



**INDIAN INSTITUTE OF INFORMATION TECHNOLOGY
SONEPAT**

Microprocessor and Interfacing Lab
(CSC508)

Practical Lab File

Submitted To:

Dr. Rajiv Verma

Submitted By

Dipankar Yadav

Roll No.- 12111070

Branch: CSE

Semester: 5

Session: 2021-25

Index


Practical No.	Topic	Page Number
1.	Addition of two 8 bit numbers	3
2.	Subtraction of two 8 bit numbers	4
3.	Addition with a carry of two 8 bit numbers	5
4.	Subtraction with a borrow of two 8 bit numbers	6
5.	Multiplication of two 8 bit numbers using repeated addition	7
6.	Multiplication of two 8 bit numbers using bit rotation	8
7.	Division of two 8 bit numbers using repeated addition	9
8.	Division of two 8 bit numbers using bit rotation	10

Practical 1: - Write a program for addition of two 8 bit numbers

CODE:

```
MVI A,01H
MVI B,05H
ADD B
STA 0000H
HLT
```

OUTPUT:

Memory view  0x Address 1

	0	1	2	3	4	5	6	7	8	9	A	B	C	D
000	06	00	00	00	00	00	00	00	00	00	00	00	00	00
001	00	00	00	00	00	00	00	00	00	00	00	00	00	00
002	00	00	00	00	00	00	00	00	00	00	00	00	00	00
003	00	00	00	00	00	00	00	00	00	00	00	00	00	00
004	00	00	00	00	00	00	00	00	00	00	00	00	00	00
005	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Practical 2: - Write a program for subtraction of two 8 bit numbers

CODE:

```
MVI A,11H
MVI B,05H
SUB B
STA 0000H
HLT
```

OUTPUT:

Load at 0x0800

main.asm

```
1 MVI A,11H
2 MVI B,05H
3 SUB B
4 STA 0000H
5 HLT
6
7
```

Memory View

	0	1	2	3	4	5	6	7	8	9	A	B
000	0C	00	00	00	00	00	00	00	00	00	00	00
001	00	00	00	00	00	00	00	00	00	00	00	00
002	00	00	00	00	00	00	00	00	00	00	00	00
003	00	00	00	00	00	00	00	00	00	00	00	00
004	00	00	00	00	00	00	00	00	00	00	00	00
005	00	00	00	00	00	00	00	00	00	00	00	00
006	00	00	00	00	00	00	00	00	00	00	00	00
007	00	00	00	00	00	00	00	00	00	00	00	00
008	00	00	00	00	00	00	00	00	00	00	00	00
009	00	00	00	00	00	00	00	00	00	00	00	00
00A	00	00	00	00	00	00	00	00	00	00	00	00

Practical 3: - Write a program for addition with a carry of two 8 bit numbers

CODE:

```
MVI A, 00H
MVI B, 23H
MVI C, 98H
MVI D, 45H
MVI E, 22H
MOV A, C
ADD E
MOV C, A
STA 0061H
MOV A, B
ADC D
STA 0060H
hlt
```

OUTPUT:

Load at 0x0800

main.asm

```
1 ;<Program title>
2 jmp start
3 ;data
4 ; code
5 start: nop
6 MVI A, 00H
7 MVI B, 23H
8 MVI C, 98H
9 MVI D, 45H
10 MVI E, 22H
11 MOV A, C
12 ADD E
13 MOV C, A
14 STA 0061H
15 MOV A, B
16 ADC D
17 STA 0060H
18 hlt
```

Memory View

	0	1	2	3	4	5	6	7	8	9
000	00	00	00	00	00	00	00	00	00	00
001	00	00	00	00	00	00	00	00	00	00
002	00	00	00	00	00	00	00	00	00	00
003	00	00	00	00	00	00	00	00	00	00
004	00	00	00	00	00	00	00	00	00	00
005	00	00	00	00	00	00	00	00	00	00
006	68	BA	00	00	00	00	00	00	00	00
007	00	00	00	00	00	00	00	00	00	00
008	00	00	00	00	00	00	00	00	00	00
009	00	00	00	00	00	00	00	00	00	00
00A	00	00	00	00	00	00	00	00	00	00
00B	00	00	00	00	00	00	00	00	00	00
00C	00	00	00	00	00	00	00	00	00	00
00D	00	00	00	00	00	00	00	00	00	00
00E	00	00	00	00	00	00	00	00	00	00
00F	00	00	00	00	00	00	00	00	00	00

Practical 4: - Write a program for subtraction of two 8 bit numbers with borrow.

