly connected to multiple memory modules.

↳ These memory modules together form a single large mem unit.

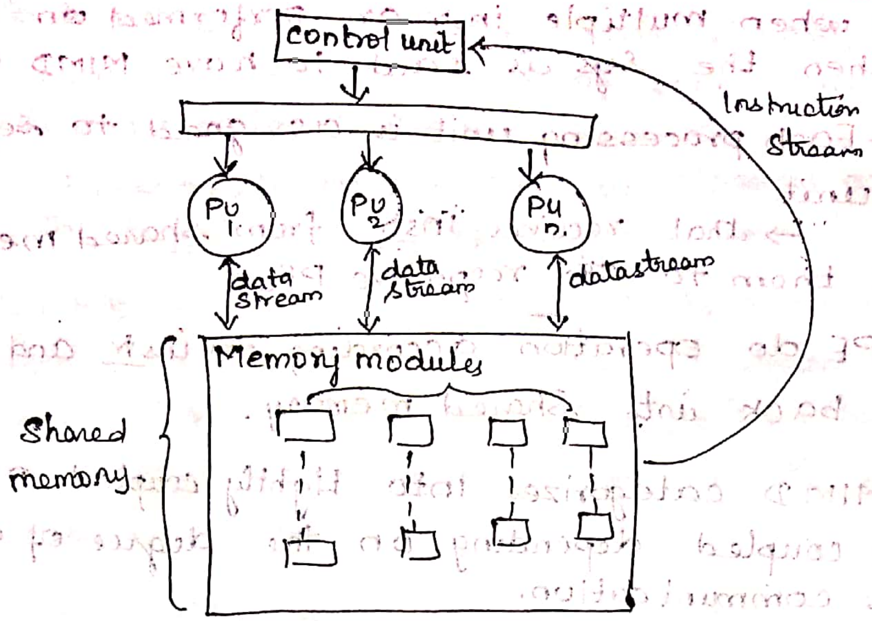
⇒ It is also referred to shared memory unit.

⇒ SIMD model suited for scientific computing since they involve lots of vector and matrix operations.

⇒ Various processor unit (PU) receive broadcasted msg from the single control unit, they work on diff data streams.

eg:- Thinking Machines CM-2, DAP, Illiac IV and STARAN.

Fig: SIMD Organization



(3) MISD System:-

⇒ It consists large NO of control lines with equal NO of processors.

⇒ These processor share single unit it is called Shared mem unit.

⇒ corresponding instruction streams gets activated from their respective mem modules, forms an i/p to their associated control unit.

⇒ The first processor receives data stream from the shared memory unit

↳ & o/p of this processor will be i/p to next consecutive processor & so on.

⇒ Finally last processor connected to memory.

⇒ It is considered as imperical architecture.

