# Sayısal Devreler

Doç. Dr. Berna Örs Yalçın

İstanbul Teknik Üniversitesi
Elektrik-Elektronik Fakültesi
Oda No: 2318
E-mail: siddika.ors@itu.edu.tr

# Değerlendirme

- 1. Yıliçi Sınavı - %35
  - CRN30538 için 16 Temmuz 2013 saat 13:30
  - CRN30734 için 17 Temmuz 2013 saat 13:30
- 5 Ödev - %15
- Final Sınavı - %50
  - CRN30538 için 06 Ağustos 2013 saat 13:30
  - CRN30734 için 07 Ağustos 2013 saat 09:30

# Amaç ve Hedefler

- Bu dersin amacı:
  - Sayısal sistemleri tanımlamak
  - Sayısal tasarımın temel tasarım bloklarını tanımlamak
  - Temel blokların daha büyük sistemlerde nasıl kullanıldığını öğretmek

- Bu dersi başarıyla tamamlamış bir öğrenci:
  - Sayısal sistemlerin önemini anlamış olacak.
  - Bir sayısal devreyi tasarlayabilir hale gelecek.
  - Temel kombinezonsal ardışıl yapı taşlarını öğrenmiş olacak.
  - Büyük sayısal sistemlerin nasıl tasarlandığını öğrenmiş olacak.

# Kaynaklar

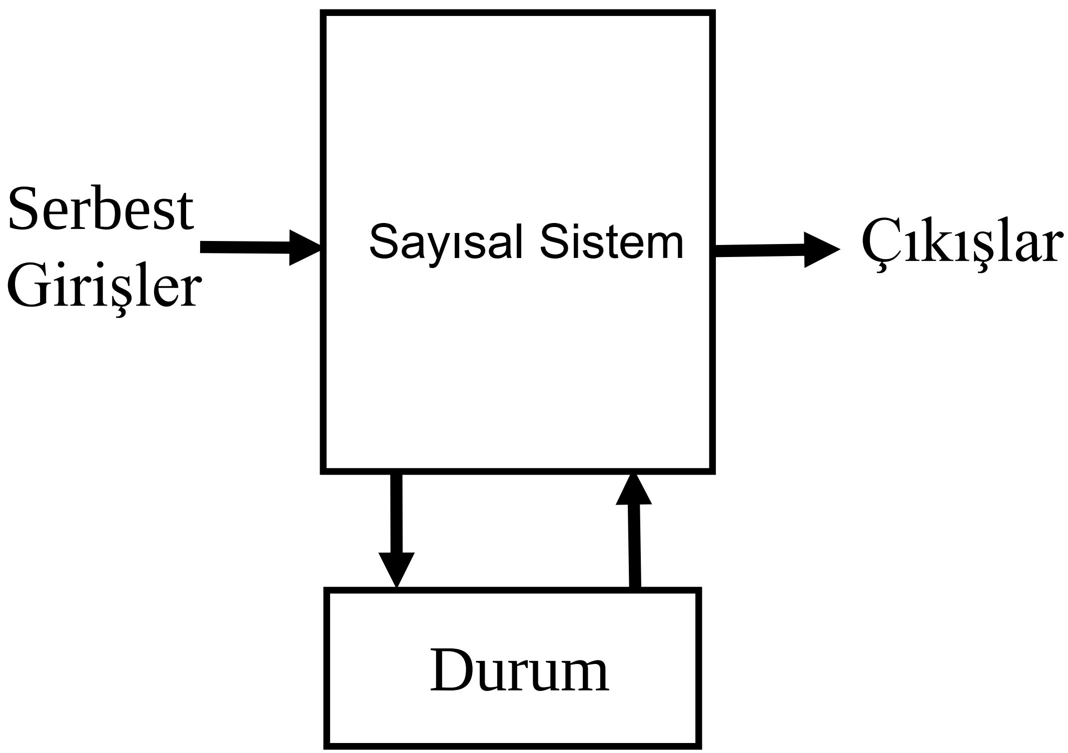
- Ders kitabı :
  - Digital Design, M. Morris Mano, Michael D. Ciletti,
  - Logic and Computer Design Fundamentals, 4/E, M. Morris Mano and Charles Kime , Prentice Hall, 2008.
- Ders sunumu, ödevler ve duyurular: ninova

# Dersin İçeriği

- 1. Sayısal Sistemler ve Bilgi
- 2. Kombinezonsal Devreler
- 3. Kombinezonsal Devre Tasarımı
- 4. Matematik Fonksiyonlar
- 5. Ardışıl Devre Elemanları
- 6. Ardışıl Devre Tasarımı

# Sayısal Sistem

- **Ayrık zamanlı serbest giriş ve sistem durumu bilgilerini kullanarak ayrık zamanlı çıkış bilgisini üretir.**

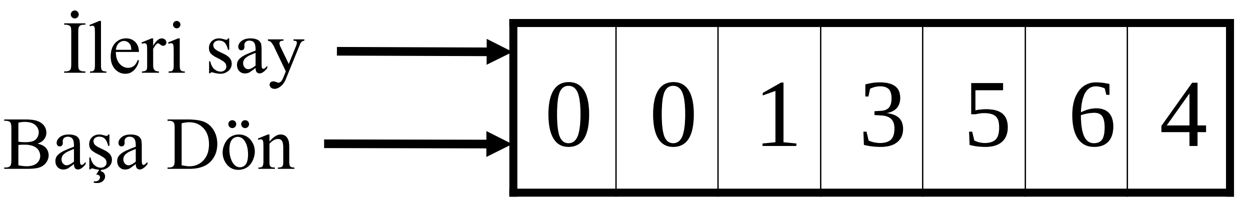


# Sayısal Sistemlerin Türleri

- **Durum Kullanılmayan**
  - **Kombinezonsal sayısal sistem**
    - **Çıkış = f(Giriş)**
- **Durum Kullanılan – Ardışıl sayısal sistem**
  - **Senkron**
    - **Durum belirli zamanlarda yenilenir**
  - **Asenkron**
    - **Durum her zaman yenilenir**
  - **Durum = f(Durum,Giriş)**
  - **Çıkış = f(Durum) veya or Çıkış = f(Durum,Giriş)**

