Lecture notes for Parallelism in Computer Architecture

Subject : Computer Architecture

Prepared by:Dr.J.Vinothkumar

TableofContents **PageNo.**

Parallelism in Computer Architecture

1. **Aim & objective**:

To make students understand the basic structure and operation of digital computer.

- Learn the concepts of parallel processing, pipelining.
- Understand the architecture and functionality of central processing unit
- Discuss about different types of peripheral devices of computer
- Learn the different types of serial communication techniques.
- Explain different pipelining processes.

2. **Prerequisite** : Digital System Design, Microprocessors & Microcontrollers

3. Pre Test- MCQtype

1. Execution of several activities at the same time.
a) Processing
b) parallel processing
c) serial processing
d) multitasking

Answer: **parallel processing**

2. A parallelism based on increasing processor word size.
a) Increasing
b) Count based
c) Bit based
d) Bit level

Answer: **Bit level**

3. The pipelining process is also called as ______
a) Superscalar operation
b) Assembly line operation
c) Von Neumann cycle
d) None of the mentioned

Answer: **Assembly line operation**

4. To increase the speed of memory access in pipelining, we make use of ________
a) Special memory locations
b) Special purpose registers

c) Cache
d) Buffers

Answer: **Cache**

4. Parallelism

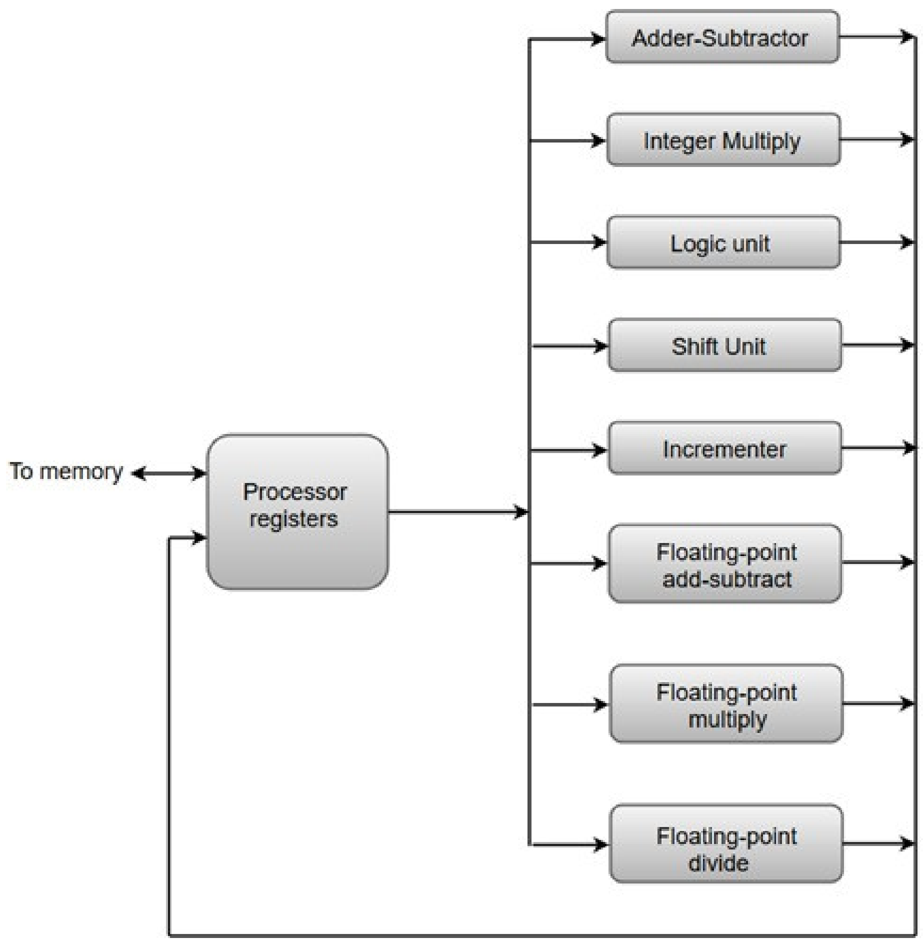
4.1 Introduction

Why Parallel Architecture?

- ✓ Parallel computer architecture adds a new dimension in the development of computer system by using more and more number ofprocessors.
- ✓ In principle, performance achieved by utilizing large number of processors is higher than the performance of a single processor at a given point oftime.

Parallel Processing

- ✓ Parallel processing can be described as a class of techniques which enables the system to achieve simultaneous data-processing tasks to increase the computational speed of a computersystem.
- ✓ A parallel processing system can carry out simultaneous data-processing to achieve faster executiontime.
- ✓ For instance, while an instruction is being processed in the ALU component of the CPU, the next instruction can be read frommemory.
- ✓ The primary purpose of parallel processing is to enhance the computer processing capability and increase itsthroughput,
- ✓ A parallel processing system can be achieved by having a multiplicity of functional units that perform identical or different operationssimultaneously.
- ✓ The data can be distributed among various multiple functionalunits.
- ✓ The following diagram shows one possible way of separating the execution unit into eight functional units operating inparallel.
- ✓ The operation performed in each functional unit is indicated in each block if the diagram:



- ✓ The adder and integer multiplier performs the arithmetic operation with integer numbers.
- ✓ The floating-point operationsare separated into three circuits operating in parallel.
- ✓ The logic, shift, and increment operations can be performed concurrently on differentdata.
- ✓ All units are independent of each other, so one number can be shifted while another number is beingincremented.

- ✓ Parallel computers can be roughly classified according to the level at which the hardware supports parallelism, with multi-core and multi-processor computers having multiple processing elements within a singlemachine.
- ✓ In some cases parallelism is transparent to the programmer, such as in bit-level or instruction-levelparallelism.
- ✓ But explicitly parallel algorithms, particularly those that use concurrency, are more difficult to write than sequential ones, because concurrency introduces several new classesofpotentialsoftwarebugs,ofwhichraceconditionsarethemostcommon.
- ✓ <u>Communication and synchronization between the different subtasks</u> are typically some of the greatest obstacles to getting optimal parallel programperformance.

Advantages of Parallel Computing over Serial Computing are as follows:

1. It saves time and money as many resources working together will reduce the time and cut potentialcosts.
2. It can be impractical to solve larger problems on SerialComputing.
3. It can take advantage of non-local resources when the localresources are finite.
4. Serial Computing 'wastes' the potential computing power, thus Parallel Computing makes better work ofhardware.

