

# BSM206 Mantıksal Devre Tasarımı

---

## 7. Hafta – BCD Toplayıcı, Büyüklük Karşılaştırıcısı, Kod Çözücüler

Dr. Öğr. Üyesi Onur ÇAKIRGÖZ  
onurcakirgoz@bartin.edu.tr

# ANAHAAT

---

- Onlu Toplayıcı (Decimal Adder)
- BCD Toplayıcı
- Büyüklük Karşılaştırıcısı (Magnitude Comparator)
- Kod Çözücüler (Decoders)
- Kod Çözücüler ile Kombinezonal Lojik Uygulaması

# Onlu Toplayıcı (Decimal Adder)

---

- Hesap makinelerinde veya bilgisayarlarda aritmetik işlemler onlu sayı sisteminde gerçekleştirilir.
- Fakat, buradaki onlu sayı sistemi, ikili kodlanmış formdaki onlu gösterimi kullanmaktadır.
- Dolayısıyla, ilgili dijital devrenin girişleri kodlanmış onlu sayılar olmalıdır.
- Her onlu haneyi (0..9) kodlamak için 4 bit gereklidir. (3 bit ?)
- Onlu bir toplayıcı devresi en az 9 giriş ve 5 çıkışa sahip olmalıdır:
  - Girişler: 4 bitlik iki ayrı sayı ve giriş eldesi
  - Çıkışlar: 4 bitlik sayı ve çıkış eldesi
- Farklı kodlamalara bağlı olarak farklı devreler tasarlanabilir.



# Onlu Toplayıcı (Decimal Adder)

---

- 9 giriş 5 çıkışlı bir kombinezonal devre için  $2^9 = 512$  satırlı bir doğruluk tablosu gerekir.
- Her bir ikili kod altı geçersiz kombinasyona (10, 11, 12, 13, 14 ve 15 desimal sayıları) sahiptir.
- Bilgisayarda oluşturulmuş bir tablo yöntemiyle devre için basitleştirilmiş Boole fonksiyonları elde edilebilir ve devre kapılarla gerçekleştirilebilir.
- **Alternatif yol:** Tam toplayıcı devreleri (ikili toplayıcı) kullanmak.

# BCD Toplayıcı

---

- BCD = Binary Coded Decimal (ikili kodlanmış onlu sayı)
- Bir BCD toplayıcı iki adet BCD hanesini paralel olarak toplayabilen ve yine BCD biçiminde bir toplam sonucu üreten bir devredir.
- Giriş eldesi de gözönünde bulundurularak, BCD'de iki onlu hanenin toplandığını varsayalım.
- Giriş hanesi en fazla 9 dur. Elde girişini ise 1 olarak alırsak, toplam çıkışı  $9 + 9 + 1 = 19$  dan büyük olamaz.
- İki BCD hanesini bir dört bitlik ikili toplayıcıya uyguladığımızı varsayalım.
- İkili toplayıcı, toplama işlemini ikili formda oluşturacak ve 0 ila 19 arasında bir sayı üretecektir.



# BCD Toplayıcı

## • (İkili Toplam – BCD Toplam) Dönüşüm Tablosu:

- Yandaki tablo bir doğruluk tablosu değildir.
- İkili toplam sütunundaki ikili sayıların BCD toplam sütunundaki BCD sayılarına dönüşümü için **kural** bulunmalıdır.
- 4-bitlik BCD kodundaki en büyük sayı 1001 dir
- Normalde 10 desimal sayısı BCD kodunda 8 bitle temsil edilir.

Binary Sum					BCD Sum					Decimal
K	Z <sub>8</sub>	Z <sub>4</sub>	Z <sub>2</sub>	Z <sub>1</sub>	C	S <sub>8</sub>	S <sub>4</sub>	S <sub>2</sub>	S <sub>1</sub>	
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	2
0	0	0	1	1	0	0	0	1	1	3
0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	1	0	0	1	0	1	5
0	0	1	1	0	0	0	1	1	0	6
0	0	1	1	1	0	0	1	1	1	7
0	1	0	0	0	0	1	0	0	0	8
0	1	0	0	1	0	1	0	0	1	9
0	1	0	1	0	1	0	0	0	0	10
0	1	0	1	1	1	0	0	0	1	11
0	1	1	0	0	1	0	0	1	0	12
0	1	1	0	1	1	0	0	1	1	13
0	1	1	1	0	1	0	1	0	0	14
0	1	1	1	1	1	0	1	0	1	15
1	0	0	0	0	1	0	1	1	0	16
1	0	0	0	1	1	0	1	1	1	17
1	0	0	1	0	1	1	0	0	0	18
1	0	0	1	1	1	1	0	0	1	19