# Sayısal Sistem Örneği:

**Bir Sayısal sayıcı:**

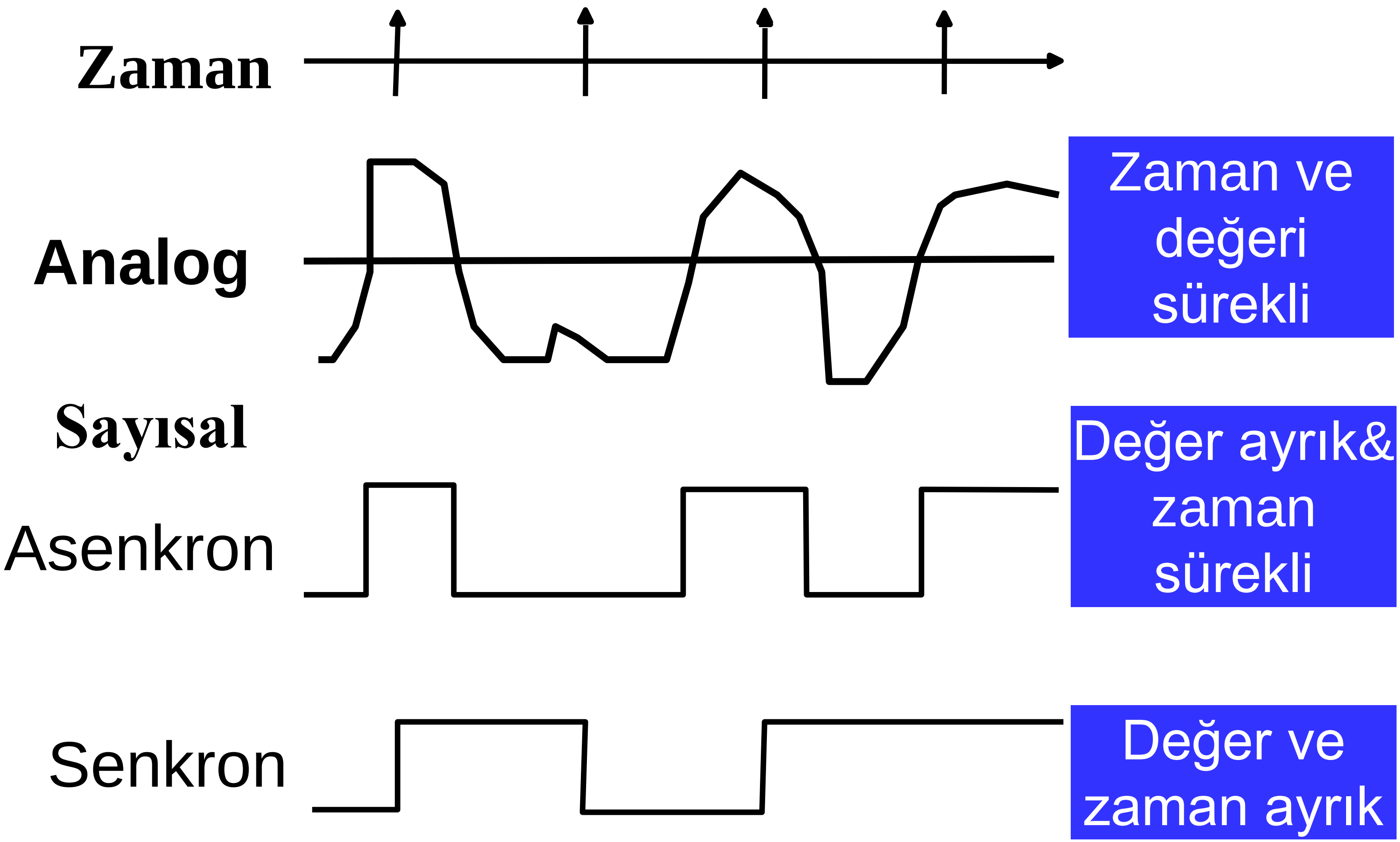



**Girişler:** **İleri say, Başa dön**
**Çıkışlar:** **Ekran**
**Durum:** **O an gösterilen değer**

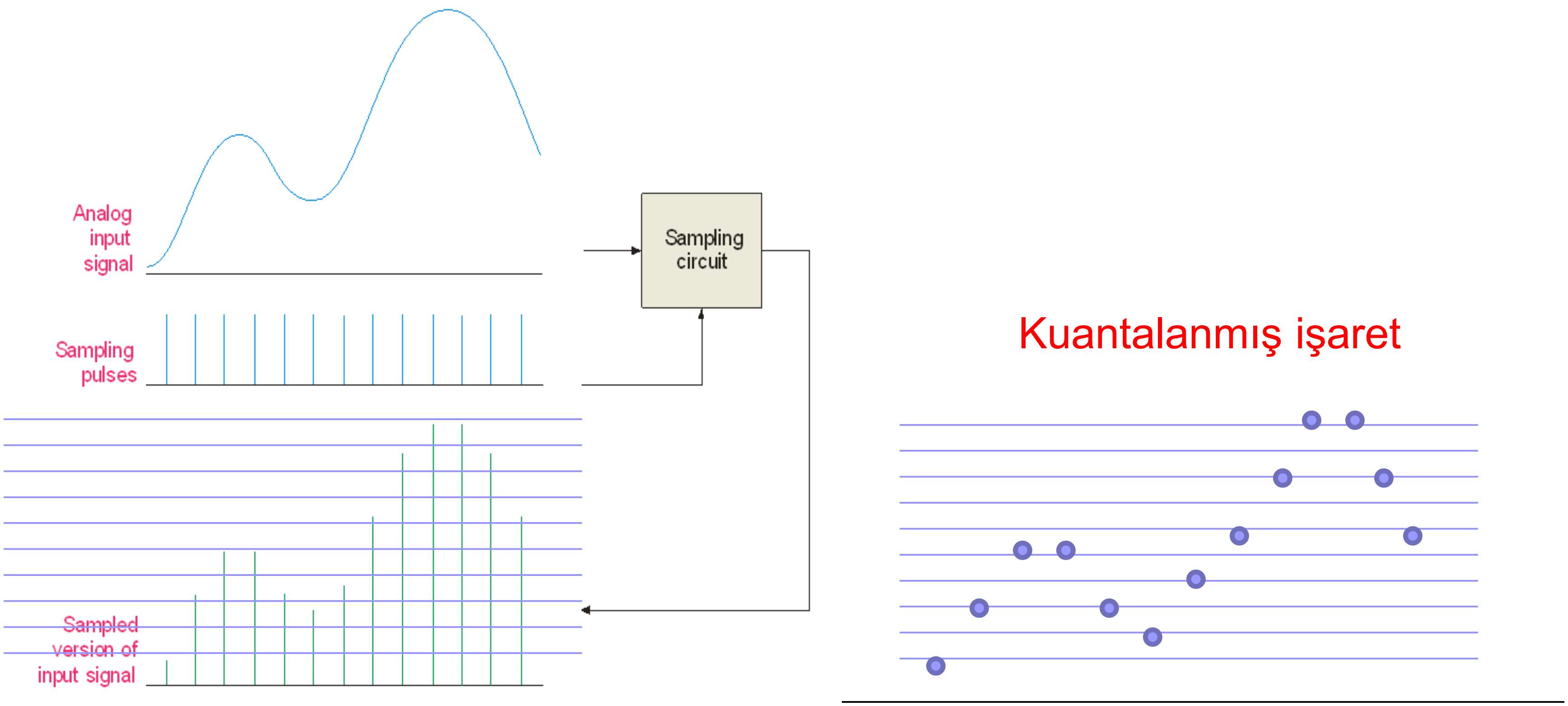
# Analog – Sayısal İşaretler

- **Gerçek dünyada karşılaştığımız birçok fiziksel büyüklük (akım, gerilim, sıcaklık, ışık şiddeti vb.) değeri sürekli bir aralık içinde değişmektedir.**
- **Sınırlar arasındaki her türlü değeri alabilen bu tür işaretlere analog işaretler denir.**
- **Sayısal sistemlerde bilgi ayrık değerler alır.**
- **İkili sayısal işaretler belli bir anda sadece olası iki değerden birini alabilirler: 0-1, yüksek – alçak, açık – kapalı.**

# Zamana göre işaret örnekleri



# Analog İşareti Sayısal İşarete Dönüştürme



# Sayısal Sistemlerin Avantajları

- Bir sayısal sisteme aynı giriş kümesi defalarca uygulandığında hep aynı çıkış kümesi elde edilir.
  - Analog sistemler ise çevre koşullarından daha çok etkilenirler ve çıkışları değişiklik gösterebilir.
- Sayısal sistem tasarımı dayandığı matematiksel temeller açısından daha kolaydır.
- Sayısal sistemleri test etmek ve hatalardan arındırmak daha kolaydır.
- Esneklik ve programlanabilirlik

# Sayısal Sistem Gerçekleme Aşamaları

Gerilim

Sıcaklık

Akım

Basınç

Hız

Görüntü

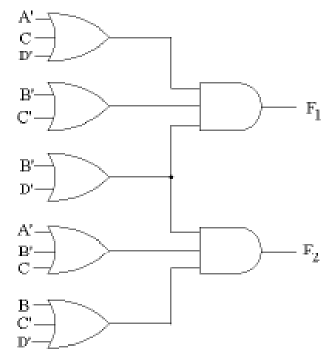
Anahtar

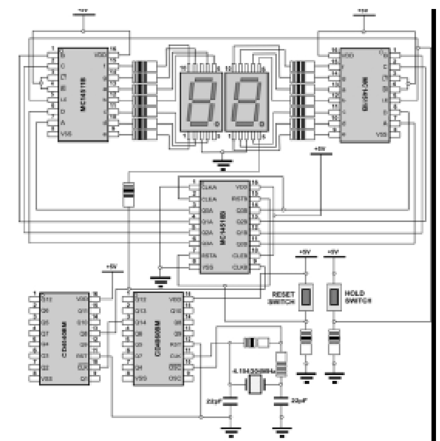
Gerçek Dünya
(Fiziksel Dünya,
Problem Domeni)

0, 1

ve, veya
tümleme

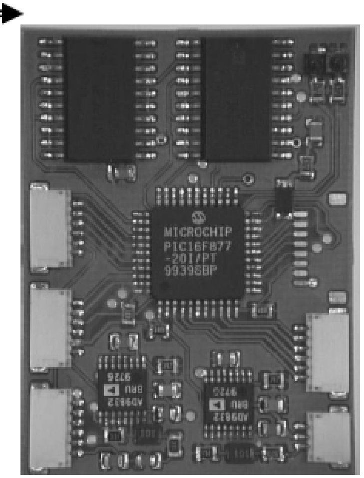
Matematiksel Dünya
(Boole Cebri)



HDL Program
(*Hardware Description Language*)

WHEN(A & B) THEN X1-C;
ELSE X2 – E # F;
X2 – (!A # C);
……




**Modelleme, soyutlama** **Tasarım** **Gerçekleme**

# Sayısal Kodlama

- Sayısal devreler yardımıyla üzerinde işlem yapılacak olan fiziksel büyüklüklere ve her türlü veriye ikili sayılar karşı düşürülür.
- Örneğin 8 basamaklı bir ikili sayı kullanarak $2^8$ tane (256) farklı "şey" ifade edebiliriz.
- Bir ikili değerin (örneğin 10001011) ne anlama geldiğine o değeri kullanacak olan sistem belirler. Bu değer bir sayı da olabilir, bir renk de, …

# BCD (Binary Coded Decimal) İkili Kodlanmış Onlu Sayılar

- 0-9 arasındaki rakamlara 4 bitlik bir ikili kod karşı düşürülür.
- Artıklı Kodlamadır: 4 bit ile 16 farklı kodlama yapılabilmekte, ancak bunlardan sadece 10 tanesi kullanılmaktadır.

**Doğal BCD:**

| Sayı: | Kod: | Sayı: | Kod: |
|---|---|---|---|
| 0: | 0000 | 5: | 0101 |
| 1: | 0001 | 6: | 0110 |
| 2: | 0010 | 7: | 0111 |
| 3: | 0011 | 8: | 1000 |
| 4: | 0100 | 9: | 1001 |

Örnek:
Sayı: 805
Kod:1000 0000 0101

- Ağırlıklı Kodlama: Bitlerin konumlarına birer ağırlık verilir.
- Doğal ikili kodlama: Sayıların ağırlıklı kodlama ile 2 tabanında gösterilmesidir.
  - $(11010)_2 = 1\cdot2^4+1\cdot2^3+0\cdot2^2+1\cdot2^1+0\cdot2^0=26$
  - Soldaki ilk basamağa en yüksek anlamlı bit (Most Significant Bit - MSB), sağdaki ilk basamağa en düşük anlamlı bit (Least Significant Bit - LSB) denir.
- Hamming Uzaklığı: n uzunluğundaki iki kod sözcüğünde aynı sırada olup değerleri farklı olan bileşenlerin sayısıdır.
  - 011 ile 101 arasındaki uzaklık 2 dir.
- Bitişik Kodlar: Birbirini izleyen sayılara karşı gelen kodlar arasındaki Hamming uzaklığı 1 ise o kodlama bitişiktir.
- Çevrimli Kodlar: Kodlama bitişik ve ayrıca son kod ile ilk kod arasında da Hamming uzaklığı 1 ise kod çevrimlidir.

Çevrimli BCD Kodu:

| Sayı: | Kod: | Sayı: | Kod: |
|---|---|---|---|
| 0: | 0000 | 5: | 1110 |
| 1: | 0001 | 6: | 1010 |
| 2: | 0011 | 7: | 1000 |
| 3: | 0010 | 8: | 1100 |
| 4: | 0110 | 9: | 0100 |

Gray Kodu: $2^n$ elemanlı bir küme için 2 tabanında artıksız ve çevrimli bir kodlama

Örnek: 2 –bitlik bir Gray kodu:

| Sayı: | Kod: |
|---|---|
| 0: | 00 |
| 1: | 01 |
| 2: | 11 |
| 3: | 10 |

# Sayıların Gösterilimi

- Pozitif taban, ağırlıklı sayı sistemleri
- r tabanında gösterilen bir sayı basamaklardan oluşan bir dizi ile gösterilir.

  $$A_{n-1}A_{n-2} \ldots A_1A_0 \,.\, A_{-1}\,A_{-2} \ldots A_{-m+1}\,A_{-m}$$

  burada $0 \leq A_i < r$ ve **.** taban noktasıdır.
- Basamak dizisi bir kuvvet serisini ifade eder:

$$(\text{sayı})_{10} = \left(\sum_{i=0}^{i=n-1} A_i \cdot r^i\right) + \left(\sum_{j=-m}^{j=-1} A_j \cdot r^j\right)$$

**(Tamsayı parçası)+ (Kesirli parça)**

# İşaretsiz Sayıların Gösterilmesi

Doğal ağırlıklı ikili kodlama kullanılır.