Types of Parallelism:

1. **Bit-level parallelism:** It is the form of parallel computing which is based on the increasing processor's size. It reduces the number of instructions that the system must execute in order to perform a task on large-sized data. *Example:* Consider a scenario where an 8-bit processor must compute the sum of two 16-bit integers. It must first sum up the 8 lower-order bits, then add the 8 higher-order bits, thus requiring two instructions to perform the operation. A 16-bit processor can perform the operation with just oneinstruction.
2. **Instruction-level parallelism:** A processor can only address less than one instructionforeachclockcyclephase.Theseinstructionscanbere-orderedand

grouped which are later on executed concurrently without affecting the result of the program. This is called instruction-levelparallelism.

3. **Task Parallelism:** Task parallelism employs the decomposition of a task into subtasks and then allocating each of the subtasks for execution. The processors perform execution of sub tasksconcurrently.
4. **Data-level parallelism (DLP)** – Instructions from a single stream operate concurrently on several data – Limited by non-regular data manipulation patterns and by memorybandwidth

Architectural Trends

- ✓ When multiple operations are executed in parallel, the number of cycles needed to execute the program isreduced.
- ✓ However, resources are needed to support each of the concurrentactivities.
- ✓ Resources are also needed to allocate localstorage.
- ✓ The best performance is achieved by an intermediate action plan that uses resources to utilize a degree of parallelism and a degree oflocality.
- ✓ Generally, the history of computer architecture has been divided into four generations having following basic technologies−
    - Vacuum tubes
    - Transistors
    - Integratedcircuits
    - VLSI
- ✓ Till 1985, the duration was dominated by the growth in bit-levelparallelism.
- ✓ 4-bit microprocessors followed by 8-bit, 16-bit, and soon.
- ✓ To reduce the number of cycles needed to perform a full 32-bit operation, the widthofthedatapathwasdoubled.Lateron,64-bitoperationswereintroduced.
- ✓ The growth in **instruction-level-parallelism** dominated the mid-80s tomid-90s.
- ✓ The RISC approach showed that it was simple to pipeline the steps of instruction processingsothatonanaverageaninstructionisexecutedinalmosteverycycle.
- ✓ Growth in compiler technology has made instruction pipelines more productive.
- ✓ In mid-80s, microprocessor-based computers consistedof

- An integer processingunit
- A floating-pointunit
- A cachecontroller
- SRAMs for the cachedata
- Tagstorage

✓ As chip capacity increased, all these components were merged into a singlechip.

✓ Thus, a single chip consisted of separate hardware for integer arithmetic, floating point operations, memory operations and branchoperations.

✓ Other than pipelining individual instructions, it fetches multiple instructions at a time and sends them in parallel to different functional units whenever possible. This type of instruction level parallelism is called **superscalarexecution**.

FLYNN‘S CLASSIFICATION

✓ Flynn's taxonomy is a specific classification of parallel computer architectures that are based on the number of concurrent instruction (single or multiple) and data streams (single or multiple) available in thearchitecture.

✓ The four categories in Flynn's taxonomy are thefollowing:

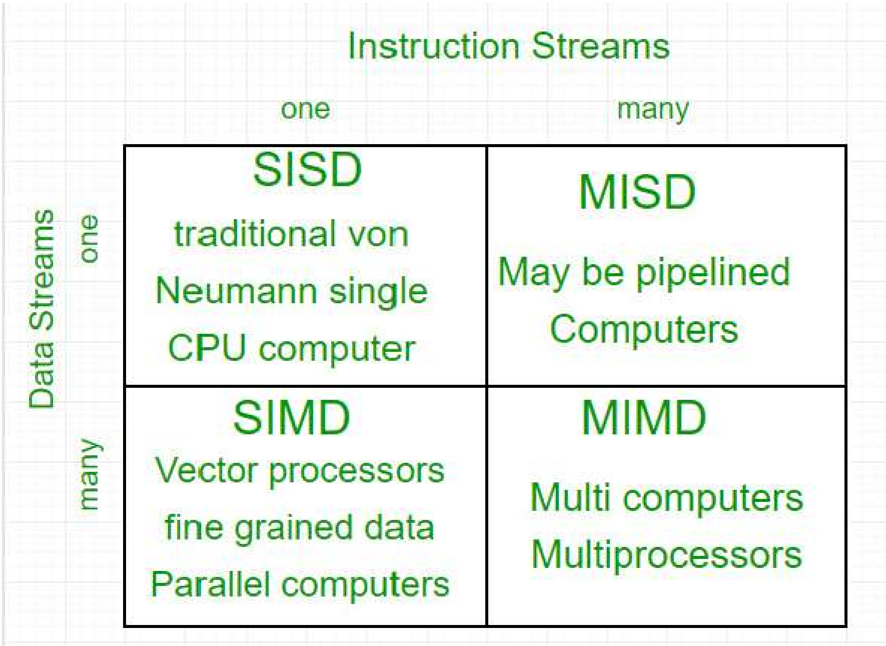
1. (SISD) single instruction, singledata
2. (SIMD) single instruction, multipledata
3. (MISD) multiple instruction, singledata
4. (MIMD) multiple instruction, multipledata

✓ Instruction stream: is the sequence of instructions asϒexecuted by themachine

✓ Data Stream is a sequence of data including input, or partialϒor temporary result, called by the instructionStream.

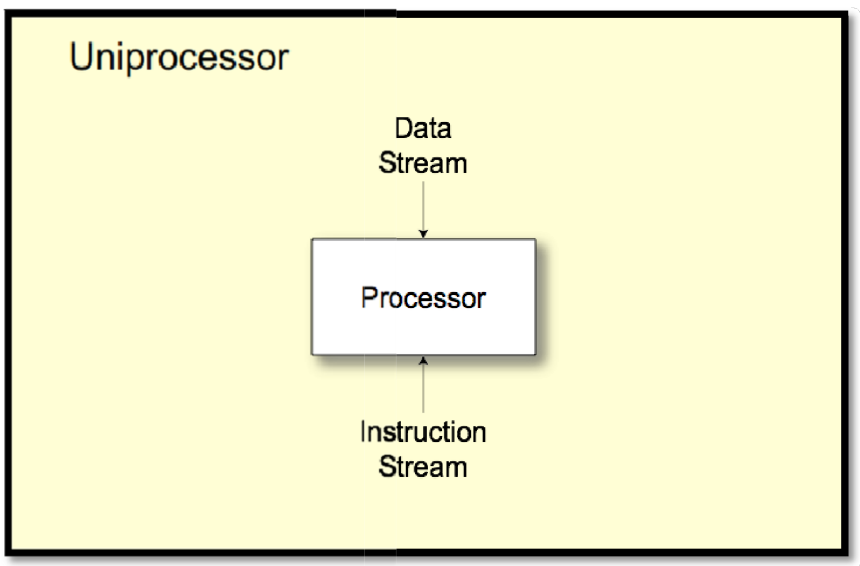
✓ Instructions are decoded by the control unit and then ctrl unit send the instructions to the processing units for execution.•

✓ Data Stream flows between the processors and memory bidirectionally.



