

Lojik Tasarım

BIL-304: Bilgisayar Mimarisi

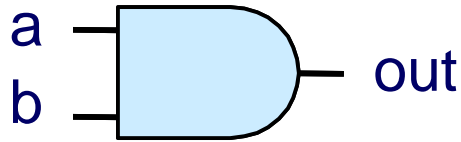
**Dersi veren öğretim üyesi:
Dr. Öğr. Üyesi Fatih Gökçe**

Ders kitabına ait sunum dosyalarından adapte edilmiştir: <http://csapp.cs.cmu.edu/>

Adapted from slides of the textbook: <http://csapp.cs.cmu.edu/>

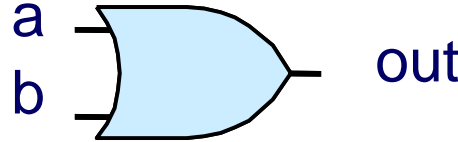
Lojik Kapılarla Hesaplama

VE
And



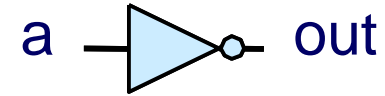
$$\text{out} = a \ \&\& \ b$$

VEYA
Or



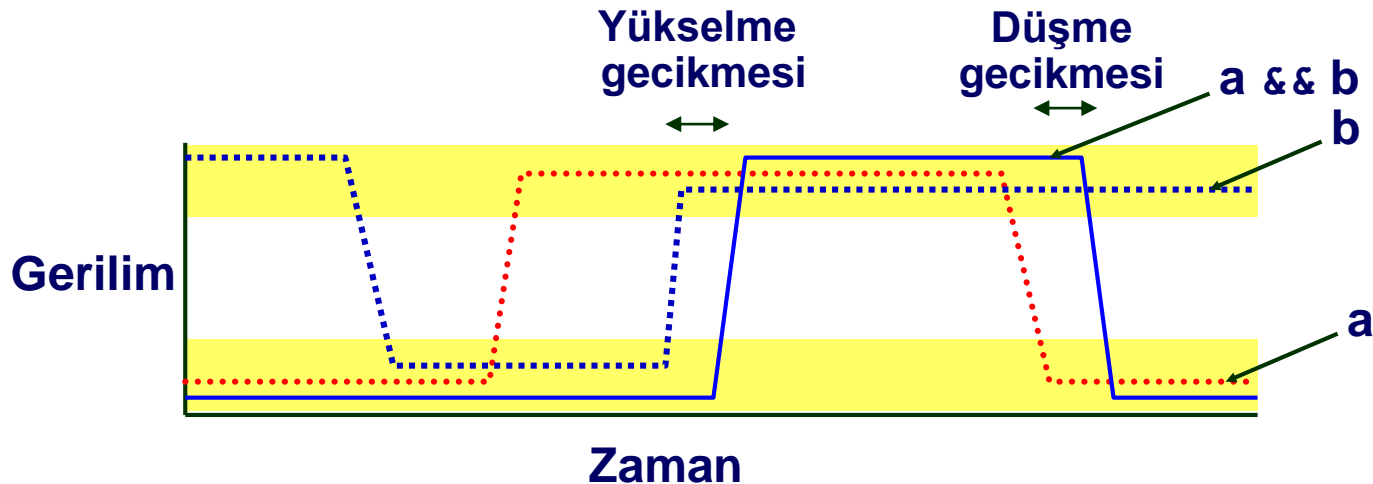
$$\text{out} = a \ || \ b$$