Örnek: $215_{10}=(1101\ 0111)_2= =1 \cdot 2^7+1\cdot 2^6+0\cdot 2^5+1\cdot 2^4+0\cdot 2^3+1\cdot 2^2+1\cdot 2^1+1\cdot 2^0$

En yüksek anlamlı bit (MSB)

En düşük anlamlı bit (LSB)

8 bit ile ifade edilebilecek en büyük işaretsiz sayı: $(1111\ 1111)_2=255_{10}$
8 bit ile ifade edilebilecek en küçük işaretsiz sayı: $(0000\ 0000)_2=0_{10}$

# Çok kullanılan tabanlar

| İsim | Taban | Basamaklar |
|---|---|---|
| İkili | 2 | 0,1 |
| Sekizli | 8 | 0,1,2,3,4,5,6,7 |
| Onluk | 10 | 0,1,2,3,4,5,6,7,8,9 |
| Onaltılık | 16 | 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F |

- **Onaltılık tabanda kullanılan 6 harf 10, 11, 12, 13, 14 ve 15 i gösterir.**

# Farklı tabanda sayıların gösterilimi

| Decimal (Base 10) | Binary (Base 2) | Octal (Base 8) | Hexadecimal (Base 16) |
|---|---|---|---|
| 00 | 00000 | 00 | 00 |
| 01 | 00001 | 01 | 01 |
| 02 | 00010 | 02 | 02 |
| 03 | 00011 | 03 | 03 |
| 04 | 00100 | 04 | 04 |
| 05 | 00101 | 05 | 05 |
| 06 | 00110 | 06 | 06 |
| 07 | 00111 | 07 | 07 |
| 08 | 01000 | 10 | 08 |
| 09 | 01001 | 11 | 09 |
| 10 | 01010 | 12 | 0A |
| 11 | 01011 | 13 | 0B |
| 12 | 01100 | 14 | 0C |
| 13 | 01101 | 15 | 0D |
| 14 | 01110 | 16 | 0E |
| 15 | 01111 | 17 | 0F |
| 16 | 10000 | 20 | 10 |

# Onluk tabandan diğer tabanlara dönüşüm

1) Tamsayı parçayı dönüştür

2) Kesirli parçayı dönüştür

3) İki sonucu bir taban noktası ile birleştir.

# Dönüşüm Kuralları

- Tam sayı parçayı dönüştürme:
  - Sayıyı dönüştürülecek taban ile tekrarlı böl.
  - Kalanları ters sırada kayıt et.
  - Yeni tabanda basamaklar ters sırada kalanlardır.
- Kesirli parçayı dönüştürme:
  - Kesiri dönüştürülecek tabanla tekrarlı çarp.
  - Tamsayı basamağını kayıt et.
  - Yeni kesirli sayının basamakları sonuçların hesaplandığı sıradaki tam sayı kısımlarıdır.

# Örnek:
# $46.6875_{10}$ sayısını 2 tabanına dönüştür

- 46 yı ikili tabana dönüştür
  - 46/2=23 kalan 0
  - 23/2=11 kalan 1
  - 11/2=5 kalan 1
  - 5/2=2 kalan 1
  - 2/2=1 kalan 0
  - 1/2=0 kalan 1
- 0.6875 yı ikili tabana dönüştür
  - 0.6875*2=1.375
  - 0.375*2=0.75
  - 0.75*2=1.5
  - 0.5*2=1
  - 0*2=0
- İki sonucu bir taban noktası ile birleştir
  - $101110.10110_2$

# Örnek:
# $46.6831_{10}$ sayısını 16 tabanına dönüştür

- 46 yı 16 tabana dönüştür
  - 46/16=2 kalan 14
  - 2/16=0 kalan 2
- 0.6831 yı 16 tabana dönüştür
  - 0.6831*16=10.9296
  - 0.9296*16=14.8736
  - 0.8736*16=13.9776
  - 0.9776*16=15.6416
  - …
- Sonuçları kesir noktası ile birleştir:
  - $2E.AEDF_{16}$

# r tabanından onluk tabana dönüşüm

- Tabanın ilgili kuvveti ile basamakların çarpımını topla
- $101110.10110_2$ sayısını onluk taban çevir

$$101110_2 = 1 \cdot 32 + 0 \cdot 16 + 1 \cdot 8 + 1 \cdot 4 + 1 \cdot 2 + 0 \cdot 1$$
$$= 32 + 8 + 4 + 2$$
$$= 46$$

$$0.1011_2 = 1/2 + 1/8 + 1/16$$
$$= 0.5000 + 0.1250 + 0.0625$$
$$= 0.6875$$

# Sekizli/onaltılı (Octal/Hex) tabandan ikili ve geriye dönüşüm

- Sekizli (onaltılı) den İkili tabana:
  - Her bir basamak ikili tabanda yazılır.
- İkiliden sekizli (onaltılı) tabanına:
  - Basamaklar taban noktasından başlanarak iki tarafa doğru üçlü (dörtlü) gruplanır.
  - Her bir grup sekizli (onaltılı) tabanına dönüştürülür.

# Örnek

- Sekizli (onaltılı) den İkili tabana:
  - $743.056_8 = 111\ 100\ 011.000\ 101\ 110_2$
  - $A49.0C6_{16} = 1010\ 0100\ 1001.0000\ 1100\ 0110_2$
- İkiliden sekizli (onaltılı) tabanına:
  - $1|011|100|011.000|101|110|1_2 = 1343.0564_8$
  - $1|1010|0100|1001.0010|1100|0110|1_2 = 1A49.2C68_{16}$

# İkili taban kullanılarak sekizli den onaltılık tabanına dönüşüm

- Octal den ikili tabana dönüştür.
- Daha önce anlatıldığı gibi hez tabanına dönüştür.

# 2'nin özel kuvvetleri

- $2^{10}$ (1024) Kilo, "K" ile gösterilir.
- $2^{20}$ (1,048,576) Mega, "M" ile gösterilir.
- $2^{30}$ (1,073, 741,824) Giga, "G" ile gösterilir.
- $2^{40}$ (1,099,511,627,776 ) Tera, "T" ile gösterilir.

# İşaretli Sayıların Gösterilmesi

- Pozitif ve negatif sayıları ayırt etmek için ikili sayının en yüksek anlamlı bitine bakılır.
  - "0" ise pozitif
  - "1" ise negatif
- 8 bit ile gösterilebilecek pozitif sayılar 0000 0000 ile 0111 1111 yani 0 ile + 127 arasında değişecektir.
- Negatif sayıların gösteriminde 2'ye tümleme yöntemi kullanılır.
  - Pozitif bir sayının 2'ye tümleyeni hesaplandığında o sayının negatif gösterilimi elde edilmiş olur.
- Bir sayının 2'ye tümleyenini elde etmek için
  - Sayı 1'e tümlenir, yani 0'lar 1, 1'ler 0 yapılır.
  - 1'e tümlenmiş sayıya 1 eklenir.

# Negatif Sayılara Örnekler

8 bitlik $5_{10}$ sayısı 5 mod 256 olarak düşünülebilir.

$-5_{10}$ mod 256 = 256-5 mod 256=251 mod 256

$$
\begin{array}{r}
1\;0000\;0000 \\
-\;\;0\;0000\;0101 \\
\hline
0\;1111\;1011
\end{array}
$$

Negatif sayı

| | |
|---|---|
| | 0000 0101 |
| 1'e tümleme | 1111 1010 |
| 1 ekleme | + 1 |
| 2'ye tümleme | 1111 1011 |

Negatif sayı

# Negatif Sayılara Örnekler

$5_{10}$ mod 256 = 256-(-5) mod 256= 256-251 mod 256

| | |
|---|---|
| | 1 0000 0000 |
| − | 0 1111 1011 |
| | 0 0000 0101 |

Pozitif sayı

| | | |
|---|---|---|
| | | 1111 1011 |
| 1’e tümleme | | 0000 0100 |
| 1 ekleme | + | 1 |
| 2’ye tümleme | | 0000 0101 |

Pozitif sayı

# İkili Sayıların Uzatılması

- Bazı durumlarda daha az bit ile ifade edilen bir sayıyı daha büyük bir yere yazmak ya da daha uzun bir sayı ile işleme sokmak gerekebilir.
- Bu durumda sayı uzatılır.
- İşaretsiz sayılar: Sayının başına gerektiği kadar sıfır ‘0’ eklenir.
  - Örnek: 4 bitlik $3_{10}$: 0011 8 bitlik $3_{10}$: 0000 0011
- İşaretli sayılar: Sayının başına sayının işareti gerektiği kadar eklenir. Buna işaret uzatma denir.
  - Örnek: 4 bitlik $3_{10}$: 0011 8 bitlik $3_{10}$: 0000 0011
  - Örnek: 4 bitlik $-7_{10}$: 1001 8 bitlik $-7_{10}$: 1111 1001

# İkili Matematik

- Elde ile bir bit uzunluklu toplama
- Birden fazla bit uzunluklu toplama
- Borç ile bir bit uzunluklu çıkartma
- Birden fazla bit uzunluklu çıkartma
- Çarpma

# Elde ile bir bit uzunluklu toplama

**Toplanacak iki basamak (X,Y), elde girişi (Z) kullanılarak toplama yapıldığında aşağıdaki toplam (S) ve elde çıkışı (C) elde edilir:**

**Elde girişi (Z) 0 ise:**

| | | | | |
|---|---|---|---|---|
| Z | 0 | 0 | 0 | 0 |
| X | 0 | 0 | 1 | 1 |
| + Y | + 0 | + 1 | + 0 | + 1 |
| C S | 0 0 | 0 1 | 0 1 | 1 0 |

**Elde girişi (Z) 1 ise:**

| | | | | |
|---|---|---|---|---|
| Z | 1 | 1 | 1 | 1 |
| X | 0 | 0 | 1 | 1 |
| + Y | + 0 | + 1 | + 0 | + 1 |
| C S | 0 1 | 1 0 | 1 0 | 1 1 |

# İşaretsiz sayıların toplanması

| | | | | |
|---|---|---|---|---|
| Elde | 00000 | | 01100 | |
| X | 01100 | 12 | 10110 | 22 |
| Y | +10001 | +17 | +10111 | +23 |
| Toplam | 11101 | 29 | 101101 | 45 |

- Not: En düşük anlamlı basamağın elde girişi her zaman ‘0’ dır.
- n-bitlik iki sayı toplandığında sonuç n+1-bitliktir.

