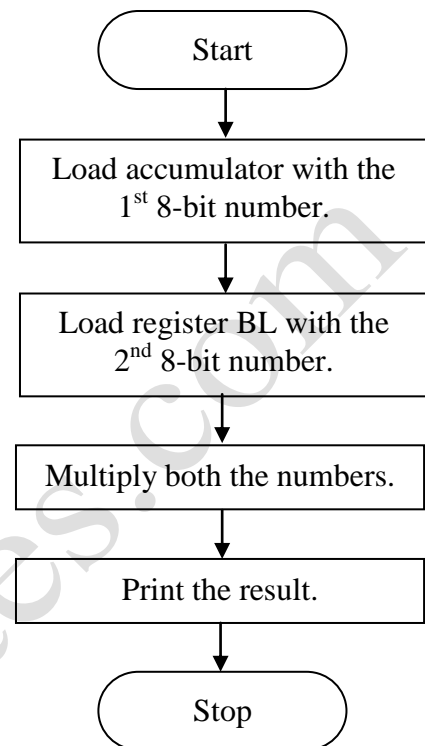


**Program 15:** Multiply two 8-bit signed numbers.**Program:**

Instructions	Comments
include "emu8086.inc"	
ORG 100h	
MOV AL, 04H	Move 1 <sup>st</sup> 8-bit number to AL.
MOV BL, FEH	Move 2 <sup>nd</sup> 8-bit number to BL.
IMUL BL	Signed multiply BL with AL and the result will be in AX.
CALL PRINT_NUM	Print the result.
RET	Return.
DEFINE_PRINT_NUM	Declare function.
END	

**Flowchart:****Explanation:**

- This program multiplies two 8-bit signed numbers.
- The program has been developed using *emu8086* emulator available at: [www.emu8086.com](http://www.emu8086.com).
- ORG 100h is a compiler directive. It tells compiler how to handle the source code.
- It tells compiler that the executable file will be loaded at the offset of 100h (256 bytes).
- The 1<sup>st</sup> 8-bit number 04H is a positive number and is moved to accumulator AL.
- The 2<sup>nd</sup> 8-bit number FEH is a negative number (-2 in decimal) and is moved to register BL.
- Then, both the numbers are multiplied.
- The multiplication of two 8-bit numbers may result into 16-bit number. So, the result is stored in AX register.
- The MSB is stored in AH and LSB is stored in AL.
- The result is printed on the screen.

**Output:****Before Execution:**

AL = 04H

BL = FEH (-2 in decimal)

**After Execution:**

AX = FFF8H (-8 in decimal)