DEĞİL
Not

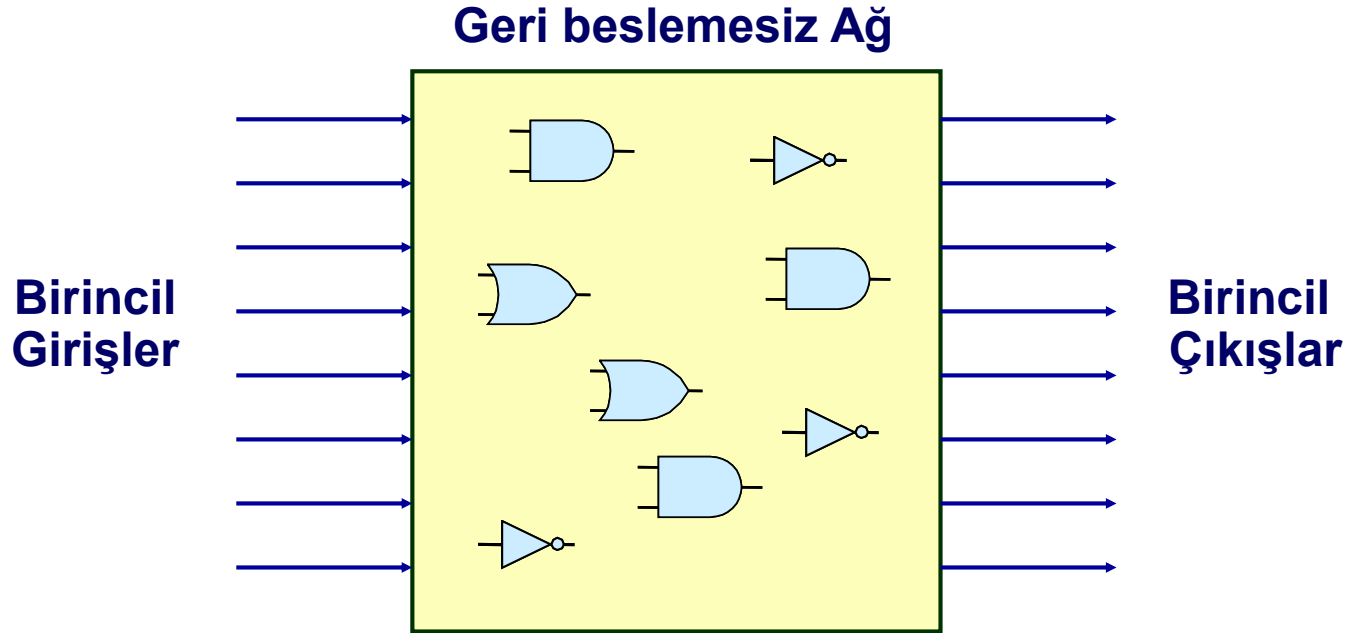


$$\text{out} = !a$$

- Çıktılar girişlerin Boolean fonksiyonlarıdır
- Girişlerdeki değişikliklere sürekli olarak tepki verirler
 - Bir miktar, küçük bir gecikmeyle



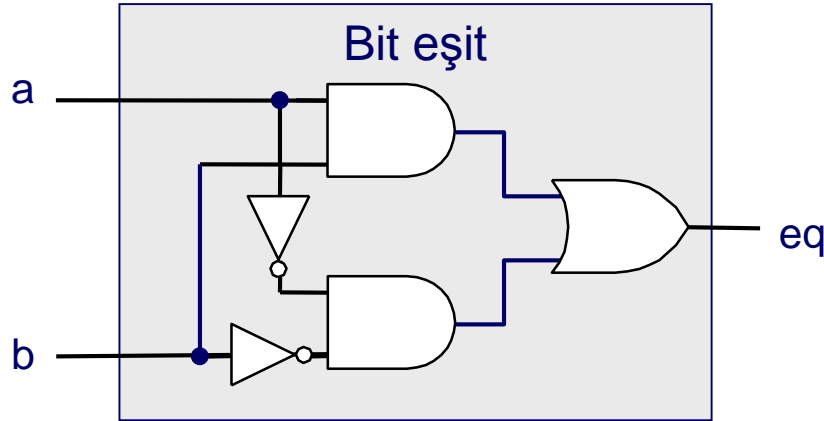
Kombinasyonel Devreler



Lojik Kapıların Geri Besleme Bulunmayan Bir Ağı

- Girişlerdeki deęişikliklere sürekli olarak tepki verir
- Birincil çıkışlar (bir miktar gecikmeden sonra) birincil girişlerin Boolean fonksiyon deęerlerini alır

