Lojik Tasarım

Ders 3

Kaynak:
M.M. Mano, M.D. Ciletti, “Digital Design with An Introduction to Verilog HDL”

Boolean Cebri ve Lojik Kapılar

Temel Tanımlar

- **Kapalılık:** İkili işlem, S'deki her eleman çiftine yine S'deki bir elemana karşı düşürecek bir kural belirliyorsa, S kümesi bu ikili işleme göre kapalıdır.

  - Örneğin N doğal sayılar kümesi N={1,2,3,4,...} aritmetik toplama kurallarıyla artı (+) işlemine göre kapalıdır. Çünkü herhangi bir a,b$\in$N, a+b =c işlemiyle tek bir c $\in$N elde edilebilir.
  - Buna karşılık, doğal sayılar kümesi aritmetik çıkarma kurallarıyla eksi (-) ikili işlemine göre kapalı değildir. Çünkü 2-3=-1 ve 2,3 $\in$N iken (-1)$\notin$N'dir.

Temel Tanımlar

Birleşme Kuralı

2. *Associative law.* A binary operator * on a set S is said to be associative whenever

$$(x*y)*z = x*(y*z) \text{ for all } x, y, z, \in S$$

Değişme Kuralı

3. *Commutative law.* A binary operator * on a set S is said to be commutative whenever

$$x*y = y*x \text{ for all } x, y \in S$$

Birim Elemanı

4. *Identity element.* A set S is said to have an identity element with respect to a binary operation * on S if there exists an element $e \in S$ with the property that

$$e*x = x*e = x \text{ for every } x \in S$$

Example: The element 0 is an identity element with respect to the binary operator + on the set of integers $I = \{\ldots, -3, -2, -1, 0, 1, 2, 3, \ldots\}$, since

$$x + 0 = 0 + x = x \text{ for any } x \in I$$

Temel Tanımlar

Ters

5. *Inverse.* A set S having the identity element e with respect to a binary operator $*$ is said to have an inverse whenever, for every $x \in S$, there exists an element $y \in S$ such that

$$x * y = e$$

Example: In the set of integers, I, and the operator $+$, with $e = 0$, the inverse of an element a is $(-a)$, since $a + (-a) = 0$.

Dağılma Kuralı

6. *Distributive law.* If $*$ and $\cdot$ are two binary operators on a set S, $*$ is said to be distributive over $\cdot$ whenever

$$x * (y \cdot z) = (x * y) \cdot (x * z)$$

Boolean Cebrinin Aksiyomatik tanımı

1. (a) The structure is closed with respect to the operator $+$.
(b) The structure is closed with respect to the operator $\cdot$.

2. (a) The element 0 is an identity element with respect to $+$; that is, $x + 0 = 0 + x = x$.
(b) The element 1 is an identity element with respect to $\cdot$; that is, $x \cdot 1 = 1 \cdot x = x$.

3. (a) The structure is commutative with respect to $+$; that is, $x + y = y + x$.
(b) The structure is commutative with respect to $\cdot$; that is, $x \cdot y = y \cdot x$.

4. (a) The operator $\cdot$ is distributive over $+$; that is, $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$.
(b) The operator $+$ is distributive over $\cdot$; that is, $x + (y \cdot z) = (x + y) \cdot (x + z)$.

5. For every element $x \in B$, there exists an element $x' \in B$ (called the *complement* of x) such that (a) $x + x' = 1$ and (b) $x \cdot x' = 0$.

6. There exist at least two elements $x, y \in B$ such that $x \neq y$.

İki Değerli Boolean Cebri

x	y	$x \cdot y$
0	0	0
0	1	0
1	0	0
1	1	1

x	y	$x + y$
0	0	0
0	1	1
1	0	1
1	1	1

x	x'
0	1
1	0

1. That the structure is *closed* with respect to the two operators is obvious from the tables, since the result of each operation is either 1 or 0 and $1, 0 \in B$.
2. From the tables, we see that

   (a) $0 + 0 = 0 \qquad 0 + 1 = 1 + 0 = 1;$
   (b) $1 \cdot 1 = 1 \qquad 1 \cdot 0 = 0 \cdot 1 = 0.$

   This establishes the two *identity elements*, 0 for + and 1 for $\cdot$, as defined by postulate 2.
3. The *commutative* laws are obvious from the symmetry of the binary operator tables.