CODE:

```
MVI A, 8DH
MVI B, 7DH
MOV C, A
SUB B
JC SUBTRACT
SUBTRACT: STA RESULT
RESULT: DB 00H
HLT;
```

OUTPUT:

Memory View



0x

Address i

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00	00	00	00	00	00	00	00	00	00	00	00	10	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Practical 5: - Write a program for multiplication of two 8 bit numbers using repeated addition.

CODE:

```
MVI b,02h
MVI c,04h
MVI a, 00h
MVI d,00h
loop: add b
jnc skip
inr d
skip: dcr c
jnz loop
mov b,d
mov c,a
mov a,c
sta 0000h
hlt
```

OUTPUT:

Memory view

0X

Address 1

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
000	08	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
001	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
002	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
003	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
004	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
005	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
006	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
007	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
008	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
009	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
00A	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00

Practical 6: - Write a program for multiplication of two 8 bit numbers using bit rotation method

CODE:


```
MVI D,06H
MVI A,05H
LXI H,0000H

LOOP:  RRC
        JNC SKIP
        DAD D

SKIP:   XCHG
        DAD H
        XCHG
        DCR C
        JNZ LOOP
        HLT
```

OUTPUT:

Register	Value
A/PSW	0x 05 56
BC	0x 00 00
DE	0x 00 00
HL	0x 1E 00
SP	0x FF FF
PC	0x 08 14

Flags 

Line	Code
1	MVI D,06H
2	MVI A,05H
3	LXI H,0000H
4	
5	LOOP: RRC
6	JNC SKIP
7	DAD D
8	
9	SKIP: XCHG
10	DAD H
11	XCHG
12	DCR C
13	JNZ LOOP
14	HLT

Practical 7: - Write a program for division of two 8 bit numbers by repeated addition method.

CODE:

```
MVI A, 27H
MVI B, 05H
MVI C, 00H
MVI D, 00H
LOOP: SUB B
      JC DONE
INR C
JMP LOOP
DONE: MOV E, A
HLT
```

OUTPUT:

A/PSW	0x FF 97
BC	0x 05 07
DE	0x 00 FF
HL	0x 00 00
SP	0x FF FF
PC	0x 08 12

Flags 

```
main.asm
1 MVI A, 27H
2 MVI B, 05H
3 MVI C, 00H
4 MVI D, 00H
5 LOOP: SUB B
6       JC DONE
7 INR C
8 JMP LOOP
9 DONE: MOV E, A
10 HLT
```

Practical 8: - Write a program for division of two 8 bit numbers using by bit rotation method.

CODE:

```

MVI E,00H;
LHLD 0008H;
LDA 0007H;
MOV B,A;
MVI C,08H;
NEXT:DAD H;
MOV A,E;
RLC
MOV E,A;
MOV A,H;
SUB B;
JC SKIP;
MOV H,A;
INR E;
SKIP:DCR C;
JNZ NEXT;
MOV A,E;
STA 0033H;
MOV A,H;
STA 0034H;
HLT;

```

OUTPUT:

	0	1	2	3	4	5	6	7	8	9
000	00	00	00	00	00	00	00	00	00	00
001	00	00	00	00	00	00	00	00	00	00
002	00	00	00	00	00	00	00	00	00	00
003	00	00	00	FF	00	00	00	00	00	00
004	00	00	00	00	00	00	00	00	00	00
005	00	00	00	00	00	00	00	00	00	00
006	00	00	00	00	00	00	00	00	00	00
007	00	00	00	00	00	00	00	00	00	00
008	00	00	00	00	00	00	00	00	00	00
009	00	00	00	00	00	00	00	00	00	00