Bit Eşitliği



HCL ifadesi

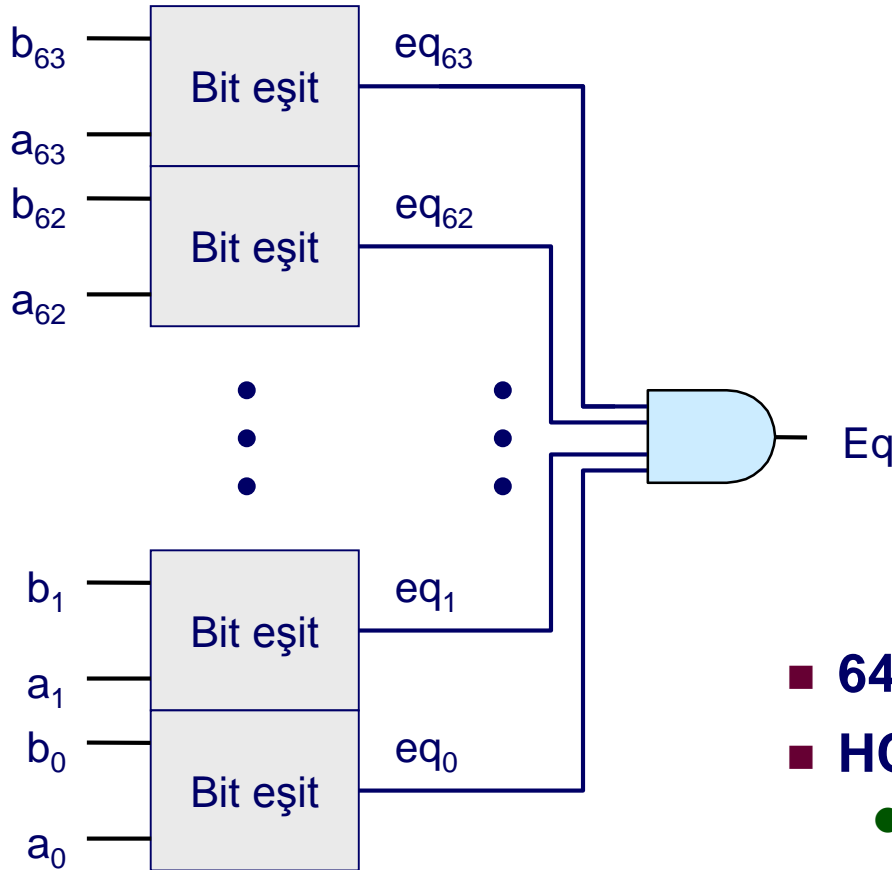
```
bool eq = (a&&b) || (!a&&!b)
```

- Eğer a ve b eşitse 1 üretir

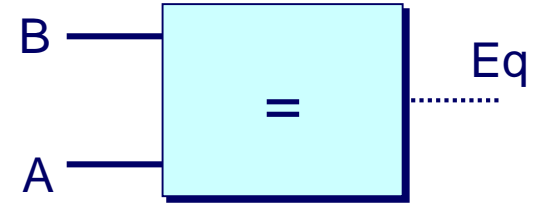
Donanım Kontrol Dili (Hardware Control Language (HCL))

- Oldukça basit bir donanım tanımlama dili (hardware description language)
 - Boolean işlemler C'deki lojik işlemlere benzer bir sözdizim kuralına sahiptir
- İşlemciye ait kontrol lojik kısmını tanımlamak için kullanacağız

