**Practical - 12**

**Aim:** Write an assembly program to perform ASCII adjust of result of addition, subtraction, multiplication and division. Also analyze the result before and after the adjustment.

**Description of instructions used:**

1. **AAA** − ASCII adjust after addition

* If the least significant four bits in AL are > 9 or if AF =1, it adds 6 to AL and 1 to AH. – Both CF and AF are set.
* In all cases, the most significant four bits in AL are cleared

1. **AAS** − ASCII adjust after subtraction

* If the least significant four bits in AL are > 9 or if AF =1, it subtracts 6 from AL and 1 from AH. – Both CF and AF are set
* In all cases, the most significant four bits in AL are cleared
* This adjustment is needed only if the result is negative

1. **AAM** − ASCII adjust after multiplication

* The AAM instruction adjusts the result of a MUL instruction.

∗ Multiplication should not be performed on ASCII

» Can be done on unpacked BCD.

* The AAM instruction works as follows

∗ AL is divided by 10

∗ Quotient is stored in AH

∗ Remainder in AL

* AAM does not work with IMUL instruction

1. **AAD** − ASCII adjust before division

* The ADD instruction adjusts the numerator in AX before dividing two unpacked decimal numbers

∗ The denominator is a single unpacked byte

* The ADD instruction works as follows

∗ Multiplies AH by 10 and adds it to AL and sets AH to 0

∗ Example:

» If AX is 0207H before AAD

» AX is changed to 001BH after AAD

* AAD instruction reverses the changes done by AAM
* ADD converts the unpacked BCD number in AX to binary form so that div can be used

