Name: Muhammad Ahmad

Reg: L1F22BSCS0634

**LAB - 02**

**Practice Exercise**

**Task-1**

Which of the following instructions is invalid? Give reasons.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr#** | **Instructions** | **Valid/ Invalid** | **Reasons** |
| 1 | MOV AX, 27 | VALID | Because we store an immediate value in AX and both have same size. |
| 2 | MOV AL, 97Fh | INVALID | Because the size of both operands are not same. |
| 3 | MOV SI, 9516 | VALID | The size of segment register is 16-bits, so both have same size. |
| 4 | MOV DS, BX | VALID | Both are 16-bits registers, they have same size. |
| 5 | MOV BX, CS | VALID | Both are 16-bits registers, they have same size. |
| 6 | MOV AX, 23FB9h | INVALID | The size of source operand is more than 16-bits. |
| 7 | MOV DS, BH | INVALID | The source operand size is 8-bits and DS is 16-bits, So the size is different. |
| 8 | MOV DS, 9BF2 | INVALID | If the destination operand is a segment register, the source operand cannot be an immediate value. |
| 9 | MOV CS, 3490 | INVALID | The destination operand cannot be a CS |
| 10 | MOV DS, ES | INVALID | Both operands cannot be segment registers |
| 11 | MOV ES, BX | VALID | Both have same size, which is 16-bits. |

**Task-2: -**

Write a program in assembly language that calculates the square of six by adding six to the accumulator six times.

**Code:**

.model small

.stack 100h

.data

.code

mov ax, 6

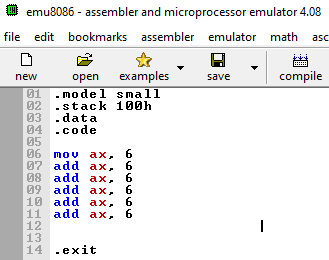
add ax, 6

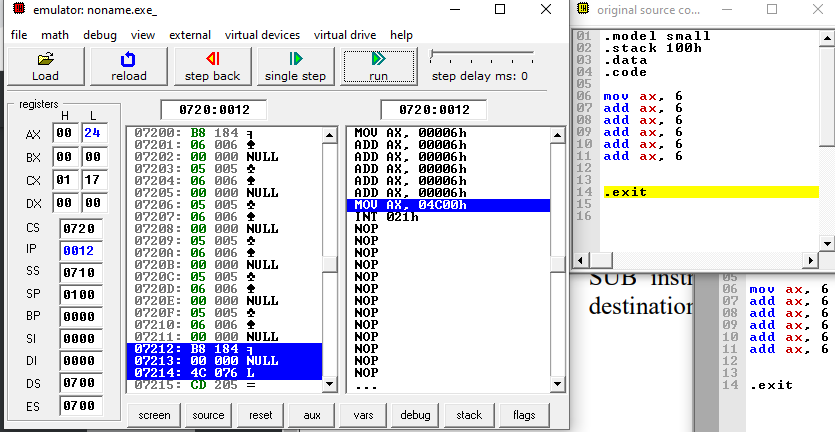
add ax, 6

add ax, 6

add ax, 6

add ax, 6

.exit



**Task-3: -**

Write a program to solve the following equation.

DX = AX + BX – CX + DX

Initialize the AX, BX, CX, and DX registers with 0100h, 55ABh, 0A11h and 0001h values, respectively.

**Code:**

.model small

.stack 100h

.data

.code

mov ax, 0100h

mov bx, 55ABh

mov cx, 0A11h

mov dx, 0001h

add dx, ax

add dx, bx

sub dx, cx

.exit

