Sayısal Sistemler-H10CD2

Durum Tabloları, Durum Diyagramları, Durum Denklemleri

Dr. Meriç Çetin

versiyon211120

Bu derste öğreneceklerimiz

5 Synchronous Sequential Logic

Flip-flop doğruluk tabloları

GİRİŞLER		ÇIKIŞ
S	R	$Q_{(t+1)}$
0	0	$Q_{(t)}$
0	1	0
1	0	1
1	1	Kullanılmaz

GİRİŞLER		ÇIKIŞ
J	K	$Q_{(t+1)}$
0	0	$Q_{(t)}$
0	1	0
1	0	1
1	1	$\overline{Q_{(t)}}$

GİRİŞ	ÇIKIŞ
D	$Q_{(t+1)}$
0	0
1	1

GİRİŞ	ÇIKIŞ
T	$Q_{(t+1)}$
0	$Q_{(t)}$
1	$\overline{Q_{(t)}}$

Flip-flop durum geçiş tabloları

DURUM GEÇİŞLERİ		GİRİŞLER	
$Q_{(t)}$	$Q_{(t+1)}$	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

S=0, R=0 veya S=0, R=1

S=0, R=0 veya S=1, R=0

RS FF Durum Geçiş Tablosu

DURUM GEÇİŞLERİ		GİRİŞLER	
$Q_{(t)}$	$Q_{(t+1)}$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

S=0, R=0 veya S=0, R=1

S=1, R=0 veya S=1, R=1

S=0, R=1 veya S=1, R=1

S=0, R=0 veya S=1, R=0

JK FF Durum Geçiş Tablosu

DURUM GEÇİŞLERİ		GİRİŞ
$Q_{(t)}$	$Q_{(t+1)}$	D
0	0	0
0	1	1
1	0	0
1	1	1

D FF Durum Geçiş Tablosu

DURUM GEÇİŞLERİ		GİRİŞ
$Q_{(t)}$	$Q_{(t+1)}$	T
0	0	0
0	1	1
1	0	1
1	1	0

T FF Durum Geçiş Tablosu

Durum Diyagramları

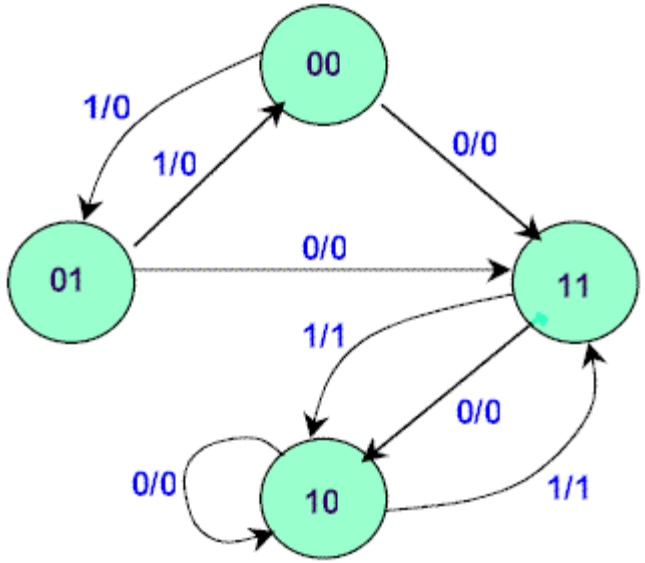
- Durum geçiş tablosunda gösterilen ifadeler durum diyagramlarında da grafiksel olarak gösterilebilirler.

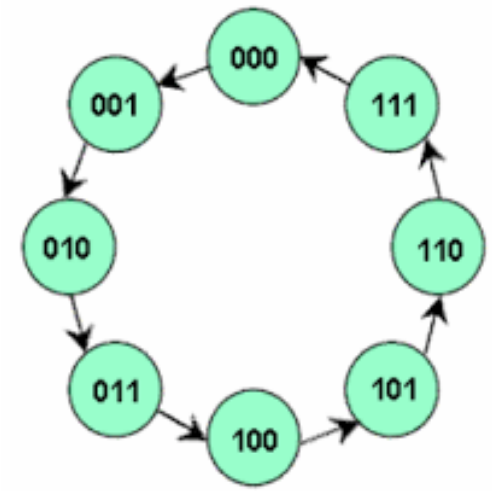
- Bu diyagramlarda **daire** içinde gösterilen değerler önceki durumları/durum değiştirmeleri veya durum geçişlerini gösterirler.
- **Direk çizgi (/)** ile birbirine bağlanan dairelerde binary sayılar « / » ile ayrılmıştır.
  - Bunlardan **1. ifade giriş** değerini, **2. ifade ise çıkış** değerini temsil eder.
  - Örneğin «1/0» ise giriş x=1, çıkış y=0 anlamındadır.