SISD

An SISD computing system is a uniprocessor machine which is capable of executing a single instruction, operating on a single datastream.

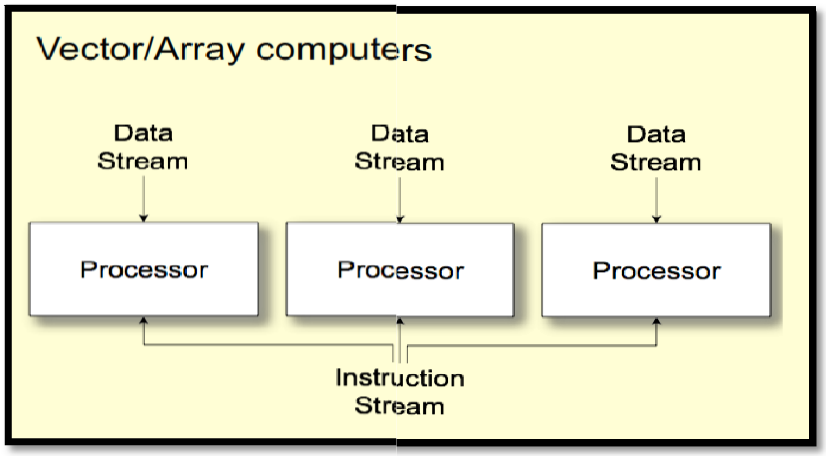


- ✓ In SISD, machine instructions are processed in a sequential manner and computers adopting this model are popularly called sequentialcomputers.
- ✓ Most conventionalcomp ι ters have SISD architecture. All the instructionsand data to be processed have to be stored in primary memory.
- ✓ The speed of the processing element in the SISD model is limited(dependent) by the rate at which the computer can transfer informationinternally.

- ✓ Dominant representative SISD systems are IBM PC,workstations.

SIMD

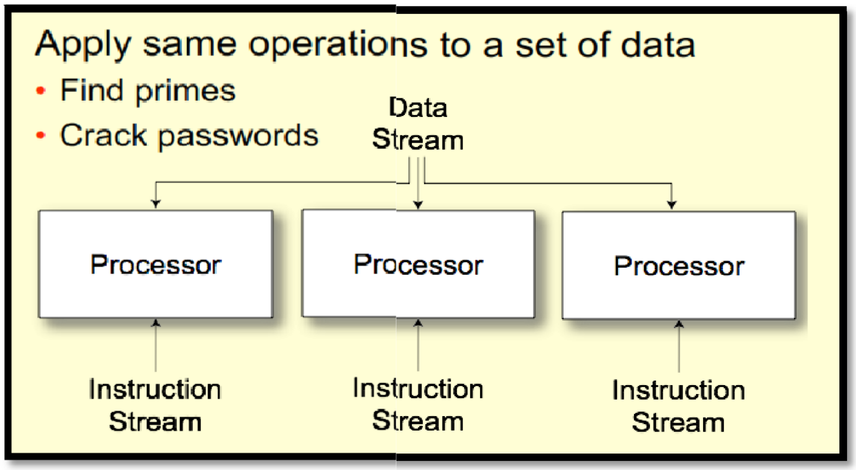
- An SIMD system is a multiprocessor machine capable of executing the same instruction on all the CPUs but operating on different datastreams



- ✓ Machines based on an SIMD model are well suited to scientific computing since they involve lots of vector and matrixoperations.
- ✓ So that theinformation can be passed to all the processing elements (PEs) organized data elements of vectors can be divided into multiple sets(N-sets for N PE systems) and each PE can process one dataset.
- ✓ Dominant representative SIMD systems is Cray's vector processingmachine.

MISD

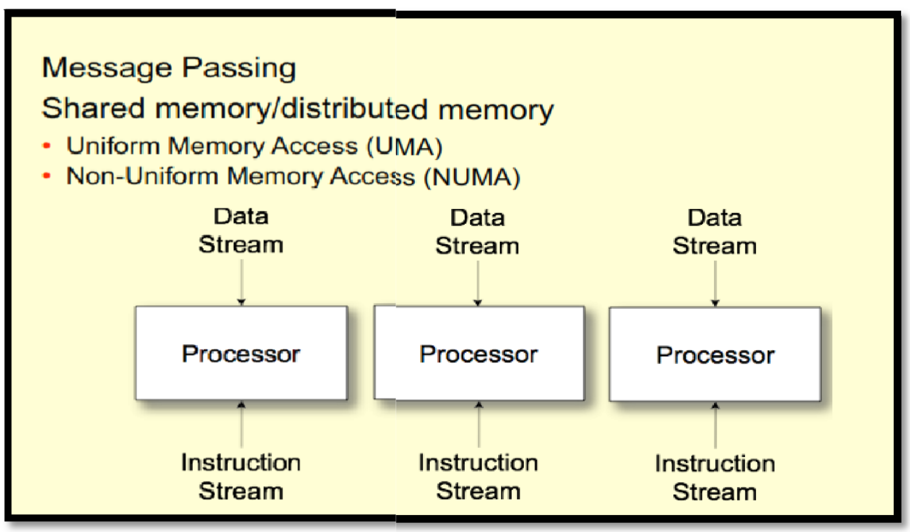
- ✓ An MISD computing system is a multiprocessor machinecapable of executing different instructions on different PEs but all of them operating on the same dataset .



- ✓ The system performs different operations on the same data set. Machines built using the MISD model are not useful in most of the application, a few machines are built, but none of them are availablecommercially.

MIMD

- ✓ An MIMD system is a multiprocessor machine which is capable of executing multiple instructions on multiple datasets.



