**MODULE2**

1. Draw and discuss the instruction format of an instruction

containing immediate operand to memory with 16 bit

displacement?

**Answer**

Instruction format – *page no.34 refer text*. Need figure of the format and explanation.

1. Explain the indexed addressing and based index addressing

modes of 8086 with examples.

**Answer**

Indexed Addressing mode : In this mode, offset of the operand is stored in one of the index registers.

Ex: MOV AX[SI]. Here data is available at the offset address stored in SI in DS. Address is found as 10H \* DS + [SI]

Based Indexed mode: In this mode, address is found by adding the content of a base register to the content of any index register.

Ex: MOV AX,[BX],[SI] Address found by 10H\*DS+{BX]+[SI] .

1. Which are the addressing modes for control transfer instructions?

**Answer**

Addressing modes for control transfer instructions

i) Intersegment

ii) Intrasegment

These are again divided as intersegment direct and intersegment

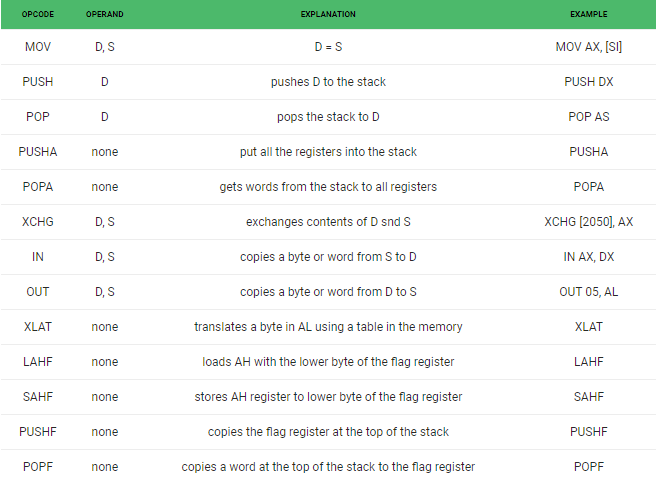
Indirect and intrasegment direct intrasegment indirect.

Refer text page no.38-39.

1. Describe the data transfer instructions associated with a stack?

**Answer**

Data transfer instructions are the instructions which transfers data in the microprocessor. They are also called copy instructions.

****

1. What is the difference between register and register indirect

addressing mode?

**Answer**

**Register mode:**In register addressing the operand is placed in one of 8 bit or 16 bit general purpose registers.

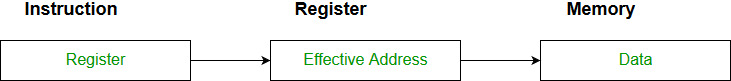
The data is in the register that is specified by the instruction.

Here one register reference is required to access the data.

am3

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

**Register Indirect mode**: In this addressing the operand’s offset is placed in any one of the registers BX,BP,SI,DI as specified in the instruction. The effective address of the data is in the base register or an index register that is specified by the instruction.  
Here two register reference is required to access the data.

[](https://media.geeksforgeeks.org/wp-content/cdn-uploads/Addressing_Modes_4.jpg)

The 8086 CPUs let you access memory indirectly through a register using the register indirect addressing modes.

MOV AX, [BX](move the contents of memory location s

addressed by the register BX to the register AX)

1. Which are the different types of instructions in 8086?

**Answer**

Different instructions in 8086 are

a. Data transfer instructions

b. Arithmetic instruction

c. String manipulation instruction

d. Logical instruction

e. Shift and rotate instruction

f. Flag manipulation instruction

1. Explain the addressing modes for sequential control transfer

instructions?

**Answer**

Sequential control flow instructions are the instructions which after execution, transfer control to the next instruction appearing immediately after it (in the sequence) in the program. For example the arithmetic, logic, data transfer and processor control instructions are Sequential control flow instructions.

The control transfer instructions on the other hand transfer control to some predefined address or the address somehow specified in the instruction, after their execution. For example INT, CALL, RET & JUMP instructions fall under this category.

The addressing modes for Sequential and control flow instructions are explained as follows.

1. **Immediate addressing mode:** In this type of addressing, immediate data is a part of instruction, and appears in the form of successive byte or bytes. Example: MOV AX, 0005H. In the above example, 0005H is the immediate data. The immediate data may be 8- bit or 16-bit in size.

2. **Direct addressing mode:** In the direct addressing mode, a 16-bit memory address (offset) directly specified in the instruction as a part of it. Example: MOV AX, [5000H].

3. **Register addressing mode:** In the register addressing mode, the data is stored in a register and it is referred using the particular register. All the registers, except IP, may be used in this mode. Example: MOV BX, AX.

4. **Register indirect addressing mode:** Sometimes, the address of the memory location which contains data or operands is determined in an indirect way, using the offset registers. The mode of addressing is known as register indirect mode. In this addressing mode, the offset address of data is in either BX or SI or DI Register. The default segment is either DS or ES. Example: MOV AX, [BX].

5. **Indexed addressing mode:** In this addressing mode, offset of the operand is stored one of the index registers. DS & ES are the default segments for index registers SI & DI respectively. Example: MOV AX, [SI] Here, data is available at an offset address stored in SI in DS.