# İşaretli sayıların toplanması

| | | | | |
|---|---|---|---|---|
| Elde | 0010 | | 0100 | |
| X | 1101 | -3 | 0011 | 3 |
| Y | +0001 | +1 | +0010 | +2 |
| Toplam | 1110 | -2 | 0101 | 5 |

Sonuç negatif

Sonuç pozitif

# İşaretli sayıların toplanması

| | | | | |
|---|---|---|---|---|
| Elde | 11110 | | 11100 | |
| X | 1101 | -3 | 0011 | 3 |
| Y | +1111 | -1 | +1110 | -2 |
| Toplam | 11100 | -4 | 10001 | 1 |

Sonuç negatif

İhmal edilir

Sonuç pozitif

İhmal edilir

| | | | | |
|---|---|---|---|---|
| Elde | 1000 | | 0000 | |
| X | 0100 | 4 | 1010 | -6 |
| Y | +0101 | +5 | +1101 | -3 |
| Toplam | 1001 | 9 | 10111 | -9 |

Sonuç negatif midir?  İhmal edilir  Sonuç pozitif midir?

- Taşma oluşmuştur. 4-bit ile gösterilebilen en büyük pozitif sayı +7 dir. Daha büyük sayılar 4-bit ile gösterilemez.
- 4-bit ile gösterilebilen mutlak değeri en büyük negatif sayı -8 dir. Mutlak değeri daha büyük olan negatif sayılar 4-bit ile gösterilemez.
- Sayıların hangi bit uzunluğu ile gösterileceğine yapılacak işlemlere ve bu işlemler sonucunda ortaya çıkması olası olan sonuçların sınırlarına göre karar verilmelidir.

# Borç ile bir bit uzunluklu çıkartma

- **Çıkarma işlemi yapılacak iki basamak (X,Y), borç girişi (Z) kullanılarak çıkarma yapıldığında aşağıdaki fark (S) ve borç çıkışı (B) elde edilir:**
- **Borç girişi (Z) 0 ise:**

| | | | | |
|---|---|---|---|---|
| Z | 0 | 0 | 0 | 0 |
| X | 0 | 0 | 1 | 1 |
| - Y | -0 | -1 | -0 | -1 |
| BS | 0 0 | 1 1 | 0 1 | 0 0 |

- **Borç girişi (Z) 1 ise:**

| | | | | |
|---|---|---|---|---|
| Z | 1 | 1 | 1 | 1 |
| X | 0 | 0 | 1 | 1 |
| - Y | -0 | -1 | -0 | -1 |
| BS | 11 | 1 0 | 0 0 | 1 1 |

# İşaretsiz sayılar ile çıkartma

| | | | | |
|---|---|---|---|---|
| Borç | 00000 | | 00110 | |
| X | 10110 | 22 | 10110 | 22 |
| Y | - 10010 | -18 | - 10011 | -19 |
| Fark | 00100 | 4 | 00011 | 3 |

- Not: En düşük anlamlı basamağın borç girişi her zaman ‘0’ dır. Eğer Y>X ise X ve Y yer değiştirilir ve sonucun başına – işareti eklenir.

# İşaretli sayılar ile 2’ye tümleme kullanılarak çıkartma

| | | | | | |
|---|---|---|---|---|---|
| X | 3 | 0011 | | 0011 | 3 |
| Y | -1 | -0001 | 2’ye tümleyen | +1111 | +(-1) |
| Fark | 2 | | | 10010 | 2 |

İhmal edilir

Sonuç pozitif

| | | | | | |
|---|---|---|---|---|---|
| X | 3 | 0011 | | 0011 | 3 |
| Y | -4 | -0100 | 2’ye tümleyen | +1100 | +(-4) |
| Fark | -1 | | | 1111 | -1 |

Sonuç negatif

| | | | | | |
|---|---|---|---|---|---|
| X | 3 | 0011 | | 0011 | 3 |
| Y | -(-1) | -1111 | 2’ye tümleyen | +0001 | +1 |
| Fark | 4 | | | 0100 | 4 |

Sonuç pozitif

| | | | | | |
|---|---|---|---|---|---|
| X | 1 | 0001 | | 0001 | 1 |
| Y | -(-7) | -1001 | 2'ye tümleyen | +0111 | +7 |
| Fark | 8 | | | 1000 | 8 |

Sonuç negatif midir?

| | | | | | |
|---|---|---|---|---|---|
| X | -5 | 1011 | | 1011 | -5 |
| Y | -4 | -0100 | 2'ye tümleyen | +1100 | +(-4) |
| Fark | -9 | | | 10111 | -9 |

İhmal edilir

Sonuç pozitif midir?

- Taşma oluşmuştur. 4-bit ile gösterilebilen en büyük pozitif sayı +7 dir. Daha büyük sayılar 4-bit ile gösterilemez.
- 4-bit ile gösterilebilen mutlak değeri en büyük negatif sayı -8 dir. Mutlak değeri daha büyük olan negatif sayılar 4-bit ile gösterilemez.

# İkili Çarpma

**İkili çarpım tablosu:**

$$0 * 0 = 0 \ | \ 1 * 0 = 0 \ | \ 0 * 1 = 0 \ | \ 1 * 1 = 1$$

**Çarpmayı birden çok bit uzunluklu sayılar ile yapma:**

| | |
|---|---|
| **Çarpılan** | **1011** |
| **Çarpan** | **x 101** |
| **Ara çarpım** | **1011** |
| | **0000 -** |
| | **1011 - -** |
| **Çarpım** | **110111** |

# İkili Lojik ve Kapılar

- İkili değişkenler iki değerden birini alırlar
- Lojik işlemler ikili değerler ve ikili değişkenler üzerinde çalışır
- Temel lojik işlemler VE, VEYA ve TÜMLEME dir
- Lojik kapılar lojik işlemleri gerçeklerler
- Boole Cebri: lojik fonksiyonları tanımlamak ve birbirine dönüştürmek için kullanılan matematik sistemidir
- Biz sayısal sistemlerin analizi ve tasarımının temelini oluşturan Boole cebrini inceleyeceğiz

# İkili Değişkenler

- İkili değişkenlere farklı isimler verilebilir
  - Doğru/Yanlış
  - Açık/Kapalı
  - Evet/Hayır
  - 1/0
- Biz bu iki değeri göstermek için 1 ve 0'ı kullanacağız.

# Lojik İşlemler

- Temel üç lojik işlem:
  - VE
  - VEYA
  - TÜMLEME
- VE (·) ile gösterilir
- VEYA (+) ile gösterilir
- TÜMLEME değişkenin üzerinde bir çizgi( ¯ ), değişkenden sonra (‘) veya değişkenden önce (~) ile gösterilir

# Gösterilim Örnekleri

- Örnekler:
  - $Y = A \cdot B \Rightarrow$ “Y A ve B dir”
  - $z = x + y \Rightarrow$ “z x veya y dir”
  - $X = \overline{A} \Rightarrow$ “X A’nın tersidir”
- Not:

  1 + 1 = 2 (“bir artı bir ikidir)

  1 + 1 = 1 ( “1 veya 1 1’e eşittir”)

  ifadeleri birbirine eşit değildir.

# İşlem Tanımları

- İşlemler ‘0’ ve ‘1’ değerleri üzerinden tanımlanırlar.

| VE | VEYA | TÜMLEME |
| --- | --- | --- |
| $0 \cdot 0 = 0$ | $0 + 0 = 0$ | $\bar{0} = 1$ |
| $0 \cdot 1 = 0$ | $0 + 1 = 1$ | $\bar{1} = 0$ |
| $1 \cdot 0 = 0$ | $1 + 0 = 1$ | |
| $1 \cdot 1 = 1$ | $1 + 1 = 1$ | |

# Doğruluk Tabloları

- *Doğruluk Tablosu* – bir fonksiyonun çıkış değerini bu fonksiyonun bütün mümkün olan giriş değerleri için gösteren tablo
- Örnek: Temel işlemlerin doğruluk tabloları

| VE | | |
|---|---|---|
| X | Y | $Z = X \cdot Y$ |
| 0 | 0 | 0 |
| 0 | 1 | 0 |
| 1 | 0 | 0 |
| 1 | 1 | 1 |

| VEYA | | |
|---|---|---|
| X | Y | $Z = X + Y$ |
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 1 |

| TÜMLEME | |
|---|---|
| X | $Z = \overline{X}$ |
| 0 | 1 |
| 1 | 0 |

# Boole Cebri

- B={0,1} kümesi üzerinde tanımlı
- İkili İşlemler : VE, VEYA (·, +)
- Birli İşlem: TÜMLEME (‾)

Aksiyomlar

a, b, c $\in$ B olmak üzere

1. Kapalılık: $a + b = c$ $a \cdot b=c$
2. Değişme: $a + b = b + a$ $a \cdot b=b \cdot a$
3. Dağılma: $a+(b \cdot c)=(a+b) \cdot (a+c)$ $a \cdot (b+c)=a \cdot b+a \cdot c$
4. Birleşme: $a+(b+c)=(a+b)+c$ $a \cdot (b \cdot c)=(a \cdot b) \cdot c$
5. Etkisiz eleman: $a+0=a$ $a \cdot 1=a$
6. Tümleme: $a+a'=1$ $a \cdot a'=0$

# Boole İşlemlerinin Sırası

1. Parantez
2. TÜMLEME
3. VE
4. VEYA

- Sonuç: VEYA ifadelerinin etrafında parantez vardır.
- Örnek: F = A(B + C)(C + D)

# Özellikler ve Teoremler

- Burada gösterilen tüm özellikler ve teoremler Boole cebrinin aksiyomları kullanılarak ispat edilebilir.

