

## ARDIŞIL DEVRELER (*Sequential Circuits*)

- Dersin ilk bölümünde **kombinezonsal (*combinational*) devreleri** inceledik. Bu tür devrelerde çıkışın değeri o andaki girişlerin değerlerine bağlıdır.  
Kombinezonsal devre : Çıkış = G (Giriş) G: çıkış fonksiyonu
- **Ardışıl (*sequential*) devrelerde** ise çıkış değeri, hem girişlerden gelen değerlere hem de devrenin bir önceki "durumuna" bağlıdır.  
Kombinezonsal devre : Çıkış = G (Giriş , Şimdiki Durum)  
Sonraki Durum = H (Giriş , Şimdiki Durum)

Durum bilgisini tutmak için bu devrelerde **bellek** elemanları bulunur.

Ardışıl devrelere örnek olarak bozuk parayla çalışan meşrubat makinelerindeki lojik devreler gösterilebilir.

Böyle bir sayısal ardışıl devre, ürünü vermek için sadece o anda atılan parayı değil, daha önce atılmış olan parayı da dikkate almalıdır.

### Ardışıl devrelerin türleri:

Ardışıl devreler iki gruba ayrılır:

#### A) Senkron (eş zamanlı) ardışıl devreler:

Bu devreler sadece belli zamanlarda durum değiştirebilirler.

Tüm bellek elemanları ortak bir saat işareti ile eş zamanlı (senkron) olarak tetiklenirler.

#### B) Asenkron ardışıl devreler:

Bu tür devreler her hangi bir zamanda girişlerdeki değişime bağlı olarak durum değiştirebilirler.

Bu derste günümüzde çok yaygın olarak kullanılan eş zamanlı devreler ele alınacaktır.

Örneğin mikroislemciler saatle tetiklenen eş zamanlı ardışıl devrelerdir.

### Sonlu Durumlu Makine (Finite State Machine- FSM) Modeli

Ardışıl devreler "sonlu durumlu makine" (Finite State Machine- FSM) modeli kullanılarak tasarlanırlar.

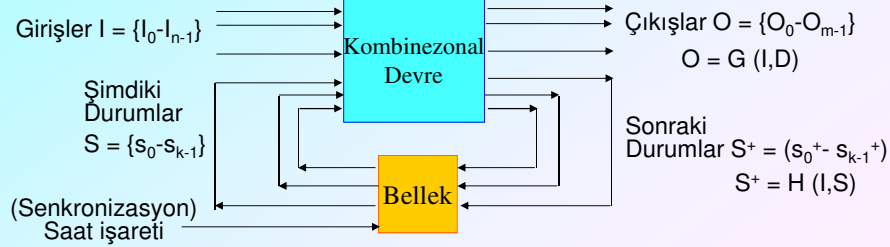
Bu **modelleme yöntemi**, bir çok başka sistemin tasarımında da kullanılır.

- Böyle bir makine ilk çalışmaya başladığında belli bir durumda bulunur.
- Gelen giriş değerine göre ve içinde bulunduğu duruma göre makine bir çıkış üretir.
- Gelen giriş değerine göre ve içinde bulunduğu duruma göre yeni bir duruma geçer.

Sonlu durumlu makineler, lojik devre olarak olarak gerçekleştirilirken iki kısımdan oluşturulurlar:

a) Lojik işlemleri yapan kombinezonel devre,

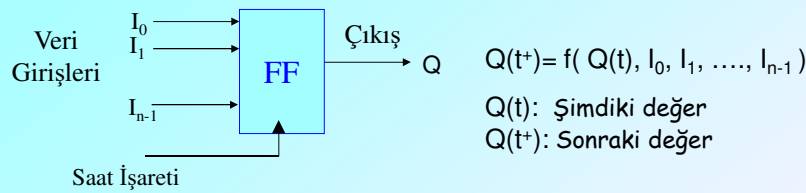
b) Durum bilgisini tutan bellek elemanları.



Bellek elemanları incelendikten sonra ardışıl devreler konusu tekrar ele alınacaktır.

### Veri Saklama (Bellek) Elemanları

'Flip-flop': Bir bitlik bellek elemanlarıdır. Çok girişli, bir çıkışlı lojik bir devre olarak tasarlanırlar.



**Q çıkışı** flip-flopun o anda içindeki ikili değeri (0,1) dışarı yansıtır. Bu değer aynı zamanda flip-flopun durum bilgisidir.

Q çıkışının alacağı yeni değer  $Q(t^+)$ , veri girişlerinin ve o andaki durumun  $Q(t)$  bir fonksiyonu olarak belirlenir.

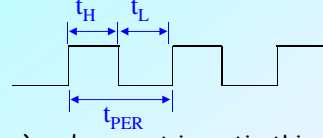
**Saat işareti**, veri girişlerindeki değerlerin ne zaman değerlendirileceğini, yani flip-flop'un ne zaman değer değiştireceğini belirten işarettir.

Sadece saat işaretinin etkin olduğu anlarda flip-flop'un içeriği yukarıdaki fonksiyona göre belirlenerek değiştirilir.

Saat işareti etkin değilse, veri girişleri değişse bile flip-flop bir önceki içeriğini korur.

**Saat (Clock) İşareti:**

Sayısal sistemlerdeki elemanların eş zamanlı (senkronize) çalışmasını sağlayan dikkörtgen dalga şeklinde bir işarettir.



Saat işaretiyle denetlenen elemanlar (örneğin flip-flop) sadece saat işareti etkin olunca işlem yaparlar. Onun dışında eski durumların korurlar.

