

SAYISAL TASARIM BOOLEAN CEBİRİ VE MANTIK KAPILARI

Hazırlayan:

Yard. Doç. Dr. Metin ARTIKLAR

Temel Tanımlamalar

- Cebir tanımlanan bir set üzerinde (bu sayılar, harfler v.s. olabilir) kuralları belirlenmiş operatörler tarafından yapılan işlemler bütünüdür.
- Örneğin tam sayılar kümesi üzerinde tanımlanmış ve toplama operatörü $+$, kullanılarak yapılan toplama işlemi cebirsel bir işlemdir.

Temel Tanımlamalar

- Buradaki set ifadesi ortak bir özelliğe sahip elemanlar kümesini belirtmektedir.
- $A = (1, 2, 3, 4)$ ortak özelliği tam sayı olmaları olan bir A setini gösterir. Bir setin bir elemanı $\in$ sembolü ile gösterilir. Örneğin $1 \in A$, $5 \notin A$

İkili Operatör

- İkili operatör bir A setinin iki elemanını tanımlanmış olan bir kurallar bütününe göre işleme sokarak sonucu yine A setinin bir elemanı olarak bulan işlemciye verilen isimdir.
- Orn: Gerçek sayılar üzerinde tanımlanan bilinen $+$ operatörü

İkili Operatör

- Örneğin bir S seti için x, y, z değişkenleri S setinin eleman değerlerini alabilen rastgele değişkenler olarak tanımlansın ve S üzerinde bir $*$ operatörü tanımlanmış olsun
- $z=x*y$ işlemi için $z \in S$ ise $*$ işlemi bir ikili operatör, $z \notin S$ ise $*$ işlemi bir ikili operatör değildir.

Aksiomlar

- Yukarıda adı geçen cebirsel işlemlerin yapılabilmesi için öncelikle bu işlemlerin belirli şartları sağladıklarının kanıtlanmış olması gerekir.
- Matematiksel yapıların uymaları zorunlu olan şartlara ***aksiom*** adı verilir.
- Aksiomları kullanmadan önce kanıtlamaya gerek yoktur.
- Cebirsel yapıları(işlemleri) tanımlarken en fazla kullanılan aksiomlar şunlardır:

Temel Cebirsel Aksiomlar

1- Kapalılık

Bir S seti ve bu set üzerinde tanımlanan herhangi bir $*$ operatörü düşünelim. $x, y \in S$ için $x * y = z$ işleminin sonucunu gösteren z elemanı $z \in S$ ise S seti $*$ operatörüne göre kapalıdır denir.

Temel Cebirsel Aksiomlar

- Eğer $*$ operatörü tüm farklı $a, b \in S$ elemanları için $a*b=c$ işleminin sonucu olarak farklı $c \in S$ elemanlarını veriyorsa S seti $*$ operatörüne göre kapalıdır denir. Örneğin doğal sayılar kümesi $D = (1, 2, 3, \dots, \infty)$ bilinen toplama operatörüne, $+$, göre kapalıdır. Ancak aynı N kümesi yine bilinen, $-$, çıkarma operatörüne göre kapalı değildir.

Temel Cebirsel Aksiomlar

2- Birleşme

İkili * operatörünün aşağıdaki şartı sağlaması durumunda S seti üzerinde birleşme özelliğinin olduğu söylenir.

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

bütün $x, y, z \in S$ için

Temel Cebirsel Aksiomlar

3- Değişme

İkili $*$ operatörünün aşağıdaki şartı sağlaması durumunda S seti üzerinde değişme özelliğinin olduğu söylenir.

$$x * y = y * x \quad \text{bütün } x, y \in S \text{ için}$$

Temel Cebirsel Aksiomlar

4- Birim Eleman Özelliği

Aşağıdaki şartın sağlanması durumunda S setinin ikili $*$ operatörüne göre e olarak tanımlanan bir $(e \in S)$ birim elemanın olduğu söylenir.

$$e * x = x * e = x \quad \text{bütün } x \in S \text{ için}$$

Temel Cebirsel Aksiomlar

- Örnek: Tamsayılar kümesi, $T = (-\infty, \dots, -2, -1, 0, 1, 2, \dots, \infty)$, üzerinde tanımlanan bilinen toplama operatörü, $+$, için 0 bir birim elemandır. Ancak doğal sayılar kümesi üzerinde tanımlanan bilinen toplama operatörünün birim elamanı yoktur. Çünkü $0 \notin D$

Temel Cebirsel Aksiomlar

5- Ters Eleman Özelliği

Bir S seti üzerinde tanımlanan bir ikili $*$ operatörü olduğunu ve S setinin $*$ operatörüne göre $e \in S$ olan bir birim elemanının olduğunu düşünelim.

Aşağıdaki şartın sağlanması durumunda S seti içerisindeki elemanların terslerinin olduğu söylenir.

$$x * y = e$$

bütün $x, y \in S$ için

Temel Cebirsel Aksiomlar

- Örnek:
- $e = 0$ birim elemanı sahip tamsayılar, T , seti içerisindeki herhangi bir $x \in T$ elemanın tersi $-x$ 'tir. ($-x \in T$). Çünkü $x + (-x) = 0 = e$ 'dir.

Temel Cebirsel Aksiomlar

6- Dağılma Özelliği

S seti üzerinde tanımlanan ikili ‘*’ ve ‘.’ operatörleri için, aşağıdaki şartın sağlanması durumunda, * operatörünün nokta operatörü üzerine dağılma özelliğinin olduğu söylenir.

$$x * (y . z) = (x * y) . (x * z)$$

Temel Cebirsel Aksiomlar

- Örn: Tamsayılar seti üzerinde tanımlanan bilinen $\cdot$ (çarpma) operatörünün bilinen $+$ (toplama) operatörü üzerine dağılma özelliği vardır. Ancak bilinen $+$ operatörünün bilinen $\cdot$ üzerine dağılma özelliği yoktur.

$$3 \cdot (2 + 5) = 3 \cdot 2 + 3 \cdot 5$$

$$2 + (3 \cdot 5) \neq (2 + 3) \cdot (2 + 5)$$

Cebirsel Yapılara Bir Örnek

- Yukarıdaki aksiomlar kullanılarak oluşturulabilecek cebirsel yapılara bir örnek *alandır*. Alan bir S seti ve bu set üzerinde tanımlanan 2 tane ikili operatörden oluşur. Operatörlerin yukarıda tanımlanan özelliklerden 1'den 5'e kadar olanlarını sağlaması gerekir. Alana bir örnek verecek olursak . Gerçek sayılar seti üzerinde tanımlanan bilinen '+' (toplama) ve '.' çarpma operatörleri bir gerçek sayılar alanını oluşturur. (Ödev: Doğruluğunu gösterin)

Boolean Cebiri

- Boolean cebiri 1854 yılında George Boole tarafından sayısal elektroniğin sistematik bir şekilde ifade edilebilmesi için geliştirilmiştir.
- 1938 yılında ise Shannon iki değerli (yani S seti içerisinde sadece 0 ve 1 sayılarının olduğu) Boolean cebirini geliştirmiştir.
- Bu cebir bir S seti üzerinde tanımlanan ‘+’ ve ‘.’ (bunlar bilinen toplama ve çarpma operatörlerinden farklıdır) oluşmaktadır.
- Bu operatörler S seti üzerinde şu aksiomları sağlamaktadırlar.