- ✓ Each PE in the MIMD model has separate instruction and data strea s; therefore machines built usingthis n odel are capable to any kind ofapplication.
- ✓ Unlike SIMD and MISD machines, PEs in MIMD mac ines work asynchronously.
- ✓ MIMD machines arebroa c ly categorized into

- **shared-memoryMIMD** and
- **distributed-memoryMIMD**

based on the way PEs are coupled to the main memory.

In the **shared memory MIMD** model (tightly coupled multiprocessor systems), all the PEs are connected to a single global memory and they all have access to it. The communication between PEs in this model takes place through the shared memory, modification of the data stored in the global memory by one PE is visible to all other PEs. Dominant representative shared memory MIMD systems are Silicon Graphics machines and Sun/IBM's SMP (SymmetricMulti-Processing).

In **Distributed memory MIMD** machines (loosely coupled multiprocessor systems) all PEs have a local memory. The communication between PEs in this model takes place through the interconnection network (the inter process communication channel, or IPC). The network connecting PEs can be configured to tree, mesh or in accordance with therequirement.

VECTORARCHITECTURES

- ✓ A multithreaded CPU is not a parallel architecture, strictly speaking; multithreading is obtained through a single CPU, but it allows a programmer to design and develop applications as a set of programs that can virtually execute in parallel: namely,threads.
- ✓ Multithreading is solution to avoid waiting clock cycles as the missing data is fetched: making the CPU manage more peer-threads concurrently; if a thread gets blocked, the CPU can execute instructions of another thread, thus keeping functional unitsbusy.
- ✓ Each thread must have a private Program Counter and a set of private registers, separate from otherthreads.
- ✓ In a traditional scalar processor, the basic data type is an n-bitword.

- The architecture often exposes a register file of words, and the instruction set is composed of instructions that operate on individualwords.
- In a vector architecture, there is support of a vector datatype, where a vector is a collection of VL n-bit words (VL is the vectorlength).
- There may also be a vector register file, which was a key innovation of the Cray architecture.
- Previously, vector machines operated on vectors stored in mainmemory.
- Figures 1 and 2 illustrate the difference between vector and scalar data types, and the operations that can be performed onthem.

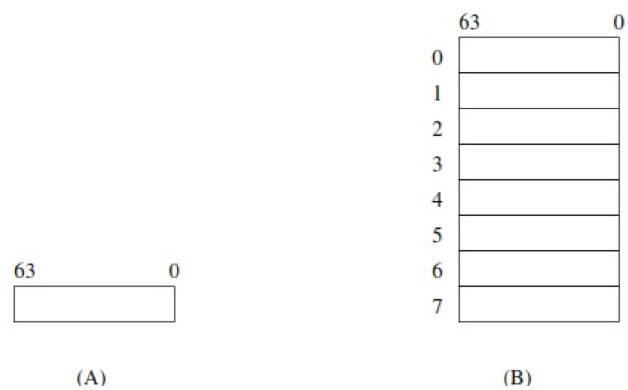


Figure 1: (A): a 64-bit word, and (B): a vector of 8 64-bit words

- Vector load/store instructions provide the ability to do strided and scatter / gather memory accesses, which take data elements distributed throughout memory and pack them into sequential vectors/streams placed in vector/streamregisters.
- This promotes datalocality.
- It results in less data pollution, since only useful data is loaded from the memory system.
- It provides latency tolerance because there can be many simultaneous outstanding memoryaccesses.
- Vector instructions such as VLD and VST provide thiscapability.

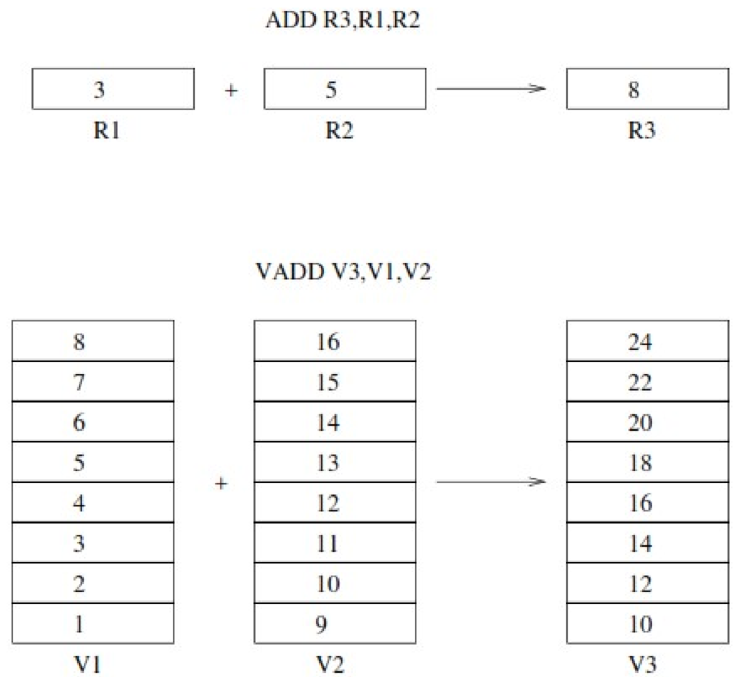


Figure 2: Difference between scalar and vector add instructions

HARDWARE MULTITHREADING

Multithreading

- A mechanism by which the instruction streams is divided into several smaller streams (threads) and can be executed in parallel is calledmultithreading.

Hardware Multithreading

- Increasing utilization of a processor by switching to another thread when one thread is stalled is known as hardwaremultithreading.

Thread

- A thread includes the program counter, the register state, and the stack. It isa lightweight process; whereas threads commonly share a single address space, processesdon't.

Thread Switch

- The act of switching processor control from one thread to another within the same process. It is much less costly than a processorswitch.

Process

- A process includes one or more threads, the address space, and the operating system state. Hence, a process switch usually invokes the operating system, but not a threadswitch.

Types of Multi-threading

1. Fine-grainedMultithreading
2. Coarse-grainedMultithreading
3. SimultaneousMultithreading

Coarse-grained Multithreading

A version of hardware multithreading that implies switching between threads only after significant events, such as a last-level cachemiss.

- This change relieves the need to have thread switching be extremely fast and is much less likely to slow down the execution of an individual thread, since instructions from other threads will only be issued when a thread encounters a costlystall.

Advantage

- To have very fast threadswitching.
- Doesn't slow downthread.

Disadvantage

- It is hard to overcome throughput losses from shorter stalls, due to pipeline start -upcosts.
- Since CPU issues instructions from 1 thread, when a stall occurs, the pipeline must beemptied.
- New thread must fill pipeline before instructions cancomplete.
- Due to this start-up overhead, coarse-grained multithreading is much more useful for reducing the penalty of high-cost stalls, where pipeline refill is negligible compared to the stalltime.

