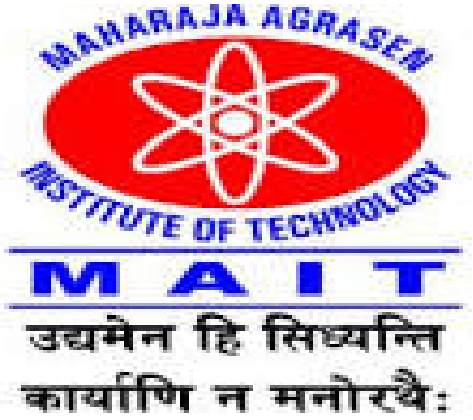
COMPUTER ORGANISATION
AND
ARCHITECTURE LAB

ETCS - 260



Faculty: Mr. Varun Goel

Submitted by: Khushi
Roll No: 03114813120
Branch: ITE
Semester: 4th
Group: ITE-2

INDEX

S.No.	Experiment Name	Date of Submission	Teacher's sign/ Remarks
1.	To draw and explain (i) Block diagram and pin diagram of 8085 (ii)Instruction set of 8085	21/02/22	
2.	Write programs to perform: - **(A)** addition of two 8-bit binary numbers with carry. (B) addition of two 8-bit binary numbers without carry.	28/02/22	
3.	Write programs to perform: - (A) Subtraction of two 8-bit binary numbers with borrows. (B) Subtraction of two 8-bit binary numbers without borrows.	7/03/22	
4.	Finding one's complement of a number.	14/03/22	
5.	Finding Two's complement of a number.	21/03/22	
6.	Multiplication of two 8-bit binary numbers.	28/03/22	
7.	Find the smallest and largest numbers from the given series in 8085.	18/04/22	
8.	Calculate the sum of a series of numbers.	25/04/22	
9.	Find the factorial of a number.	2/05/22	

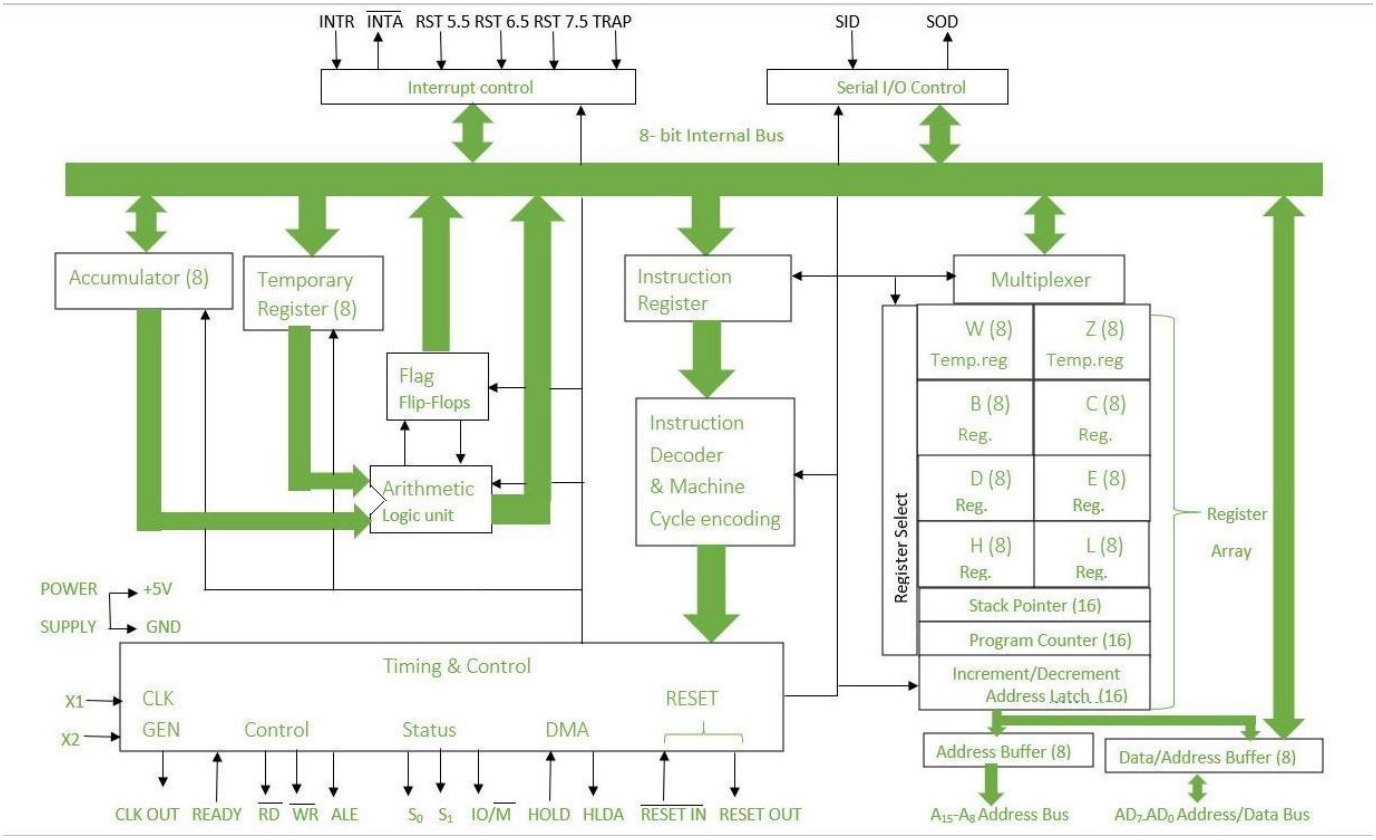
10.	Reverse an 8-bit number in 8085.	9/05/22	
11.	Sort an array in ascending and descending order in 8085.	23/05/22	
12.	Division of two 8 bit numbers in 8085	30/05/22	

EXPERIMENT - 1

AIM: To draw and explain

(i) Block diagram and pin diagram of 8085

Block diagram of 8085:



Arithmetic and Logic Unit (ALU):

It is used to perform mathematical operations like addition, multiplication, subtraction, division, decrement, increment, etc. Different operations are carried out in ALU: ***Logical operations, Bit-Shifting Operations, and Arithmetic Operations.***

Flag Register: It is an 8-bit register that stores either 0 or 1 depending upon which value is stored in the accumulator. Flag Register contains 8-bit out of which 5-bits are important and the rest of 3-bits are Don't Care conditions. The flag register is a dynamic register because after each operation to check whether the result is zero, positive or negative whether there is any overflow occurred or not, or for comparison of two 8-bit numbers carry flag is checked .so for numerous operations to check the contents of the accumulator and from that contents, if we want to check the behavior of given result then we can use Flag register to verify and check.

Different fields of a flag register:

1. Carry Flag
2. Parity Flag
3. Auxiliary Carry Flag
4. Zero Flag
5. Sign Flag

Accumulator: Accumulator is used to performing I/O, arithmetic, and logical operations. It is connected to ALU and the internal data bus.