# BCD Toplayıcı

---

- Tablodaki 5 bit, devrenin çıkışlarıdır. Toplam (sum) bitleri ve elde (C) çıkışı.
- Tablo içeriği incelendiğinde, ikili toplamamanın sonucu 1001'e eşit veya ondan küçükse, dönüşüme gerek olmadığı görülür. Yani, sayı yine BCD kodundadır. (Ortadaki çizginin yukarısı)
- İkili toplamamanın sonucu 1001'den büyük olduğunda (desimal 10 veya yukarısı – ortadaki çizginin alt tarafı), doğru BCD gösterimini elde etmek için 2 dönüşüm yapılmalıdır:
  - 6 sayısı (ikili eşdeğeri: 0110) toplama sonucuna ilave edilmelidir.
  - C (carry) çıkışı 1 yapılmalıdır.
- Çizginin yukarısıyla altını birbirinden ayırt etmek için, alt tarafı sağlayan, yukarıyı sağlamayan kurallar bulunmalıdır.



# BCD Toplayıcı

- Tabloya bakılarak, toplama sonucunda gereken düzeltmeleri sezecek bir lojik devre tasarlanabilir. (2 farklı durum var.)
- **1. durum:** Çıkış eldesi  $K = 1$  ise bir düzeltme işlemi gereklidir.
- **2. durum:** 1010'dan 1111'e kadar düzeltme gerektiren 6 kombinasyonun  $Z_8$  konumunda 1 vardır. (ortak özellikleri)
- 1000 ve 1001 kombinasyonlarının  $Z_8$  i de 1'dir ve bunları düzeltme gerektirenlerden ayırabilmek için  $Z_4$  veya  $Z_2$  den birinin ayrıca 1 olmasına bakılmalıdır.

Binary Sum					BCD Sum					Decimal
K	Z <sub>8</sub>	Z <sub>4</sub>	Z <sub>2</sub>	Z <sub>1</sub>	C	S <sub>8</sub>	S <sub>4</sub>	S <sub>2</sub>	S <sub>1</sub>	
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	2
0	0	0	1	1	0	0	0	1	1	3
0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	1	0	0	1	0	1	5
0	0	1	1	0	0	0	1	1	0	6
0	0	1	1	1	0	0	1	1	1	7
0	1	0	0	0	0	1	0	0	0	8
0	1	0	0	1	0	1	0	0	1	9
0	1	0	1	0	1	0	0	0	0	10
0	1	0	1	1	1	0	0	0	1	11
0	1	1	0	0	1	0	0	1	0	12
0	1	1	0	1	1	0	0	1	1	13
0	1	1	1	0	1	0	1	0	0	14
0	1	1	1	1	1	0	1	0	1	15
1	0	0	0	0	1	0	1	1	0	16
1	0	0	0	1	1	0	1	1	1	17
1	0	0	1	0	1	1	0	0	0	18
1	0	0	1	1	1	1	0	0	1	19