Boolean Cebirinin Sağladığı Aksiomlar

- 1 (a) + operatörüne göre kapalılık özelliği
(b) . operatörüne göre kapalılık özelliği
- 2 (a) + operatörüne göre değişme özelliği $x+y=y+x$
(b) . operatörüne göre değişme özelliği $x.y=y.x$
- 3 (a) + operatörüne göre “0” olarak gösterilen birim eleman $x + 0 = 0 + x = x$
(b) . operatörüne göre “1” olarak gösterilen eleman $x . 1 = 1 . x = x$

Boolean Cebirinin Sağladığı Aksiomlar

4 (a) $+$ operatörü için S seti tüm $x \in S$ değerleri için $x+x' = 1$ olacak ve x' 'in **tümleyeni** olarak adlandırılan $x' \in S$ elemanını içermektedir.

Not: Ters olabilmesi için $x + x' = 0$ olmalıydı

(b) $.$ operatörü için S seti tüm $x \in S$ elemanları için $x . x' = 0$ olacak ve x' 'in tümleyeni olarak adlandırılan $x' \in S$ elemanını içermektedir.

Not: tersi olması için $x . x' = 1$ olmalıydı.

Boolean Cebirinin Sağladığı Aksiomlar

5 (a) $+$ operatörünün nokta üzerine dağılma özelliği vardır.

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

Not: Bu özellik bilinen toplama ve çarpma operatörleri için geçerli değildir

(b) $\cdot$ operatörünün $+$ üzerine dağılma özelliği vardır.

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

6) S seti içerisinde en az iki eleman olmak zorundadır. $x, y \in S \quad x \neq y$

Boolean Cebirinin Sağladığı Aksiomlar

- Boolean cebirinin tanımı için kabul gören bu 6 aksiom 1904 yılında E.V Huntigton tarafından ortaya atıldığından Huntigton aksiomları olarak bilinir.
- Boolean cebirini gerçek sayılar seti, G , üzerinde bilinen toplama ve çarpma operatörleri ile oluşturulan cebirle karşılaştırırsak şu farkların olduğunu gözleriz:

Boolean Cebiri ile Kullandığımız Cebirin Karşılaştırılması

- 1- Huntington aksiomları birleşme özelliğini içermez. Ancak bu özelliğin boolean cebiri için geçerli olduğu diğer aksiomlar kullanılarak gösterilebilir.
- 2- Bilinen $+$ operatörün bilinen çarpma işlemi üzerine dağılma özelliği yoktur. (tersi geçerli). Ancak boolean cebirindeki $+$ operatörünün nokta operatörü üzerine dağılma özelliği vardır.
- 3- Boolean cebirinde toplama ve çarpma operatörleri için elemanların tersleri yoktur(tümleyenleri vardır). Dolayısıyla Boolean cebirinde çıkarma ve bölme işlemleri tanımlanmış değildir.
- 4- Dördüncü aksiom Boolean cebiri için set elemanlarının tümleyenini tanımlar. Bilinen cebirde bu şekliyle sayıların tümleyenleri tanımlı değildir.
- 5- Bilinen cebir içinde sonsuz tane eleman bulunan gerçel sayılar seti üzerinde tanımlanmıştır. Boolean cebiri içinde böyle bir set tanımlanabilir. Ancak bizim kullanacağımız Boolean cebiri, 2 değerli boolean cebir olarak adlandırılacak ve $B = \{ 0, 1 \}$ elemanlarını içeren bir set üzerinde tanımlanmış işlemlerden oluşacaktır.

Boolean Cebiri

- Tekrar edecek olursak, Boolean cebirinin tanımlanabilmesi için şunlar gereklidir:
 - Bir B seti içerisinde tanımlanmış elemanlar
 - Bu set üzerinde tanımlanmış 2 adet ikili operatör.
 - B seti ve bu set üzerinde tanımlanmış 2 adet ikili operatörün Huntington'un 6 adet aksiomunu sağlıyor olması.

İki Elemanlı Boolean Cebiri

- Bu cebir $B = \{0, 1\}$ seti üzerinde tanımlı $+$ ve $.$ ikili operatörlerinden oluşur.
- Günümüzde sayısal devre tasarımı tüm dünyada kullanılan cebirdir
- Bu açıdan bundan sonra iki elemanlı Boolean cebiri tabiri yerine sadece Boolean cebiri tabirini kullanacağız
- $+$ ve $.$ operatörleri tanımlayacağımız şu kurallara göre işlem görmektedirler

Boolean Cebirinde Tanımlı İkili Operatörlerin İşlem Kuralları

VE

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

VEYA

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

TÜMLEYEN

x	x'
0	1
1	0

Boolean Cebiri

- Bu cebiri kullanmaya başlamadan önce B seti ile bu set üzerinde tanımladığımız $+$ ve $.$ operatörlerinin Huntington'un 6 adet aksiomunu sağladığını göstermemiz gerekir:

Boolean Cebirinin Kapalılık ve Değişme Özelliği

- 1- Kapalılık özelliği sağlanmaktadır. Çünkü tablolardan görüleceği üzere tüm işlemlerin sonucu yine B setinin elemanıdır.
- 2- Değişme özelliğinin sağlandığı aşağıdaki tablodan görülmektedir.

Boolean Cebirinin Değişme Özelliği

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

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

Boolean Cebirinin Birim Eleman ve Dağılma Özelliği

- 3- Birim eleman özelliği
 - $0+0 = 0$; $1+0 = 1$ olduğundan, $+$ operatörü için 0 birim elemandır.
 - $0.1 = 0$; $1.1 = 1$ olduğundan . operatörü için 1 birim elemandır.
- 4- Dağılma özelliğinin sağlandığı
aşağıdaki tablolardan görülmektedir.

Boolean Cebirinin Dağılma Özelliği

x	y	z	$y+z$	$x.(y+z)$	$x.y$	$x.z$	$(x.y) + (x.z)$
0	0	0	0	0	0	0	0
0	0	1	1	0	0	0	0
0	1	0	1	0	0	0	0
0	1	1	1	0	0	0	0
1	0	0	0	0	0	0	0
1	0	1	1	1	0	1	1
1	1	0	1	1	1	0	1
1	1	1	1	1	1	1	1

Boolean Cebirinin Tümleyen Özelliği

- 5- Tümleyen özelliğini

Bu özelliğin sağlandığı aşağıdaki şekilde gösterilebilir.

+ operatörü için: $x + x' = 1$

$$0+0'=0+1=1 \quad ; \quad 1+1'=1+0=1$$

. operatörü için: $x . x' = 0$

$$0.0'=0.1=0 \quad ; \quad 1.1'=1.0=0$$

- 6- B kümesi içerisinde iki eleman bulunmaktadır. $B = \{0, 1\}$

Boolean Cebiri

- Boolean cebirinde tanımladığımız $+$, $.$ ve $'$ operatörlerinin işlem kuralları birinci bölümde tanımladığımız VE , $VEYA$ ve $DEĞİL$ mantık kapılarının işlem kuralları ile aynıdır.

Boolean cebirinde Çokça Kullanılan Teoremler

- ***Teorem*** cebirsel işlemler yaparken (cebirsel fonksiyonların toplanması, daha sade ifadelerin bulunması v.s) işlemleri kolaylaştırmak amacı ile kullanılan ancak aksiomların aksine doğruluklarının kanıtlanması gereken kurallardır.
- Boolean cebirinde en çok kullanılan teoremler ve kurallar şunlardır

Eşdeğerlik (Duality) Kuralı

- Boolean cebirinde herhangi bir cebirsel fonksiyonun içindeki operatörler ve birim elemanlar değiştirilirse fonksiyon geçerliliğini korur.