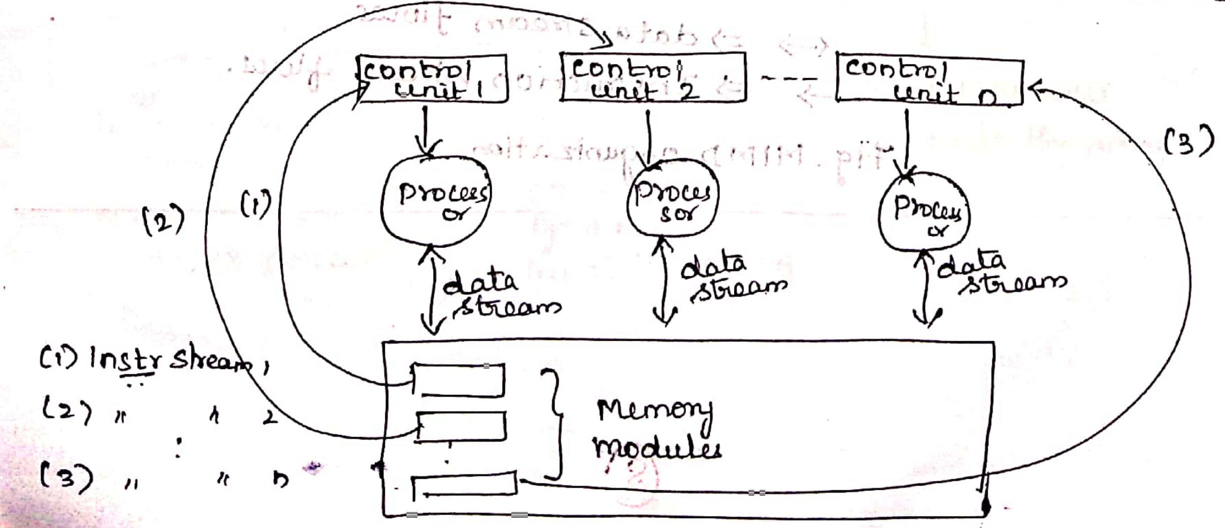


Fig: MISD Organization

(4) MIMD :-

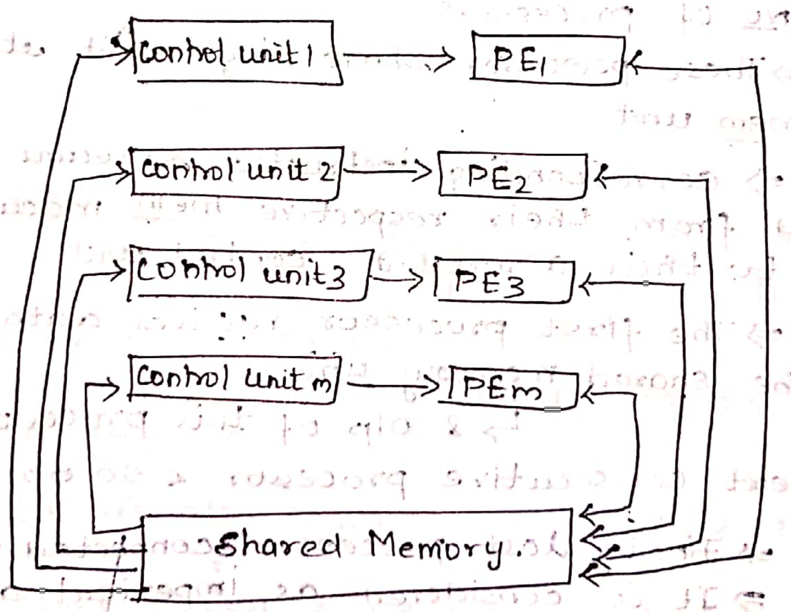
=> when multiple instr are performed on multiple data then the sys is said to have MIMD architecture.

=> Each processing unit is assigned to seperate control unit
↳ that receives instr from shared memory & passes them to the respective PEs.

=> PE do operation according to instr and stores result back into shared memory.

=> MIMD categorizes into tightly coupled or loosely coupled depending on the degree of inter-process communication.

If the degree of communication is high then the MIMD architecture is called tightly coupled.



<-> => data stream flows

-> => Instruction stream flows.

Fig: MIMD organization.

(4) Explain in detail about hardware multithreading.

⇒ A multithreading processor is able to pursue two or more threads of control in parallel within the processor pipeline.

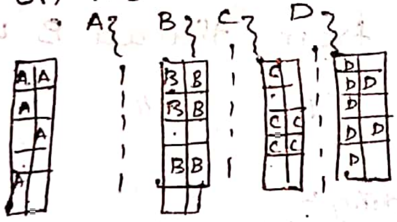
↳ Contexts of two or more threads are often stored in seperate on-chip register sets.

⇒ CMP (chip MultiThreading), is a processor technology that allows multiple hardware threads of execution (also known as strands) on the same chip, through multiple cores per chip, multiple threads per core or combination of both.

Hardware Multithreading techniques/types.

[1] Multiple cores per chip

CMP (chip multiprocessing (or) Multicore), is a processor technology that combines multiple processors (a.k.a cores) on the same chip.



chip Multiprocessor.

[2] Multiple Threads per Core.

Types

Vertical multithreading

- Interleaved (or) Fine Grained multithreading
- Blocked or Coarse-grained multithreading

horizontal Multithreading.

- Simultaneous multithreading (SMT)

① Interleaved Multithreading (Fine Grained)

⇒ The thread are switched on each instruction.