# BCD Toplayıcı

---

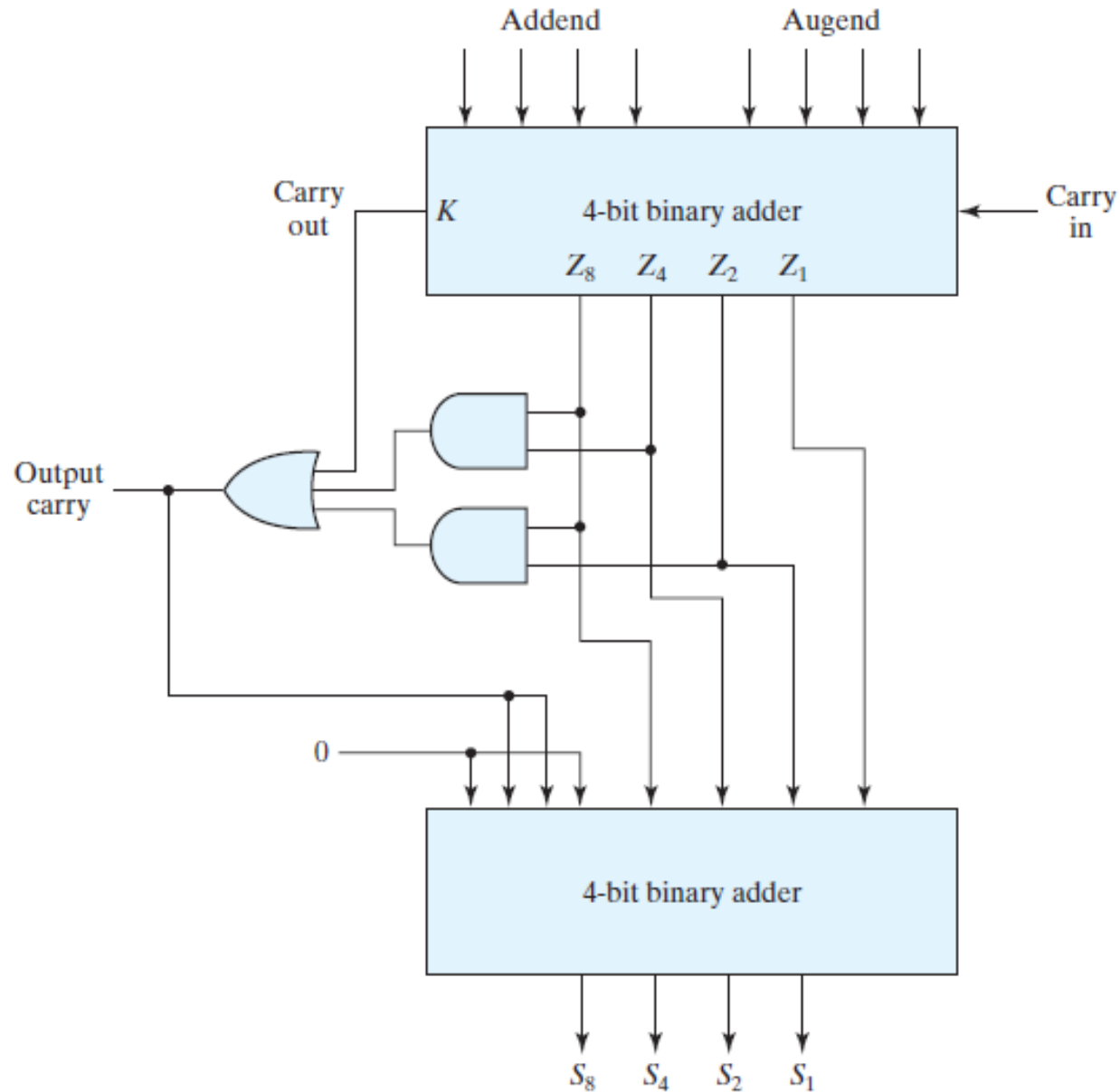
- Gereken düzeltmeler için koşullar ve çıkış eldesi aşağıdaki Boole fonksiyonuyla ifade edilir:

$$C = K + Z_8Z_4 + Z_8Z_2$$

- Dolayısıyla,  $C = 1$  olduğunda ikili toplam sonucuna 0110 eklenmeli ve bir sonraki kademe için çıkış eldesi üretilmelidir.
- Devrede, düzeltme işlemi için, ikili toplam sonucuna 0110 değerini ekleyecek ikinci bir dört bitlik toplayıcı kullanılır.
- Çıkış eldesi (C) sıfır olduğunda, ikili toplama 0000 eklenir fakat sonuç değişmez.
- Çıkış eldesi (C) bir olduğunda, devrenin alt kısmında yer alan 4-bitlik toplayıcı tarafından ikili toplama 0110 eklenir.

# BCD Toplayıcı

- K: Elde
- $Z_8$ : ikili toplamdaki 8 ağırlığı
- $Z_4$ : ikili toplamdaki 4 ağırlığı
- $Z_2$ : ikili toplamdaki 2 ağırlığı
- $Z_1$ : ikili toplamdaki 1 ağırlığı
- Not: Bu devre BCD formunda kodlanmış iki onlu haneyi (iki tek basamaklı desimal sayıyı) toplamaktadır.



# Büyüklik Karşılaştırıcısı (Magnitude Comparator)

---

- İki sayıyı karşılaştırma işlemi:
  - $A > B$  (A, B'den büyük mü)
  - $A < B$  (A, B'den küçük mü)
  - $A = B$  (A, B'ye eşit mi)olduğunu belirler.
- Bir büyüklik (genlik) karşılaştırıcısı A ve B gibi iki sayıyı karşılaştıran ve onların bağıl genliklerini elde eden bir kombinezonal devredir.
- Karşılaştırma sonucu, devrenin üç çıkışı tarafından belirlenir ve her bir çıkış yukarıdaki üç durumdan birini gösterir.
- n bitlik iki sayıyı karşılaştıran devre için  $2^{2n}$  içerikli bir doğruluk tablosu oluşturmak gerekir.



# Büyüklik Karşılaştırıcısı (Magnitude Comparator)

- Her biri 4 haneli olan A ve B gibi iki sayıyı gözönüne alalım:

$$A = A_3A_2A_1A_0$$

$$B = B_3B_2B_1B_0$$

- $A_3 = B_3$ ,  $A_2 = B_2$ ,  $A_1 = B_1$  ve  $A_0 = B_0$  ise iki sayı **eşittir**.
- Her bit çiftinin ( $A_i$  ve  $B_i$ ) **eşitlik** ilişkisi bir eşdeğer fonksiyonuyla lojik olarak ifade edilebilir:

$$x_i = A_iB_i + A_i'B_i' \quad i = 0, 1, 2, 3$$

- Bu fonksiyonda,  $i$  indisli bit çiftleri eşitse,  $x_i = 1$  olur.
- $A = B$  olması için, tüm  $x_i$  değişkenleri 1'e eşit olmalıdır. Bu, tüm değişkenlere bir VE işlemi uygulanmasını gerektirir:

$$(A = B) \text{ çıkışı} = x_3x_2x_1x_0$$

- $(A = B)$  çıkışı, yalnızca iki sayının tüm haneleri eşitse 1 olur.