Örn: $x + 1 = 1$

$$x \cdot 0 = 0$$

Teoremler

Aksiom ve Teoremler		$+$ operatörü		$.$ operatörü
Aksiom 2 (Birim Eleman)	a	$x + 0 = x$	b	$x . 1 = x$
Aksiom 5 (Tümleyen)	a	$x + x' = 1$	b	$x . x' = 0$
Teorem 1	a	$x + x = x$	b	$x . x = x$
Teorem 2	a	$x + 1 = 1$	b	$x . x = x$
Teorem 3	a	$(x')' = x$	b	
Aksiom 3 (Değişme)	a	$x + y = y + x$	b	$x . y = y . x$
Teorem 4 (Birleşme)	a	$x + (y + z) = (x + y) + z$	b	$x . (y . z) = (x . y) . z$
Aksiom 3 (Dağılma)	a	$x . (y + z) = x.y + x.z$	b	$x + y.z = (x + y) . (x + z)$
Teorem 5 (DeMorgan)	a	$(x + y)' = x' . y'$	b	$(x . y)' = x' + y'$
Teorem 6 (Yutma)	a	$x + x.y = x$	b	$x . (x + y) = x$

Teoremler

- Teorem 1.a:

$x+x=x \rightarrow x(1+1) \rightarrow x.1 = x$ (çarpmanın toplama üzerinde dağılma özelliği olduğundan ve birim elemanı özellikleri kullanıldı)

- Teorem 1.b:

$x.x = x \rightarrow x.x + 0 = x$ (birim eleman)

$xx+xx' = x$ (Çarpmanın toplama üzerine dağılma özelliği)

$x(x+x') = x \rightarrow x . 1 = x$

Teoremler

- Teorem 2.a

$$x+1=1 \rightarrow (\text{çarpmanın birim elemanı özelliğinden})$$

$$1(x+1)=1 \rightarrow (x+x').(x+1)=1 \text{ (Dağılma özelliğinden)}$$

$$x.x + x.1 + x'.x + x'.1 = 1$$

$$x + x + 0 + x'.1 = 1 \rightarrow x + x' = 1$$

- Teorem 2.b

$$x.0=0 \quad (x+1=1'e \text{ eşdeğerlik kuralı uygulanarak bulunur})$$

Teoremler

- Teorem 6.a

$$x + xy = x \rightarrow x.1 + x.y = x$$

$$x(1+y) = x \rightarrow x.1 = x \rightarrow x = x$$

- Teorem 6.b

$$x . (x + y) = x \quad (x+xy = x' \text{ e eşdeğerlik kuralı uygulanarak})$$

$$x + x.y = x$$

Teorem 3.

$$(x')' = x$$

De Morgan Kuralı

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

De Morgan Kuralı

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

Operatör Öncelikleri

- Boolean cebirinde operatör öncelikleri şu şekildedir.

Parantez

TÜMLEYEN (DEĞİL)

. (VE)

+ (VEYA)

Operatör Öncelikleri

Örn: $(x+y)'$

Önce parantezin içerisindeki $(x+y)$ işlemi yapılır, daha sonra sonucun ***tümleyeni*** (***değili***) alınır.

Örn: $x'.y'$

Önce x in ***tümleyeni*** (***değili***) ve y 'nin ***tümleyeni*** (***değili***) alınır daha sonrada ***(VE)*** işlemi gerçekleştirilir.

Boolean Fonksiyonları

- Boolean fonksiyonları 0 veya 1 değerini alabilen değişkenler ve bunların değil'lerinin, + ve . operatörleri ile işleme alınmaları sonucunda oluşan fonksiyonlardır
- Örnek:

$$F(x,y,z) = x.y + x'.y'.z + x.y'.z + z$$

Boolean Fonksiyonları

- Boolean fonksiyonları doğruluk tabloları ile temsil edilirler.
- F fonksiyonu içindeki değişkenlerin alabileceği tüm farklı değerler için fonksiyonun alacağı değerlerin listelendiği tablolara doğruluk tabloları denir.

Boolean Fonksiyonları

- Bir F fonksiyonu içerisinde n adet değişken varsa F fonksiyonun doğruluk tablosunda 2^n satır bulunur.
- Tabloda değişkenlerin olduğu taraf 0'dan 2^n-1 'e kadar ikili sayıların her satıra yazılması ile oluşturulur.
- Sağ taraftaki sütuna ise, her satırdaki değişken değerlerinin fonksiyona yerleştirilmesi ile elde edilen değerler yazılır.

