

# **HARDWARE ARCHITECTURES FOR PARALLEL PROCESSING**

The main parts of parallel processing are CPUs. Based on how many instructions and data streams can be processed at the same time, computer systems are grouped into four types:

- 1. Single-instruction, Single-data (SISD) systems**
- 2. Single-instruction, Multiple-data (SIMD) systems**
- 3. Multiple-instruction, Single-data (MISD) systems**
- 4. Multiple-instruction, Multiple-data (MIMD) systems**

## **1. Single-instruction, Single-data (SISD) Systems**

### **What it means:**

- The system has one processor that can handle one instruction at a time.

- This instruction processes one set of data at a time.

### How it works:

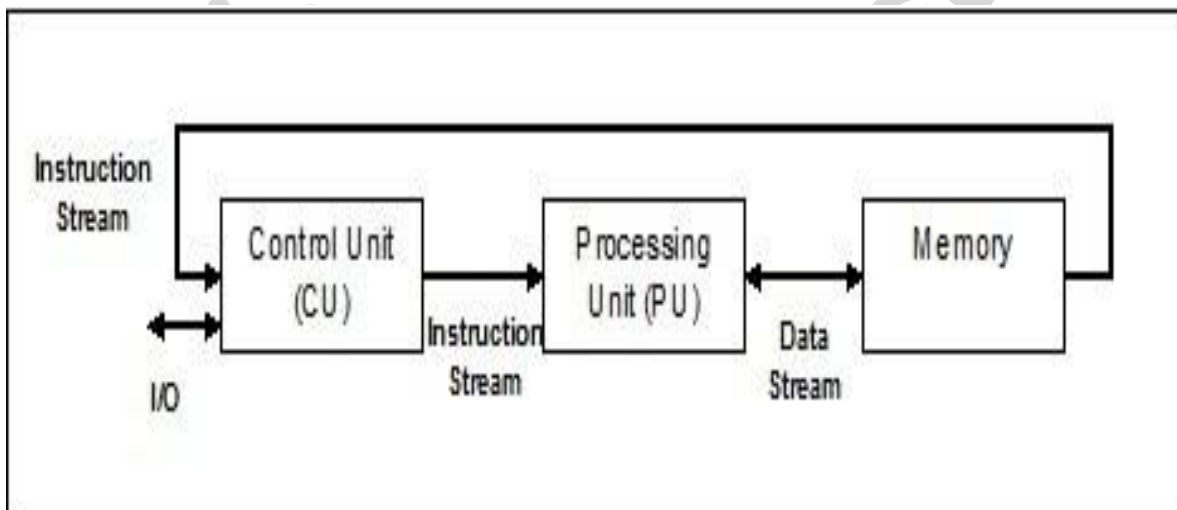
- The computer performs operations in order, step by step.
- It's like a person reading and solving math problems one by one without skipping any.

### Common usage:

- Most of the regular computers we use, like personal laptops or desktops, work this way. They are also called sequential computers because they work step by step.

### Examples:

- Early PCs (like the IBM PC), Macintosh computers, and regular workstations use this system.



## 2. Single-instruction, Multiple-data (SIMD) Systems

### What it means:

- The system has one instruction that is applied to many data points at the same time.