Fine-grained Multithreading

- A version of hardware multithreading that implies switching between threads after every instruction resulting in interleaved execution of multiple threads. It switches from one thread to another at each clockcycle.
- This interleaving is often done in a round-robin fashion, skipping any threads that are stalled at that clockcycle.

To make fine-grained multithreading practical, the processor must be able to switch threads on every clockcycle.

Advantage

- Vertical waste iseliminated.
- Pipeline hazards cannotarise.
- Zero switchingoverhead
- Ability to hide latency within a thread i.e., it can hide the throughput losses that arise from both short and longstalls.
- Instructions from other threads can be executed when one threadstalls.
- High executionefficiency
- Potentially less complex than alternative high performanceprocessors.

Disadvantage

- Clock cycles are wasted if a thread has little operation toexecute.
- Needs a lot of threads toexecute.
- It is expensive than coarse-grainedmultithreading.
- It slows down the execution of the individual threads, since a thread that is ready to execute without stalls will be delayed by instructions from other threads.

Simultaneous multithreading (SMT)

- It is a variation on hardware multithreading that uses the resources of a multiple-issue, dynamically scheduled pipelined processor to exploit thread-level parallelism at the same time it exploitsinstruction level parallelism.
- The key insight that motivates SMT is that multiple-issue processors often have more functional unit parallelism available than most single threads can effectively use.

Since SMT relies on the existing dynamic mechanisms, it does not switch resources every cycle.

- Instead, SMT is always executing instructions from multiple threads, to associate instruction slots and renamed registers with their properthreads.

Advantage

- It is ability to boost utilization by dynamically scheduling functional units among multiplethreads.
- It increases hardware designfacility.
- It produces better performance and add resources to a fine grainedmanner.

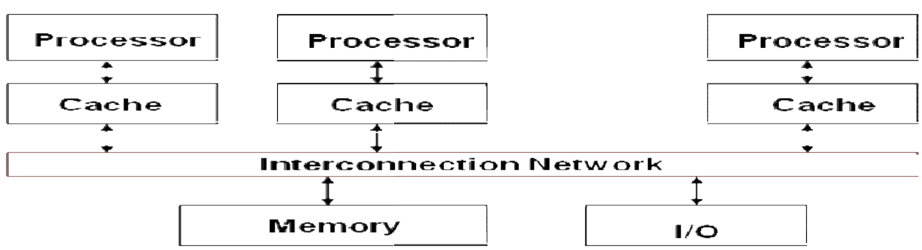
Disadvantage

It cannot improve performance if any of the shared resources are the limiting bottlenecks for theperformance.

MULTICORE AND OTHER SHARED MEMORYMULTIPROCESSORS

Multiprocessor: A computer system with at least two processors

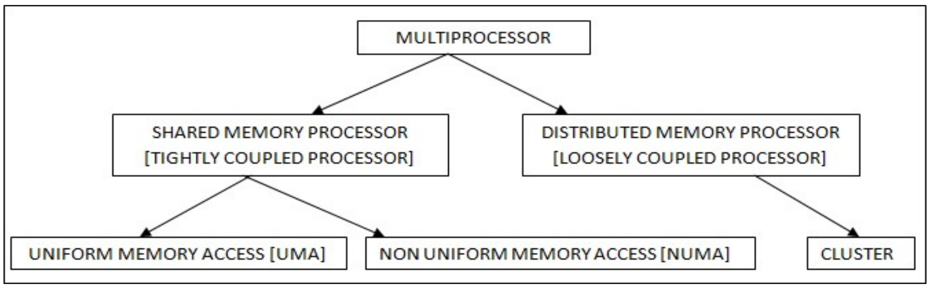
- **Multicore:** More than one processor available within a singlechip.



The conventional multiprocessor system used iscommonly referred as shared memory multiprocessorsystem.

- **Shared Memory Multiprocessor (SMP)** is one that offers the programmer a single is physical address space across all processors which case nearly always the for multicorechips.
- Processors communicate throughshared variables in memory, with all processors capable of accessing any memory location via loads andstores.

- Systems can still run independent jobs in their own virtual address spaces, even if they all share a physical addressspace.
- Use of shared data must be coordinated via synchronization primitives (locks) that allow access to data to only one processor at atime



Shared Memory Multiprocessor System.[Tightly coupled processor]

- The conventional multiprocessor system used iscommonly referred as shared memory multiprocessorsystem.
- Single address space shared by all processors. Because every processor communicates through a shared globalmemory.
- For high speed real time processing, these systems are preferable as their throughput is high as compared to loosely coupledsystems
- In tightly coupled system organization, multiple processors share a global main memory, which may have manymodules.
- Tightly coupled systems use a common bus, crossbar, or multistage network to connect processors, peripherals, andmemories.

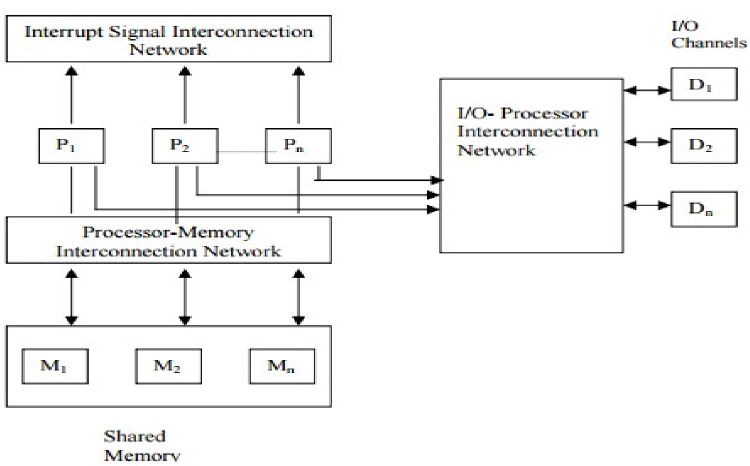
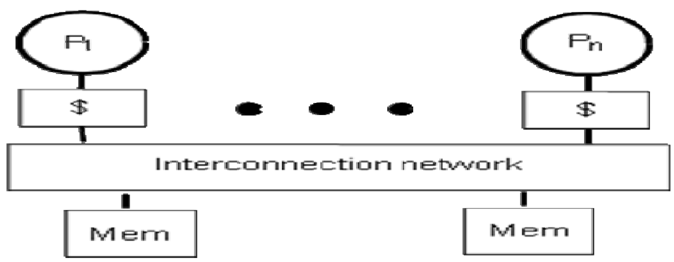


Figure : Tightly coupled system organization

- Two common styles of implementing Shared Memory Multiprocessors (SMP) are,

Uniform memory access (UMA) multiprocessors

- In this model, main memory is uniformly shared by all processors in multiprocessor systems and each processor has equal access time to sharedmemory.
- Thismodelisusedfortime-sharingapplicationsinamulti userenvironment
- Tightly-coupled systems (high degree of resource sharing) suitable for general purpose and time-sharing applications by multipleusers



Physical memory uniformly shared by all processors, with equal access time to all words.