Saat işaretinin kullanılması açısından elemanlar ikiye ayrılır.

a) Düzey tetiklemeli elemanlar, b) Kenar tetiklemeli elemanlar

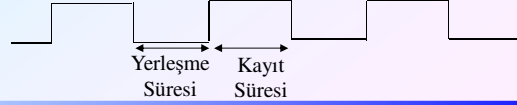
**Düzey tetiklemeli elemanlar:** Saat işaretinin bir düzeyini (pozitif lojikte "1" düzeyini) etkin düzey olarak kabul ederler.

Bu elemanlar saat işareti "1" düzeyindeyken işlem yaparak durumlarını ve çıkışlarını değiştirirler; saat işareti "0" düzeyindeyken eski durumlarını korurlar.

Saat işaretinin "1" düzeyindeyken girişler işleme sokulduğundan, bu süre boyunca giriş değerleri sabit tutulmalıdır.

Aksi durumda ardışıl elemanın çıkışının alacağı değer belirsiz olur. Bu süreye **kayıt süresi** denir.

Saat işaretinin "0" olduğu sürede ise girişler değiştirilebilir. Bu süreye **yerleşme süresi** denir.

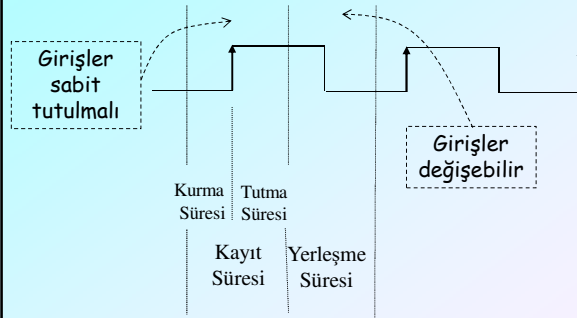
**Kenar tetiklemeli elemanlar:**

Saat işaretinin bir kenarını (pozitif lojikte çıkan kenar) etkin kenar olarak kabul ederler.

Pozitif kenar tetiklemeli elemanlar saat işareti 0→1 geçişi yapınca (çıkan kenar) işlem yaparak durumlarını ve çıkışlarını değiştirirler; saat işareti geçiş yapmazsa eski durumlarını korurlar.

Negatif lojikte ise işlemler 1→0 geçişinde (inen kenar) yapılır.

Saat işaretinin 0→1 geçişi yaparken girişler işleme sokulduğundan, bu kenardan belli bir süre önce ve sonra giriş değerleri sabit tutulmalıdır. Aksi durumda ardışıl elemanın çıkışının alacağı değer belirsiz olur.



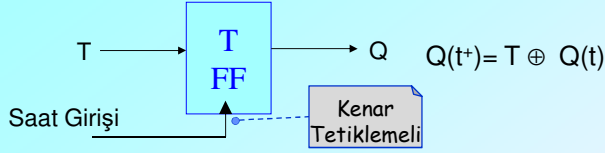
Kurma süresi "Set-up time"

Tutma süresi "Hold time"

Kayıt süresi, kurma ve tutma sürelerinin toplamından oluşur. Ardışıl devrenin sağlıklı çalışması için bu süre boyunca girişlerin sabit kalması gerekir.

**Örnek: Kenar tetiklemeli T Flip-Flopu (Toggle Flip-flop)**

Bellek elemanlarının ve flip-flopların ayrıntılarına geçmeden önce, bu bölümde örnek olarak bir T flip-flopu gösterilmiştir.



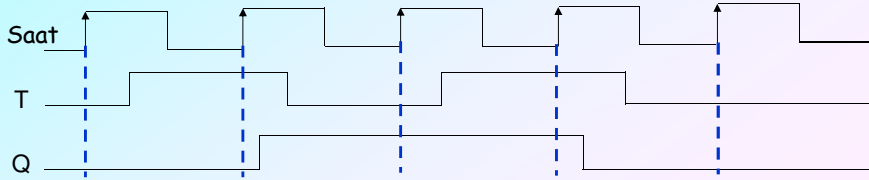
T flip-flopunun çıkışının (içeriği) alacağı değer  $Q(t^+)$ , o andaki değer  $Q(t)$  ile girişinin (T) YA DA işlemine sokulmasıyla bulunur.

Buna göre girişine  $T=0$  uygulanırsa flip-flopun içeriği değişmez.

Çünkü:  $0 \oplus x = x$

Flip-flopun girişine  $T=1$  uygulanırsa flip-flopun içeriği tümlenir.

Çünkü:  $1 \oplus x = x'$

**Geri Beslemeli Bağlantılar (Feedback Connections)**

Bellek elemanları oluşturabilmek için devrelerde geri besleme bağlantılarına gerek duyulur.

**Geri besleme**nin anlamı; devrenin bazı çıkışlarının aynı devrenin bazı girişlerine bağlanmasıdır.

Örnek:

Belli bir anda tümleyicinin A girişi "0" ise bu "0" değeri devrenin Z çıkışının "1" değeri almasına neden olacaktır.

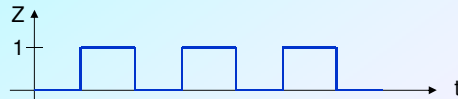
Bu "1" değeri tekrar devrenin girişine uygulandığından (**geri besleme**) yayılma gecikmesi sonrası tümleyicinin Z çıkışı "0" olacaktır.

Bu "0" değeri A girişine uygulanacağından Z çıkışı tekrar "1", olacak ve bu değişim bu şekilde devam edecektir.

