

# Bölüm 1. Ardışıl Devreler

## **Geçen Hafta**

**Flip Flopların Uyarma Tablolarının Oluşturulması**

**Ardışıl Devrelerin Analizi**

**Ardışıl bir devrenin durum tablosuna bakılarak gerçekleştirilmesi**

## **Bu Hafta**

**Senkron Ardışıl Devre Tasarımının Adımları**

# Senkron Ardışıl Devre Tasarımının Adımları

---

1. Problemin sözlü tanımından hangi durumların olacağına ya da bellek elemanlarına hangi değerlerin atanacağına karar verilir.
2. Sistemin davranışını modelleyebilmek için durum tablosu ve durum diyagramı oluşturulur.
3. Flip flop seçiminin yapılması (belirtilmemişse) ve uyarma tablolarının oluşturulması.
4. Çıkış ve uyarma tablolarından, lojik ifadelerin elde edilmesi ve devrenin çizimi.

# Senkron Ardışıl Devre Tasarımının Adımları

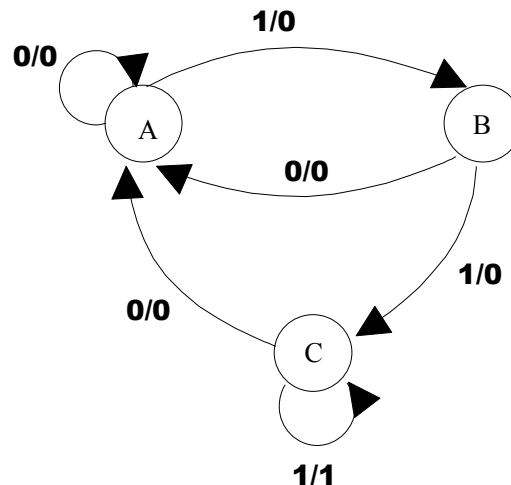
**Örnek:** 1 giriş ve 1 çıkışa sahip Mealy tipi bir sistemde, son 3 giriş verisi 1 ise çıkışın 1 olması isteniyor.

Bu problemin çözümünde 3 durum vardır; bu durumlar bellek elemanlarında saklanır.

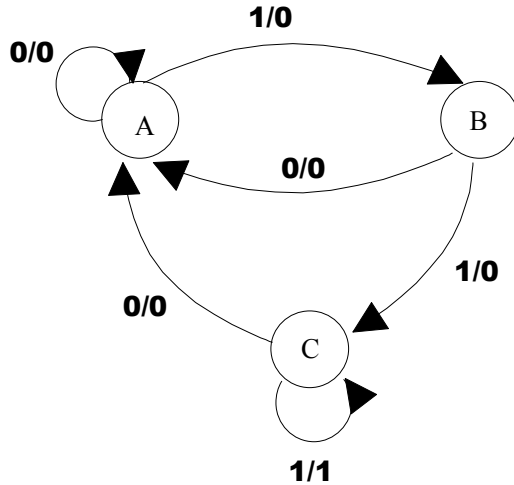
A durumu, girişin 1 olmadığı durumu,

B durumu, girişin bir defa 1 olduğu durumu,

C durumu, girişin iki ya da daha fazla 1 olduğu durumu gösterir.



## Örnek (devamı-1):



Durum diyagramı

Şimdiki Durum q	Sonraki Durum(Q)		Çıkış (z)	
	x=0	x=1	x=0	x=1
A	A	B	0	0
B	A	C	0	0
C	A	C	0	1

Devrenin durum tablosu

Durum tablosu 3 durum içerdiğinden 2 tane flip flop kullanmak gerekir.

A durumuna 00, B durumuna 01 ve C durumuna 10 atayalım.

