Syeda Reeha Quasar

14114802719

4C7

Aim

Write an assembly language program to add two numbers:  
a) numbers are of 8 - bit  
b) numbers are of 16 - bit

Experiment - 2

Computer Organization and Architecture

# **EXPERIMENT – 2**

## **Aim:**

Write an assembly language program to add two numbers:

a) numbers are of 8 - bit  
b) numbers are of 16 - bit

## **Theory:**

Instructions Used-

* **MVI** (Move immediate 8-bit) - The 8-bit data is stored in the destination register or M, data memory. If the operand is a memory location, its location is specified by the contents of the HL registers. Example: MVI B, 57H or MVI M, 57H
* **ADD** (Add register or memory to accumulator) - The contents of the operand (register or memory) are added to the contents of the accumulator and the result is stored in the accumulator. If the operand is a memory location, its location is specified by the contents of the HL registers. All flags are modified to reflect the result of the addition. Example: ADD B or ADD M
* **MOV** (Copy from source to destination) - This instruction copies the contents of the source register into the destination register; the contents of the source register are not altered. If one of the operands is a memory location, its location is specified by the contents of the HL registers. Example: MOV B, C or MOV B, M
* **HLT** (Halt and enter wait state) - The CPU finishes executing the current instruction and halts any further execution. An interrupt or reset is necessary to exit from the halt state. Example: HLT
* **LXI** (Load register pair immediate) - The instruction loads 16-bit data in the register pair designated in the operand. Example: LXI H, 2034H or LXI H, XYZ
* **DAD** (Add register pair to H and L registers) - The 16-bit contents of the specified register pair are stored in the added to the contents of the HL register and the sum is HL register. The contents of the source register pair are not altered. If the result is larger than 16 bits, the CY flag is set. No other flags are affected. Example: DAD H