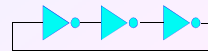
Aşağıdaki şekilde gösterildiği gibi tümleyicinin Z çıkışı sürekli "0", "1" değerleri arasında değişecek ve hiç bir zaman kararlı bir duruma ulaşmayacaktır.

Bu **kararsız (unstable)** devre bellek olarak kullanılamaz.

Devrenin değişim (osilasyon) periyodu tümleyicinin gecikme süresine bağlıdır.



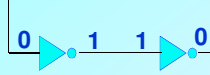
Başka bir **kararsız (unstable)** devre:



## İki Tümlleme Elemanı ile Oluşturulan geri Beslemeli Devre İki Kararlı Devre (Bistable Circuit)

Bu devrede iki tümlleme elemanı kullanılmıştır.  
Bu durumda devrenin iki kararlı durum olur (*bistable*).

Kararlı durum 1:



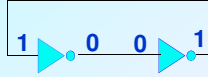
Birinci tümlleyicinin girişi "0" ise çıkışı "1" olur.

Bu durumda ikinci tümlleyicinin girişi "1" olur, çıkışı da "0" olur.

Bu "0" değeri ilk tümlleyicinin girişine uygulanır (**geri besleme**). Bu girişte zaten "0" değeri olduğundan devrenin durumunda bir değişiklik olmaz (kararlı).

Devre kararlı bir durumdadır.

Kararlı durum 2:



Eğer birinci tümlleyicinin girişi "1" ise ikinci tümlleyicinin girişi "0" olur ve şekilde gösterildiği gibi ikinci kararlı durum oluşur.

## İki Kararlı Durumlu (Bistable) Devre (devamı)

Yansı 6.9'da gösterilen iki kararlı devre yandaki gibi de çizilebilir.

Bu devre iki **kararlı** durumdan birinde bulunur.

**Durum 1:**  $V_{in1} = 1, V_{out1} = V_{in2} = 0, V_{out2} = 1$

**Durum 2:**  $V_{in1} = 0, V_{out1} = V_{in2} = 1, V_{out2} = 0$

Bu elemanın iki kararlı durumu vardır.  $Q=0$  ve  $Q=1$

$Q_L$  tümlleyen çıkıştır  $Q_L = \bar{Q}$ .

Bunlar belleklerde olması gereken özelliklerdir.

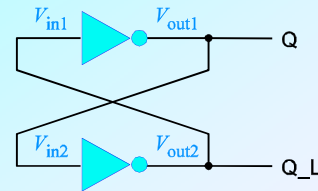
Ancak bu devrenin girişi yoktur.

Girişi olmadığından elemanın durumunu dışarıdan denetlemek (değiştirmek) mümkün değildir. İlk gerilim verildiğinde eleman rastgele bir duruma geçer.

Bu nedenle bu devre bellek olarak kullanılamaz.

Bellek elemanına aşağıdaki iki özelliğe sahip olması gerekir:

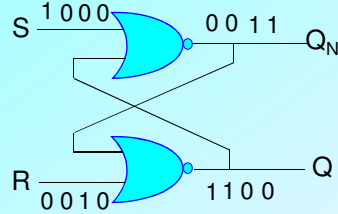
1. İki kararlı durum
2. Durumları değiştirmeyi veya eski durumu korumayı sağlayan denetim girişleri



**S-R (Set-Reset) Bilgi Saklama Elemanı**

İki adet TVEYA veya iki adet TVE bağlacı ile oluşturulabilen bir bitlik saklama elemanıdır.

Tüm flip-floplar, bu temel saklama elemanına yapılan eklemeler ile oluşturulabilir.

**TVEYA ile oluşturulan S-R Saklama Elemanı:**

S: Set (Birleme)  
R: Reset (Sıfırlama)  
Q: Çıkış (Durum)  
Q<sub>N</sub>: Tümlenmiş Çıkış (Q')

**Hatırlatma:** Bir TVEYA bağlacının bir girişi "1" olduğunda çıkışı mutlaka "0" olur

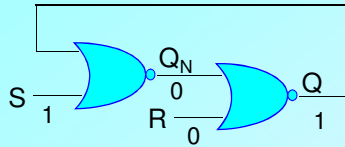
S	R	Q	Q <sub>N</sub>	
1	0	1	0	
0	0	1	0	S=1, R=0'dan sonra
0	1	0	1	
0	0	0	1	S=0, R=1'den sonra
1	1	0	0	Yasaklı girişler

- S girişi saklama elemanına "1" yazmak için, R girişi de "0" yazmak için kullanılır.
- Her iki girişi de "0" olduğunda SR elemanı bir önceki durumunu korur.
- Girişlerin her ikisine birden "1" verilmez.

**S-R (Set-Reset) Tutucunun Durum Değişimleri**

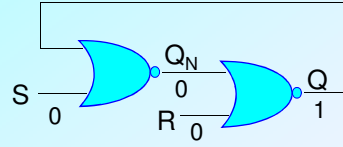
S-R tutucunun durum değişimlerini göstermek için devre aşağıdaki gibi çizilebilir

S=1, R=0 :



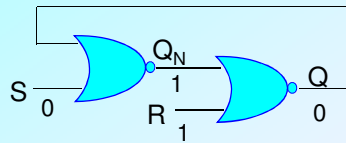
Eğer S = 1, R = 0 ise Q<sub>N</sub> "0" olur. İkinci bağlacın her iki girişi de "0" olduğundan, Q çıkışı "1" olur.

