

# HPC Concepts and Terminology

## CS 540- High Performance Computing

*Spring 2017*

# Flynn's Taxonomy

- Taxonomy: the field of science responsible for classification
- There are different ways to classify parallel computers.
- One of the more popular and widely adopted classifications, is called Flynn's Taxonomy.

# Flynn's Classical Taxonomy

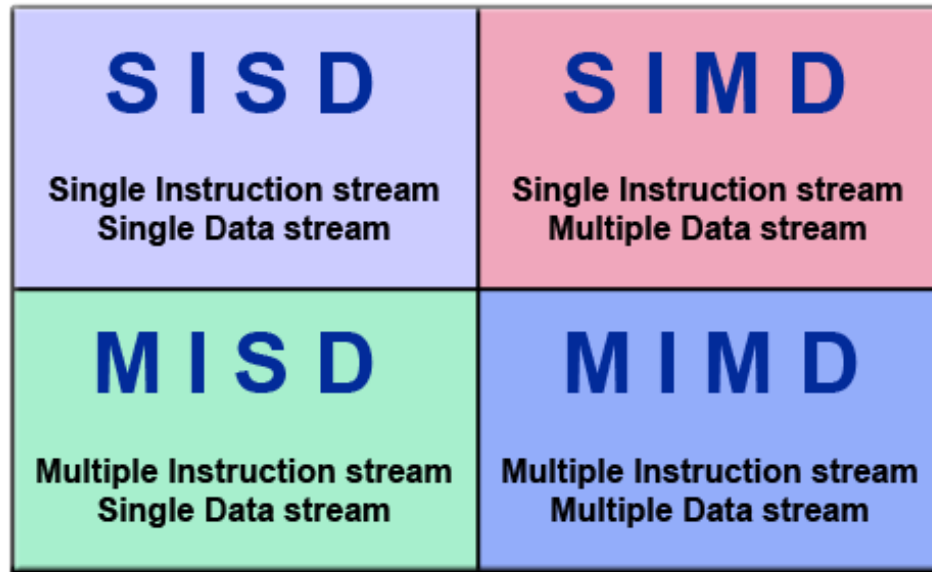
- Proposed by Michael Flynn, Flynn's taxonomy is a classification of computer architectures
- Created 1966
- The classification system has remained, and has been used as a tool in design of modern processors and their functionalities.

# Flynn's Taxonomy cont.

- Flynn's taxonomy distinguishes multi-processor computer architectures according to how they can be classified along the two independent dimensions.
- **Instruction Stream:** a flow of instructions from main memory (RAM) to the CPU.
- **Data Stream:** a flow of operands between processor and memory (bi-directional)
- These dimensions can have only one of two possible states: *Single* or *Multiple*

# Flynn's Taxonomy cont.

- the 4 possible classifications according to Flynn:



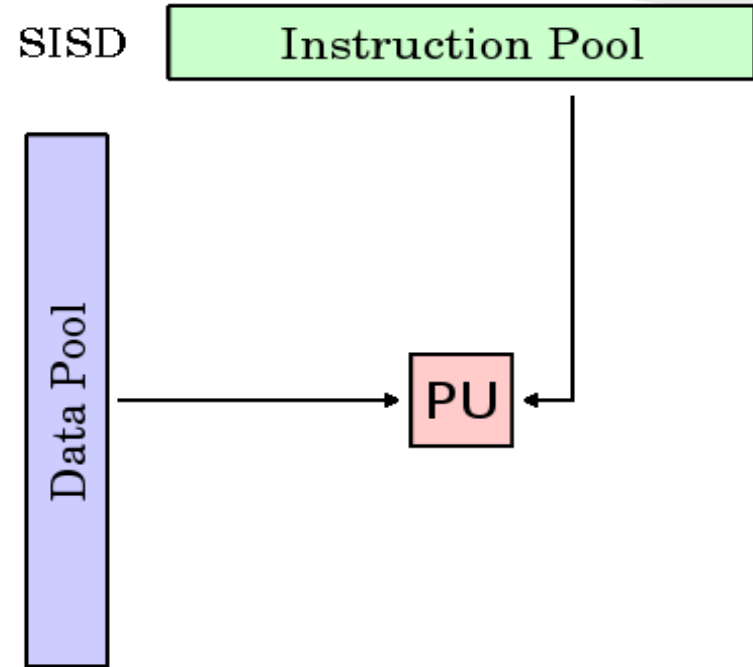
# Flynn's Taxonomy, SISD

Single Instruction, Single Data (SISD):