# Büyüklik Karşılaştırıcısı (Magnitude Comparator)

---

- $(A > B)$  veya  $(A < B)$  durumunu bulmak için, en yüksek anlamlı konumdan başlayarak iki sayının hanelerinin bağıl genlikleri incelenir.
- İki hane birbirine eşit olduğunda bir sonraki daha düşük anlamlı hane karşılaştırılır.
- Bu karşılaştırma birbirine eşit olmayan haneye ulaşıncaya kadar devam ettirilir.
- Eşit olmayan hanede, A'nın biti 1, B'nin biti 0 ise  $A > B$  dir.
- Eşit olmayan hanede, A'nın biti 0, B'nin biti 1 ise  $A < B$  dir.
- $(A > B)$  veya  $(A < B)$  durumları, iki Boole fonksiyonuyla ifade edilebilir:

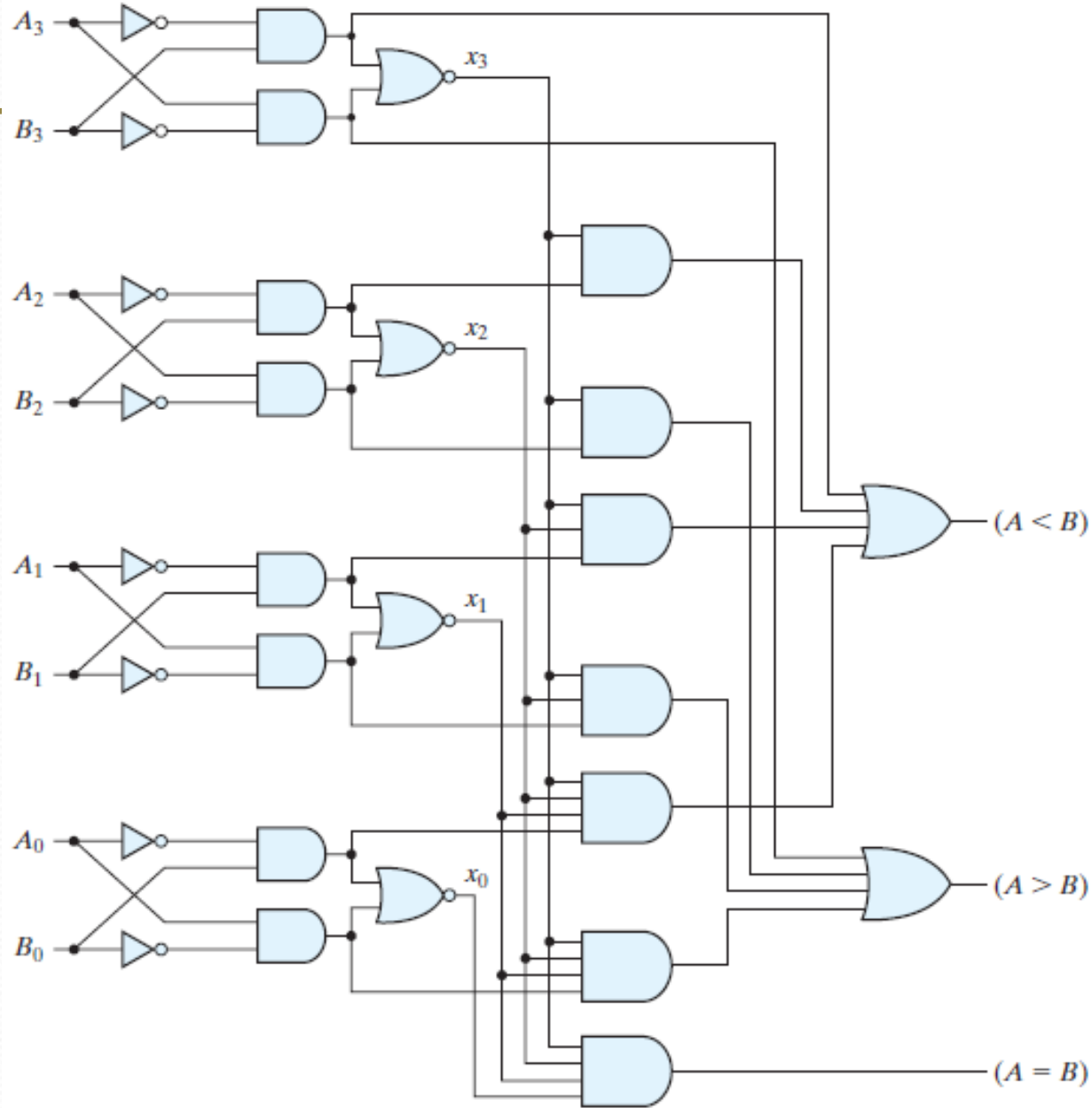
$$(A > B) = A_3B_3' + x_3A_2B_2' + x_3x_2A_1B_1' + x_3x_2x_1A_0B_0'$$