- Processors may have ocal cache memories. Peripherals also shared in some fashion.
- UMA architecture models are of two20types,

Symmetric:

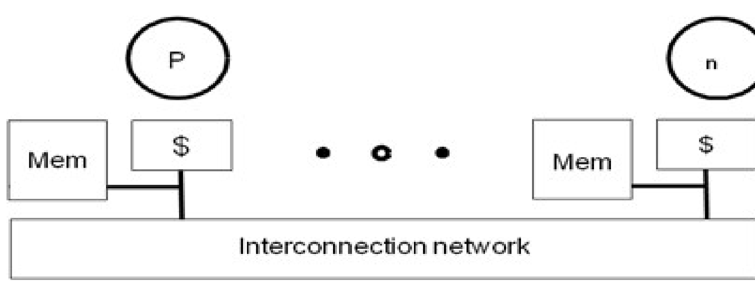
- All processors have equal access to allperipheral devices. All processors are identical.

Asymmetric:

- One processor (master) executes the operating system other processors may be of different types and may be dedicated to specialtasks.

Non Uniform Memory Access (NUMA) multiprocessors

- In shared memory multiprocessor systems, local memories can be connected with every processor. The collections of all local memories form the global memory beingshared.
- In this way, global memory is distributed to all the processors. In this case, the access to a local memory is uniform for its corresponding processor as itisattached to the localmemory.
- But if one reference is to the local memory of some other remote processor, then the access is notuniform.
- It depends on the location of the memory. Thus, all memory words are not accessed uniformly. All local memories form a global address space accessible by allprocessors
- Programming NUMAs are harder but NUMAs can scale to larger sizes and have lower latency to localmemory
- Memory is common to all the processors. Processors easily communicate by means of sharedvariables.
- These systems differ in how the memory and peripheral resources are shared ordistributed
- The access time varies with the location of the memoryword.

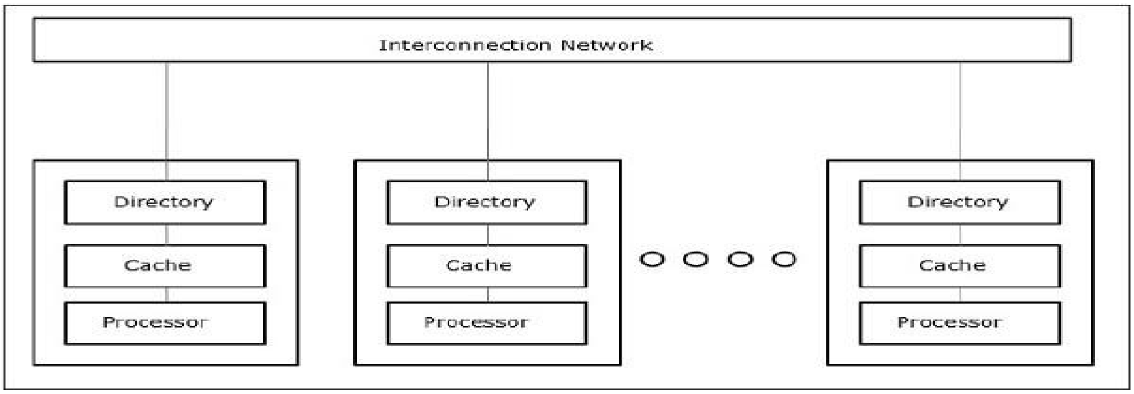


Distributed Memory (NUMA)

- Cache Only Memory Architecture. The COMA model is a special case of the NUMA

Here all the distributed memories are converted to cachememories.

• The local memories for the processor at each node are used as cache instead ofactual



Distributed Memory [Loosely Coupled Systems]

- These systems do not share the global memory because shared memory concept gives rise to the problem of memory conflicts, which in turn slows down the execution ofinstructions.
- Therefore, to alleviate this problem, each processor in loosely coupled systems is having a large local memory (LM), which is not shared by any otherprocessor.
- Thus, such systems have multiple processors with their own local memory and a set of I/Odevices.
- This set of processor, memory and I/O devices makes a computersystem.

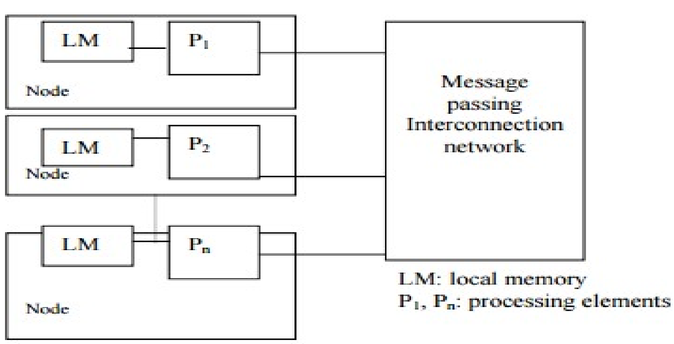


Figure 14: Loosely coupled system organisation

- ✓ Therefore, these systems are also called multi-computersystems.
- ✓ These computer systems are connected together via message passing interconnection network through which processes communicate by passing messages to oneanother.

Since every computer system or node inmulticomputersystemshasa separate memory, they are called distributed multicomputer systems. These are also called loosely coupledsystems.

GPU (Graphics Processing Unit)

- ✓ A graphics processing unit (GPU) is a computer chip that performs rapid mathematical calculations, primarily for the purpose of renderingimages.
- ✓ In the early days of computing, the central processing unit (CPU) performed these calculations.
- ✓ As more graphics-intensive applications such as AutoCAD were developed, however, their demands put strain on the CPU and degradedperformance.
- ✓ GPUs came about as a way to offload those tasks from CPUs and free up processingpower.
- ✓ Today, graphics chips are being adapted to share the work of CPUs and train deep neural networks for AIapplications.
- ✓ A GPU may be found integrated with a CPU on the same circuit, on a graphics card or in the motherboard of a personal computer orserver.
- ✓ NVIDIA,AMD,IntelandARMaresomeofthemajorplayersintheGPUmarket.