**Code:**

1. **AAA:**

SUB AH, AH ; clear AH

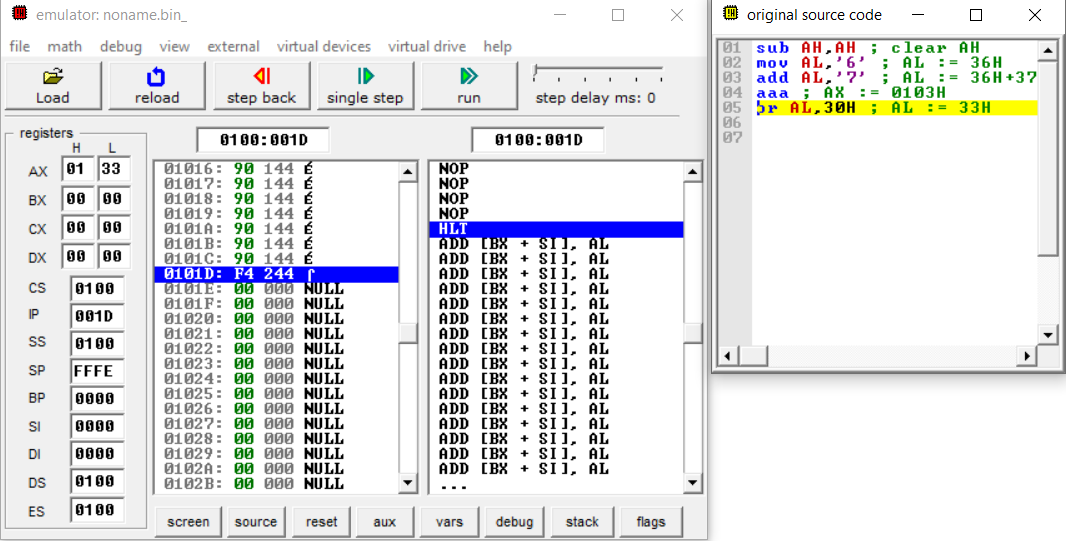
MOV AL,'6' ; AL: 36H

ADD AL,'7' ; AL: 36H+37H = 6DH

AAA ; AX: 0103H

OR AL,30H ; AL: 33H

**Output:**



1. **AAS:**
   1. **POSITIVE RESULT:**

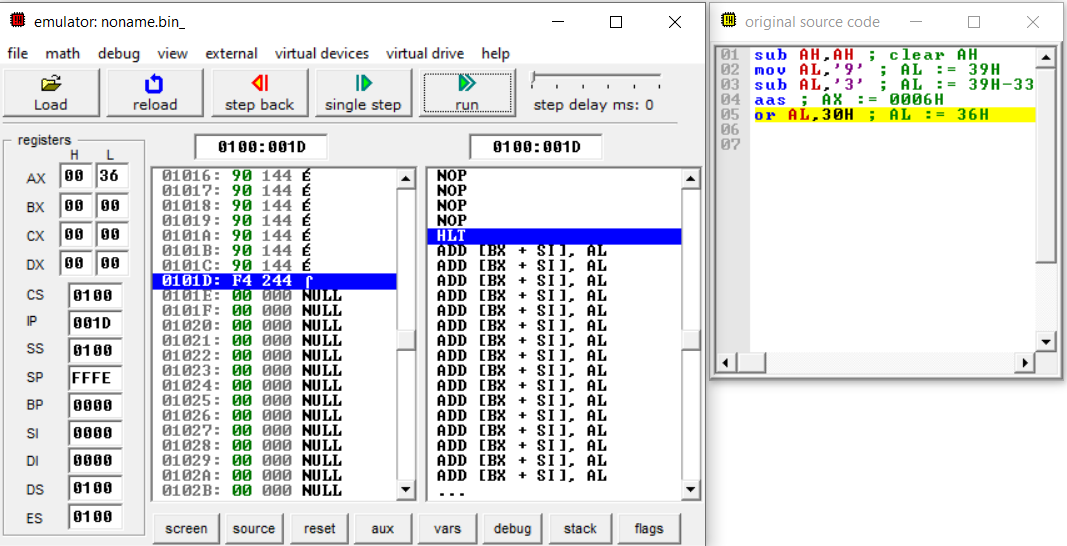
SUB AH, AH ; clear AH

MOV AL,'9' ; AL: 39H

SUB AL,'3' ; AL: 39H-33H = 6H

AAS ; AX: 0006H

OR AL,30H ; AL: 36H

**Output:**  

* 1. **NEGATIVE RESULT:**

SUB AH, AH ; clear AH

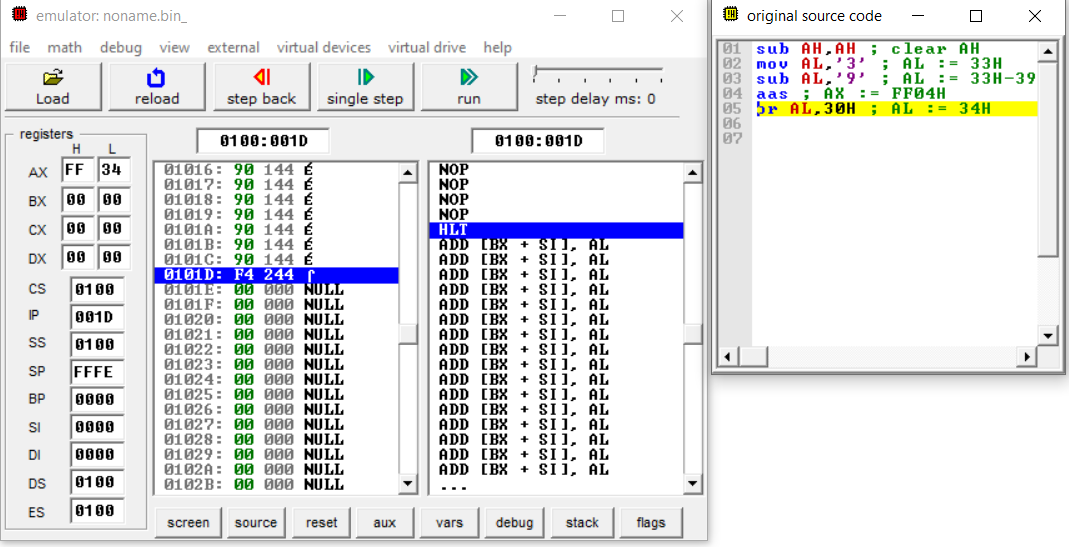
MOV AL,'3' ; AL: 33H

SUB AL,'9' ; AL: 33H-39H = FAH

AAS ; AX: FF04H

OR AL,30H ; AL: 34H

**Output:**



1. **AAM:**
   1. **Example 1:**

MOV AL,3 ; multiplier in unpacked BCD form

