Sayısal Sistemler-H3CD2

Boolean Cebri

Dr. Meriç Çetin

versiyon250925

Bu derste öğreneceklerimiz

2 Boolean Algebra and Logic Gates

- İkili mantık günümüzün tüm dijital bilgisayarlarında ve cihazlarında kullanıldığından, onu uygulayan devrelerin **maliyeti**, bilgisayar mühendisleri, elektrik mühendisleri yada devre tasarımcıları tarafından ele alınan önemli bir konudur.
- Daha basit ve daha ucuz, ancak eşdeğer bir devreyi gerçekleştirmek, tasarımın toplam maliyetini düşürmede büyük kazançlar sağlayabilir.

- Devreleri basitleştiren matematiksel yöntemler, öncelikle **Boole cebirine** dayanır.
- Bu bölüm, basit devreleri ve milyonlarca mantık kapısını içeren karmaşık devreleri optimize etmek için yazılım araçları tarafından kullanılan algoritmaların amacını anlamanızı sağlayacak Boole cebirini bir özet şeklinde sunmaktadır.

- Boole cebri, diğer tümdengelimli matematiksel sistemler gibi, bir dizi öğe, bir dizi operatör ve bir dizi kanıtlanmamış aksiyom veya varsayımla tanımlanabilir.

Temel teoremler ve Boolean cebrinin özellikleri

Table 2.1
Postulates and Theorems of Boolean Algebra

Postulate 2	(a)	$x + 0 = x$	(b)	$x \cdot 1 = x$
Postulate 5	(a)	$x + x' = 1$	(b)	$x \cdot x' = 0$
Theorem 1	(a)	$x + x = x$	(b)	$x \cdot x = x$
Theorem 2	(a)	$x + 1 = 1$	(b)	$x \cdot 0 = 0$
Theorem 3, involution		$(x')' = x$		
Postulate 3, commutative	(a)	$x + y = y + x$	(b)	$xy = yx$
Theorem 4, associative	(a)	$x + (y + z) = (x + y) + z$	(b)	$x(yz) = (xy)z$
Postulate 4, distributive	(a)	$x(y + z) = xy + xz$	(b)	$x + yz = (x + y)(x + z)$
Theorem 5, DeMorgan	(a)	$(x + y)' = x'y'$	(b)	$(xy)' = x' + y'$
Theorem 6, absorption	(a)	$x + xy = x$	(b)	$x(x + y) = x$

THEOREM 1(a): $x + x = x$.

Statement	Justification
$x + x = (x + x) \cdot 1$	postulate 2(b)
$= (x + x)(x + x')$	5(a)
$= x + xx'$	4(b)
$= x + 0$	5(b)
$= x$	2(a)

THEOREM 1(b): $x \cdot x = x$.

Statement	Justification
$x \cdot x = xx + 0$	postulate 2(a)
$= xx + xx'$	5(b)
$= x(x + x')$	4(a)
$= x \cdot 1$	5(a)
$= x$	2(b)

THEOREM 2(a): $x + 1 = 1.$

Statement	Justification
$x + 1 = 1 \cdot (x + 1)$	postulate 2(b)
$= (x + x')(x + 1)$	5(a)
$= x + x' \cdot 1$	4(b)
$= x + x'$	2(b)
$= 1$	5(a)

THEOREM 2(b): $x \cdot 0 = 0$ by duality.

THEOREM 6(a): $x + xy = x.$

Statement	Justification
$x + xy = x \cdot 1 + xy$	postulate 2(b)
$= x(1 + y)$	4(a)
$= x(y + 1)$	3(a)
$= x \cdot 1$	2(a)
$= x$	2(b)

THEOREM 6(b): $x(x + y) = x$ by duality.