Sözcük (Word) Eşitliği



Sözcük-seviyesinde ifadesi

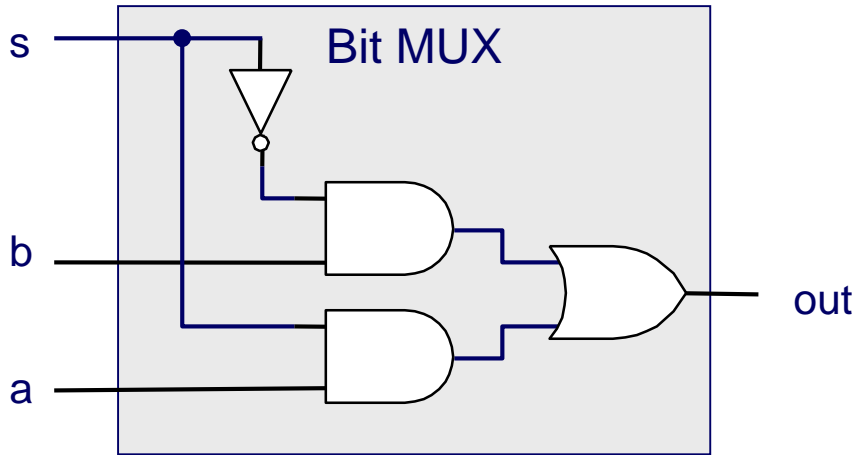


HCL İfadesi

```
bool Eq = (A == B)
```

- 64-bit sözcük boyutu
- HCL ifadesi
 - Eşitlik işlemi
 - Boolean değer oluşturur

Bit-Seviyesinde Veri Seçici (Multiplexor)

