# Practical-2

- Aim: Write an assembly language code in GNUsim8085 to implement arithmetic instructions.
- Objective : Study about Arithmetic Instructions

Following is the showing the list of Arithmetic instructions with their meanings.

1. ADD :- Operand:- R,M
   Meaning :- Add register or memory, to the accumulator .
   Explanation :-
   The contents of the register or memory are added to the contents of the accumulator and the result is stored in the accumulator. Example − ADD B.

2. ADC :-
   Operand :- R,M
   Meaning :- Add register to the accumulator with carry.
   Explanation :- The contents of the register or memory & M the Carry flag are added to the contents of the accumulator and the result is stored in the accumulator.
   Example – ADC B.

3. ADI :-
   Operand :- 8-bit data
   Meaning :- Add the immediate to the accumulator.
   Explanation :-The 8-bit data is added to the contents of the accumulator and the result is stored in the accumulator.
   Example – ADI 03.

4. ACI :-
   Operand :- 8-bit data
   Meaning :- Add the immediate to the accumulator with carry.
   Explanation :- The 8-bit data and the Carry flag are added to the contents of the accumulator and the result is stored in the accumulator. Example − ACI 10.

5. LXI :-
   Operand :- Reg. pair, 16bit data
   Meaning :- Load the register pair immediate.
   Explanation :- The instruction stores 16-bit data into the register pair designated in the operand.

Example − LXI B,2005.

6. DAD :-
   Operand :- Reg. pair
   Meaning :- Add the register pair to H and L registers.
   Explanation :- The 16-bit data of the specified register pair are added to the contents of the HL register..
   Example – DAD B.

7. SBB :-
   Operand :- R,M
   Meaning :- Subtract the source and borrow from the accumulator
   Explanation :- The contents of the register or the memory & M the Borrow flag are subtracted from the contents of the accumulator and the result is placed in the accumulator.
   Example − SBB B.

8. SUB :-
   Operand :- R,M
   Meaning :- Subtract the register or the memory from the accumulator
   Explanation :- The contents of the register or the memory are subtracted from the contents of the accumulator, and the result is stored in the accumulator.. Example – SUB B.

9. INR :-
   Operand :- R,M
   Meaning :- Increment the register or the memory by 1
   Explanation :- The contents of the designated register or the memory are incremented by 1 and their result is stored at the same place.
   Example – INR B.

10. INX :-
    Operand :- R
    Meaning :- Increment register pair by 1
    Explanation :- The contents of the designated register pair are incremented by 1 and their result is stored at the same place.
    Example – INX H.