The theorems of Boolean algebra can be proven by means of truth tables. In truth tables, both sides of the relation are checked to see whether they yield identical results for all possible combinations of the variables involved. The following truth table verifies the first absorption theorem:

x	y
0	0
0	1
1	0
1	1

xy	$x + xy$
0	0
0	0
0	1
1	1

The algebraic proofs of the associative law and DeMorgan's theorem are long and will not be shown here. However, their validity is easily shown with truth tables. For example, the truth table for the first DeMorgan's theorem, $(x + y)' = x'y'$, is as follows:

x	y	$x + y$	$(x + y)'$
0	0	0	1
0	1	1	0
1	0	1	0
1	1	1	0

x'	y'	$x'y'$
1	1	1
1	0	0
0	1	0
0	0	0

Operator Precedence

The operator precedence for evaluating Boolean expressions is (1) parentheses, (2) NOT, (3) AND, and (4) OR. In other words, expressions inside parentheses must be evaluated before all other operations. The next operation that holds precedence is the complement, and then follows the AND and, finally, the OR. As an example, consider the truth table for one of DeMorgan's theorems. The left side of the expression is $(x + y)'$. Therefore, the expression inside the parentheses is evaluated first and the result then complemented. The right side of the expression is $x'y'$, so the complement of x and the complement of y are both evaluated first and the result is then ANDed. Note that in ordinary arithmetic, the same precedence holds (except for the complement) when multiplication and addition are replaced by AND and OR, respectively.

Boolean fonksiyonları

- Boole cebri, ikili değişkenler ve mantık işlemleriyle ilgilenen bir cebirdir.
- Cebirsel bir ifade ile tanımlanan bir Boole fonksiyonu, ikili değişkenlerden, 0 ve 1 sabitlerinden ve mantık işlem sembollerinden oluşur.
- İkili değişkenlerin belirli bir değeri için, fonksiyon 1 veya 0'a eşit olabilir.
- Örnek olarak,

$$F_1 = x + y'z$$

$$F_2 = x'y'z + x'yz + xy' = x'z(y' + y) + xy' = x'z + xy'$$

Boolean fonksiyonları

$$F_1 = x + y'z$$

x	y	z	F_1
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

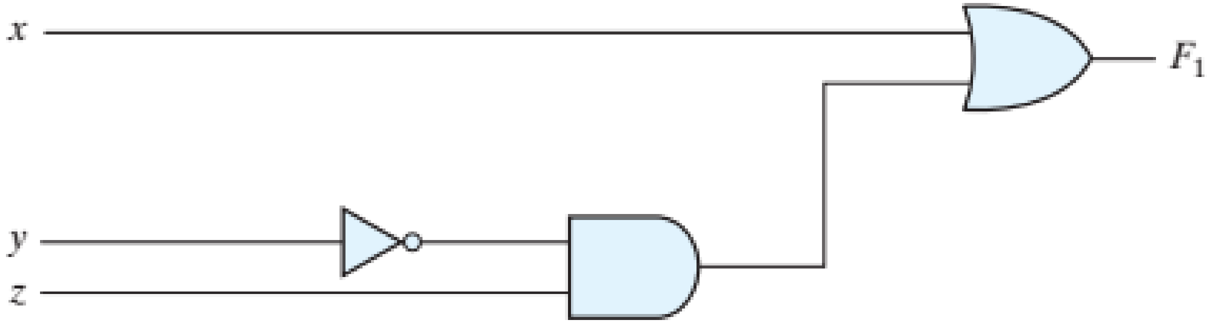
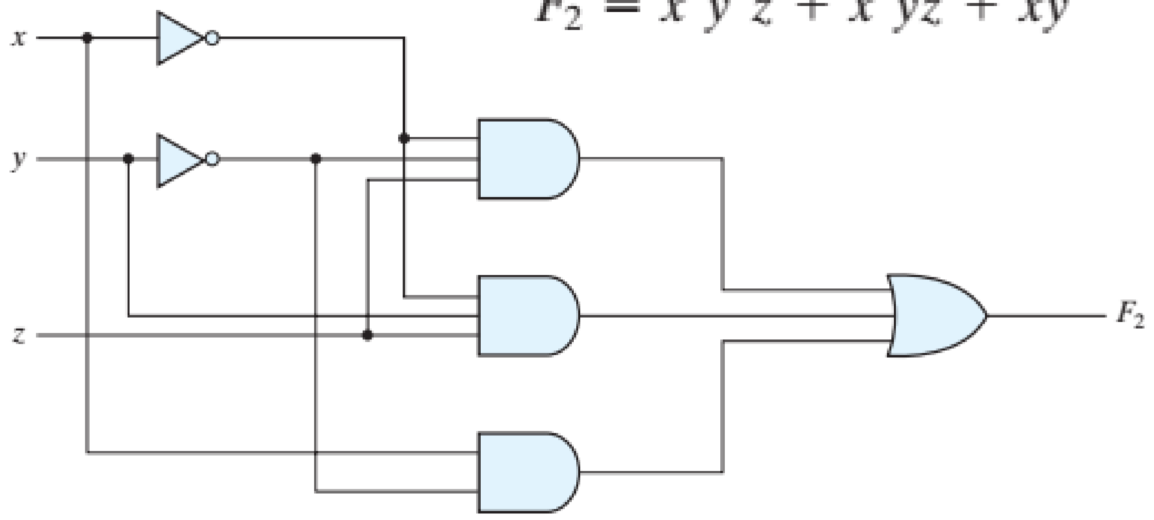


