

AIM:

To compute various logical operations using 8085 processor.

ALGORITHM:

- 1) Load data to accumulator.
- 2) Load another data in register
- 3) Perform logical operations like AND, OR and XOR (Use ANA, ORA, XRA) with the accumulator content.
- 4) Store the result in specified memory location.

PROGRAM:**AND
OPERATION:**

```
MVI A,06
```

MVI B,04

ANA B

STA 2500

HLT

**OR
OPERATION:**

MVI A,07

MVI B,06

ORA B

STA 2000

HLT

**XOR
OPERATION:**

MVI A,03

MVI B,08

XRA B

STA 2000

HLT

INPUT:

Data Stack KeyPad Memory I/O Ports			
Start		2000	OK
Address (Hex)	Address	Data	
07D0	2000	32	
07D1	2001	0	
07D2	2002	0	
07D3	2003	0	
07D4	2004	0	
07D5	2005	0	
07D6	2006	0	
07D7	2007	0	
07D8	2008	0	
07D9	2009	0	
07DA	2010	0	
07DB	2011	0	
07DC	2012	0	
07DD	2013	0	

Data Stack KeyPad Memory I/O Ports			
Start		2000	OK
Address (Hex)	Address	Data	
07D0	2000	11	
07D1	2001	0	
07D2	2002	0	
07D3	2003	0	
07D4	2004	0	
07D5	2005	0	
07D6	2006	0	
07D7	2007	0	
07D8	2008	0	
07D9	2009	0	
07DA	2010	0	
07DB	2011	0	
07DC	2012	0	
07DD	2013	0	

Data Stack KeyPad Memory I/O Ports			
Start 2000			OK
Address (Hex)	Address	Data	
07D0	2000	48	
07D1	2001	0	
07D2	2002	0	
07D3	2003	0	
07D4	2004	0	
07D5	2005	0	
07D6	2006	0	
07D7	2007	0	
07D8	2008	0	
07D9	2009	0	
07DA	2010	0	
07DB	2011	0	
07DC	2012	0	
07DD	2013	0	

OUTPUT:

FileResetAssemblerDebugHelp

Registers

A30

BC0000

DE0000

HL0000

PSW0000

PC420E

SPFFFF

Int-Reg00

Flag

S0

Z0

AC0

P0

C0

Decimal - Hex Conversion

Decimal

0

Hex

0

To Hex

To Dec

I/O Ports

0

-

+

00

Update Port Value

Memory

0

-

+

00

Update Memory

Load me at

1<Program title>

2

3

4jmp start

5

6;data

7

8

9;code

10start: nop

11MVI A,03

12

13

14RRC

15

16

17RRC

18

19

20RRC

21

22

23RRC

24

25

26STA 2000

27

28

29HLT

DataStackKeyPadMemoryI/O Ports

Start2000OK

Address (Hex)

Address

Data

07D0200048

07D120010

07D220020

07D320030

07D420040

07D520050

07D620060

07D720070

07D820080

07D920090

07DA20100

07DB20110

07DC20120

07DD20130

Line No

Assembler Message

0

Program assembled successfully

File Reset Assembler Debug Help

Registers: A 0B, BC 08 00, DE 00 00, HL 00 00, PSW 00 00, PC 42 0D, SP FF FF, Int-Reg 00. Flag: S 0, Z 0, AC 0, P 0, C 0.

Decimal - Hex Conversion: Decimal 0, Hex 0. Buttons: To Hex, To Dec.

I/O Ports: 0, -, +, 00. Update Port Value.

Memory: 0, -, +, 00. Update Memory.

Load me at: []

```

1 ;<Program title>
2
3
4 jmp start
5
6 ;data
7
8
9 ;code
10 start: nop
11 MVI A,03
12
13
14 MVI B,08
15
16
17 XRA B
18
19
20 STA 2000
21
22
23 HLT
24
25

```

Start 2000 OK

Address (Hex)	Address	Data
07D0	2000	11
07D1	2001	0
07D2	2002	0
07D3	2003	0
07D4	2004	0
07D5	2005	0
07D6	2006	0
07D7	2007	0
07D8	2008	0
07D9	2009	0
07DA	2010	0
07DB	2011	0
07DC	2012	0
07DD	2013	0

Line No Assembler Message
0 Program assembled successfully

File Reset Assembler Debug Help

Registers: A 20, BC 00 00, DE 00 00, HL 00 00, PSW 00 00, PC 42 0E, SP FF FF, Int-Reg 00. Flag: S 0, Z 0, AC 0, P 0, C 0.

Decimal - Hex Conversion: Decimal 0, Hex 0. Buttons: To Hex, To Dec.

I/O Ports: 0, -, +, 00. Update Port Value.

Memory: 0, -, +, 00. Update Memory.

Load me at: []

```

1 ;<Program title>
2
3
4 jmp start
5
6 ;data
7
8
9 ;code
10 start: nop
11 MVI A,02
12
13
14 RLC
15
16
17 RLC
18
19
20 RLC
21
22
23 RLC
24
25
26 STA 2000
27
28
29 HLT
30
31
32

```

Start 2000 OK

Address (Hex)	Address	Data
07D0	2000	32
07D1	2001	0
07D2	2002	0
07D3	2003	0
07D4	2004	0
07D5	2005	0
07D6	2006	0
07D7	2007	0
07D8	2008	0
07D9	2009	0
07DA	2010	0
07DB	2011	0
07DC	2012	0
07DD	2013	0

Line No Assembler Message
0 Program assembled successfully

RESULT: Thus the program was executed successfully using 8085 processor simulator.