

* ⑧ Add two 8 bit using direct addressing method.
 ⑨ Subtract two 8 bit "

<u>Address</u>	<u>Mnemonic</u>	<u>opcode</u>	<u>M.C / Hex code</u>	<u>comment</u>
----------------	-----------------	---------------	-----------------------	----------------

C000, C001, C002	LDA	C100H		
------------------	-----	-------	--	--

C003

MOV

C, A

C004, C005, C006	LDA	C101H		
------------------	-----	-------	--	--

C007

MOV

B, A

C008, C009, C00A	LDA	C102H		
------------------	-----	-------	--	--

C00B

ADD
SUB

B

C103H

C00C, C00D, C00E	STA	C013		
------------------	-----	------	--	--

C00F, C010, C011	JNC			
------------------	-----	--	--	--

C012

INR

C013

MOV

STA

C
A, C

C104

Input →

✓ C100 → 00H

✓ C101 → 05H 03H

✓ C102 → 02H 05H

Output

✓ C103 → 07H (sum)

✓ C104 → 00H (any)

without

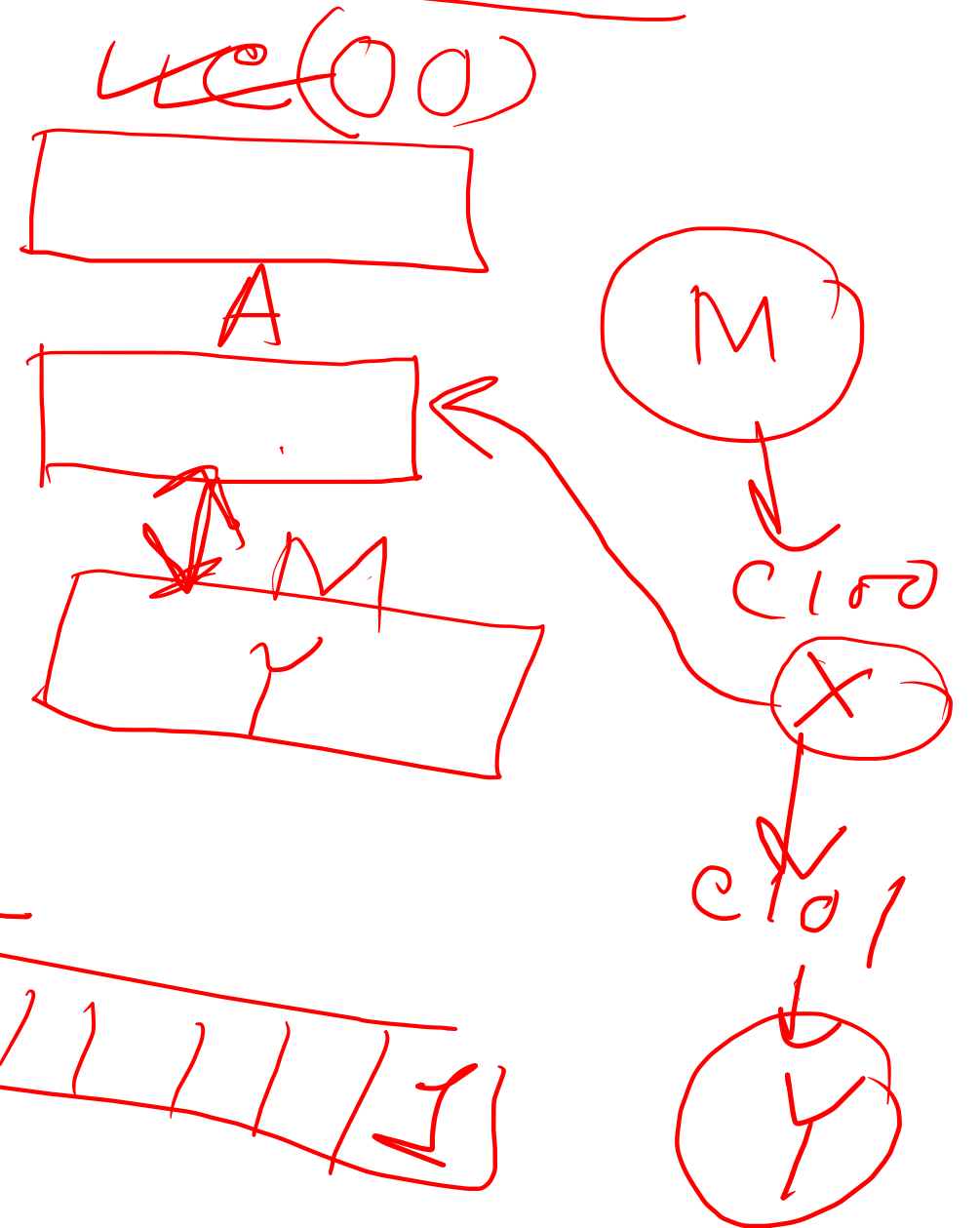
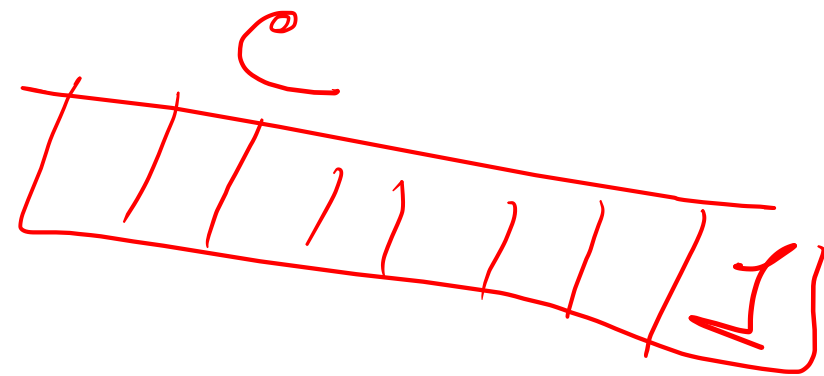
HLT

10 Add two 8 bit numbers using indirect addressing

11 mode
Sub

<u>Address</u>	<u>Mnemonic</u>	<u>Op code</u>	<u>M.C</u>	<u>Comments</u>
C000, C001, C002	LXI	H, C100 H		
C003	MVI	C, 00 H		
C004, C005	MOV	A, M		
C006	INX			
C007	ADD <u>SUB</u>	M		
C008, C009, C00A, C00B	JNC			
C00C, C00D, C00E, C00F	STA			
C010, C011, C012	MOV			
C013	STA			
	HLT			

@00C
@103
A, C
@104 H



data input $\rightarrow$ C100 $\rightarrow$ 97
C101 $\rightarrow$ 9A

Output C103 $\rightarrow$ 31H (same)

C104 $\rightarrow$ 01 (carry)
H

10010111
1001010

0000	0001
0	1 H

① 0011 0001
3 1