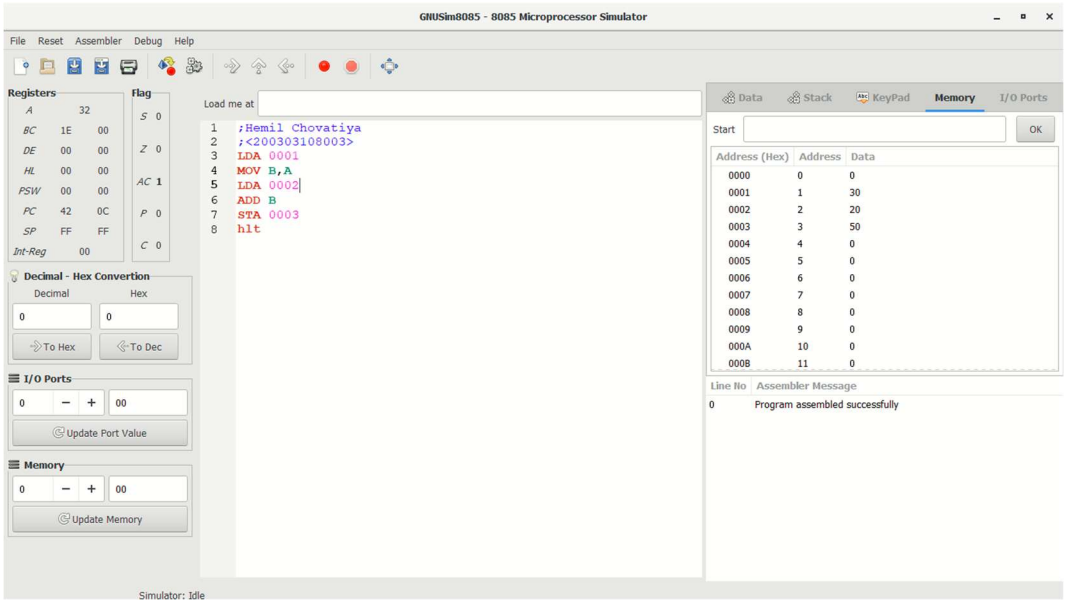
PRACTICAL 3

AIM: Write an assembly language code in GNUsim8085 to implement Addition of two 8 bit Numbers.

Theory:

Code	Meaning
LDA 0001	Load value of memory location 0001 in Accumulator A
MOV B,A	Move data from memory to accumulator
STA 0003	Store accumulator contents in memory
ADD B	Add data of memory with accumulator
HLT	Hold the program

Implementation:



Input:

0001 = 30

0002 = 20

Output:

0003 = 50

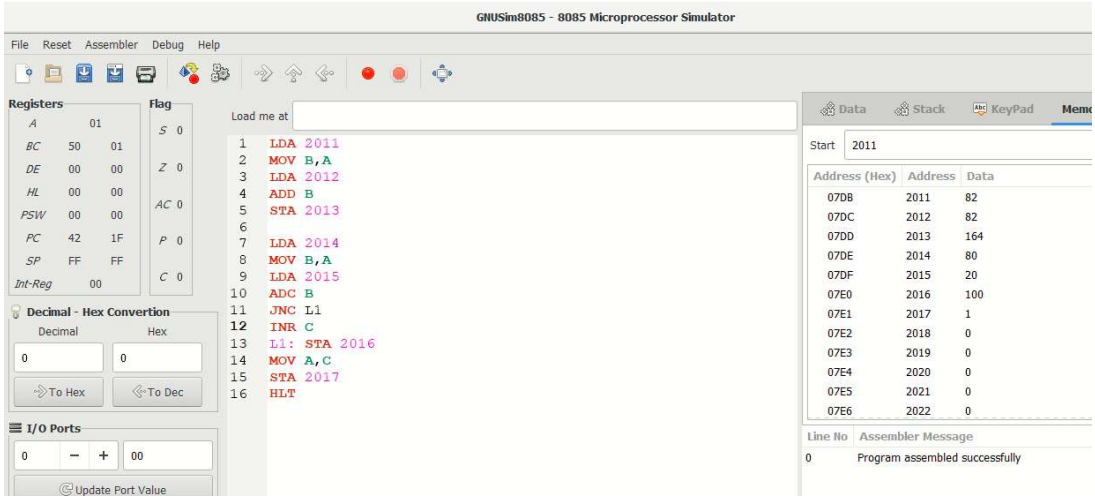
PRACTICAL 4

AIM: Write an assembly language code in GNUsim8085 to implement Addition of two 16-bit Numbers.

Theory:

Code	Meaning
LDA 2011	Load value of memory location 0001 in Accumulator A
MOV B, A	Move data from memory to accumulator
STA 0003	Store accumulator contents in memory
JNC L1	Jump if no carry CY = 0
INR C	Increment register or memory by 1
ADC B	Add register or memory to accumulator with carry
ADD B	Add data of memory with accumulator
HLT	Hold the program

IMPLEMENTATION:



INPUT: 2011=82
2012=82
2014=80
2015=20

OUTPUT: 2013=164
2016=100
2017=2