Şimdiki Durum q <sub>1</sub> q <sub>2</sub>	Sonraki Durum(Q <sub>1</sub> Q <sub>2</sub> )		Çıkış (z)	
	x=0	x=1	x=0	x=1
0 0	0 0	0 1	0	0
0 1	0 0	1 0	0	0
1 0	0 0	1 0	0	1

## Örnek (devamı-2):

Devremizde bellek elemanı olarak JK tipi flip floplar kullanacağımızı farz edelim. Bundan sonraki adım, flip flopların uyarma işlevlerini bulmaktır.

Şimdiki Durum $q_1q_2$	Sonraki Durum( $Q_1Q_2$ ) $x=0$ $x=1$	Uyarma İşlevleri				Çıkış (z) $x=0$ $x=1$	
		$x=0$ için $J_1K_1$ $J_2K_2$	$x=1$ için $J_1K_1$ $J_2K_2$				
0 0	0 0 0 1	0 x 0 x	0 x 1 x			0	0
0 1	0 0 1 0	0 x x 1	1 x x 1			0	0
1 0	0 0 1 0	x 1 0 x	x 0 0 x			0	1



q Q	J K
0 0	0 x
0 1	1 x
1 0	x 1
1 1	x 0

## Örnek (devamı-3):

Uyarma işlevlerindeki her bir sütunu, Karnaugh haritasına taşıyıp indirgeme işlemlerini yapalım;

Şimdiki Durum $q_1q_2$	Sonraki Durum( $Q_1Q_2$ )		Uyarma İşlevleri				Çıkış (z)	
	x=0	x=1	x=0 için $J_1K_1 \quad J_2K_2$		x=1 için $J_1K_1 \quad J_2K_2$		x=0	x=1
0 0	0 0	0 1	0 x	0 x	0 x	1 x	0	0
0 1	0 0	1 0	0 x	x 1	1 x	x 1	0	0
1 0	0 0	1 0	x 1	0 x	x 0	0 x	0	1

$q_1q_2$ x \	00	01	11	10
0			x	x
1		1	x	x

$$J_1 = x.q_2$$

$q_1q_2$ x \	00	01	11	10
0		x	x	
1	1	x	x	

$$J_2 = x.q_1'$$

$q_1q_2$ x \	00	01	11	10
0	x	x	x	1
1	x	x	x	

$$K_1 = x'$$

$q_1q_2$ x \	00	01	11	10
0	x	1	x	x
1	x	1	x	x

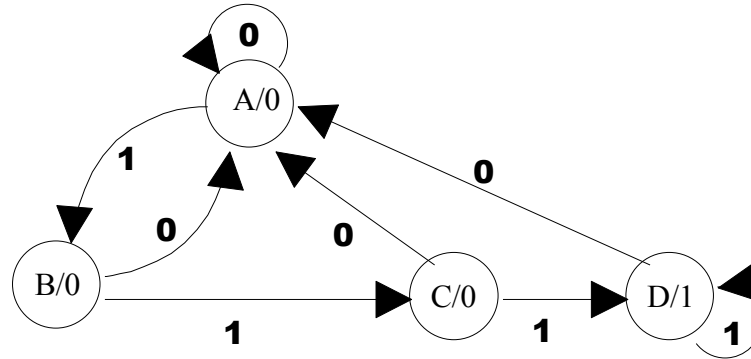
$$K_2 = 1$$

$q_1q_2$ x \	00	01	11	10
0			x	
1			x	1

$$z = x.q_1$$

# Senkron Ardışıl Devre Tasarımının Adımları

**Örnek:** Ard arda gelen 3 clock saykılında giriş 1 ise çıkışın 1 olmasını sağlayacak Moore türü devre tasarlanması isteniyor.