$$(A < B) = A_3'B_3 + x_3A_2'B_2 + x_3x_2A_1'B_1 + x_3x_2x_1A_0'B_0$$



# Büyüklik Karşılaştırıcısı (Magnitude Comparator)

- "Eşit" çıkışının üretildiği kapıları ( $x_3$ ,  $x_2$ ,  $x_1$ ,  $x_0$ ), "eşit olmayan" çıkışlar da kullanmıştır.
- Bu devre iki BCD sayının bağlı genliklerini karşılaştırmak için de kullanılabilir.
- Dört bitten daha büyük sayılar için genlik karşılaştırma devresi aynı mantıkla kolayca gerçekleştirilebilir.





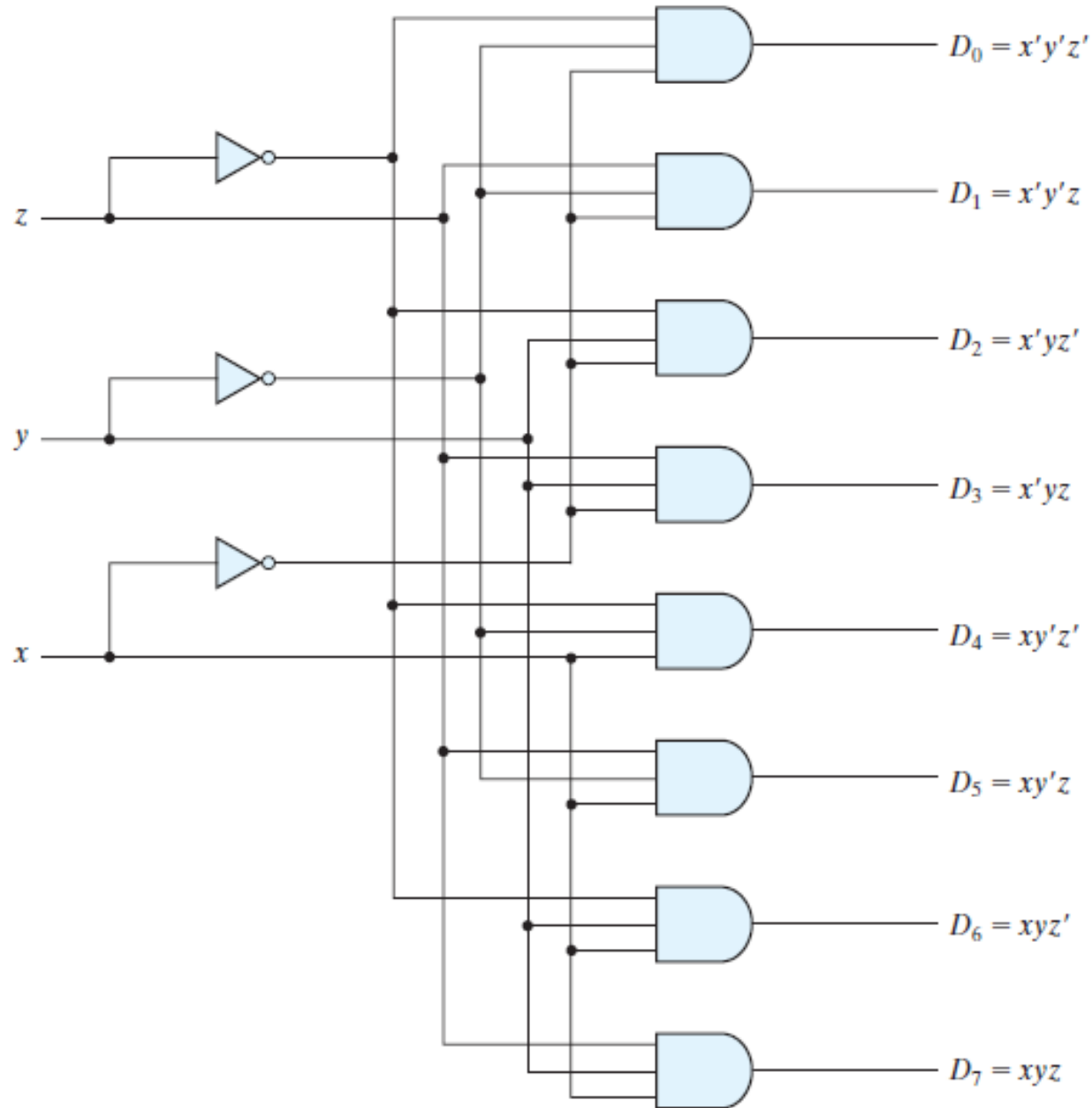
# Kod Çözücüler (Decoders)