FIGURE 2.1
Gate implementation of $F_1 = x + y'z$

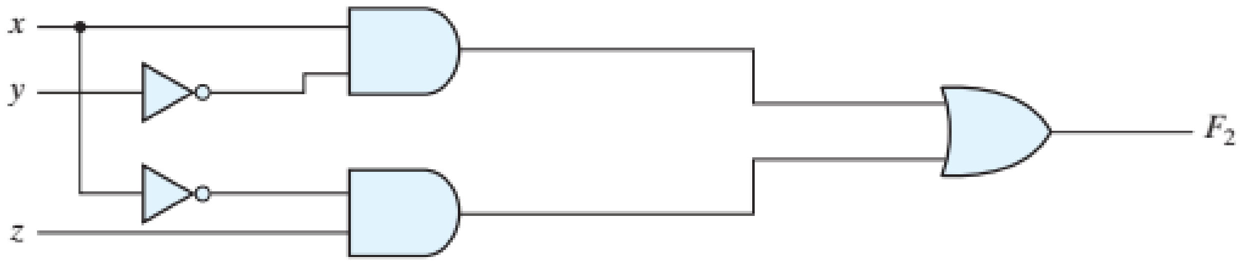
Boolean fonksiyonları

$$F_2 = x'y'z + x'yz + xy'$$



Doğruluk Tablosu ???

$$F_2 = x'y'z + x'yz + xy' = x'z(y' + y) + xy' = x'z + xy'$$



Boolean fonksiyonları

- "**Digital Design**", M. Morris Mano

EXAMPLE 2.1

Simplify the following Boolean functions to a minimum number of literals.

1. $x(x' + y) = xx' + xy = 0 + xy = xy.$
2. $x + x'y = (x + x')(x + y) = 1(x + y) = x + y.$
3. $(x + y)(x + y') = x + xy + xy' + yy' = x(1 + y + y') = x.$
4. $$\begin{aligned} xy + x'z + yz &= xy + x'z + yz(x + x') \\ &= xy + x'z + xyz + x'yz \\ &= xy(1 + z) + x'z(1 + y) \\ &= xy + x'z. \end{aligned}$$
5. $(x + y)(x' + z)(y + z) = (x + y)(x' + z)$, by duality from function 4.

Bir fonksiyonun tümleyeni

$$
\begin{aligned}
(A + B + C)' &= (A + x)' && \text{let } B + C = x \\
&= A'x' && \text{by theorem 5(a) (DeMorgan)} \\
&= A'(B + C)' && \text{substitute } B + C = x \\
&= A'(B'C') && \text{by theorem 5(a) (DeMorgan)} \\
&= A'B'C' && \text{by theorem 4(b) (associative)}
\end{aligned}
$$