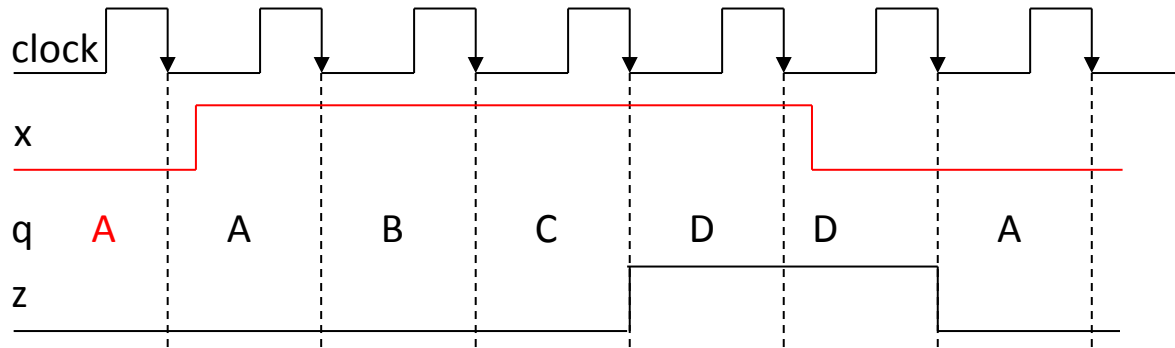


Sistemin başlangıçta A durumunda olduğunu farz ederek verilen x girişi için ardışıl devrenin durum geçişlerini ve çıkışı inceleyelim;

	A	B	C	D	D	D	D	A	B	C	A	B	C	D	A	B
x	0	1	1	1	1	1	1	0	1	1	0	1	1	1	0	1
z	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	1

## Örnek (devamı-1):

Ardışıl devrenin davranışını verilen x girişi için zaman ekseninde inceleyelim;

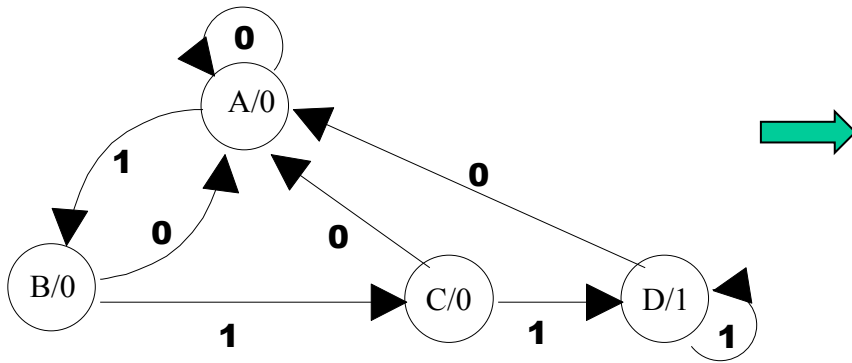




## Örnek (devamı-2):

Şimdi de A,B,C ve D durumlarına durum ataması yapıp, uyarma işlevlerini de içeren durum tablosunu oluşturalım;

A=00, B=01, C=10, D=11 durumlarını atayalım.



Şimdiki Durum x q <sub>1</sub> q <sub>2</sub>	Sonraki Durum Q <sub>1</sub> Q <sub>2</sub>	Uyarma İşlevleri		Çıkış z
		J <sub>1</sub> K <sub>1</sub>	J <sub>2</sub> K <sub>2</sub>	
0 0 0	0 0	0 x	0 x	0
0 0 1	0 0	0 x	x 1	0
0 1 0	0 0	x 1	0 x	0
0 1 1	0 0	x 1	x 1	1
1 0 0	0 1	0 x	1 x	0
1 0 1	1 0	1 x	x 1	0
1 1 0	1 1	x 0	1 x	0
1 1 1	1 1	x 0	x 0	1

## Örnek (devamı-3):

Uyarma işlevlerindeki her bir sütun, Karnaugh haritası yardımıyla indirgendiğinde, flip flopların girişlerine uygulanması gereken lojik ifadeler ve çıkış aşağıdaki gibi olacaktır;

