

# PROGRAMMING IN 8085

8 Bit Addition:

LXI H 4000H : HL points 4000H

MOV A, M : Get first operand

INX H : HL points 4001H

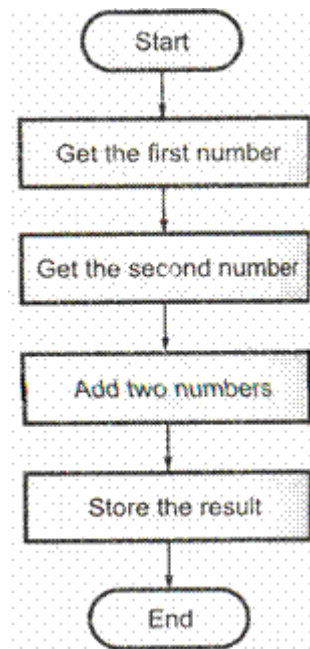
ADD M : Add second operand

INX H : HL points 4002H

MOV M, A : Store result at 4002H

HLT : Terminate program execution

Flowchart



8 Bit Subtraction:

Source program:

LXI H, 4000H : HL points 4000H

MOV A, M : Get first operand

INX H : HL points 4001H

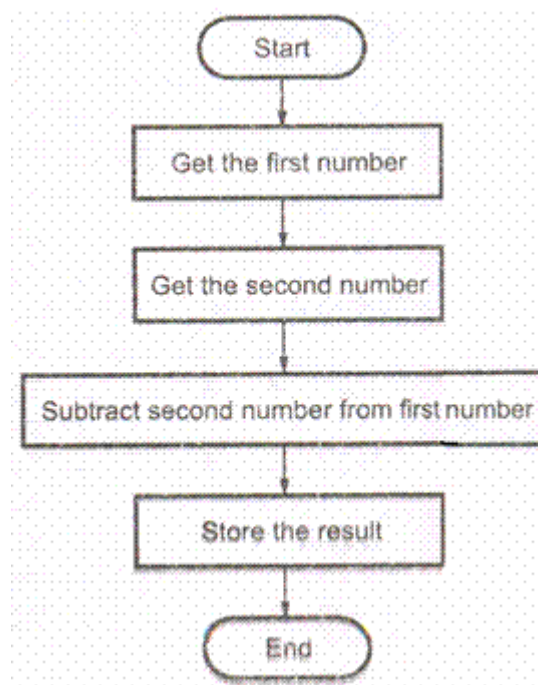
SUB M : Subtract second operand

INX H : HL points 4002H

MOV M, A : Store result at 4002H.

HLT : Terminate program execution

Flowchart

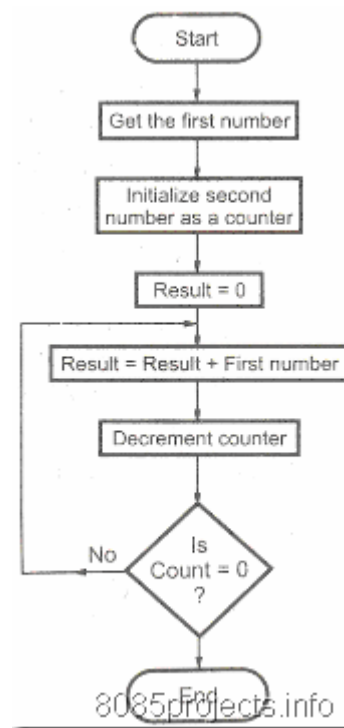


### 8Bit Multiplication

#### Source program :

- LDA 2200H
- MOV E, A
- MVI D, 00 : Get the first number in DE register pair
- LDA 2201H
- MOV C, A : Initialize counter
- LX IH, 0000 H : Result = 0
- BACK: DAD D : Result = result + first number
- DCR C : Decrement count
- JNZ BACK : If count 0 repeat
- SHLD 2300H : Store result
- HLT : Terminate program execution

#### Flowchart for program



### 8 BIT DIVISION

MVIE, 00 : Quotient = 0  
 LHLD 2200H : Get dividend

```

LDA 2300      : Get divisor
MOV B, A      : Store divisor
MVIC, 08     : Count = 8
NEXT: DAD H   : Dividend = Dividend x 2
MOV A, E
RLC
MOV E, A     : Quotient = Quotient x 2
MOV A, H
SUB B       : Is most significant byte of Dividend > divisor
JC SKIP    : No, go to Next step
MOV H, A    : Yes, subtract divisor
INR E      : and Quotient = Quotient + 1
SKIP: DCR C  : Count = Count - 1
JNZ NEXT   : Is count =0 repeat
MOV A, E
STA 2401H   : Store Quotient
Mov A, H
STA 2410H   : Store remainder
HLT        : End of program.

```

