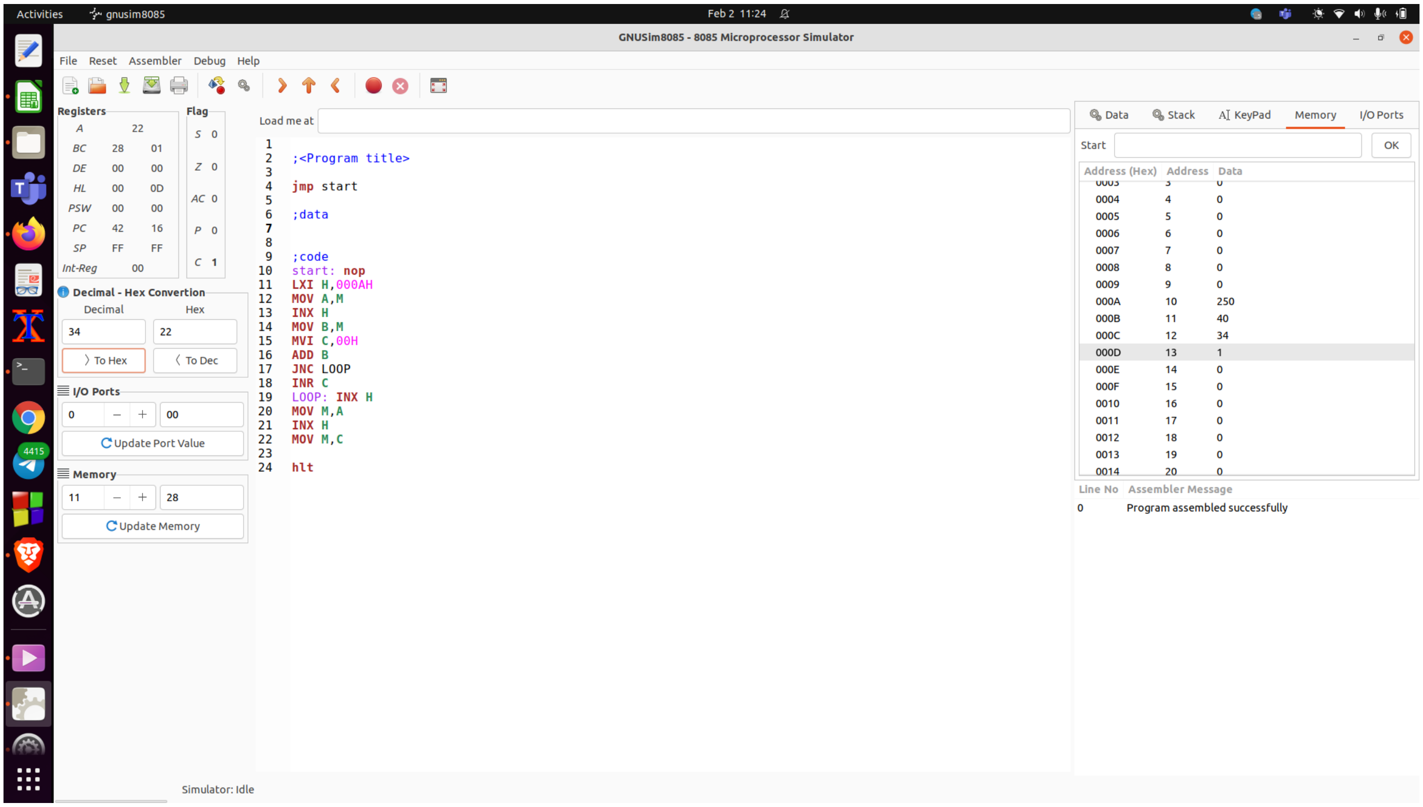
ANUSHKA GUPTA
CSE B.TECH
ROLL NO : 19075088

LAB ASSIGNMENT-1

1. ADDITION OF TWO 8 BIT NUMBERS

Case:1 performing addition with carry



```
;<Program title>

jmp start

;data


;code
start: nop
LXI H,000AH
MOV A,M
INX H
MOV B,M
MVI C,00H
ADD B
JNC LOOP
INR C
LOOP: INX H
MOV M,A
INX H
MOV M,C

hlt
```

Registers		
A	22	
BC	28	01
DE	00	00
HL	00	0D
PSW	00	00
PC	42	16
SP	FF	FF
Int-Reg	00	