İki Değerli Boolean Cebri

4. (a) The *distributive* law $x \cdot (y + z) = (x \cdot y) + (x \cdot z)$ can be shown to hold from the operator tables by forming a truth table of all possible values of x, y, and z. For each combination, we derive $x \cdot (y + z)$ and show that the value is the same as the value of $(x \cdot y) + (x \cdot z)$:

x	y	z
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$y + z$	$x \cdot (y + z)$
0	0
1	0
1	0
1	0
0	0
1	1
1	1
1	1

$x \cdot y$	$x \cdot z$	$(x \cdot y) + (x \cdot z)$
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	1	1
1	0	1
1	1	1

(b) The *distributive* law of + over $\cdot$ can be shown to hold by means of a truth table similar to the one in part (a).

5. From the complement table, it is easily shown that

(a) $x + x' = 1$, since $0 + 0' = 0 + 1 = 1$ and $1 + 1' = 1 + 0 = 1$.
(b) $x \cdot x' = 0$, since $0 \cdot 0' = 0 \cdot 1 = 0$ and $1 \cdot 1' = 1 \cdot 0 = 0$.

Thus, postulate 1 is verified.

Boolean Cebrine İlişkin Temel Teoremler

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)	$x(x + y) = x$

İşlem Önceliği

1. Parantez
2. DEĞİL (NOT)
3. VE (AND)
4. VEYA (OR)

Boolean Fonksiyonlarının Doğruluk Tabloları

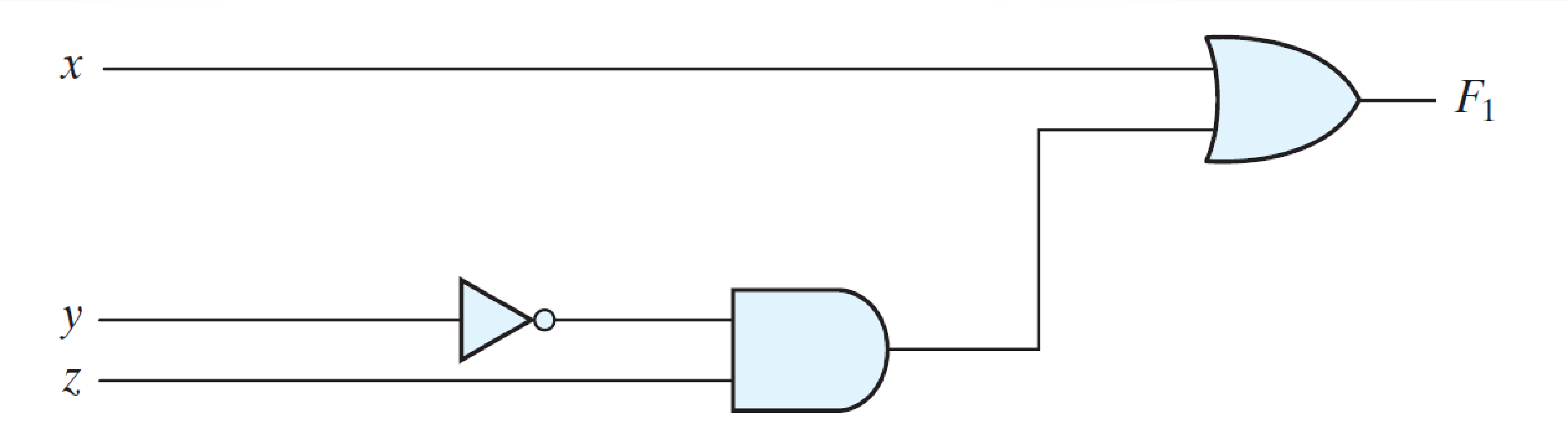
$F_1 = x + y'z$

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

x	y	z	F_1	F_2
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	0	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	0
1	1	1	1	0

