MICROPROCESSOR CLASS+LAB **DATE: 12/3/21**

Example 1: Writing a data to a Memory location

Suppose, the data is: 25_H and Memory location is : 5000_H.

We need to follow the following steps to write the data:

a. Loading the data into **accumulator register** (Ac or A).
b. Writing the data into the memory location.

a. Loading the data into accumulator register (Ac or A).

Here we use **MVI Register DATA – This instruction loads the DATA value into Register. MVI A 25_H**

b. Writing the data into the memory location

Here we use **STA 16-bit Memory location – This instruction writes the content of accumulator into the specified memory location. STA 5000_H**

So the Program Table Looks like:

Memory Address	Label	Mnemonic	Opcode	Operand	Hex Code	Comments
2000		**MVI A 25_H**	MVI	A 25_H	3E	**A<- 25**
2001					25	
2002		**STA 5000_H**	STA	5000_H	32	5000<- CONTENT OF A
2003					00	
2004					50	
2005		**HLT**	HLT		76	EOP

Example 2: Reading a data from a Memory location

Suppose, the Memory location is : 4000_H and it is loaded with some unknown data.

We need to follow the following step to read the data:

a. Read the data using instruction **LDA 16 bit Memory location. This instruction reads the content of the specified memory location and loads it to accumulator. LDA 4000_H. [use "User Data Grid" Section to give any arbitrary data into 4000_H memory location.]**

So the Program Table Looks like:

Memory Address	Label	Mnemonic	Opcode	Operand	Hex Code	Comments
2000		**LDA 4000_H**	LDA	4000_H	3A	**A<- CONTENT OF 4000**
2001					00	
2002					40	
2003		**HLT**			76	EOP

Example 3: Moving a data from source to destination.

Here source and destination can be different types like:

Source can be any Register – A, B,C D,E H,L

Destination can be any Register - A, B,C D,E H,L

The instruction that can be used here: **MOV Destination Reg., Source Reg.**

Here MOV is opcode.

Destination Reg., Source Reg. are the two operands.

So following instructions can be formed:

MOV B A

MOV C A

MOV D A
MOV E A
MOV H A
MOV L A
MOV M A
Here Source is fixed (Accumulator) but destination is variable. You can also take any other forms.

Suppose the source is accumulator which is to be loaded with data 88_H. Next, the content of this register needs to move into another register B.

We need to follow the following steps to move the data:

a. Loading the data into accumulator register (Ac or A).
b. Moving the data into register B.

a. Loading the data into accumulator register (Ac or A).
   Here we use **MVI A DATA – This instruction loads the DATA value into Accumulator (A). MVI A 88_H**
b. Moving the data into register B: Here we use **MOV B A**

So the Program Table Looks like:

Memory Address	Label	Mnemonic	Opcode	Operand	Hex Code	Comments
2000		**MVI A 88_H**	MVI	A 88_H	3E	A<-88
2001					88	
2002		**MOV B A**	MOV	B A	47	B<-A
2003		**HLT**			76	EOP

Example 4: DATA CAN BE MOVED FROM MEMORY TO SINGLE REGISTER USING REGISTER PAIR (M=H-L).

Suppose, the Memory location is: 4000_H and it is loaded with some unknown data. Now using M= H-L register pair we move the content of memory 4000 to register A

We need to follow the following step :

The instruction that can be used here is : **MOV Destination Reg., Source Reg. Pair**
Here MOV is opcode.
Destination Reg., Source Reg. pair are the two operands.
Here source register pair is always fixed. That is H-L pair which has another name: M
So following instructions can be formed:
MOV A M
MOV B M
MOV C M
MOV D M
MOV E M
MOV H M ----------MOV L M
Here Source is fixed (M) but destination is variable. You can also take any other forms.

We need to follow the following steps to move the data:

a. Loading the Address value into register pair (M).
b. Moving the data into register A.

a. Loading Address value into register pair (M).
   Here we use, **LXI H 16 bit Address – This instruction loads the 16 bit Address value into H-L pair. LXI H 4000_H**
b. Moving the data into register A: Here we use **MOV A M – Moving the content of memory location of 4000 to accumulator.**

So the Program Table Looks like:

Memory Address	Label	Mnemonic	Opcode	Operand	Hex Code	Comments
2000		**LXI H 4000_H**	LXI	H 4000_H	21	H-L<-4000
2001					00	
2002					40	
2003		**MOV A M**	MOV	A M	7E	A<-M OR A<- CONTENT OF MEMORY M/4000
2004		**HLT**			76	EOP

Example 5: DATA CAN BE MOVED FROM SINGLE REGISTER TO MEMORY USING REGISTER PAIR (M=H-L).