Flag	
S	0
Z	0
AC	0
P	0
C	1

Decimal - Hex Convertion

Decimal	Hex
34	22

〉To Hex 〈 To Dec

I/O Ports: 0 – + 00 Update Port Value

Memory: 11 – + 28 Update Memory

Address (Hex)	Address	Data
0003	3	0
0004	4	0
0005	5	0
0006	6	0
0007	7	0
0008	8	0
0009	9	0
000A	10	250
000B	11	40
000C	12	34
000D	13	1
000E	14	0
000F	15	0
0010	16	0
0011	17	0
0012	18	0
0013	19	0
0014	20	0

Line No	Assembler Message
0	Program assembled successfully

Code:

```
;<Program title>

jmp start

;data


;code
start: nop
LXI H,000AH
MOV A,M
INX H
MOV B,M
MVI C,00H
ADD B
JNC LOOP
INR C
LOOP: INX H
MOV M,A
INX H
MOV M,C
hlt
```

Algorithm:

1) Load H-L pair with address 000AH
2) Move the first operand from memory to register A
3) Increment H-L pair
4) Move the second operand from memory to register B
5) Move immediate or initialize register C with 0 i.e 00H in hex format
6) Add B with A
7) Jump to the address denoted by LOOP if there is no carry else increment C by 1 to consider the case of carry
8) Increment H-L pair
9) Move the result of addition from register A or accumulator to the memory
10) Increment H-L pair
11) Move the vale of register C which may be 0 if no carry or 1 if there if carry to the memory

Results:

Before execution of program:
Data at address 000AH -> 0FAH = 250 in decimal format

Data at address 000BH -> 028H = 40 in decimal format

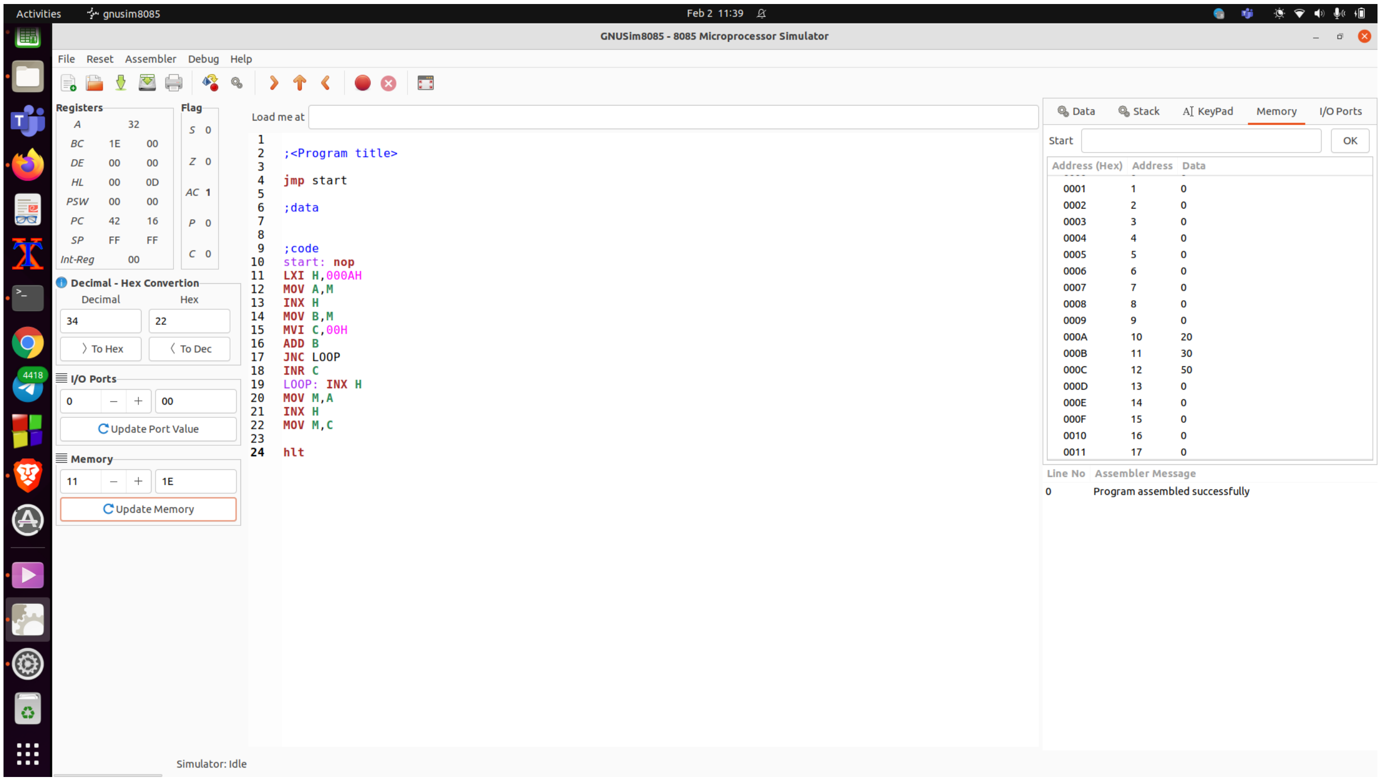
After execution of program:

Data at address 000CH -> 34 which when converted to hex format is 22H
Data at address 000DH -> 01 which when converted to hex format is 1H

On addition of 0FAH and 028H we get 122H. Which is the same result as shown by the program

The carry part is stored at 000DH address and the remaining portion is stored at 000CH on combining both, we get 122H, **hence the addition stands verified.**

Case:2 addition without carry



```
;<Program title>

jmp start

;data


;code
start: nop
LXI H,000AH
MOV A,M
INX H
MOV B,M
MVI C,00H
ADD B
JNC LOOP
INR C
LOOP: INX H
MOV M,A
INX H
MOV M,C

hlt
```

Registers

A	32	
BC	1E	00
DE	00	00
HL	00	0D
PSW	00	00
PC	42	16
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	0
AC	1
P	0
C	0

Decimal - Hex Convertion

Decimal	Hex
34	22

I/O Ports: 0 — + 00

Memory: 11 — + 1E

Address (Hex)	Address	Data
0001	1	0
0002	2	0
0003	3	0
0004	4	0
0005	5	0
0006	6	0
0007	7	0
0008	8	0
0009	9	0
000A	10	20
000B	11	30
000C	12	50
000D	13	0
000E	14	0
000F	15	0
0010	16	0
0011	17	0

Line No	Assembler Message
0	Program assembled successfully

Results:

Before execution of program:
Data at address 000AH -> 14H = 20 in decimal format

Data at address 000BH -> 1EH = 30 in decimal format

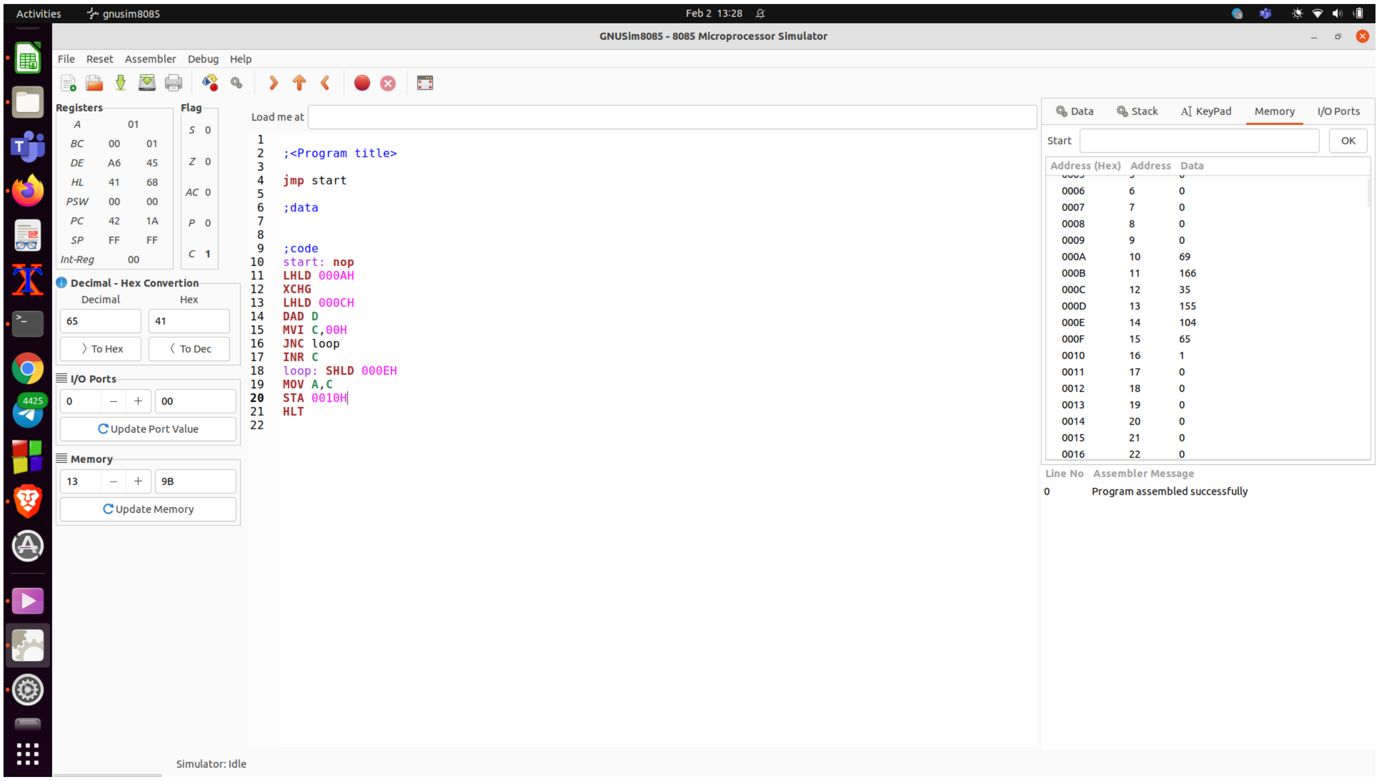
After execution of program:

Data at address 000CH -> 50 which when converted to hex format is 32H
Data at address 000DH -> 00 which when converted to hex format is 000H

On addition of 14H and 1EH we get 32H. Which is the same result as shown by the program

The carry part is stored at 000DH address and the remaining portion is stored at 000CH on combining both, we get 32H, **hence the addition stands verified**

2. Addition of two 16 bit numbers with carry



Registers

A	01	
BC	00	01
DE	A6	45
HL	41	68
PSW	00	00
PC	42	1A
SP	FF	FF
Int-Reg	00	

Flag

S	0
Z	0
AC	0
P	0
C	1

Decimal - Hex Convertion

Decimal	Hex
65	41

I/O Ports: 0 — 00

Memory: 13 — 9B

Load me at

```
;<Program title>

jmp start

;data


;code
start: nop
LHLD 000AH
XCHG
LHLD 000CH
DAD D
MVI C,00H
JNC loop
INR C
loop: SHLD 000EH
MOV A,C
STA 0010H
HLT
```

Address (Hex)	Address	Data
0006	6	0
0007	7	0
0008	8	0
0009	9	0
000A	10	69
000B	11	166
000C	12	35
000D	13	155
000E	14	104
000F	15	65
0010	16	1
0011	17	0
0012	18	0
0013	19	0
0014	20	0
0015	21	0
0016	22	0

Line No	Assembler Message
0	Program assembled successfully

Simulator: Idle

Algorithm:

1) Load H-L pair with address 000AH
2) Exchange H-L pair with D-E pair
3) Load H-L pair with data from 000CH
4) Load Register C with 0
5) Add D-E pair with H-L pair
6) Jump to the address denoted by LOOP if there is no carry else increment C by 1 to consider the case of carry
7) Store the result at location 000EH

8) Move the carry from register C to accumulator
9) Store the carry at location 0010H

Code:

```
;<Program title>
jmp start
;data
;code
start: nop
LHLD 000AH
XCHG
LHLD 000CH
DAD D
MVI C,00H
JNC loop
INR C
loop: SHLD 000EH
MOV A,C
STA 0010H
HLT
```

Results:

Before execution of program:
Data at address 000AH -> 45H
Data at address 000BH -> 0A6H
Data at address 000CH -> 23H
Data at address 000DH -> 9BH

Therefore, operation to be performed is : A645H + 9B23H

After execution of program:

Data at address 000EH -> 104 which when converted to hex format is 68H
Data at address 000FH -> 65 which when converted to hex format is 41H
Data at address 0010H -> 1 which when converted to hex format is 1H

On addition of A645H and 9B23H we get 14168H. Which is the same result as shown by the program.
The carry part is stored at 0010H address , 68H is stored at 000EH address, 41H is stored at 000FH. On combining all, we get 14168H, **hence the addition stands verified.**