---

- Ayırık bilgiler sayısal sistemlerde ikili kodlarla temsil edilir.
- $n$  bitlik ikili kod,  $2^n$  bağımsız kodlanmış bilgiyi temsil edebilir.
- Kod çözücü,  $n$  bitlik bir ikili bilgiyi maksimum  $2^n$  çıkış hattına dönüştüren bir kombinezonal devredir.
- $n$  bitlik kodu çözülmüş olan ayırık bilgiler içerisinde önemli olmayan veya kullanılmayan kombinasyonlar varsa, kod çözücünün çıkışı  $2^n$  çıkıştan daha az olacaktır.
- Burada,  $m \leq 2^n$  olmak üzere,  $n$ 'den  $m$ 'ye kod çözücüler işlenecektir.
- Bu kod çözücülerde  $n$  giriş değişkeninden  $2^n$  veya daha az sayıda terim üretilebilir.
- Kod çözücü ismi BCD'den 7 parçalıya kod çözücüde olduğu gibi, başka kod dönüştürme işlemleri için de kullanılmaktadır

# Kod Çözücüler (Decoders)

- Yanda, 3'ten 8'e bir kod çözücü devresi yer almaktadır.
- Devrede, 3 girişten 8 çıkışa kod çözülür ve her bir çıkış 3 giriş değişkeninin **minterimlerinden** birini temsil eder.
- Devrede sekiz VE kapısının her birinden bir **minterim** üretilir.
- Bu devrenin tipik bir uygulaması ikili sayı sisteminden sekizli sayı sistemine bir dönüşüm olabilir.



# Kod Çözücüler (Decoders)

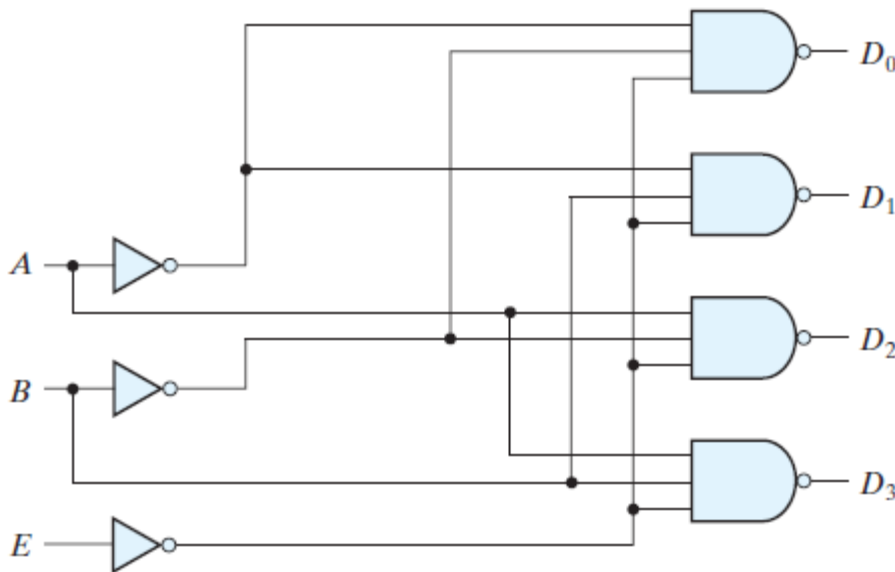
- Aşağıda 3'ten 8'e kod çözücünün doğruluk tablosu görülmektedir.
- Tablodan, herhangi bir anda **sadece** bir çıkışın 1' e eşit olduğu görülebilir.

Inputs			Outputs							
<i>x</i>	<i>y</i>	<i>z</i>	<i>D</i> <sub>0</sub>	<i>D</i> <sub>1</sub>	<i>D</i> <sub>2</sub>	<i>D</i> <sub>3</sub>	<i>D</i> <sub>4</sub>	<i>D</i> <sub>5</sub>	<i>D</i> <sub>6</sub>	<i>D</i> <sub>7</sub>
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1