General Purpose Register: There are six general-purpose registers. These registers can hold 8-bit values. These 8-bit registers are B,C,D,E,H,L. These registers work as 16-bit registers when they work in pairs like B-C, D-E, and H-L. Here registers W and Z are reserved registers

Program Counter: Program Counter holds the address value of the memory to the next instruction that is to be executed. It is a 16-bit register.

Stack Pointer: It works like a stack. In stack, the content of the register is stored that is later used in the program. It is a 16-bit special register. The stack pointer is part of memory but it is part of Stack operations, unlike random memory access. Stack pointer works in a continuous and contiguous part of the memory, whereas Progam Counter works in random memory locations.

Temporary Register: It is an 8-bit register that holds data values during arithmetic and logical operations.

Instruction register and decoder:

It is an 8-bit register that holds the instruction code that is being decoded. The instruction is fetched from the memory.

Timing and control unit:

The timing and control unit comes under the CPU section, and it controls the flow of data from the CPU to other devices. It is also used to control the operations performed by the microprocessor and the devices connected to it. There are certain timing and control signals like Control signals, DMA Signals, RESET signals, and Status signals.

Interrupt control:

Whenever a microprocessor is executing the main program and if suddenly an interrupt occurs, the microprocessor shifts the control from the main program to process the incoming request. After the request is completed, the control goes back to the main program. There are 5 interrupt signals in 8085 microprocessors: INTR, TRAP, RST 7.5, RST 6.5, and RST 5.5.

Priorities of Interrupts: TRAP > RST 7.5 > RST 6.5 > RST 5.5 > INTR

Address bus and data bus:

The data bus is bidirectional and carries the data which is to be stored. The address bus is unidirectional and carries the location where data is to be stored.

Serial Input/output control:

It controls the serial data communication by using Serial input data and Serial output data.

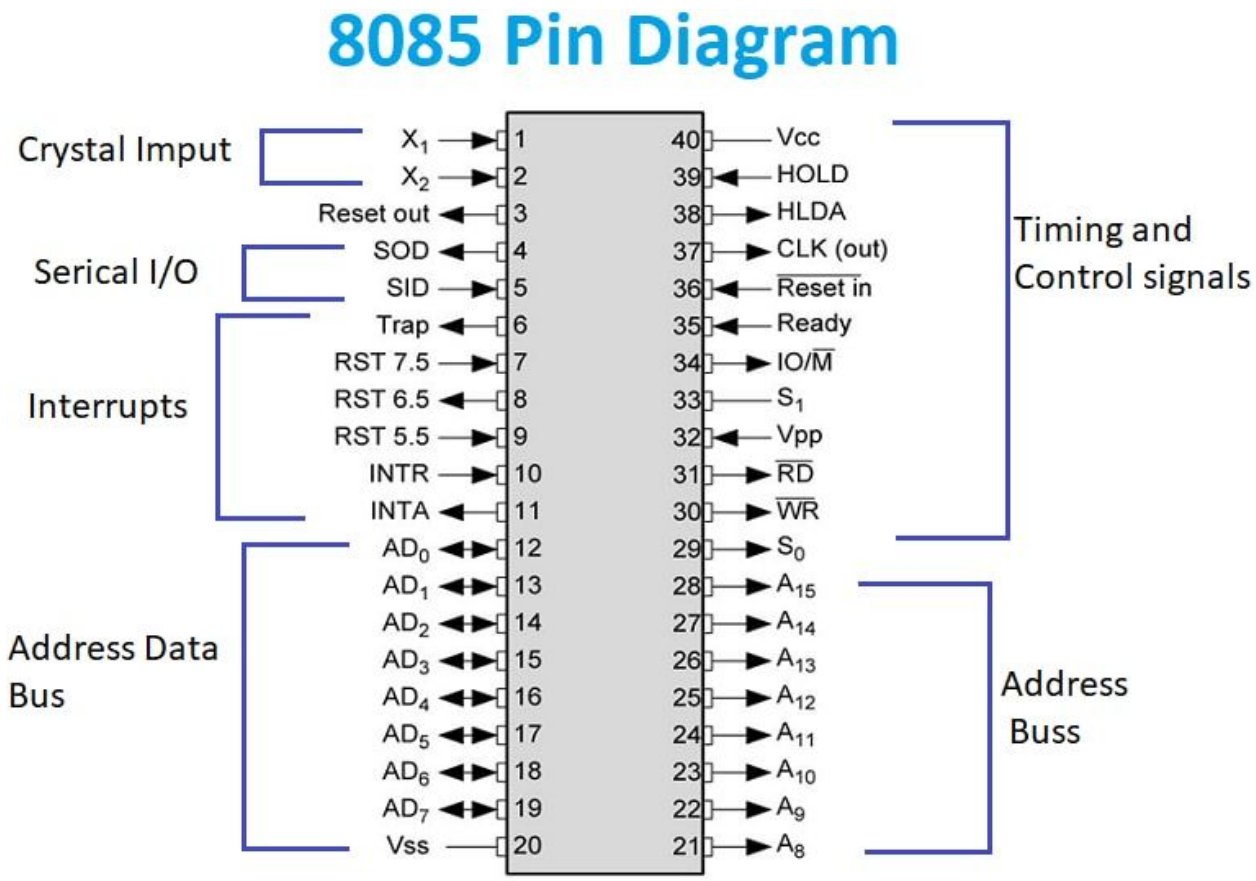
The flow of an Instruction Cycle in 8085 Architecture:

1. Execution starts with Program Counter. It starts program execution with the next address field. it fetches an instruction from the memory location pointed by Program Counter.
2. For address fetching from the memory, multiplexed address/data bus acts as an address bus and after fetching instruction this address bus will now acts as a data bus and extract data from the specified memory location and send this data on an

8-bit internal bus For multiplexed address/data bus Address Latch Enable(ALE) Pin is used. If ***ALE = 1 (Multiplexed bus is Address Bus otherwise it acts as Data Bus).***

3. After data fetching data will go into the Instruction Register it will store data fetched from memory and now data is ready for decoding so for this Instruction decoder register is used.
4. After that timing and control signal circuit comes into the picture. *It sends control signals all over the microprocessor to tell the microprocessor whether the given instruction is for READ/WRITE and whether it is for MEMORY/I-O Device activity.*
5. Hence according to timing and control signal pins, logical and arithmetic operations are performed and according to that data fetching from the different registers is done by a microprocessor, and mathematical operation is carried out by ALU. And according to operations Flag register changes dynamically.
6. With the help of Serial I/O data pin(SID or SOD Pins) we can send or receive input/output to external devices .in this way execution cycle is carried out.
7. ***While execution is going on if there is any interrupt detected then it will stop the execution of the current process and Invoke Interrupt Service Routine (ISR)*** Function. Which will stop the current execution and do execution of the current occurred interrupted after that normal execution will be performed.

Pin Diagram of 8085:



1. Address Bus and Data Bus:

An address bus is a group of sixteen lines i.e A0-A15. The address bus is unidirectional, i.e., bits flow in one direction from the microprocessor unit to the peripheral devices and uses the high order address bus.

