Faculty of Engineering and Technology

University of Lucknow

Er. Priyanka Singh

B.Tech –IV Sem

Fundamentals of Microprocessor (EC-403)

Additional Data transfer 16 bit Arithmetic Instruction

1) 16 bit Data transfer to register pair (LXI)

opcode	operand
LXI	Rp, 16 bit
LXI	B, 16 bit
LXI	D, 16 bit
LXI	H, 16 bit

Load register pair
- This is a 3 byte instruction
The second byte is loaded in the low order register of the register pair
The third byte is loaded in the high order register pair

LXI SP, 16 bit → ~~There are four such instruction in the set as shown~~ the operands B, D & H represent BC, DE & HL registers & SP represents the stack pointer register.

2) Data transfer from memory to the Microprocessor.

a) MOV R, M (Move) from memory to Register
This is 1 byte instruction.
It copies data byte from memory location into a register
R represent microprocessor register A, B, C, D, E, H & L.
→ the memory location is specified by the contents of HL register.
→ Indirect addressing mode.

b) LDAX B/D : Load Acc. Indirect
→ This is a 1 byte instruction
LDAX B → It copies the data byte from memory location into the accumulator.
LDAX D → The memory location is specified by content of register BC or DE.
→ indirect addressing mode.

c) LDA 16 bit :- Load Accumulator Direct.
→ This is a 3 byte instruction.
It copies data byte from memory location specified by bit address in second & third byte
→ direct.

The memory location 2050H holds the data byte F7H. write instructions to transfer the data byte to the accumulator using three different opcodes MOV, LDAX & LDA.

1) LXI H 2050H.
   MOV A M

2) LXI B, 2050H.
   LDAX B

3) LDA 2050H

Data transfer from Microprocessor to Memory

1) MOV M, R (Move from Register to memory)
This is 1 byte instruction that copies data from a register R into the memory location specified by content of HL register.

2) STAX B/D Store Accumulator Indirect
STAX B / STAX D This is 1 byte instruction that copies data from accumulator into the memory location specified by content of either BC or DE registers.

3) STA, 16 bit :- Store Accumulator Direct.
This is a 3 byte instruction that copies data from accumulator into memory location specified by 16 bit operand.

4) MVI M, 8 bit → load 8 bit data in memory.
→ this is a two byte instruction, the second byte specifies 8 bit data. the memory location is specified by content of HL register.

Arithmetic operation Related to 16 bit or Register pair → these instructions do not affect flags.

INX Rp → 1 byte
B
D
H
SP → treats the content of two registers as one 16 bit number & increases the content by 1.

DCX Rp → decreases the 16 bit content of register pair by 1.

Compare

CMP → Compare with Acc.
CPI → Compare Immediate (with Accumulator)

1) CMP R/M : Compare (Register or Memory) with Accumulator
→ this is a 1 byte instruction,
→ It compares the data byte in register or memory with contents of the accumulator.
1) if (A) < (R/M), the CY flag is set & the zero flag is reset.
2) if A = R/M, the zero flag is set & CY flag is reset.
3) if A > (R/M), the CY & zero flags are reset.

when memory is operand it address is specified by HL
S, P, AC

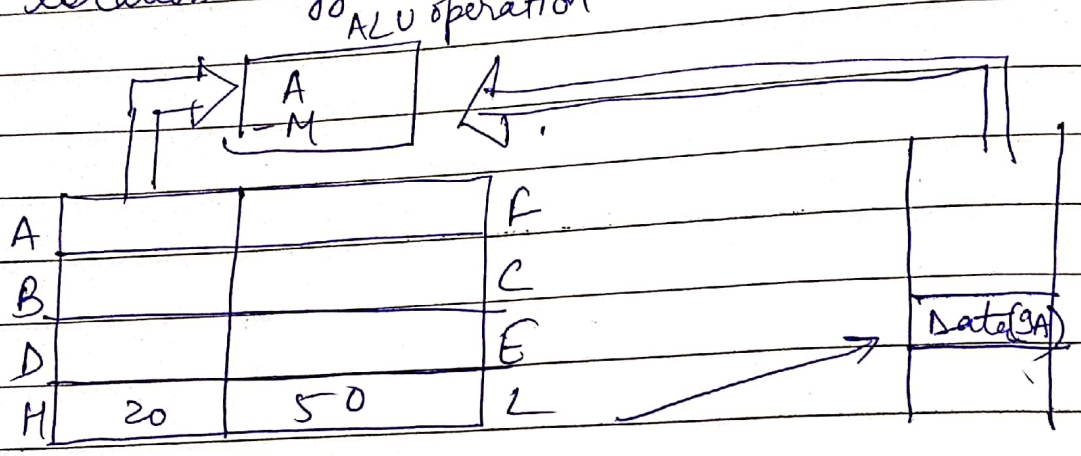
CPI 8 bit Compare immediate with Accumulator
this is a 2 byte instruction, the second byte being 8 bit data.
→ It compares the second byte with A.
→ if A < 8 bit data, the CY flag is set & zero flag is reset.
→ if A = 8 bit data, the zero flag is set & CY flag is reset
→ if A > 8 bit data CY & zero flag are reset.

... instruction to load the acc. with data ... byte in memory ...

Date __/__/____

Page _____

*. Write an Instruction to load the accumulator with data byte 64H & verify whether the data byte in memory location 2050 is equal to the accumulator contents. if both data are equal, jump to memory location Buffer.



LXI H, 2050H

MVI A, 64H.

CMP M

JZ BUFFER

64 0110 0100

2's 9A 0110 0110

0 1100 1010

if there is no Carry then in case of Substraction using 2' Complement we complement the Carry.

So, we get

CY = 1

Z = 0

S = 1

* A Set of three reading is Stored in memory Starting at XX50H, Sort the reading in ascending order.
Data 87H, 56H & 42H

Program:-

```
START : LXI H, XX50H
        MVI D, 00H    Clear register D to set up flags.
        MVI C, 02H    Set register C for Comparison Count
CHECK : MOV A, M      Get data byte
        INX H
        CMP M
        JC NXTBYT     if A < Second, do no Carry.
        MOV B, M      Get Second byte for exchange
        MOV M, A      Store first byte in Second location.
        DCX H         point to first location.
        MOV M, B      Store second byte in first location
        INX H         Get ready for next Comparison.
        MVI D, 01H    Load 1 in D as reminder for exchange.
NXTBYT : DCR C
        JNZ CHECK
        MOV A, D
        RRC
        JC START
```

Data :	1st	2nd	3rd
XX50H	87	56	42	42
XX51H	56	42	56	56
XX52H	42	87	87	87
Register D	0	1	1	0

arranged in ascending order.

Date / /
Page

Additional logic operations

- Rotate
  - → Rotate the contents of the accumulator one position to the left or right

→ RLC → Rotate the accumulator left.
Bit 7 goes to bit 0 AND the Carry flag.

→ RAL Rotate the accumulator left through the Carry. Bit 7 goes to the Carry & Carry goes to bit 0

→ RRC Rotate the accumulator right
Bit 0 goes to bit 7 AND & the Carry flag

→ RAR Rotate the accumulator right through the Carry
Bit 0 goes to the Carry & Carry goes to Bit 7