### How it works:

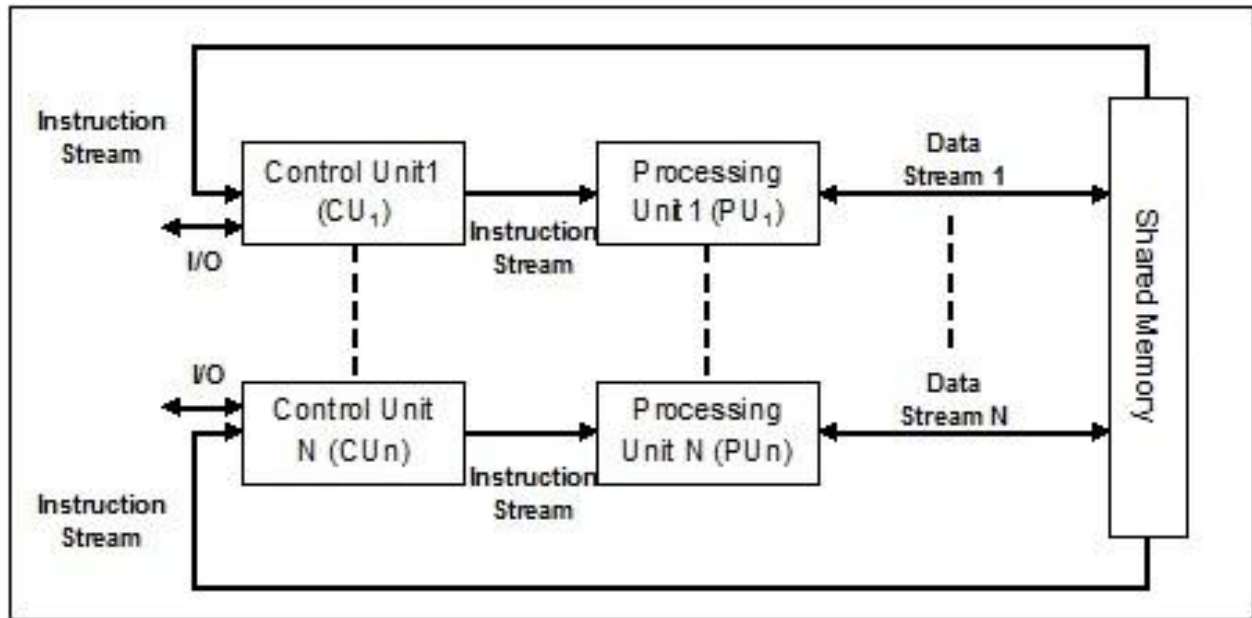
- Imagine a teacher explaining a math problem to a whole class, and everyone in the class solves the same type of problem but with different numbers. The teacher (instruction) is the same, but each student (data) has a different set of numbers to solve.
- This system can process large amounts of similar data at once.

### Common usage:

- This is useful in tasks like image processing or scientific calculations, where the same operation needs to be applied to many data points (like pixels in an image).

### Examples:

- Graphics cards (GPUs) used for gaming or video editing are based on SIMD systems.



### 3. Multiple-instruction, Single-data (MISD) Systems

#### What it means:

- The system can process many different instructions on one set of data.

#### How it works:

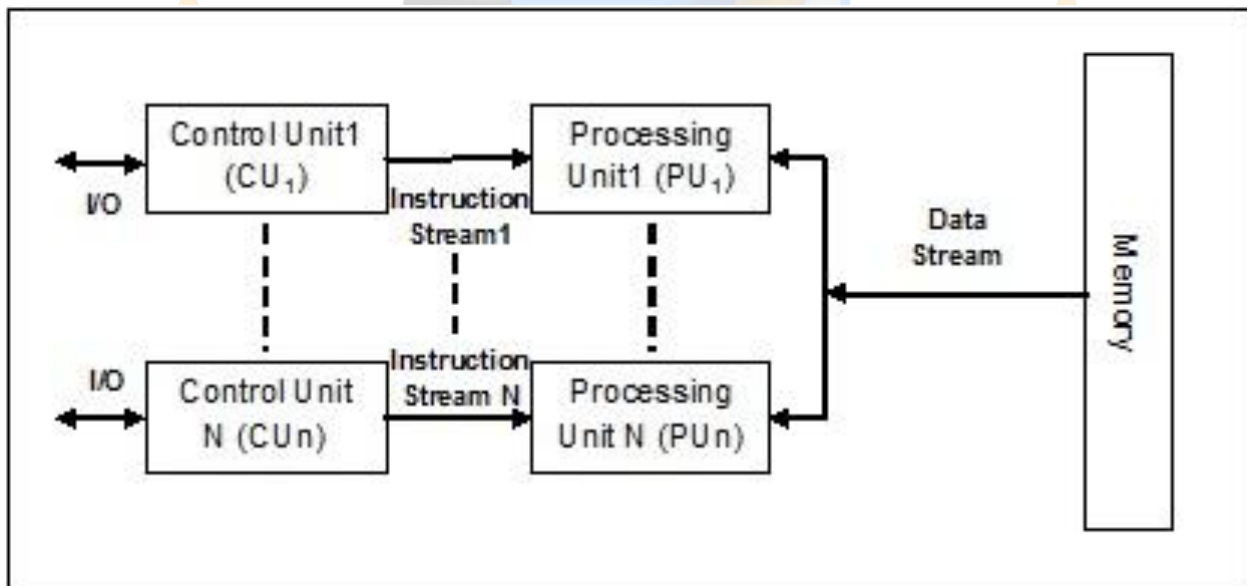
- Imagine several chefs all working on the same dish but using different techniques. Each chef (instruction) might handle

one part of the recipe, but all are working on the same meal (data).