2. Control and Status Signals:

- **ALE** – It is an Address Latch Enable signal. It goes high during first T state of a machine cycle and enables the lower 8-bits of the address, if its value is 1 otherwise data bus is activated.
- **IO/M'** – It is a status signal which determines whether the address is for input-output or memory. When it is high(1) the address on the address bus is for input-output devices. When it is low(0) the address on the address bus is for the memory.

- **SO, S1** – These are status signals. They distinguish the various types of operations such as halt, reading, instruction fetching or writing.
- **RD'** – It is a signal to control READ operation. When it is low the selected memory or input-output device is read.
- **WR'** – It is a signal to control WRITE operation. When it goes low the data on the data bus is written into the selected memory or I/O location.
- **READY** – It senses whether a peripheral is ready to transfer data or not. If READY is high(1) the peripheral is ready. If it is low(0) the microprocessor waits till it goes high. It is useful for interfacing low-speed devices.

3. Power Supply and Clock Frequency:

- **Vss** – Ground Reference
- **Vcc** – +5v power supply
- **XI, X2** – A crystal is connected at these two pins. The frequency is internally divided by two, therefore, to operate a system at 3MHZ the crystal should have a frequency of 6MHZ.
- **CLK (OUT)** – This signal can be used as the system clock for other devices.

4. Interrupts and Peripheral Initiated Signals:

The 8085 has five interrupt signals that can be used to interrupt program execution.

(i) INTR

(ii) RST 7.5

(iii) RST 6.5

(iv) RST 5.5

(v) TRAP

The microprocessor acknowledges Interrupt Request by INTA signal. In addition to Interrupts, there are three externally initiated signals: RESET, HOLD, and READY. To respond to the HOLD requests, it has one signal called HLDA.

- **INTR** – It is an interrupt request signal.
- **INTA'** – It is an interrupt acknowledgment sent by the microprocessor after INTR is received.

5. Reset Signals:

- **RESET IN'** – When the signal on this pin is low(0), the program counter is set to zero, the buses are tri-stated and the microprocessor unit is reset.
- **RESET OUT** – This signal indicates that the MPU is being reset. The signal can be used to reset other devices.

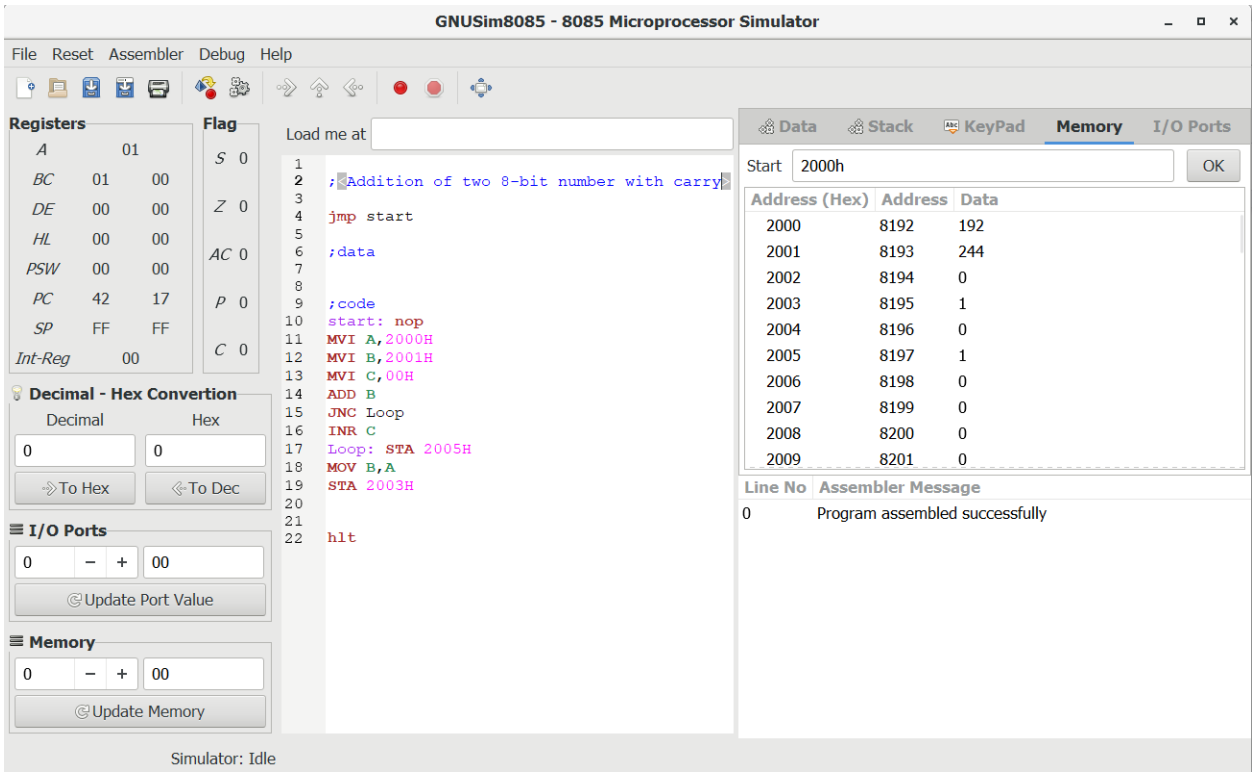
6. DMA Signals:

- **HOLD** – It indicates that another device is requesting the use of the address and data bus. Having received the HOLD request the microprocessor relinquishes the use of the buses as soon as the current machine cycle is completed. Internal processing may continue. After the removal of the HOLD signal, the processor regains the bus.
- **HLDA** – It is a signal which indicates that the hold request has been received after the removal of a HOLD request, the HLDA goes low.

