



Fenerbahçe Üniversitesi

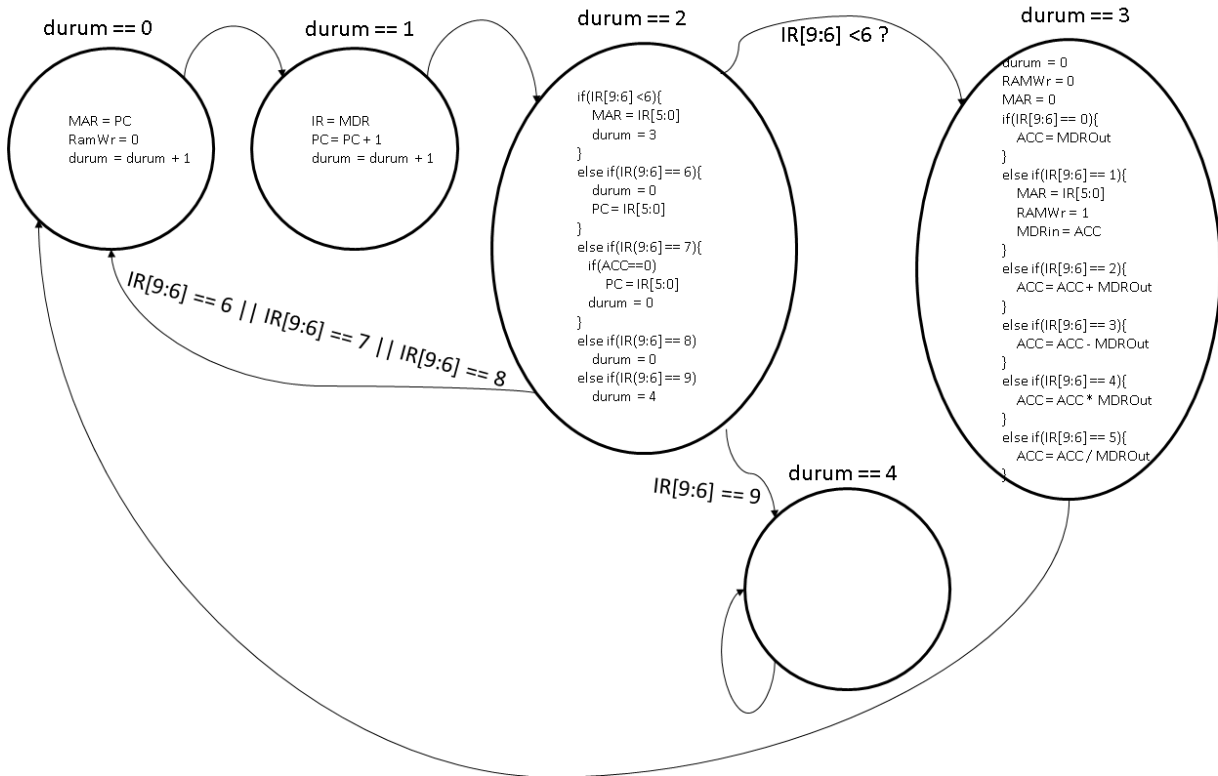
BLM 101 – Bilgisayar Mühendisliğine Giriş

LAB 7: FB-CPU Donanım ve Makine Dili Tasarımı

Önemli Not: Aşamaları tamamladıkça, dersin hocası veya asistanı yanınıza çağırarak, tamamladığınız aşamayı gösterdikten sonra diğer aşamaya geçiniz.

LAB Hakkında:

FB-CPU'nun tasarım ve makine dili ile kod geliştirmeye başlanacaktır. FB-CPU'nun desteklediği komut seti tanıtılacaktır. Durum makinesi ile 4 durumla ifade edilmektedir. Verilen başlangıç tasarımında durum 0 ve durum 1 verilen durum diyagramına göre gerçekleşmiştir. Durum diyagramı Şekil 1'de verilmiştir. Durum 2'nin tasarımı yapılacaktır.



