DEPARTMENT OF
ACADEMIC AFFAIRS
Discover. Learn. Empower.

NAAC
GRADE A+
ACCREDITED UNIVERSITY

Experiment - 1

Student Name: **UID:**

Branch: CSE   **Section/Group:**

Semester: 4   **Date of Performance:**

Subject Name: Microprocessor and Interfacing Lab

Subject Code: 22E-20CSP-253
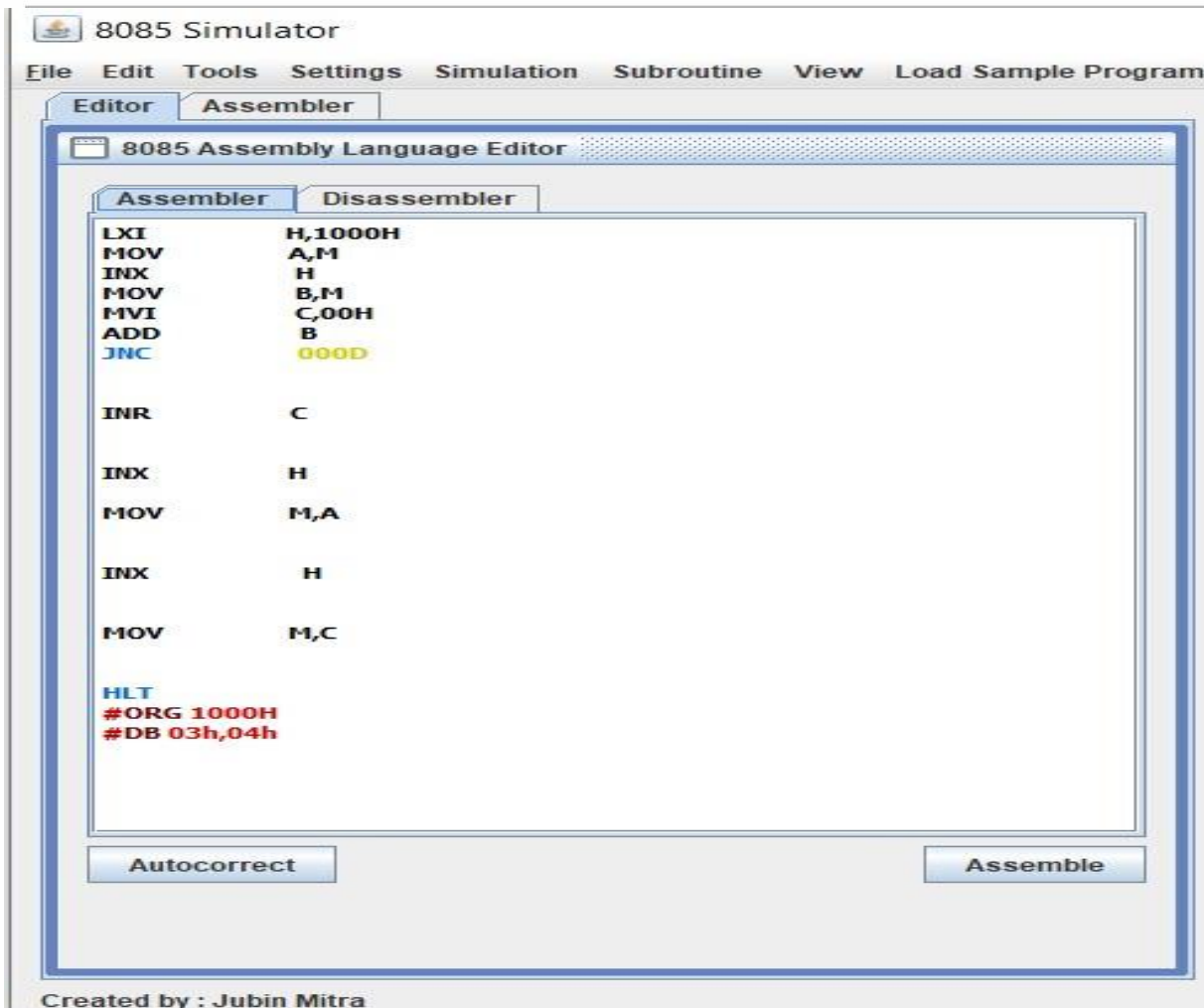
1. **Aim/Overview of the practical:**

   Addition of two 8bit numbers, sum 8 bit.

2. **Task to be done:**

   Adding 8 bit number using the Jubin application with help of some lines of codes to get the desirable result

3. Apparatus/Simulator used (For applied/experimental sciences/materials based labs)**:**

8085 Simulator, Java

8085 Simulator

File   Edit   Tools   Settings   Simulation   Subroutine   View   Load Sample Program

Editor | Assembler

8085 Assembly Language Editor

Assembler | Disassembler

```
LXI        H,1000H
MOV        A,M
INX        H
MOV        B,M
MVI        C,00H
ADD        B
JNC        000D

INR        C

INX        H

MOV        M,A

INX        H

MOV        M,C

HLT
#ORG 1000H
#DB 03h,04h
```

Autocorrect   Assemble

Created by : Jubin Mitra

4. Algorithm/Flowchart (For programming based labs):

1. Load H-L pair with address 1000H.

2. Lower-order of 1000H.

3. Higher-order of 1000H.

4. Move the 1st operand from memory to reg. A.

DEPARTMENT OF
ACADEMIC AFFAIRS
Discover. Learn. Empower.

NAAC
GRADE A+
ACCREDITED UNIVERSITY

5. Increment H-L pair.

6, Move the 2nd operand from memory to reg. B.

