**Addressing Modes**

**Addressing Modes– The term addressing modes refers to the way in which the operand of an instruction is specified. The addressing mode specifies a rule for interpreting or modifying the address field of the instruction before the operand is actually executed.**

**The operation field of an instruction specifies the operation to be performed. This operation will be executed on some data which is stored in computer registers or the main memory. The way any operand is selected during the program execution is dependent on the addressing mode of the instruction.**

**The purpose of using addressing mode techniques is as follows:**

* **To give the programming flexibility to the user by providing some facilities.**
* **To reduce the number of bits in an addressing field of the instruction.**

**To understand the various addressing modes, it is imperative that we should understand the basic operation cycle of the computer.**

**Instruction Cycle**

**An instruction cycle, also known as fetch-decode-execute cycle is the basic operational process of a computer. This process is repeated continuously by CPU from boot up to shut down of computer.**

**Following are the steps that occur during an instruction cycle:**

1. **Fetch the Instruction from memory**
2. **Decode the Instruction**
3. **Execute the Instruction**

**1. Fetch the Instruction from memory**

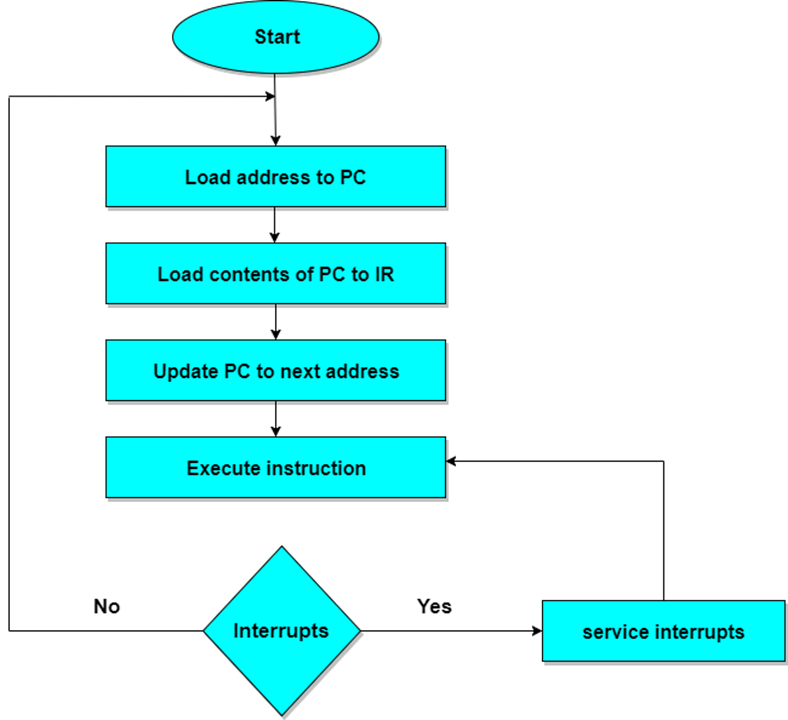
**There is one register in the computer called the program counter or PC that keeps track of the instructions in the program stored in the memory. PC holds the address of the instruction to be executed next and is incremented each time an instruction is fetched from memory.**

**2. Decode the Instruction**

**The decoding done in step 2 determines the operation to be performed, the addressing mode of the instruction, and the location of the operands.**

**3. Execute the Instruction**

**The computer then executes the instruction and returns to step 1 to fetch the instruction in sequence. The Control Unit passes the information in the form of control signals to the functional unit of CPU. The result generated is stored in main memory or sent to an output device.**

****

**The cycle is then repeated by fetching the next instruction. Thus in this way the instruction cycle is repeated continuously.**

**An example of instruction format with a distinct addressing mode field is shown in Fig 8-6.**

****

**The opcode specifies the operation to be performed.**

**The mode field is used to locate the operands needed for the operation.**

**There may or may not be an address field in the instruction. If there is an address field, it may designate a memory address or processor register.**

**Types of addressing modes**

**Implied mode: In implied addressing the operands are specified implicitly in the definition of the instruction. In this mode the data is 8 bits or 16 bits long and data is the part of instruction. Zero address instructions are designed with implied addressing mode.**

**Example: CMA means Complement Accumulator**

**CLC (used to reset Carry flag to 0)**

**Immediate addressing mode: In this mode the operand is specified in the instruction itself. Designed like one address instruction format.**

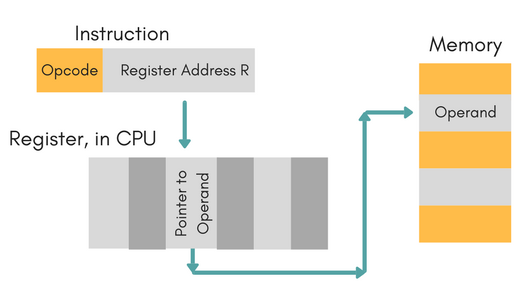
**Example: MOV AL, 35H (move the data 35H into AL register)**

**Register mode: In this mode the operands are in registers that reside within the CPU. The data is in the register that is specified by the instruction.**

**Example: MOV AX, CX (move the contents of CX register to AX register)**

**Register indirect mode: In this mode the instruction specifies a register in the CPU whose contents give the address of the operand in memory. In other words, the selected register contains the address of the operand rather than the operand itself.**

**MOV AX, [BX] (move the contents of memory locations addressed by the register BX to the register AX)**

****