



# Lojik Tasarım

Ders 13

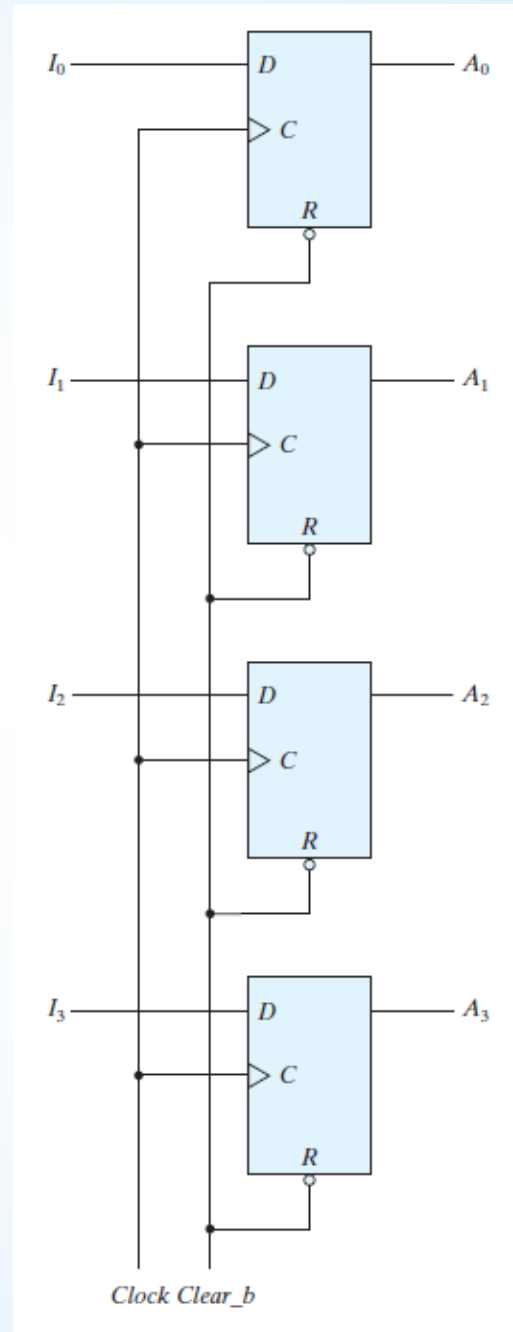
Kaynak:

M.M. Mano, M.D. Ciletti, "Digital Design with An Introduction to Verilog HDL"

# Registers (Yazıcılar – Kaydediciler)

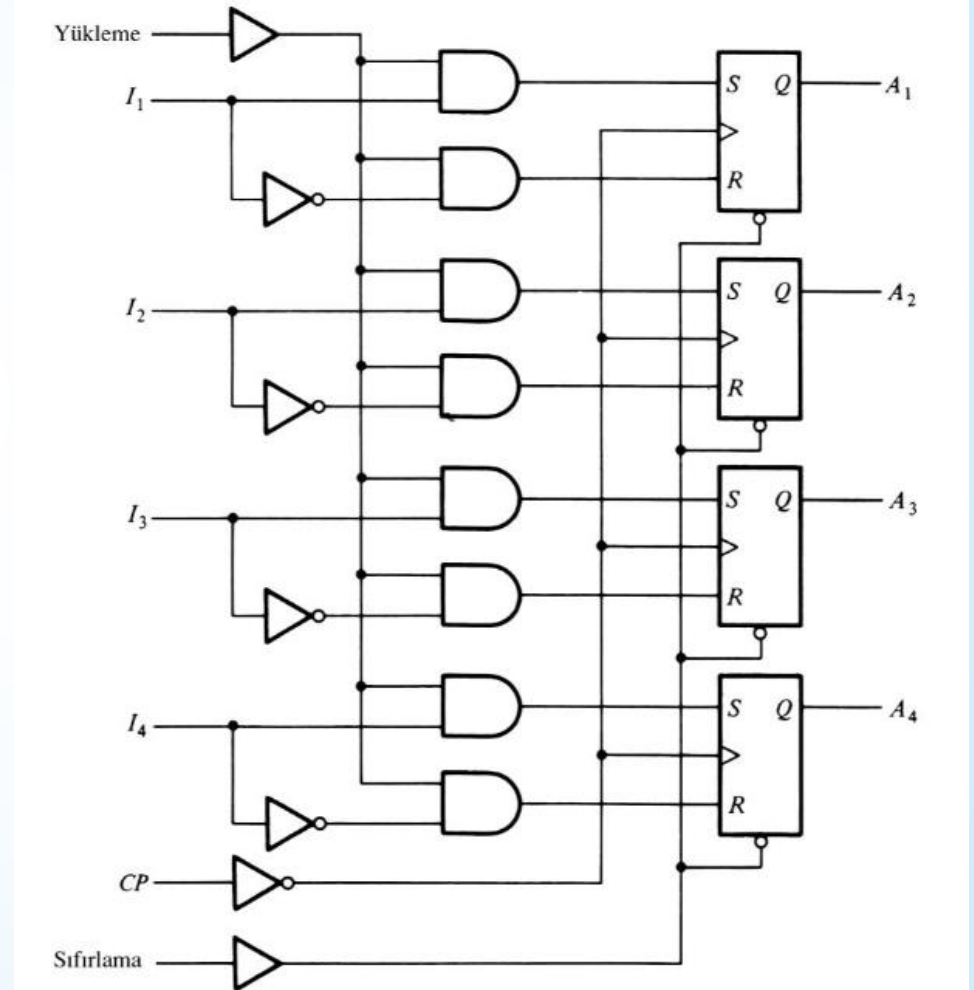
- İki yada daha fazla bitten oluşan bilgi yüklenebilen, RESET işareti verilene kadar veya yeni bir bilgi yüklenene kadar üzerindeki bilgiyi koruyabilen fonksiyonel yapılara register denir.
- Sayıcılar, durumları daha önceden belirlenen durumlara göre değişen registerlardır.
- $n$  bitlik bir register  $n$  adet flip-floptan oluşur