Boolean Fonksiyonları

x	y	z	$F(x,y,z)$
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

$$F(x,y,z) = x + y'.z$$

Boolean Fonksiyonları

$$F(x,y,z) = x + y'.z$$

Boolean fonksiyonu devre üzerinde mantık kapıları kullanılarak gerçekleştirilir

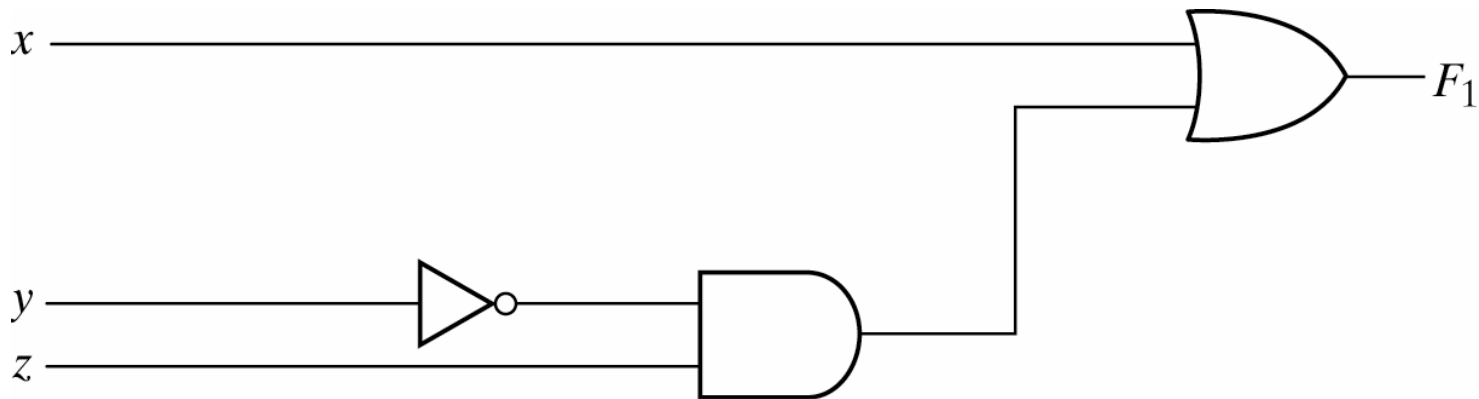


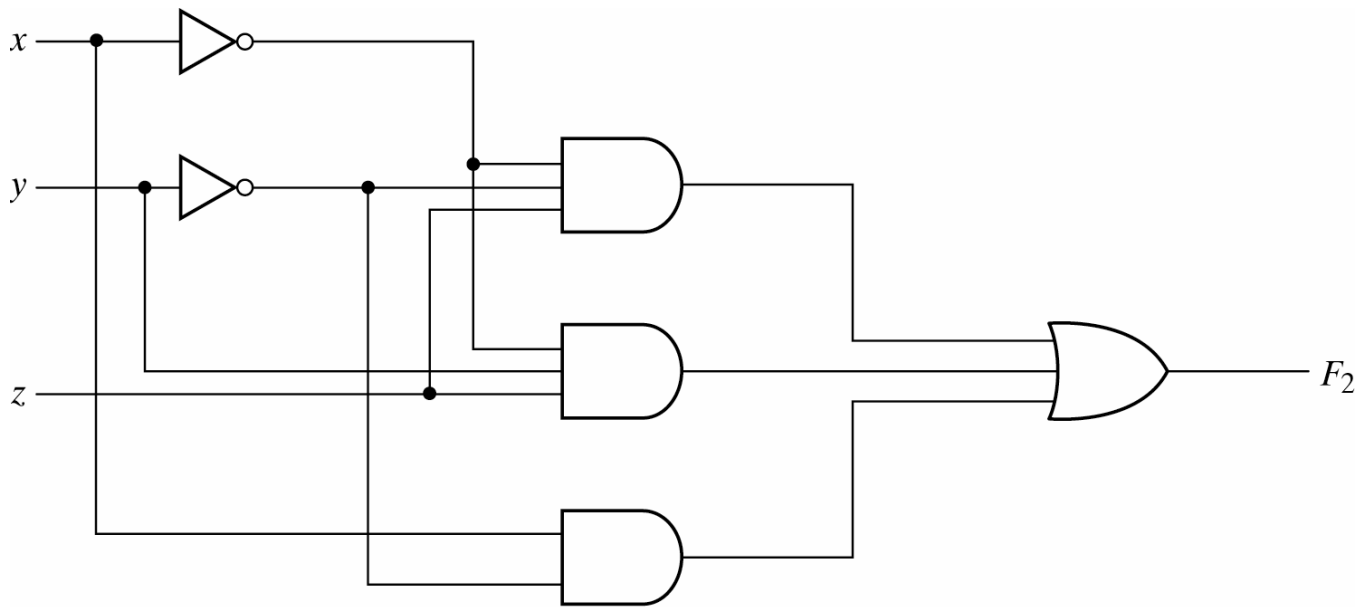
Fig. 2-1 Gate implementation of $F_1 = x + y'.z$

Boolean Fonksiyonları

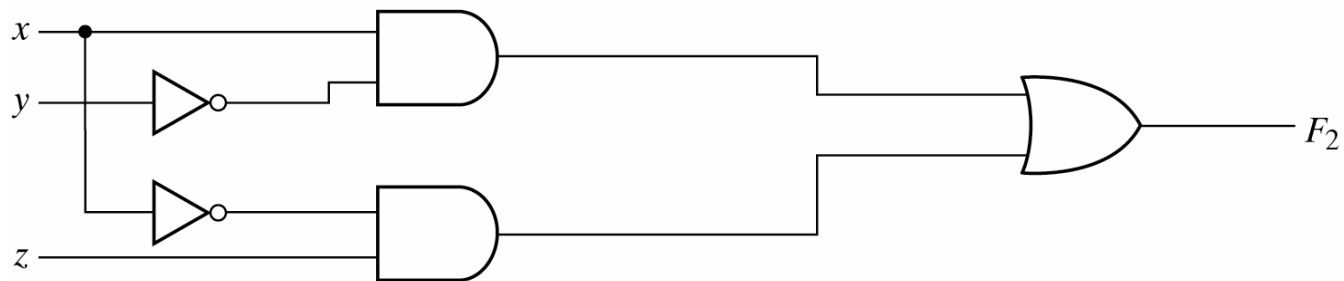
- Herhangi bir Boolean fonksiyonunun bir tane doğruluk tablosu vardır. Ancak doğruluk tabloları aynı olan birden fazla Boolean fonksiyonu olabilir. Bunun sebepleri:
 1. Doğruluk tablosu verilen bir fonksiyon hepside doğru olan farklı şekillerde temsil edilebilir
 2. Fonksiyon daha önce görmüş olduğumuz kurallar ve teoremlerle daha sade bir forma dönüştürülmüş olabilir.

Boolean Fonksiyonları

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



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



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

Fig. 2-2 Implementation of Boolean function F_2 with gates

Cebirsel Sadeleştirme İşlemleri

- Cebirsel sadeleştirme işlemleri ile herhangi bir Boolean fonksiyonu içerisindeki değişken sayısı teoremler kullanılarak azaltılmaya çalışılır.
- Böylece fonksiyon daha az sayıda mantık kapısı kullanılarak gerçekleştirilebilir.

Cebirsel Sadeleştirme İşlemleri

Örnekler:

$$1. \quad x(x'+y) = x x' + xy = 0 + xy = xy$$

$$2. \quad x + x'y = (x+x')(x+y) = 1.(x+y) = x + y$$

$$\begin{aligned} 3. \quad & (x+y)(x+y') = xx + xy' + xy + yy' \\ & = x + xy' + xy = x + x(y'+y) \\ & = x + x = x \quad (\text{veya şöylede sadeleşir}) \\ & = x + xy' + xy \quad (x = x + xz \text{ olduğundan}) \\ & = x + xy \\ & = x \end{aligned}$$

Cebirsel Sadeleştirme İşlemleri

$$\begin{aligned} 4. \quad & xy + x'z + yz \\ &= xy + x'z + yz(x+x') \\ &= xy + x'z + xyz + x'yz \\ &= xy(1+z) + x'z(1+y) \\ &= x.y.1 + x'.z.1 \\ &= xy + x'z \end{aligned}$$

$$\begin{aligned} 5. \quad & (x+y)(x'+z)(y+z) \\ &= (x+y)(x'+z) \quad (4'\text{'ün eşdeğeri, duality, olduğu için}) \end{aligned}$$

Fonksiyonların Tümleyeni

- Herhangi bir F fonksiyonunun tümleyeni De Morgan kuralı ile bulunabilir.
- De Morgan kuralının genelleştirilmesi:

$$(A+B+C+\dots+Z)' = A'B'C'\dots Z'$$

$$(ABCD\dots Z)' = A'+B'+C'+\dots+Z'$$

x	y	F	F'
0	0	0	1
0	1	0	1
1	0	1	0
1	1	0	1

Örn: $F_1 = x'yz' + x'y'z$ ve
 $F_2 = x(y'z' + yz)$ fonksiyonlarının
tümleyenlerini bulunuz.

Çözüm:

$$\begin{aligned} F_1' &= (x'yz' + x'y'z)' \\ &= (x'yz')' \cdot (x'y'z)' \\ &= (x + y' + z) \cdot (x + y + z') \end{aligned}$$

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

Kanonik ve Standart Biçimler

			<i>Mintörm</i> ler			<i>Makstörm</i> ler	
<i>x</i>	<i>y</i>	<i>z</i>	<i>Mintörm</i>	<i>Gösterimi</i>		<i>Makstörm</i>	<i>Gösterimi</i>
0	0	0	$x'.y'.z'$	m_0		$x + y + z$	M_0
0	0	1	$x'.y'.z$	m_1		$x + y + z'$	M_1
0	1	0	$x'.y.z'$	m_2		$x + y' + z$	M_2
0	1	1	$x'.y.z$	m_3		$x + y' + z'$	M_3
1	0	0	$x.y'.z'$	m_4		$x' + y + z$	M_4
1	0	1	$x.y'.z$	m_5		$x' + y + z'$	M_5
1	1	0	$x.y.z'$	m_6		$x' + y' + z$	M_6
1	1	1	$x.y.z$	m_7		$x' + y' + z'$	M_7