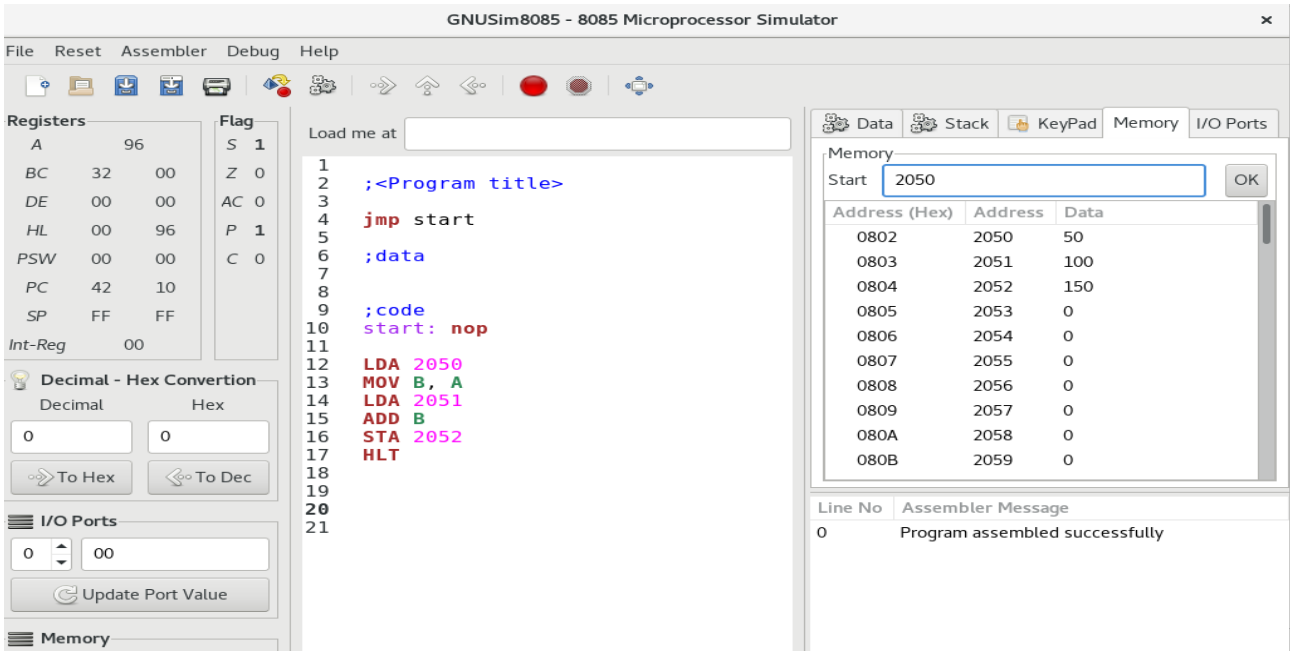
EXPERIMENT - 2

AIM: Write programs to perform

i) addition of two 8-bit binary numbers with carry.



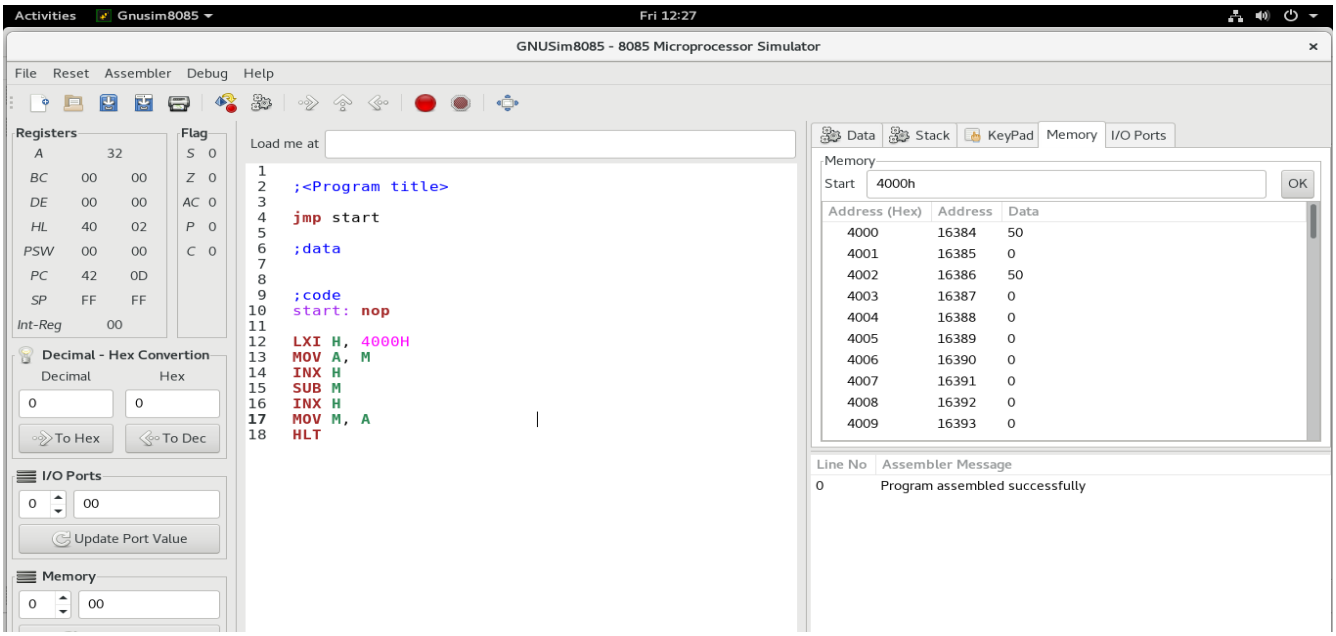
ii) addition of two 8-bit binary numbers without carry.



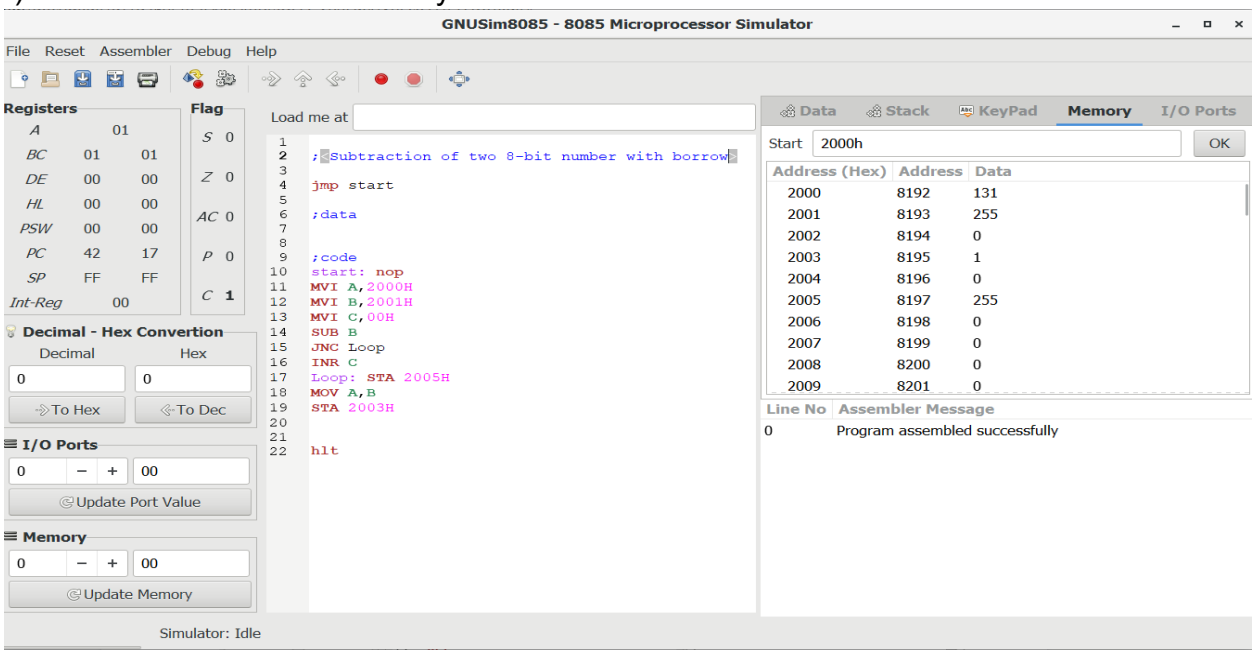
EXPERIMENT - 3

AIM: Write programs to perform

i) subtraction of two 8-bit binary numbers without borrow.