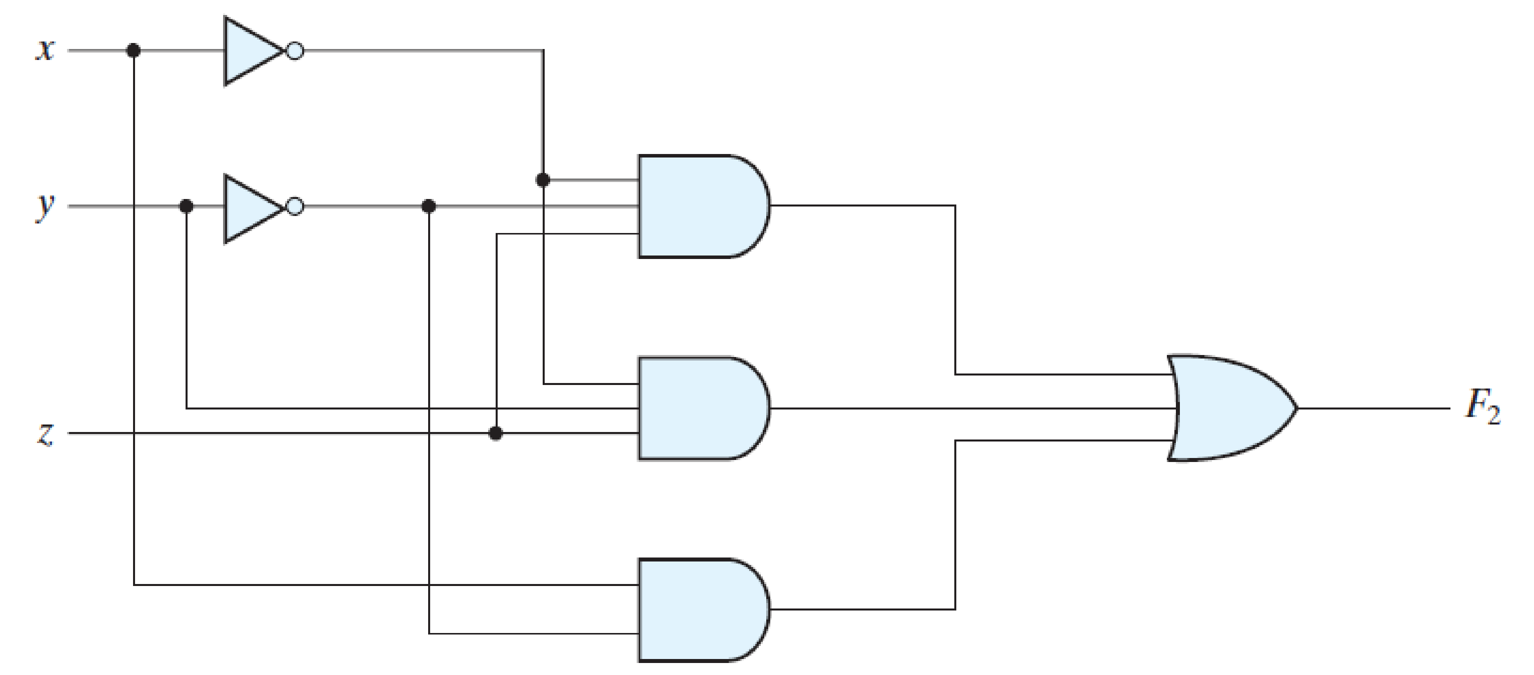
Boolean Fonksiyonlarının Lojik Kapılar ile Gerçeklenmesi

$F_1 = x + y'z$



Boolean Fonksiyonlarının Lojik Kapılar ile Gerçeklenmesi

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



Boolean Fonksiyonlarının Sadeleştirilmesi

Aşağıdaki Boolean fonksiyonunu minimum sayıdaki değişkene indirgeyerek sadeleştirin.

$x\,(x'+y) = ?$ (xy)

Boolean Fonksiyonlarının Sadeleştirilmesi

Aşağıdaki Boolean fonksiyonunu minimum sayıdaki değişkene indirgeyerek sadeleştirin.

$x + x'y = ?$ $(x+y)$

Boolean Fonksiyonlarının Sadeleştirilmesi

Aşağıdaki Boolean fonksiyonunu minimum sayıdaki değişkene indirgeyerek sadeleştirin.

$$(x + y)\,(x + y') = \; ? \quad (x)$$

Boolean Fonksiyonlarının Sadeleştirilmesi

Aşağıdaki Boolean fonksiyonunu minimum sayıdaki değişkene indirgeyerek sadeleştirin.

$$xy + x'z + yz = \ ? \ _{(xy+x'z)}$$

Boolean Fonksiyonlarının Sadeleştirilmesi

Aşağıdaki Boolean fonksiyonunu minimum sayıdaki değişkene indirgeyerek sadeleştirin.

