Unit - 4
Instruction Set Architecture

Instruction Set:

- Instruction set is the interface between hardware and software.
- The instruction set provides commands to the processor, to tell it what it needs to do.
- The instruction set consists of addressing modes, instructions, native data types, registers, memory architecture, interrupt, and exception handling, and external I/O.
- The instruction set, also called ISA (instruction set architecture), is part of a computer that pertains to programming, which is basically machine language.

Example:

An example of an instruction set is the x86 instruction set, which is common to find on computers today. Different computer processors can use almost the same instruction set while still having very different internal design.

Both the Intel Pentium and AMD Athlon processors use nearly the same x86 instruction set. An instruction set can be built into the hardware of the processor, or it can be emulated in software, using an interpreter. The hardware design is more efficient and faster for running programs than the emulated software version.

Examples of instruction set

ADD - Add two numbers together.
COMPARE - Compare numbers.
IN - Input information from a device, e.g., keyboard.
JUMP - Jump to designated RAM address.
JUMP IF - Conditional statement that jumps to a designated RAM address.
LOAD - Load information from RAM to the CPU.
OUT - Output information to device, e.g., monitor.
STORE - Store information to RAM.

Choice of Instruction Set:

An instruction set can be defined as a group of instruction that a processor can execute to perform different operations. On the basis of complexity and the number of instruction used the instruction set can be classified as

- **Complex Instruction set**
- **Reduced Instruction set**

Complex Instruction set

The complex instruction set is the set of instruction which includes very complex and large number of instructions. The number of instruction in this set varies from 100 to 250. The instruction in this set is mostly memory based instruction, which involves frequent to the memory. The complex instruction set makes use of a large number of addressing modes because of the frequent references to register as well as memory. The instruction in this instruction takes a lot of time because the instructions format which is not limited to only 32 bit.

The complex which makes use of complex instruction set is called Complex Instruction set Computer (CISC). The instruction set of CISC has a large number of instruction and for each instruction type.

Advantages of Complex Instruction Set

- There is no need to invent an instruction set for each new design. The instruction set of the predecessor is used by the new user.
- A program written in CISC require less memory space as the code is confined to less number of instruction.
- CISC makes the job of a computer easier by facilitating the implementation of high-level language constructs.

Disadvantages of Complex Instruction set

- The inheritance of old instruction into new processor increases the complexity.
- Many CISC instructions are not frequently used.
- CISC command is translated into a large number of lines of microcode which makes the CPU processing slower.
- CISC system has a complex hardware, so they require more times for designing.

Reduced Instruction Set

The reduced instruction set refers to a set of instruction that contains very few instructions ranging from 0 to 100. It comprises only that instruction that is frequently used by the processor for the execution of a program. The instruction is generally very simple to execute. The memory-based instruction includes the only load and stores instructions. The instruction in this instruction set has fixed length instruction format of 32 bits. An instruction format divides the bits of instructions into small group fields. Generally, an instruction has the following fields:

Op code field

It represents the operations to be performed by the instruction.

Operand field

It represents the data on which the operations are to be performed or the memory location or registers where the data is stored.

Mode field

It represents the method of fetching the operands stored at specified memory location or register.

The comparison of RISC and CISC processor indicated that the RISC processor is always preferred over the CISC processor because of their compact size and small instruction set.

Advantages of Reduced Instruction set computer

The other advantages of RISC processor over the CISC processor are as follow:

- In RISC processor the instructions are executed by decoding but in CISC processors, the instructions are executed by first and then translating them into equivalent microcode instructions. The conversion of instruction into microcode consumes a lot of space in the memory thereby reducing the speed of executions.
- The RISC processor executes an instruction in a single clock cycle, while the CISC processor requires multiple clocks cycles for the execution of an instruction.

Disadvantages of Reduced Instruction set computer

The only disadvantages of RISC in comparison to CISC is that the number of instructions required to perform an operation is comparatively large.

Instruction Word Formats

The type of internal storage in the CPU is the most basic differentiation. The major choices are:

A stack (the operands are implicitly on top of the stack) .
An accumulator (one operand is implicitly the accumulator) .

A set of registers (all operands are explicit either registers or memory locations) .

The code segment C = A + B how it would appear on the classes of instruction sets

Stack	Accumulator	Register
PUSH A	Load A	Load R1,A
PUSH B	ADD B	ADD R1,B
ADD	Store C	Store C,R1
POP C		

Primary advantages and disadvantages of each class of machine

Machine Type	Advantages	Disadvantages
Stack	Simple model of expression evaluation. Good code density.	A stack can't be randomly accessed. It makes it difficult to generate efficient code.
Accumulator	Minimizes internal state of machine. Short instructions	Since accumulator is only temporary storage, memory traffic is highest.
Register	Most general model for code generation	All operands must be named, leading to longer instructions.

The 8085 instruction set is classified into 3 categories by considering the length of the instructions.

In 8085, the length is measured in terms of “byte” rather then “word” because 8085 microprocessor has 8-bit data bus.

Three types of instruction are:

- **1-byte instruction**
- **2-byte instruction**
- **3-byte instruction**

1. One-byte instructions –
In 1-byte instruction, the opcode and the operand of an instruction are represented in one byte.

Example-1:
Task- Copy the contents of accumulator in register B.

Mnemonic- MOV B, A
Opcode- MOV
Operand- B, A
Hex Code- 47H

Binary code- 0100 0111

Example-2:
Task- Add the contents of accumulator to the contents of register B.

Mnemonic- ADD B
Opcode- ADD
Operand- B
Hex Code- 80H
Binary code- 1000 0000

Example-3:
Task- Invert (complement) each bit in the accumulator.

Mnemonic- CMA
Opcode- CMA
Operand- NA
Hex Code- 2FH
Binary code- 0010 1111

Note – The length of these instructions is 8-bit; each requires one memory location. The mnemonic is always followed by a letter (or two letters) representing the registers (such as A, B, C, D, E, H, L and SP).

2. Two-byte instructions –

Two-byte instruction is the type of instruction in which the first 8 bits indicates the opcode and the next 8 bits indicates the operand.

Example-1:
Task- Load the hexadecimal data 32H in the accumulator.

Mnemonic- MVI A, 32H
Opcode- MVI
Operand- A, 32H
Hex Code- 3E
32
Binary code- 0011 1110
0011 0010

Example-2:
Task- Load the hexadecimal data F2H in the register B.

Mnemonic- MVI B, F2H
Opcode- MVI
Operand- B, F2H
Hex Code- 06

F2
Binary code- 0000 0110
1111 0010

Note – This type of instructions need two bytes to store the binary codes. The mnemonic is always followed by 8-bit (byte) data.

3. Three-byte instructions –

Three-byte instruction is the type of instruction in which the first 8 bits indicates the opcode and the next two bytes specify the 16-bit address. The low-order address is represented in second byte and the high-order address is represented in the third byte.

- **Example-1:**
  Task- Load contents of memory 2050H in the accumulator.

Mnemonic- LDA 2050H
Opcode- LDA
Operand- 2050H
Hex Code- 3A
50
20
Binary code- 0011 1010
0101 0000
0010 0000

Example-2:
Task- Transfer the program sequence to the memory location 2050H.

Mnemonic- JMP 2085H
Opcode- JMP
Operand- 2085H
Hex Code- C3
85
20
Binary code- 1100 0011
1000 0101
0010 0000