**Computer Organization and Assembly Language (CS-3G Fall 2020)**

**ASSIGNMENT 1**

**Due Date: Monday, 26th October 2020**

**Q1.** List the seven addressing modes available in the 8088 architecture. Giveexample of each in terms of instruction/code.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Q2.** Take the 4 digits of your Roll no. Replace all the '1s' (if any) with A and all the'0s' (if any) with B. For Example, 1182 becomes AA82, 1027 becomes AB27, 4208 becomes 42B8 etc

Make four different combinations by left shifting/rotating digits of your Roll no. as follows:

AA82: AA82, A82A, 82AA, 2AA8 and consider this array1

*Example* array1: dw AA82h, A82Ah, 82AAh, 2AA8h

Now make four 2digit combinations by taking

1. 1st and 2nd digit
2. 3rd and 4th digit
3. 1st and 4th digit
4. 2nd and 3rd digit and consider this array2

*Example*array2: db AAh, 82h, A2h, A8h

Consider another array

array3: dw 0000h, 0000h, 0000h, 0000h

* Write a program in assembly language to subtract each element of array2 from its corresponding element in array1 and store in array3
* You must solve the same problem with at least 3 different addressing modes
* Note and write down values of Carry, Zero, Sign and OverFlow Flags upon each subtraction
* Write down the effective addresses of each element of array3 in all 3 programs

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Q3.** Given an array of words with last element as -1, write a code to delete theeven numbers from array. Example case is given below. (you can assume that -1

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| will not be in the array, only at end) | | | | |  |  |  |  |  |  |
| Array: | dw | 10, | 13, 96, | 16, | 18, | 51, | 88, | 45, 2, | 4, 3, | -1 |
| After your code finishes executing Array should be as follow. | | | | | | | | |  |  |
| 13, | 51, | 45, | -1, | -1, | -1, | -1, | -1, | -1, | -1, | -1, |

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Q4.** Given that: CS= 5645h, DS= 1000h, ES= 6783h, SS= FFFFh, BX= 4567h, SI=FFFFh, DI= 2000h, BP= 4700h, SP = 4500h

Write the physical address of the memory locations. (Show your working)

|  |  |
| --- | --- |
|  | Memory Location Physical Address in hex |
|  |  |
| I | [CS:bx + di] 5 C9B7 |

1. [bp + si + 10] 04701

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Q5.** Encryption and Decryption algorithms are very commonly used in NetworkSecurity. Your task is to write code for Encryption algorithm written below

Encryption

There is a 32 bit PlainText and a 32 bit Encryption Key. Use the key to perform 4 rounds to encrypt Text.

In 1st round, Take the key,

1- complement every 2nd bit, starting from 1st bit

2- Rotate Left

3- Add key to the plaintext

In 2nd round, take the updated key

1- Complement every 2nd bit starting from 2nd bit

2- Rotate Right

3- Add to the text obtained from round 1

. In 3rd round, Take the updated key from round 2, 1- complement every 2nd bit, starting from 1st bit 2- Rotate Left

3- Add to text obtained from round 2

In 4th round, take updated key from round 3

1- Complement every 2nd bit starting from 2nd bit

2- Rotate Right

3- Add to text obtained from round 3

Example

Plaintext: 15 D3 C2 57

Key: 23 CD E6 89

*Round 1*

KEY: 0010 0011 1100 1101 1110 0110 1000 1001

Bit cpl: 1000 1001 0110 0111 0100 1100 0010 0011

Rotate left: 0001 0010 1100 1110 1001 1000 0100 0111

|  |  |  |  |
| --- | --- | --- | --- |
| Resultant key in hex | | | 12 CE 98 47 |
| Result after addition: | | | 28 A2 5A 9E |
| *Round 2* | |  |  |
| KEY: | 0001 0010 1100 1110 1001 1000 0100 0111 | | |
| Bit cpl: | | 0100 0111 1001 1011 1100 1101 0001 0010 | |
| Rotate Right: 0010 0011 1100 1101 1110 0110 1000 1001 | | | |
| Resultant key in hex | | | 23CDE689 |
| Result after addition: | | | 4C 70 41 27 |

*Round 3*

KEY: 0010 0011 1100 1101 1110 0110 1000 1001

Bit cpl: 1000 1001 0110 0111 0100 1100 0010 0011

Rotate left: 0001 0010 1100 1110 1001 1000 0100 0111

Resultant key in hex 12 CE 98 47

Result after addition: 5F 3E D9 6E

|  |  |  |  |
| --- | --- | --- | --- |
| *Round 4* | |  |  |
| KEY: | 0001 0010 1100 1110 1001 1000 0100 0111 | | |
| Bit cpl: | | 0100 0111 1001 1011 1100 1101 0001 0010 | |
| Rotate Right: 0010 0011 1100 1101 1110 0110 1000 1001 | | | |
| Resultant key in hex | | | 23 CD E6 89 |
| Result after addition: | | | 83 0C BF F7 |

Note: You will begin with Key = 23 CD E6 89 and PlainText = 15 D3 C2 57 and after four encryption rounds your result will be Key = 23 CD E6 89 and Encrypted Text = 83 0C BF F7

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

You are required to submit all .asm files on Xeon (by 5pm on due date) and also submit hard copy of the assignment (print out the codes that you write) in class.

***Good luck!***