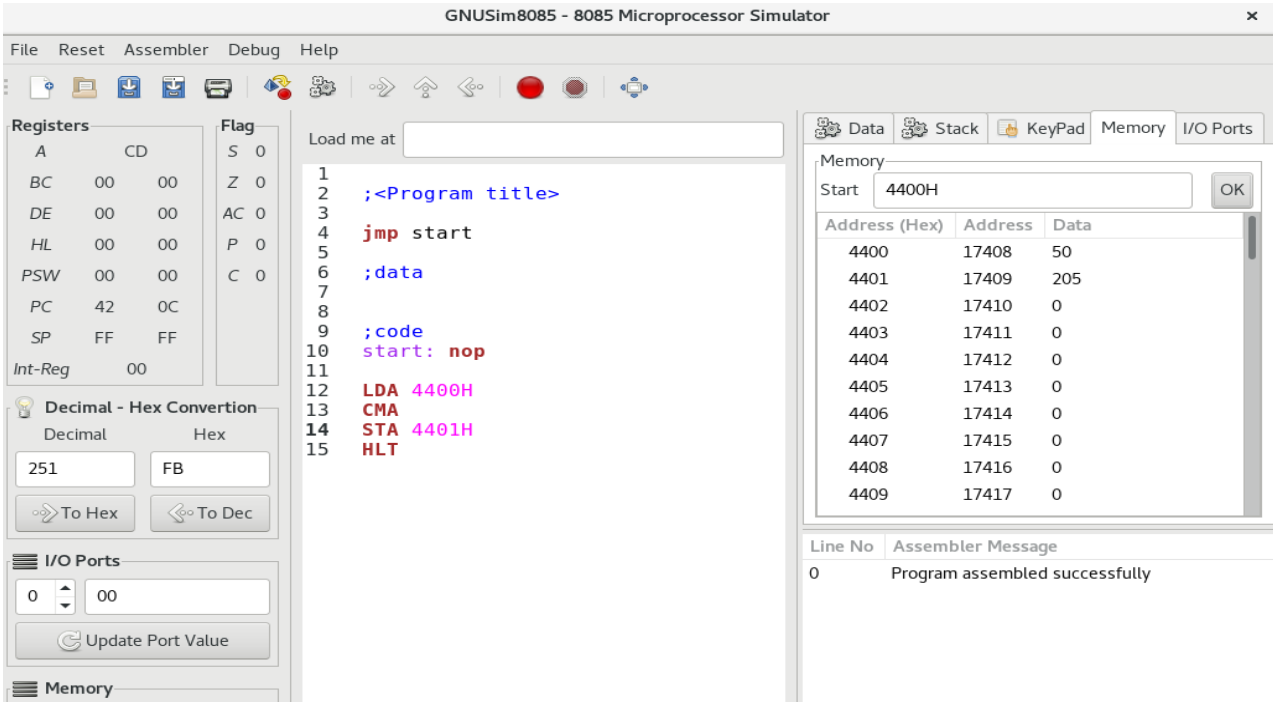


ii) subtraction of two 8-bit binary numbers with borrow.



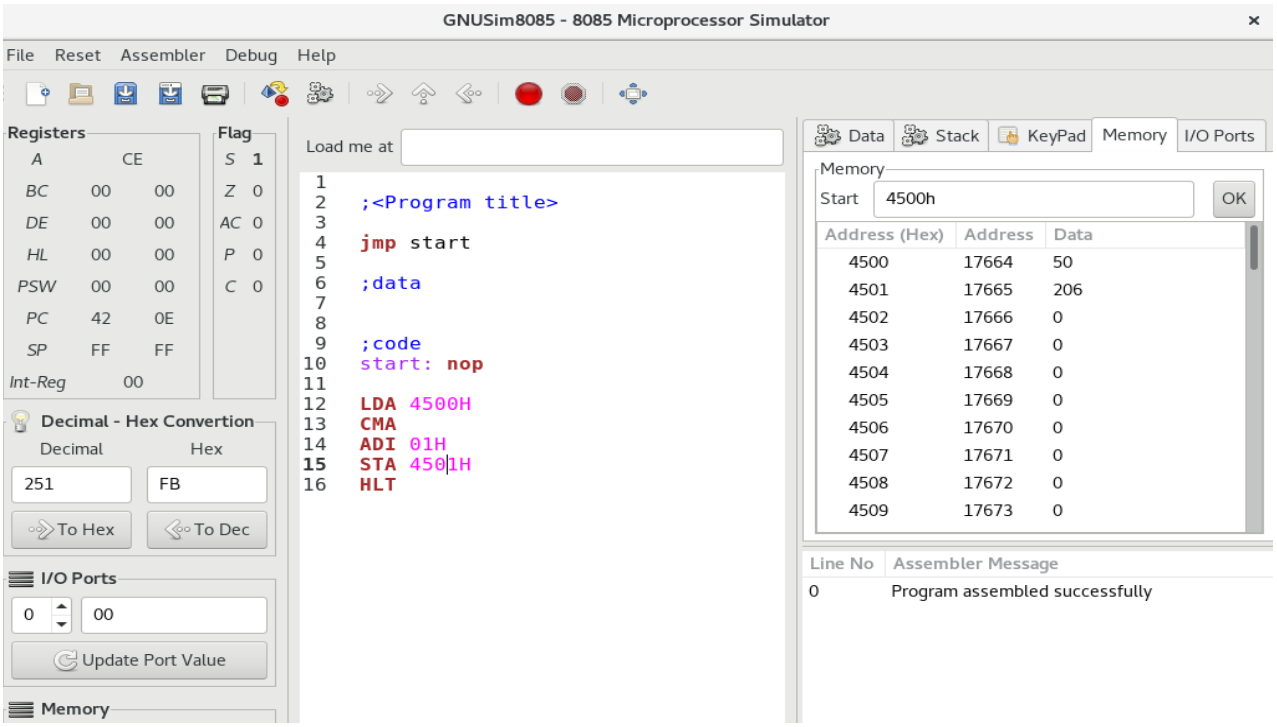
EXPERIMENT - 4

AIM: Finding one's complement of a number.