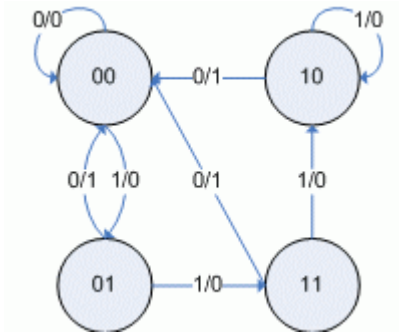
Durum Diyagramları

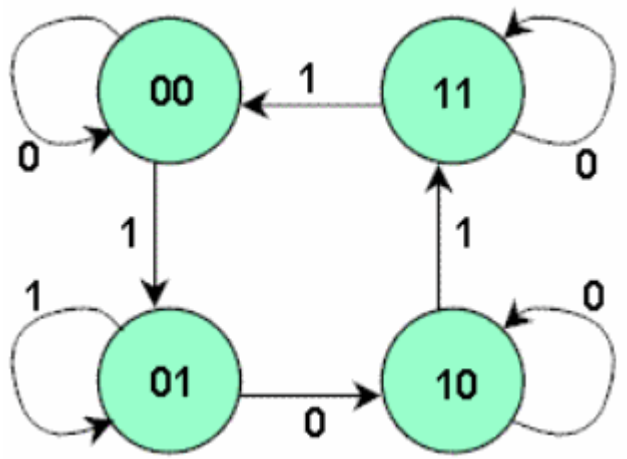
- Gösterim şekli hariç durum tablosu ile durum diyagramları arasında bir fark yoktur.
- Durum tabloları basitçe verilen lojik devreden çıkarılabilir.
- Bunun yanında durum diyagramlarındaki değişmeler de durum tablolarından direkt olarak elde edilebilir.
- Durum diyagramları ardışık lojik devre tasarımının ilk adımını oluşturur.

