UNIT-7

Syllabus Contents

- Flynn's Classification of Parallel Processing Systems
- Superscaler Processors

Flynn's Classification

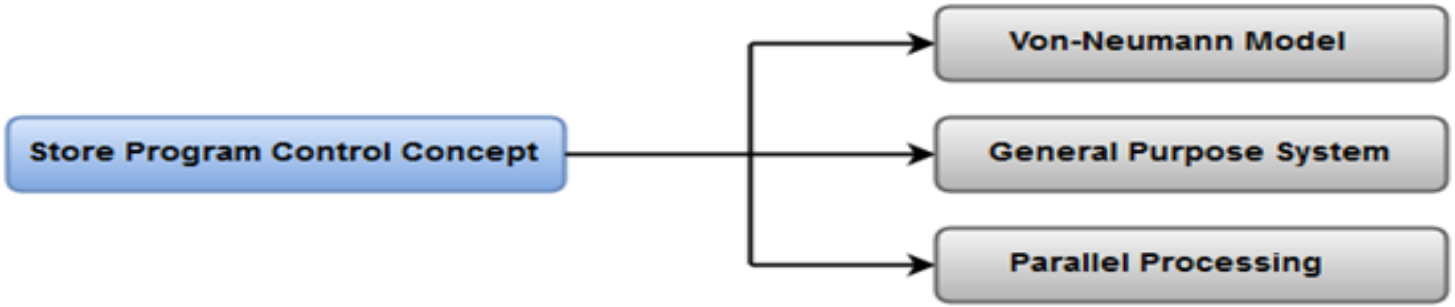
- **Flynn's Classification:**
- Multiprocessing can be defined using Flynn's classification, it is based on **multiplicity of instruction stream and data streams in a computer system.**
- An instruction stream is a sequence of instruction executed by computer.
- A data stream in a sequence of data which includes input data or temporary results.

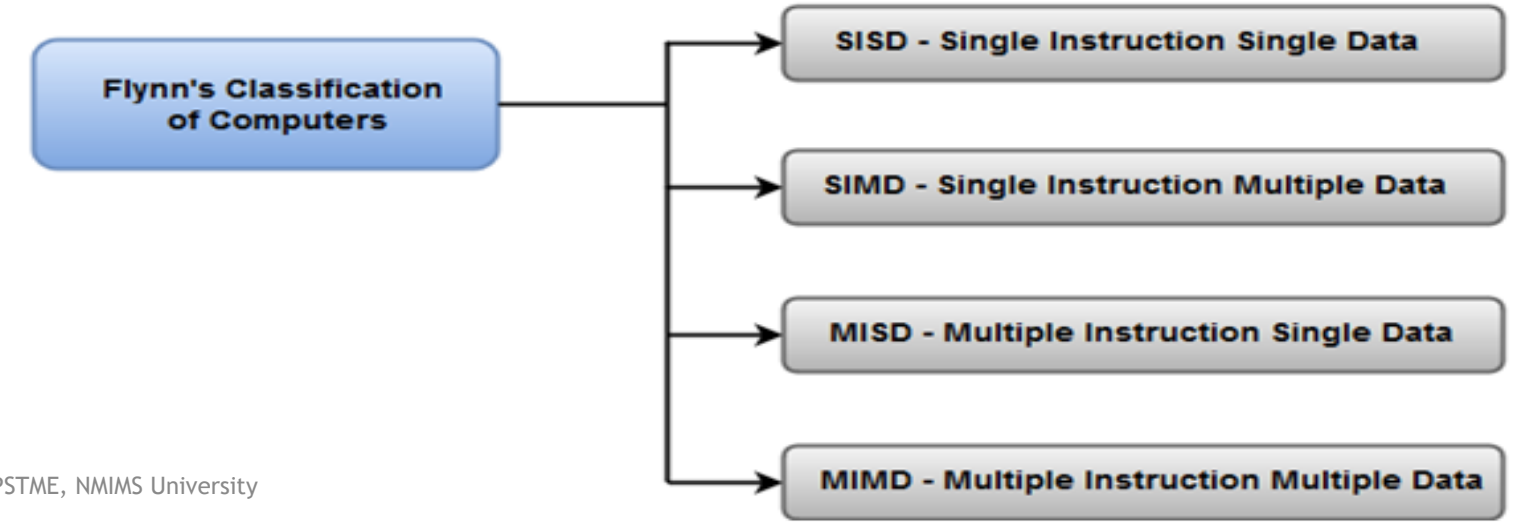
General System Architecture…

- In Computer Architecture, the General System Architecture is divided into two major classification units.

