parallel processing

What is parallel processing?

Parallel processing in computer architecture is a technique used in advanced computers to get improved performance of computer systems by performing multiple tasks simultaneously.

In modern world, there is huge demand for high performance computer systems.

This has increased popularity of parallel processing technique use among computer systems.

Advantages of parallel processing:

- Parallel processing systems has capability to perform multiple task simultaneously.
- Parallel processing systems offer high performance.
- Parallel processing architecture reduces the computer costs.
- Parallel processing architecture offers sustained productivity.

So along with the high performance, Parallel processing systems also helps in lower computer costs and sustained productivity.

Parallel computers architecture is **categorized** as:

- Pipelined computers
- Multiprocessor systems

Flynn's classification architecture

Michael J. Flynn in 1996 proposed a scheme for classifying computer organizations.

Flynn's idea for classification of computer organizations stated that based on the number of simultaneous instruction and data streams used by CPU during a program execution, digital computers can be classified into four categories. This is popularly known as **Flynn's classification of computer architecture**.

The four classification of digital computer according to Flynn's are:

1. SISD or **S**ingle **I**nstruction stream-**S**ingle **D**ata stream.
2. SIMD or **S**ingle **I**nstruction stream-**M**ultiple **D**ata stream.
3. MISD or **M**ultiple **I**nstruction stream-**S**ingle **D**ata stream.
4. MIMD or **M**ultiple **I**nstruction stream-**M**ultiple **D**ata stream.

SISD Architecture

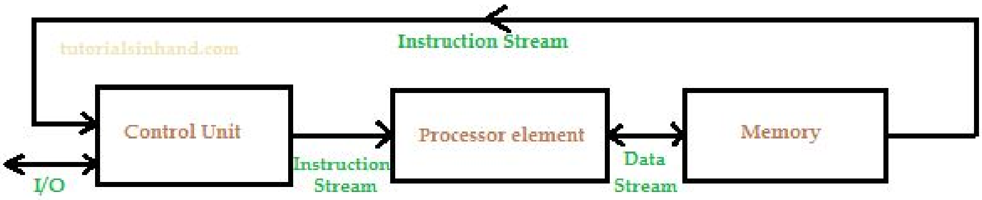
Full form of **SISD** is **Single Instruction stream-Single Data stream**.

In SISD computer architecture, instructions are executed sequentially but may be overlapped in their execution stages. In other words pipelining technique can be used in CPU.

Modern day SISD uniprocessor systems are mostly pipelined. An SISD computer may have more than one functional unit in them, but all are under supervision of one control unit. SISD architecture computers can process only scalar type instructions.

Most serial computers available today follow in SISD architecture.

Given below is the diagram depicting SISD computer architecture.



SISD Computer

Examples of SISD computers

- SISD computer using one functional units – IBM 701, IBM 1620; IBM 7090
- SISD computer using multiple functional units- IBM 360/91; CDC Star-100; TI-ASC

SIMD Architecture

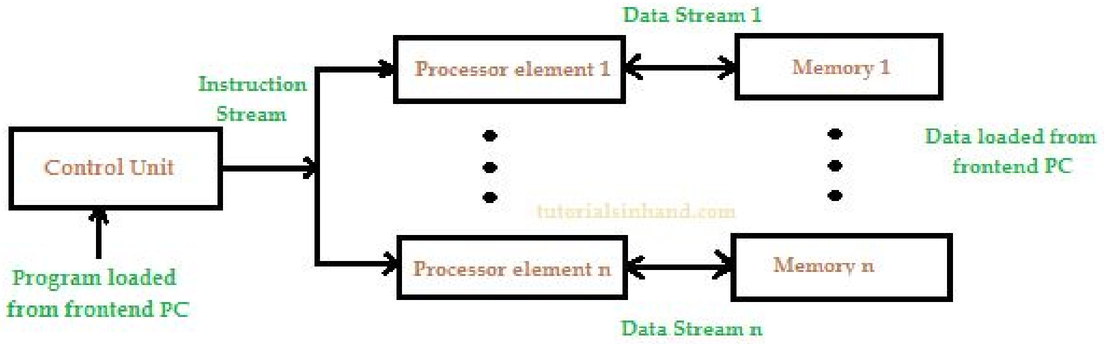
SIMD full form is **Single Instruction stream-Multiple Data stream**.