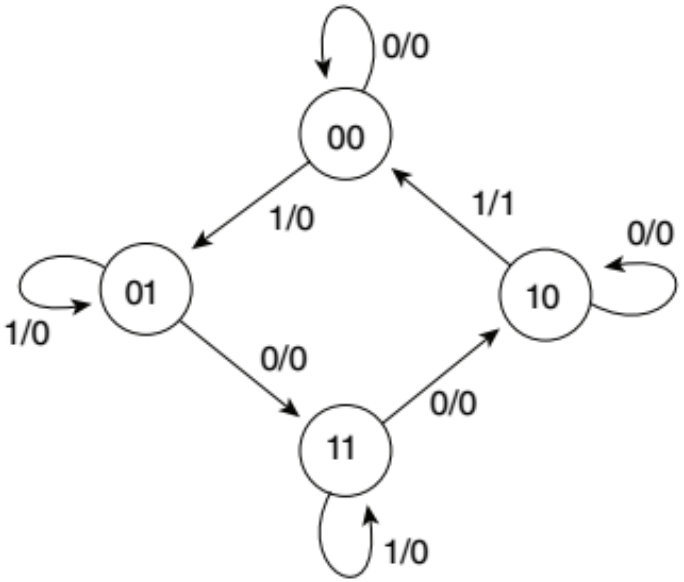
Durum diyagramlarına örnek











Durum tablosundan durum diyagramına geçiş

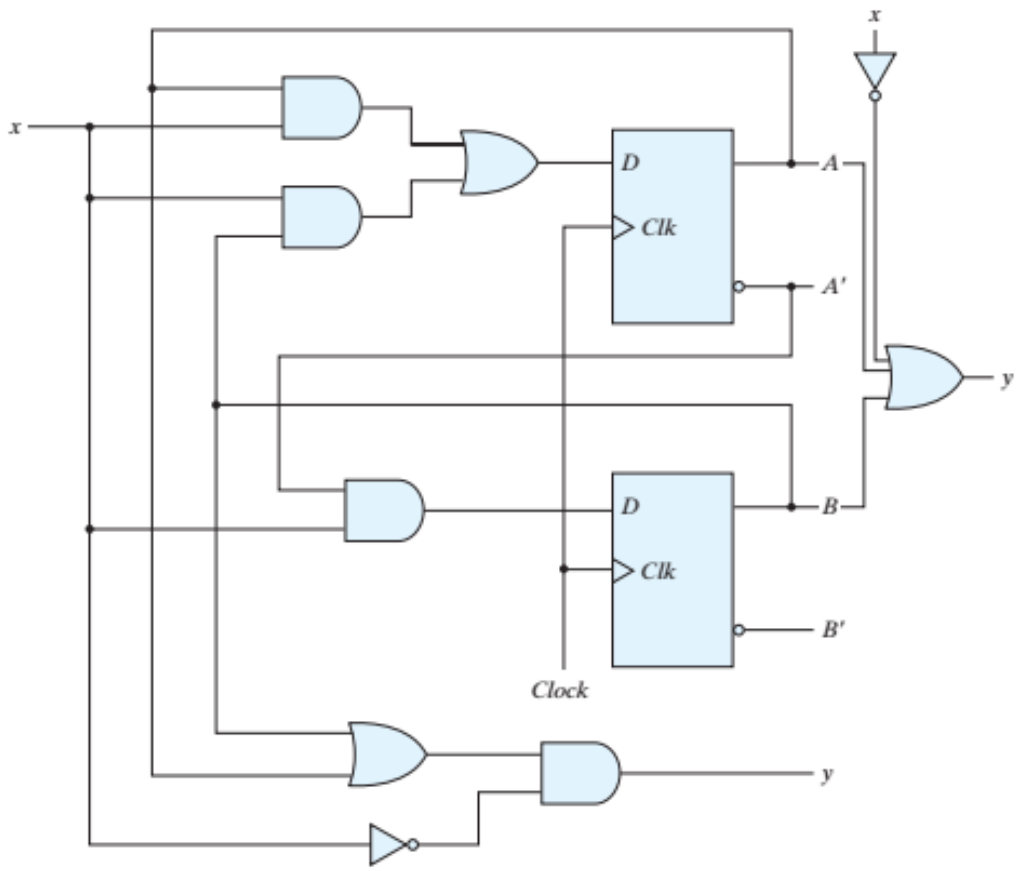


FIGURE 5.15
Example of sequential circuit

Table 5.3
Second Form of the State Table

Present State		Next State				Output	
		$x = 0$		$x = 1$		$x = 0$	$x = 1$
A	B	A	B	A	B	y	y
0	0	0	0	0	1	0	0
0	1	0	0	1	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	0	1	0

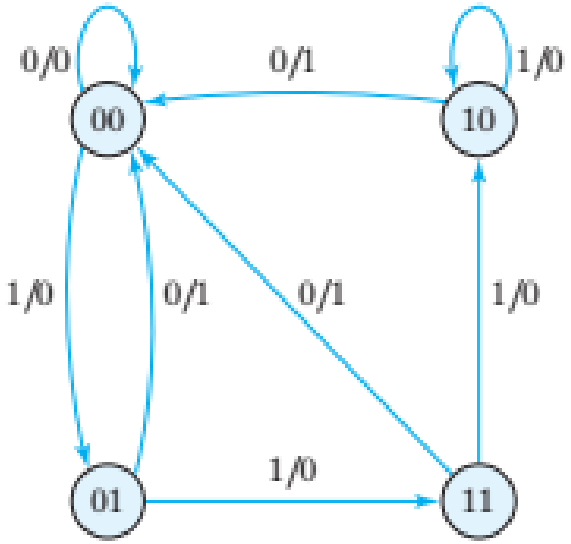
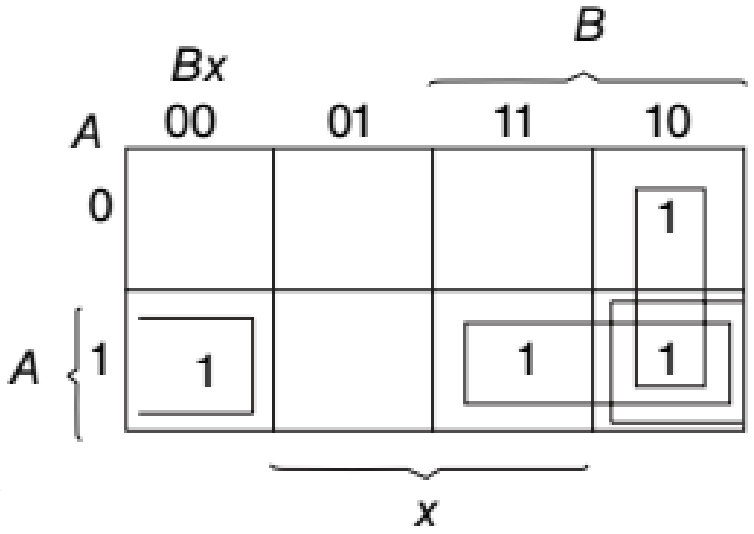