# 4 bitlik register



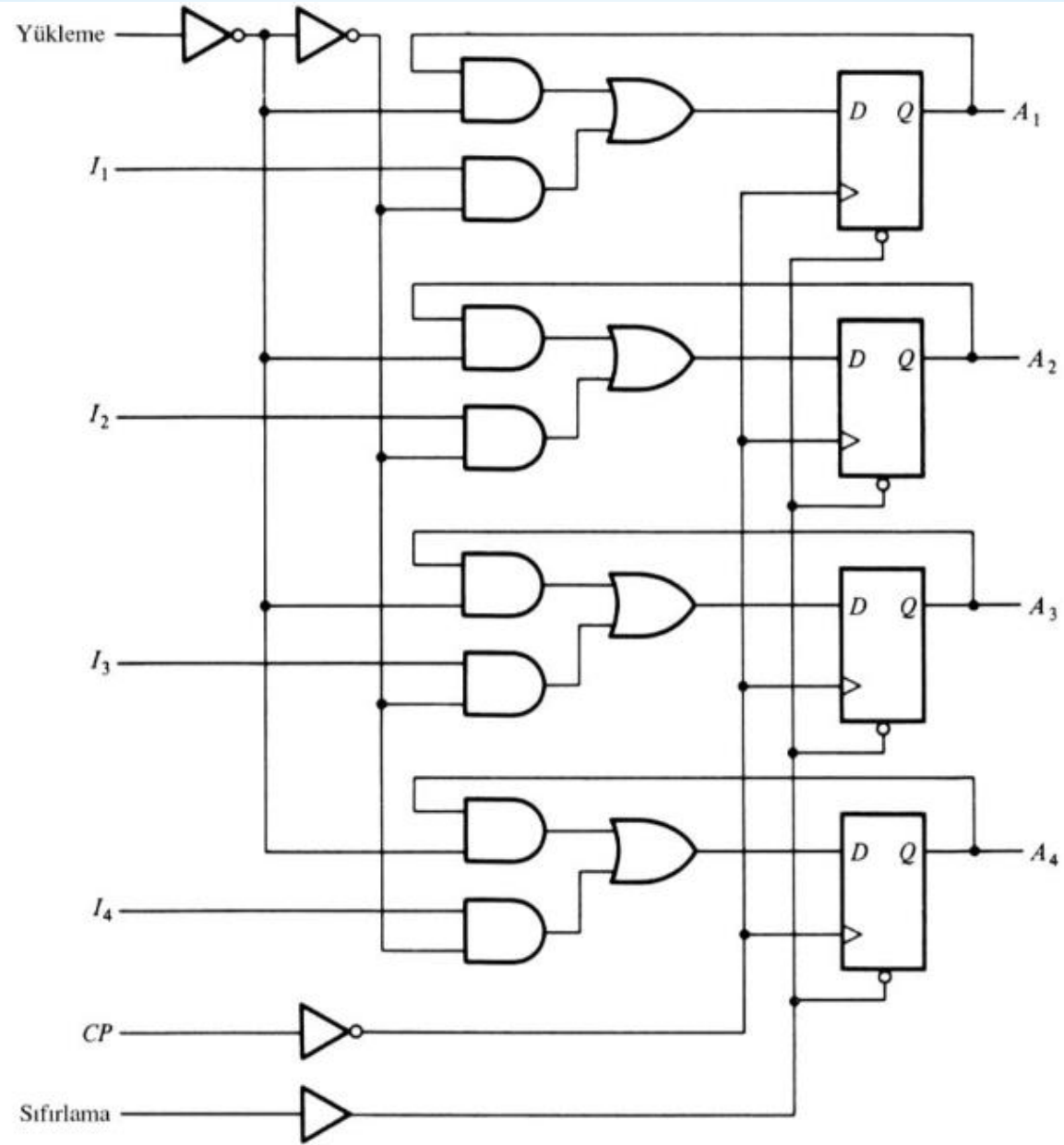
# Paralel Yüklemeli Registerlar

- Sayısal bir sistemde sürekli bir saat palsi vardır
- Saat palsini VE'lemek yayılım gecikmesinden dolayı senkronizasyonu bozar
- Tampon kapılar yükü azaltmak için kullanılmıştır



ŞEKİL 7-2

Paralel yüklemeli 4 bitlik yazıcı



**ŞEKİL 7-3**

Paralel yüklemeli D flip-flopları kullanılan yazıcı

**Örnek  
7-1**

Durum tablosu Şekil 7-5(a)'da verilen ardışıl devreyi tasarlayın.

Tabloda,  $A_1$  ile  $A_2$  flip-flopları,  $x$  girişi ve  $y$  de çıkışı göstermektedir. Sonraki durum ve çıkış bilgisi doğrudan tablo yardımıyla elde edilebilir:

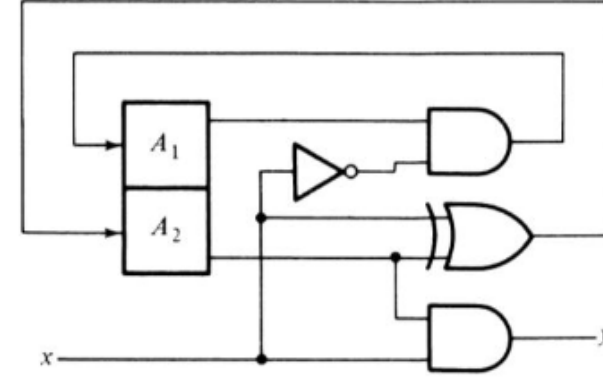
$$A_1(t+1) = \Sigma(4, 6)$$

$$A_2(t+1) = \Sigma(1, 2, 5, 6)$$

$$y(A_1, A_2, x) = \Sigma(3, 7)$$

Şimdiki durum		Giriş $x$	Sonraki durum		Çıkış $y$
$A_1$	$A_2$		$A_1$	$A_2$	
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	0	1
1	0	0	1	0	0
1	0	1	0	1	0
1	1	0	1	1	0
1	1	1	0	0	1

(a) Durum tablosu



(b) Lojik devre

**ŞEKİL 7-5**

Ardışıl devre uygulama örneği

Minterimlerdeki değişkenler  $A_1$  ve  $A_2$ 'nin şimdiki durumlarıyla  $x$  girişidir. Sonraki durum ve çıkışa ilişkin fonksiyonlar diyagramlar kullanılarak aşağıdaki gibi basitleştirilebilir:

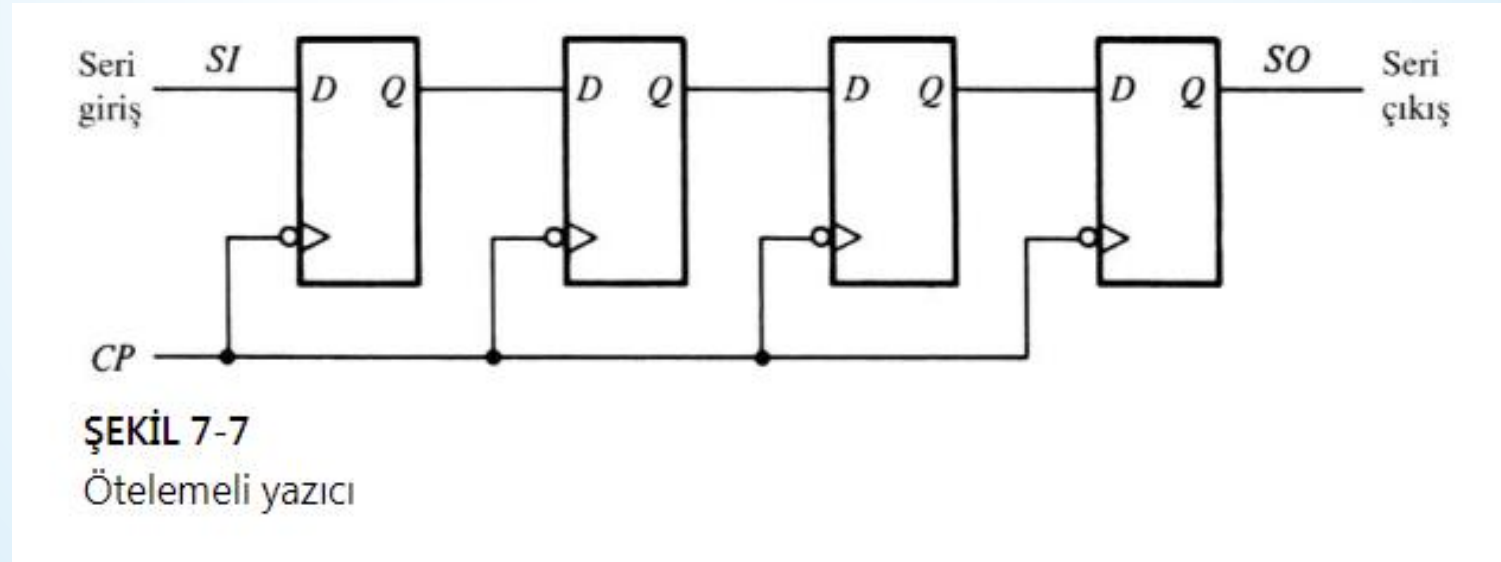
$$A_1(t+1) = A_1 x'$$

$$A_2(t+1) = A_2 \oplus x$$

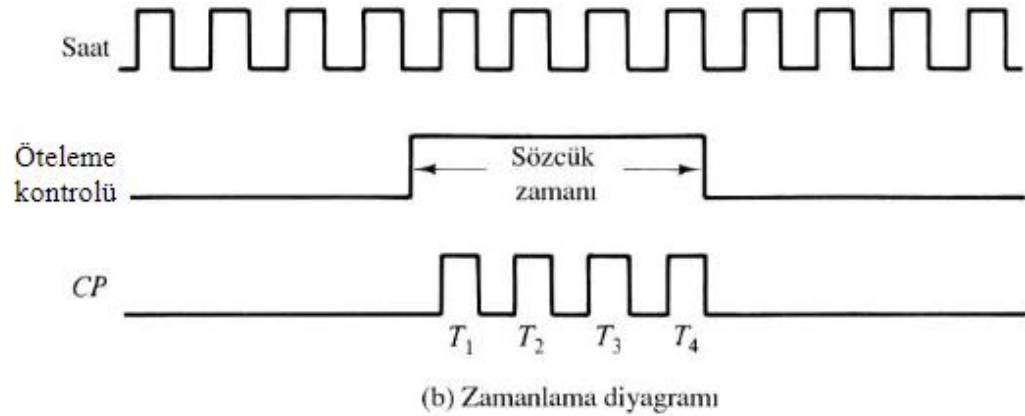
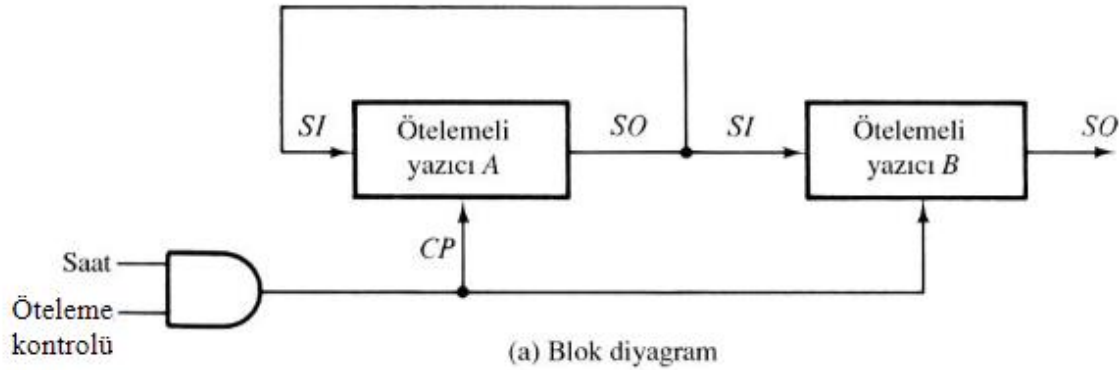
$$y = A_2 x$$

