**Gebze Teknik Üniversitesi**

**Bilgisayar Mühendisliği Bölümü**

**BİL101 – Bilgisayara Giriş**

**Ödev #2**

**Ömer Faruk BİTİKÇİOĞLU**

**161044010**

**1-)** **a)** 8A9

8 – 8, A – 10, 9 – 9

8 = 20.0 + 21.0 + 22.0 + 23.1 – (**1000**)2

A = 20.0 + 21.1 + 22.0 + 23.1 – (**1010**)2

9 = 20.1 + 21.0 + 22.0 + 23.1 – (**1001**)2

* 100010101001

**b-)** EF3

E – 14, F – 15, 3 – 3

E = 20.0 + 21.1 + 22.1 + 23.1 – (**1110**)2

F = 20.1 + 21.1 + 22.1 + 23.1 – (**1111**)2

3 = 20.1 + 21.1 + 22.0 + 23.0 – (**0011**)2

* 111011110011

**c-)** 0001 1110 0001

0001 = 20.1 + 21.0 + 22.0 + 23.0 = **1**

1110 = 20.0 + 21.1 + 22.1 + 23.1 = 14 – **E**

0001 = 20.1 + 21.0 + 22.0 + 23.0 = **1**

* 1E1

**d-)** 1111 1110 1101 1011

1111 = 20.1 + 21.1 + 22.1 + 23.1 = 15 – **F**

1110 = 20.0 + 21.1 + 22.1 + 23.1 = 14 – **E**

1101 = 20.1 + 21.0 + 22.1 + 23.1 = 13 – **D**

1011 = 20.1 + 21.1 + 22.0 + 23.1 = 11 – **B**

* FEDB

**2-)** 436F6D7075746572

4 = 20.0 + 21.0 + 22.1 + 23.0 – **0100**

3 = 20.1 + 21.1 + 22.0 + 23.0 – **0011**

* **01000011 = C**

6 = 20.0 + 21.1 + 22.1 + 23.1 – **0110**

F = 20.0 + 21.1 + 22.1 + 23.1 – **1111**

* **01101111 = o**

6 = 20.0 + 21.1 + 22.1 + 23.1 – **0110**

D = 20.0 + 21.1 + 22.1 + 23.1 – **1101**

* **01101101 = m**

7= 20.0 + 21.1 + 22.1 + 23.1 – **0111**

0 = 20.0 + 21.1 + 22.1 + 23.1 – **0000**

* **01110000 = p**

7 = 20.0 + 21.1 + 22.1 + 23.1 – **0111**

5 = 20.0 + 21.1 + 22.1 + 23.1 – **0101**

* **01110101 = u**

7 = 20.0 + 21.1 + 22.1 + 23.1 – **0111**

4 = 20.0 + 21.1 + 22.1 + 23.1 – **0100**

* **01110100 = t**

6 = 20.0 + 21.1 + 22.1 + 23.1 – **0110**

5 = 20.0 + 21.1 + 22.1 + 23.1 – **0101**

* **01100101 = e**

7 = 20.0 + 21.1 + 22.1 + 23.1 – **0111**

2 = 20.0 + 21.1 + 22.1 + 23.1 – **0010**

* **01110010 = r**

**3-)** 5 bitlik gösterimde 25 = 32 sayı gösterilebilir. Bunlar [15,-16] kapalı aralığında değer alabilir. Eğer yapacağımız işlemlerin sonucu bu aralıkta değilse *overflow* olmuştur.

**a)** 5 = 00101

1 = 00001

-1 = 11111

00101 + 11111 = 00100

00100 = 20.0 + 21.0 + 22.1 + 23.0 + 24.0 = 4 – bu sayı tanımlı aralığın üyesi olduğundan *overflow* **olmamıştır.**

Aynı işlem onluk tabanda : 5-1 = 4

**b)** 5 = 00101

11 = 01011

-11 = 10101

00101 + 10101 = 11010

11010 = 20.0 + 21.1 + 22.0 + 23.1 + 24.1 = 26 – bu sayı tanımlı aralığın üyesi olmadığından bu işlemde *overflow* **olmuştur.**

Aynı işlem onluk tabanda : 5-11 = -6

**4-)** 01001011 *AND* 10101011 = 00001011

01001011 *OR* 10101011 = 11101011

01001011 *XOR* 10101011 = 11100000

**5-) a)** **7**123

Op-code: 7 -> OR R1, R2, R3

R2 ve R3’e girilen 2 ve 3 değerlerine OR işlemi yaptırıp sonucu R1’e yani 1’in yerine yazar.

**b) 2**BCD

Op-code: 2 -> LOAD R, XY

XY’nin bit şablonunu Register’a yükler. Burada CD’nin bit şablonu B’ye atanır.

**6-)** LOAD A3 , F0

LOAD A4 , 0F

AND R1,A0 ,A3

AND R2, A1, A4

ADD A2,R1 R2