# Kod Çözücüler (Decoders)

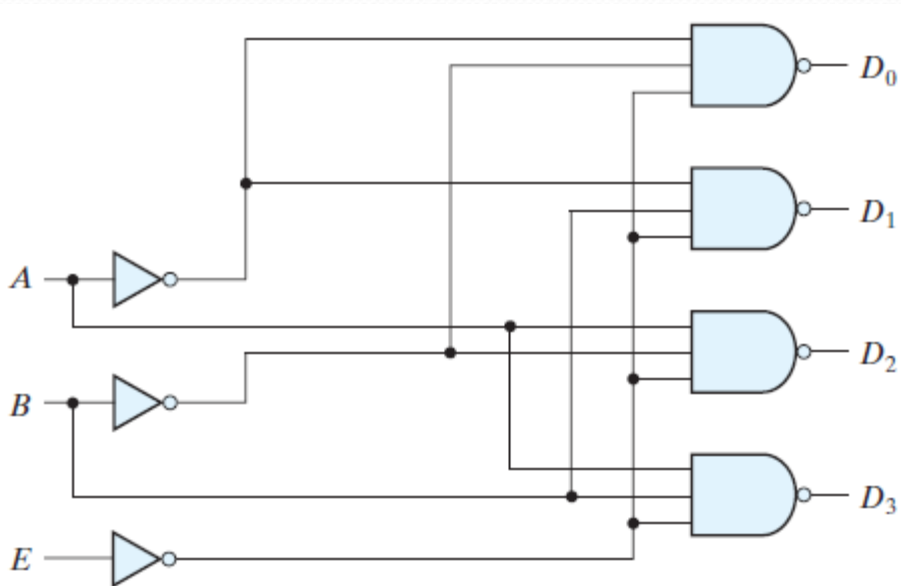
- Bazı kod çözücüler VEDEĞİL kapılarıyla oluşturulur.
- Bir VEDEĞİL kapısı tümleyen çıkışlı bir VE işlemini gerçekleştirdiğinden, kod çözücü minterimlerini tümleyen formda üretmek daha ekonomik olur.
- Çoğu tümdevre kod çözücüler, devrenin çalışmasını kontrol etmek için bir veya daha fazla izin girişi (Enable) içerir.



<i>E</i>	<i>A</i>	<i>B</i>	<i>D</i> <sub>0</sub>	<i>D</i> <sub>1</sub>	<i>D</i> <sub>2</sub>	<i>D</i> <sub>3</sub>
1	<i>X</i>	<i>X</i>	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

# Kod Çözücüler (Decoders)

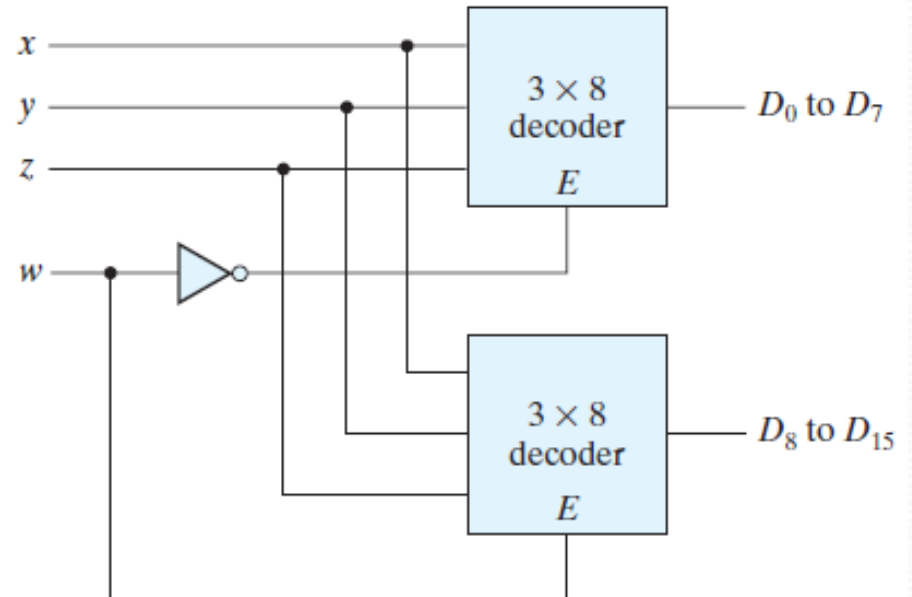
- VEDEĞİL kapılarıyla oluşturulmuş izin girişli bir 2'den 4'e kod çözücü aşağıda verilmiştir.
- E izin girişi 1 olduğunda çıkışlar?
- Devre, E izin girişi 0 olduğunda, tümlenmiş çıkışlı bir kod çözücü olarak çalışır.
- Çıkışlar hangi değere sahipse aktif (seçilmiş) olur?