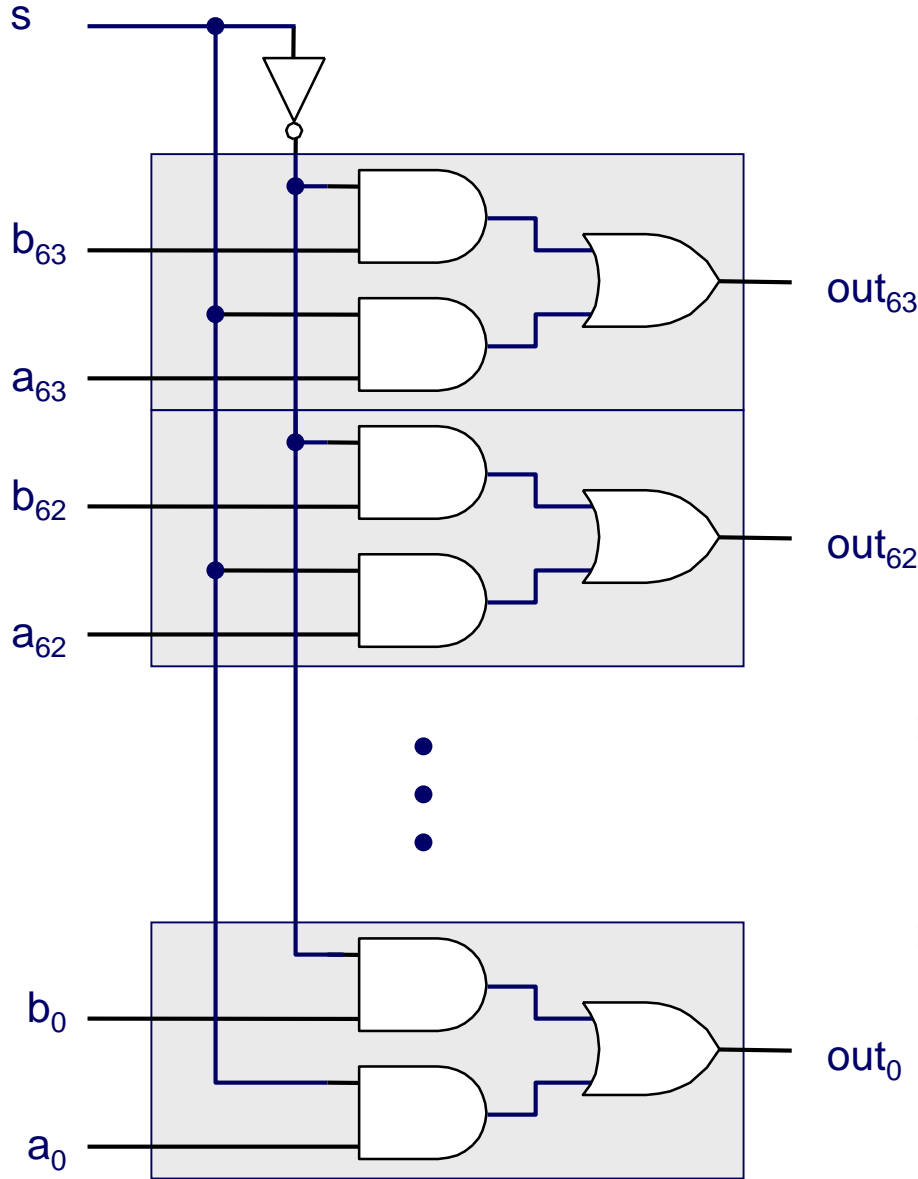


HCL İfadesi

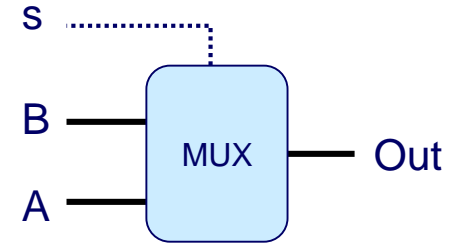
```
bool out = (s&&a) || (!s&&b)
```

- Kontrol sinyali s
- Veri sinyalleri: a ve b
- Çıkış $s=1$ iken a , $s=0$ iken b 'dir.

Sözcük Veri Seçici (Word Multiplexor)



Sözcük seviyesinde ifadesi



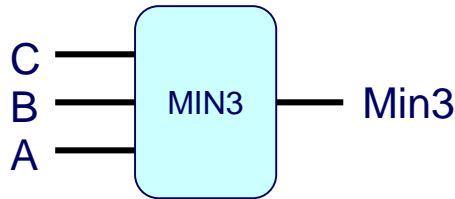
HCL ifadesi

```
int Out = [  
  s : A;  
  1 : B;  
];
```

- s kontrol sinyaline bağlı olarak A veya B giriş sözcüklerinden birini seçmeyi sağlar
- HCL ifadesi
 - C'deki Case benzeri yapı
 - Bir dizi test:değer ikilisi
 - İlk başarılı testin değerini çıkışa aktarır

HCL Sözcük Seviyesinde Örnekler

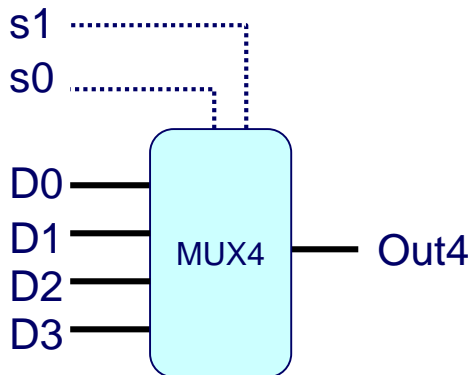
3 Sözcüğün en küçüğü



```
int Min3 = [  
    A < B && A < C : A;  
    B < A && B < C : B;  
    1                : C;  
];
```

- 3 giriş sözcüğünün minimumunu bulur
- HCL case ifadesi
- En son case eşitliği garantiler