# Shift Register (Ötelemeli – Kaydırmalı Yazıcılar)

- Register içerisindeki bilgini sağa – sola kaydırılması amacıyla kullanılır.
- Aşağıda sağa ötelemeli bir shift register devresi görülmektedir.



# Seri veri transferi



ŞEKİL 7-8

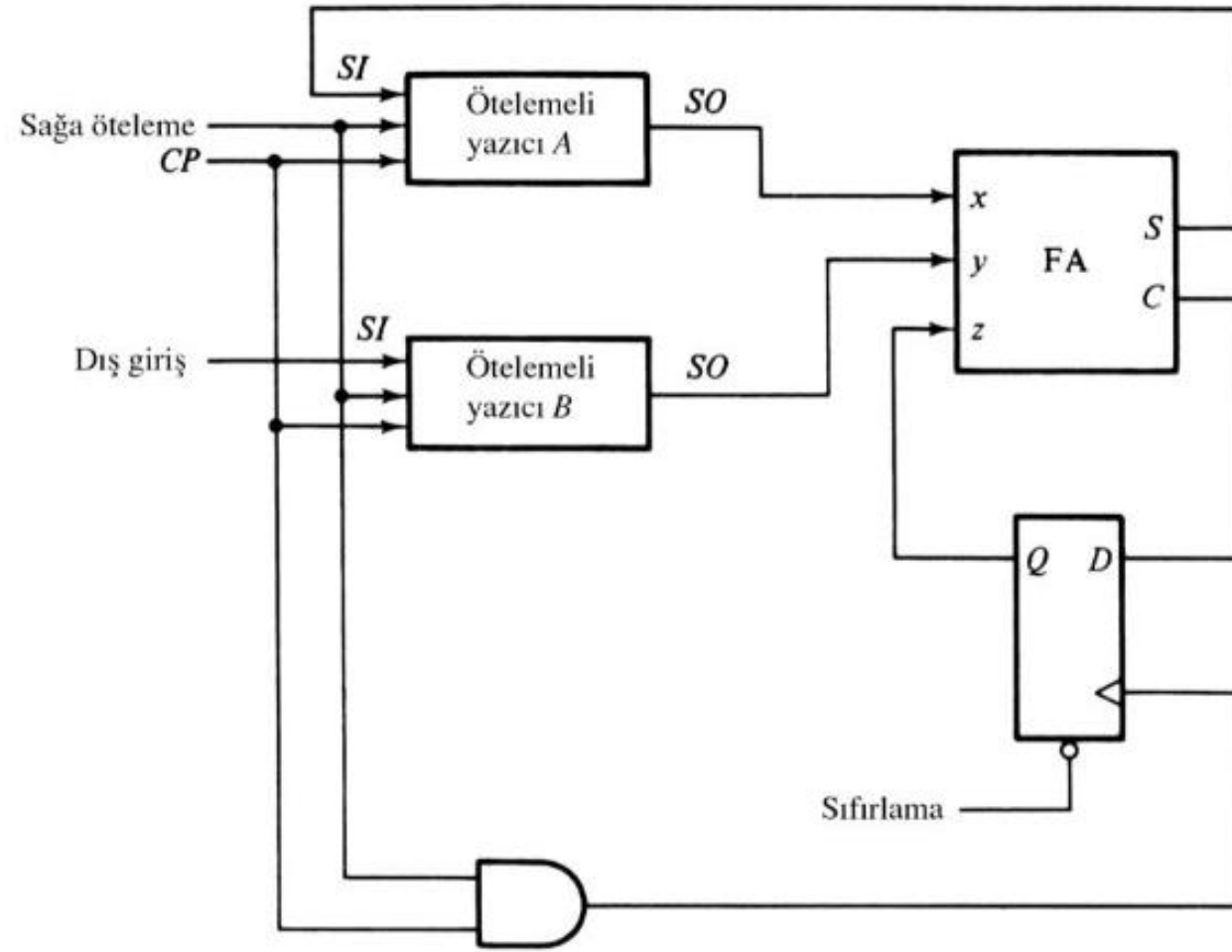
A'dan B yazıcısına seri transfer

TABLO 7-1  
Seri Transfer Örneği

Zamanlama Darbesi	Ötelemeli Yazıcı A	Ötelemeli Yazıcı B	B'nin Seri Çıkışı
Başlangıç değeri	1 0 1 1	0 0 1 0	0
$T_1$ sonrası	1 1 0 1	1 0 0 1	1
$T_2$ sonrası	1 1 1 0	1 1 0 0	0
$T_3$ sonrası	0 1 1 1	0 1 1 0	0
$T_4$ sonrası	1 0 1 1	1 0 1 1	1



# Seri Toplama



ŞEKİL 7-10  
Seri toplayıcı

**Soru:** Bir seri toplayıcıyı JK tipi FF kullanarak tasarlayınız

*State Table for Serial Adder*

Present State	Inputs		Next State	Output	Flip-Flop Inputs	
	$x$	$y$			$J_Q$	$K_Q$
0	0	0	0	0	0	X
0	0	1	0	1	0	X
0	1	0	0	1	0	X
0	1	1	1	0	1	X
1	0	0	0	1	X	1
1	0	1	1	0	X	0
1	1	0	1	0	X	0
1	1	1	1	1	X	0