Şimdiki Durum $x \ q_1 \ q_2$	Sonraki Durum $Q_1 \ Q_2$	Uyarma İşlevleri		Çıkış $z$
		$J_1 \ K_1$	$J_2 \ K_2$	
0 0 0	0 0	0 x	0 x	0
0 0 1	0 0	0 x	x 1	0
0 1 0	0 0	x 1	0 x	0
0 1 1	0 0	x 1	x 1	1
1 0 0	0 1	0 x	1 x	0
1 0 1	1 0	1 x	x 1	0
1 1 0	1 1	x 0	1 x	0
1 1 1	1 1	x 0	x 0	1

Karnaugh



$$\begin{aligned} J_1 &= q_2 \cdot x \\ K_1 &= x' \\ J_2 &= x \\ K_2 &= x' + q_1' \\ z &= q_1 \cdot q_2 \end{aligned}$$

Devremiz Moore türü olduğundan dolayı, çıkış sadece durumlara bağlıdır. D durumunda çıkış 1 olduğundan ve D durumuna da 11 atadığımızdan dolayı  $z = q_1 \cdot q_2$  olmuştur

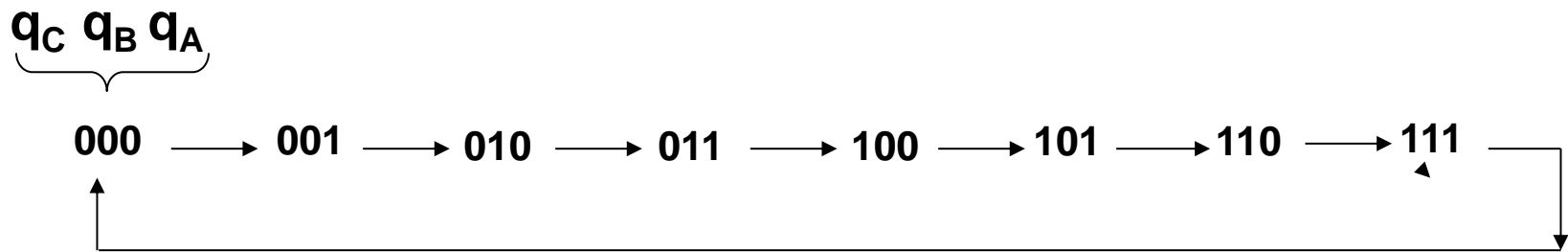
# Senkron Ardışıl Devre Tasarımının Adımları

## Örnek: Senkron Sayıcı Tasarımı

Sayıcı devrelerinin genellikle girişleri yoktur; saat geçişiyle durumlar arasında geçiş sağlanır. Çıkışlar, sistemin durumlarıdır, başka bir ifadeyle flip flopların tuttuğu değerlerdir. Bir sayıcının hem ileri hem de geri sayması istenirse, bu işlev için bir giriş kullanılması gerekir.

3 bit ikili bir sayıcının tasarım aşamaları:

Bu sayıcı 0 ile 7 arasında ileri yönde sayma yapacağından 3 adet flip flop kullanılacaktır. Bu flip flopların isimleri  $q_C$  (MSB),  $q_B$  ve  $q_A$  (LSB) olsun.



## Örnek (devamı-1):

Şimdiki Durum $q_C q_B q_A$	Sonraki Durum $Q_C Q_B Q_A$
0 0 0	0 0 1
0 0 1	0 1 0
0 1 0	0 1 1
0 1 1	1 0 0
1 0 0	1 0 1
1 0 1	1 1 0
1 1 0	1 1 1
1 1 1	0 0 0

Şimdiki Durum $q_C q_B q_A$	Sonraki Durum $Q_C Q_B Q_A$	Uyarma İşlevleri		
		$J_C K_C$	$J_B K_B$	$J_A K_A$
0 0 0	0 0 1	0 x	0 x	1 x
0 0 1	0 1 0	0 x	1 x	x 1
0 1 0	0 1 1	0 x	x 0	1 x
0 1 1	1 0 0	1 x	x 1	x 1
1 0 0	1 0 1	x 0	0 x	1 x
1 0 1	1 1 0	x 0	1 x	x 1
1 1 0	1 1 1	x 0	x 0	1 x
1 1 1	0 0 0	x 1	x 1	x 1

