SNS COLLEGE OF TECHNOLOGY

(An Autonomous Institution)

COIMBATORE – 641035

19MCE304- DESIGN OF EMBEDDED SYSTEMS

Computer architecture taxonomy classifies the various architectural designs of computer systems. It helps in understanding the structure, function, and performance of different computer architectures. Here's an overview of the primary classifications:

1. Based on Instruction Set Architecture (ISA)

a. Complex Instruction Set Computing (CISC)

- **Characteristics**: Complex instructions that can perform multiple operations; variable-length instruction format.
- **Examples**: x86 architecture (used in most personal computers).

b. Reduced Instruction Set Computing (RISC)

- **Characteristics**: Simple and small set of instructions; fixed-length instruction format; emphasis on software to handle complex operations.
- **Examples**: ARM architecture (used in many mobile devices), MIPS, PowerPC.

c. Very Long Instruction Word (VLIW)

- **Characteristics**: Instructions explicitly specify multiple operations to be performed in parallel; relies heavily on the compiler for parallelism.
- **Examples**: Intel Itanium architecture.

d. Explicitly Parallel Instruction Computing (EPIC)

- **Characteristics**: Similar to VLIW but with additional features to handle instruction-level parallelism.
- **Examples**: Intel Itanium.

2. Based on Processor Design

a. Single Instruction, Single Data (SISD)

- **Characteristics**: A single processor executes a single instruction stream on a single data stream.
- **Examples**: Traditional uniprocessor systems.

b. Single Instruction, Multiple Data (SIMD)

- **Characteristics**: A single instruction stream operates on multiple data streams; useful in applications with lots of data-level parallelism.
- **Examples**: Vector processors, Graphics Processing Units (GPUs).

c. Multiple Instruction, Single Data (MISD)

- **Characteristics**: Multiple instruction streams operate on a single data stream; rarely used in practice.
- **Examples**: Fault-tolerant systems like the Space Shuttle’s flight control computer.

d. Multiple Instruction, Multiple Data (MIMD)

- **Characteristics**: Multiple processors execute different instructions on different data streams; supports task-level parallelism.
- **Examples**: Multicore processors, distributed systems.

3. Based on Parallelism

a. Instruction-Level Parallelism (ILP)

- **Characteristics**: The ability to execute multiple instructions simultaneously within a single CPU cycle.
- **Techniques**: Pipelining, superscalar execution, out-of-order execution.
- **Examples**: Modern x86 processors, ARM Cortex-A series.

b. Data-Level Parallelism (DLP)

- **Characteristics**: The ability to perform the same operation on multiple data points simultaneously.
- **Examples**: SIMD architectures, GPUs.

c. Task-Level Parallelism (TLP)

- **Characteristics**: The ability to execute different tasks or threads in parallel.
- **Examples**: Multithreading, multiprocessing systems.

4. Based on Memory Architecture

a. Von Neumann Architecture

- **Characteristics**: Single memory space for instructions and data; the processor fetches instructions and data over a single bus.
- **Examples**: Traditional computer systems.

b. Harvard Architecture

- **Characteristics**: Separate memory spaces and buses for instructions and data; allows simultaneous access to both.
- **Examples**: Many microcontrollers, DSPs (Digital Signal Processors).

c. Modified Harvard Architecture

- **Characteristics**: Combines elements of both Von Neumann and Harvard architectures; typically has separate caches for instructions and data but unified main memory.
- **Examples**: Most modern CPUs.

5. Based on Specialization

a. General-Purpose Processors (GPP)

- **Characteristics**: Designed for a wide range of applications.
- **Examples**: x86 processors, ARM processors.

b. Application-Specific Integrated Circuits (ASICs)

- **Characteristics**: Custom-designed for a specific application; highly efficient but inflexible.
- **Examples**: Chips used in embedded systems, certain parts of smartphones.

c. Field-Programmable Gate Arrays (FPGAs)

- **Characteristics**: Reconfigurable hardware; can be programmed to perform specific tasks.
- **Examples**: Used in prototyping, telecommunications, and signal processing.

d. Digital Signal Processors (DSPs)

- **Characteristics**: Specialized for signal processing tasks; optimized for high-speed numeric calculations.
- **Examples**: Audio and video processing chips.

6. Based on Performance and Power

a. High-Performance Computing (HPC)

- **Characteristics**: Optimized for maximum performance; often used in supercomputers.
- **Examples**: Cray supercomputers, IBM Blue Gene.

b. Low-Power Computing

- **Characteristics**: Optimized for minimal power consumption; often used in mobile and embedded devices.
- **Examples**: ARM Cortex-M series, Intel Atom processors.

7. Based on Emerging Architectures

a. Quantum Computing

- **Characteristics**: Utilizes quantum bits (qubits) for computation; can solve certain problems exponentially faster than classical computers.
- **Examples**: IBM Q, Google’s quantum processors.

b. Neuromorphic Computing

- **Characteristics**: Mimics the neural structure of the human brain; used for AI and machine learning applications.
- **Examples**: Intel Loihi, IBM TrueNorth.

c. Optical Computing

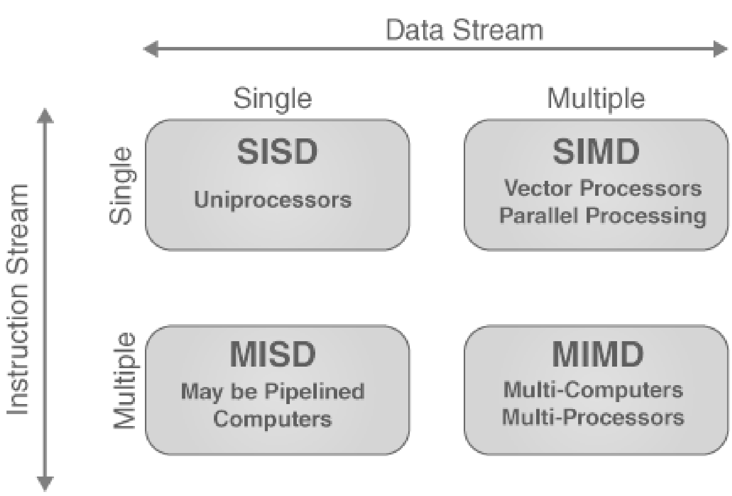
- **Characteristics**: Uses light instead of electrical signals for computation; potential for very high-speed processing.
- **Examples**: Research prototypes and specialized applications.

This taxonomy helps categorize and understand the various approaches to designing and implementing computer systems, each suited to different applications and requirements.

Flynn's Classification

M.J. Flynn offered a classification for a computer system's organisation based on the number of instructions as well as data items that are changed at the same time.

An instruction stream is a collection of instructions read from memory. A data stream is the result of the actions done on the data in the processor. The term 'stream' refers to the flow of data or instructions. Parallel processing can happen in the data stream, the instruction stream, or both.



Flynn's Classification of Computers

Computers can be divided into the following major groups according to Flynn's Classification:

SISD

It depicts the structure of a single computer, which includes a control unit, a memory unit, and a processor unit. Read more on SISD here.

SIMD

It symbolises an organisation with a large number of processing units overseen by a central control unit. Read more on SIMD here.

MISD

Because no real system has been built using the MISD structure, it is primarily of theoretical importance. Read more on MISD here.

MIMD

All processors of a parallel computer could execute distinct instructions and act on different data at the same time in this configuration.