

Gebze Teknik Üniversitesi
Bilgisayar Mühendisliği Bölümü
BİL101 – Bilgisayara Giriş
Ödev #2

Ömer Faruk BİTİKÇİOĞLU

161044010

1-) a) 8A9

8 – 8, A – 10, 9 – 9

$$8 = 2^0 \cdot 0 + 2^1 \cdot 0 + 2^2 \cdot 0 + 2^3 \cdot 1 = (\mathbf{1000})_2$$

$$A = 2^0 \cdot 0 + 2^1 \cdot 1 + 2^2 \cdot 0 + 2^3 \cdot 1 = (\mathbf{1010})_2$$

$$9 = 2^0 \cdot 1 + 2^1 \cdot 0 + 2^2 \cdot 0 + 2^3 \cdot 1 = (\mathbf{1001})_2$$

➔ 100010101001

b-) EF3

E – 14, F – 15, 3 – 3

$$E = 2^0 \cdot 0 + 2^1 \cdot 1 + 2^2 \cdot 1 + 2^3 \cdot 1 = (\mathbf{1110})_2$$

$$F = 2^0 \cdot 1 + 2^1 \cdot 1 + 2^2 \cdot 1 + 2^3 \cdot 1 = (\mathbf{1111})_2$$

$$3 = 2^0 \cdot 1 + 2^1 \cdot 1 + 2^2 \cdot 0 + 2^3 \cdot 0 = (\mathbf{0011})_2$$

➔ 111011110011

c-) 0001 1110 0001

$$0001 = 2^0 \cdot 1 + 2^1 \cdot 0 + 2^2 \cdot 0 + 2^3 \cdot 0 = \mathbf{1}$$

$$1110 = 2^0 \cdot 0 + 2^1 \cdot 1 + 2^2 \cdot 1 + 2^3 \cdot 1 = 14 - \mathbf{E}$$

$$0001 = 2^0 \cdot 1 + 2^1 \cdot 0 + 2^2 \cdot 0 + 2^3 \cdot 0 = \mathbf{1}$$

➔ 1E1

d-) 1111 1110 1101 1011

$$1111 = 2^0 \cdot 1 + 2^1 \cdot 1 + 2^2 \cdot 1 + 2^3 \cdot 1 = 15 - \mathbf{F}$$

$$1110 = 2^0 \cdot 0 + 2^1 \cdot 1 + 2^2 \cdot 1 + 2^3 \cdot 1 = 14 - \mathbf{E}$$

$$1101 = 2^0 \cdot 1 + 2^1 \cdot 0 + 2^2 \cdot 1 + 2^3 \cdot 1 = 13 - \mathbf{D}$$

$$1011 = 2^0 \cdot 1 + 2^1 \cdot 1 + 2^2 \cdot 0 + 2^3 \cdot 1 = 11 - \mathbf{B}$$

➔ FEDB

2-) 436F6D7075746572

$$4 = 2^0 \cdot 0 + 2^1 \cdot 0 + 2^2 \cdot 1 + 2^3 \cdot 0 = \mathbf{0100}$$

$$3 = 2^0 \cdot 1 + 2^1 \cdot 1 + 2^2 \cdot 0 + 2^3 \cdot 0 = \mathbf{0011}$$

➔ 01000011 = C

$$6 = 2^0 \cdot 0 + 2^1 \cdot 1 + 2^2 \cdot 1 + 2^3 \cdot 1 = \mathbf{0110}$$

$$F = 2^0 \cdot 0 + 2^1 \cdot 1 + 2^2 \cdot 1 + 2^3 \cdot 1 = \mathbf{1111}$$

➔ 01101111 = o

$$6 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0110}$$

$$D = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{1101}$$

$$\rightarrow \mathbf{01101101 = m}$$

$$7 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0111}$$

$$0 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0000}$$

$$\rightarrow \mathbf{01110000 = p}$$

$$7 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0111}$$

$$5 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0101}$$

$$\rightarrow \mathbf{01110101 = u}$$

$$7 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0111}$$

$$4 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0100}$$

$$\rightarrow \mathbf{01110100 = t}$$

$$6 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0110}$$

$$5 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0101}$$

$$\rightarrow \mathbf{01100101 = e}$$

$$7 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0111}$$

$$2 = 2^0.0 + 2^1.1 + 2^2.1 + 2^3.1 - \mathbf{0010}$$

$$\rightarrow \mathbf{01110010 = r}$$

3-) 5 bitlik gösterimde $2^5 = 32$ sayı gösterilebilir. Bunlar [15,-16] kapalı aralığında değer alabilir. Eğer yapacağımız işlemlerin sonucu bu aralıkta değilse *overflow* olmuştur.

$$\mathbf{a) 5 = 00101}$$

$$1 = 00001$$

$$-1 = 11111$$

$$00101 + 11111 = 00100$$

$$00100 = 2^0.0 + 2^1.0 + 2^2.1 + 2^3.0 + 2^4.0 = 4 - \text{bu sayı tanımlı aralığın üyesi olduğundan } \textit{overflow} \text{ olmamıştır.}$$

$$\text{Aynı işlem onluk tabanda : } 5-1 = 4$$

b) 5 = 00101

11 = 01011

-11 = 10101

00101 + 10101 = 11010

$11010 = 2^0 \cdot 0 + 2^1 \cdot 1 + 2^2 \cdot 0 + 2^3 \cdot 1 + 2^4 \cdot 1 = 26$ – 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) 7123

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) 2BCD

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