## **Source Codes:**

## Adding 8 – bit numbers

### Source Code

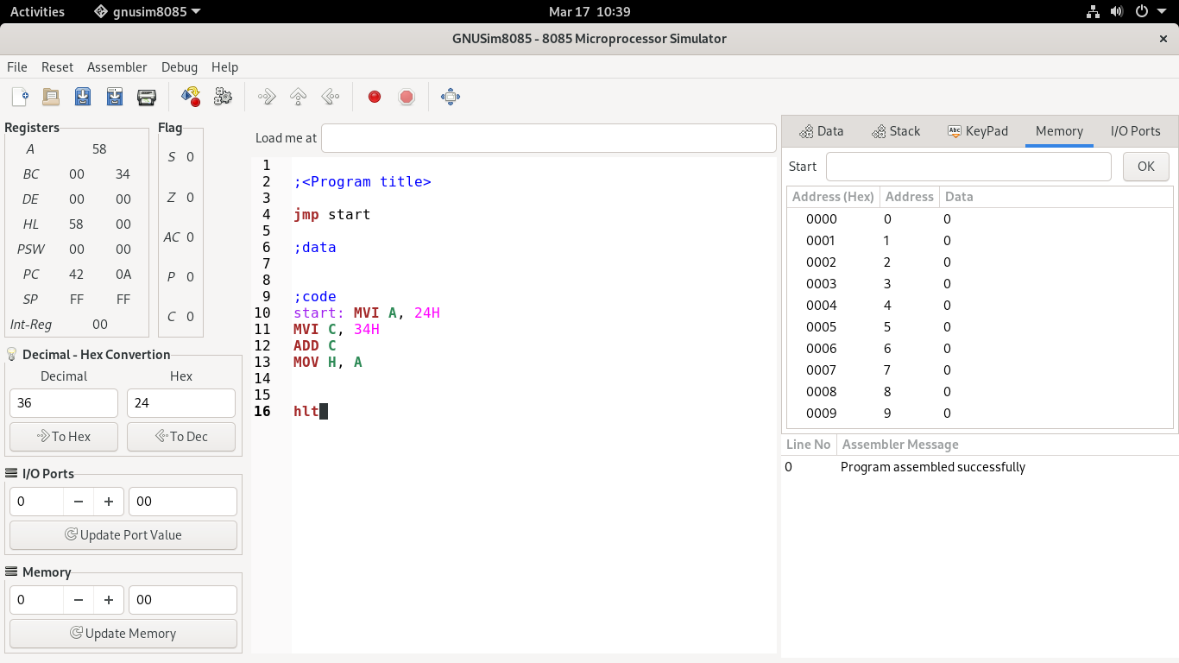
; Adding 8 – bit numbers  
  
jmp start  
  
;data  
  
  
;code  
start: MVI A, 24  
MVI C, 34  
ADD C  
MOV H, A

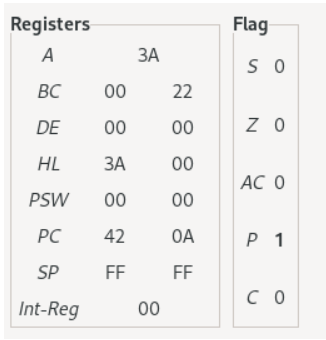
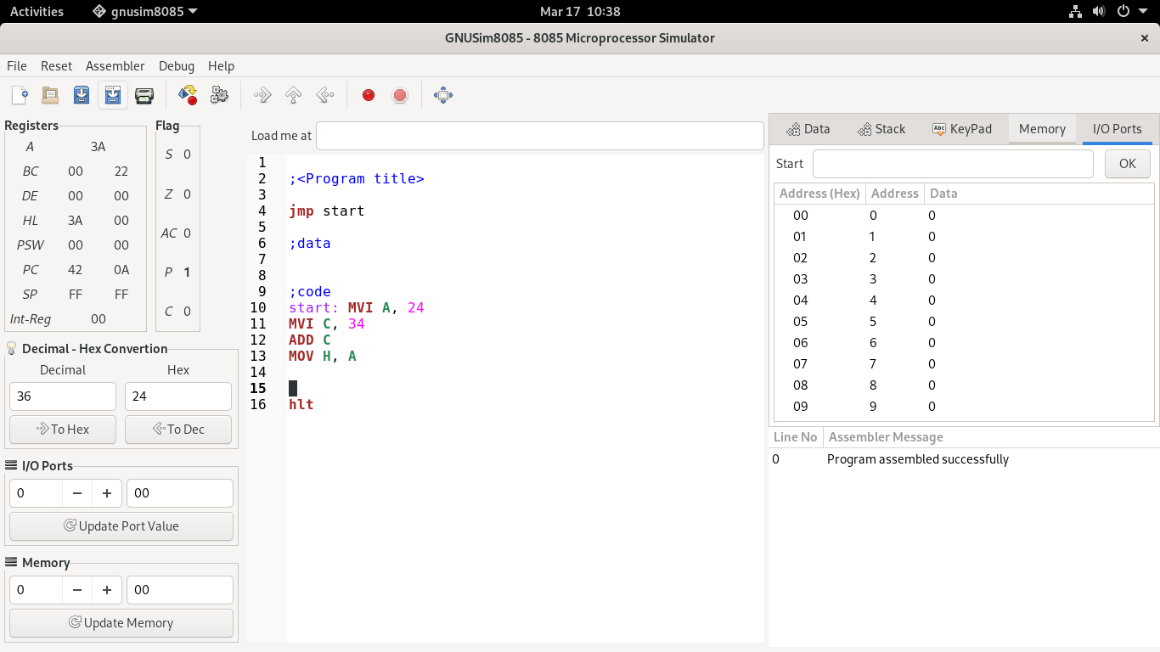
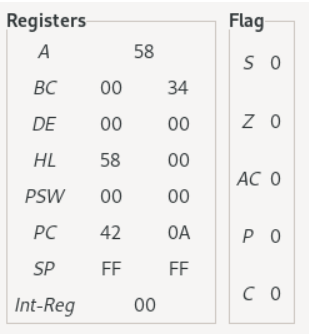
hlt

------------------------------------------------------------------------------------------------------------

; Adding 8 – bit numbers  
  
jmp start  
  
;data  
  
  
;code  
start: MVI A, 24H  
MVI C, 34H  
ADD C  
MOV H, A  
  
  
hlt

## **A) Output:**





## Adding 16 – Bit Numbers

### Source Code

;Adding 16 – bit Numbers  
  
jmp start  
  
;data

;code  
start: LXI H, 2434h  
LXI D, 3417h  
DAD D

hlt

## **B) Output**