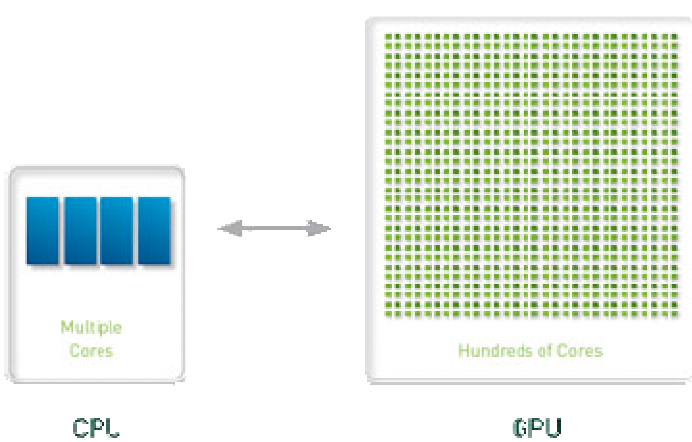
GPU vs. CPU

- ✓ A GPU is able to render images more quickly than a CPU because of its parallel processing architecture, which allows it to perform multiple calculations at the sametime.
- ✓ A single CPU does not have this capability, although multicore processors can perform calculations in parallel by combining more than one CPU onto the same chip.

- ✓ A CPU also has ahigher clock speed, meaning it can perform an individual calculation faster than a GPU so it is often better equippedto handle basic computing tasks.
- ✓ In general, a GPU is designed for data-parallelism and applying the same operation to multiple data-items(SIMD).
- ✓ A CPU is designed fortas -parallelism and doing different operations.

How a GPU works

- ✓ CPU and GPUarchitectures are also differentiated by the number of cores.
- ✓ The core is essentially the processor within theprocessor.
- ✓ MostCPUshavebetweenfourandeightcores,thoughsomehaveupto32cores.
- ✓ Each core can process its own tasks, orthreads.
- ✓ Because some processors have multithreading capability -- inwhich the core is dividedvirtually,allowing asinglecoretoprocesstwothreads--thenumberof threads can be much higher than the number of cores.
- ✓ This can be useful in video editing andtranscoding.
- ✓ CPUs can run two threads (independent instructions) per core (the independent processor unit). GPUs can have four to 10 threads percore.

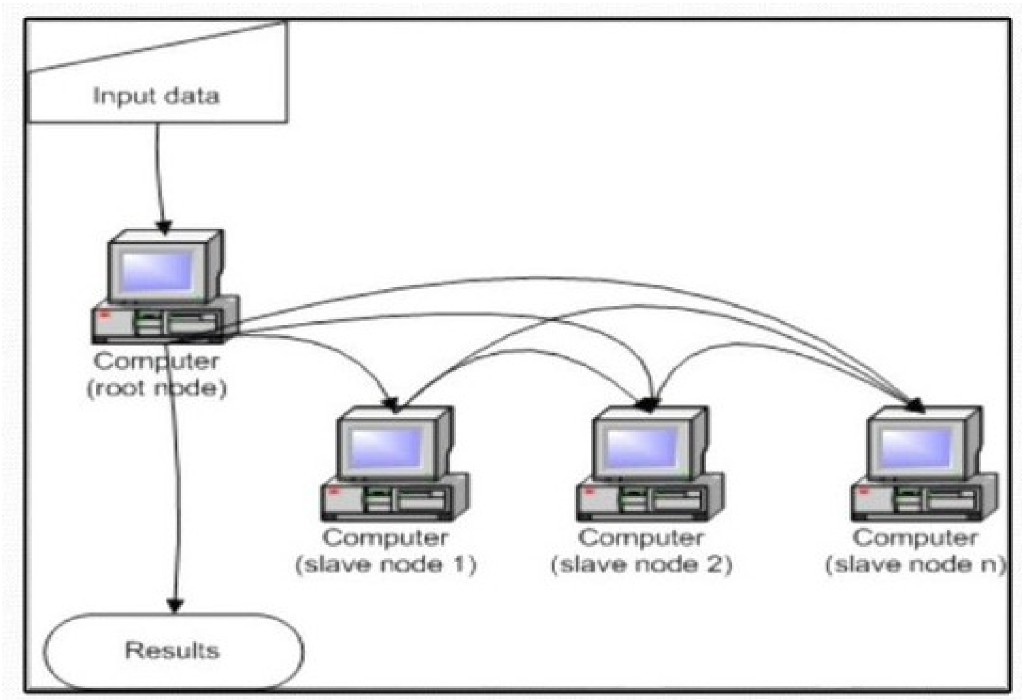


- ✓ GPU computing is the use of a GPU (graphics processing unit) as a co-processor to accelerate CPUs for general-purpose scientific and engineering computing.
- ✓ The GPU accelerates applications running on the CPU by offloading some of the compute-intensive and time consuming portions of thecode.
- ✓ The rest of the application still runs on the CPU. From a user's perspective, the application runs faster because it's using the massively parallel processing power of the GPU to boost performance. This is known as "heterogeneous" or "hybrid"computing.
- ✓ A CPU consists of four to eight CPU cores, while the GPU consists of hundreds of smallercores.
- ✓ Together, they operate to crunch through the data in theapplication.
- ✓ This massively parallel architecture is what gives the GPU its high compute performance.
- ✓ There are a number of GPU-accelerated applications that provide an easy way to access high-performance computing(HPC).

CLUSTER SYSTEM

- ✓ ClusteredsystemsaresimilartoparallelsystemsastheybothhavemultipleCPUs.
- ✓ However a major difference is that clustered systems are created by two or more individual computer systems mergedtogether.
- ✓ Basically, they have independent computer systems with a common storage and the systems worktogether.

A diagram to better illustrate this is:



The clustered systems are a combination of hardware clusters and software clusters. The hardware clusters help in sharing of high performance disks between the systems. The software clusters makes all the systems work together.

Each node in the clustered systems contains the cluster software. This software monitors the cluster system and makes sure it is working as required. If any one of the nodes in the clustered system fail, then the rest of the nodes take control of its storage and resources and try torestart.