- Store Program Control Concept
- Flynn's Classification of Computers

General System Architecture…



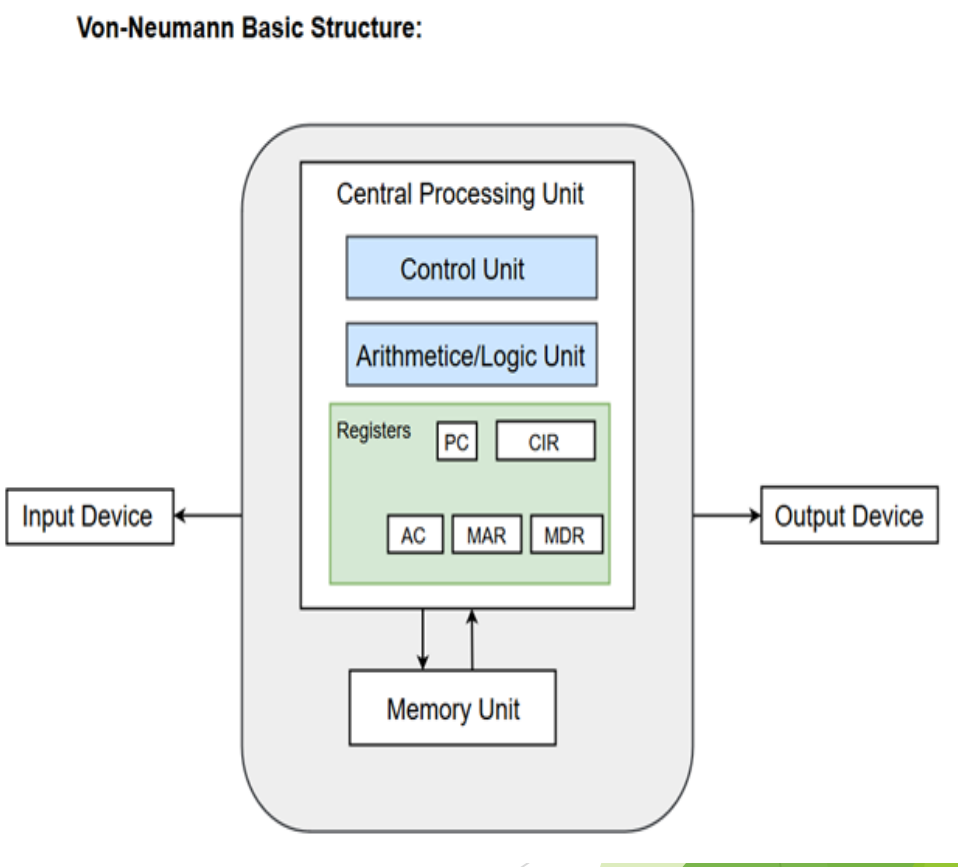


General System Architecture…

- Store Program Control Concept
- The term **Stored Program Control Concept** refers to the storage of instructions in computer memory to enable it to perform a variety of tasks in sequence or intermittently.
- The idea was introduced in the late 1040s by John von Neumann who proposed that a program be electronically stored in the binary-number format in a memory device so that instructions could be modified by the computer as determined by intermediate computational results.
- **ENIAC (Electronic Numerical Integrator and Computer)** was the first computing system designed in the early 1940s. It was based on Stored Program Concept in which machine use memory for processing data.
- Stored Program Concept can be further classified in three basic ways:
- Von-Neumann Model
- General Purpose System
- Parallel Processing

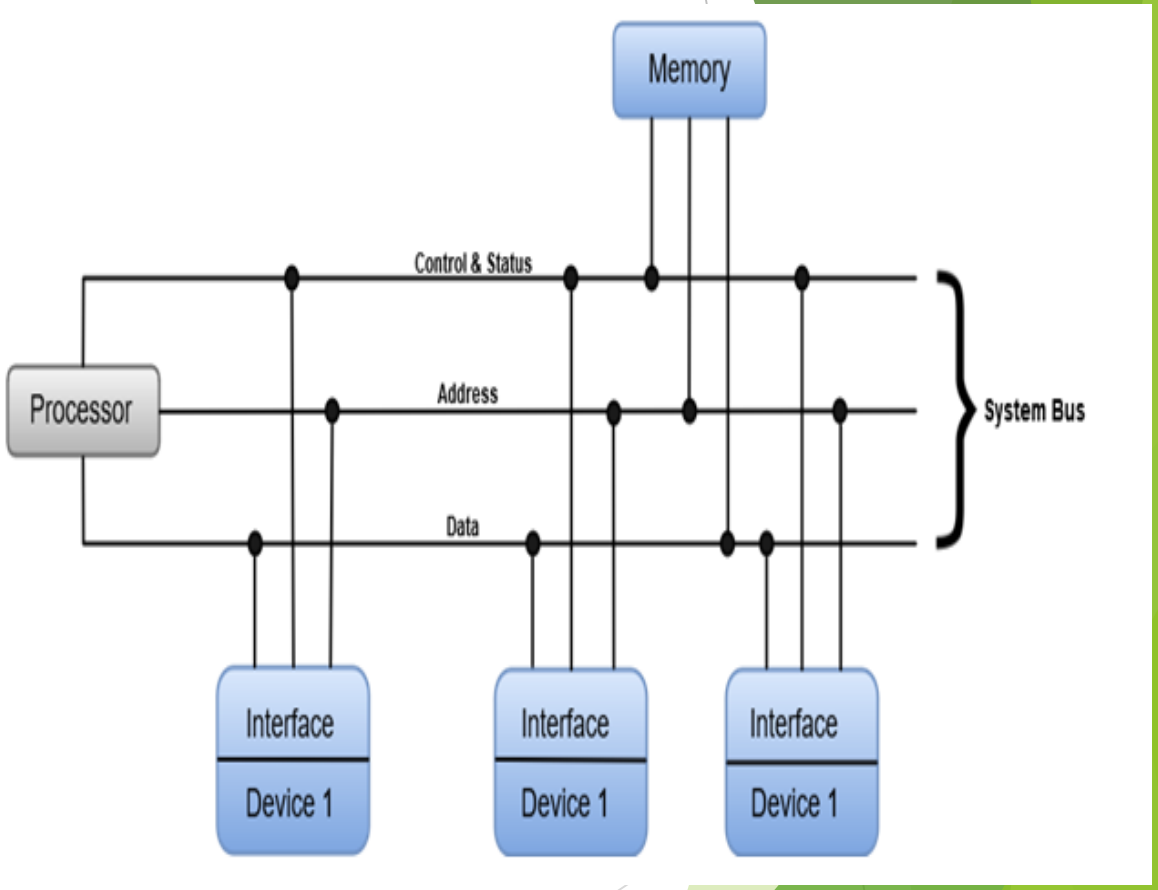
General System Architecture-Von-Neumann Model