SIMD computer is one among the four Flynn's classification of computer. Other three are SISD, MISD, MIMD computer.

In **SIMD computer**, we can see from the below diagram there are multiple processing elements supervised by the common control unit.

All the processing elements, which are ALUs, receive the same instruction broadcast from the control unit but operate on different data sets from distinct data streams. As seen in the diagram, control unit sends common instruction stream to each processor element. The shared memory sub-system containing multiple modules is very essential.

Given below is **SIMD architecture diagram**:



SIMD Computer

SIMD computer is used to process **vector type data**.

Array processor falls into SIMD computer.

Examples of SIMD computer

Give below are examples of SIMD computer:

- Illiac-IV
- PEPE
- BSP

MISD architecture

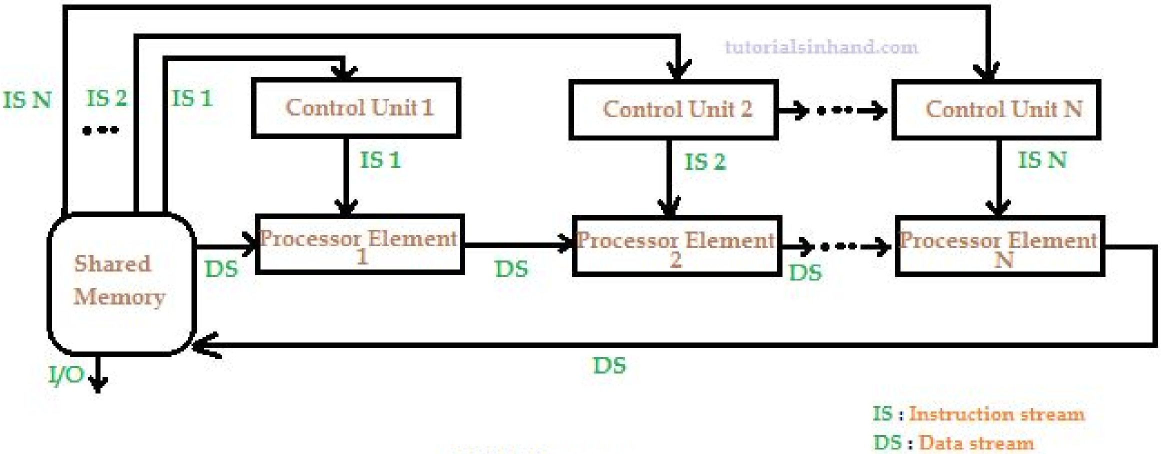
MISD full form is **Multiple Instruction stream-Single Data stream**.

MISD architecture is also known as **systolic arrays**.

MISD is one among the four machine that falls under the Flynn's classification of computer. Other three are SISD, SIMD and MIMD computers.

In MISD computer architecture, there are n processor elements. Each processor elements receives distinct instructions to execute on the same data stream and its derivatives. Here the output of one processor element becomes the input of the next processor element in the series.

Given below is the **diagram of MISD architecture**:



MISD Computer

MISD architecture is not that popular though some fault tolerance machines can be used in this class.

Examples of MISD computer

No practical machine of MISD class exists.

MIMD architecture

MIMD full form is **Multiple Instruction stream-Multiple Data stream**.

MIMD computer is one among four of the Flynn's classification of computer. Other three are SIMD, SISD, MISD computer.

MIMD computer category covers multiple computer system and multiprocessor systems.

