Republic of Iraq
Ministry of Higher Education and Scientific Research
Al-Furat Al-Awsat Technical University
Engineering Technical College/Najaf
Al Najaf Al Ashraf, 31001. Iraq.

8085 Microprocessor

Lecture 5

المدرس ضرغام الخفاف الاسدي

Third Year lecture notes

Avionics Engineering Dept.

Engineering Technical College/ NAJAF 2020-2021

Lecturer: Dhurgham Al-Khaffaf Alasady

INSTRUCTION SET

The instruction set has been classified into following groups.

1. Data Transfer Group
2. Arithmetic Group
3. Branch Group
4. Stack, Input/Output and Machine Control Group

Data Transfer Group

The function of data transfer group of instructions is to transfer the data from register to register, register to memory and also immediate transfer of data (given) to memory location.

The 8085 includes all instructions which will be discussed below.

***1- LHLD address* (Loads the H-L pair direct)** This instruction loads the H-L pair direct with two bytes already stored in two consecutive memory locations starting at the specified memory address. The contents stored in the memory location whose address is given with the instruction will be loaded to the L-register; and the contents stored in the next memory location (address + 1) will be loaded to the H-register.

$$\text{i.e.} \quad [L] \leftarrow [M_{address}]$$
$$\text{and} \quad [H] \leftarrow [M_{address+1}]$$

For example, let 2A H is stored in the memory location 2100 H and 2B H is stored in the memory location 2101 H, then after the execution of the instruction LHLD 2100 H , the L-register will have 2A H and H-register will have 2B H. None of the flags is affected with this instruction.

***2- SHLD address* (Stores the H-L pair direct)** This instruction does the reverse operation of LHLD. The instruction *SHLD address* stores the contents of L-register to memory location whose address is given with the instruction; and the contents of H-register are stored in the next consecutive memory location (address + 1).

$$\text{i.e.} \quad [M_{address}] \leftarrow [L]$$
$$\text{and} \quad [M_{address+1}] \leftarrow [H]$$

***3- LDAX rp* (Loads the Accumulator Indirect)** This instruction loads the accumulator, the contents already stored in the memory location addressed by the register pair (rp). Here rp represents B-C or D-E register pair. The H-L register pair is not included in this instruction.

i.e. $[A] \leftarrow [M_{rp}]$

4- STAX rp (Stores the Accumulator Indirect) The STAX rp instruction does the reverse operation of LDAX rp. This instruction stores the accumulator contents in the memory location addressed by the register pair (rp). Here too rp represents B-C or D-E register pair. The H-L register pair is not included in this instruction.

i.e. $[M_{rp}] \leftarrow [A]$

5- MOV destination, source (Transfer data)

MOV A, B	78	Register	4	1	None	$[A] \leftarrow [B]$
MOV A, C	79	Register	4	1	None	$[A] \leftarrow [C]$
MOV E, M	5E	Register indirect	7	1	None	$[A] \leftarrow [M_{H-L}]$

6- MVI R/M, 8-bit (Loads 8-bit to Req. or Memory location)

MVI L, data	2E	Immediate	7	2	None	$[L] \leftarrow data$
MVI M, data	36	Immediate	10	2	None	$[M_{H-L}] \leftarrow data$

7- LDA address (Loads the accumulator direct). It transfers the content stored in the addressed memory location (given by address) to accumulator.

$$[A] \leftarrow [M_{address}]$$

No flag is affected in this instruction. It is three byte instruction. For example if 2AH data is stored in memory location 2500H before the execution of LDA 2500H instruction, then after the execution of this instruction, the data 2AH will be transferred to accumulator.