S=0, R=0 :



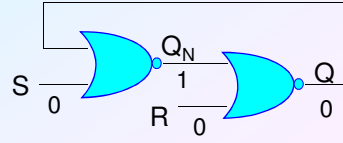
Eğer S "0" olarak değişirse tutucu o andaki durumunu (1) korur, çünkü Q = 1 değeri geri besleme ile ilk bağlacın girişine uygulandığından Q<sub>N</sub> "0" olarak kalır.

S=0, R=1 :



Eğer R "1" olarak değiştirilirse Q "0" olur ve Q<sub>N</sub> "1" olarak değişir.

S=0, R=0



R tekrar "0" yapılırsa tutucu o andaki durumunu (0) korur (değiştirmez).

**S-R Tutucunun Doğruluk Tablosu ve Karakteristik Denklemi:**

Q çıkışının bir sonraki değeri (sonraki durumu)  $Q(t+1)$ , girişlere ve saklama elemanının o anki çıkışına (durumuna)  $Q(t)$  bağlıdır.

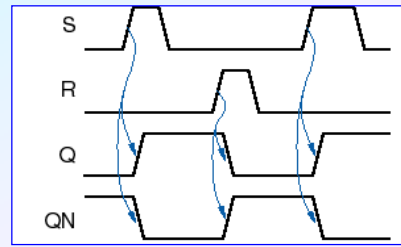
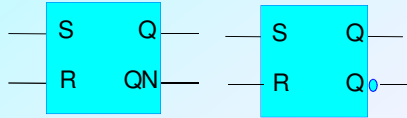
**Doğruluk Tablosu:**

Q(t)	S	R	Q(t+1)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	Yasak ( $\Phi$ )
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	Yasak ( $\Phi$ )

		S			
		00	01	11	10
Q(t)	0			$\Phi$	1
	1	1		$\Phi$	1

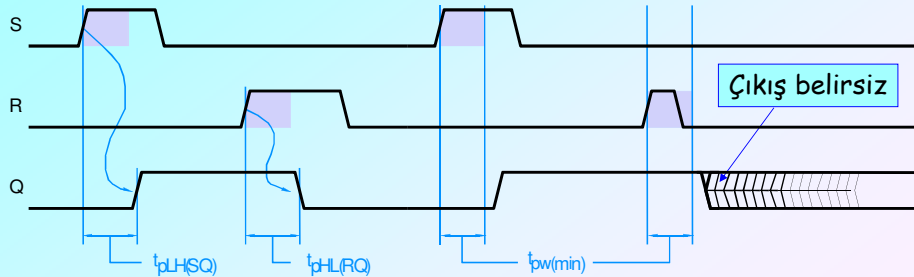
**Karakteristik Denklem:**

$$Q(t+1) = S + Q(t)R', \quad SR=0$$

**S-R Tutucunun Blok Diyagramları:****S-R Tutucunun Zamanlama Özellikleri:**

Elemanın içindeki yayılma (propagasyon) gecikmesinden dolayı S veya R girişlerindeki değişimlerin etkisi belli bir süre geçtikten sonra çıkışta etkili olur. Bu süre boyunca girişler sabit kalmalıdır. Aksi durumda çıkışın alacağı değer belirsiz olur.

S (veya R) girişindeki darbenin uzunluğu en az gecikme kadar olmalıdır.



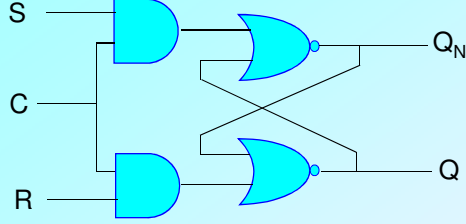
$t_{pLH(SQ)}$ : S değiştiğinde çıkışın 0-1 değişim yapması için geçen süre.

$t_{pHL(RQ)}$ : R değiştiğinde çıkışın 1-0 değişim yapması için geçen süre.

$t_{pw(min)}$ : Girişlerin sabit kalması gereken en küçük süre.

**İzin Girişli S-R Bilgi Saklama Elemanı (Tutucu)**

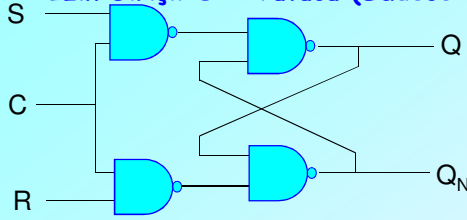
S ve R girişlerinin sadece istenen (izin verilen) zamanlarda etkili olabilmesi için bu girişlere VE kapıları bağlanır.



S: Set (Birleme)  
R: Reset (Sıfırlama)  
Q: Çıkış (Durum)  
QN: Tümlleyen Çıkış (Q')  
C: İzin girişi

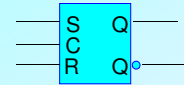
Ancak C=1 olduğunda elemanın içeriği değiştirilebilir. C=0 olduğunda elemanın içeriği korunur.

S  
R  
C  
Q  
QN

**İzin Girişli S-R Tutucu (Sadece TVE kapıları ile Tasarım)**

Yansı 6.15'te, TVEYA ve VE kapıları kullanılarak gerçekleştirilmiş olan izin girişli S-R tutucu sadece TVE bağlaçları kullanılarak yandaki şekilde gerçekleştirilebilir.

C	S	R	Q(t+1)
0	X	X	Q(t)
1	1	0	1
1	0	1	0
1	0	0	Q(t)
1	1	1	Yasak



Tutucunu girişine yasaklı değerler (SR=11) uygulanırsa Q ve Q' çıkışlarının ikisi de 1 olur. Bu durumdayken izin kaldırılırsa tutucunun değeri belirsiz olur.