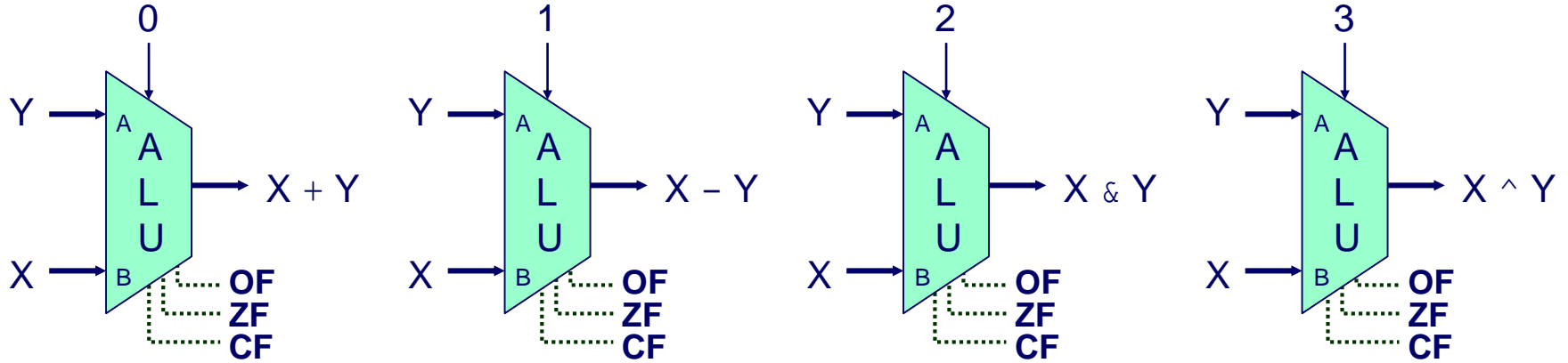
4-yollu Veri seçici



```
int Out4 = [  
    !s1&&!s0: D0;  
    !s1      : D1;  
    !s0      : D2;  
    1        : D3;  
];
```

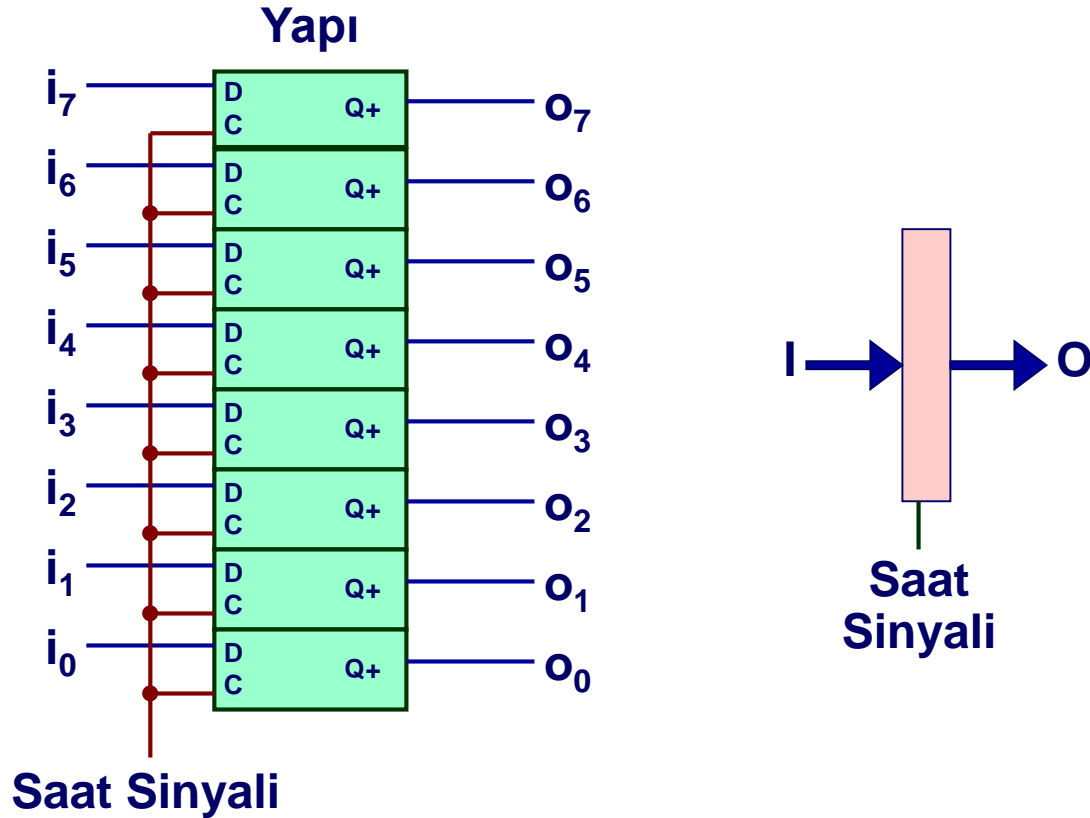
- 2 Kontrol bitine bağlı olarak 4 girişten birini seçer
- HCL case ifadesi
- Ardışıl eşleme kabul edilerek testler basitleştirilir