⇒ The thread is switched to other when running thread encounters stall.

⇒ This thread remove all data dependency stalls from the execution pipeline.

⇒ It is similar to pre-emptive multitasking used in OS.

⇒ It is first called as Barrel processing

⇒ also called as

Interleave (or)
preemptive (or)
Fine Grained Multithreading (or)
Time - sliced "

eg :-

Cycle i+1 : An instr from thread B is issued.
cycle i+2 : " " " " C " "

② Block Multi-Threading :-

⇒ Thread switch to another thread when costly stall encountered.

⇒ Occurs when one thread runs until it is blocked by an event that normally would create a long latency stall.

⇒ Such stall might be a cache miss that has to access off-chip memory.

⇒ Instead of waiting for the stall to resolve, a thread processor would switch execution to another thread that was ready to run.

=> Only when previous thread receives data it can be placed back on the list of ready-to-run threads

=> It is similar to cooperative multitasking used in OS.

=> It is also called Blocked or cooperative or coarse grained multithreading.

eg:-

1. cycle i : instr j from thread A is issued
2. cycle i+1 : " j+1 " " " " "
3. cycle i+2 : " j+2 " " " " "

load instr which misses in all caches

4. cycle i+3 : thread scheduler invoked, switches to thread B.
5. " i+4 : Instr k from thread B is issued
6. " i+5 : " k+1 " " " " "

⑧ Simultaneous Multithreading : SMT.

↳ It is a technique for improving the overall efficiency of superscalar CPU with h/w multithreading.

↳ Multiple threads utilizes the resources properly.

↳ It uses big & deep pipelining for executing multiple instr parallely across multiple threads.

eg:

cycle i : Instr j & j+1 from thread A;
Instr k from thread B all issued simultaneously

cycle i+1 : Instr j+2 from thread A;
Instr k+1, k+2 " " B;
Instr m from " C; issued simultaneously.

cycle i+3 : instr j+3 from thread A;
" m+1, m+2 " " C issued simultaneously

(5) Write a note on Multi core porocessor.

Ans :- A Multi-core processor is one which combines two or more independent processors into a single package, often a single Integrated circuit (IC)

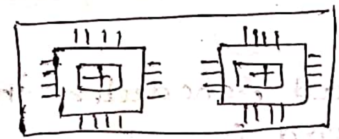
→ also called core Multi processor (CMP) or Single chip multiporocessor.

→ Multi core are made to core in parallel to acheive better performance

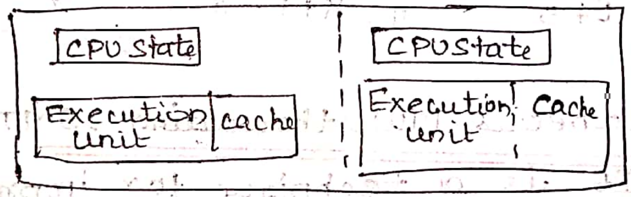
So

More core = Better performance

Structure of Multi-core Processor :



⇒ A simple multi-core architecture consists of two independent working processors.



→ Each core or CPU consists of its own set of execution unit & cache.

→ There are other multi-core architecture

1. Multi-core with shared memory.

2 - " " " hyper threading technology.

Downside of multiprocessor :-

(1) All the pgms might not run effectively in a multi-core system. Sometimes it even might result in loosd of performance.

(2) parallelizing the pgm is not simple task.

(3) Speed of the system depends on what the users is doing with it

(4) Multi core Processor are very expensive.