8- STA address (Stores the accumulator direct). It transfers the content stored in the accumulator to addressed memory location (given by address).

$$[M_{address}] \leftarrow [A]$$

No flag is affected in this instruction. It is also three byte instruction. For example if 16H data is stored in the accumulator before the execution of STA 2100H instruction, then after the execution of this instruction, the data 16H will be transferred to the addressed memory location.

i.e. $[M_{2100}] \leftarrow 16$

***9- XCHG* (Exchange the contents of H-L register with D-E register)** This is one byte instruction and no operand is needed with it. It exchanges the contents of H and L register with D and E registers respectively.

i.e. $[H] \leftrightarrow [D]$ and $[L] \leftrightarrow [E]$

For example:

If H = 25 H, L = 32 H

and D = 12 H, E = 1B H

then after the execution of *XCHG* instruction, we have:

H = 12 H, L = 1B H

and D = 25 H, E = 32 H

Example: Let us write a program to add two numbers stored in memory locations 2100 H and 2201 H without using *LDAX* and *STAX* instructions. The answer is to be loaded in the memory location 2100 H.

```
LXI H, 2201 H  ; Loads H = 22 H and L = 01 H
LXI D, 2100 H  ; Loads D = 21 H and E = 00 H
MOV A, M       ; [A] ← [M2201H]
XCHG           ; H = 21 H, L = 00 H and D = 22 H, E = 01 H
ADD M          ; [A] ← [A] + [M2100H]
MOV M, A       ; [M2100H] ← [A]
HLT
```

10- LXI Rp,16-bit (Loads 16-bit to req. pairs)

LXI H,2000 H (H=20 and L=00)

LXI D,2000

LXI B,2000

LXI SP,2000

11- ***PCHL*** **(Copies H-L to PC)** This is one byte instruction and no operand is needed with this instruction. It copies the contents of H-register to high-order byte of the program counter (PC) and the contents of L-register to low order byte of the program counter.

12- ***SPHL*** **(Copies HL to Stack Pointer SP)** This is also one byte instruction as no operand is used. The *SPHL* instruction copies the contents of H-register to high order byte stack pointer (SP) and the contents of L-register to low order byte of stack pointer (SP).

***13- XTHL (*Exchanges the top of the stack with H-L register pair)** The *XTHL* is one byte instruction and does not require any operand. The top byte of the stack is exchanged with L-register and next byte of the stack is exchanged with H register.

i.e. $[L] \leftrightarrow [M_{SP}]$

and $[H] \leftrightarrow [M_{SP+1}]$

For example if $H = 21\ H$ $L = 02\ H$

and $M_{SP} = 1A\ H$ $M_{SP+1} = 2C\ H$

as shown in figure 5.8(a), then after the execution of the instruction *XTHL* will result:

$H = 2C\ H$ $L = 1A\ H$

and $M_{SP} = 02\ H$ $M_{SP+1} = 21\ H$

as shown in figure 5. 8(b).

No flag is affected with the instruction.

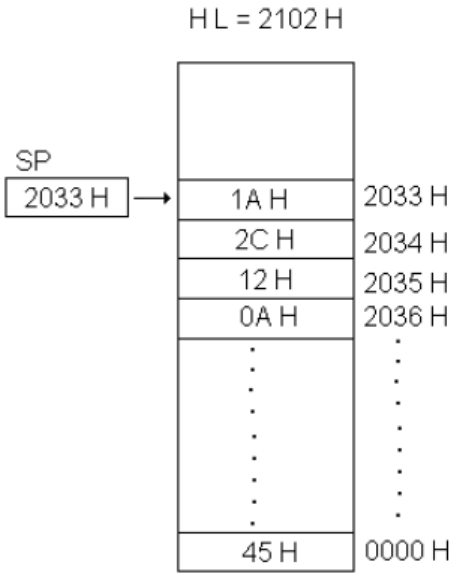

Fig. 5.8 (a)

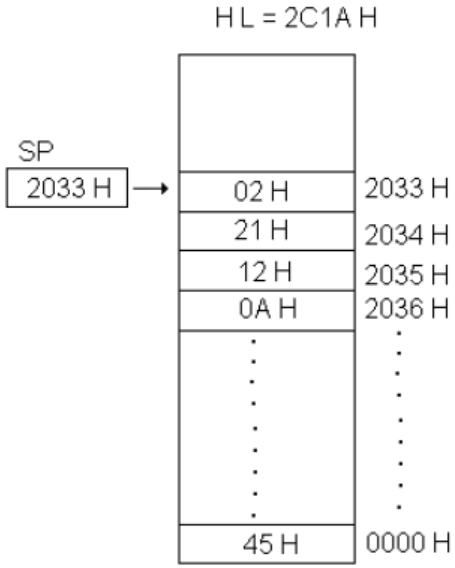

Fig. 5.8 (b)

Thank you