

ASIAN SCHOOL OF MANAGEMENT AND TECHNOLOGY

[Affiliated to Tribhuvan University]



ASSIGNMENT: KIT LAB REPORT OF MICROPROCESSOR

Submitted By

Name: Manalika Shrestha

Roll No: 13

Program: B.Sc. CSIT

Submission Date: / /2080

Registration No:

Submitted To:

Er. Manoj Giri

Kathmandu, Nepal

Table of contents

Page

1. WAP to add 56h and 37h and store the sum in memory address 8050h.	3
2. WAP to add two number stored in 8050h and store sum in 8052h.	4
3. WAP to subtract 37h from 52h and store the difference in memory address 8050.	5
4. WAP to find the difference of number stored in location 9050 and 9051 and stored the difference in 9052.	6
5. WAP to find greater number between two numbers stored in location 8050 and 8051 and stored the bigger number in memory address 9050.	7
6. WAP a program to find a given number is positive or negative and if negative stored in location 8051 and positive in 8050.	8
7. WAP to find a given number is even or odd and if even store in location 9050h and if odd store in 9051h.	10
8. WAP to add two 16 bits numbers.	12
9. WAP to add two numbers stored in location 8000 and 8001 and store the sum in 8003 and carry in 8002.	13
10. WAP to multiply two number stored in location 8002 and 8003 and stored the product in 8084.	14
11. WAP to find quotient and remainder and store quotient at location 8050 and remainder at location 8051.	15
12. Find compliment of a number store in the location 9051and store its compliment in location 9052.	17
13. Find compliment 2's of a number stored in location 8050 and store its compliment at location.	18
14. Find the greatest number out of 10 number stored from location 9010 and number stored the greatest number at location 9060.	19
15. Find the greatest number out of 10 number stored from location 9010 and store the smallest at location 9060.	21
16. WAP to copy 10 data from memory location 8050 to new memory location 9050.	23
17. WAP to generate 10 even numbers.	25
18. WAP to generate 10 odd number.	26
19.WAP to count even number or odd number out of 10 number and store even number in location 8050H and odd number in location 8060H.	27
20.WAP to sort given 10 numbers from memory location 8200H in ascending order.	29

1. Write a program to add 56H and 37H and store the sum in memory address 8050H.

MVI A, 56H

ADI 37H

STA 8050H

HLT

OUTPUT:

Data	Stack	KeyPad	Memory	I/O Ports
Start	8050H	OK		
Address (Hex)	Address	Data		
8050	32848	141		
8051	32849	0		
8052	32850	0		
8053	32851	0		
8054	32852	0		
8055	32853	0		
8056	32854	0		
8057	32855	0		
8058	32856	0		
8059	32857	0		
805A	32858	0		
805B	32859	0		
805C	32860	0		
805D	32861	0		
Line No	Assembler Message			
0	Program assembled successfully			

Registers			Flag
A	8D		S 1
BC	06	03	
DE	02	05	Z 0
HL	23	32	
PSW	00	00	AC 0
PC	42	08	P 1
SP	FF	FF	
Int-Reg	00		C 0

2. Write a program to add two number store in 8050H and 8051H and store the sum in 8052H.

```
LDA 8050H
MOV H, A
LDA 8051H
ADD H
MOV L, A
MVI A, 00
MOV H, A
SHLD 8052H
HLT
```

OUTPUT:

Registers			Flag	
<i>A</i>	00		<i>S</i>	0
<i>BC</i>	06	03		
<i>DE</i>	02	05	<i>Z</i>	0
<i>HL</i>	00	6E	<i>AC</i>	0
<i>PSW</i>	00	00		
<i>PC</i>	42	10	<i>P</i>	0
<i>SP</i>	FF	FF		
<i>Int-Reg</i>	00		<i>C</i>	0

Data

Stack

KeyPad

Memory

I/O Ports

Start

8050h

OK

Address (Hex)	Address	Data
8050	32848	50
8051	32849	60
8052	32850	110
8053	32851	0
8054	32852	0
8055	32853	0
8056	32854	0
8057	32855	0
8058	32856	0
8059	32857	0
805A	32858	0
805B	32859	0
805C	32860	0
805D	32861	0

Line No	Assembler Message
0	Program assembled successfully

3. Write a program to subtract 37H from 52H and store the difference in memory address 8050.

MVI A, 52H

SUI 37H

STA 8050

HLT

OUTPUT:

Registers			Flag
A	1B		S 0
BC	06	03	Z 0
DE	02	05	
HL	00	6E	AC 0
PSW	00	00	P 1
PC	42	08	
SP	FF	FF	C 0
Int-Reg	00		

Registers			Flag	
<i>A</i>	01		<i>S</i>	0
<i>BC</i>	05	00	<i>Z</i>	0
<i>DE</i>	00	00	<i>AC</i>	0
<i>HL</i>	00	00	<i>P</i>	0
<i>PSW</i>	00	00	<i>C</i>	0
<i>PC</i>	42	0C		
<i>SP</i>	FF	FF		
<i>Int-Reg</i>	00			

5. Write a program to find greater number between two number stored in location 8050H and 8051H and stored the bigger number in memory address 9050H.