<i>E</i>	<i>A</i>	<i>B</i>	<i>D</i> <sub>0</sub>	<i>D</i> <sub>1</sub>	<i>D</i> <sub>2</sub>	<i>D</i> <sub>3</sub>
1	<i>X</i>	<i>X</i>	1	1	1	1
0	0	0	0	1	1	1
0	0	1	1	0	1	1
0	1	0	1	1	0	1
0	1	1	1	1	1	0

# Kod Çözücüler (Decoders)

- İzin giriřli kod çözücüler daha büyük bir kod çözücü devre oluşturmak için bağlanabilir.
- Ařağıda, izin giriřli iki 3x8 kod çözücü 4x16 kod çözücüyü oluşturmak için bağlanmıřtır.
- $w = 0$  olduėunda üstteki kod çözücüye izin verilir ve alttaki yasaklanır. Altteki kod çözücü çıkıřlarının hepsi 0 olur. Yukarıdaki sekiz çıkıř ise 0000'dan  $(wx'y'z')$  0111'e  $(wxyz)$  kadar olan **minterimleri** üretir.
- $w = 1$  olduėunda alttaki kod çözücüye izin verilir ve üstteki yasaklanır. Üstteki kod çözücü çıkıřlarının hepsi 0 olur. Altteki sekiz çıkıř ise 1000'dan 1111'e kadar olan dört giriř deėiřkeninin **minterimlerini** üretir.





# Kod Çözücüler (Decoders)

---

- Tümdevrede yer alan **izin** girişi çok kullanışlı bir özelliktir.
- Genelde **izin uçları**, az giriş az çıkışlı lojik fonksiyonları çok giriş çok çıkışlı benzer fonksiyonlara genişletme amacıyla iki yada daha fazla tümdevrenin birbirine **bağlanmasını** sağlayan çok önemli bir özelliktir.

# Kod Çözücüler ile Kombinezonal Lojik Uygulaması

---

- Bir kod çözücü,  $n$  giriş değişkeninden  $2^n$  **minterim** oluşturur.
- Herhangi bir Boole fonksiyonu kanonik formda minterimlerin toplamı biçiminde ifade edilebildiğinden;
  - minterimleri üretmek için bir kod çözücü,
  - toplamı oluşturmak için de ekstradan bir VEYA kapısı kullanılabilir.
- Bu yolla,  $n$  giriş  $m$  çıkışlı herhangi bir kombinezonal devre;
  - $n$ 'den  $2^n$ 'ye kod çözücü ve
  - $m$  adet VEYA kapısından oluşabilir.
- **Kısacası, kod çözücüler herhangi bir kombinezonal devrenin gerçekleştirilmesinde kullanılabilir.**
- **Yani, kod çözücüler evrensel bir tasarım elemanıdır.**



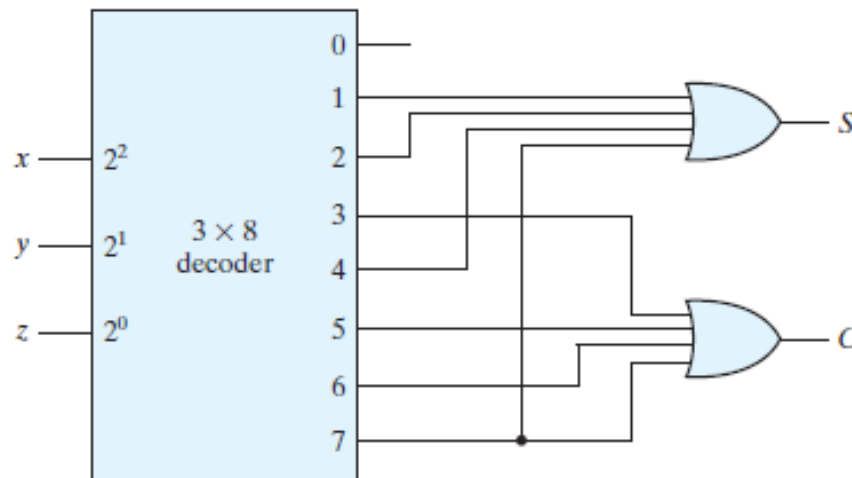
# Kod Çözücüler ile Kombinezonal Lojik Uygulaması - Örnek

- **Örnek:** Bir tam toplayıcıyı, bir kod çözücü ve iki VEYA kapısıyla gerçekleştirin.
- Tam toplayıcının doğruluk tablosundan, fonksiyonlar minterimlerin toplamı biçiminde elde edilir:

$$S(x,y,z) = \Sigma(1, 2, 4, 7)$$

$$C(x,y,z) = \Sigma(3, 5, 6, 7)$$

- İstenen gerçekleştirme üç giriş ve sekiz minterim içerdiğinden, 3x8 lik bir kod çözücü gerektirir.



x	y	z	C	S
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	0
1	1	0	1	0
1	1	1	1	1



# Kod Çözücüler ile Kombinezonal Lojik Uygulaması

---

- Kod çözücüyü herhangi bir kombinezonal devrenin gerçekleştirilmesinde kullanmak her zaman mantıklı mıdır?
- Her zaman kullanmak mantıklı olmayabilir. Diğer olası tasarımlarla mukayese ederek hangisinin maliyeti daha az ise o tasarım yaklaşımı kullanılmalıdır.