$$(x+y)(x'+z)(y+z) = \ ? \quad (x+y)(x'+z)$$

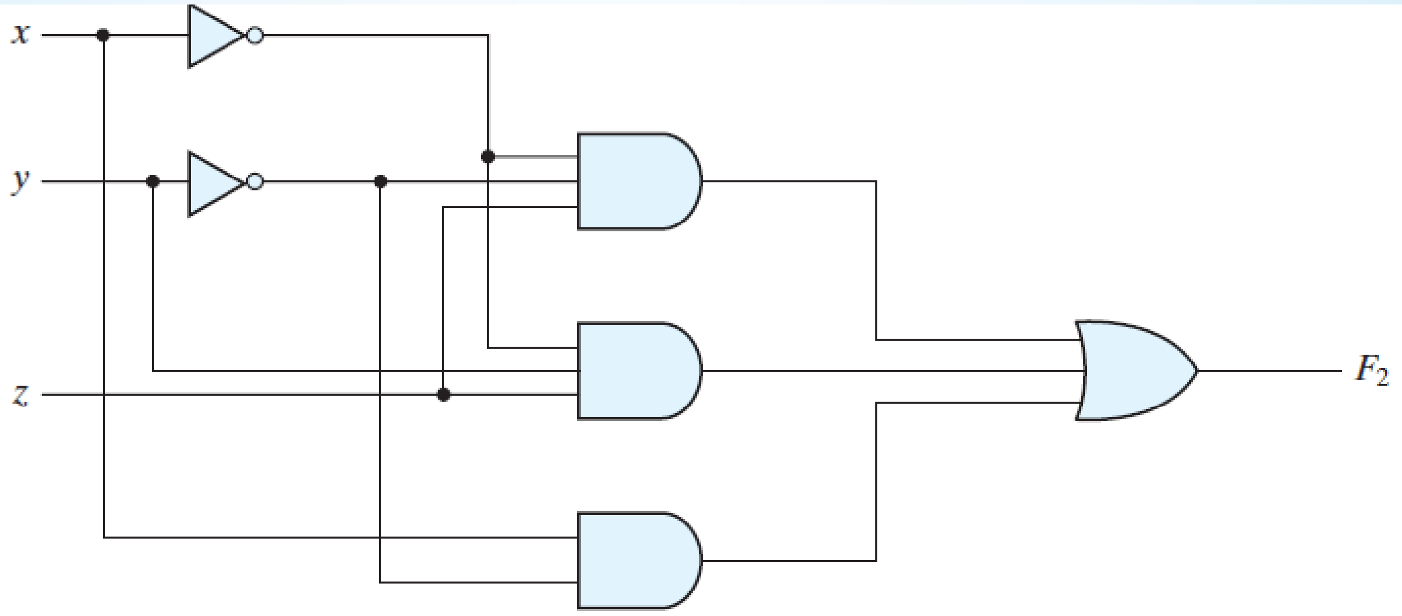
Boolean Fonksiyonlarının Sadeleştirilmesi

Aşağıdaki Boolean fonksiyonunu minimum sayıdaki değişkene indirgeyerek

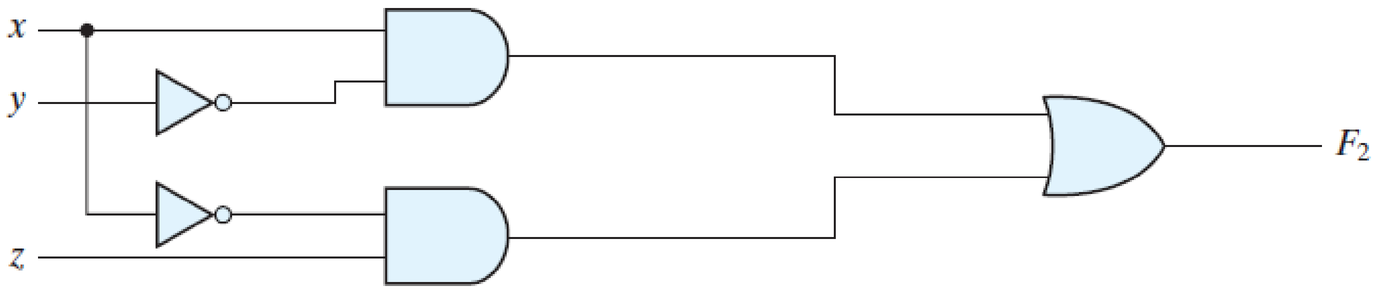
$$F_2 = x'y'z + x'yz + xy'$$

$x'y'z + x'yz + xy' = \ ?$ $(x'z+xy')$

Boolean Fonksiyonlarının Sadeleştirilmesi



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



(b) $F_2 = xy' + x'z$

Fonksiyonun Tümleyeni

Aşağıdaki Boolean fonksiyonunun tümleyenini hesaplayınız

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

Fonksiyonun Tümleyeni

Aşağıdaki Boolean fonksiyonlarının tümleyenini hesaplayınız

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

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

Gelecek Hafta

- Minterim (MINTERM) ve Maksterim (MAKSTERM)