Suppose, the data is: 25_H is moved to register A. Then moved to Memory location is: 5000_H.

We need to follow the following steps to write the data:

a. Loading the Address value into register pair (M).
b. Loading the data into **accumulator register** (Ac or A).
c. Moving the data into the memory location.

a. Loading the Address value into register pair (M)
   Here we use, **LXI H 16 bit Address – This instruction loads the 16 bit Address value into H-L pair. LXI H 5000_H**
b. Loading the data into accumulator register (Ac or A).
   Here we use **MVI Register DATA – This instruction loads the DATA value into Register. MVI A 25_H**
c. Here we use **MOV M A - This instruction writes the content of accumulator into the specified memory location.**

So the Program Table Looks like:

Memory Address	Label	Mnemonic	Opcode	Operand	Hex Code	Comments
2000		**LXI H 5000_H**	LXI	5000_H	21	H-L<-5000
2001					00	
2002					50	
2003		**MVI A 25_H**	MVI	A 25_H	3E	A<-25
2004					25	
2005		**MOV M A**	MOV	M A	77	M<-A OR M<- CONTENT OF ACCUMULATOR
2006		**HLT**			76	EOP

Memory Address	Label	Mnemonic	Opcode	Operand	Hex Code	Comments
2000		**LXI H 5000_H**	LXI	5000_H	21	H-L<-5000
2001					00	
2002					50	
2003		**MVI M 25_H**	MVI	A 25_H	3E	M'S CONTENT<- 25
2004					25	
2005		**HLT**			76	EOP

PROBLEM: Suppose register B and C have the data 81_H and 82 $_H$ respectively. Now transfer content of B and C register into memory locations: 2000_H and 2001_H respectively.

A.

Memory Address	Label	Mnemonic	Opcode	Operand	Hex Code	Comments	MACHINE CYCLE	T STATE
3000		**MOV A B**	MOV	A B	78	A<-B	OF = MR+DECODING	4
3001		**STA 2000**	STA	20 00_H	32	2000<- A	OF = MR+DECODING	4
3002					00		MR	3
3003					20		MR	3
							MW	3
3004		MOV A C	MOV	A C	79	A<-C	OF= MR+DECODING	4
3005		**STA 2001**	STA	2001	32	2001<-A	OF = MR+DECODING	4
3006					01		MR	3
3007					20		MR	3
							MW	3
3008		**HLT**			76	EOP	OF = MR+DECODING	4

TOTAL T STATE: 38T , M CYCLES: 11 , BYTES: 9

EXECUTION TIME: 38 x 0.392 micro second

SUMMARIZED TABLE OF DATA MOVING OPERATIONS:

1. DATA CAN BE MOVED FROM ANY SINGLE REGISTER (A, **B,C,D,E,H,L**) TO MEMORY – WRITING TO MEMORY LOCATION.
2. DATA CAN BE MOVED FROM MEMORY TO ANY SINGLE REGISTER (A, **B,C,D,E,H,L**) – READING FROM MEMORY LOCATION
3. DATA CAN BE MOVED FROM ANY SINGLE REGISTER TO OTHER SINGLE REGISTER.
4. DATA CAN BE MOVED FROM MEMORY TO SINGLE REGISTER (**A, B,C,D,E**) USING REGISTER PAIR **(M=H-L)**
5. DATA CAN BE MOVED FROM SINGLE REGISTER (A, B,C,D,E) TO MEMORY USING REGISTER PAIR (M=H-L)

H==40

L==00

H-L==M=4000 ---16 BIT ADDRESS

SCENARIO1: READ 4000, 4001 MEMORY LOCATIONS AND PUT DATA INTO REGISTER B AND C RESPECTIVELY.

1ST APPROACH:

LDA 4000 (3)

MOV B A (1)

LDA 4001(3)

MOV C A(1)

TOTAL MEMORY LOCATIONS REQUIRED: 8

2ND APPROACH:

LXI H 4000 (3)

MOV B M (1)

INX H (1) H-L= H-L+01

MOV C M (1)

TOTAL MEMORY LOCATIONS REQUIRED: 6

SCENARIO2: READ 4001, 4000 MEMORY LOCATIONS AND PUT DATA INTO REGISTER H AND L RESPECTIVELY.

1ST APPROACH:

LDA 4000 (3)

MOV L A (1)

LDA 4001(3)

MOV H A(1)

TOTAL MEMORY LOCATIONS REQUIRED: 8

2ND APPROACH:

LHLD 4000 (3)

TOTAL MEMORY LOCATIONS REQUIRED: 3

LXI H 4000 HL<- 4000 MVI H 40 H<- 40 MVI L 00 L<-00