- Von-Neumann proposed his computer architecture design in 1945 which was later known as Von-Neumann Architecture. It consisted of a Control Unit, Arithmetic, and Logical Memory Unit (ALU), Registers and Inputs/Outputs.
- Von Neumann architecture is based on the stored-program computer concept, where instruction data and program data are stored in the same memory. This design is still used in most computers produced today.
- A Von Neumann-based computer:
- Uses a single processor
- Uses one memory for both instructions and data.
- Executes programs following the fetch-decode-execute cycle



General System Architecture-Von-Neumann Model

- **General Purpose System**
- The General Purpose Computer System is the modified version of the Von-Neumann Architecture. In simple words, we can say that a general purpose computer system is a modern day architectural representation of Computer System.
- The CPU (Central Processing Unit) consists of the ALU (Arithmetic and Logic Unit), Control Unit and various processor registers.
- The CPU, Memory Unit and I/O subsystems are interconnected by the system bus which includes data, address, and control-status lines.
- The following image shows how CPU, Memory Unit and I/O subsystems are connected through common single bus architecture.



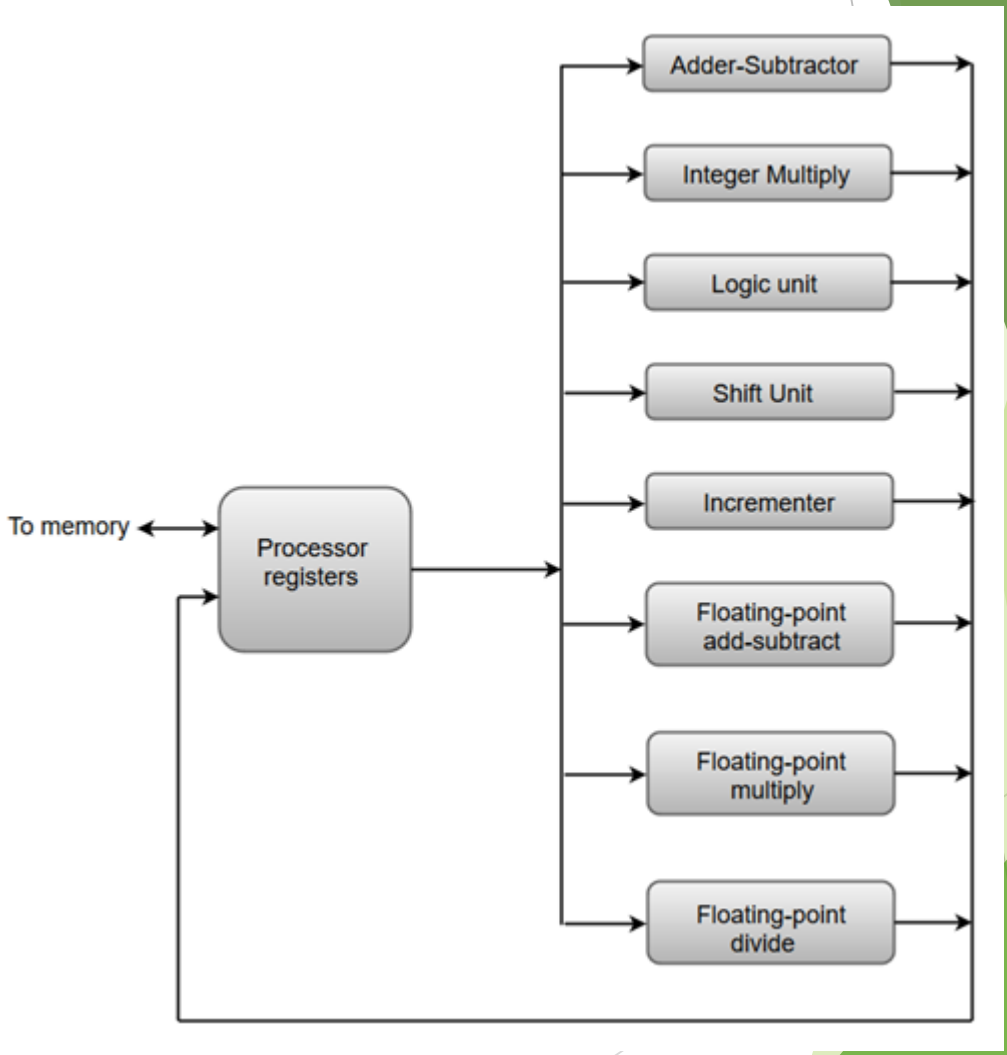
General System Architecture-Von-Neumann Model

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 computer system.
- A parallel processing system can carry out simultaneous data-processing to achieve faster execution time. For instance, while an instruction is being processed in the ALU component of the CPU, the next instruction can be read from memory.
- The primary purpose of parallel processing is to enhance the computer processing capability and increase its throughput, i.e. the amount of processing that can be accomplished during a given interval of time.
- A parallel processing system can be achieved by having a multiplicity of functional units that perform identical or different operations simultaneously. The data can be distributed among various multiple functional units.