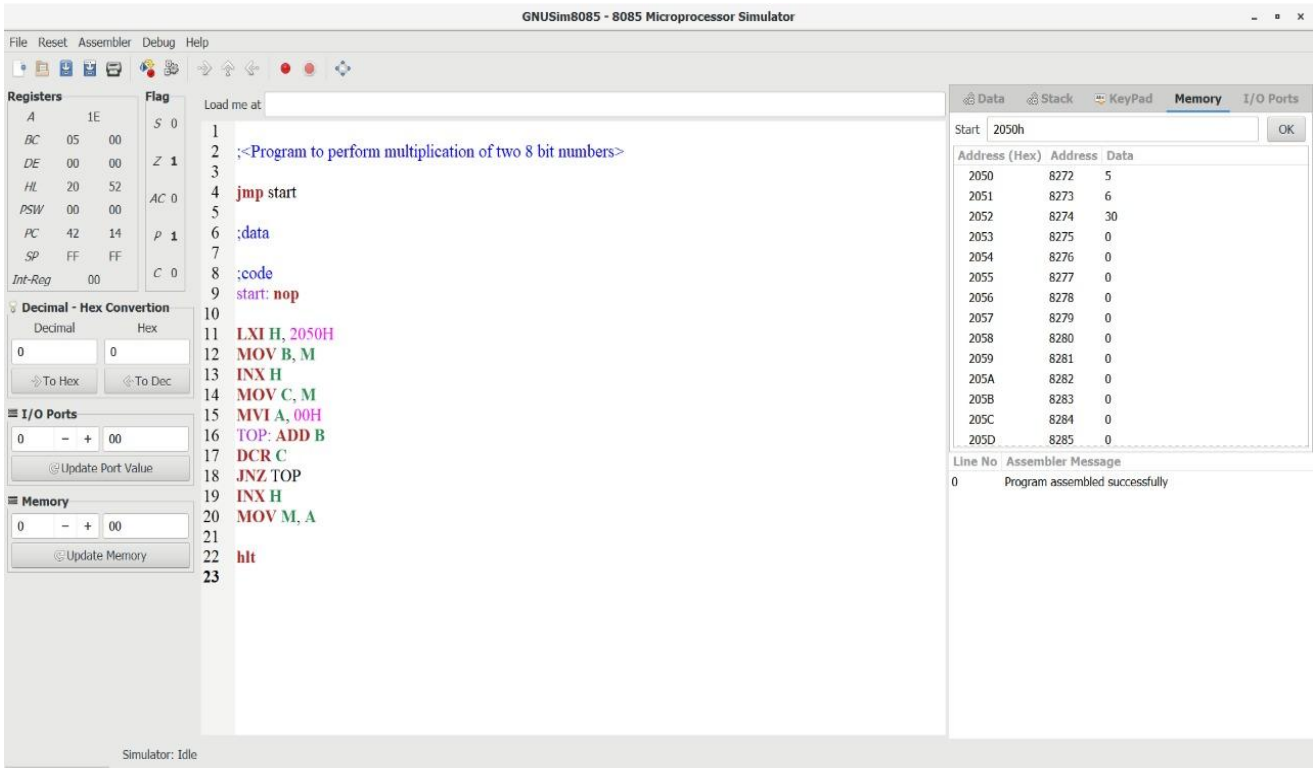
EXPERIMENT - 5

AIM: Finding two's complement of a number



EXPERIMENT - 6

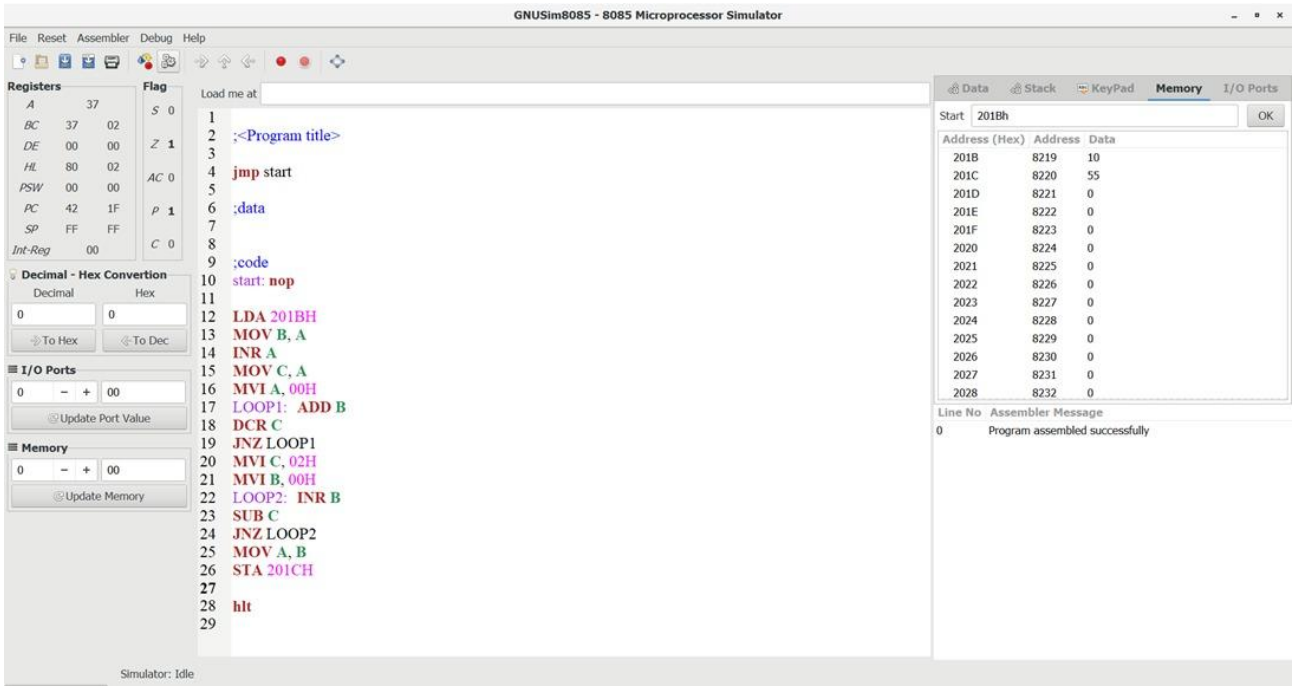
AIM: Multiplication of two 8-bit binary numbers.




```
;<Program to perform multiplication of two 8 bit numbers>

jmp start

;data

;code
start: nop

LXI H, 2050H
MOV B, M
INX H
MOV C, M
MVI A, 00H
TOP: ADD B
DCR C
JNZ TOP
INX H
MOV M, A

hlt
```