Types of Clustered Systems

- **High performanceCluster**
  - 1000 nodes, high level parallelprocess
- **Load BalancingCluster**
  - Balance the workloads
- **Web service Cluster**
  - Web pages &applications
- **StorageCluster**

  - Parallel filesystems
- **DatabaseCluster**
  - Oracle parallelserver

WSC

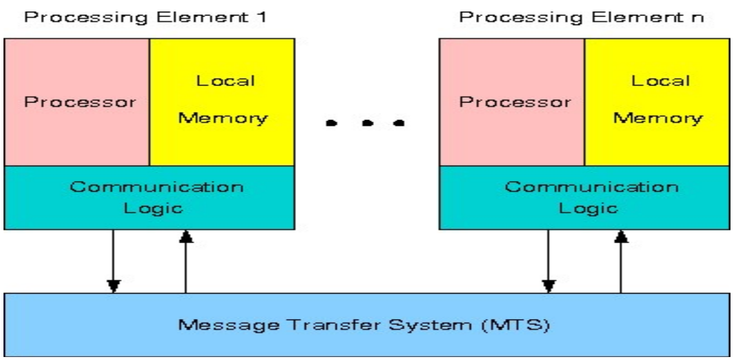
- ✓ Warehouse-scale computers (WSCs) form the foundation of internet services that people use for search, social networking, online maps, video sharing, online shopping, email, cloud computing,etc.
- ✓ The ever increasing popularity of internet services has necessitated the creation of WSCs in order to keep up with the growing demands of thepublic.
- ✓ Although WSCs may seem to be large datacenters, their architecture and operation are different fromdatacenters.
- ✓ The WSC is a descendant of the supercomputer. Today's WSCs act as one giant machine.
- ✓ The main parts of a WSC are the building with the electrical and cooling infrastructure, the networking equipment and the servers, about 50000 to 100000 of them.
- ✓ The costs are of the order of $150M to build such an infrastructure . WSCs have many orders of magnitude more users than high performance computing and play a very important roletoday.

Message Passing Multiprocessor

Communicating between multiple processors by explicitly sending and receiving information.

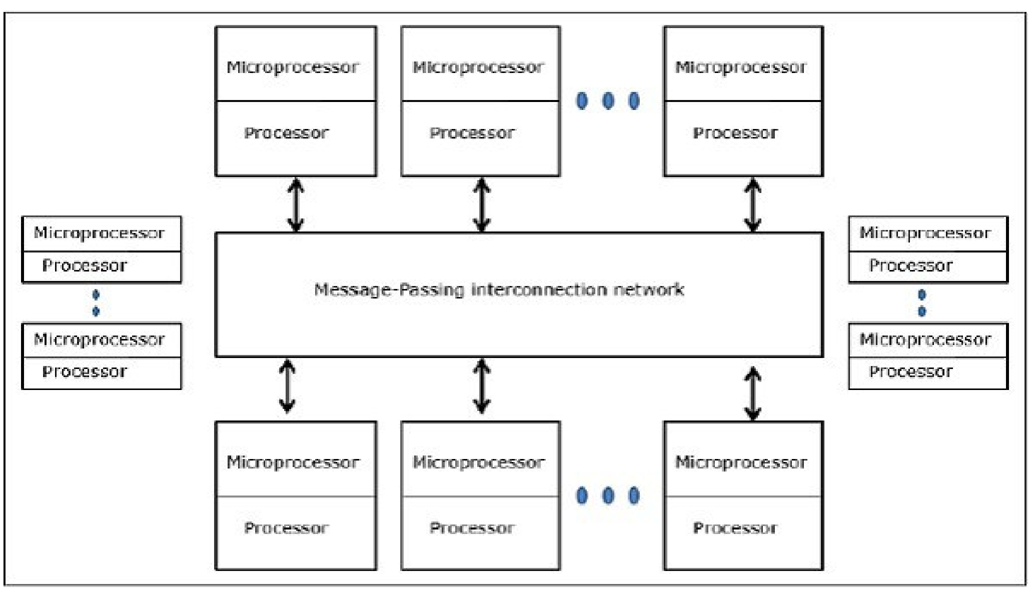
<u>Send messageroutine:</u> A routine used by a processor in machines with private memories to pass a message to anotherprocessor.

- **<u>Receive messageroutine:</u>** A routine used by a processor in machines with private memories to accept a message from anotherprocessor.



- Distributed memory multicomputer system consists of multiple computers, known as nodes, inter-connected by message passingnetwork.
- Each node acts as an autonomous computer having a processor, a local memory and sometimes I/Odevices.

- In this case, all local memories are private and are accessible only to the local processors.
- This is why, the traditional machines are called **no-remote-memory-access (NORMA)** machines.



6. Post MCQ Test

1. Which of the following processor has a fixed length of instructions?
   a) CISC
   b) RISC
   c) EPIC
   d) Multi-core
   Answer: RISC
2. Which one is not benefit of multiprocessors?
   (A) Multiple independent jobs can be made to operate in parallel
   (B) A single job can be partitioned into multiple parallel tasks
   (**C)** Multiple jobs can be made to operate in serial
   (D) All are benefits
   Answer: Multiple jobs can be made to operate in serial
3. MISD data stream is the abbreviation of
   a) Multiple instruction single data stream
   b) Multiple instruction streams, single data stream
   c) Multiple instruction streams, data stream
   d) Many instruction streams, single data stream

   Answer: Multiple instruction streams, single data stream
4. Data-level parallelism/task-level parallelism in a tightly coupled hardware which allows interaction among parallel threads, are processed by
   a) instruction-Level Parallelism
   b) Request-Level Parallelism
   c) Thread-Level Parallelism
   d) Vector Architectures and Graphic Processor Units
   Answer: Thread-Level Parallelism
5. An alternative towards the fine-grained multithreading, the devised technique was
   a) Buffer-grained multi-threading
   b) Miss-grained multi-threading
   c) Coarse-grained multi-threading
   d) Coarse-grained single threading
   Answer: Coarse-grained multi threading

7.Reference:

1. Miles J. Murdocca and Vincent P. Heuring, ―Computer Architecture and Organization: An Integrated approach‖, Second edition, Wiley India Pvt Ltd, 2015.
2. William Stallings, "Computer Organization and Architecture – Designing for Performance", Eighth Edition, Pearson Education, 2010
3. John P. Hayes, Computer Architecture and Organization, Third Edition, Tata McGraw Hill, 2012.