LDA 8050H

MOV B, A

LDA 8051

CMP B

JNC GREATER

MOV A, B

GREATER: STA 9050

HLD

OUTPUT:

Data	Stack	KeyPad	Memory	I/O Ports
Start 9050 OK				
Address (Hex)	Address	Data		
235A	9050	50		
235B	9051	0		
235C	9052	0		
235D	9053	0		
235E	9054	6		
235F	9055	9		
2360	9056	0		
2361	9057	0		
2362	9058	0		
2363	9059	0		
2364	9060	0		
2365	9061	0		
2366	9062	0		
2367	9063	0		

Line No	Assembler Message
0	Program assembled successfully

Registers	Flag
A 32	S 1
BC 32 00	Z 0
DE 02 05	AC 0
HL 09 06	P 0
PSW 00 00	C 1
PC 42 10	
SP FF FF	
Int-Reg 00	

Decimal - Hex Conversion

6. Write a program to find a given number is positive or negative and if positive store in location 8050H if negative store in 8051H.

LDA 9050H

MOV B, A

RAL

JC NEGATIVE

MVI A,00H

RAR

MOV A, B

STA 8050H

NEGATIVE: MOV A, B

STA 8051H

HLT

OUTPUT:

Registers			Flag
A	D6		S 0
BC	D6	00	Z 0
DE	00	00	AC 0
HL	00	00	P 0
PSW	00	00	C 1
PC	42	15	
SP	FF	FF	
Int-Reg	00		

Decimal - Hex Conversion	
Decimal	Hex
214	d6h
To Hex	To Dec

Data
Stack
KeyPad
Memory
I/O Ports

Start

Address (Hex)	Address	Data
9050	36944	214
9051	36945	0
9052	36946	0
9053	36947	0
9054	36948	0
9055	36949	0
9056	36950	0
9057	36951	0
9058	36952	0
9059	36953	0
905A	36954	0
905B	36955	0
905C	36956	0
905D	36957	0

Line No	Assembler Message
0	Program assembled successfully

Data
Stack
KeyPad
Memory
I/O Ports

Start

Address (Hex)	Address	Data
8050	32848	0
8051	32849	214
8052	32850	0
8053	32851	0
8054	32852	0
8055	32853	0
8056	32854	0
8057	32855	0
8058	32856	0
8059	32857	0
805A	32858	0
805B	32859	0
805C	32860	0
805D	32861	0

Line No	Assembler Message
0	Program assembled successfully

7. Write a program to find a given number is even or odd and if even stored in location 9050H and if odd stored in 9051H.

LDA 8050H

ANI 01H

JZ EVEN

MVI A, 01H

STA 9050H

EVEN: MVI A, 00H

STA 9050H

HLT

OUTPUT:

Data Stack KeyPad Memory I/O Ports			
Start		9050h	
		OK	
Address (Hex)	Address	Data	
9050	36944	0	
9051	36945	1	
9052	36946	0	
9053	36947	0	
9054	36948	0	
9055	36949	0	
9056	36950	0	
9057	36951	0	
9058	36952	0	
9059	36953	0	
905A	36954	0	
905B	36955	0	
905C	36956	0	
905D	36957	0	

Line No	Assembler Message
0	Program assembled successfully

Registers			Flag
A	00		S 0
BC	D6	00	Z 0
DE	00	00	
HL	00	00	AC 1
PSW	00	00	
PC	42	13	P 0
SP	FF	FF	C 0
Int-Reg	00		

Data
Stack
KeyPad
Memory
I/O Ports

Start
OK

Address (Hex)	Address	Data
8050	32848	5
8051	32849	0
8052	32850	0
8053	32851	0
8054	32852	0
8055	32853	0
8056	32854	0
8057	32855	0
8058	32856	0
8059	32857	0
805A	32858	0
805B	32859	0
805C	32860	0
805D	32861	0

Line No	Assembler Message
0	Program assembled successfully

8. Write a program to add two 16 bits numbers stored in location 9050,9051 and 9052,9053 and then stored the sum in 9054 and 9055.