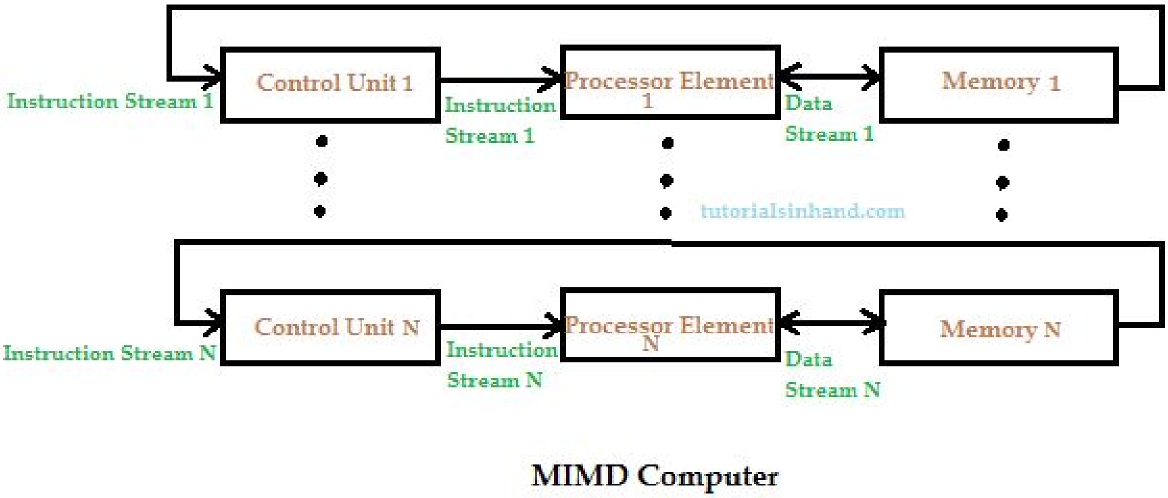
MIMD computer is of two types:

- tightly coupled or Uniform Memory Access (UMA),
- loosely coupled or Non-Uniform Memory Access (NUMA)

MIMD computer is called **tightly coupled or Uniform Memory Access (UMA)** if the degree of interaction among the processor is high.

MIMD computer is called **loosely coupled or Non-Uniform Memory Access (NUMA)** if the degree of interaction among processors is low.

Given below is the **MIMD computer architecture diagram**:



Example of MIMD computer

- MIMD loosely coupled examples: IBM 370/168 MP, Univac 1100/80
- MIMD tightly coupled examples: C.mmp, Cray-3, S-1, Cray-X MP

Pipeline architecture

We are going to discuss an important topic related to parallel processing of computer organization and architecture - "**Pipelining architecture of computer**"

Pipelining is a method in which a sequential task is decomposed into the subtasks and then each of the subtask is executed in a specialized dedicated stage while operating concurrently with all other stages.
Every stage performs partial processing of the task that it is supposed to do. And then the result obtained is passed to the next stage in pipeline. Once the instruction has passed through all the stages, the final result is obtained.

All the stages in pipelining are synchronized by a common clock.
Each of the stages can be understood as a combinational circuit that keeps performing arithmetic or logic operations over the data stream flowing through the pipelining stages. Each stage is seperated by high-speed interface latches (collection of registers).

Performance of pipeline processor

Performance of a pipeline processor is measured with help of following parameters:

- **efficiency**→ The efficiency of a linear pipeline is measured by the percentage of busy time-space spans over the total time-space span, which equal the sum of all busy and idle time-space spans.
- **throughput**→ The number of tasks that can be completed by a pipeline per unit time is known as throughput.
- **speed-up**→ speed-up = (time to execute n tasks in k-stage non-pipeline processor)/(time to execute n tasks in k-stage pipeline processor)

Classification of pipeline processor

Pipeline processors are classified as:

1. Arithmetic pipeline
2. Instruction pipeline
3. Processor pipeline

Classification of pipeline processor

The pipeline processors were classified based on the levels of processing by **Handler** in 1977.

Given below are the **classification of pipeline processor** by given by Handler:

1. Arithmetic pipeline
2. Processor pipeline
3. Instruction pipeline

Lets read about each of them one by one.

Arithmetic pipeline

An arithmetic pipeline generally breaks an arithmetic operation into multiple arithmetic steps. So in arithmetic pipeline, an arithmetic operation like multiplication, addition, etc. can be divided into series of steps that can be executed one by one in stages in Arithmetic Logic Unit (ALU).

Example of Arithmetic pipeline

Listed below are examples of arithmetic pipeline processor:

- 8 stage pipeline used in TI-ASC
- 4 stage pipeline used in Star-100

Processor pipeline

In processor pipeline processing of the same data stream is done by a cascade of processors. In this each cascade of processor is assigned and process a specific task.

There is no practical example found for processor pipeline.

Instruction pipeline

In instruction pipeline processor, the execution of a stream of instructions can be pipelined by overlapping the execution of the current instruction with the fetch, decode and operand fetch of subsequent instructions.

Example of Instruction pipeline

- All high-performance computers nowadays are equipped with instruction pipeline processor.