Lecture Overview: Classifications of Computer Architecture

This lecture covers two main classifications of computer architecture: Von Neumann vs. non-Von Neumann (Harvard family) and Flynn's Taxonomy based on instruction/data streams.

The content is structured into two primary approaches for classifying architectures.

Classification 1: Von Neumann vs. Harvard Architectures

Step 1: Von Neumann Architecture (Princeton Architecture)

Von Neumann architecture features **three basic hardware subsystems**: **CPU**, **main memory**, and **I/O system**.

Key characteristics:

- **Stored program computer**: Main memory holds the program controlling operations; CPU can manipulate its own program and data.
- Executes **instructions sequentially** (one operation at a time).
- **Single path** between main memory and processor (source of debate).
- Origin: Proposed by Von Neumann at **Princeton University**.

Limitation: **Von Neumann bottleneck** - Instructions and data share the same memory path, preventing simultaneous access.

> Systems having pure Von Neumann architecture stores the instructions and the data in the same memory unit thus both the data and the instructions are fetched over the same path.

Step 2: Harvard Architecture

Harvard architecture uses **two separate memory units**: one for **instructions**, one for **data**.

Advantages:

- Processor can **read instruction AND access data simultaneously**.
- **Faster** than Von Neumann (no competition for memory pathway).
- Origin: Proposed by **Harvard University**, adopting most Von Neumann features but with parallelism.

Step 3: Modified Harvard Architecture

Modified Harvard combines both approaches with a **cache** (small, fast memory).

Behavior:

- **From cache**: Acts like pure **Harvard** (separate instruction/data).
- **From main memory**: Acts like pure **Von Neumann**.
- **Common in modern processors**.

This Venn diagram shows core overlaps and key differences between Von Neumann and Harvard architectures.

Classification 2: Flynn's Taxonomy (1966)

Flynn's classification uses **instruction streams** and **data streams**: SISD, SIMD, MISD, MIMD.

Step 1: SISD (Single Instruction Single Data)

- **One CPU** executes **one instruction** at a time on **one data item**.
- **Von Neumann architecture** belongs here.

Step 2: SIMD (Single Instruction Multiple Data)

- **One control unit** (Von Neumann-like) broadcasts **single instruction** to **multiple ALUs**.
- ALUs execute same operation on **different data** (in lockstep).

- Example: **Processor arrays**.

Step 3: MISD (Multiple Instruction Single Data)

- **Multiple programs** on **same data** (logical concept).
- **No practical implementations**, but some MIMD machines can simulate.

Step 4: MIMD (Multiple Instruction Multiple Data)

- **Multiple independent processors** (multiprocessors).
- Each executes **different instructions** on **own data**.