CMPs Come in Multiple flavours:

① two processor (dual core)
② Four " (quad ")
③ Eight " (octa ")

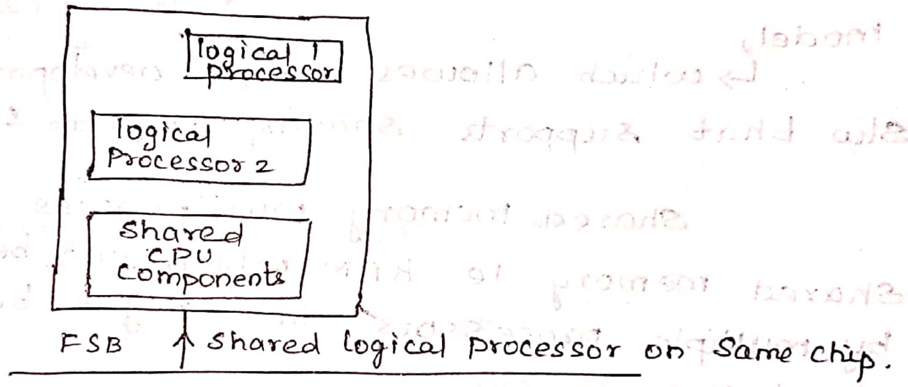
Three common configuration:-

Configuration 1:-

→ uses hyperthreading. Hyperthreading processor allows more or more threads to execute on single chip.

→ In hyperthreaded package the multiple processor are logical instead of physical.

→ Hyperthreading allows the processor to present itself to os as complete multiple processor when infact there is a single processor running multiple thread.

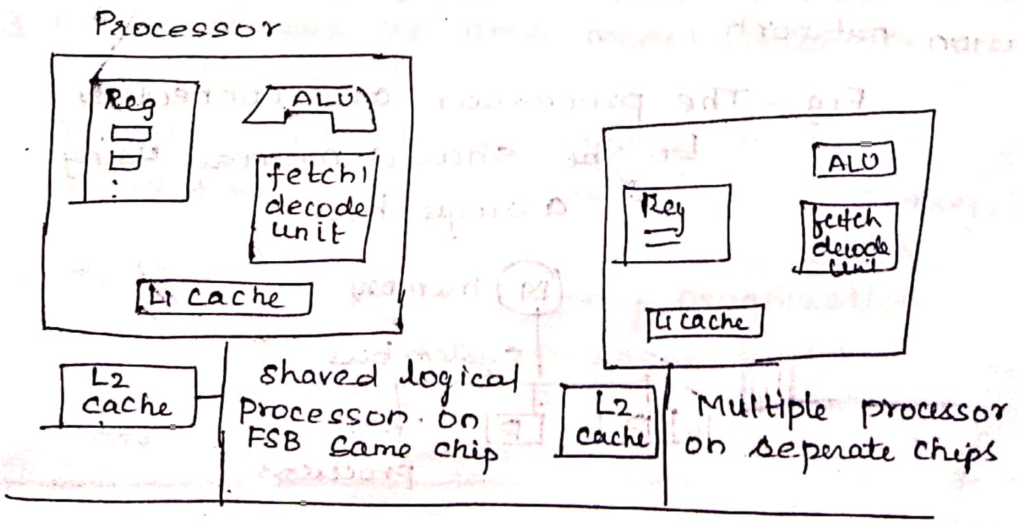


Shared logical processor on Same chip.

Configuration 2:-

→ It is a classic multiprocessor.

→ Each processor is on seperate chip with its own hardware.



Shared logical Processor on same chip

Multiple processor on seperate chips

configuration 3 :

↳ Represent current trend in multiprocessors.
↳ It provides complete processor on single chip.

Advantage Of Multicore porocessor

(1) It porovides great energy efficiency.
(2) It provides high performance
(3) It provides absolutes reliability & robustness.
(4) It helps in executing the given, tasks with fewer computers and processor.

---

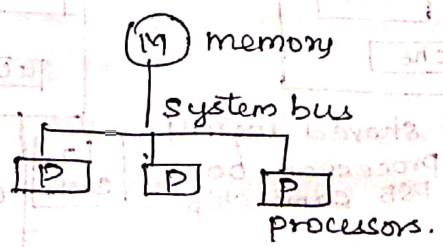
⑥ Discuss about shared Memory Multiprocessor. based on memory access latency.