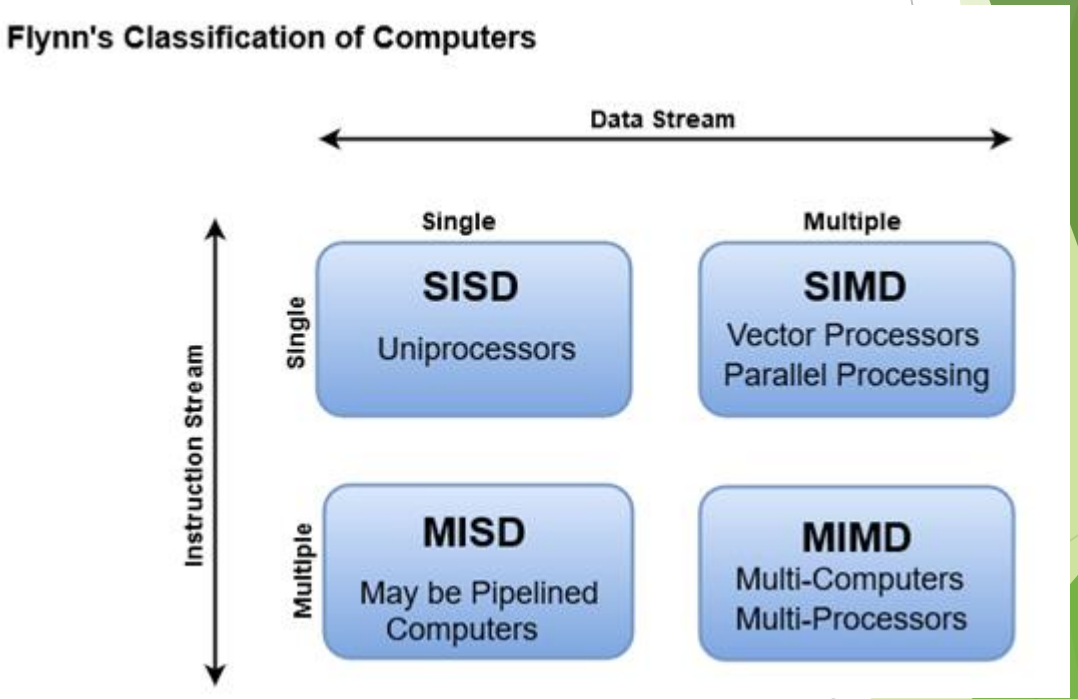
General System Architecture-Von-Neumann Model

- Parallel Processing-The following diagram shows one possible way of separating the execution unit into eight functional units operating in parallel.
- The operation performed in each functional unit is indicated in each block of the diagram:



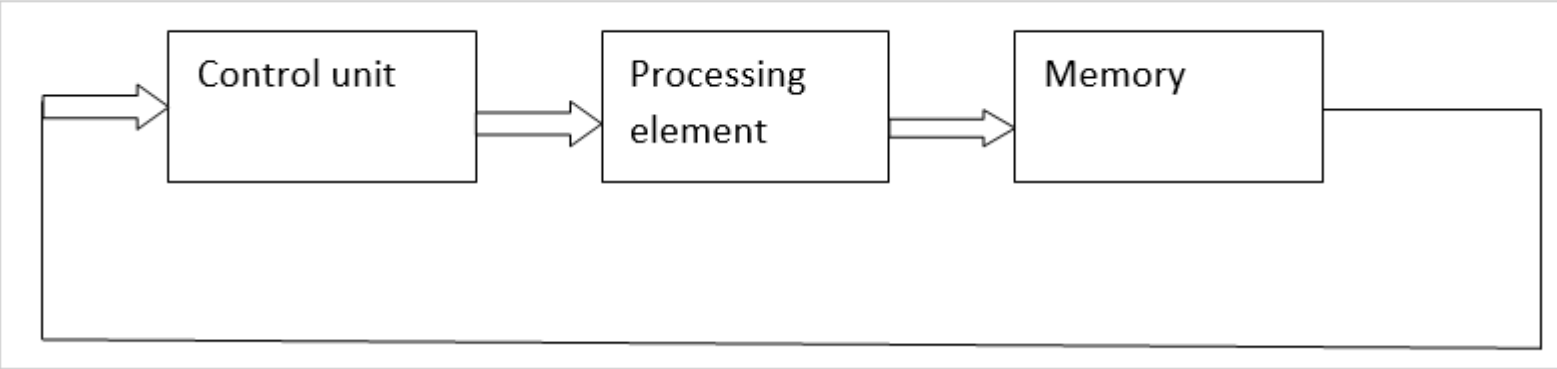
Flynn's classification

- Flynn's classification divides computers into four major groups that are:
- Single instruction stream, single data stream (SISD)
- Single instruction stream, multiple data stream (SIMD)
- Multiple instruction stream, single data stream (MISD)
- Multiple instruction stream, multiple data stream (MIMD)



Classification of computer architecture

- 1) **SISD (Single Instruction Single Data Stream)**
- **Single instruction:** Only one instruction stream is being acted or executed by CPU during one clock cycle.
- **Single data stream:** Only one data stream is used as input during one clock cycle.

