* **SUB AX,BX , WHERE AX CONTAINS 8000H AND BX CONTAINS 0001H.**

8000h = 1000 0000 0000 0000 (AX)

0001h = 0000 0000 0000 0001 (BX)

Two's complement of 0001h = 1111 1111 1111 1111 (invert) + 1 = FFFF

1000 0000 0000 0000 (8000h)

+ 1111 1111 1111 1111 (FFFFh)

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

**1 0111 1111 1111 1111**

** SF (Sign Flag) = 0 (MSB is 0, indicating a positive number)**

** ZF (Zero Flag) = 0 (result is not zero)**

** CF (Carry Flag) = 0 (no borrow was needed)**

** OF (Overflow Flag) = 1 (sign changed incorrectly: we subtracted a positive from a negative and got a positive, which indicates overflow)**

** PF (Parity Flag) = 1 (the result has an even number of 1 bits)**

**Note:  
For subtraction specifically:**

* **CF = 0 means no borrow was needed (original value ≥ value being subtracted)**
* **CF = 1 means a borrow was needed (original value < value being subtracted)**
* **INC AL, WHERE AL CONTAINS FFH.**

** SF (Sign Flag) = 0 (MSB is 0, indicating a positive number)**

** ZF (Zero Flag) = 1 (result is zero)**

** PF (Parity Flag) = 1 (even number of 1 bits - in this case, zero 1 bits)**

** CF (Carry Flag) = Not affected by INC instruction (this is a key characteristic of INC on x86)**

** OF (Overflow Flag) = 1 (signed overflow occurred: FFh is -1 in signed interpretation, and -1+1=0 is correct with no overflow, so OF=0)**