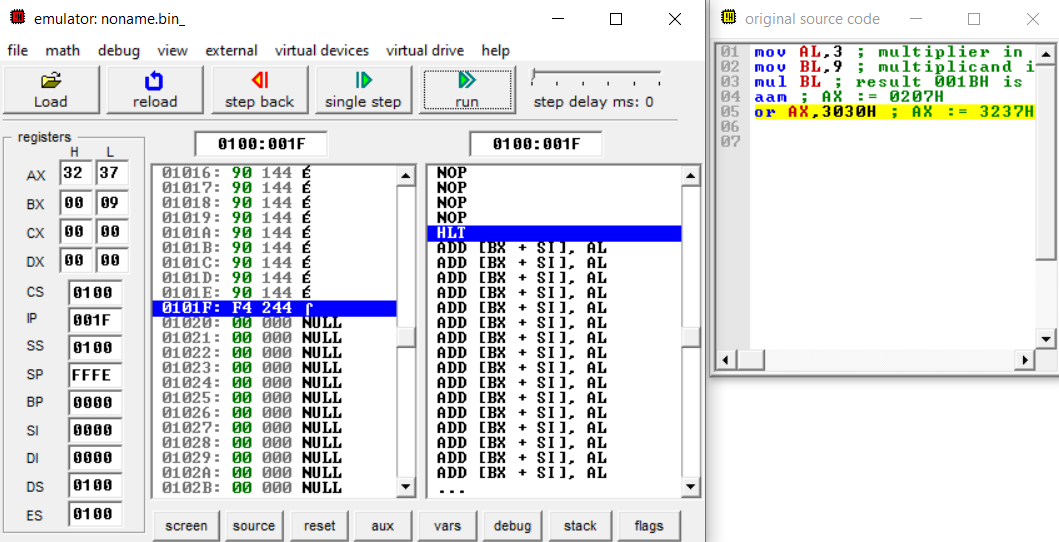
MOV BL,9 ; multiplicand in unpacked BCD form

MUL BL ; result 001BH is in AX

AAM ; AX: 0207H

OR AX,3030H ; AX: 3237H

**Output:**



* 1. **Example 2:**

MOV AL,'3' ; multiplier in ASCII

MOV BL,'9' ; multiplicand in ASCII

AND AL,0FH ; multiplier in unpacked BCD form

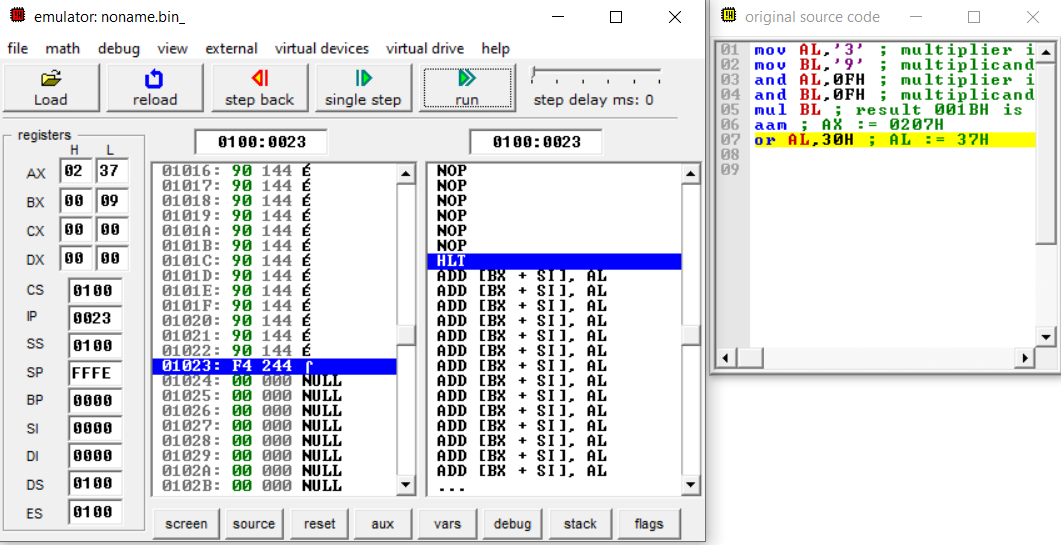
AND BL,0FH ; multiplicand in unpacked BCD form

MUL BL ; result 001BH is in AX

AAM ; AX: 0207H

OR AL,30H ; AL: 37H

**Output:**



1. **AAD:**

MOV AX,0207H ; dividend in unpacked BCD form

MOV BL,05H ; divisor in unpacked BCD form

AAD ; AX: 001BH

DIV BL ; AX: 0205H

**Output:**