11. DCX:-
    Operand:- R
    Meaning :- Decrement the register pair by 1

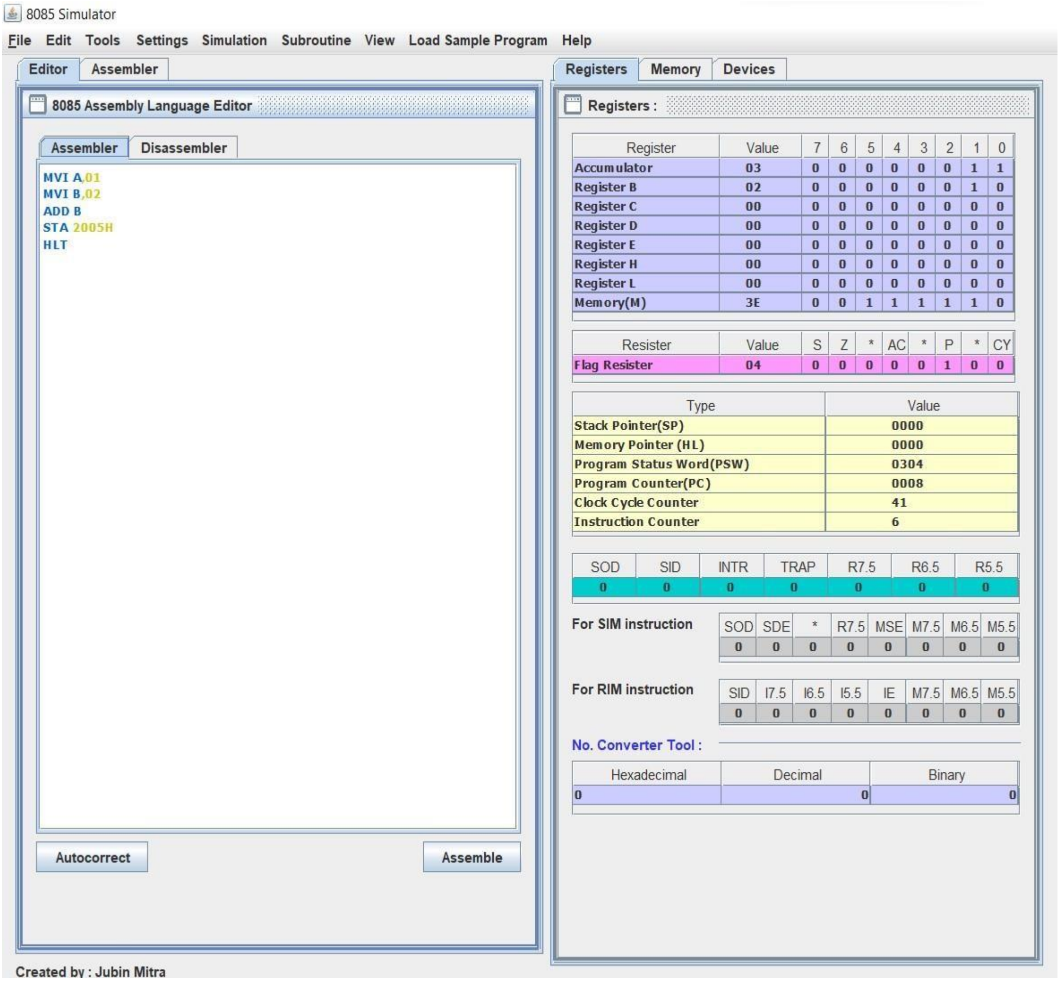
Explanation :- The contents of the designated register pair are decremented by 1 and their result is stored at the same place..
Example – DCX H.

12. DCR :-
Operand:- R,M
Meaning :- Decrement the register or the memory by 1
Explanation :- The contents of the designated register or memory are decremented by 1 and their result is stored at the same place.. Example − DCR B.

13. DAA :-
Operand :- none
Meaning :- Decimal adjust accumulator
Explanation :- The contents of the accumulator are changed from a binary value to two 4bit BCD digits.
If the value of the low-order 4-bits in the accumulator is greater than 9 or if AC flag is set, the instruction adds 6 to the low-order four bits.
If the value of the high-order 4-bits in the accumulator is greater than 9 or if the Carry flag is set, the instruction adds 6 to the high-order four bits.
Example − DAA.

- Programs of implementation arithmetic instruction.

1)



```
MVI A,01
MVI B,02
ADD B
STA 2005H
HLT
```

| Register | Value | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| Accumulator | 03 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Register B | 02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Register C | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register D | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register E | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register H | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register L | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Memory(M) | 3E | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |

| Resister | Value | S | Z | * | AC | * | P | * | CY |
|---|---|---|---|---|---|---|---|---|---|
| Flag Resister | 04 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

| Type | Value |
|---|---|
| Stack Pointer(SP) | 0000 |
| Memory Pointer (HL) | 0000 |
| Program Status Word(PSW) | 0304 |
| Program Counter(PC) | 0008 |
| Clock Cycle Counter | 41 |
| Instruction Counter | 6 |

| SOD | SID | INTR | TRAP | R7.5 | R6.5 | R5.5 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For SIM instruction

| SOD | SDE | * | R7.5 | MSE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