LHLD 9050

XCHG

LHLD 9052

DAD D

MVI C,00H

JNC LOOP

INR C

LOOP: SHLD 9054

MOV A, C

STA 9056

HLT

OUTPUT:

The screenshot displays an 8085 assembly simulator interface. The 'Memory' tab is active, showing a memory dump starting at address 9050. The data at 9050 is 5, at 9051 is 6, at 9052 is 8, at 9053 is 15, at 9054 is 13, and at 9055 is 21. The registers window shows the following values: A=00, BC=D6 00, DE=06 05, HL=15 0D, PSW=00 00, PC=42 16, SP=FF FF, and Int-Reg=00. The flag register shows S=0, Z=0, AC=1, P=0, and C=0. The assembly message window at the bottom indicates 'Program assembled successfully'.

Address (Hex)	Address	Data
235A	9050	5
235B	9051	6
235C	9052	8
235D	9053	15
235E	9054	13
235F	9055	21
2360	9056	0
2361	9057	0
2362	9058	0
2363	9059	0
2364	9060	0
2365	9061	0
2366	9062	0
2367	9063	0

Registers	Flag
A 00	S 0
BC D6 00	
DE 06 05	Z 0
HL 15 0D	
PSW 00 00	AC 1
PC 42 16	
SP FF FF	P 0
Int-Reg 00	
	C 0

Line No	Assembler Message
0	Program assembled successfully

9. Write a program to add two numbers stored in locations 8000 and 8001, store the sum of addition in 8003 and carry in 8002.

MVI C,00

LDA 8000H

MOV B, A

LDA 8001H

ADD B

STA 8003H

INR C

MOV A, C

STA 8002H

HLT

OUTPUT:

MOV C, M

BACK: DCR C

JZ SKIP

ADD D

JMP BACK

SKIP: STA 8004

HLT

OUTPUT:

Registers			Flag		Memory	I/O Ports
A	04		S	0		OK
BC	05	00				
DE	02	05	Z	1		
HL	1F	43	AC	0		
PSW	00	00				
PC	42	14	P	1		
SP	FF	FF				
Int-Reg	00		C	0		

11.WAP to find quotient and remainder and store quotient at location 8050 and remainder at location 8051.

LXI H,8000

MOV B, M

MVI C,00

INX H

MOV A, M

NEXT: CMP B

JC LOOP

SUB B

INR C

JMP NEXT

LOOP: STA 8081

MOV A, C

STA 8050

HLT

OUTPUT:

Registers			Flag	
<i>A</i>	04		<i>S</i>	1
<i>BC</i>	08	04	<i>Z</i>	0
<i>DE</i>	02	05	<i>AC</i>	0
<i>HL</i>	1F	41	<i>P</i>	0
<i>PSW</i>	00	00	<i>C</i>	1
<i>PC</i>	42	19		
<i>SP</i>	FF	FF		
<i>Int-Reg</i>	00			

Data
Stack
Keypad
Memory
I/O Ports

Start

Address (Hex)	Address	Data
1F72	8050	4
1F73	8051	3
1F74	8052	0
1F75	8053	0
1F76	8054	0
1F77	8055	0
1F78	8056	0
1F79	8057	0
1F7A	8058	0
1F7B	8059	0
1F7C	8060	0
1F7D	8061	0
1F7E	8062	0
1F7F	8063	0

Line No

Assembler Message

0

Program assembled successfully

Line No

Assembler Message

0

Program assembled successfully

12. Find compliment of a number store in the location 9051and store its compliment in location 9052.

LDA 9051

CMA

STA 9052

HLT

OUTPUT:

Registers			Flag	
<i>A</i>	9B		<i>S</i>	1
<i>BC</i>	08	04		
<i>DE</i>	02	05	<i>Z</i>	0
<i>HL</i>	1F	41	<i>AC</i>	0
<i>PSW</i>	00	00		
<i>PC</i>	42	08	<i>P</i>	0
<i>SP</i>	FF	FF		
<i>Int-Reg</i>	00		<i>C</i>	1

Data

Stack

ABC

Keypad

Memory

I/O Ports

Start

9051

OK

Address (Hex)	Address	Data
235B	9051	100
235C	9052	155
235D	9053	0
235E	9054	0
235F	9055	0
2360	9056	0
2361	9057	0
2362	9058	0
2363	9059	0
2364	9060	0
2365	9061	0
2366	9062	0
2367	9063	0
2368	9064	0

