

**IMPORTANT ---- OPCODE NOT REQUIRED**

**UNIT - 4**

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR 8 BIT ADDITION**

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MOV A,M
	INX H
	MOV B,M
	ADD B
	INX H
	MOV M,A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	05
4101	07

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4102	0C

## ASSEMBLY LANGUAGE PROGRAM (ALP) FOR 8 – BIT SUBTRACTION

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MOV A, M
	INX H
	MOV B, M
	SUB B
	INX H
	MOV M, A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	09
4101	04

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4102	05

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR 8-BIT  
MULTIPLICATION**

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MOV B,M
	INX H
	MOV C,M
	SUB A
NEXT	ADD B
	DCR C
	JNZ NEXT
	INX H
	MOV M,A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	04
4101	03

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4102	0C

## ASSEMBLY LANGUAGE PROGRAM (ALP) FOR 8-BIT DIVISION

LABEL	MNEMONIC
	LXI H,4100
	MOV A,M
	INX H
	MOV B,M
	MVI C,00
NEXT	CMP B
	JC DONE x x
	SUB B
	INR C
	JMP NEXT
DONE	INX H
	MOV M,C
	INX H
	MOV M,A
	HLT

INPUT	
ADDRESS	DATA
4100	08
4101	03

OUTPUT	
ADDRESS	DATA
4102	02
4103	02

## ASSEMBLY LANGUAGE PROGRAM (ALP) FOR MULTIBYTE ADDITION

<b>LABEL</b>	<b>MNEMONIC</b>
	STC
	CMC
	LXI D,4100
	LXI H, 4300
	MVI C,05
NEXT	LDAX D
	ADC M
	MOV M,A
	INX H
	INX D
	DCR C
	JNZ NEXT
	JNC LAST
	MOV M,01
LAST	HLT

<b>INPUT</b>			
<b>ADDRESS</b>	<b>DATA</b>	<b>ADDRESS</b>	<b>DATA</b>
<b>4100</b>	<b>01</b>	<b>4300</b>	<b>05</b>
<b>4101</b>	<b>02</b>	<b>4301</b>	<b>06</b>
<b>4102</b>	<b>03</b>	<b>4302</b>	<b>01</b>
<b>4103</b>	<b>05</b>	<b>4303</b>	<b>03</b>
<b>4104</b>	<b>03</b>	<b>4304</b>	<b>01</b>

<b>OUTUPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4300	06
4301	08
4302	05
4303	08
4304	04

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR MULTIBYTE SUBTRACTION**

<b>LABEL</b>	<b>MNEMONIC</b>
	STC
	CMC
	LXI D,4100
	LXI H, 4300
	MVI C,05
NEXT	LDAX D
	SBB M
	MOV M,A
	INX H
	INX D
	DCR C
	JNZ NEXT
	HLT

<b>INPUT</b>			
<b>ADDRESS</b>	<b>DATA</b>	<b>ADDRESS</b>	<b>DATA</b>
<b>4100</b>	<b>02</b>	<b>4300</b>	<b>05</b>
<b>4101</b>	<b>05</b>	<b>4301</b>	<b>07</b>
<b>4102</b>	<b>08</b>	<b>4302</b>	<b>09</b>
<b>4103</b>	<b>01</b>	<b>4303</b>	<b>0A</b>
<b>4104</b>	<b>03</b>	<b>4304</b>	<b>0C</b>

<b>OUTUPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
<b>4300</b>	<b>03</b>
<b>4301</b>	<b>02</b>
<b>4302</b>	<b>01</b>
<b>4303</b>	<b>09</b>
<b>4304</b>	<b>09</b>

### ASSEMBLY LANGUAGE PROGRAM (ALP) FOR SUM OF SERIES

LABEL	MNEMONIC
NEXT	LXI H,4100
	MVI C,05
	SUB A
	ADD M
	INX H
	DCR C
	JNZ NEXT
	MOV M,A
	HLT

INPUT	
ADDRESS	DATA
4100	02
4101	07
4102	01
4103	05
4104	03

OUTPUT	
ADDRESS	DATA
4105	12



**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR BLOCK DATA TRANSFER**

<b>LABEL</b>	<b>MNEMONIC</b>
NEXT	LXI H,4100
	LXI D,4300
	MVI C,05
	MOV A,M
	STAX D
	INX H
	INX D
	DCR C
	JNZ NEXT
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	05
4101	09
4102	0A
4103	02
4104	1B

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4300	05
4301	09
4302	0A
4303	02
4304	1B

