# Unit-3 part-1

# Topic :a.Addressing Modes

# b.Data Transfer and Manipulation

Most computer instructions can be classified into three categories:

1. Data transfer instructions.
2. Data manipulation instructions.
3. Program control instructions.

# Addressing Modes

Computers use addressing mode techniques for the purpose of accommodating one or both of the following provisions:

1. To give programming versatility to the user by providing such facilities as pointers to memory, counters for loop control, indexing of data, and program relocation.
2. To reduce the number of bits in the addressing field of the instruction.

PC holds the address of the instruction to be executed next and is incremented each time an instruction is fetched from memory. An example of an instruction format with a distinct addressing mode field is shown in Fig(27).



**Immediate Mode:** In this mode the operand is specified in the instruction itself. In other words, an immediate-mode instruction has an operand field rather than an address field.

**Register Mode:** In this mode the operands are in registers that reside within the CPU. The particular register is selected from a register field in the instruction.

**Register Indirect Mode:** In this mode the instruction specifies a register in the CPU whose contents give the address of the operand in memory.

**Auto-increment or Auto-decrement Mode:** This is similar to the register indirect mode except that the register is incremented or decremented after (or before) its value is used to access memory.

The effective address is defined to be the memory address obtained from the computation dictated by the given addressing mode.

**Direct Address Mode:** In this mode the effective address is equal to the address part of the instruction.

**Indirect Address Mode:** In this mode the address field of the instruction gives the address where the effective address is stored in memory.

The effective address in these modes is obtained from the following computation:

*effective address = address part of instruction + content of CPU register*

**Relative Address Mode:** In this mode the content of the program counter is added to the address part of the instruction in order to obtain the effective address.

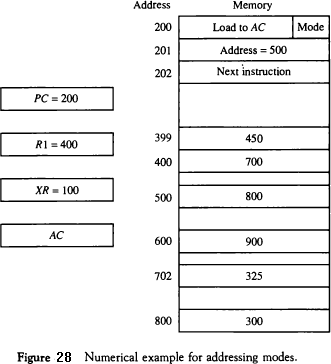
**Indexed Addressing Mode:** In this mode the content of an index register is added to the address part of the instruction to obtain the effective address.

**Base Register Addressing Mode:** In this mode the content of a base register is added to the address part of the instruction to obtain the effective address.

## Numerical Example:

* 1. The two-word instruction at address 200 and 201 is a "load to AC" instruction with an address field equal to 500.
  2. The first word of the instruction specifies the operation code and mode, and the second word specifies the address part.
  3. PC has the value 200 for fetching this instruction.
  4. The content of processor register Rl is 400, and the content of an index register XR is 100.

AC receives the operand after the instruction is executed. The Fig(28) lists a few pertinent addresses and shows the memory content at each of these addresses.

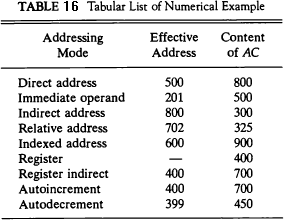


## Answer:

1. *In the direct address mode* the effective address is the address part of the instruction 500 and the operand to be loaded into AC is 800.
2. *In the immediate mode* the second word of the instruction is taken as the operand rather than an address, so 500 is loaded into AC. (The effective address in this case is 201)
3. *In the indirect mode* the effective address is stored in memory at address 500. Therefore, the effective address is 800 and the operand is 300.
4. *In the relative mode* the effective address is 500 + 202 = 702 and the operand is 325. (Note that the value in PC after the fetch phase and during the execute phase is 202)
5. *In the index mode* the effective address is XR + 500 = 100 + 500 = 600 and the operand is 900.
6. *In the register mode* the operand is in Rl and 400 is loaded into AC. (There is no effective address in this case)
7. *In the register indirect mode* the effective address is 400, equal to the content of Rl and the operand loaded into AC is 700.
8. *The auto-increment mode* is the same as the register indirect mode except that Rl is incremented to 401 after the execution of the instruction.
9. *The auto-decrement mode* decrements Rl to 399 prior to the execution of the instruction.

The operand loaded into AC is now 450.

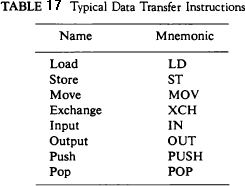
Table (16) lists the values of the effective address and the operand loaded into AC for the nine addressing modes.



# Data Transfer and Manipulation

Most computer instructions can be classified into three categories:

1. Data transfer instructions.
2. Data manipulation instructions.
3. Program control instructions.

*Data transfer instructions* cause transfer of data from one location to another without changing the binary information content. The table(17) list the Data transfer instructions:

*Data manipulation instructions* are those that perform arithmetic, logic, and shift operations. The data manipulation instructions in a typical computer are usually divided into three basic types:

1- Arithmetic instructions.

1. Logical and bit manipulation instructions.
2. Shift instructions.

**Reduced Instruction Set Computer (RISC)**

An important aspect of computer architecture is the design of the instruction set for the processor. The instruction set chosen for a particular computer determines the way that machine language programs are constructed. A computer with a large number of instructions is classified as a complex instruction set computer, abbreviated CISC. In the early 1980s, a

number of computer designers recommended that computers use fewer instructions with simple constructs so they can be executed much faster within the CPU without having to use memory as often. This RISC type of computer is classified as a reduced instruction set computer or RISC.

In summary, the major characteristics of CISC architecture are:

1. A large number of instructions—typically from 100 to 250 instructions.
2. Some instructions that perform specialized tasks and are used infrequently.
3. A large variety of addressing modes—typically from 5 to 20 different modes.
4. Variable-length instruction formats.
5. Instructions that manipulate operands in memory. The major characteristics of a RISC processor are:
6. Relatively few instructions.
7. Relatively few addressing modes.
8. Memory access limited to load and store instructions.
9. All operations done within the registers of the CPU.
10. Fixed-length, easily decoded instruction format.
11. Single-cycle instruction execution.
12. Hardwired rather than microprogrammed control.