Bir fonksiyonun tümleyeni

- "**Digital Design**", M. Morris Mano

EXAMPLE 2.2

Find the complement of the functions $F_1 = x'yz' + x'y'z$ and $F_2 = x(y'z' + yz)$. By applying DeMorgan's theorems as many times as necessary, the complements are obtained as follows:

$$F_1' = (x'yz' + x'y'z)' = (x'yz')'(x'y'z)' = (x + y' + z)(x + y + z')$$

$$\begin{aligned} F_2' = [x(y'z' + yz)]' &= x' + (y'z' + yz)' = x' + (y'z')'(yz)' \\ &= x' + (y + z)(y' + z') \\ &= x' + yz' + y'z \end{aligned}$$

Lojik Devre Tasarımına Giriş

- Bu bölümde geri besleme içermeyen ve kombinasyonel lojik olarak adlandırılan lojik devrelerin tasarımı anlatılacaktır. Lojik devre tasarımı aşağıdaki 4 adım gerçekleştirilerek yapılır.
  1. Lojik devrenin yapacağı tüm işlemler detaylarıyla belirlenir.
  2. İşlem doğruluk tablosu hazırlanır.
  3. İşlemin lojik ifadesi yani matematiksel modeli yazılır.
  4. Lojik devre basitleştirilebiliyorsa sadeleştirilir.
- Lojik ifadeler iki formda belirtilir.
  - 1. Form: Temel işlem toplama, ara işlemler çarpma
  - 2. Form: Temel işlem çarpma, ara işlemler toplama

Lojik Devre Tasarımına Örnek

- Bir banka hırsız alarmı şöyle çalışıyor: Alarm devresinin 3 giriş elemanı vardır (Kasa kapısı, pencere, oda kapısı). Sadece pencere veya sadece oda kapısı veya sadece kasa kapısı açık ise bu kapıların açık olduğu düşünülecektir. En az iki giriş elemanı açık ise içeride hırsız olduğu düşünülecek ve alarm devreye girecektir.
- Bu durumları dikkate aldığınızda doğruluk tablosu nasıl olur? Alarm devresini çiziniz !

Girişler			Çıkış
A Kasa	B Pencere	C Oda	f (Alarm)

Lojik Devre Tasarımına Örnek

- Bir banka hırsız alarmı şöyle çalışıyor: Alarm devresinin 3 giriş elemanı vardır (Kasa kapısı, pencere, oda kapısı). Sadece pencere veya sadece oda kapısı veya sadece kasa kapısı açık ise bu kapıların açık olduğu düşünülecektir. En az iki giriş elemanı açık ise içeride hırsız olduğu düşünülecek ve alarm devreye girecektir.
- Bu durumları dikkate aldığınızda doğruluk tablosu nasıl olur? Alarm devresini çiziniz !

Girişler			**Çıkış**
A **Kasa**	**B** **Pencere**	**C** **Oda**	**f** **(Alarm)**
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	**1**
1	0	0	0
1	0	1	**1**
1	1	0	**1**
1	1	1	**1**

Lojik Devre Tasarımına Örnek

- Matematiksel ifade doğruluk tablosunda çıkışı «1» yapan kombinasyonların toplamı şeklindedir.

$$f = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

Girişler			Çıkış
A Kasa	B Pencere	C Oda	f (Alarm)
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	**1**
1	0	0	0
1	0	1	**1**
1	1	0	**1**
1	1	1	**1**

Lojik Devre Tasarımına Örnek

$$f = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

$f = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$

Matematiksel ifade yukarıdaki gibi çıkışı "1" yapan kombinasyonların toplamı şeklindedir.

- Sadeleştirme yapalım:

$$f = C\underbrace{(\bar{A}B + A\bar{B})}_{A \oplus B} + AB\underbrace{(C + \bar{C})}_{1}$$

$$\boxed{f = C(A \oplus B) + AB}$$

- farklı bir yol:

$$f = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC + \underbrace{ABC + ABC}$$

Bunları eklemek sonucu etkilemez (terim farkı olduğu için)

$$f = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC + ABC + ABC$$

(3) (2) (1)

$$f = BC\underbrace{(A + \bar{A})}_{1} + AC\underbrace{(\bar{B} + B)}_{1} + AB\underbrace{(\bar{C} + C)}_{1}$$

$$\boxed{f = BC + AC + AB}$$

Lojik Devresi;