FIGURE 5.16
State diagram of the circuit of Fig. 5.15

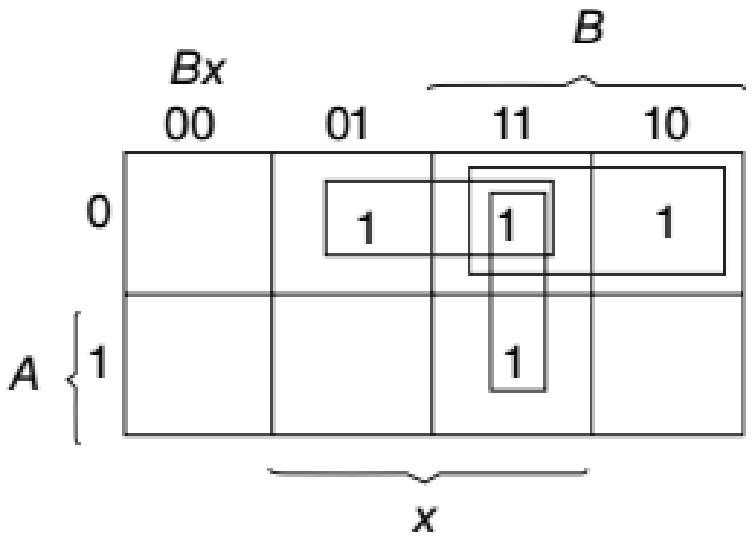
Durum Denklemleri

- Durum denklemleri flip-flop'ların durum değiştirmeleri için gerekli şartları belirleyen cebrik ifadelerdir.
- Denklemlerin sol tarafı flip-flop'ların sonraki durumlarını (clock pulse sonrası değerlerini) temsil eder.
- Denklemlerin sağ tarafı ise Boolean fonksiyonudur.
- Durum denklemleri direkt durum tablolarından elde edilir.
- Durum tablolarında ilgili flip-flop'ların sonraki durumlarına ilişkin sütunlardan ve çıkış ifadelerinde değeri «1» olan değişkenlere göre Karnaugh haritaları ile sadeleştirmeler yapılır. Flip-flop'ların sonraki durumları bu şekilde hesaplanmış olur.

Durum denklemlerine örnek



$$A(t+1) = Bx' + (B + x')A$$
$$= Bx' + (B'x)'A$$



$$B(t+1) = A'x + (A' + x)B$$
$$= Ax + (Ax')'B$$

$$A(t+1) = B'x + (B'x)'A$$

State equation for flip-flops A and B

$$A(t+1) = S + R'A$$

Durum denklemlerine örnek-devam

The state equation for flip-flop A is simplified by means of a map as shown in Fig. 6-17(a). With some algebraic manipulation, the function can be expressed in the following form:

$$A(t+1) = B'x + (B'x)'A$$

If we let $Bx' = S$ and $B'x = R$. we obtain the relationship:

$$A(t+1) = S + R'A$$

$$B(t+1) = A'x + (Ax')'B$$

The state equation can be derived directly from the logic diagram. From Fig. 6-15, we see that the signal for input S of flip-flop B is generated by the function $A'x$ and the signal for input R by the function Ax'. Substituting $S = A'x$ and $R = Ax'$ into an RS flip-flop characteristic equation given by:

$$B(t+1) = S + R'B$$

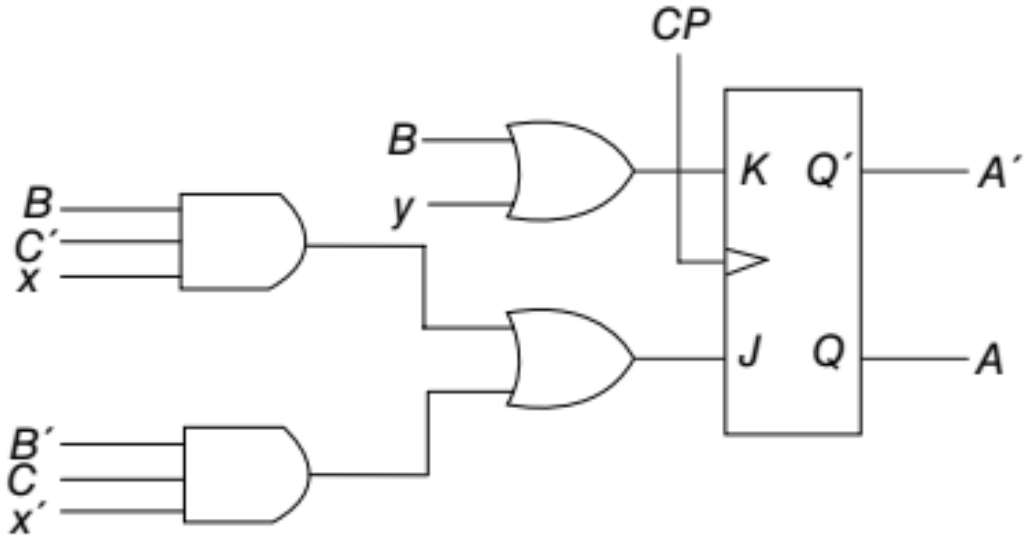
Flip-Flop Giriş Fonksiyonları

- Ardışık lojik devreler, hafıza elemanları ve mantık kapılarından meydana gelir.
- Flip-flop türü ve onların karakteristik tabloları hafıza elemanlarının lojik özelliklerini belirler.
- Bu kapıların birbiriyle ara bağlantıları bir kombinasyonel devre meydana getirir ve bu devrenin özellikleri de Boolean cebri ile tanımlanır.
- Çıkış fonksiyonu için cebrik ifadeler devre çıkış fonksiyonuyla belirlenir.

Flip-Flop Giriş Fonksiyonları-devam

- Örneğin
- Yandaki devre için flip-flop giriş fonksiyonları ve flip-flop çıkış ifadesi aşağıdaki gibidir.

$$JA = BC'x + B'Cx'$$
$$KA = B + y$$



Flip-Flop Giriş Fonksiyonları-devam

- Başka bir örnek inceleyelim:

- Yandaki devre için flip-flop giriş fonksiyonları ve flip-flop çıkış ifadesi aşağıdaki gibidir.