Mintörmler

- Yukarıdaki tabloda dördüncü sütundaki terimler mintörm yada standart çarpım terimleri olarak adlandırılır
- Her bir mintörm kendisine karşılık gelen değişkenlerden *0 olanların tümleyenli şekli ve 1 olanların ise tümleyensiz şekliyle* $\sum (\cdot)'$ lenmesinden oluşur ve m_i olarak gösterilir (i sayısı mintörmü oluşturan değişkenlerin aldığı değerlerin onluk tabandaki karşılığı olarak)
- n değişkenli bir Boolean fonksiyonunun 2^n tane mintörmü vardır. $0 \leq i \leq 2^n - 1$

Makstörmler

- Yukarıdaki tabloda altıncı sütundaki terimler makstörm yada standart toplam terimleri olarak adlandırılır
- Her bir makstörm kendisine karşılık gelen değişkenlerden 0 olanların tümleyensiz şekli ve 1 olanların ise tümleyenli şekliyle VEYA (+)' lanmasından oluşur ve M_i olarak gösterilir (i sayısı makstörmü oluşturan değişkenlerin aldığı değerlerin onluk tabandaki karşılığı olarak)
- n değişkenli bir Boolean fonksiyonunun 2^n tane makstörmü vardır. $0 \leq i \leq 2^n - 1$

Mintörmeler ve Makstörmeler

- De Morgan kuralı uygulandığında

$$(m_i)' = M_i$$

olduğu görülür.

- Örnek:

$$m_0 = x'y'z'$$

$$m_0' = (x'y'z')'$$

$$= x + y + z = M_0$$

Boolean Fonksiyonlarının Mintörmeler ile Gerçeklenmesi

- Kural:
Herhangi bir Boolean fonksiyonu 1 olan mintermlerinin VEYA'lanması (+ operatörü ile) olarak gerçekleştirilebilir. Bu durumda fonksiyon:
 - $F = \sum (1 \text{ olan mintermlerin listesi})$ olarak gösterilir

Not: Bu tür bir gerçekleştirme fonksiyonun minimum sayıda mantık kapısı kullanılarak gerçekleştirildiğini garanti etmez

Boolean Fonksiyonlarının Mintörmeler ile Gerçeklenmesi

A	B	C	F(A,B,C)
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

$$F(A,B,C) = A + B'.C$$

$$F(A,B,C) = m_1 + m_4 + m_5 + m_6 + m_7$$

$$F(A,B,C) = A'.B'.C + A.B'.C' + A.B'.C + A.B.C' + A.B.C$$

$$F(A,B,C) = \sum (1, 4, 5, 6, 7)$$

Boolean Fonksiyonlarının Makstörmler ile Gerçeklenmesi

A	B	C	F(A,B,C)	F'(A,B,C)
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	1	0
1	1	1	1	0

$$F'(A,B,C) = m_0 + m_2 + m_3$$

$$(F'(A,B,C))' = (m_0 + m_2 + m_3)'$$

$$F(A,B,C) = (m_0)' \cdot (m_2)' \cdot (m_3)'$$

$$F(A,B,C) = M_0 \cdot M_2 \cdot M_3$$

$$F(A,B,C) = \prod (0, 2, 3)$$

$$F(A,B,C) = (A+B+C) \cdot (A+B'+C) \cdot (A+B'+C')$$

Boolean Fonksiyonlarının Makstörmler ile Gerçeklenmesi

- Kural:
Herhangi bir Boolean fonksiyonu 0 olan makstermlerinin VE' lenmesi (. operatörü ile) olarak gerçekleştirilebilir. Bu durumda fonksiyon:
$$F = \prod (0 \text{ olan makstörmlerin listesi})$$
 olarak gösterilir

Not: Bu tür bir gerçekleştirme fonksiyonun minimum sayıda mantık kapısı kullanılarak gerçekleştirildiğini garanti etmez

Boolean Fonksiyonlarının Mintörmeler ve Makstörmeler ile Gerçeklenmesi

- Örnek: $F = xy + x'z'$ fonksiyonunu mintörmelerin toplamı (VEYA'lanması) ve makstörmelerin çarpımı (VE'lenmesi) olarak yazınız.
- Not: Boolean fonksiyonları için tanımlanan VE (.) ve VEYA (+) operatörleri için bundan sonra çarpım ve toplam ifadeleri de kullanılacaktır

Boolean Fonksiyonlarının Mintörmeler ve Makstörmeler ile Gerçeklenmesi

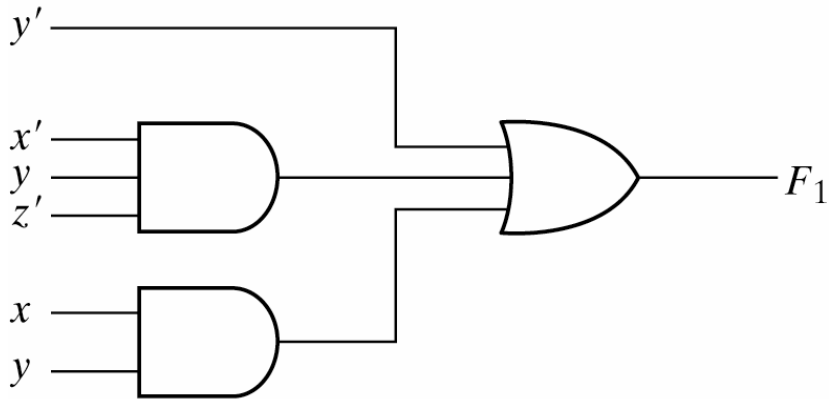
x	y	z	$F(x,y,z)$
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

$$F(x,y,z) = xy + x'z'$$

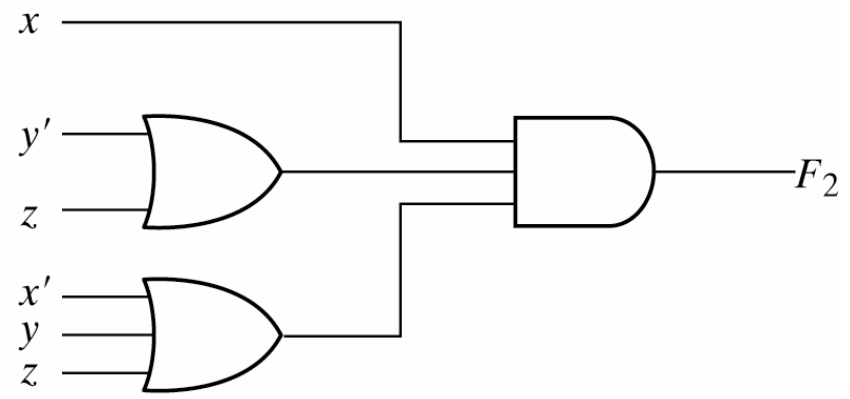
$$F(x,y,z) = \sum (0, 2, 6, 7) = m_0 + m_2 + m_6 + m_7$$