$$J_Q = xy$$

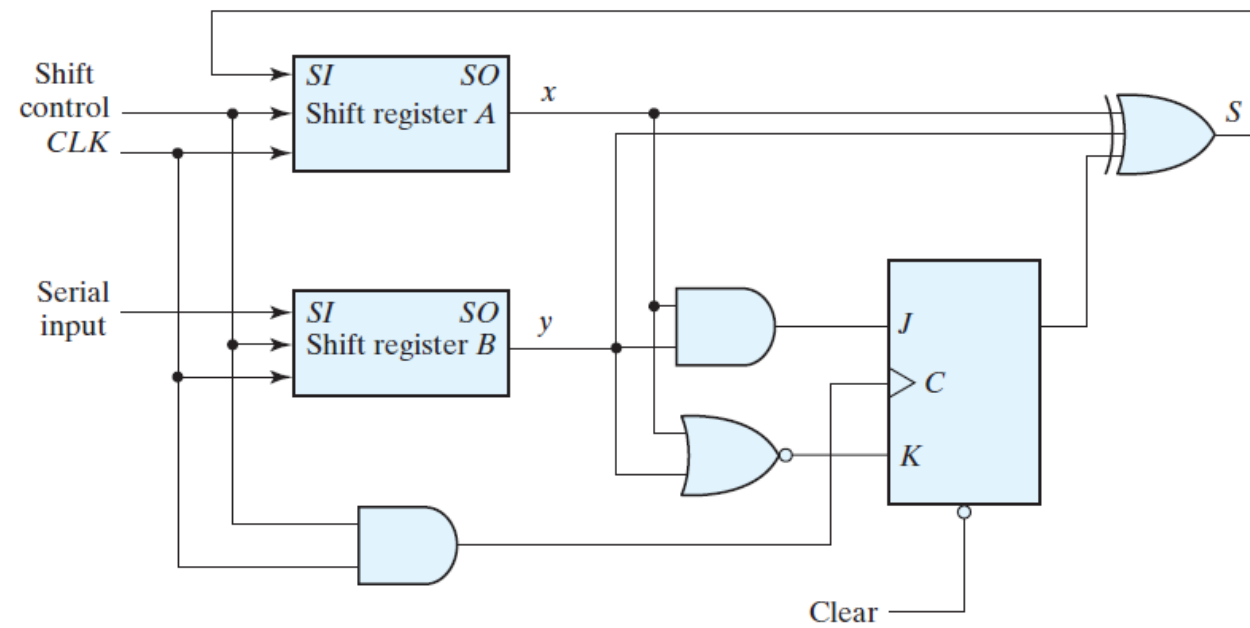
$$K_Q = x'y' = (x + y)'$$

$$S = x \oplus y \oplus Q$$

$$J_Q = xy$$

$$K_Q = x'y' = (x + y)'$$

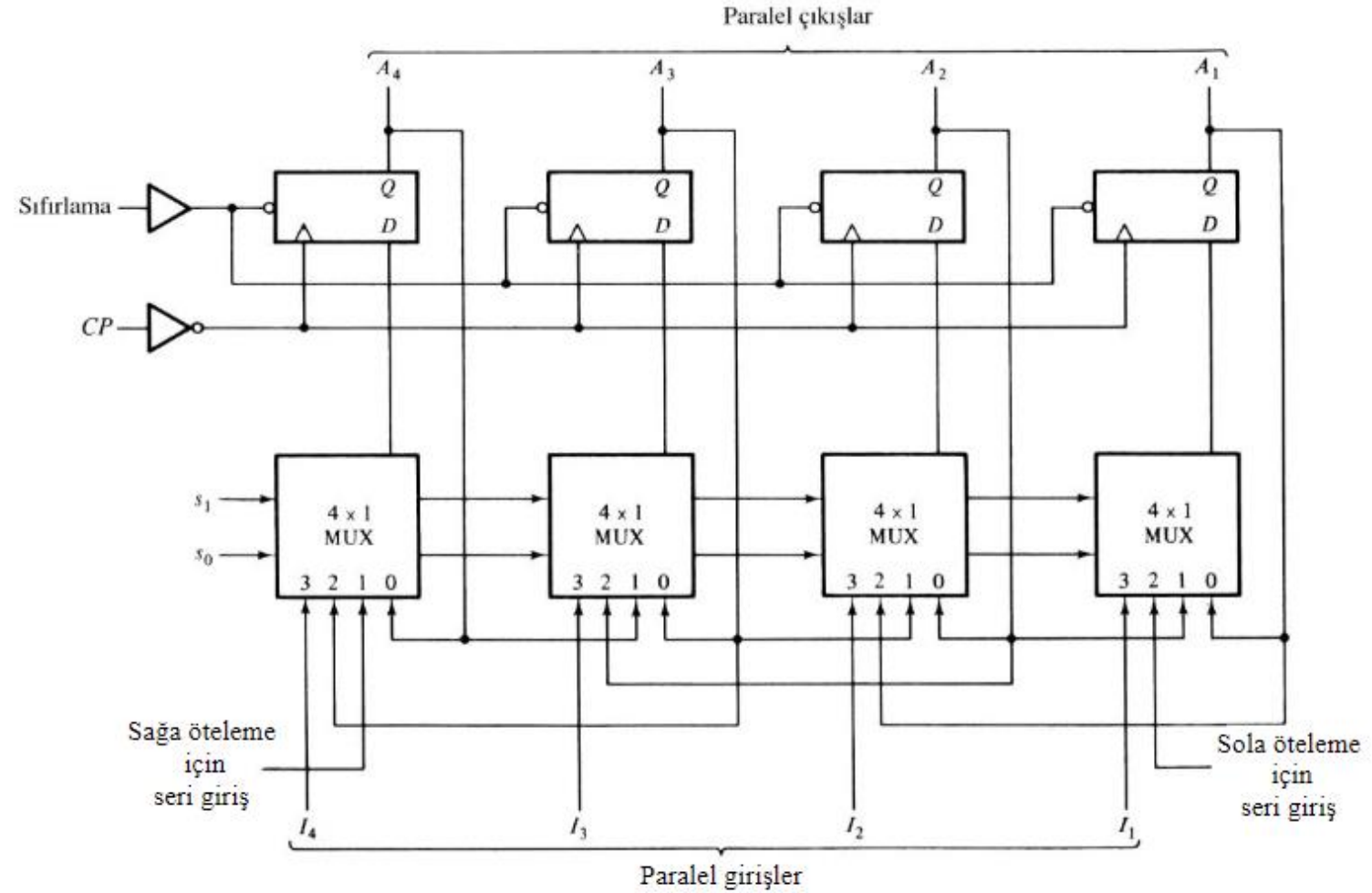
$$S = x \oplus y \oplus Q$$



**FIGURE 6.6**  
Second form of serial adder

# Çok Fonksiyonlu (Universal) Shift Register

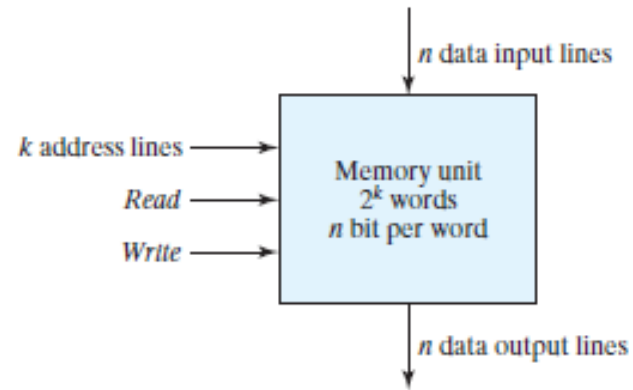