$$D_A = Ax + Bx$$
$$D_B = A'x$$
$$y = (A + B)x'$$

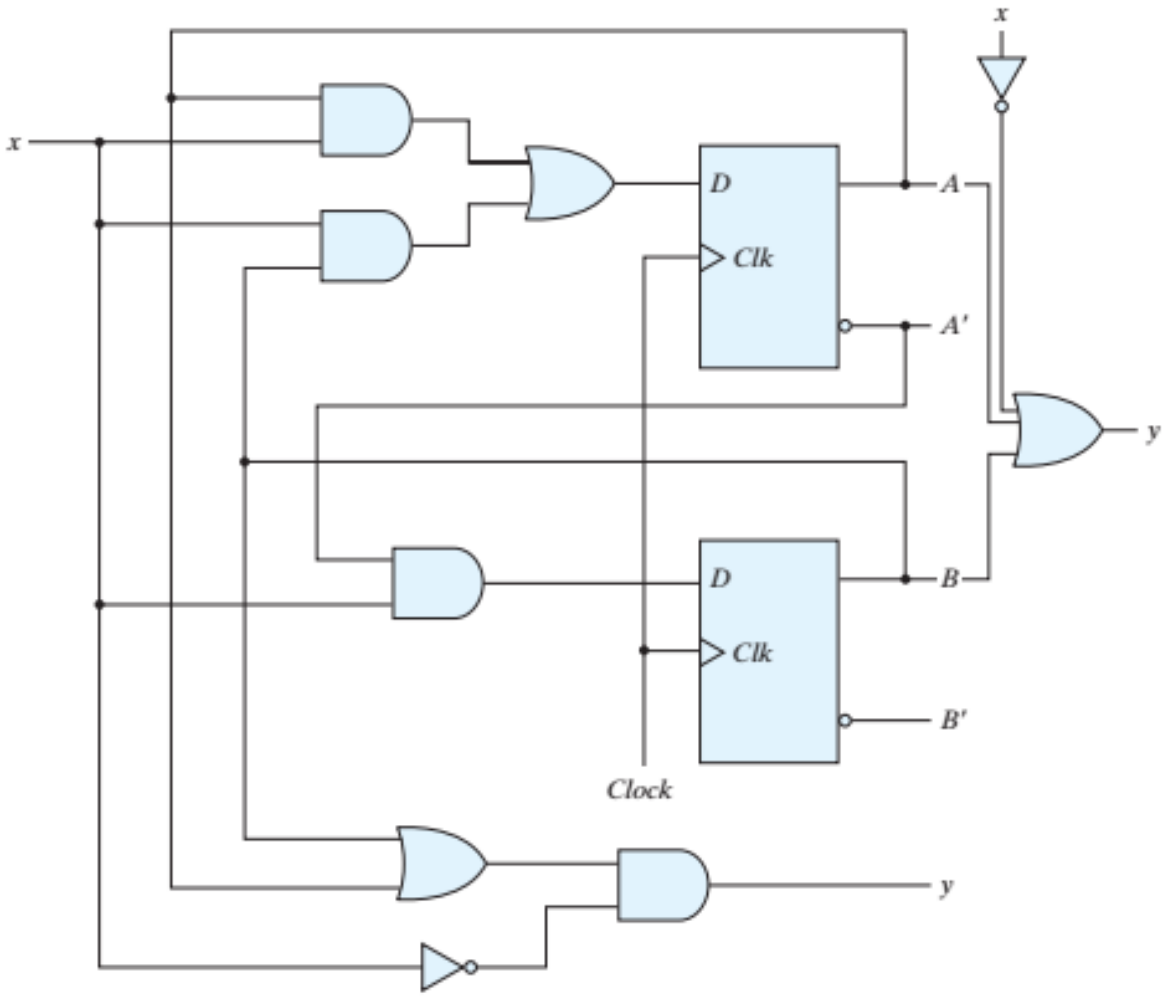


FIGURE 5.15
Example of sequential circuit

Flip-Flop Giriş Fonksiyonları-devam

- Başka bir örnek inceleyelim:

- Yandaki devre için flip-flop giriş fonksiyonları ve flip-flop çıkış ifadesi aşağıdaki gibidir.

$$J_A = B \quad K_A = Bx'$$
$$J_B = x' \quad K_B = A'x + Ax' = A \oplus x$$

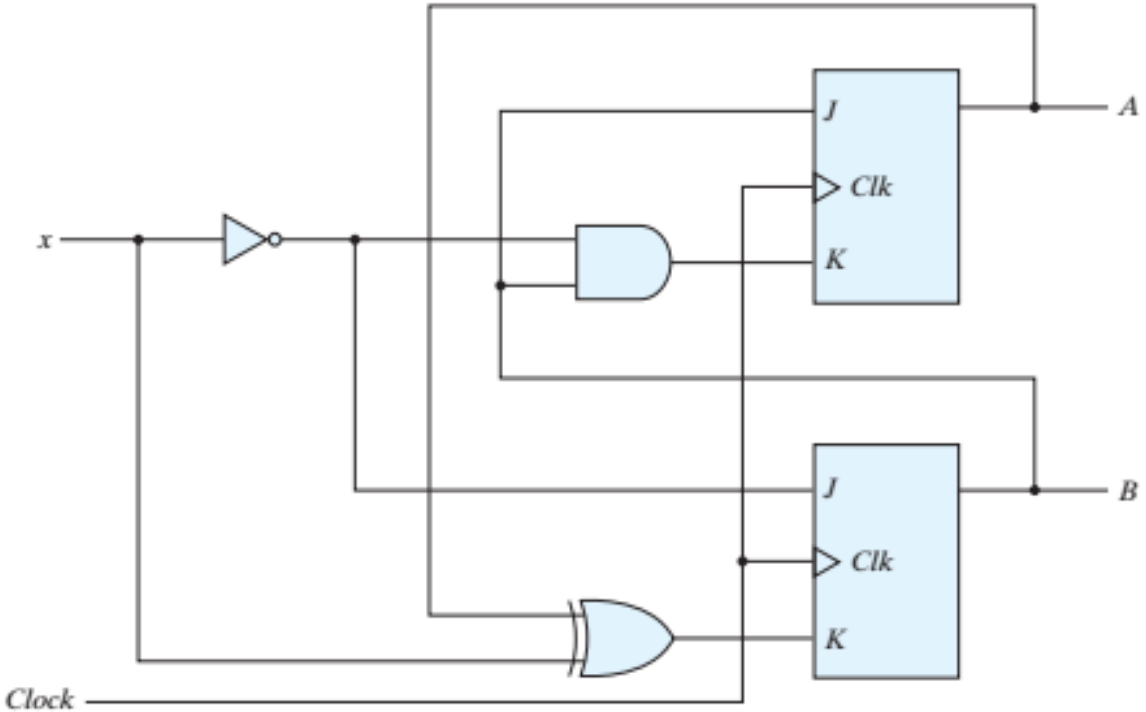


FIGURE 5.18
Sequential circuit with *JK* flip-flop

Ardışık Lojik Devre Tasarımı

Tasarım prosedürü

- Ardışık lojik devre tasarımı için şu yol takip edilmelidir:

- Devre davranışı tanımlanır. Bu, durum diyagramlarıyla belirlenir.
- Elde edilen değerler durum tablosuna taşınır.
- Gerekli flip-flop sayısı ve flip-flop türü belirlenir.
- Karnaugh veya diğer indirgeme metotları kullanılarak kombinasyonel devre çıkış ve flip-flop giriş denklemleri elde edilir.
- Elde edilen bu sonuçlara göre lojik devre tasarımı yapılır.