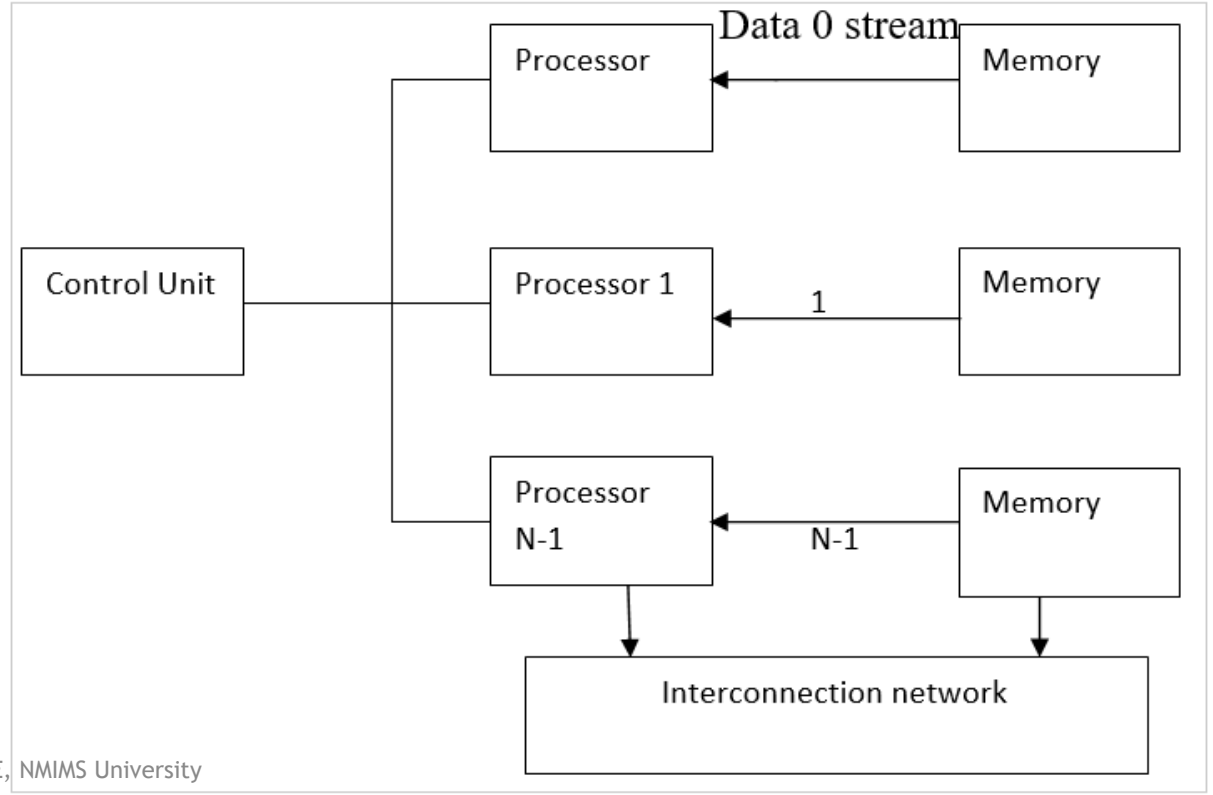


A SISD computing system is a uniprocessor machine that is capable of executing a single instruction operating on a single data stream. Most conventional computers have SISD architecture where all the instruction and data to be processed have to be stored in primary memory.

Classification of computer architecture

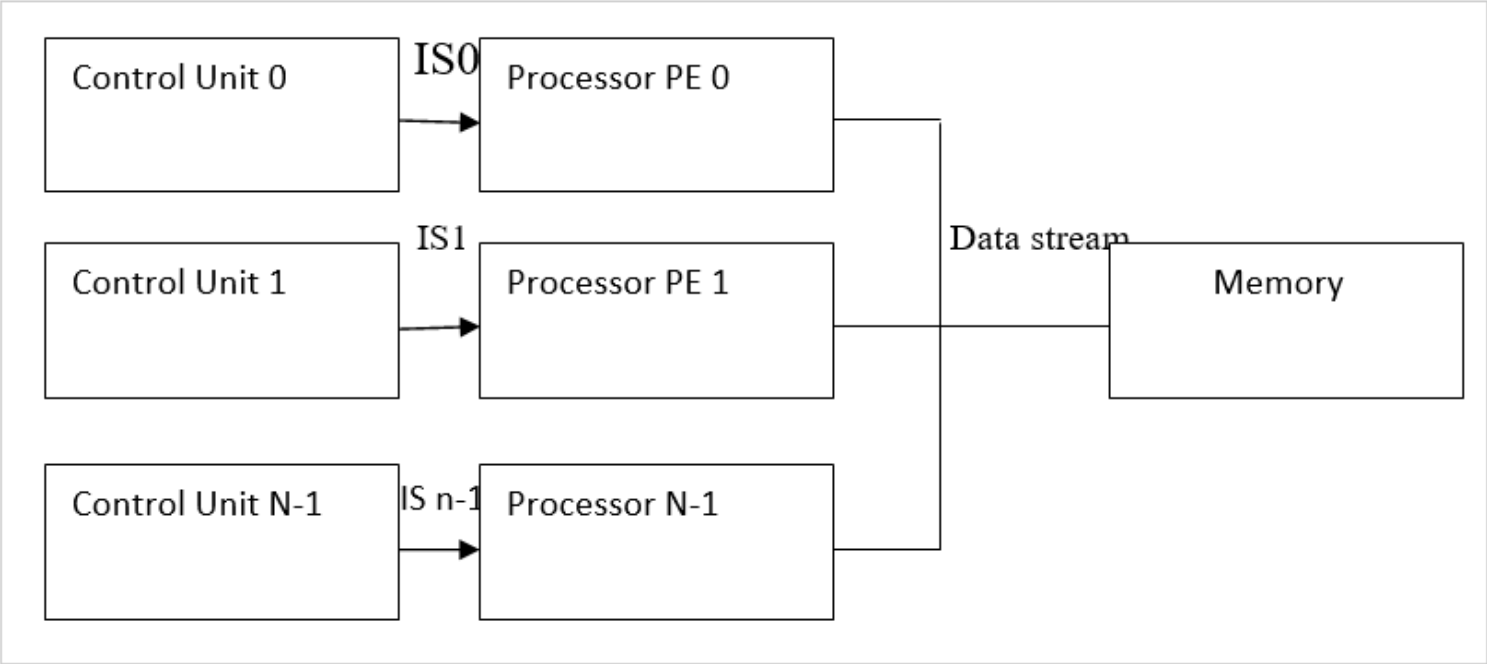
2) SIMD (Single Instruction Multiple Data Stream)

- A SIMD system is a multiprocessor machine, capable of executing the same instruction on all the CPUs but operating on the different data stream.
- **IBM 710 is the real life application of SIMD.**