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR BIGGEST NUMBER IN AN ARRAY**

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MVI C,05
	DCR C
	MOV A,M
LOOP	INX H
	CMP M
	JNC NEXT
	MOV A,M
NEXT	DCR C
	JNZ LOOP
	INX H
	MOV M,A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	01
4101	02
4102	03
4103	04
4104	05

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4105	05

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR SMALLEST NUMBER IN AN ARRAY**

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MVI C,05
	DCR C
	MOV A,M
LOOP	INX H
	CMP M
	JC NEXT
	MOV A,M
NEXT	DCR C
	JNZ LOOP
	INX H
	MOV M,A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	09
4101	00
4102	03
4103	0A
4104	2B

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4105	00

## ASSEMBLY LANGUAGE PROGRAM (ALP) FOR ASCENDING ORDER

LABEL	MNEMONIC
	MVI D,04
START	LXI H,4100
	MVI C,04
LOOP	MOV A,M
	INX H
	MOV B,M
	CMP B
	JC NEXT
	MOV M,A
	DCX H
	MOV M,B
	INX H
NEXT	DCR C
	JNZ LOOP
	DCR D
	JNZ START
	HLT

INPUT	
ADDRESS	DATA
4100	05
4101	07
4102	02
4103	08
4104	03

OUTPUT	
ADDRESS	DATA
4300	02
4301	03
4302	05
4303	07
4304	08

### ASSEMBLY LANGUAGE PROGRAM (ALP) FOR DESCENDING ORDER

<b>LABEL</b>	<b>MNEMONIC</b>
	MVI D,04
START	LXI H,4100
	MVI C,04
LOOP	MOV A,M
	INX H
	MOV B,M
	CMP B
	JNC NEXT
	MOV M,A
	DCX H
	MOV M,B
	INX H
NEXT	DCR C
	JNZ LOOP
	DCR D
	JNZ START
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4300	07
4301	02
4302	00
4303	05
4304	09

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4300	09
4301	07
4302	05
4303	02
4304	00

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR DISASSEMBLY OF A BYTE**

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MOV A,M
	MOV B,A
	ANI 0F
	INX H
	MOV M,A
	MOV A,B
	RRC
	RRC
	RRC
	RRC
	ANI 0F
	INX H
	MOV M,A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	38

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4101	08
4102	03

### ASSEMBLY LANGUAGE PROGRAM (ALP) FOR ASSEMBLY OF A BYTE

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MOV B,M
	INX H
	MOV A,M
	RRC
	RRC
	RRC
	RRC
	ADD B
	INX H
	MOV M,A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	08
4101	03

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4102	38



## ASSEMBLY LANGUAGE PROGRAM (ALP) FOR DECIMAL ADDITION

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MOV A,M
	INX H
	MOV B,M
	ADD B
	DAA
	INX H
	MOV M,A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	05
4101	07

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4102	12

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR DECIMAL  
SUBTRACTION**

<b>LABEL</b>	<b>MNEMONIC</b>
	LXI H,4100
	MOV A,M
	INX H
	MOV B,M
	SUB B
	DAA
	INX H
	MOV M,A
	HLT

<b>INPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4100	09
4101	04

<b>OUTPUT</b>	
<b>ADDRESS</b>	<b>DATA</b>
4102	05

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR ONE'S COMPLEMENT  
OF 8 BIT NUMBER**

<b>LABEL</b>	<b>MNEMONIC</b>
	LDA 2501 H
	CMA
	STA 2502 H
	HLT

**Ones complement is nothing but changing 1 to zero and zero to 1. It is nothing but complementation.**

**Example :**

**5485 = 0101 0100 1000 0101**  
**(5) (4) (8) (5)**

**One's complement is**

**1010 1011 0111 1010**  
**(A) (B) (7) (A)**

**Hence the one's complement of 5485 is AB7A**

**ASSEMBLY LANGUAGE PROGRAM (ALP) FOR TWO'S COMPLEMENT  
OF 8 BIT NUMBER**

<b>LABEL</b>	<b>MNEMONIC</b>
	LDA 2501 H
	CMA
	INR A
	STA 2502 H
	HLT

**Two's complement is nothing but adding one to the one's complement of the number.**

**Example : Find one's complement of 96**

**96 = 1001 0110**  
       (9)    (6)

**One's complement**

**0110 1001**  
       (6)    (9)

**Two's complement**

**0110 1001 (69)**  
                   1  
 -----  
**0110 1010 (6A)**

**The two's complement of 69 is 6A**