Şekil 1. FB-CPU Durum Diyagramı

Başlangıç tasarımı: http://levent.tc/courses/blm101/proje/BLM101_proje_fbCPUBaslangic.circ

Kullanılacak simülatör: <http://levent.tc/courses/blm101/proje/logisim-evolution.jar>

LAB'ın aşamaları ve puanlar:

1- Aşağıda verilen ifadenin donanım karşılığını çiziniz (0 Puan):

A sayısı 5 bit, Y sayısı ise 8 bittir.

```
if(A == 5)
    Y = 15;
else
    Y = 20;
```

2- Şekil 2'de verilen durum diyagramının donanım karşılığını çiziniz (0 Puan):

Durum isminde bir bitlik saklayıcı bulunmalıdır. Durum saklayıcısı 0 konumunda iken, 1 bitlik push isminde bir sinyal geliyorsa halen 0 konumunda kalmalıdır. Ancak 1 bitlik coin sinyali aktif olduysa, durum 1'e geçmelidir. Durum 1'de ise push sinyali varsa durum 0'a dönmeli, coin sinyali varsa durum 1'de kalmalıdır. Aşağıda programlama dilinde ifadesi bulunmaktadır.

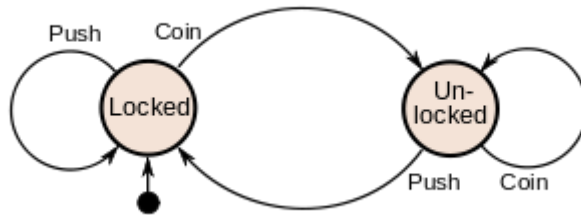
Durum 0 için

```
if(coin == 1)
    durumNext0 = 1;
else
    durumNext0 = 0;
```

Durum 1 için

```
if(push == 1)
    durumNext1 = 1;
else
    durumNext1 = 0;
```

Durum sinyalinin üzerine durumNext sinyallerinin atanması için bir MUX'a ihtiyaç bulunmaktadır. Bu MUX'un select girişine, durum sinyali, girişlerine ise durumNext0 ve durumNext1 bağlandığında devre tamamlanır.



Şekil 2. Durum Makinası

3- İlk 6 operasyon olan, RAM'e adresin verilmesi gerekli olan operasyonların RAM'den veri isteme bölümünü yapınız (0 Puan).

RAM'e 0000_000011 (LOD 3 komutu), Hex = 3, yazarak, MAR deęişkeninde 3 olduęu ve durum deęişkeninin 3'e geçtięini gözlemleyiniz.

```
if(IR[9:6] < 6){  
    MAR = IR[5:0]  
    durum = 3;  
}
```

4., 5., 6. Tasarımları test etmek için RAM'e komut yerleřtirmek gerekmektedir. RAM'in 0 nolu lokasyonuna,

4. tasarım için, 0110_000101 (JMP 5), Hex = 185, yazarak, durum'un 0, PC'nin 5'e atandıęını gözlemleyiniz.

5. tasarım için, 0111_000111 (JMZ 7), Hex = 1C7, yazarak, PC'nin 9'a atladıęını ve durum'un 0 olduęunu gözlemleyiniz (PC, ACC bařlangıçta 0 ise, 9 atanacaktır, deęilse deęiřtirilmez).

6. tasarım için, 1000_000000 (NOP), Hex = 200, 1001_000000 (HLT), Hex = 240, deęeri yazıp, NOP ve HLT iřlemlerini yaptıęını test ediniz.

4- JMP (Jump) Operasyonunu tamamlayınız. Operasyonu bir önceki bölümde istenen donanımın üzerine kurunuz (40 Puan).

```
if(IR[9:6] < 6){  
    MAR = IR[5:0]  
    durum = 3  
}  
else if(IR[9:6] == 6){  
    durum = 0  
    PC = IR[5:0]  
}
```

5- JMZ (Jump if Zero) Operasyonunu tamamlayınız. Operasyonu bir önceki bölümde istenen donanımın üzerine kurunuz (40 Puan).

```
if(IR[9:6] < 6){  
    MAR = IR[5:0]  
    durum = 3  
}  
else if(IR[9:6] == 6){  
    durum = 0  
    PC = IR[5:0]  
}  
else if(IR[9:6] == 7){  
    if(ACC==0)  
        PC = IR[5:0]  
    durum = 0  
}
```

6- NOP ve HLT operasyonlarını tamamlayınız. Operasyonu bir önceki bölümde istenen donanımın üzerine kurunuz (20 Puan).

```
if(IR[9:6] < 6){
    MAR = IR[5:0]
    durum = 3
}
else if(IR[9:6] == 6){
    durum = 0
    PC = IR[5:0]
}
else if(IR[9:6] == 7){
    if(ACC == 0)
        PC = IR[5:0]
    durum = 0
}
else if(IR[9:6] == 8)
    durum = 0
else if(IR[9:6] == 9)
    durum = 4
```