$$F(x,y,z) = \prod (1, 3, 4, 5) = M_1 \cdot M_3 \cdot M_4 \cdot M_5$$

Boolean Fonksiyonlarının Gerçeklenmesi için Kullanılan Standart Formlar



(a) Sum of Products



(b) Product of Sums

Fig. 2-3 Two-level implementation

Boolean Fonksiyonlarının Gerçeklenmesi için Kullanılan Standart Formlar

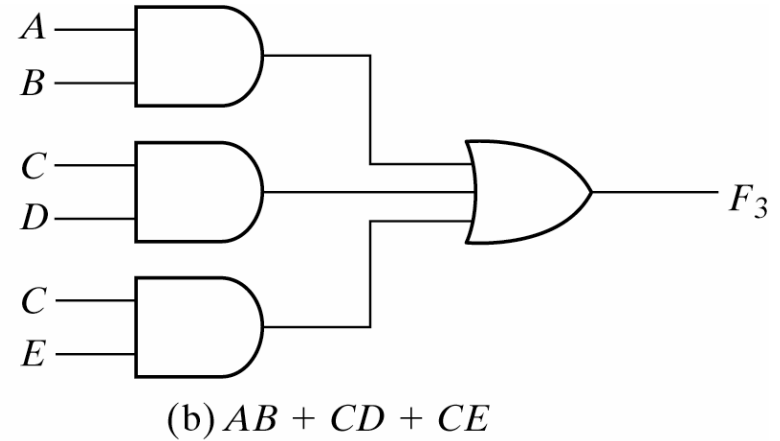
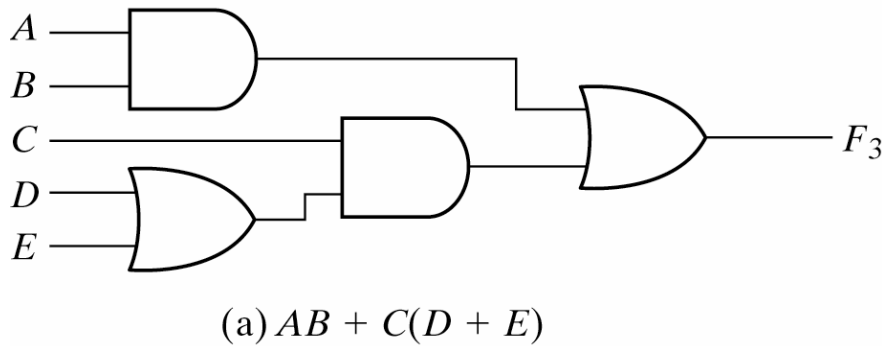


Fig. 2-4 Three- and Two-Level implementation

Diğer Mantıksal Operatörler

- Şimdiye kadar Boolean fonksiyonlarını oluşturmak için (VE), + (VEYA) ve ' (TÜMLEYEN) operatörlerini kullandık.
- Ancak ikili sayılar üzerinde yandaki tabloda göreceğimiz toplam 16 operatör tanımlayabiliriz

Diğer Mantıksal Operatörler

x	y	F_0	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}	F_{14}	F_{15}
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

Not: n tane iki tabanlı değişkenin oluşturabileceği 2^{2^n} farklı fonksiyon vardır

Diğer Mantıksal Operatörler

- F_1 ve F_7 daha önce tanımlamış olduğumuz $.$ ve $+$ Boolean operatörlerini gerçekleyen fonksiyonlardır
- F_{12} ve F_{10} x ve y ' nin tümleyenlerini gösteren fonksiyonlardır
- Aynı şekilde geriye kalan fonksiyonları da gerçekleyen Boolean operatörleri tanımlanabilir

Diğer Mantıksal Operatörler

- KURAL:
- VE (), VEYA(+) VE TÜMLEYEN (‘) operatörleri kullanılarak diğer operatörler elde edilebilir
- Dolayısıyla bu 3 işlev temel işlevler olarak kabul edilir

Diğer Mantıksal Operatörler

Booean Fonksiyonu	Operatör Sembolü	Operatörün Adı	Açıklama
$F_0 = 0$		SIFIR	Her zaman 0
$F_1 = x . y$	$x . y$	VE	x ve y
$F_2 = x . y'$	x / y	YUTMA	x, fakat y değil
$F_3 = x$		TRANSFER	x
$F_4 = x' . y$	y / x	YUTMA	y, fakat x değil
$F_5 = y$		TRANSFER	y
$F_6 = x.y' + x'y$	$x \ominus y$	ÖZELVEYA	x veya y, fakat ikisi beraber değil
$F_7 = x + y$	$x + y$	VEYA	x veya y
$F_8 = (x+y)'$	$x \downarrow y$	DEĞİLVEYA	değilveya
$F_9 = x.y + x'y'$	$(x \ominus y)'$	EŞİTLİK	x ve y aynı (eşit)
$F_{10} = y'$	y'	TÜMLEYEN	y'nin tümleyeni
$F_{11} = x + y'$	$x \subset y$	GEREKTİRME	
$F_{12} = x'$	x'	TÜMLEYEN	x'nin tümleyeni
$F_{13} = x' + y$	$x \supset y$	GEREKTİRME	
$F_{14} = (x . y)'$	$x \uparrow y$	DEĞİLVE	değilve
$F_{15} = 1$	1	BİRİM	Her zaman 1

MANTIK KAPILARI

- Tabloda verilen 16 Boolean fonksiyonundan F_1 , F_7 , F_8 , F_{14} , F_6 , F_9 , F_{10} , F_{12} , F_3 , F_5 sayısal devre tasarımında çokça kullanıldıkları için mantık kapıları olarak entegre devreler içerisinde üretilirler. Bu kapılar şunlardır:
 - **VE** (AND)
 - **VEYA** (OR)
 - **DEĞİLVEYA** (NOR)
 - **DEĞİLVE** (NAND)
 - **ÖZELVEYA** (Ex-OR)
 - **EŞİTLİK** (Ex-NOR)
 - **TÜMLEYEN** (COMPLEMENT)
 - **TRANSFER** (TRANSFER)

MANTIK KAPILARI

- F_2 , F_{11} , F_{13} ise değişme ve birleşme özelliklerinin ikisine birden sahip olamadıklarından mantık kapısı olarak kullanılmazlar

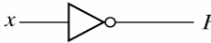
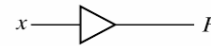
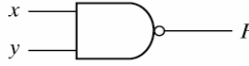
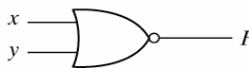
Name	Graphic symbol	Algebraic function	Truth table															
AND		$F = xy$	<table><tr><th>x</th><th>y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	F	0	0	0	0	1	0	1	0	0	1	1	1
x	y	F																
0	0	0																
0	1	0																
1	0	0																
1	1	1																
OR		$F = x + y$	<table><tr><th>x</th><th>y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	F	0	0	0	0	1	1	1	0	1	1	1	1
x	y	F																
0	0	0																
0	1	1																
1	0	1																
1	1	1																
Inverter		$F = x'$	<table><tr><th>x</th><th>F</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	x	F	0	1	1	0									
x	F																	
0	1																	
1	0																	
Buffer		$F = x$	<table><tr><th>x</th><th>F</th></tr><tr><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td></tr></table>	x	F	0	0	1	1									
x	F																	
0	0																	
1	1																	
NAND		$F = (xy)'$	<table><tr><th>x</th><th>y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	x	y	F	0	0	1	0	1	1	1	0	1	1	1	0
x	y	F																
0	0	1																
0	1	1																
1	0	1																
1	1	0																
NOR		$F = (x + y)'$	<table><tr><th>x</th><th>y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	x	y	F	0	0	1	0	1	0	1	0	0	1	1	0
x	y	F																
0	0	1																
0	1	0																
1	0	0																
1	1	0																
Exclusive-OR (XOR)		$F = xy' + x'y$ $= x \oplus y$	<table><tr><th>x</th><th>y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	x	y	F	0	0	0	0	1	1	1	0	1	1	1	0
x	y	F																
0	0	0																
0	1	1																
1	0	1																
1	1	0																
Exclusive-NOR or equivalence		$F = xy + x'y'$ $= (x \oplus y)'$	<table><tr><th>x</th><th>y</th><th>F</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	x	y	F	0	0	1	0	1	0	1	0	0	1	1	1
x	y	F																
0	0	1																
0	1	0																
1	0	0																
1	1	1																