**Tutucu (Latch), Flip-flop Farkı:**

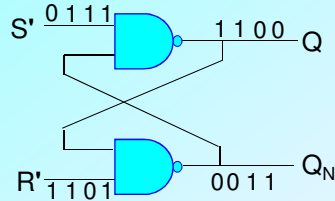
Buraya kadar tanıtılan S-R saklama elemanının bir saat işareti ile tetiklenmesi söz konusu değildir. İzin girişi etkin olduğu sürece bu elemanın içeriği değiştirilebilir.

Bu tip elemanlara **tutucu (latch)** denir.

Saat işareti ile tetiklenen saklama elemanlarına ise **flip-flop** denir.

**TVE Bağlı S'-R' Tutucu (Latch)**

S-R veri saklama elemanları TVEYA kapıları yerine TVE kapıları kullanılarak da tasarlanabilir. Bu elemanlar **S'-R'** tutucu olarak adlandırılır.



S': Set (Birleme) Tümlenyeni  
R': Reset (Sıfırlama) Tümlenyeni  
Q: Çıkış (Durum)  
Q<sub>N</sub>: Tümlen Çıkış (Q')

6.15 ve  
6.16'daki  
elemandan  
farklıdır.

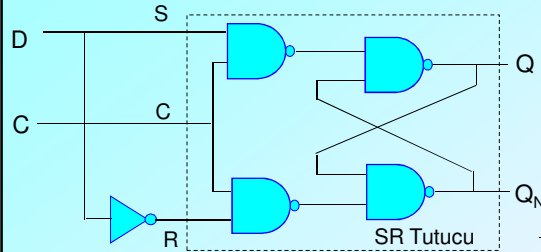
S'	R'	Q	Q <sub>N</sub>	
0	1	1	0	
1	1	1	0	S'=0, R'=1'den sonra
1	0	0	1	
1	1	0	1	S'=1, R'=0'dan sonra
0	0	1	1	Yasaklı girişler

**Hatırlatma:** Bir TVE bağlacının bir girişi "0" olduğunda çıkışı mutlaka "1" olur.

**D tipi Tutucu (Delay Latch)**

S-R tutucunun yapısına bazı eklemeler yaparak değişik fonksiyonlara sahip başka tipte tutucular elde edilebilir.

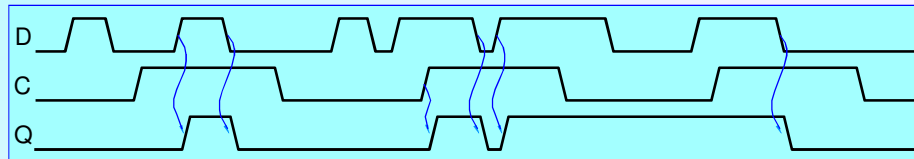
S-R tutucunun S ve R girişleri bir tümlenme kapısı ile birleştirilirse D tipi tutucu elde edilir.



C=1 olduğu sürece D'den gelen değer tutucuya yazılır.  
C=0 olduğu sürece tutucu bir önceki değerini korur.

**Karakteristik Denklem:**  $Q(t+1)=D$

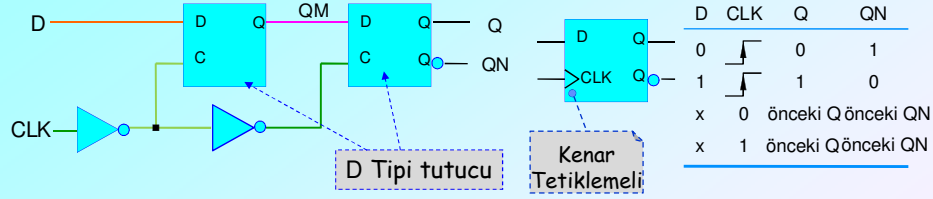
C	D	Q(t+)	Q <sub>N</sub> (t+)
1	0	0	1
1	1	1	0
0	X	Q(t)	Q <sub>N</sub> (t)



### Pozitif (çıkan) kenar tetiklemeli D tipi Flip-flop

Tutucular izin girişleri etkin olduğu sürece veri girişlerindeki değerlere göre içeriklerini değiştirirler.

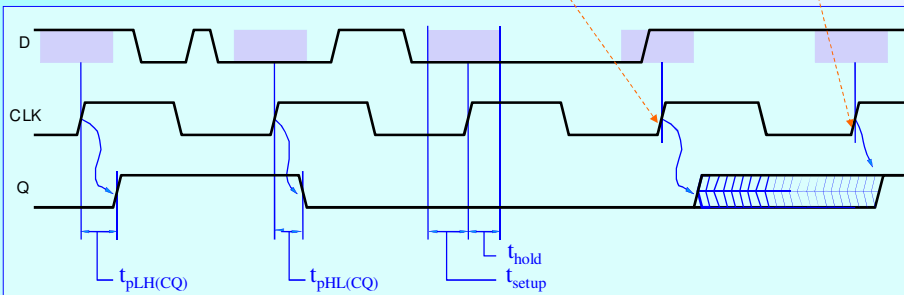
Flip-floplar ise ancak bir saat işareti etkin olduğunda veri girişlerindeki değerlerden etkilenirler.



### Pozitif kenar tetiklemeli D tipi flip-flopunun zamanlama özellikleri

Kurma süresine (*setup time*) uyulmadığı için çıkış belirsizdir.