7. Initialize reg. C with 00H.

8. Immediate value 00H.

9. Add B with A.

10. Jump to address 000DH if there is no carry.

11. Lower-order of 000DH.

12. Higher-order of 000DH.

13. Increment reg. C.

14. Increment H-L pair.

15. Move the result from reg. A to memory.

16. Increment H-L pair.

17. Move carry from reg. C to memory.

18. Halt.

5. Description/ Code:

CODE:-

```
LXI        H,1000H
MOV        A,M
INX        H
MOV        B,M
MVI        C,00H
ADD        B
JNC        000D


INR        C


INX        H



MOV        M,A


INX        H


MOV        M,C


HLT
#ORG 1000H
#DB 03h,04h
```
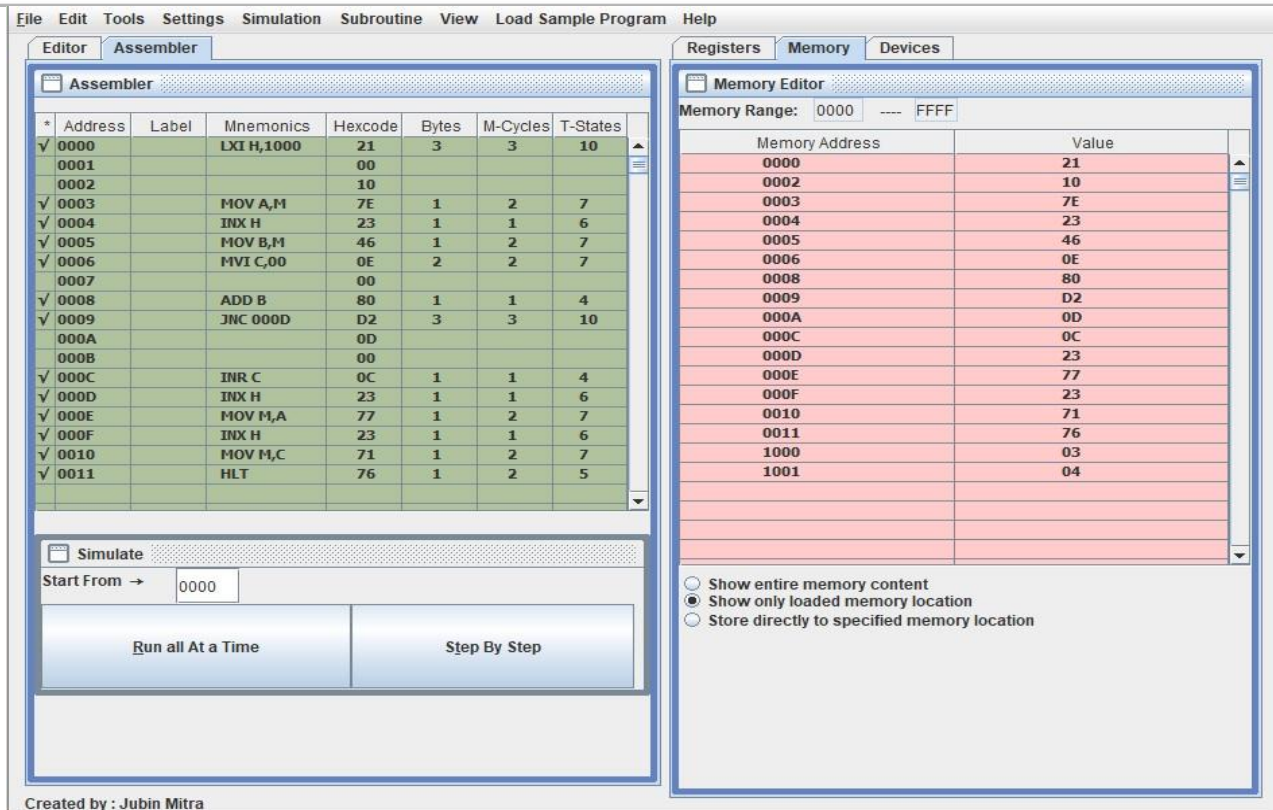
S

File Edit Tools Settings Simulation Subroutine View Load Sample Program Help

Editor | Assembler

Assembler

*	Address	Label	Mnemonics	Hexcode	Bytes	M-Cycles	T-States
√	0000		LXI H,1000	21	3	3	10
	0001			00			
	0002			10			
√	0003		MOV A,M	7E	1	2	7
√	0004		INX H	23	1	1	6
√	0005		MOV B,M	46	1	2	7
√	0006		MVI C,00	0E	2	2	7
	0007			00			
√	0008		ADD B	80	1	1	4
√	0009		JNC 000D	D2	3	3	10
	000A			0D			
	000B			00			
√	000C		INR C	0C	1	1	4
√	000D		INX H	23	1	1	6
√	000E		MOV M,A	77	1	2	7
√	000F		INX H	23	1	1	6
√	0010		MOV M,C	71	1	2	7
√	0011		HLT	76	1	2	5

Simulate

Start From →   0000

| Run all At a Time | Step By Step |

Created by : Jubin Mitra

Registers | Memory | Devices

Memory Editor

Memory Range: 0000 ---- FFFF

Memory Address	Value
0000	21
0002	10
0003	7E
0004	23
0005	46
0006	0E
0008	80
0009	D2
000A	0D
000C	0C
000D	23
000E	77
000F	23
0010	71
0011	76
1000	03
1001	04

○ Show entire memory content
● Show only loaded memory location
○ Store directly to specified memory location

Learning outcomes (What I have learnt):

1. **Handling the microprocessors**

2. **Doing mathematical operations in Microprocessor**

3. **learning about the working of microprocessors**

4. **Oerations of two 8 bit numbers**

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			