1. Yutma: $a+1=1$ $\quad a \cdot 0=0$
2. Dönüşme: $(a')'=a$
3. Sabit kuvvet: $a+a+\ldots+a=a$ $\quad a \cdot a \cdot \ldots \cdot a=a$
4. Soğurma: $a+a \cdot b=a$ $\quad a \cdot (a+b)=a$
5. De Morgan Teoremi: $(a+b)'=a' \cdot b'$ $\quad (a \cdot b)'=a'+b'$
6. Genel De Morgan Teoremi:
   $f'(X1,X2,\ldots,Xn,0,1,+,\cdot) \Leftrightarrow f(X1',X2',\ldots,Xn',1,0,\cdot,+)$

# Örnek1: Boole Teoremlerinin İspatı

- $A + A \cdot B = A$ (Yutma)

| İspat adımları | Aksiyomlar |
| --- | --- |
| $A + A \cdot B$ | |
| $= A \cdot 1 + A \cdot B$ | $X = X \cdot 1$ |
| $= A \cdot (1 + B)$ | $X \cdot Y + X \cdot Z = X \cdot (Y + Z)$ (Dağılma) |
| $= A \cdot 1$ | $1 + X = 1$ |
| $= A$ | $X \cdot 1 = X$ |

- İspatları yapmamızın sebebi:
  - Boole cebrinin aksiyom ve teoremlerini kullanmayı öğrenmek
  - Boole fonksiyonlarıyla işlem yapmak için doğru aksiyom ve teoremi seçmeyi öğrenmek

# Örnek2: Boole Teoremlerinin İspatı

- $AB + A'C + BC = AB + A'C$ (Consensus Theorem)

| İspat adımları | Aksiyomlar |
|---|---|
| $AB + A'C + BC$ | |
| $=AB + A'C + 1 \cdot BC$ | $1 . X = X$ |
| $=AB + A'C + (A + A') \cdot BC$ | $X + X' = 1$ |
| $=AB + A'C + ABC + A'BC$ | $X(Y + Z) = XY + XZ$ |
| $=AB + ABC + A'C + A'BC$ | $X + Y = Y + X$ |
| $=AB + A'(C + BC)$ | $X(Y + Z) = XY + XZ$ |
| $=AB + A'C$ | |

# Örnek3: Boole Teoremlerinin İspatı

- $(\overline{X+Y})Z + X\overline{Y} = \overline{Y}(X+Z)$

| İspat adımları | Aksiyomlar |
|---|---|
| $(\overline{X+Y})Z + X\overline{Y}$ | |
| $= X'Y'Z+XY'$ | De Morgan Teoremi |
| $= (X'Z+X)Y'$ | Dağılma |
| $= (X'+X)(Z+X)Y'$ | Dağılma |
| $= (Z+X)Y'$ | Tümleme |

# Boole Fonksiyonlarının Değerlendirilmesi

$F1 = xy\overline{z}$

$F2 = x + \overline{y}z$

$F3 = \overline{x}\overline{y}\overline{z} + \overline{x}\,y\,z + x\overline{y}$

$F4 = x\overline{y} + \overline{x}\,z$

Giriş sayısı=n olmak üzere

$2^{2^n}$ farklı n değişkenli Boole fonksiyonu tanımlanabilir.

| x | y | z | F1 | F2 | F3 | F4 |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 1 | 1 |
| 1 | 0 | 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | 1 | 0 | 0 |

# Boole Fonksiyonlarının İndirgenmesi

- Amaç en az sayıda değişken bırakmak.

$AB+A'CD+A'BD+A'CD'+ABCD$

$= AB + ABCD + A'CD + A'CD' + A'BD$

$= AB + AB(CD) + A'C(D+D') + A'BD$

$= AB + A'C + A'BD = B(A + A'D) + A'C$

$= B(A + D) + A'C$

- 5 değişken

# Kanonik Gösterilimler

- Kanonik gösterilimler nelerdir?
- Çarpım terimleri (Minterms) ve toplam terimleri (Maxterms)
- Çarpım terimleri ve toplam terimlerin indis ile gösterilimi
- Çarpımlar toplamı gösterilim
- Toplamlar çarpımı gösterilim
- Fonksiyonların tümlemelerinin gösterilimi
- Gösterilimler arası dönüşümler

# Kanonik Gösterilimler

- Boole fonksiyonları aşağıdaki kolaylıkları sağlayacak bir gösterilimle tanımlanır:
  - Eşitliğin karşılaştırılması
  - Doğruluk tablosu ile birebir olma
- Çok kullanılan kanonik gösterilimler:
  - Çarpımlar toplamı
  - Toplamlar çarpımı

# Çarpım terimleri

- Çarpım terimleri bütün değişkenlerin veya tümleyenlerinin göründüğü VE terimleridir.
- *n değişkenli bir Boole fonksiyonunun $2^n$ çarpım terimi vardır.*
- Örnek: İki değişkenli bir Boole fonksiyonunun çarpım terimleri 2 x 2 = 4 tanedir:

$$XY$$
$$X\overline{Y}$$
$$\overline{X}Y$$
$$\overline{X}\,\overline{Y}$$

# Toplam terimleri

- Toplam terimleri bütün değişkenlerin veya tümleyenlerinin göründüğü VEYA terimleridir.
- *n değişkenli bir Boole fonksiyonunun $2^n$ toplam terimi vardır.*
- Örnek: İki değişkenli bir Boole fonksiyonunun çarpım terimleri 2 x 2 = 4 tanedir:

$$X + Y$$
$$X + \overline{Y}$$
$$\overline{X} + Y$$
$$\overline{X} + \overline{Y}$$

# Çarpım ve Toplam Terimleri

- Örnek: İki değişkenli çarpım ve toplam terimleri

| İndis | Çarpım Terimi | Toplam Terimi |
|---|---|---|
| 0 | $\bar{x}\,\bar{y}$ | $x + y$ |
| 1 | $\bar{x}\,y$ | $x + \bar{y}$ |
| 2 | $x\,\bar{y}$ | $\bar{x} + y$ |
| 3 | $x\,y$ | $\bar{x} + \bar{y}$ |

- İndis hangi değişkeninin kendisinin hangi değişkenin tümleyeninin yer aldığını gösterir.

# Normal Sıralama

- Çarpım ve toplam terimlerine bir sıra numarası karşılık düşer.
- Bu sıra numarası bir ikili sayı ile gösterilir.
- İkili sayının bitleri değişkenlerin kendisinin veya tümleyenin terim içinde yer alacağını gösterir.
- Çarpım ve toplam terimlerinin içinde değişkenler hep aynı sırada yer alırlar
- Örnek: a, b, c değişkenleri için:
  - Toplam terimleri: $(a + \bar{b} + c)$, $(a + b + c)$
  - Terimler: $(b + a + c)$, $a\,c\,b$ ve $(c + b + a)$ normal sıralamada değiller.
  - Çarpım terimleri: $a\,b\,c$, $\bar{a}\,\bar{b}\,c$, $a\,\bar{b}\,c$
  - Terimler : $(a + c)$, $b\,\bar{c}$ ve $(a + \bar{b})$ bütün değişkenleri içermiyorlar.

# İndisin Kullanılma Sebebi

- İkili sayı ile gösterilen indis çarpım veya toplam terimindeki değişkenlerin kendisinin mi yoksa tümleyeninin mi kullanılacağını gösterir.
- Çarpım terimleri için:
  - “1” değişkenin kendisinin
  - “0” değişkenin tümleyeninin yer aldığını gösterir.
- Toplam terimleri için:
  - “0” değişkenin kendisinin
  - “1” değişkenin tümleyeninin yer aldığını gösterir.

# Üç değişken için indis örneği

- Değişkenler X, Y ve Z.
- Normal sıralama X, Y, Z.
- İndis $0_{10}=(000)_2$ ise çarpım teriminde bütün değişkenlerin tümleyeni görülür, toplam teriminde bütün değişkenlerin kendileri görülür.
- Çarpım terimi 0, $m_0$ ile adlandırılır $\overline{X}\overline{Y}\overline{Z}$ .
- Toplam terimi 0, $M_0$ ile adlandırılır (X + Y + Z).
- Çarpım terimi 6 ?
- Toplam terimi 6 ?

# İndis Örnekleri – Dört Değişken

| İndis i | İkili Sayı | Çarpım $m_i$ | Toplam $M_i$ |
|---|---|---|---|
| 0 | 0000 | $\overline{a}\,\overline{b}\,\overline{c}\,\overline{d}$ | $a+b+c+d$ |
| 1 | 0001 | $\overline{a}\,\overline{b}\,\overline{c}\,d$ | ? |
| 3 | 0011 | ? | $a+b+\overline{c}+\overline{d}$ |
| 5 | 0101 | $\overline{a}\,b\,\overline{c}\,d$ | $a+\overline{b}+c+\overline{d}$ |
| 7 | 0111 | ? | $a+\overline{b}+\overline{c}+\overline{d}$ |
| 10 | 1010 | $a\,\overline{b}\,c\,\overline{d}$ | $\overline{a}+b+\overline{c}+d$ |
| 13 | 1101 | $a\,b\,\overline{c}\,d$ | ? |
| 15 | 1111 | $a\,b\,c\,d$ | $\overline{a}+\overline{b}+\overline{c}+\overline{d}$ |

# Çarpım ve Toplam Terimlerinin İlişkisi

- DeMorganTeoremi

  $$\overline{x \cdot y} = \overline{x} + \overline{y} \quad \text{ve} \quad \overline{x + y} = \overline{x} \cdot \overline{y}$$

- İki değişkenli örnek:

  $$M_2 = \overline{x} + y \quad \text{ve} \quad m_2 = x \cdot \overline{y}$$

  Yani $M_2$ $m_2$ nin tümleyenidir. $m_2$ de $M_2$ nin tümleyenidir.

$$M_i = \overline{m_i} \qquad m_i = \overline{M_i}$$

# Çarpımlar Toplamı Gösterilim

- Örnek: $F_1 (x,y,z) = m_1 + m_4 + m_7$
- $F_1 = \overline{x}\ \overline{y}\ z + x\ \overline{y}\ \overline{z} + x\ y\ z$

| x y z | index | $m_1$ | + | $m_4$ | + | $m_7$ | $= F_1$ |
|---|---|---|---|---|---|---|---|
| 0 0 0 | 0 | 0 | + | 0 | + | 0 | = 0 |
| 0 0 1 | 1 | 1 | + | 0 | + | 0 | = 1 |
| 0 1 0 | 2 | 0 | + | 0 | + | 0 | = 0 |
| 0 1 1 | 3 | 0 | + | 0 | + | 0 | = 0 |
| 1 0 0 | 4 | 0 | + | 1 | + | 0 | = 1 |
| 1 0 1 | 5 | 0 | + | 0 | + | 0 | = 0 |
| 1 1 0 | 6 | 0 | + | 0 | + | 0 | = 0 |
| 1 1 1 | 7 | 0 | + | 0 | + | 1 | = 1 |

# Çarpımlar Toplamı Örneği

- F(A, B, C, D, E) = $m_2 + m_9 + m_{17} + m_{23}$
- F(A, B, C, D, E) = $A'B'C'DE'+A'BC'D'E+AB'C'D'E+AB'CDE$

# Toplamlar Çarpımı Örneği

- Örnek:

$F_1 (x,y,z) = M_0 \cdot M_2 \cdot M_3 \cdot M_5 \cdot M_6$

$$F_1 = (x+y+z)\cdot(x+\overline{y}+z)\cdot(x+\overline{y}+\overline{z})$$
$$\cdot(\overline{x}+y+\overline{z})\cdot(\overline{x}+\overline{y}+z)$$

| x y z | i | $M_0 \cdot M_2 \cdot M_3 \cdot M_5 \cdot M_6 = F1$ |
|---|---|---|
| 0 0 0 | 0 | $0 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 0$ |
| 0 0 1 | 1 | $1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 1$ |
| 0 1 0 | 2 | $1 \cdot 0 \cdot 1 \cdot 1 \cdot 1 = 0$ |
| 0 1 1 | 3 | $1 \cdot 1 \cdot 0 \cdot 1 \cdot 1 = 0$ |
| 1 0 0 | 4 | $1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 1$ |
| 1 0 1 | 5 | $1 \cdot 1 \cdot 1 \cdot 0 \cdot 1 = 0$ |
| 1 1 0 | 6 | $1 \cdot 1 \cdot 1 \cdot 1 \cdot 0 = 0$ |
| 1 1 1 | 7 | $1 \cdot 1 \cdot 1 \cdot 1 \cdot 1 = 1$ |

# Toplamlar Çarpımı Örneği

- $F(A,B,C,D) = M_3 \cdot M_8 \cdot M_{11} \cdot M_{14}$
- F(A,B,C,D)=

(A+B+C′+D′)(A′+B+C+D)(A′+B+C′+D′)(A′+B′+C′+D)

# Çarpımlar Toplamı Gösterilim

- Her Boole fonksiyonu çarpımlar toplamı ile gösterilebilir.
  - Kullanılan çarpım terimleri doğruluk tablosundaki 1"lere karşılık düşer.
  - Çarpımlar toplamı şeklinde gösterilmemiş Boole fonksiyonlarında bütün terimleri değişkenlerin hepsi görülecek şekilde genişletmek gerekir. Bu eksik olan terim v ise terimi ( $v+\overline{v}$ ) ile çarpılarak yapılır.
- Örnek: $f = x + \overline{x}\ \overline{y}$ fonksiyonunun çarpımlar toplamı gösterilimini bulunuz.
  - Terimleri genişlet $f = x(y+\overline{y}) + \overline{x}\ \overline{y}$
  - Terimleri dağıt: $f = xy + x\overline{y} + \overline{x}\,\overline{y}$
  - Çarpımlar toplamı şeklinde göster: $f = m_3 + m_2 + m_0$

# Çarpımlar Toplamı Gösterilim Örneği

- Örnek: $F = A + \overline{B}\,C$
- Üç değişken var: A, B, C
- Terimler eksik değişkenler ile genişletilir:

$$
\begin{aligned}
F &= A(B + B')(C + C') + (A + A')\,B'C \\
&= ABC + ABC' + AB'C + AB'C' + AB'C + A'B'C \\
&= ABC + ABC' + AB'C + AB'C' + A'B'C \\
&= m_7 + m_6 + m_5 + m_4 + m_1 \\
&= m_1 + m_4 + m_5 + m_6 + m_7
\end{aligned}
$$

# Çarpımlar Toplamının Kısa Gösterilimi

- Önceki örnekte $F = A + \overline{B}C$ ile başladık.
- $F = m_1+m_4+m_5+m_6+m_7$ bulduk.
- Bu kısa olarak aşağıdaki gibi gösterilebilir:

$$\mathbf{F(A,B,C) = \Sigma m(1,4,5,6,7)}$$

# Toplamlar Çarpımı Gösterilimi

- Her Boole fonksiyonu toplamlar çarpımı ile gösterilebilir.
  - Kullanılan toplam terimleri doğruluk tablosundaki 0"lara karşılık düşer.
  - Toplamlar çarpımı şeklinde gösterilmemiş Boole fonksiyonlarında bütün terimleri değişkenlerin hepsi görülecek şekilde genişletmek gerekir. Bu eksik olan terim v ise terimi ($v \cdot \bar{v}$ ) ile toplanarak yapılır.
- Örnek: $f(x, y, z) = x + \bar{x}\,\bar{y}$ fonksiyonunun toplamlar çarpımı ifadesini bulunuz.
  - Dağılma özelliğini kullan
    $$x + \bar{x}\,\bar{y} = (x + \bar{x})(x + \bar{y}) = 1 \cdot (x + \bar{y}) = x + \bar{y}$$
  - Eksik olan değişken z’yi ekle
    $$x + \bar{y} + z \cdot \bar{z} = (x + \bar{y} + z)(x + \bar{y} + \bar{z})$$
  - Toplamlar çarpımı olarak göster:

    $f = M_2 \cdot M_3$

# Toplamlar Çarpımı Örneği

- Aşağıdaki fonksiyonun toplamlar çarpımı gösterilimini bulunuz.

$$f(A, B, C) = A\,\overline{C} + B\,C + \overline{A}\,\overline{B}$$

$f=(AC'+BC+A')\ (AC'+BC+B')$

$f=((AC'+B)(AC'+ C)+A')((AC'+B)(AC'+ C)+B')$

$f=((A+B)(C'+B)(A+C)(C'+ C)+A')((A+B)(C'+B)(A+C)(C'+C)+B')$

$f=((A+B)(C'+B)(A+C)+A')\ ((A+B)(C'+B)(A+C)+B')$

$f=(A+B+A')(C'+B+A')(A+C+A')(A+B+B')(C'+B+B')(A+C+B')$

$f=(A'+B +C')(A+B'+C)$

$f = M_5 \cdot M_2$

# Fonksiyonların Tümleyenleri

- Çarpımlar toplamı ile gösterilen bir fonksiyonun tümleyeni çarpımlar toplamında görünmeyen terimler kullanılarak ifade edilir.
- Ya da aynı indislere sahip toplamlar çarpımı ifade ile gösterilir.
- Örnek: $F(x, y, z) = \Sigma_m(1,3,5,7)$

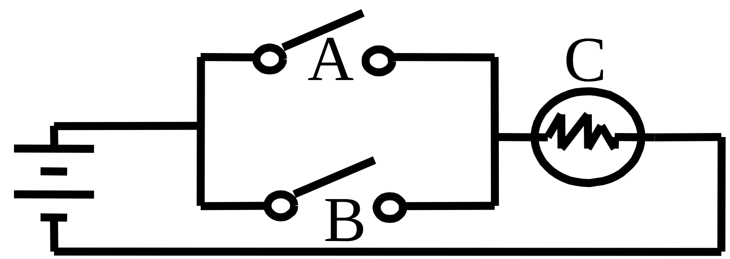
$$\overline{F}(x, y, z) = \Sigma_m(0,2,4,6)$$

$$\overline{F}(x, y, z) = \Pi_M(1,3,5,7)$$

# Boole Fonksiyonlarının Anahtar Devreleri İle Gerçeklenmesi

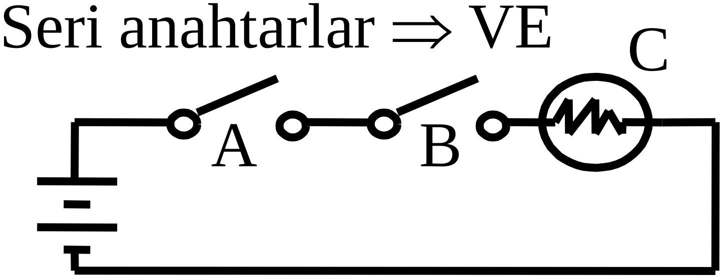
- Anahtarları Kullanarak
  - Girişler için:
    - lojik 1 <u>anahtar kapalı</u>
    - lojik 0 <u>anahtar açık</u>
  - Çıkışlar için:
    - lojik 1 <u>ışık açık</u>
    - lojik 0 <u>ışık kapalı</u>
  - TÜMLEME
    - lojik 1 <u>anahtar açık</u>
    - lojik 0 <u>anahtar kapalı</u>

Paralel Anahtarlar $\Rightarrow$ VEYA



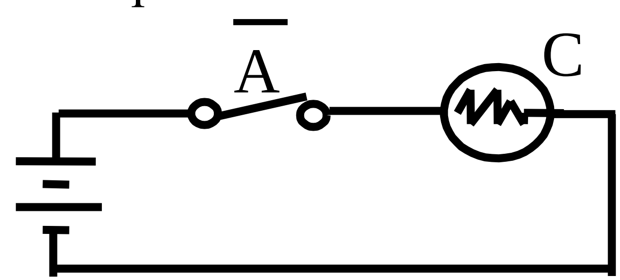

C=A VEYA B

Seri anahtarlar $\Rightarrow$ VE




C=A VE B

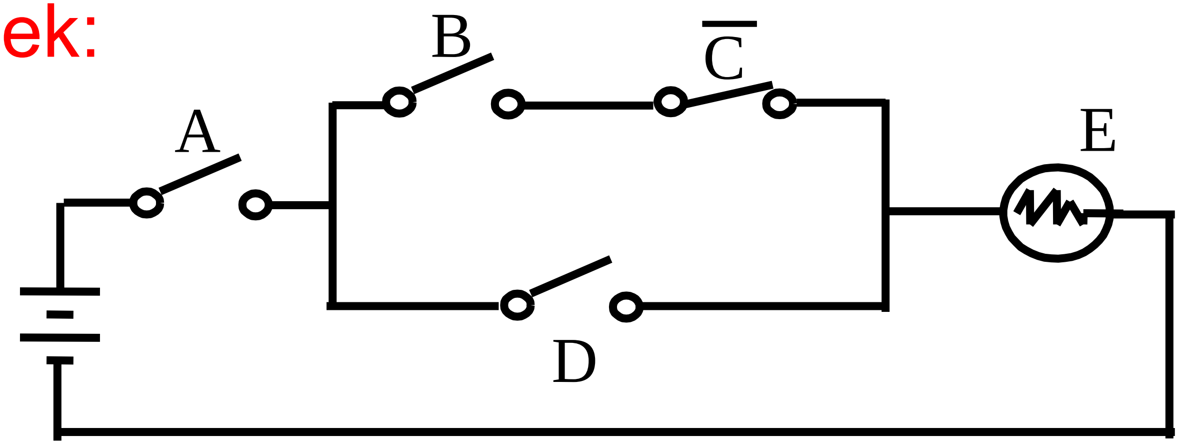
Normalde kapalı anahtar $\Rightarrow$ TÜMLEME