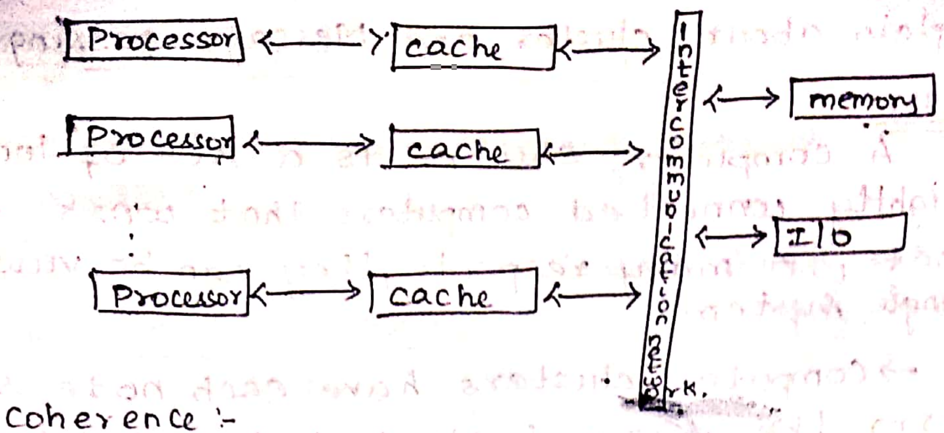
⇒ shared memory processors are popular due to their simple and general programming model,
↳ which allowes simple development of parallel s/w that supports sharing of code &data.

Shared memory multiprocess uses a common shared memory ie . RAM which can be accessed by multiple processors that might be carrying local memory.

All the processors in the shared memory architecture can access the same address space of a common memory through an intercommunication network.

Fig :- The processor are connected to the shared memory using a single bus





Cache coherence :-

=> In order to increase speed, a small memory known as cache is introduce.

↳ processor searched data from cache.

Each processor have local memory as a cache memory.

=> To write data in these memories, multiprocessor system adopts two types of mechanism.

① write through
② write back.

Shared Memory Multiprocessors Model:-

(1) Uniform Memory access (UMA)
(2) Non - " " " (NUMA)
(3) cache-only Memory architecture (COMA)
(4) Heterogeneous system architecture.

UMA :- ① UMA is a shared memory arch used in ||el computers.
② All processors in UMA model share the physical memory uniformly.

NUMA :- In NUMA, a processor can access its own local memory faster than non-local memory.

COMA :- It is a computer memory organization for used in multiprocessor in which local memories (typically DRAM) at each node are used as cache.

(7) Explain about cluster and Message passing.

Cluster :-

A computer cluster is a set of loosely or tightly connected computers that work together so that, in many respects, they can be viewed as a single system.

→ Computer clusters have each node set to perform the same task, controlled & scheduled by h/w.

→ The components of a cluster are usually connected to each other through fast local area network,

↳ with each node (computer used as a server) running its own instance of an os.

↳ In Most circumstances, all of the nodes use the same h/w and same os.

⇒ Clusters are usually deployed to improve performance & availability over that of a single computer.

Benefits :-

→ Clusters are primarily designed with performance in mind, but installations are based on many other factors.

↳ fault tolerence (the ability for a system to continue working with a malfunctioning node). allows for scalability, & in high performance situations, low frequency of maintenance routines, resource consolidation (RAID) & centralized mgmt.

Adv :-

→ Include enabling data recovery in the event of a disaster.

→ & provide parallel data processing & high ~~performance~~ processing capacity.

Message passing in Computer Cluster.

→ Msg passing is an inherent element of all computer cluster.

→ Msg passing in computer clusters built with commodity servers & switches is used by virtually every internet service.

→ As the NO of nodes in the cluster increases, the rapid growth in the cap complexity of the communication subsystem makes msg passing delays over the interconnect a serious performance issue in the execution of ||el programs.

Approaches to Msg Passing :-

(1) PVM, the parallel virtual Machine

(2) MPI, the message Passing Interface.

PVM ⇒ It provides a set of s/w libraries that allow a computing node to act as a "parallel virtual m/c.

⇒ It provides run-time environment for msg-passing, task and resource mgmt, fault notification & must be directly installed on every cluster node.

⇒ PVM can be used by user pgm written in C, C++ etc.

MPI ⇒ MPI Specification gave rise to specific implementation.

⇒ It typically use TCP/IP & socket connection

⇒ MPI is now widely available communication model that enables parallel programs to be written in larger. like C, python etc.