Lecture#1

Instruction set :

The **instruction set** is a fundamental part of a CPU's architecture that defines the commands the processor can execute. It includes two primary components:

1. **Format of the Instruction**: This refers to the structure of an individual instruction, specifying how the instruction is encoded in binary. The format typically includes:
   - **Opcode**: Identifies the operation to be performed (e.g., ADD, LOAD, STORE).
   - **Operands**: Specify the data to be operated on. These may be registers, memory locations, or immediate values.
   - **Addressing Mode**: Determines how the operands are accessed or interpreted (e.g., directly, through a memory address, or through a register).
2. **Semantics of the Instruction**: This refers to the meaning or the actual operation that the instruction performs when executed. The semantics explain how the CPU will execute the instruction and manage data, including the addressing modes used.

Addressing Modes

The **addressing modes** describe how an instruction specifies the location of its operands. Common addressing modes include:

- **Immediate Addressing**: The operand is a constant value embedded within the instruction.
- **Register Addressing**: The operand is in a CPU register.
- **Direct Addressing**: The operand is located at a specific memory address.
- **Indirect Addressing**: The instruction specifies a memory location that contains the actual memory address of the operand.
- **Indexed Addressing**: Combines a base address from a register and an offset value.

The Register Set:

A **register set** is a group of registers that the CPU can use to temporarily store and manipulate data. Registers are fast storage locations inside the processor, and instructions can operate directly on these registers. Registers are much faster to access than memory and play a critical role in executing programs efficiently.

- **Registers** can store data such as numbers, memory addresses, or other instructions.
- The instruction set defines which registers can be used for specific operations.

For example, the **MIPS** architecture uses registers named $r0, $r1, $r31, etc., and instructions are designed to operate on these registers. In MIPS, some registers may have specific roles, like holding temporary values, program counters, or function return addresses.

In summary, the **instruction set** provides the blueprint for the operations the CPU can perform, while the **register set** defines the working space where these operations take place.

Microarchitecture:

Microarchitecture is a layer between the instruction set architecture (ISA), which defines the commands a CPU can execute, and the physical implementation of the CPU.

DRAM:

Dynamic RAM (DRAM) and its variants are common types of memory used in PCs and servers. Here's a breakdown of each type:

1. **DRAM (Dynamic RAM)**:

- **Function**: DRAM is a type of memory that stores each bit of data in a tiny capacitor. These capacitors leak electricity over time, so the data needs to be constantly refreshed (hence "dynamic").
- **Usage**: It's used as the main memory in computers because it can store a lot of data at a relatively low cost.
- **Speed**: DRAM is slower than other types of memory because of the constant refreshing.

2. **SDRAM (Synchronous Dynamic RAM)**:

- **Function**: SDRAM is a type of DRAM that is synchronized with the system clock of the CPU, meaning it operates in sync with the processor’s operations. This synchronization improves performance.
- **Usage**: Commonly used in PCs in the late 1990s and early 2000s.
- **Speed**: Faster than standard DRAM because it can respond to memory requests more predictably.

3. **DDR SDRAM (Double Data Rate SDRAM)**:

- **Function**: DDR SDRAM is a type of SDRAM that doubles the data rate by transferring data on both the rising and falling edges of the clock cycle, allowing it to send and receive data twice as fast.
- **Usage**: Widely used in modern PCs and servers.
- **Versions**: It has several versions, like DDR2, DDR3, DDR4, and DDR5, with each version being faster and more efficient than the previous one.
- **Speed**: Significantly faster than regular SDRAM.

4. **RDRAM (Rambus Dynamic RAM)**:

- **Function**: RDRAM was developed by Rambus Inc. and was designed to be faster than SDRAM by using a high-speed data bus. It has multiple data channels and allows for faster data transfers.

- **Usage**: It was briefly used in high-end PCs and gaming consoles like the PlayStation 2 but has largely been phased out due to its high cost and the rise of DDR SDRAM.
- **Speed**: It was fast for its time but is now considered outdated compared to modern DDR SDRAM.

Note:

A register is composed of **multiple flip-flops**, which are electronic circuits capable of storing a single bit of information, which is represented through binary data – either a 0 or a 1.