- A serial computer
- Only one instruction stream is being processed by the CPU during any one clock cycle (Single Instruction)
- Only one data stream is being used as input during any one clock cycle (Single Data)
- Deterministic execution

# Flynn's Taxonomy, SISD

- Older generation mainframes, workstations and single processor/core PCs.



# Flynn's Taxonomy, SIMD

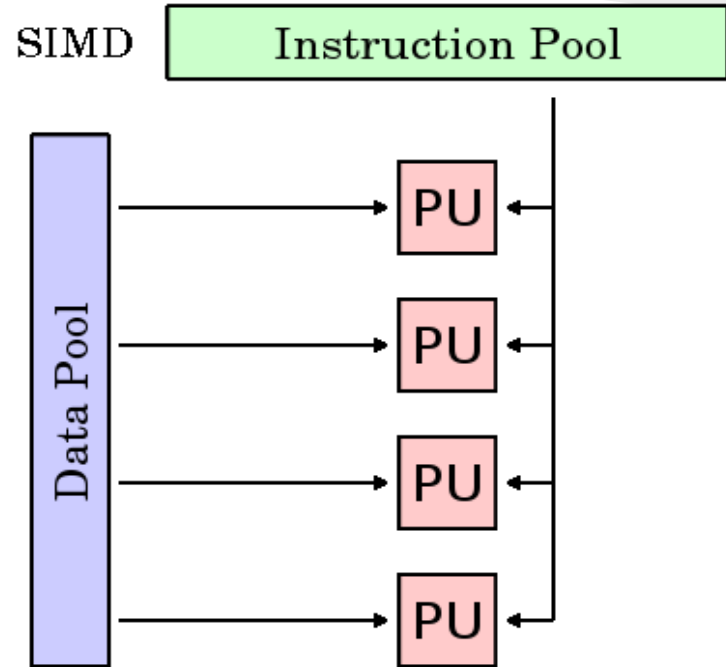
## Single Instruction, Multiple Data (SIMD):

- A type of parallel computer
- All processing units execute the same instruction at any given clock cycle (Single Instruction)
- Each processing unit can operate on a different data element (Multiple Data)
- Best suited for specific problems characterized by a high degree of regularity (i.e. graphics/image processing).
- Synchronous (lockstep) and deterministic execution



# Flynn's Taxonomy, SIMD

- The majority of modern computers, specifically those with graphics processor units (GPUs) utilize SIMD.



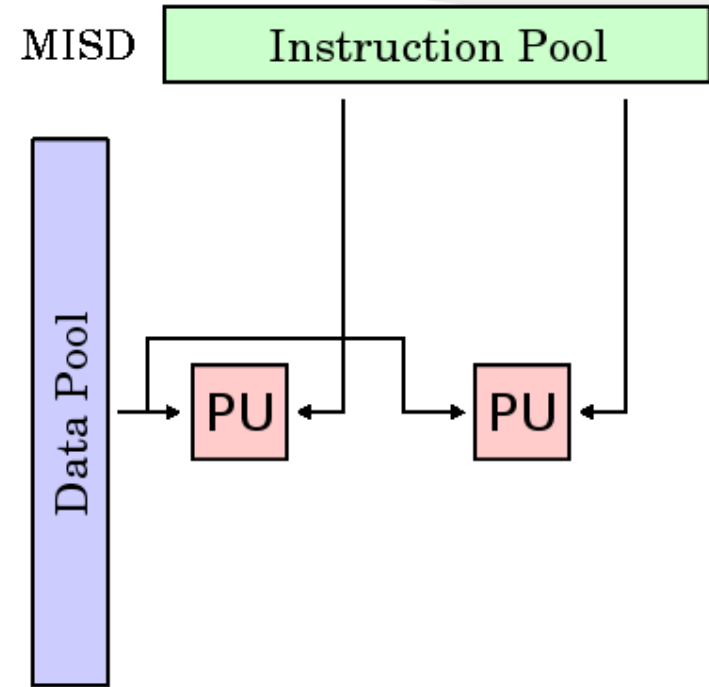
# Flynn's Taxonomy, MISD

Multiple Instruction, Single Data (MISD):

