

VIVEKANANDA COLLEGE
THAKURPUKUR
KOLKATA-700063
NAAC ACCREDITED 'A' GRADE



Topic: Addressing Modes in 8085

Course Title: Microprocessor and its Application

Paper: CMS-A-CC-4-10-TH

Unit: **Microprocessor Architecture and Memory Interfacing**

Semester: 4th Semester

Name of the Teacher: Amitava Biswas

Name of the Department: Computer Science

Addressing Modes of 8085 Microprocessor

- Introduction:

Every instruction is divided into two parts, one part is opcode and another part is operand. Opcode tells the type of operation and operand is the data on which operation is to be performed. The way in which operand is specified in an instruction is called addressing mode.

Addressing modes specifies the way, the effective address of an operand is represented in the instruction. Some addressing mode efficiently allows referring to a large range of area like a linear array of addresses and list of addresses. Addressing mode describes a flexible and efficient way to define complex effective address.

- Addressing Modes of 8085 Microprocessor:

The process of identifying the operands for a particular instruction can be carried out in several ways. The various types of addressing modes are as shown below.

1. Immediate addressing mode
2. Direct addressing mode
3. Register addressing mode
4. Register indirect addressing mode
5. Implicit addressing mode
6. Immediate Addressing Mode:

In this type of addressing mode the operand is specified within the instruction itself.

Let us discuss with an example.

Consider this instruction:

ADI 12H – This instruction adds the immediate data, 34H to the accumulator.

12H is the data here. H represents Hexadecimal value and the immediate value is added to the accumulator. In this case 12H is added to the accumulator.

Suppose if accumulator has a value 2H and when this instruction is executed, 12H is added to the 2H and the result is stored in accumulator. In the above instruction the operand is specified within instruction itself.

1. Direct Addressing Mode:

In this mode of addressing, the address of the data (operand) is specified within the instruction.

There is a subtle difference between the direct addressing modes and immediate addressing modes. In immediate addressing mode the data itself is specified within instruction, but in direct addressing mode the address of the data is specified in the instruction.

Example:

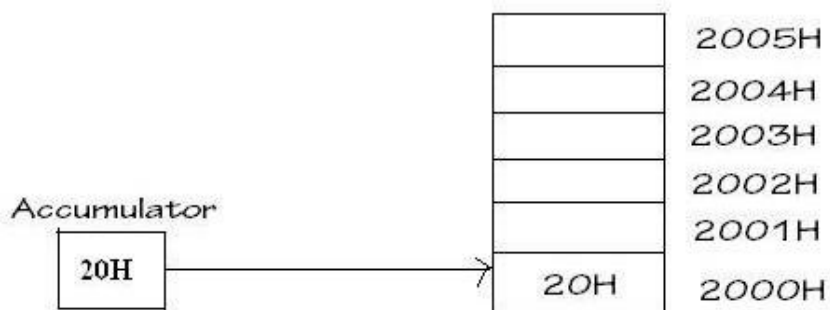
OUT 10H

LDA F100H

STA 2000H

Consider the instruction STA 2000H

When this instruction is executed, the contents of the accumulator are stored in the memory location specified. In the above example the contents of accumulator are stored in memory location 2000H.



2. Register Addressing Mode:

In this type of addressing mode the instruction specifies the name of the register in which the data is available and Opcode specifies the name (or) address of the register on which the operation would be performed.

Example:

MOV A, B

Here the Opcode is MOV. If the above instruction is executed, the contents of Register B are moved to the Register A, which is nothing but the accumulator.



Other examples:

ANA B

On executing the above instruction the contents of Register B or logically ANDed with contents of register A (accumulator).

SUB H

If we execute the above instruction the contents of Register H will be subtracted from the contents of the accumulator.

3. Register Indirect Addressing Mode:

This is indirect way of addressing. In this mode the instruction specifies the name of the register in which the address of the data is available.

Example:

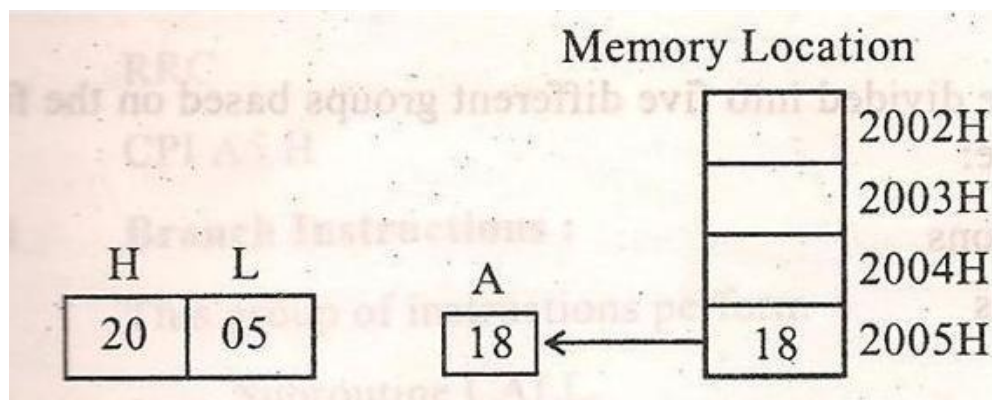
MOV A, M

SUB M DCR

M

Consider MOV A, M. This instruction will move the contents of memory location, whose address is in H-L register pair to the accumulator.

M represents the address present in the H-L register pair. So when MOV A, M is executed, the contents of the address specified in H-L register pair are moved to accumulator.



4. Implicit Addressing Mode:

There are certain instructions in 8085 which does not require the address of the operand to perform the operation. They operate only upon the contents of accumulator.

Example:

CMA RAL

RAR

CMA complements the contents of accumulator.

If RAL is executed the contents of accumulator is rotated left one bit through carry.

If RAR is executed the contents of accumulator is rotated right one bit through carry.

Type	Instruction	Source	Destination
Direct	STA 2005 H	Accumulator	Memory 2005H
Register	MOV A, B	Register B	Accumulator
Register Indirect	MOV A, M	Memory [[HL]]	Accumulator
Immediate	MVI A, 18H	Data 18H	Accumulator
Implicit	CMA		