$C=\overline{A}$

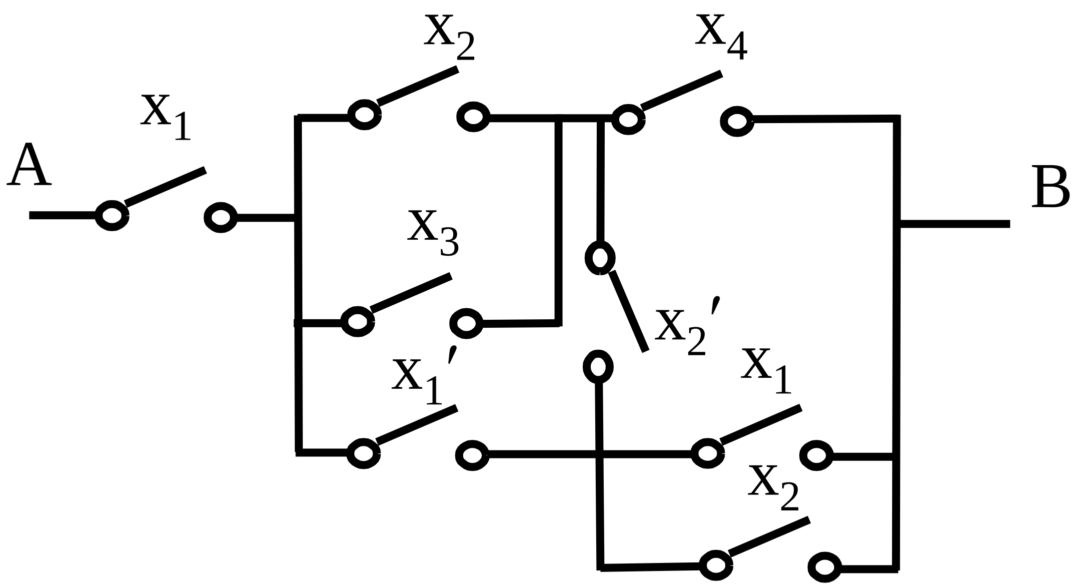
# Boole Fonksiyonlarının Anahtar Devreleri İle Gerçeklenmesi

- Örnek:




- Işık (E = 1) ise açıktır. (E = 0) ise kapalıdır.
  - Yol fonksiyonlarının toplamı:
    - $f(A, B, C, D) = ABC' + AD$
  - Kesitleme fonksiyonlarının çarpımı:
    - $f(A, B, C, D) = A (B+D) (C'+D)$

# Örnek: $f_{AB}$=?

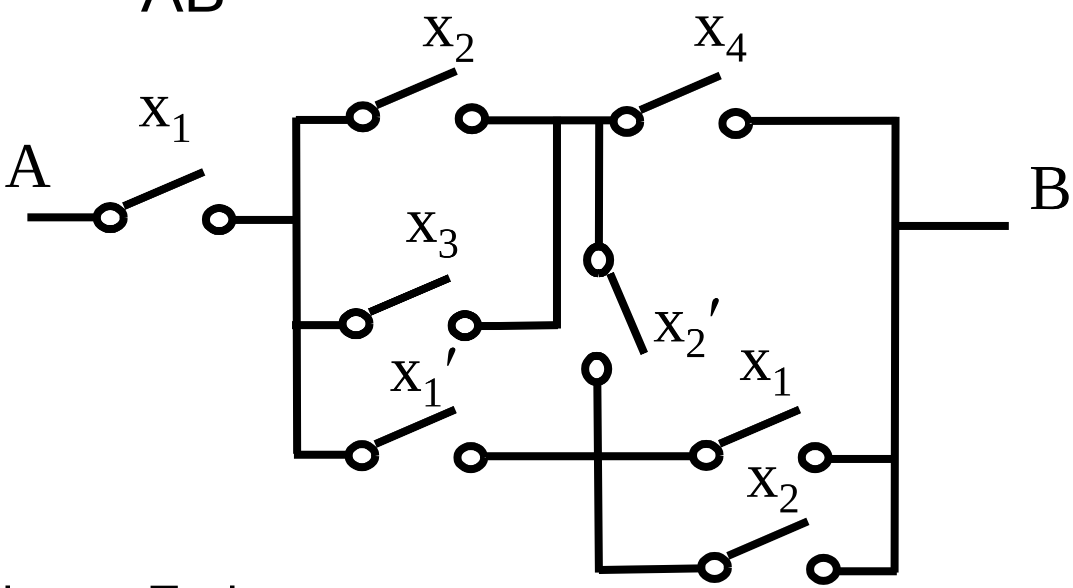



$f_{AB}=\Sigma_m(10,11,13,15)$

$f_{AB}=\Pi_M(0,1,2,3,4,5,6,7,8,9,12,14)$

| $x_1$ | $x_2$ | $x_3$ | $x_4$ | $f_{AB}$ |
|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 0 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 |
| 0 | 0 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 |
| 1 | 1 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 0 | 0 |
| 1 | 1 | 1 | 1 | 1 |

# Örnek: $f_{AB}$=?




Yol Fonksiyonlarının Toplamı:

$$f_{AB} = x_1x_2x_4+x_1x_2x_2'x_1+x_1x_2x_2'x_2+x_1x_3x_4+x_1x_3x_2'x_1+x_1x_3x_2'x_2+x_1x_1'x_2'x_4+x_1x_1'x_1+x_1x_1'x_2$$
$$= x_1x_2x_4+x_1x_3x_4+x_1x_3x_2'$$

Kesitleme Fonksiyonlarının Çarpımı:

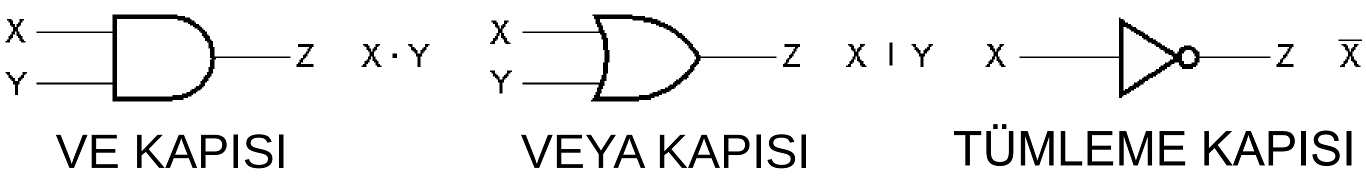
$$f_{AB} = x_1(x_2+x_3+x_1')(x_2+x_3+x_2'+x_1+x_2)(x_4+x_2'+x_1')(x_4+x_1+x_2)$$
$$= x_1(x_2+x_3+x_1')(x_4+x_2'+x_1')(x_4+x_1+x_2)$$

# Lojik Kapılar

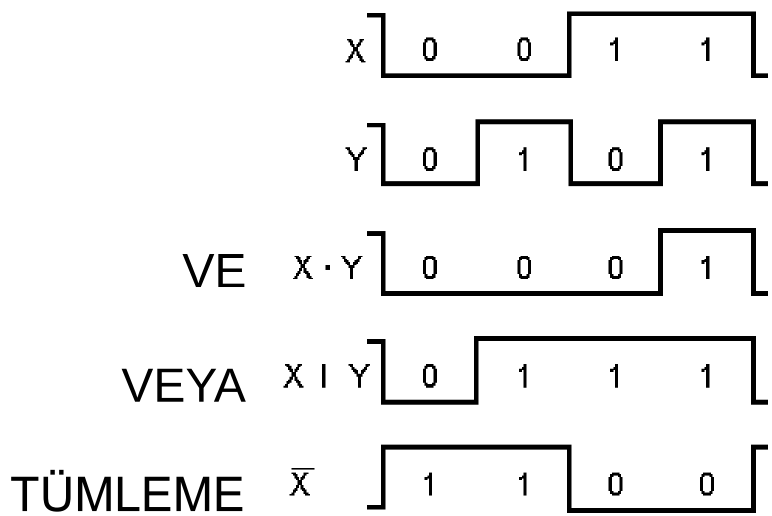
- İlk bilgisayarlarda anahtarlar röleler tarafından kontrol edilen elektromanyetik alanlar yardımı ile açılıp kapanıyordu. Anahtarlar da akım yollarını açıp – kapamada kullanılıyorlardı.
- Daha sonra vakum tüpleri akım yollarını açıp kapamada rölelerin yerini aldılar.
- Günümüzde tranzistörler elektronik anahtarlar olarak kullanılmaktadır.

# Lojik Kapılar ve sembolleri

- Lojik kapıların özel sembolleri vardır.
- Davranış biçimleri aşağıdaki gibidir.



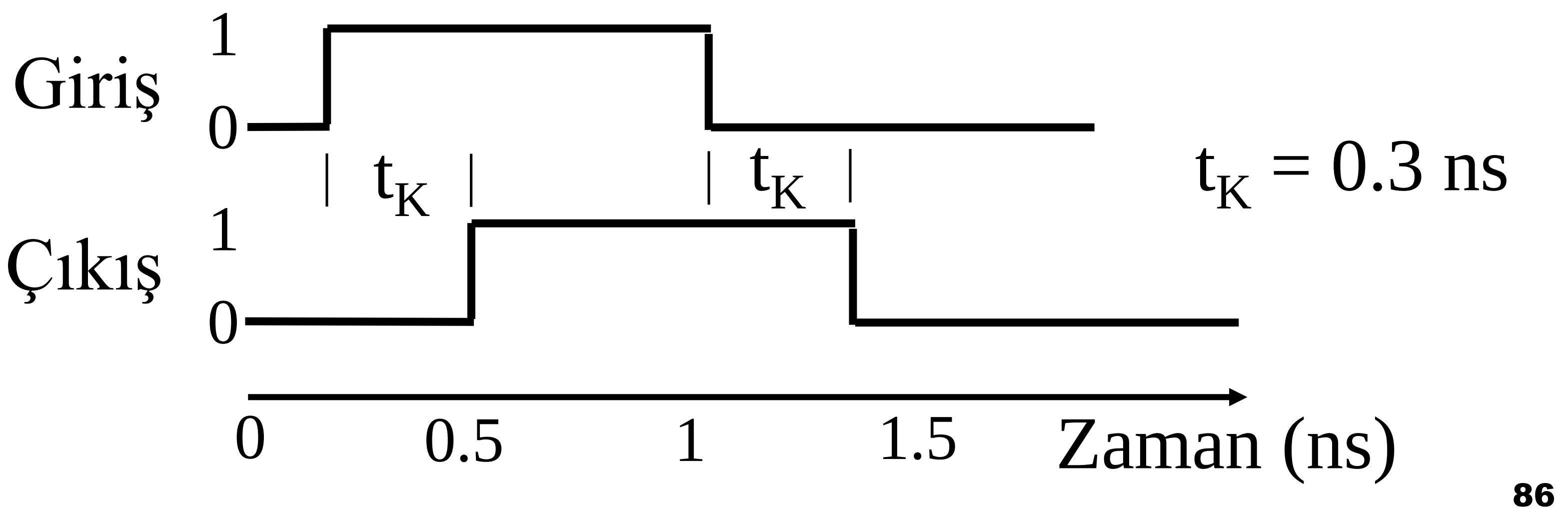
(a) Grafik Semboller



(b) Zamanlama Diyagramı

# Kapı Gecikmesi

- Fiziksel kapılarda bir veya birden fazla giriş değiştiğinde çıkış hemen değişmez.
- Girişlerden herhangi birindeki değişimden sonra çıkıştaki değişime kadar geçen süreye kapı gecikmesi denir ve $t_K$ ile gösterilir.