Line No	Assembler Message
0	Program assembled successfully

13. Find compliment 2's of a number stored in location 8050 and store its compliment at location.

LDA 8050

CMA

INR A

STA 8051

HLT

OUTPUT:

Registers			Flag		ad		Memory	I/O Ports	
A	9C		S	1				<input type="text"/>	OK
BC	08	04							
DE	02	05	Z	0					
HL	1F	41	AC	0					
PSW	00	00							
PC	42	09	p	1					
SP	FF	FF							
Int-Reg	00		C	1					
</									

14. Find the greatest number out of 10 number stored from location 9010 and number stored the greatest number at location 9060.

LXI H,9010

MOV C, M

INX H

DCR C

MOV A, M

Skip: INX H

CMP M

JNC LOOP

MOV A, M

LOOP: DCR C

JNC Skip

STA 9060

HLT

OUTPUT:

Data	Stack	Keypad	Memory	I/O Ports
Start	9060	OK		
Address (Hex)	Address	Data		
2364	9060	70		
2365	9061	0		
2366	9062	0		
2367	9063	0		
2368	9064	0		
2369	9065	0		

Registers

A	46
BC	08 02
DE	02 05
HL	23 34
PSW	00 00
PC	42 15
SP	FF FF
Int-Reg	00

Flag

S	0
Z	0
AC	0
P	0
C	1

ad

Memory

I/O Ports

OK

💡

Decimal - Hex Conversion

ssfully

15. Find the smallest number out of 10 numbers stored from location 9010 and store the smallest at location 9060.

LXI H, 9010

MOV C, M

INX H

MOV B, M

DCR C

LOOP: INX H

MOV A, M

CMP B

JNC Skip

MOV B, A

Skip: DCR C

JNZ LOOP

LXI H, 9060H

MOV M, B

HLT

OUTPUT:

Registers			Flag	
A	00		S	0
BC	00	00	Z	1
DE	00	00	AC	0
HL	90	60	P	1
PSW	00	00	C	0
PC	42	16		
SP	FF	FF		
Int-Reg	00			

Data
 Stack
 Keypad
 Memory
 I/O Ports

Start

Address (Hex)	Address	Data
9060	36960	2
9061	36961	0
9062	36962	0
9063	36963	0
9064	36964	0
9065	36965	0
9066	36966	0
9067	36967	0
9068	36968	0
9069	36969	0
906A	36970	0
906B	36971	0
906C	36972	0
906D	36973	0

Line No	Assembler Message
0	Program assembled successfully

16. WAP to copy 10 data from memory location 8050 to new memory location 9050.

LXI H,8050H

LXI D,9050H

MVI B,10H

LOOP:MOV A, M

STAX D

INX H

INX D

DCR B

JNZ LOOP

HLT

OUTPUT:

Registers			Flag
A	00		S 0
BC	00	00	
DE	90	60	Z 1
HL	80	60	
PSW	00	00	AC 0
PC	42	11	P 1
SP	FF	FF	
Int-Reg	00		C 0

Data
Stack
KeyPad
Memory
I/O Ports

Start

Address (Hex)	Address	Data
8050	32848	1
8051	32849	2
8052	32850	3
8053	32851	4
8054	32852	5
8055	32853	6
8056	32854	7
8057	32855	8
8058	32856	9
8059	32857	10
805A	32858	0
805B	32859	0
805C	32860	0
805D	32861	0

Line No	Assembler Message
0	Program assembled successfully

Data
Stack
KeyPad
Memory
I/O Ports

Start

Address (Hex)	Address	Data
9050	36944	1
9051	36945	2
9052	36946	3
9053	36947	4
9054	36948	5
9055	36949	6
9056	36950	7
9057	36951	8
9058	36952	9
9059	36953	10
905A	36954	0
905B	36955	0
905C	36956	0
905D	36957	0

Line No	Assembler Message
0	Program assembled successfully

17. WAP to generate 10 even numbers.

MVI A,00H

MVI B,02H

MVI C, 0AH

LXI H,9200H

LOOP: ADDB

MOV M, A

INX H

DCR C

JNZ LOOP

HLT

OUTPUT:

Registers			Flag	
<i>A</i>	14		<i>S</i>	0
<i>BC</i>	02	00		
<i>DE</i>	90	60	<i>Z</i>	1
<i>HL</i>	92	0A		
<i>PSW</i>	00	00	<i>AC</i>	0
<i>PC</i>	42	11	<i>P</i>	1
<i>SP</i>	FF	FF		
<i>Int-Reg</i>	00		<i>C</i>	0