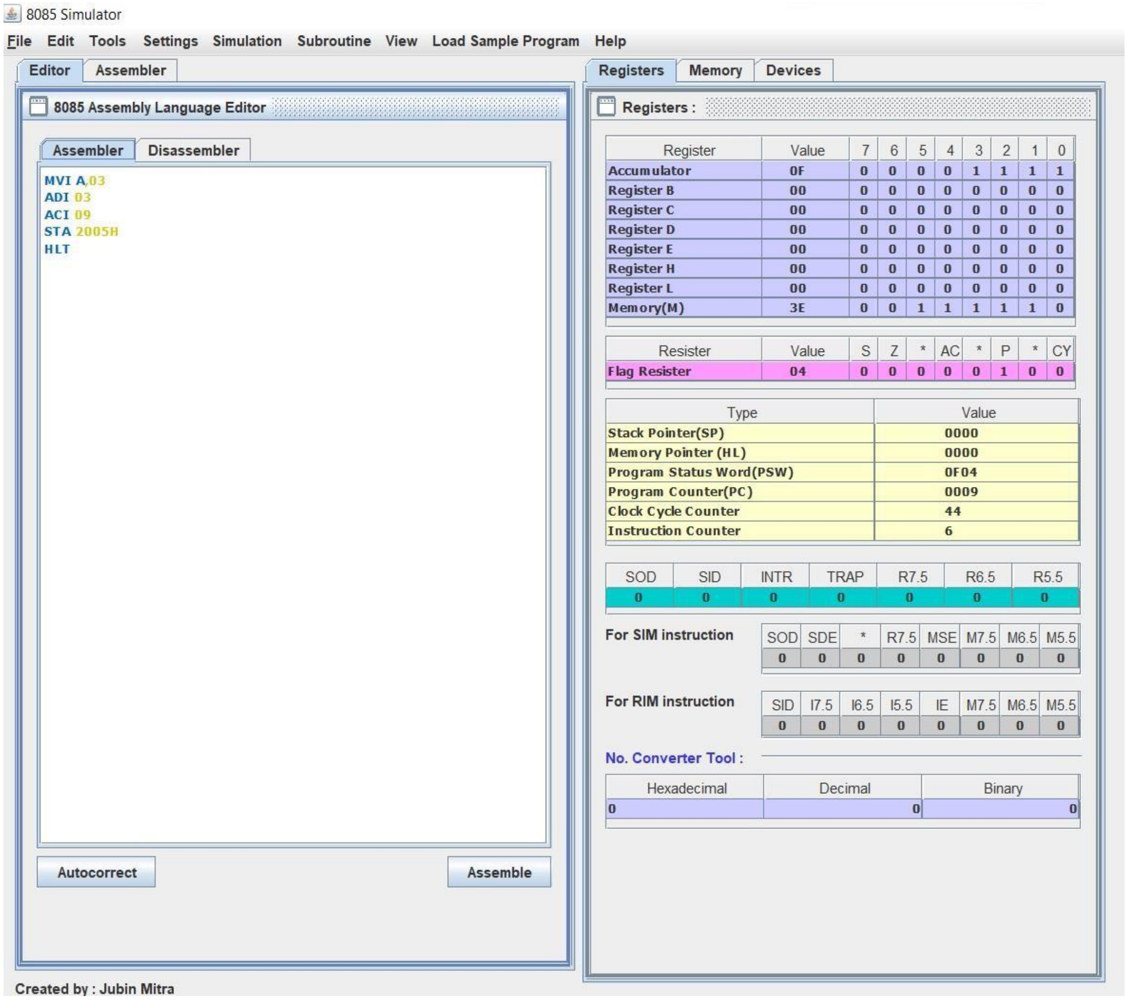
For RIM instruction

| SID | I7.5 | I6.5 | I5.5 | IE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

No. Converter Tool :

| Hexadecimal | Decimal | Binary |
|---|---|---|
| 0 | 0 | 0 |

2)



```
MVI A,03
ADI 03
ACI 09
STA 2005H
HLT
```

| Register | Value | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| Accumulator | 0F | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| Register B | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register C | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register D | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register E | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register H | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register L | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Memory(M) | 3E | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |

| Resister | Value | S | Z | * | AC | * | P | * | CY |
|---|---|---|---|---|---|---|---|---|---|
| Flag Resister | 04 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |

| Type | Value |
|---|---|
| Stack Pointer(SP) | 0000 |
| Memory Pointer (HL) | 0000 |
| Program Status Word(PSW) | 0F04 |
| Program Counter(PC) | 0009 |
| Clock Cycle Counter | 44 |
| Instruction Counter | 6 |

| SOD | SID | INTR | TRAP | R7.5 | R6.5 | R5.5 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For SIM instruction