Mod Kontrol		Register İşlevi
S1	S0	
0	0	Değişim Yok
0	1	Sağa Öteleme
1	0	Sola Öteleme
1	1	Paralel Yükleme



ŞEKİL 7-9

Paralel yüklemeli 4 bitlik iki yönlü ötelemeli yazıcı

# Hafıza Ünitesinin Blok Diyagramı

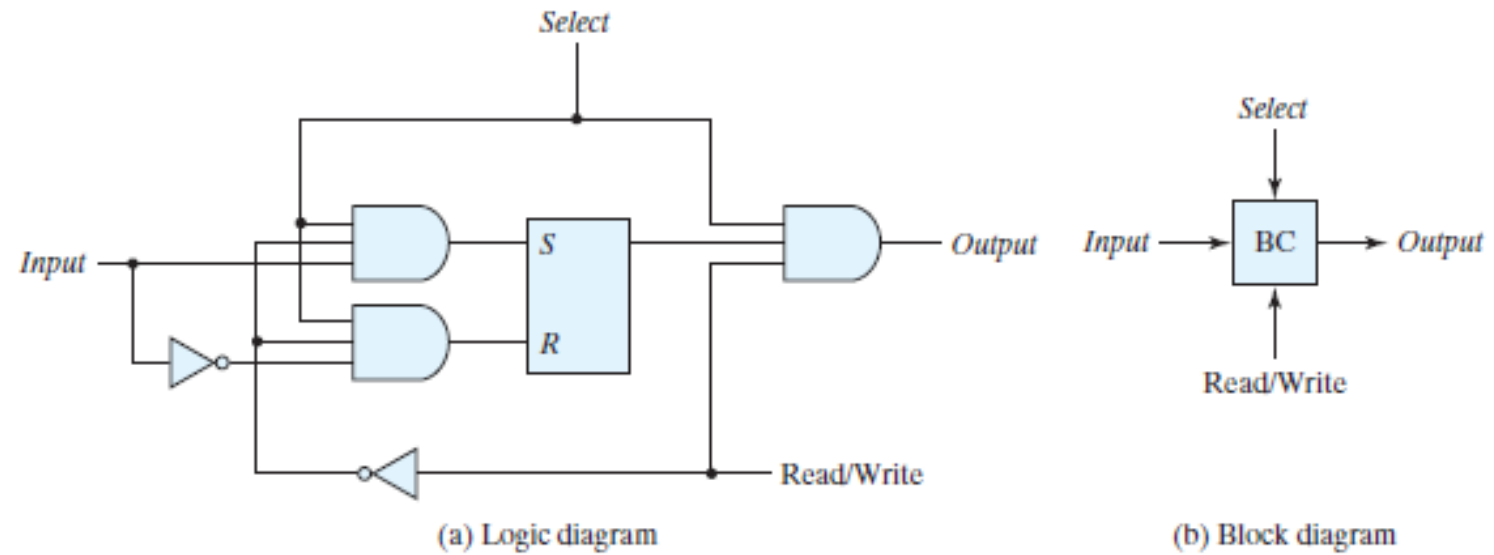


**FIGURE 7.2**  
Block diagram of a memory unit

Memory address		Memory content
Binary	Decimal	
000000000	0	1011010101011101
000000001	1	1010101110001001
000000010	2	0000110101000110
	⋮	⋮
111111101	1021	1001110100010100
111111110	1022	0000110100011110
111111111	1023	1101111000100101

**FIGURE 7.3**  
Contents of a  $1024 \times 16$  memory

# 1 Bitlik Hafıza Hücresi



**FIGURE 7.5**  
Memory cell

# 4x4 RAM Diyagramı