# Lojik Diyagramlar ve İfadeler

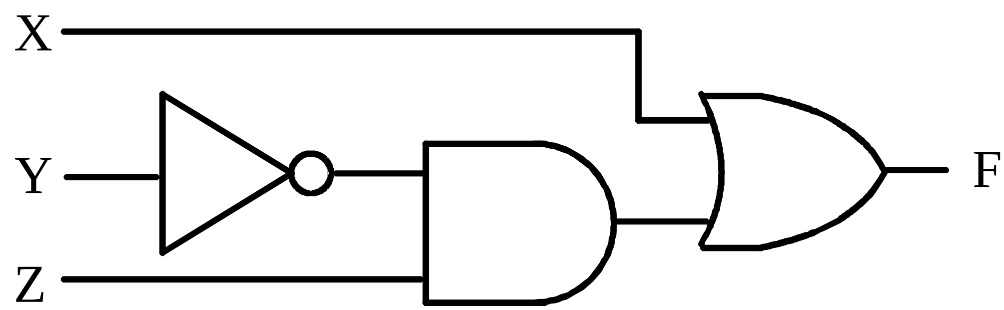
Doğruluk Tablosu

| X Y Z | $F = X + \overline{Y} \cdot Z$ |
|---|---|
| 0 0 0 | 0 |
| 0 0 1 | 1 |
| 0 1 0 | 0 |
| 0 1 1 | 0 |
| 1 0 0 | 1 |
| 1 0 1 | 1 |
| 1 1 0 | 1 |
| 1 1 1 | 1 |

Fonksiyon

$$F = X + \overline{Y}\ Z$$

Lojik Diyagram




- Boole fonksiyonları, doğruluk tabloları ve lojik diyagramlar aynı fonksiyonu gösterir.
- Her fonksiyonun doğruluk tablosu tektir. Ancak Boole fonksiyonu ve lojik diyagramı tek değildir. Bu gerçeklemede esneklik sağlar.

# Çarpımlar Toplamı Gösteriliminin İndirgenmesi

- Örnek: $\mathbf{F(A,B,C)=\Sigma m(1,4,5,6,7)}$
- Çarpımlar toplamı ifade:

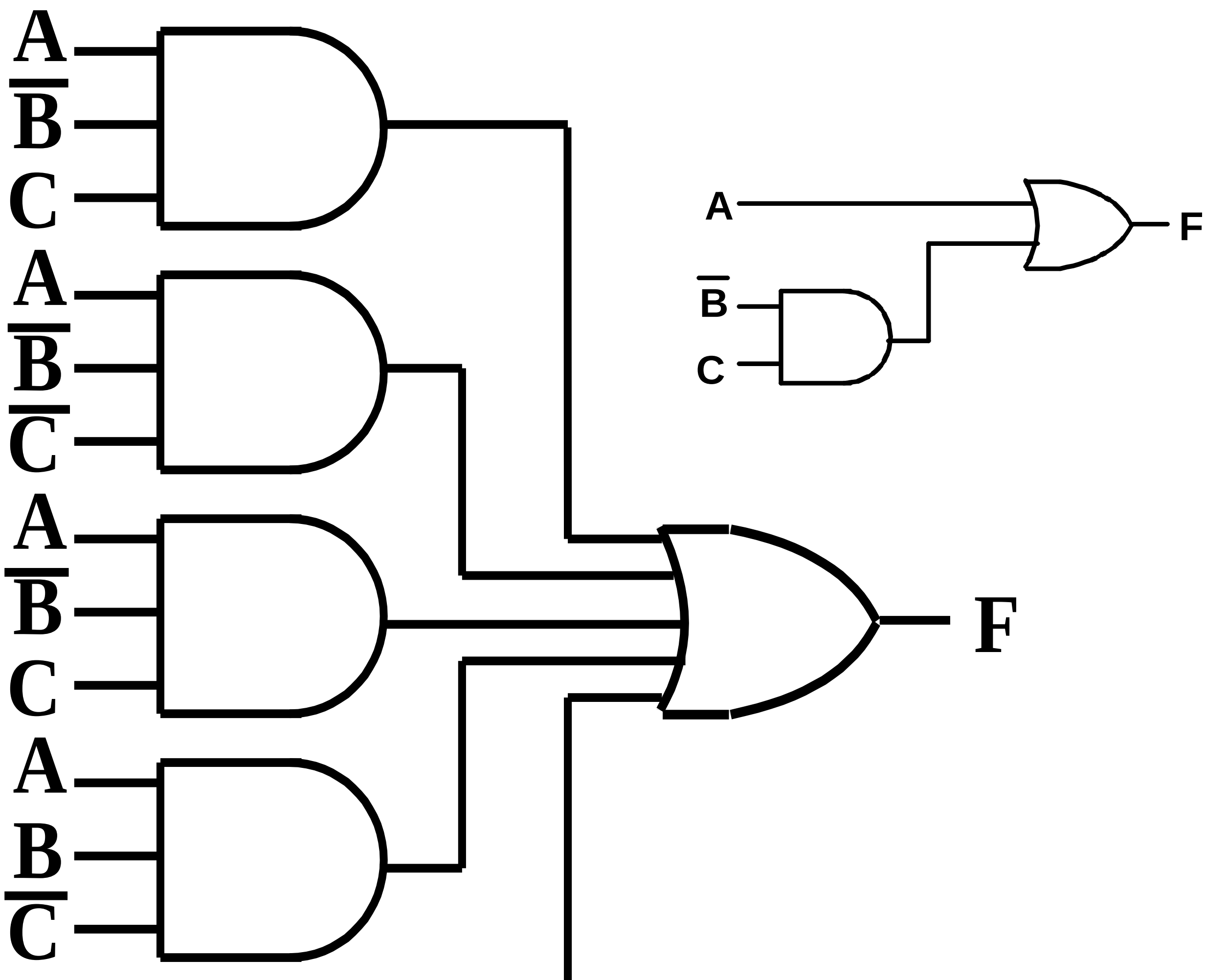
  $F = A'B'C + AB'C' + AB'C + ABC' + ABC$
- İndirgeme:

$$
\begin{aligned}
F &= A'B'C + A\,(B'C' + B'C + BC' + BC) \\
&= A'B'C + A\,(B' + B)(C' + C) \\
&= A'B'C + A\;1\;1 \\
&= B'C + A
\end{aligned}
$$

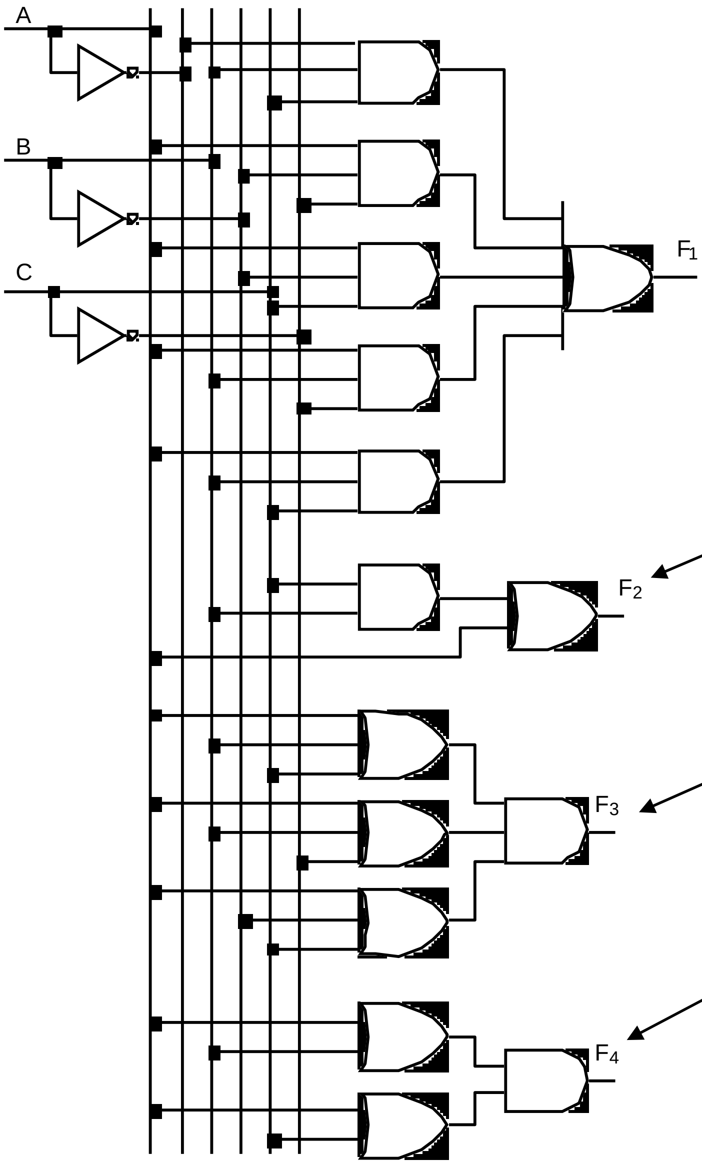
- İndirgenmiş ifade 3 değişken içerir.

# Çarpımlar Toplamı İfadenin VE/VEYA İki Seviyeli Gerçeklemesi

- F’in iki ayrı gerçeklemesi



# F Fonksiyonunun 4 farklı gerçeklemesi




**Çarpımlar toplamı**

**İndirgenmiş çarpımlar toplamı**

**Toplamlar çarpımı**

**İndirgenmiş toplamlar çarpımı**