Data

Stack

ABC

Keypad

Memory

I/O Ports

Start

9200h

OK

Address (Hex)	Address	Data
9200	37376	2
9201	37377	4
9202	37378	6
9203	37379	8
9204	37380	10
9205	37381	12
9206	37382	14
9207	37383	16
9208	37384	18
9209	37385	20
920A	37386	0
920B	37387	0
920C	37388	0
920D	37389	0

Line No	Assembler Message
0	Program assembled successfully

18. WAP to generate 10 odd number.

MVI A,01H

MVI B,02H

MVI C,0AH

LXI H, 9050H

LOOP: ADD B

MOV M, A

INX H

DCR C

HLT

OUTPUT:

The screenshot displays an 8085 assembly simulator interface. On the left, a 'Registers' table shows the current state of the processor's registers. To its right, a 'Flag' section shows the status of various flags. The main window is divided into three tabs: 'ad', 'Memory', and 'I/O Ports'. The 'Memory' tab is currently selected, showing an empty memory dump. Below the registers, a table shows the assembly process results.

Registers	Value
A	15
BC	02 00
DE	90 60
HL	90 5A
PSW	00 00
PC	42 11
SP	FF FF
Int-Reg	00

Flag	Value
S	0
Z	1
AC	0
P	1
C	0

Line No	Assembler Message
0	Program assembled successfully

19.WAP to count even number or odd number out of 10 number and store even number in location 8050H and odd number in location 8060H.

LXI H,8000H

MOV C, M

MVI B,00H

MVI D,00H

LOOP2: INX H

MOV A, M

RAR

JC LOOP1

INR B

JMP LOOP3

LOOP1: INR D

LOOP3: DCR C

JNZ LOOP2

MOV A, B

STA 8050h

MOV A, D

STA 8060h

HLT

OUTPUT:

Data	Stack	KeyPad	Memory	I/O Ports
Start	8050h		OK	
Address (Hex)	Address	Data		
8050	32848	5		
8051	32849	0		
8052	32850	0		
8053	32851	0		
8054	32852	0		

Data	Stack	KeyPad	Memory	I/O Ports
Start	8060h		OK	
Address (Hex)	Address	Data		
8060	32864	5		
8061	32865	0		
8062	32866	0		
8063	32867	0		
-----	-----	-		

Registers	Flag	
A 05	S 0	
BC 05 00	Z 1	
DE 05 00	AC 0	
HL 80 0A	P 1	
PSW 00 00	C 1	1
PC 42 20		1
SP FF FF		1
Int-Reg 00		1

Data	Stack	KeyPad	Memory	I/O Ports
Start	8000h		OK	
Address (Hex)	Address	Data		
8000	32768	10		
8001	32769	5		
8002	32770	6		
8003	32771	55		
8004	32772	4		
8005	32773	3		
8006	32774	2		
8007	32775	1		
8008	32776	12		
8009	32777	14		
800A	32778	17		
800B	32779	0		
800C	32780	0		
800D	32781	0		
-----	-----	-		
Line No	Assembler Message			
0	Program assembled successfully			

20.WAP to sort given 10 numbers from memory location 8200H in ascending order.

MVI B, 09

START: LXI H, 8200H

MVI C,0AH

BACK: MOV A, M

INX H

CMP M

JNC SKIP

MOV D, M

MOV M, A

DCX H

MOV M, D

INX H

SKIP: DCR C

JNZ BACK

DCR B

JNZ START

HLT

OUTPUT:

Registers			Flag
<i>A</i>	02		<i>S</i> 0
<i>BC</i>	00	00	<i>Z</i> 1
<i>DE</i>	0D	00	
<i>HL</i>	82	0A	<i>AC</i> 0
<i>PSW</i>	00	00	<i>P</i> 1
<i>PC</i>	42	1B	
<i>SP</i>	FF	FF	<i>C</i> 0
<i>Int-Reg</i>	00		

Decimal - Hex Conversion	
Decimal	Hex

Data	Stack	Keypad	Memory	I/O Ports
Start 8200h				OK
Address (Hex)	Address	Data		
8200	33280	12		
8201	33281	13		
8202	33282	9		
8203	33283	8		
8204	33284	7		
8205	33285	6		
8206	33286	5		
8207	33287	4		
8208	33288	3		
8209	33289	2		
820A	33290	1		
820B	33291	0		
820C	33292	0		
820D	33293	0		

Line No	Assembler Message
0	Program assembled successfully