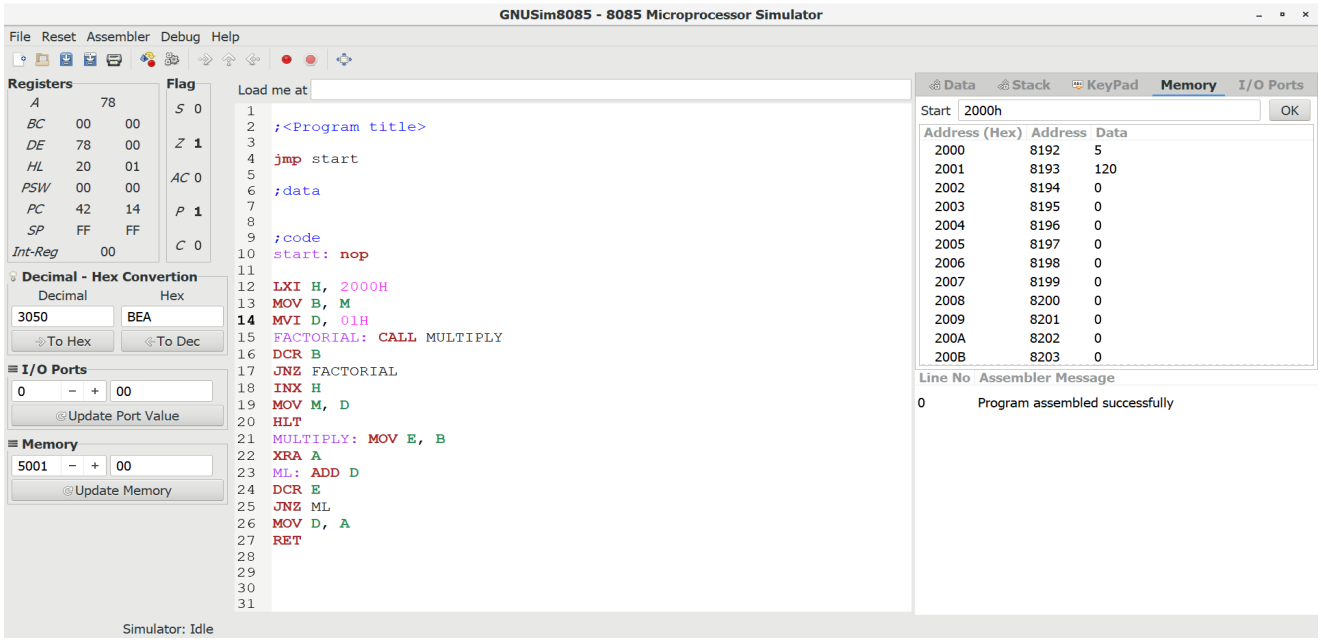
EXPERIMENT - 8

AIM: Write a program to find the sum of a series of n consecutive numbers.



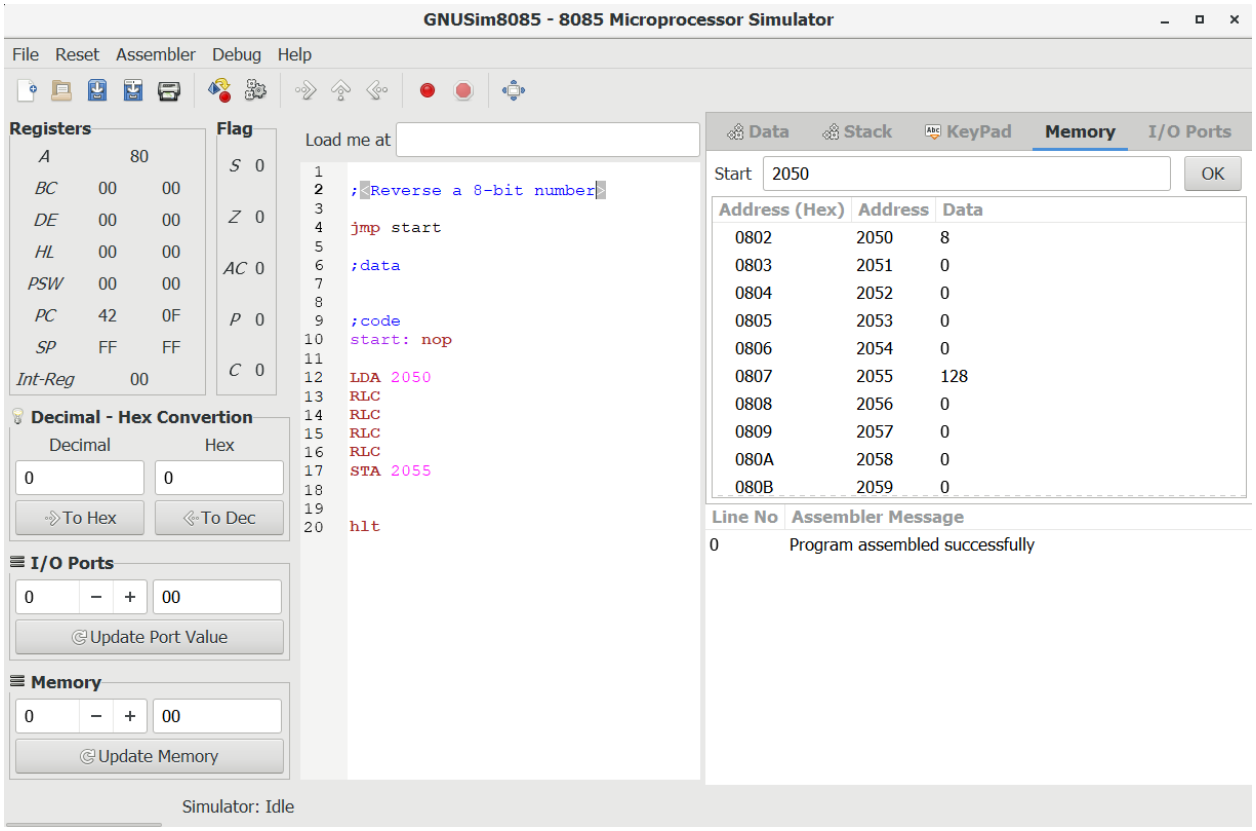
EXPERIMENT - 9

AIM: Find the factorial of a number.



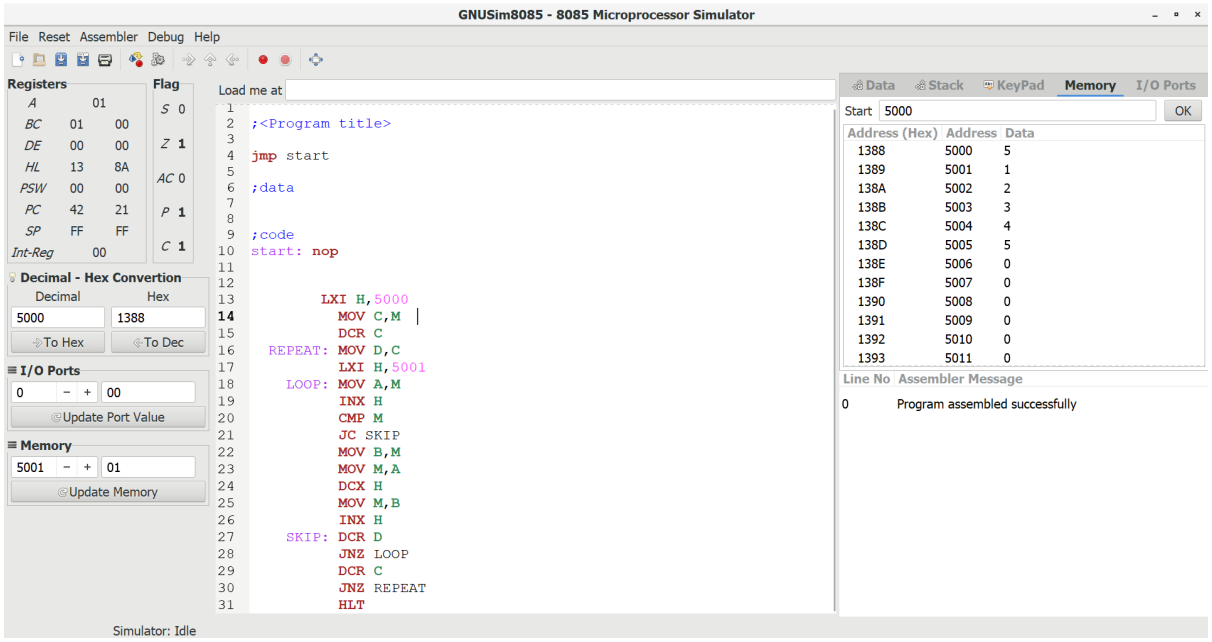
EXPERIMENT - 10

AIM: Reverse an 8-bit number in 8085.