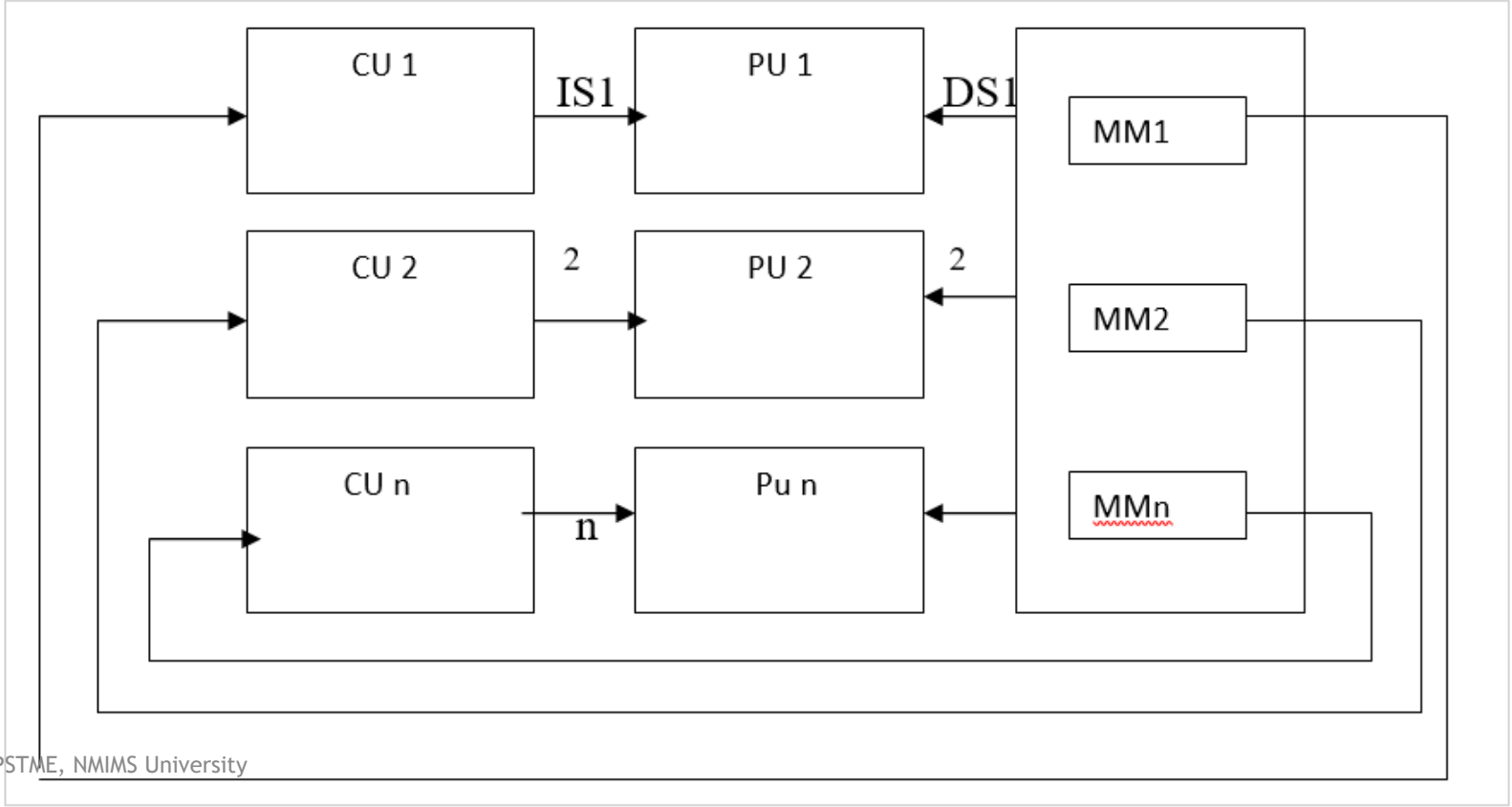
Classification of computer architecture

- **3) MISD (Multiple Instruction Single Data stream)**
- An MISD computing is a multiprocessor machine capable of executing different instructions on process
- single elements but all of them operating on the same data set.



Classification of computer architecture

- **4) MIMD (Multiple Instruction Multiple Data Stream)**
- A MIMD system is a multiprocessor machine that is capable of executing multiple instructions over multiple data streams. Each processing element has a separate instruction stream and data stream.



Superscaler Architecture

- Superscalar architecture is **a method of parallel computing used in many processors**. In a superscalar computer, the central processing unit (CPU) manages multiple instruction pipelines to execute several instructions concurrently during a clock cycle.
- A superscalar processor is created to produce an implementation rate of more than one instruction per clock cycle for a single sequential program. Superscalar processor design defines as a set of methods that enable the central processing unit (CPU) of a computer to manage the throughput of more than one instruction per cycle while performing a single sequential program.
- While there is not a global agreement on the interpretation, superscalar design techniques involve parallel instruction decoding, parallel register renaming, speculative execution, and out-of-order execution. These techniques are usually employed along with complementing design techniques including pipelining, caching, branch prediction, and multi-core in current microprocessor designs.
- Superscalar processor emerged in three consecutive phases as first, the idea was conceived, then a few architecture proposals and prototype machines appeared, and finally, in the last phase, the commercial products reached the market.
- The concept of the superscalar issue was first developed as early as 1970 (Tjaden and Flynn, 1970). It was later reformulated more precisely in the 1980s (Torng, 1982, Acosta et al, 1986).