Durum İndirgeme

- Ardışık bir devrede flip-flop sayısındaki azalma, durum indirgeme problemi olarak adlandırılır.
- Durum indirgeme algoritmaları, harici giriş-çıkış gereksinimlerini değiştirmeden tutarken, bir durum tablosundaki durumların sayısını azaltmak için prosedürlerle ilgilidir.
- m adet flip-flop 2^m adet durum ürettiğinden, durum sayısındaki bir azalma, flip-flopların sayısında bir azalmaya neden olabilir (veya olmayabilir).
- Flip-flop sayısını azaltmada öngörülemeyen bir etki, bazen eşdeğer devrenin (daha az flip-flop ile) bir sonraki durumunu ve çıkış mantığını gerçekleştirmek için daha fazla kombinasyon kapısı gerektirmesidir.

Durum İndirgeme-örnek

- Durum indirgeme prosedürünü inceleyeceğimiz bu örnekte, yalnızca girdi-çıktı dizileri önemlidir; iç durumlar yalnızca gerekli dizileri sağlamak için kullanılır.
- Bu nedenle, dairelerin içinde işaretlenen durumlar ikili değerler yerine harf sembolleriyle gösterilmiştir.

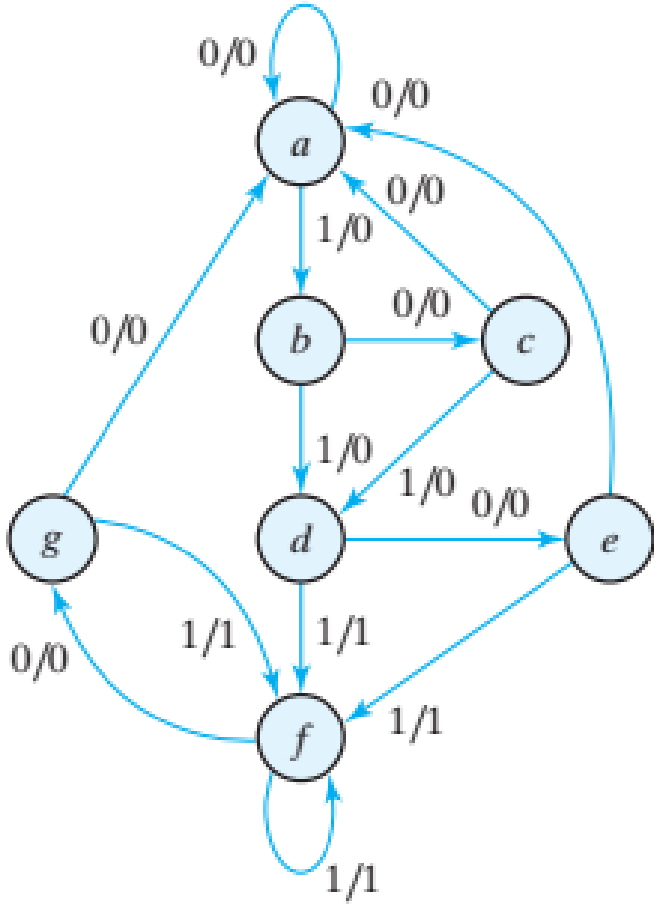


FIGURE 5.25
State diagram

state	*a*	*a*	*b*	*c*	*d*	*e*	*f*	*f*	*g*	*f*	*g*	*a*
input	0	1	0	1	0	1	1	0	1	0	0	
output	0	0	0	0	0	1	1	0	1	0	0	

Durum İndirgeme-örnek

- Durum diyagramından durum tablosunu düzenleyelim:

Table 5.6
State Table

	Next State		Output	
Present State	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	g	f	0	1
g	a	f	0	1