Fig. 2-5 Digital logic gates

Mantık Kapılarının İkiden Fazla Girişe Sahip Olabilmesi

- Herhangi bir mantık kapısının ikiden fazla girişe sahip olabilmesi için hem değişme hem de birleşme özelliğine sahip olması gerekir.
- Örn: F_3 fonksiyonunu gerçekleyen YUTMA operatörünün değişme ve birleşme özelliğine sahip olup olmadığını gösteriniz

Mantık Kapılarının İkidenden Fazla Girişe Sahip Olabilmesi

x	y	xy'	yx'
0	0	0	0
0	1	0	1
1	0	1	0
1	1	0	0

$$x/y \neq y/x$$

$$x.y' \neq y.x'$$

$$(x / y) / z \neq x / (y / z)$$

$$(x . y') / z \neq x / (y . z')$$

$$x . y' . z' \neq x . (y . z')'$$

$$x . y' . z' \neq x . (y' + z)$$

Mantık Kapılarının İkiden Fazla Girişe Sahip Olabilmesi

- VE (AND) ve VEYA (OR) operatörleri değişme ve birleşme özelliklerini beraber sağladıklarından birden fazla girişe sahip olabilirler
- Çok kullanılan DEĞİLVE (NAND) ve DEĞİLVEYA (NOR) operatörleri ise değişme özelliğine sahip oldukları fakat birleşme özelliğine sahip olmadıklarından çok girişli NAND ve NOR kapıları çok girişli AND ve OR kapılarının tümleyenleri olarak tanımlanır

Mantık Kapılarının İkiden Fazla Girişe Sahip Olabilmesi

- Örn: NOR (DEĞİLVEYA) kapısını için:
- $(x \downarrow y) \downarrow z = [(x+y)' + z]' = (x+y).z' = xz' + yz'$
- $x \downarrow (y \downarrow z) = [(x + (y+z)')]' = x'(y+z) = x'y + x'z$
- Dolayısıyla çok girişli NAND ve NOR kapıları şu şekilde tanımlanır:
- $(x \downarrow y) \downarrow z = (x + y + z)'$
- $(x \uparrow y) \uparrow z = (x . y . z)'$

Mantık Kapılarının İkiden Fazla Girişe Sahip Olabilmesi

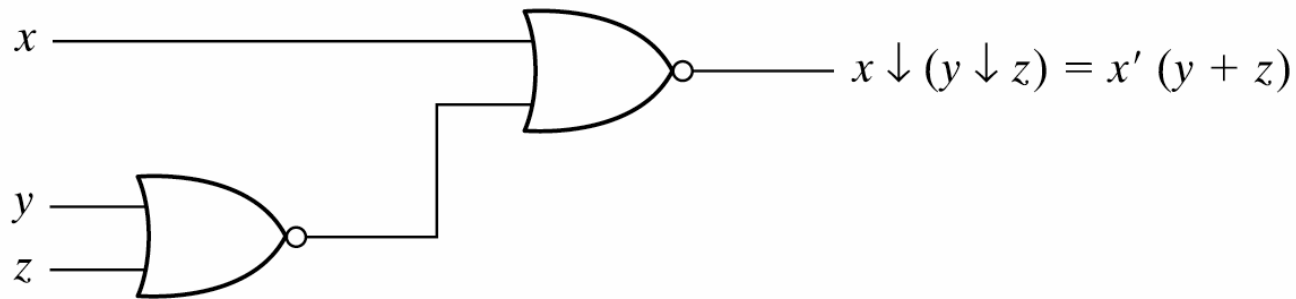
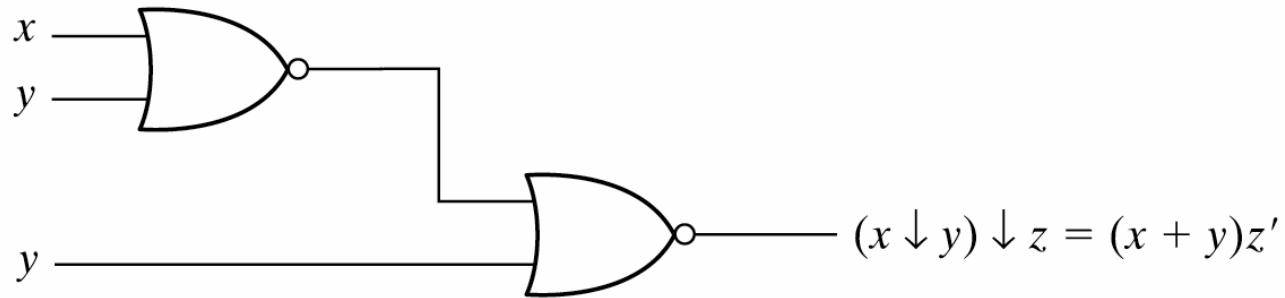
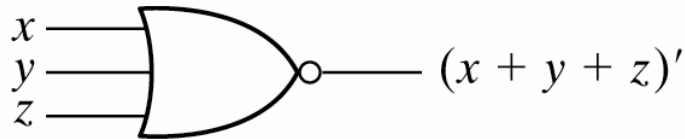
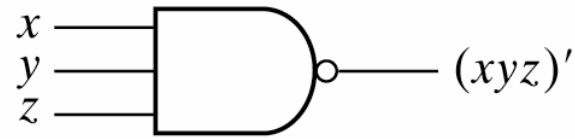


Fig. 2-6 Demonstrating the nonassociativity of the NOR operator; $(x \downarrow y) \downarrow z \neq x \downarrow (y \downarrow z)$

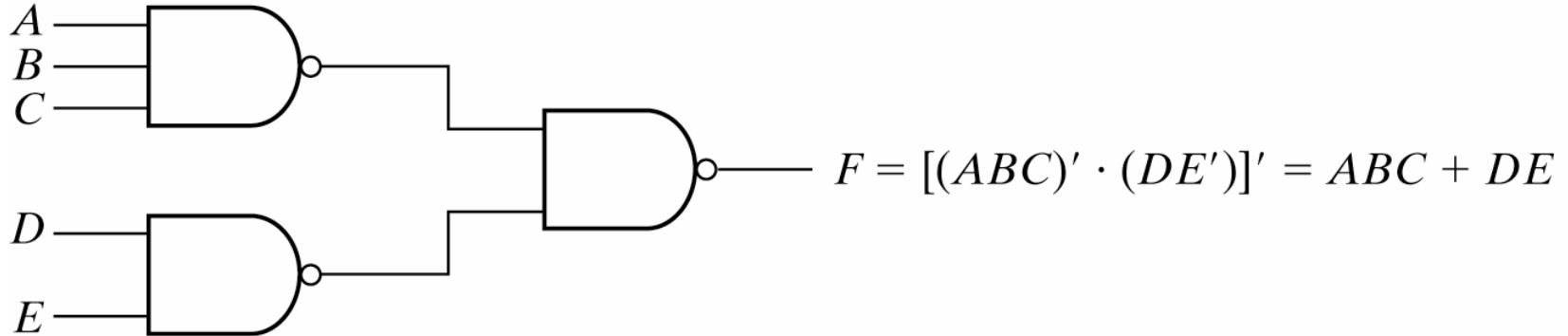
Mantık Kapılarının İki'den Fazla Girişe Sahip Olabilmesi