Superscaler Architecture

- As far as prototype machines are concerned IBM was the first with two significant superscalar developments called the Cheetah and America project. The Cheetah project (1982-83) and the subsequent America project (from 1985 on) were the testbeds for IBM to study superscalar execution.
- The four-way Cheetah machine served as a base for the America processor, which spawned the RS/6000 (1990), which was later renamed the Power1. The Power 1 is almost identical to the America machine (Grohoski, 1990).
- The term superscalar processor is assumed to have first appeared in connection with these developments in an internal IBM Technical Report (Agarwala, T and Cocke, J. High-Performance Reduced Instruction Set Processors, 1987).
- A second early player in the area of superscalar developments was DEC with its Multititan project, carried out from 1985 to 1987. While the Multititan project was the continuation of project Titan (1984), whose goal was to construct a very high-speed RISC processor, this project did not contribute much to the development of the α line of processors.
- The Intel 960CA embedded RISC processor was the first commercial superscalar machine, introduced in 1989. To boost performance subsequently all major manufactures were forced to introduce the superscalar issue in their commercial processor lines.
- Superscalar RISC processors emerged according to two different approaches. Some appeared as the result of transferring a current (scalar) RISC line into a superscalar one. Examples of this are the Intel 960, MC 88000, HP PA (Precision Architecture), SunSparc, MIPS R, and AMD Am29000 RISC lines. The another significant approach was to perceive a new architecture and to execute it from the very starting as a superscalar line. This happened when IBM announced its RS/6000 processor in 1990, later renamed the Power1.

Superscalar Processor

- **Superscalar Processor**
- A Superscalar Processor is a processor designed to exploit Data-Level Parallelism combined with Instruction-Level Parallelism, through the implementation of a Pipeline with multiple Executing Units, an execution resource within a single CPU developed for a specific set of operations.
- Superscalar Processors execute multiple instructions within the implemented Executing Units in the Execution Stage, providing true parallelism within the CPU and, consequently, a CPI < 1.
- In a superscalar design, the processor must read the instructions from memory and decide which ones can be run in parallel, dispatching them to the available Executing Units. The Superscalar Processor can be envisioned as having multiple parallel pipelines, as illustrated in figure 1.

Superscalar Processor

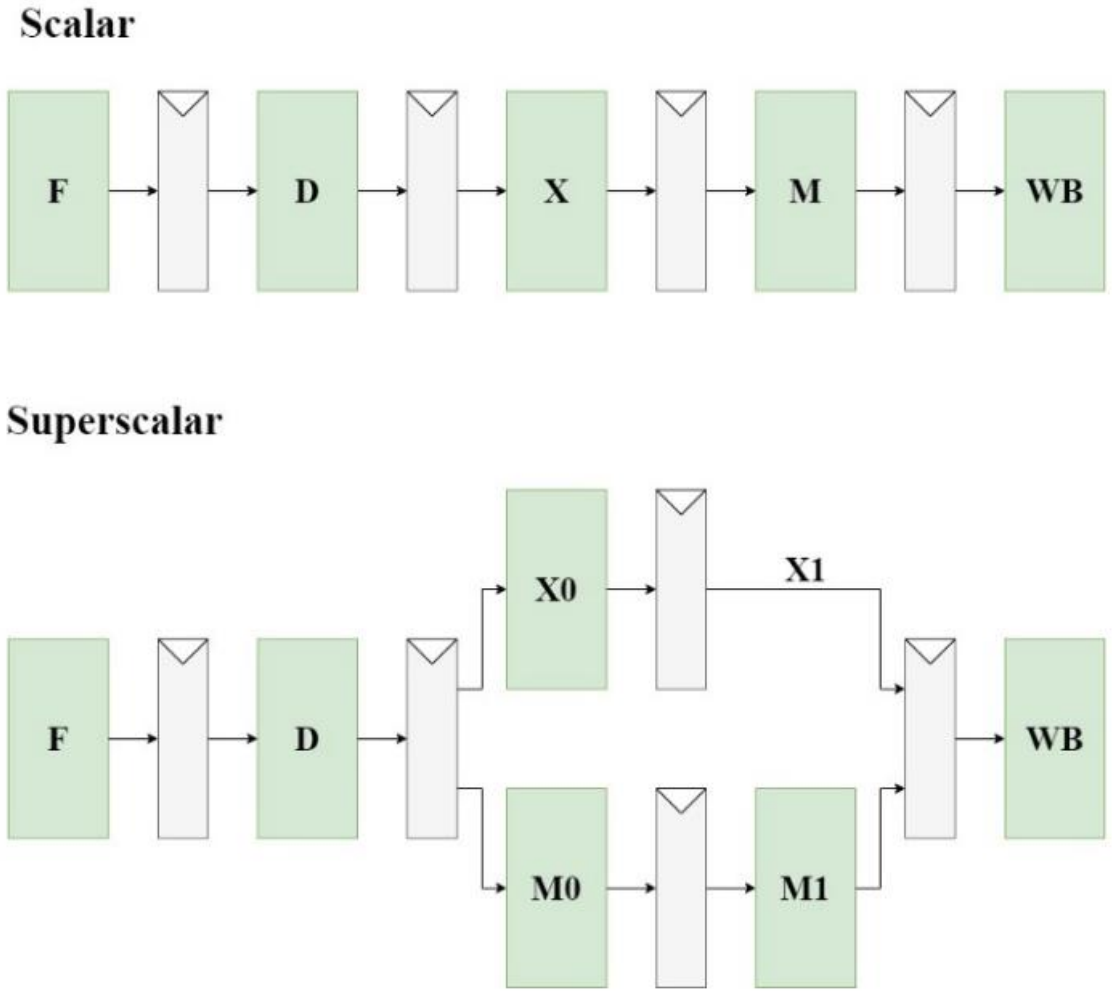
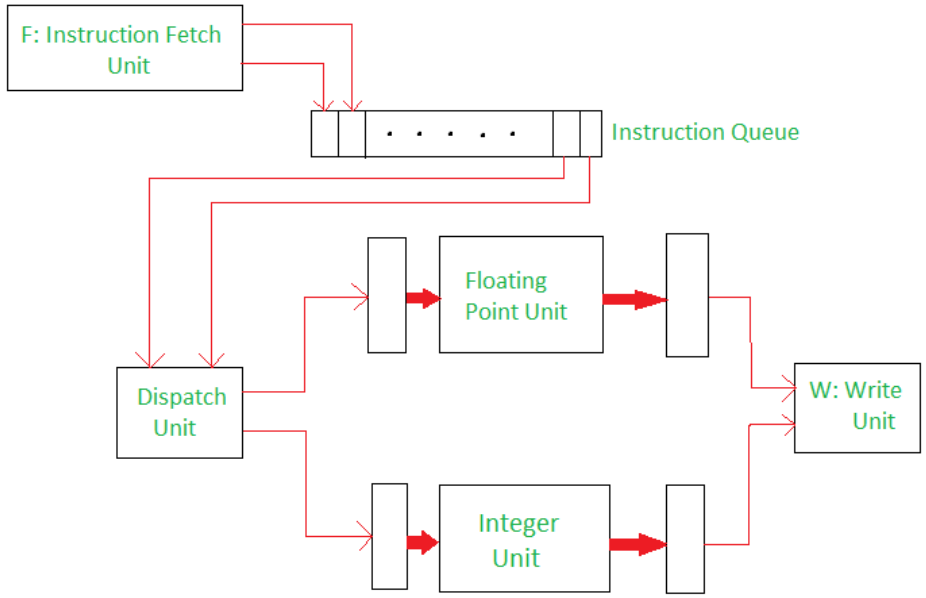