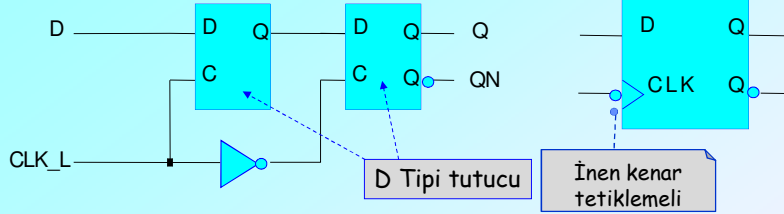
Çıkış tekrar belirli bir değer (örnekte "1") alır.



- $t_{pLH(CQ)}$ : Etkin kenardan sonra çıkışın 0-1 geçişi yapması için geçen süre.
- $t_{pHL(CQ)}$ : Etkin kenardan sonra çıkışın 1-0 geçişi yapması için geçen süre.
- $t_{setup}$ : Etkin kenardan önce girişin sabit kalması gereken süre.
- $t_{hold}$ : Etkin kenardan sonra girişin sabit kalması gereken süre.

### Negatif (inen) kenar tetiklemeli D tipi Flip-flop

Saat işaretinin inen kenarlarında D girişindeki veri flip-flopa yazılır.

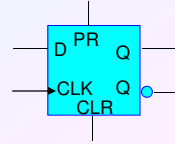


D	CLK_L	Q	QN
0	↓	0	1
1	↓	1	0
x	0	eski Q	eski QN
x	1	eski Q	eski QN

Flip-flopla, özellikle başlangıç değeri yazabilmek için saat işaretinden bağımsız olarak (**asenكرون**) çalışan girişler de bulunabilir.

Flip-flopa 1 yazmak için PR (*Preset*), 0 yazmak için CLR (*Clear*) girişi kullanılır.

Asenkron girişler, saat işareti etkin olmasa da flip-flopu etkilerler.

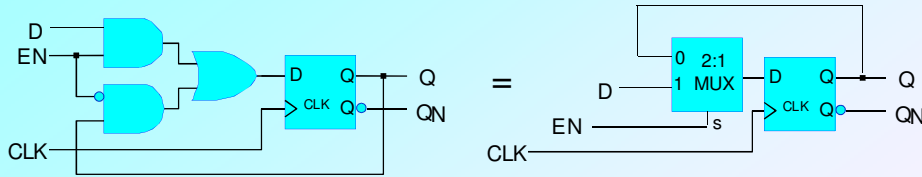


### Kenar tetiklemeli ve izin girişli D tipi Flip-flop

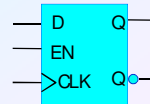
Flip-floplarda da izin girişi (EN) (*enable*) bulunabilir.

Flip-flopun içeriğinin değiştirilebilmesi için izin girişi EN etkin olmalıdır.

Aksi durumda flip-flopun içeriği korunur.

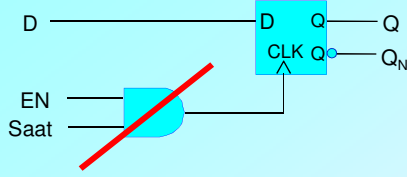


D	EN	CLK	Q	QN
0	1	↑	0	1
1	1	↑	1	0
x	0	↑	eski Q	eski QN
x	x	0	eski Q	eski QN
x	x	1	eski Q	eski QN



**Kenar tetiklemeli ve izin giriшли D tipi Flip-flopu (devamı)**

Flip flopun saat girişine (CLK) bir VE kapısı bağlayarak daha basit bir devre elde edebilir miyiz?



Bu uygun bir çözüm değildir.

Bu devrede aşağıdaki sorunlar vardır.

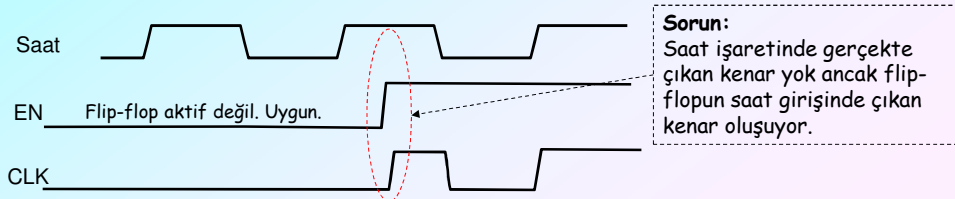
1. Flip flopun saat girişindeki gecikmeyi arttırmak iyi bir fikir değildir.

Bu durum senkronizasyon sorunlarına neden olabilir.

2. Eğer Saat=1 ve EN=0 ise CLK=0 olur ve flip flop çalışmaz (uygun bir durum).

Ancak Saat=1 iken EN=1 olursa flip-flopun CLK girişinde bir aktif kenar oluşur ve flip flop D girişindeki değeri işler.

Halbuki saat işareti o anda 1'dir; 0-1 geçişi yapılmamıştır.

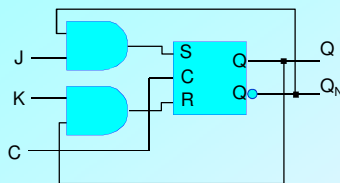
**J-K Tutucu**

J-K Tutucu, S-R tutucunun ve T tutucunun özelliklerini kendi üstünde toplar.

Sadece J veya K girişine "1" değeri uygulanırsa bu eleman S-R tutucu gibi davranır.

Eğer sadece J = 1 ise tutucu birleşir (set) Q = 1; eğer sadece K = 1 ise tutucu sıfırlanır (reset) Q = 0.