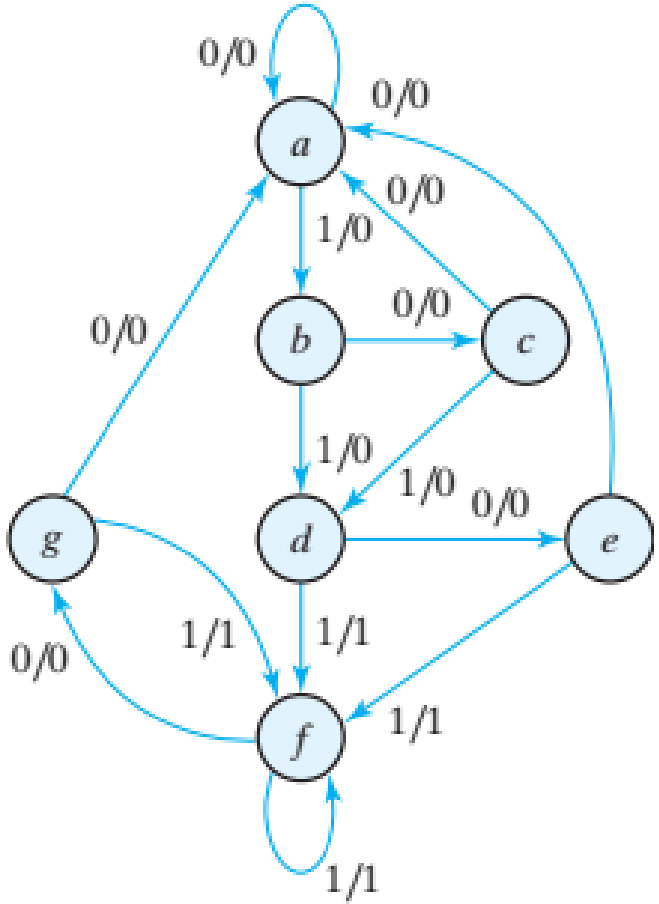


FIGURE 5.25
State diagram

Durum İndirgeme-örnek

- Durum tablosunu inceleyerek, aynı sonraki duruma giden ve her iki giriş kombinasyonu için aynı çıktıya sahip iki mevcut durumu ararız.

Table 5.6
State Table

Present State	Next State $x=0$	Next State $x=1$	Output $x=0$	Output $x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	g	f	0	1
g	a	f	0	1

e,g →

Table 5.7
Reducing the State Table

Present State	Next State $x=0$	Next State $x=1$	Output $x=0$	Output $x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	e	f	0	1

d,f →

Table 5.8
Reduced State Table

Present State	Next State $x=0$	Next State $x=1$	Output $x=0$	Output $x=1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

Durum İndirgeme-örnek

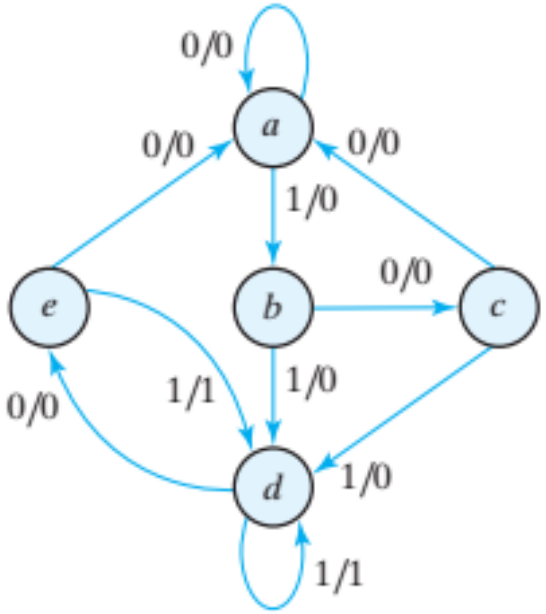


FIGURE 5.26
Reduced state diagram

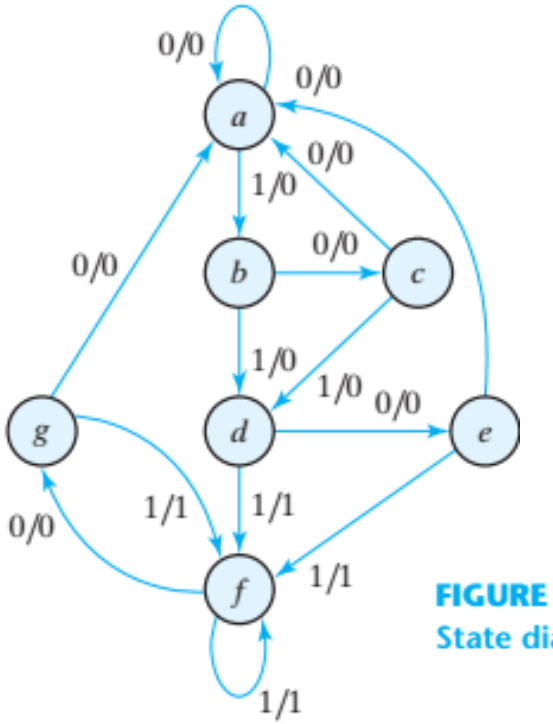


FIGURE 5.25
State diagram

Table 5.8
Reduced State Table

	Next State		Output	
Present State	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	d	0	1
e	a	d	0	1

Table 5.6
State Table

	Next State		Output	
Present State	$x = 0$	$x = 1$	$x = 0$	$x = 1$
a	a	b	0	0
b	c	d	0	0
c	a	d	0	0
d	e	f	0	1
e	a	f	0	1
f	g	f	0	1
g	a	f	0	1