- A type of parallel computer
- Each processing unit operates on the data independently through separate instruction streams (Multiple Instruction).
- A single data stream is served into multiple processing units (Single Data).

# Flynn's Taxonomy, MISD

- Very few (if any) actual samples of this type of parallel computer have ever existed.



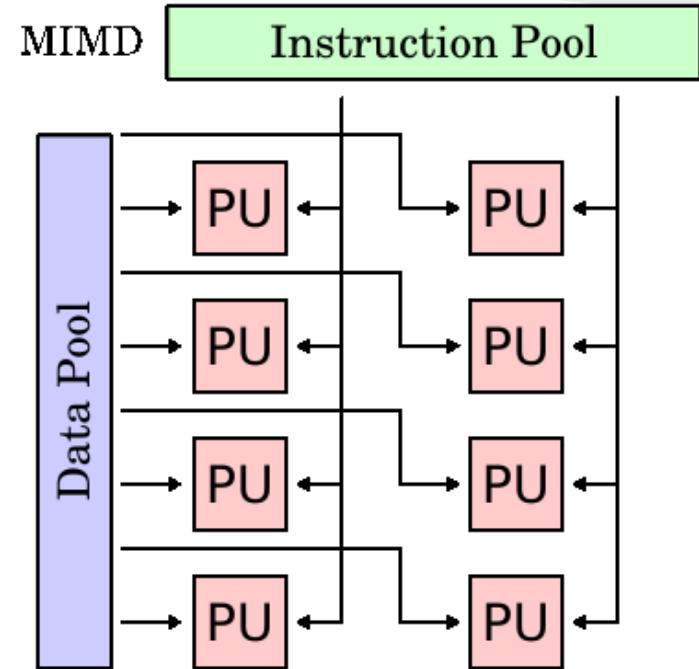
# Flynn's Taxonomy, MIMD

Multiple Instruction, Multiple Data (MIMD):

- A type of parallel computer
- Every processor may be executing a different instruction stream (Multiple Instruction).
- Every processor may process different data streams (Multiple Data).
- Execution can be synchronous or asynchronous, deterministic or non-deterministic

# Flynn's Taxonomy, MIMD

- Most of today's supercomputers fall into this category.



Source: Lawrence Livermore National Laboratory

# Terminology

- **Node**- A standalone computer. It usually consists of multiple CPUs/processors/cores, memory, network interfaces, etc. Nodes are networked together to comprise a supercomputer.
- **Task** - A logically discrete section of computational work. A task is typically a program or program-like set of instructions that is executed by a processor. A parallel program consists of multiple tasks running on multiple processors.
- **Pipelining** - Breaking a task into steps performed by different processor units, with inputs streaming through, much like an assembly line; a type of parallel computing.
- **Shared Memory** - describes a computer architecture where all processors have direct (usually bus based) access to common physical memory.
- **Symmetric Multi-Processor (SMP)** - Shared memory hardware architecture where multiple processors share a single address space and have equal access to all resources.

# Terminology cont.

- **Distributed Memory** - refers to network based memory access for physical memory that is not common. As a programming model, tasks can only logically "see" local machine memory and must use communications to access memory on other machines where other tasks are executing.

# References

1. Blaise Barney, Lawrence Livermore National Laboratory,  
[https://computing.llnl.gov/tutorials/parallel\\_comp/#Overview](https://computing.llnl.gov/tutorials/parallel_comp/#Overview)
2. [https://en.wikipedia.org/wiki/Flynn's\\_taxonomy](https://en.wikipedia.org/wiki/Flynn's_taxonomy)
3. <https://www.citutor.org/index.php>, Parallel Computing Explained