(a) 3-input NOR gate



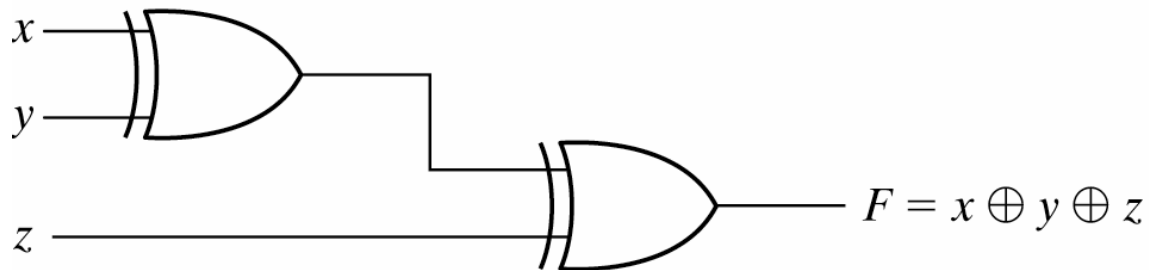
(b) 3-input NAND gate



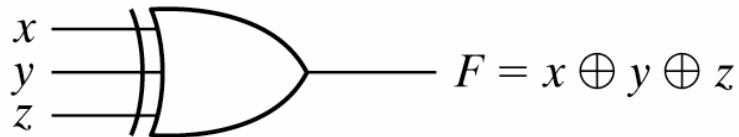
(c) Cascaded NAND gates

Fig. 2-7 Multiple-input and cascaded NOR and NAND gates

Mantık Kapılarının İkidenden Fazla Girişe Sahip Olabilmesi



(a) Using 2-input gates



(b) 3-input gate

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

(c) Truth table

Fig. 2-8 3-input exclusive-OR gate

Pozitif ve Negatif Mantık

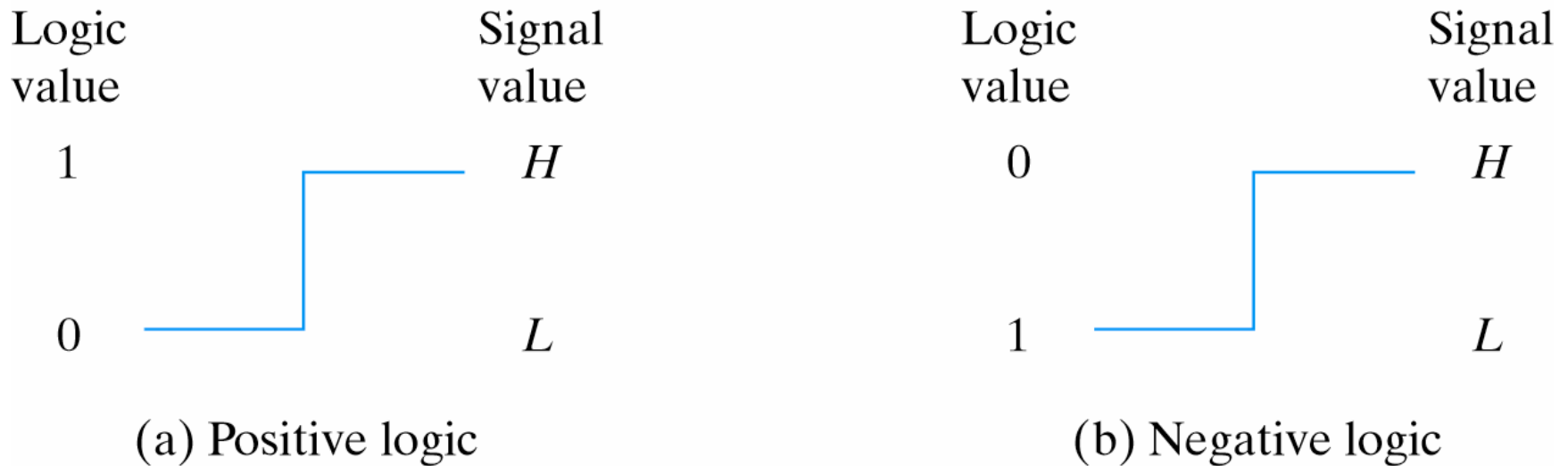
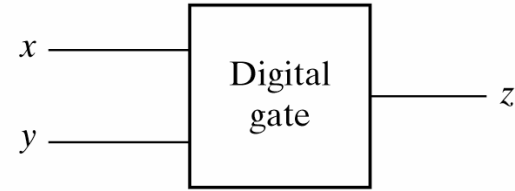


Fig. 2-9 signal assignment and logic polarity

x	y	F
L	L	L
L	H	L
H	L	L
H	H	H

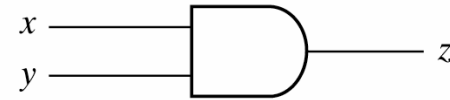
(a) Truth table with H and L



(b) Gate block diagram

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

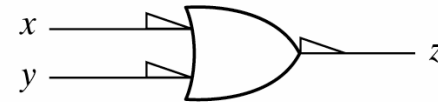
(c) Truth table for positive logic



(d) Positive logic AND gate

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

(e) Truth table for negative logic



(f) Negative logic OR gate

Fig. 2-10 Demonstration of positive and negative logic

Entegre Devreler

- Küçük Ölçekli Entegre Devreler (Small Scale Integrated Circuits, SSI):

İçlerinde onlarla ifade edilebilen mantık kapısı bulunduran entegre devrelerdir

- Orta Ölçekli Entegre Devreler (Medium Scale Integrated Circuits, MSI):

İçlerinde 10 ila 100 arasında mantık kapısı bulunduran entegre devrelerdir. Örn: Kod çözücüler (Multiplexers)

Entegre Devreler

- Büyük Ölçekli Entegre Devreler (Large Scale Integrated Circuits, LSI):
İçlerinde 100 ila birkaç bin arasında mantık kapısı bulunduran entegre devrelerdir. Örnek: Programlanabilir hafızalar
- Çok Büyük Ölçekli Entegre Devreler (Very Large Scale Integrated Circuits, VLSI):
İçlerinde yüz binler yada milyonlarca mantık kapısı bulunduran entegre devrelerdir. Örnek: Mikroişlemciler

Entegre Devre Üretim Teknolojileri

- Entegre devreler aynı zamanda mantık kapılarını oluşturan temel elektronik devrelerin farklılıklarına görede sınıflandırılırlar
- Bu teknolojiler NAND, NOR, NOT mantık devrelerinin hangisinin daha karmaşık devrelerin oluşturulmasında kullanıldığına göre değişir
- En yaygın olan teknolojiler şunlardır

Entegre Devre Üretim Teknolojileri

- TTL : NAND kapılarının transistörler kullanılarak gerçekleştirildiği devrelerdir. Çok uzun süredir kullanılmaktadırlar
- ECL : Yüksek hız gerektiren uygulamalarda kullanılırlar
- CMOS : Düşük güç kullanımları ile bilinirler. Daha çok VLSI devrelerinde kullanılırlar

Bilgisayar Destekli Devre Dizaynı

- Günümüzde karmaşık sayısal devreler farklı yöntemlerle tasarımları yapıp entegre devreler yada kartlar üzerinde gerçekleştirilmektedirler. En çok gerçekleştirildikleri metodlar:
 - PLD (Programmable Logic Devices)
 - FPGA (Field Programmable Gate Arrays)
 - ASIC (Application Specific Integrated Circuits)
 - Full Custom IC

Bilgisayar Destekli Devre Dizaynı

- Bu sayısal devrelerin tasarımlarında farklı yöntemler kullanılarak yapılabilmektedir.
Bunlar:
 - ✓ Bilgisayar Paket Programları
 - ✓ HDL (Hardware Description Language)