Aritmetik Lojik Birim



- **Kombinasyonel lojik**
 - Girişlere sürekli olarak tepki verir
- **Kontrol sinyali hesaplama fonksiyonunu seçer**
 - Y86-64'teki 4 aritmetik/lojik işleme karşılık gelir
- **Aynı zamanda durum kodlarının değerlerini de hesaplar**

Kaydediciler (Registers)



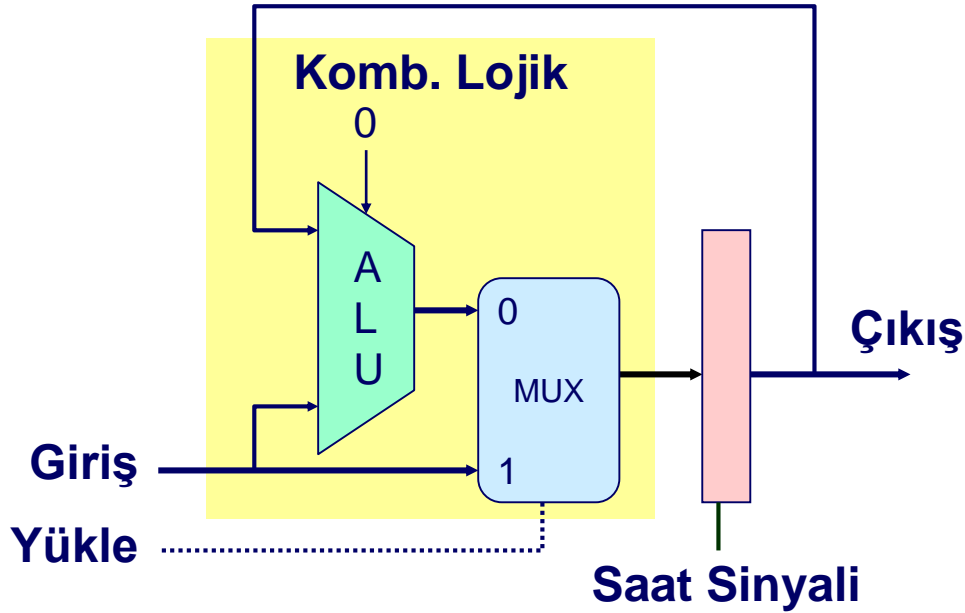
- Veri sözcüğünü kaydeder
 - Assembly kodunda görülen *program kaydedicilerinden* farklıdır
- Kenar tetiklemeli tutucuların bir araya gelmesiyle oluşur
- Girişi saat sinyalinin yükselen kenarında yükler

Kaydedicinin Çalışması



- Veri bitlerini kaydeder
- Çoğu zaman giriş ve çıkış arasında bir bariyer görevi görür
- Saat sinyali yükselirken, girişi yükler

Durum Makinası Örneği



- Biriktirici devre
- Her bir saat palsinde yükleme veya biriktirme yapar

