**Computer Organization and Assembly Language**

|  |  |
| --- | --- |
| **Lab 06** | |
| **Topic** | * Add,sub * Memory addressing * Flags * Logical operations * Loops * Branching (JUMPS) |

**Problem #1:**

Write an assembly language program that calculate the decimal value of the word value and stores the individual digits in different variables. Consider the number is signed.

Hint: Think how binary to decimal can be calculated. Use repeated addition for multiplication purpose.

For example:

1. Let 0XABAB

Decimal value will be

-21589

Solution:

Digit\_unit: db 9

Digit\_ten: db 8

Digit\_hundred: db 5

Digit\_thousand: db 1

Digit\_tenthousand: db 2

1. Let 0X78CA

Decimal value will be

30922

Digit\_unit: db 2

Digit\_ten: db 2

Digit\_hundred: db 9

Digit\_thousand: db 0

Digit\_tenthousand:3

**Problem #2:**

Write an assembly language program that will add the odd value digits of all the numbers within each element of a word array.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Index | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| Value | 0xABCD | 0x1234 | 0X5678 | 0x9876 | 0x5432 | 0X7766 | 0XACE3 | 0x2536 |

B+D+1+3+5+7+9+7+5+3+7+7+3+5+3

Sol:

Sum: dw 63

**Problem # 3:**

Write an assembly language program to clear the nth bit of a number of word size. Code should be generic.

1. Binary of F37E is : 1111 0011 0111 1110

For-example

Number: dw 0xF37E

Position db 5 (5th bit starting from LSB as 0th bit)

After Execution

Number: dw 0xF35E

1. Binary of F37E is : 1111 0011 0111 1110

For-example

Number: dw 0xF37E

Position db 7 (7th bit starting from LSB as 0th bit)

After Execution

Number: dw 0xF37E