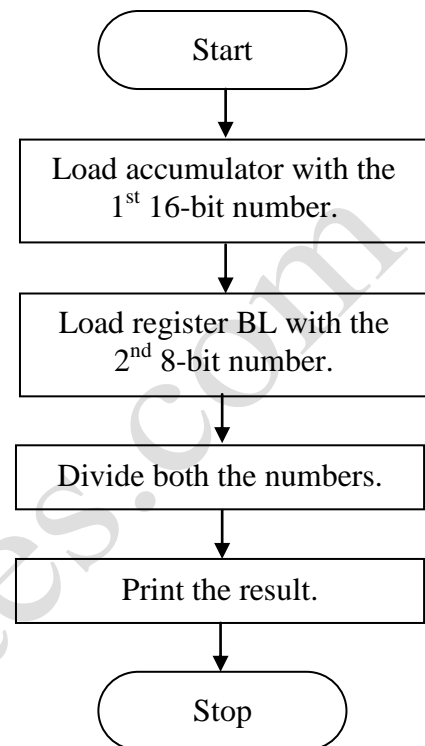


**Program 17:** Divide 16-bit unsigned number by an 8-bit unsigned number.**Program:**

Instructions	Comments
include "emu8086.inc"	
ORG 100h	
MOV AX, 0008H	Move 1 <sup>st</sup> 16-bit number to AX.
MOV BL, 02H	Move 2 <sup>nd</sup> 8-bit number to BL.
DIV BL	Divide AX with BL 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 divides a 16-bit unsigned number by an 8-bit unsigned number.
- 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> 16-bit number 0008H, i.e. dividend, is moved to accumulator AX.
- The 2<sup>nd</sup> 8-bit number 02H, i.e. divisor, is moved to register BL.
- Then, both the numbers are divided.
- The result of division is stored in AX. AL contains the quotient and AH contains the remainder.
- The result is printed on the screen.

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

AX = 0008H

BL = 02H

**After Execution:**

AL = 04H (Quotient)

AH = 00H (Remainder)