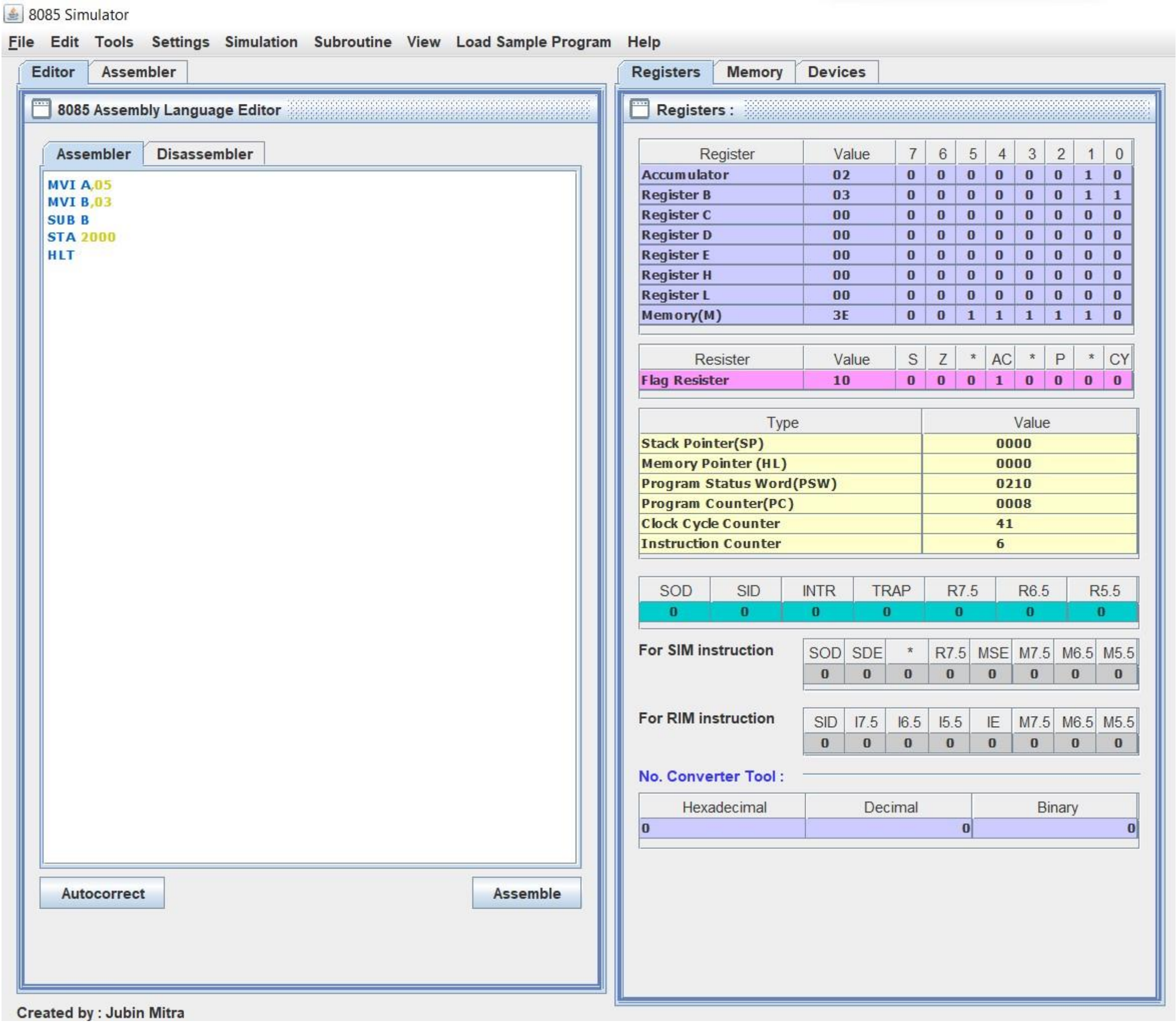
| SOD | SDE | * | R7.5 | MSE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For RIM instruction

| SID | I7.5 | I6.5 | I5.5 | IE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

| Hexadecimal | Decimal | Binary |
|---|---|---|
| 0 | 0 | 0 |

3)



```
MVI A,05
MVI B,03
SUB B
STA 2000
HLT
```

| Register | Value | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| Accumulator | 02 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Register B | 03 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| Register C | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register D | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register E | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register H | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register L | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Memory(M) | 3E | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 0 |

| Resister | Value | S | Z | * | AC | * | P | * | CY |
|---|---|---|---|---|---|---|---|---|---|
| Flag Resister | 10 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |

| Type | Value |
|---|---|
| Stack Pointer(SP) | 0000 |
| Memory Pointer (HL) | 0000 |
| Program Status Word(PSW) | 0210 |
| Program Counter(PC) | 0008 |
| Clock Cycle Counter | 41 |
| Instruction Counter | 6 |

| SOD | SID | INTR | TRAP | R7.5 | R6.5 | R5.5 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For SIM instruction

| SOD | SDE | * | R7.5 | MSE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