J=1, K=1 girişi uygulanması durumunda eleman T tipi tutucu gibi davranır ve içeriği tüklenir (toggle).

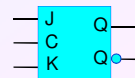


J	K	C	Q	QN
x	x	0	eski Q	eski QN
0	0	1	eski Q	eski QN
0	1	1	0	1
1	0	1	1	0
1	1	1	eski QN	eski Q

C=1	J	K	Q(t)	S	R	Q(t+1)	İşlem
	0	0	0	0	0	0	Değişim yok
	0	0	1	0	0	1	
	0	1	0	0	0	0	Sıfırlama (Reset)
	0	1	1	0	1	0	
	1	0	0	1	0	1	Birleme (Set)
	1	0	1	0	0	1	
	1	1	0	1	0	1	Tümleme
	1	1	1	0	1	0	

**Karakteristik Denklem:**

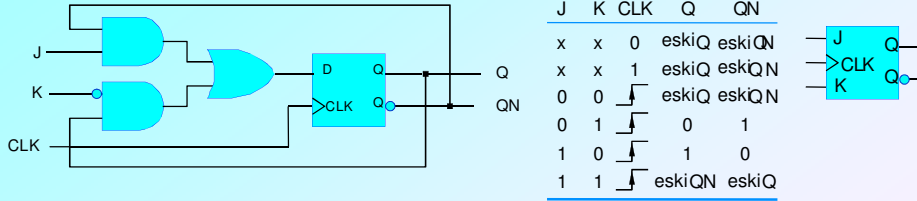
$$Q(t+1) = J \cdot Q(t)' + K' \cdot Q(t)$$



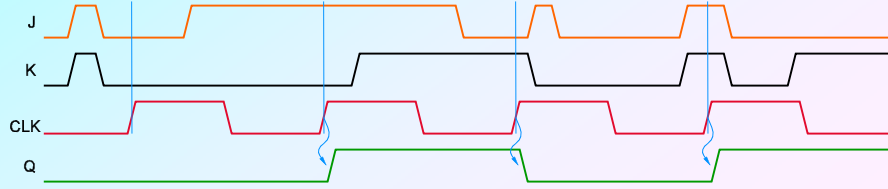
### Kenar Tetiklemeli J-K Flip-Flop

Kenar tetiklemeli bir D flip-flop ve lojik bağlaçlar kullanılarak kenar tetiklemeli bir JK flip-flop tasarlanabilir.

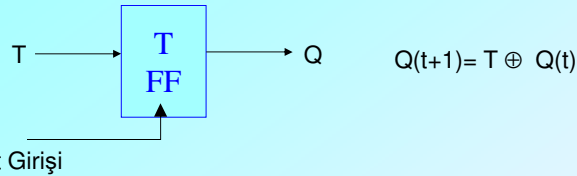
Bu flip-flopta JK girişleri sadece saat işaretinin etkin geçişlerinde (kenarlarında) değerlendirilir.



$$Q(t+1) = J \cdot Q(t)' + K' \cdot Q(t)$$



### Kenar tetiklemeli T Flip-Flop (Toggle Flip-flop)



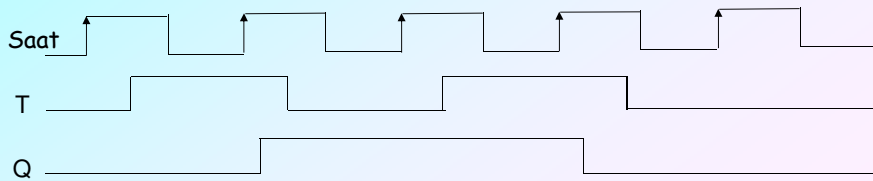
T flip-flopunun çıkışının (içeriği) bir saat darbesi sonra alacağı değer  $Q(t+1)$ , şimdiki değer  $Q(t)$  ile girişinin (T) YA DA işlemine sokulmasıyla bulunur.

Buna göre girişine  $T=0$  uygulanırsa flip-flopun içeriği değişmez.

Çünkü:  $0 \oplus x = x$

Flip-flopun girişine  $T=1$  uygulanırsa flip-flopun içeriği tümlenir.

Çünkü:  $1 \oplus x = x'$



**Flip-Flop ve Tutucuları Karakteristik Denklemleri:**

Flip-Flop ve tutucuların işlevsel davranışları **karakteristik denklemleri** ile tarif edilir.

Bu denklemler; elemanın sonraki durumunun değerini, şimdiki durumu ve giriş değeri cinsinden nasıl hesaplanacağını gösterirler.

**Characteristic equations for the flip-flops:**

$$\text{S-R FF: } Q(t+1) = S + R' \cdot Q(t), \quad SR=0$$

$$\text{J-K FF: } Q(t+1) = J \cdot Q(t)' + K' \cdot Q(t)$$

$$\text{D FF: } Q(t+1) = D$$

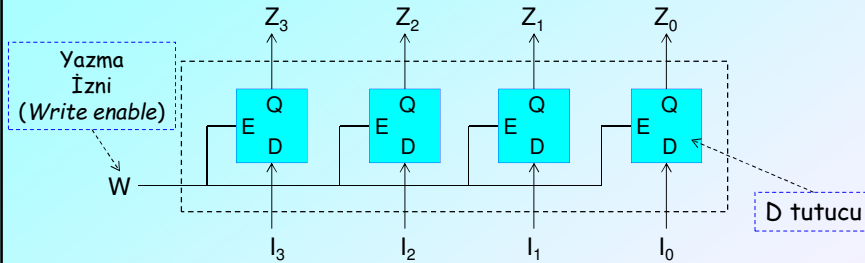
$$\text{T FF: } Q(t+1) = T \oplus Q(t)$$

**Saklayıcılar (Registers)**

Saklayıcılar n bitlik ikili sayıları saklayabilen (tutabilen) bellek elemanlarıdır.

