ADDRESSING MODES

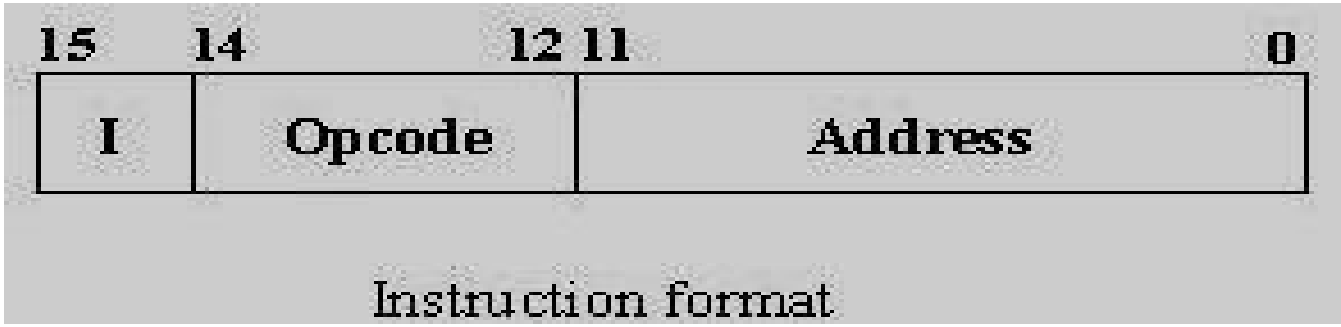
Addressing Mode

Addressing mode: It specifies The rule for interpreting or modifying the address field of an instruction. It provides flexibility with respect to number of instructions and execution time.

Effective address: The address of operand produced by interpreting or modifying the address field of the instruction before the operand is actually referenced.

Addressing Mode

- To give programming flexibility to user
- To reduce the number of bits in the address fields of the instruction



Instruction format

Types of Addressing Mode

- **Implied Mode**
- **Immediate Mode**
- **Register Direct Mode**
- **Register Indirect mode**
- **Direct addressing mode**
- **Indirect addressing mode**
- **Relative addressing mode**
- **Indexed addressing mode**

Addressing Mode

- Implied mode: Needs no address field, the operand is specified implicitly in the definition of the opcode.
- For example, **ADD in a stack computer**, CLA, CLE etc
- Immediate mode: An instruction has an operand field rather than an address field
- For example, **Add R4, #3**
- This instruction adds the content of R4 with 3 and stores the result in R4 by overwriting its previous content.

Addressing Mode

Register direct mode:

– The address field specifies a processor register
  For example: **Add R4, R3**
– This instruction adds the content of registers R4 and R3 and stores the results in R4

Register-indirect mode:

– This instruction specifies a register in the processor whose content gives the address of the operand in the memory.
  - the address field uses fewer bits to select a register than would have been required to specify a memory address directly
  - For example**, Add R4, (R1)**
  - This instruction performs the microoperation R4 <- R4 + M[R1]

Addressing Mode

- **Direct addressing mode**:
- The address field of the instruction gives the address of the operand in memory directly in the instruction itself.
- For example:Add R4, (1000)
- This instruction performs the microoperation

R4 <- R4 + M[1000]

Addressing Mode

- **Indirect addressing mode**:
- The address field of the instruction gives the address at which the effective address is stored in memory. For example: **Add R4, @(1000)**
- This instruction performs the microoperation

R4 <- R4 + M[M[1000]]

- **Relative addressing mode**:
- Effective address= address part of the instruction + contents of PC
- Eg. If PC=825, if address of instruction =24, If PC is fetched, then now PC=826+24=850.
- It is used with branch type instr.

Addressing Mode

- **Indexed addressing mode**:
- The content of an indexed register is added to obtain the effective address.
- The index register contains the index value.

Each operand in array is stored in memory relative to beginning address.

- The distance between the beginning address and the address of the operand is the index value stored in the index register.
- For example: **Add R4, (R1 + R2)**
- This instruction performs the microoperation

R4 <- R4 + M[R1 + R2]

Addressing Mode

Addressing Mode	Example	Action
1. Register direct	Add R4, R3	R4 <- R4 + R3
2. Immediate	Add R4, #3	R4 <- R4 + 3
3. Displacement	Add R4, 100(R1)	R4 <- R4 + M[100 + R1]
4. Register indirect	Add R4, (R1)	R4 <- R4 + M[R1]
5. Indexed	Add R4, (R1 + R2)	R4 <- R4 + M[R1 + R2]
6. Direct	Add R4, (1000)	R4 <- R4 + M[1000]
7. Memory Indirect	Add R4, @(1000)	R4 <- R4 + M[M[1000]]