Figure 1 Comparison between a Scalar and a Superscalar Processor. The Superscalar Processor implements one pipeline dedicated for Memory Access and one pipeline for Arithmetic operations.

Superscaler Architecture

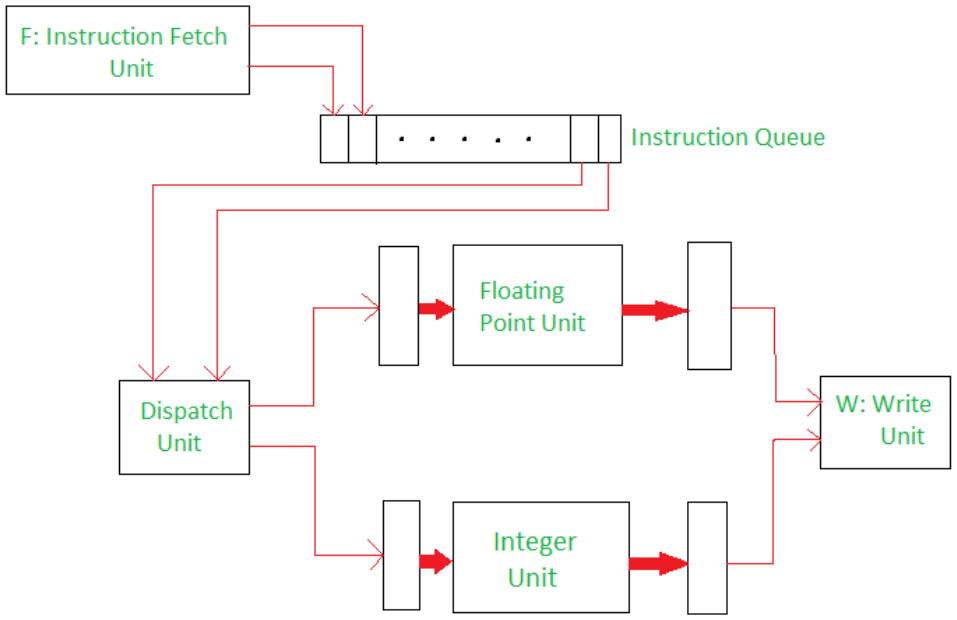
- The approach is to equip the processor with multiple processing units to handle several instructions in parallel in each processing stage.
- With this arrangement, several instructions start execution in the same clock cycle and the process is said to use multiple issue.
- Such processors are capable of achieving an instruction execution throughput of more than one instruction per cycle.
- They are known as 'Superscalar Processors'.



Processor with Two Execution Units

In the above diagram, there is a processor with two execution units; one for integer and one for floating point operations.
The instruction fetch unit is capable of reading the instructions at a time and storing them in the instruction queue.
In each cycle, the dispatch unit retrieves and decodes up to two instructions from the front of the queue.
If there is one integer, one floating point instruction and no hazards, both the instructions are dispatched in the same clock cycle.

Superscaler Architecture



Processor with Two Execution Units

- In the diagram, there is a processor with two execution units; one for integer and one for floating point operations.
- The instruction fetch unit is capable of reading the instructions at a time and storing them in the instruction queue.
- In each cycle, the dispatch unit retrieves and decodes up to two instructions from the front of the queue.
- If there is one integer, one floating point instruction and no hazards, both the instructions are dispatched in the same clock cycle.

Superscaler Architecture

- **Advantages of Superscalar Architecture :**
- The compiler can avoid many hazards through judicious selection and ordering of instructions.
- The compiler should strive to interleave floating point and integer instructions. This would enable the dispatch unit to keep both the integer and floating point units busy most of the time.
- In general, high performance is achieved if the compiler is able to arrange program instructions to take maximum advantage of the available hardware units
- **Disadvantages of Superscalar Architecture :**
- In a Superscalar Processor, the detrimental effect on performance of various hazards becomes even more pronounced.
- Due to this type of architecture, problem in scheduling can occur.