- This is not commonly used in everyday computers.

### Common usage:

- It's very rare but might be used in specialized systems where a single data stream needs to be processed in different ways, such as in fault-tolerant systems (systems designed to keep working even if something goes wrong).



## 4. Multiple-instruction, Multiple-data (MIMD) Systems

**What it means:**

- The system can handle many instructions at the same time, each working on different sets of data.

### **How it works:**

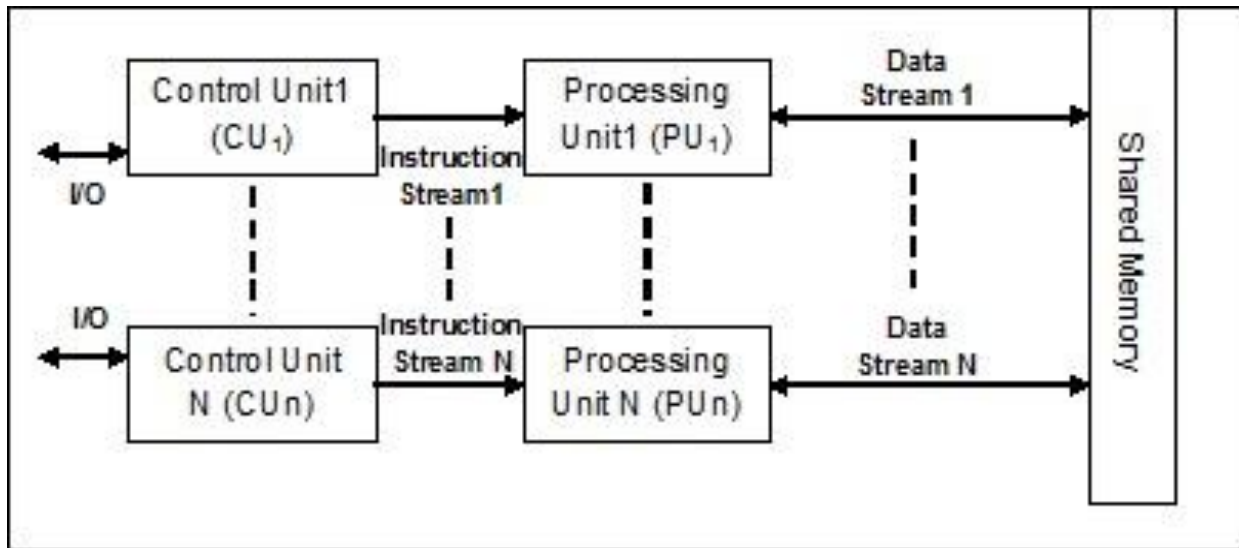
- Imagine a team of builders working on different parts of a house simultaneously. One builder works on the walls, another on the roof, and another on the plumbing. They are all doing different tasks (instructions) on different parts of the house (data).
- MIMD systems are very powerful because they can perform multiple complex operations at once.
- MIMD is broadly categorized into two categories : 1. Shared Memory Systems 2. Distributed Memory Systems

### **Common usage:**

- These systems are often used in supercomputers or advanced multi-core processors that require a lot of power to solve complex problems quickly.

### **Examples:**

- Modern multicore processors in servers, gaming consoles, and high-performance computers are examples of MIMD systems.



### SHARED VS. DISTRIBUTED MEMORY:

- **Shared Memory Systems:**
  - All processors share the same memory.
  - Easy to program.
  - Difficult to manage when too many processors are used, as they might interfere with each other when accessing the memory.

- **Distributed Memory Systems:**

- Each processor has its own memory.
- Easier to expand as more processors are added.
- Avoids memory traffic jams.
- Harder to program because processors must communicate by sending messages.

**In summary:**

- **SISD:** One instruction, one data (step by step, like most personal computers).
- **SIMD:** One instruction, many data (great for processing many similar things at once, like in images).
- **MISD:** Many instructions, one data (rare, used for specialized purposes).
- **MIMD:** Many instructions, many data (very powerful, used in advanced computing tasks).