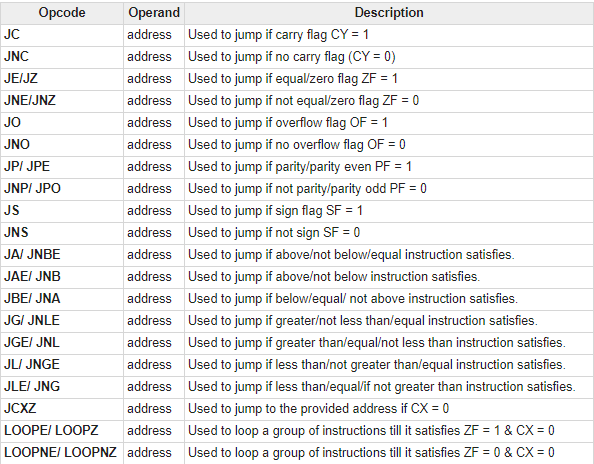
6. **Register relative addressing mode:** In this addressing mode, the data is available at an effective address formed by adding an 8-bit or 16-bit displacement with the content of any one of the register BX, BP, SI & DI in the default (either in DS & ES) segment. Example: MOV AX, 50H [BX].

7. **Based indexed addressing mode**: The effective address of data is formed in this addressing mode, by adding content of a base register (any one of BX or BP) to the content of an index register (any one of SI or DI). The default segment register may be ES or DS. Example: MOV AX, [BX][SI].

8. **Relative based indexed:** The effective address is formed by adding an 8 or 16-bit displacement with the sum of contents of any of the base registers (BX or BP) and any one of the index registers, in a default segment. Example: MOV AX, 50H [BX] [SI] For the control transfer instructions, the

1. Explain conditional branch instructions of 8086.

**Answer**



1. What are the different addressing modes supported by 8086?

Give explanation with suitable examples.

**Answer**

**Refer question 7 and,**

Addressing Modes for control transfer instructions:

* Intersegment :

Intersegment direct

Intersegment indirect

* Intrasegment:

Intrasegment direct

Intrasegment indirect

* **Intersegment direct:** In this mode, the address to which the control is to be transferred is in a different segment. This addressing mode provides a means of branching from one code segment to another code segment. Here, the CS and IP of the destination address are specified directly in the instruction.
* **Intersegment indirect:** In this mode, the address to which the control is to be transferred lies in a different segment and it is passed to the instruction indirectly, i.e. contents of a memory block containing four bytes, i.e. IP(LSB), IP(MSB), CS(LSB) and CS(MSB) sequentially. The starting address of the memory block may be referred using any of the addressing modes, except immediate mode.

Example: JMP [2000H]. Jump to an address in the other segment specified at effective address 2000H in DS

* **Intrasegment direct mode:** In this mode, the address to which the control is to be transferred lies in the same segment in which the control transfers instruction lies and appears directly in the instruction as an immediate displacement value. In this addressing mode, the displacement is computed relative to the content of the instruction pointer. The effective address to which the control will be transferred is given by the sum of 8 or 16 bit displacement and current content of IP. In case of jump instruction, if the signed displacement (d) is of 8-bits (i.e. -128<d< +127), it as short jump and if it is of 16 bits (i.e. -32768<d< +32767), it is termed as long jump. Example: JMP SHORT LABEL.
* **Intrasegment indirect mode:** In this mode, the displacement to which the control is to be transferred is in the same segment in which the control transfer instruction lies, but it is passed to the instruction directly. Here, the branch address is found as the content of a register or a memory location. This addressing mode may be used in unconditional branch instructions. Example: JMP [BX]; Jump to effective address stored in BX.

1. The contents of different registers are given below:-

Offset (displacement)=5000H, [AX]=1002H, [BX]=3024H,

[SI]=2000H, [DS]=1012H.

Calculate the effective addresses for different addressing

modes.

a) Direct b) Register indirect c) Register relative d) Based

indexed e) Relative based indexed.

**Answer**

1. EA = 10H \* DS + 5000H

= 10 \* 1012 + 5000

**EA = 15120H**

1. EA = 10H \* DS + [BX]

= 10 \* 1012 + 3024

**EA = 13144H**

1. EA = (10H \* DS) + 5000H + [BX]

= (10\*1012) + 5000 + 3024

**EA = 18144H**

1. EA = (10H \* DS) + [BX] + [SI]

= (10 \* 1012) + 3024 + 2000

**EA = 15144H**

1. EA = (10 \* DS) +5000H + [BX] + [SI]

=(10 \* 1012) + 5000 + 3024 +2000

**EA = 1A144H**

1. Explain Data Copy/Transfer instruction set in 8086.

**Answer**

Instruction to transfer a word

* **MOV** − Used to copy the byte or word from the provided source to the provided destination.
* **PPUSH** − Used to put a word at the top of the stack.
* **POP** − Used to get a word from the top of the stack to the provided location.
* **PUSHA** − Used to put all the registers into the stack.
* **POPA** − Used to get words from the stack to all registers.
* **XCHG** − Used to exchange the data from two locations.
* **XLAT** − Used to translate a byte in AL using a table in the memory.

Instructions for input and output port transfer

* **IN** − Used to read a byte or word from the provided port to the accumulator.
* **OUT** − Used to send out a byte or word from the accumulator to the provided port.

Instructions to transfer the address

* **LEA** − Used to load the address of operand into the provided register.
* **LDS** − Used to load DS register and other provided register from the memory
* **LES** − Used to load ES register and other provided register from the memory.