**Örnek:** 4 bit paralel saklayıcı

D tutucular (latch) kullanılmıştır.

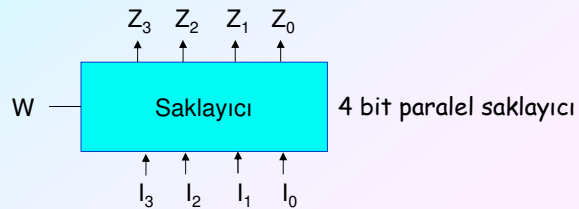


W=0 ise the saklayıcı o anki durumunu (içeriğini) korur.

W=1 ise girişteki veri saklayıcıya yazılır.

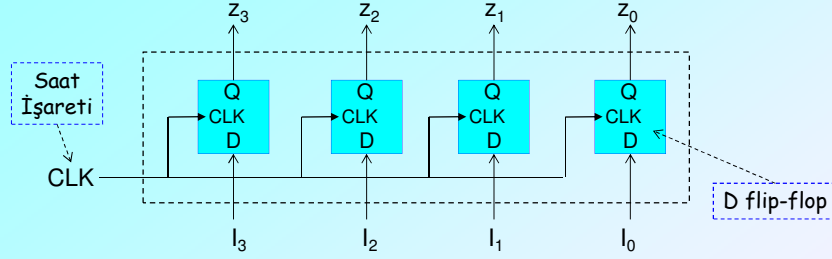
$$W = 0, Z(t+1) = Z(t)$$

$$W = 1, Z(t+1) = I(t)$$



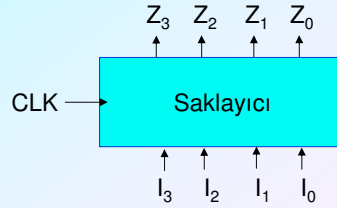
**Örnek: Saat ile tetiklenen 4 bit paralel saklayıcı**

D flip-flopları kullanılmıştır.



Saat işareti etkin olduğunda girişteki veri saklayıcıya yazılır.

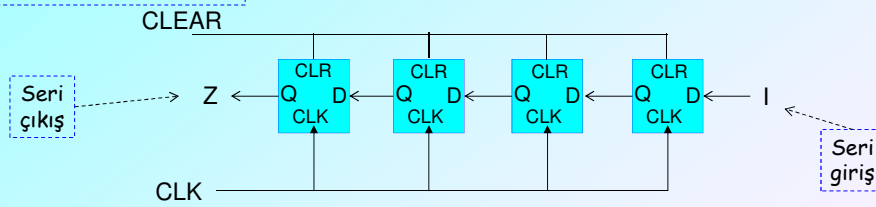
Saat ile tetiklenen 4bit paralel saklayıcı:

Not: Eğer tasarımda izin girişli (E) D flip-flopları kullanılırsa elde edilen saklayıcı hem saat işareti (CLK) hem de yazma izni (*write enable -W*) içerir.**Seri Ötelemeli Saklayıcılar (Serial Shift Registers)**

Seri ötelemeli saklayıcılar ikili sayıları saklayabilirler ve saat işaretinin her etkin geçişinde bir bit sola veya sağa öteleyebilirler.

**Örnek:** 4-bit seri sola ötelemeli saklayıcı

Tüm bitleri sıfırlamak için

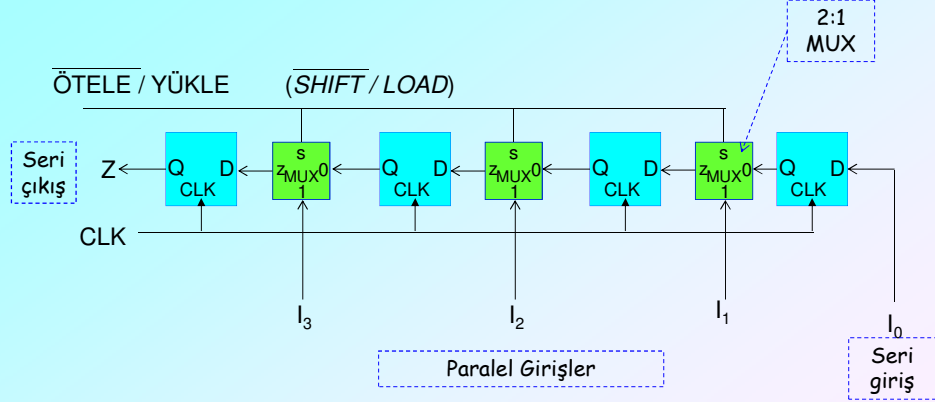


Saat işaretinin her etkin geçişinde veri bir bit sola ötelenir.

Flip-flopların bağlantıları (giriş/çıkış) yönleri değiştirilerek sağa ötelemeli saklayıcı gerçekleştirilebilir.

Yukarıdaki tasarımda saklayıcıya başlangıç değeri (sıfırdan başka) yüklemek mümkün değildir.

**Örnek:** Paralel yüklenebilen 4-bit seri sola ötelemeli saklayıcı  
Bu tasarımda saklayıcıya bir başlangıç değeri paralel olarak yüklenebilir.



SHIFT / LOAD = 0 : ötele

SHIFT / LOAD = 1 : yükle