Uyarma işlevleri

Karnaugh ile indirgendğinde;

$$J_C = K_C = q_B \cdot q_A$$

$$J_B = K_B = q_A$$

$$J_A = K_A = 1$$



4 bitlik bir sayıcı yapmak isteseydik;

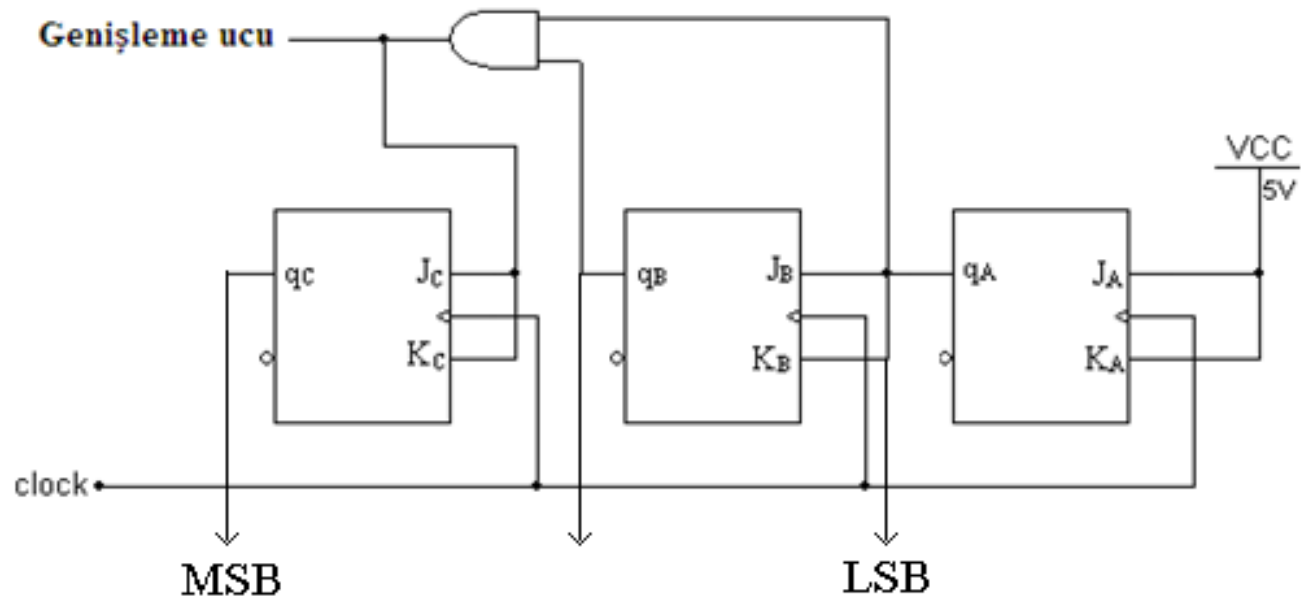
$J_D = K_D = q_C \cdot q_B \cdot q_A$  olacağı tahmin edilebilir.

## Örnek (devamı-2):

$$J_C = K_C = q_B \cdot q_A$$

$$J_B = K_B = q_A$$

$$J_A = K_A = 1$$



Sayıc1 tasarımlının devre şeması

## NOTLAR:

---

1. İleri yönde sayan sayıcı devrelerin  $q$  çıkışları yerine  $q'$  çıkışlarından uç alınması durumunda geri yönde sayan sayıcı elde edilir (tersi de geçerlidir).
2. Şayet sayıcı uygulaması sıralı değilse, tasarım aşamaları yine aynıdır. Ancak kullanılmayan durumlar varsa, sonraki durum önemsiz durum olarak ele alınacağından, devrenin böyle bir durumdan başlaması halinde, bir sonraki durumu tahmin etmek mümkün değildir (bu durumu önlemek için flip flopların clear ve preset girişleri vardır).