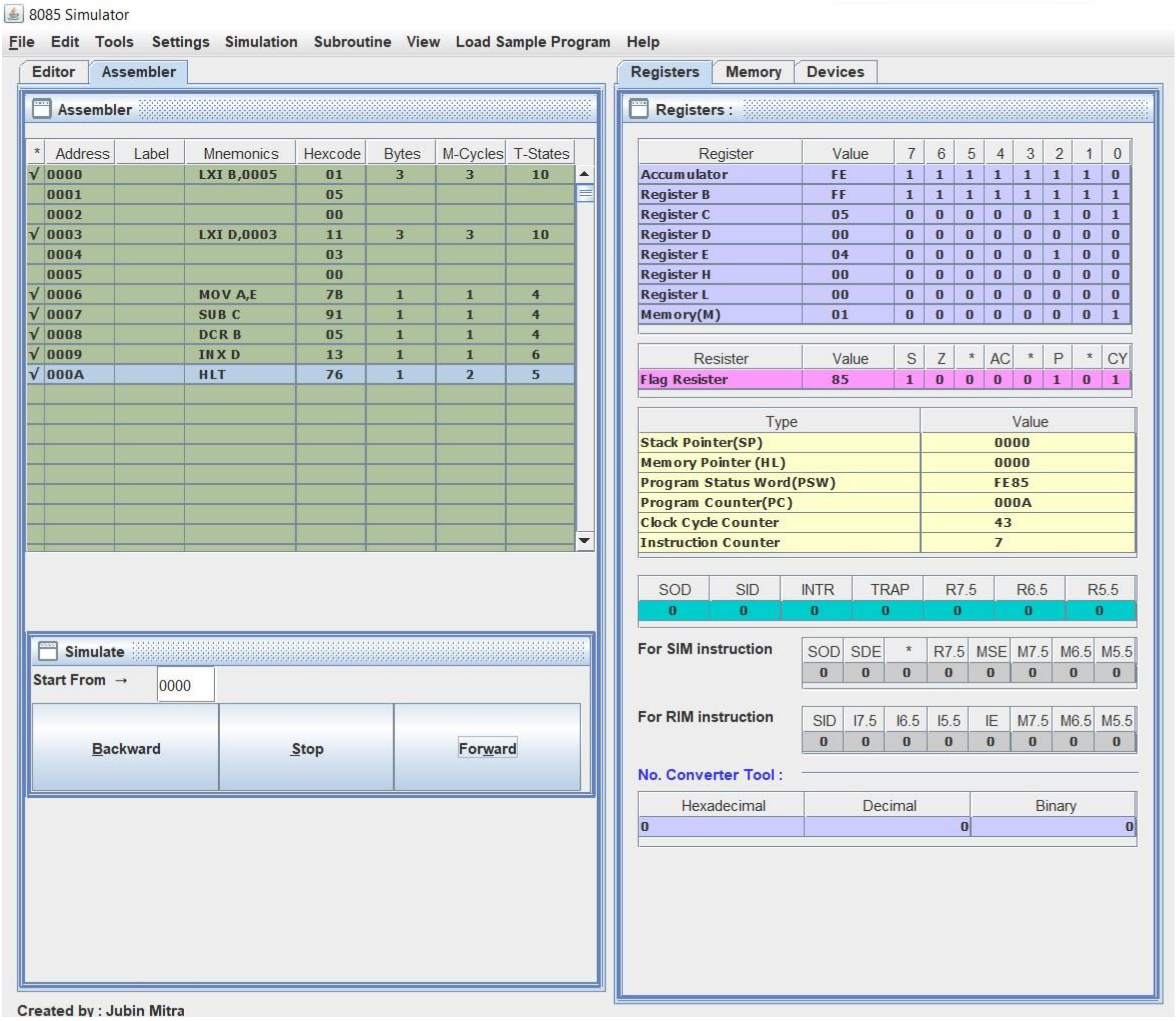
For RIM instruction

| SID | I7.5 | I6.5 | I5.5 | IE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