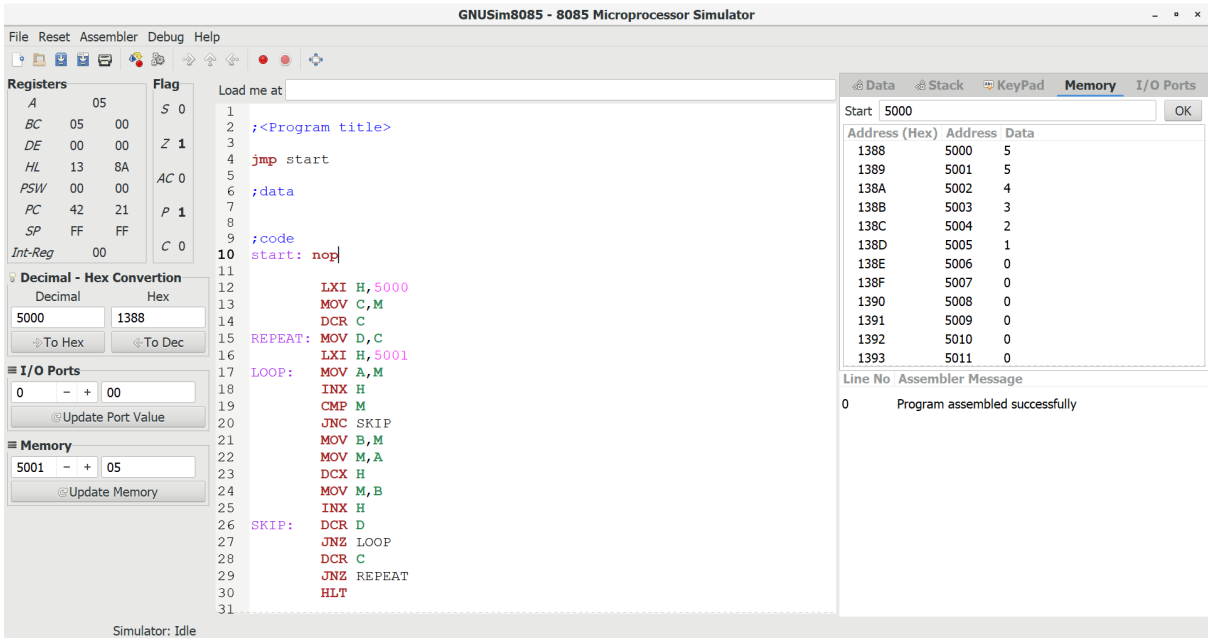


EXPERIMENT - 11

AIM: i) Sort an array in ascending order in 8085.

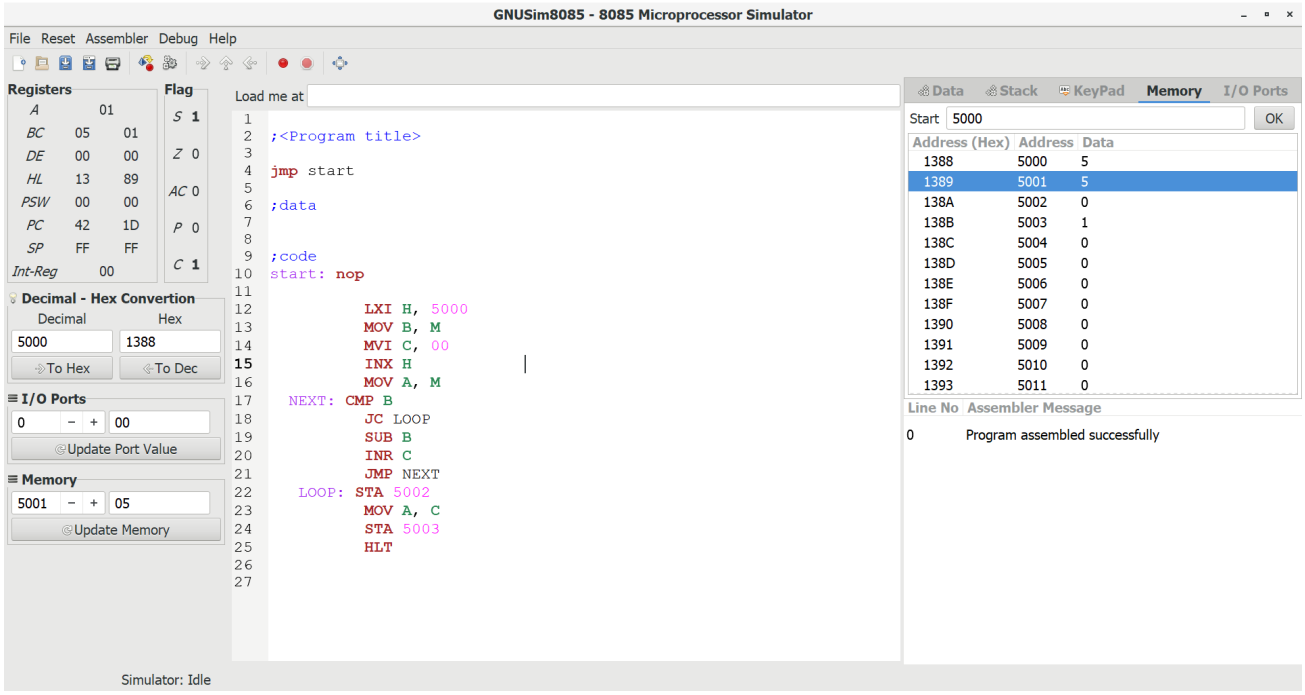


ii) Sort an array in descending order in 8085.



EXPERIMENT - 12

AIM: division of two 8-bit numbers in 8085.




```
;<Program title>

jmp start

;data


;code
start: nop

          LXI H, 5000
          MOV B, M
          MVI C, 00
          INX H
          MOV A, M
  NEXT: CMP B
          JC LOOP
          SUB B
          INR C
          JMP NEXT
   LOOP: STA 5002
          MOV A, C
          STA 5003
          HLT
```