No. Converter Tool :

| Hexadecimal | Decimal | Binary |
|---|---|---|
| 0 | 0 | 0 |

4)


8085 Simulator
File Edit Tools Settings Simulation Subroutine View Load Sample Program Help
Editor
Assembler
Registers
Memory
Devices

Assembler

| * | Address | Label | Mnemonics | Hexcode | Bytes | M-Cycles | T-States |
|---|---|---|---|---|---|---|---|
| √ | 0000 | | LXI B,0005 | 01 | 3 | 3 | 10 |
| | 0001 | | | 05 | | | |
| | 0002 | | | 00 | | | |
| √ | 0003 | | LXI D,0003 | 11 | 3 | 3 | 10 |
| | 0004 | | | 03 | | | |
| | 0005 | | | 00 | | | |
| √ | 0006 | | MOV A,E | 7B | 1 | 1 | 4 |
| √ | 0007 | | SUB C | 91 | 1 | 1 | 4 |
| √ | 0008 | | DCR B | 05 | 1 | 1 | 4 |
| √ | 0009 | | INX D | 13 | 1 | 1 | 6 |
| √ | 000A | | HLT | 76 | 1 | 2 | 5 |

Simulate

Start From → 0000

Registers :

| Register | Value | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| Accumulator | FE | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| Register B | FF | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| Register C | 05 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| Register D | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register E | 04 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| Register H | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register L | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Memory(M) | 01 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |

| Resister | Value | S | Z | * | AC | * | P | * | CY |
|---|---|---|---|---|---|---|---|---|---|
| Flag Resister | 85 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

| Type | Value |
|---|---|
| Stack Pointer(SP) | 0000 |
| Memory Pointer (HL) | 0000 |
| Program Status Word(PSW) | FE85 |
| Program Counter(PC) | 000A |
| Clock Cycle Counter | 43 |
| Instruction Counter | 7 |

| SOD | SID | INTR | TRAP | R7.5 | R6.5 | R5.5 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For SIM instruction

| SOD | SDE | * | R7.5 | MSE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For RIM instruction

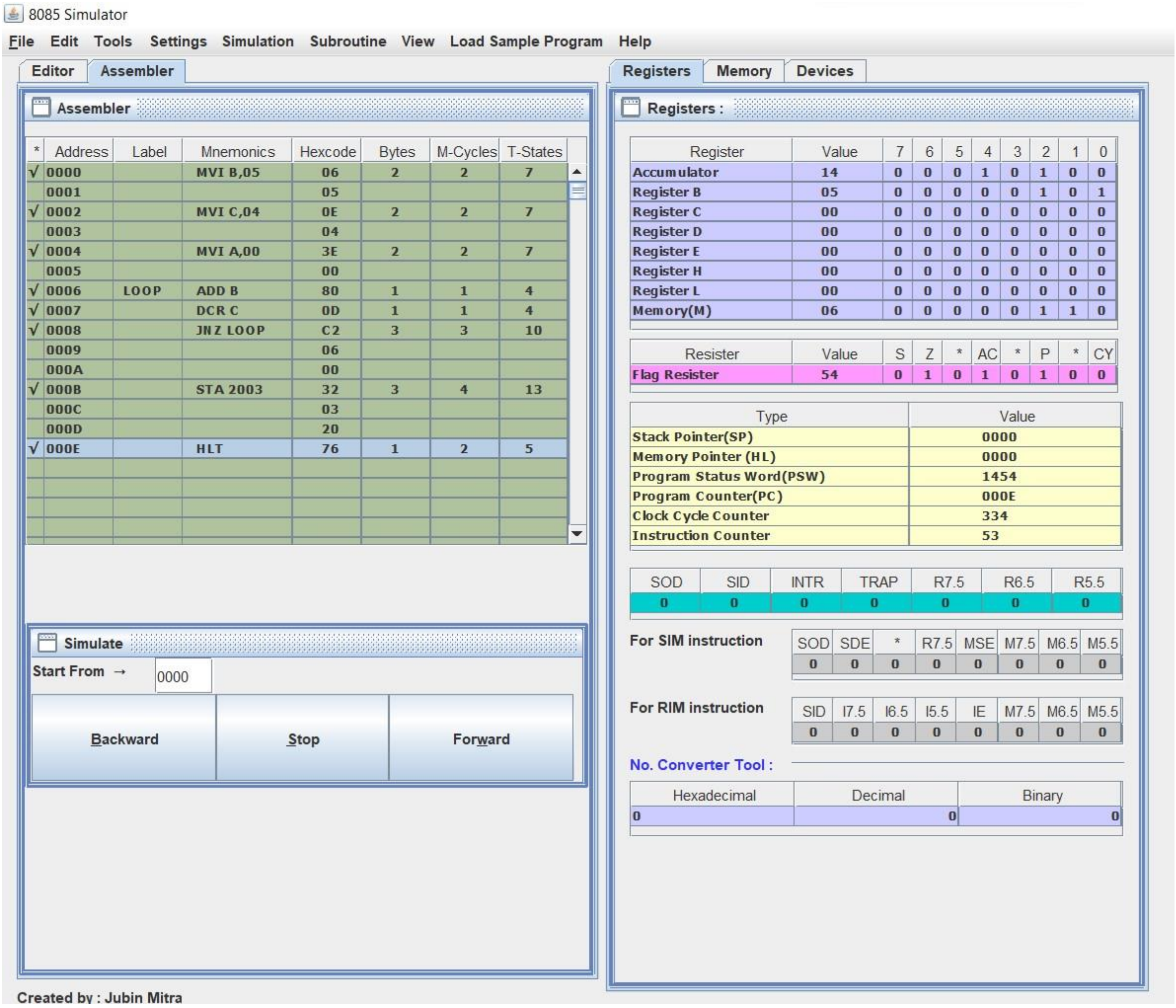
| SID | I7.5 | I6.5 | I5.5 | IE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

No. Converter Tool :

| Hexadecimal | Decimal | Binary |
|---|---|---|
| 0 | 0 | 0 |

Created by : Jubin Mitra

5)



| * | Address | Label | Mnemonics | Hexcode | Bytes | M-Cycles | T-States |
|---|---|---|---|---|---|---|---|
| √ | 0000 | | MVI B,05 | 06 | 2 | 2 | 7 |
| | 0001 | | | 05 | | | |
| √ | 0002 | | MVI C,04 | 0E | 2 | 2 | 7 |
| | 0003 | | | 04 | | | |
| √ | 0004 | | MVI A,00 | 3E | 2 | 2 | 7 |
| | 0005 | | | 00 | | | |
| √ | 0006 | LOOP | ADD B | 80 | 1 | 1 | 4 |
| √ | 0007 | | DCR C | 0D | 1 | 1 | 4 |
| √ | 0008 | | JNZ LOOP | C2 | 3 | 3 | 10 |
| | 0009 | | | 06 | | | |
| | 000A | | | 00 | | | |
| √ | 000B | | STA 2003 | 32 | 3 | 4 | 13 |
| | 000C | | | 03 | | | |
| | 000D | | | 20 | | | |
| √ | 000E | | HLT | 76 | 1 | 2 | 5 |

Start From → 0000

| Register | Value | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|---|---|
| Accumulator | 14 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| Register B | 05 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |
| Register C | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register D | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register E | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register H | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Register L | 00 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| Memory(M) | 06 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

| Resister | Value | S | Z | * | AC | * | P | * | CY |
|---|---|---|---|---|---|---|---|---|---|
| Flag Resister | 54 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 |

| Type | Value |
|---|---|
| Stack Pointer(SP) | 0000 |
| Memory Pointer (HL) | 0000 |
| Program Status Word(PSW) | 1454 |
| Program Counter(PC) | 000E |
| Clock Cycle Counter | 334 |
| Instruction Counter | 53 |

| SOD | SID | INTR | TRAP | R7.5 | R6.5 | R5.5 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For SIM instruction

| SOD | SDE | * | R7.5 | MSE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

For RIM instruction

| SID | I7.5 | I6.5 | I5.5 | IE | M7.5 | M6.5 | M5.5 |
|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

No. Converter Tool :

| Hexadecimal | Decimal | Binary |
|---|---|---|
| 0 | 0 | 0 |