Boole-algebra
Digitális információ ábrázolása

A digitális technika feladata: információ ábrázolása és feldolgozása a digitális technika eszközeivel

Szakterület	Jelkészlet
Digitális technika	"0" és "1"
Fizika	"*low*" és "*high*" feszültség ill. áramerősség
Logika	"igaz" és "hamis" ill. "*true*" és "*false*"

1: megfelelője a gyakorlatban általában „aktív”
0: megfelelője a gyakorlatban általában „passzív”

digitális, lat. „digitus”= ujj, ujjak → tízes számrendszer, információmennyiség megszámolható, adott véges számú érték fordulhat elő
analóg - végtelen számú értékszint fordulhat elő
bináris (ang. *binary*)- kétértékű rendszer, kettős számrendszer
bit: (ang. "***bi***nary digi***t***”) legkisebb bináris információmennyiség
(***B***asic ***I***ndissoluble Uni***t***)

Gépi szinten **bitsorozat** hordozza az információt kódolva, ami ember számára nehézkesen értelmezhető

Példa:

0100000101000001001000011

Boole-algebra
Digitális információ ábrázolása

bináris:	01000001	01000010	01000011
ASCII:	A	B	C

ASCII kód: A biteket nyolcasával csoportosítjuk és a csoportokon belül előforduló minden egyes bitsorozat-változathoz egy-egy karaktert rendelünk hozzá.

1 Byte = 8 bit-ből álló csoport
1 kByte = 2^{10} Byte = 1024 Byte

A bitek csoportosítása és az adott csoportok információval (pl. helyiértékkel) való felruházása a gépi szint érthetőbbé tételét segíti
(lásd hexadecimális számábrázolás).

Természetes számok ábrázolása:

$$n = \sum_{i=0}^{N} b_i \cdot B^i$$

B: a számrendszer alapja $B \in N, \quad B \geq 2$
és
b$_i$: a helyiértékek együtthatói $b_i \in N \cup \{0\}, \; 0 \leq b_i < B$

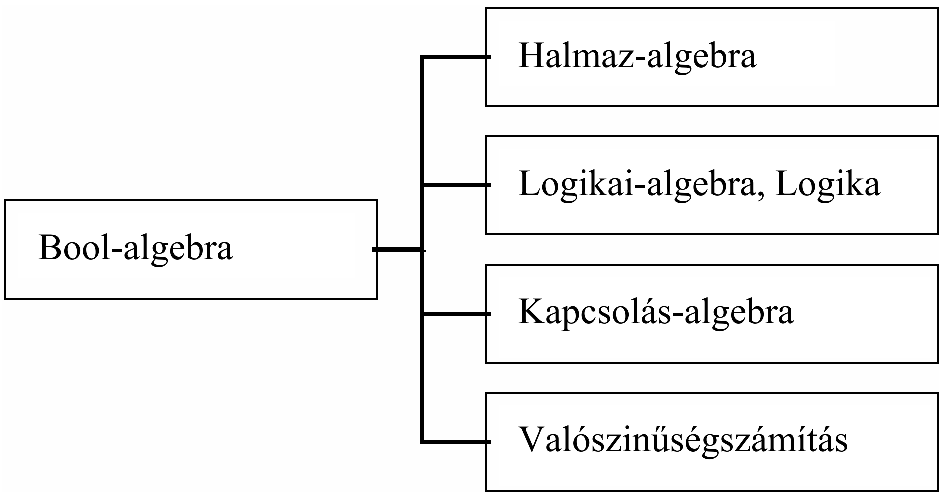
Az elterjedt írásmód szerint a számokat az alábbi formában, csak a b_i együtthatók felsorolásával (csökkenő helyiérték szerinti sorrendben) ábrázolják: [Mivel több számrendszer ugyanazokat a számjegyeket (szimbólumokat) használja, esetenként a számrendszer alapjának megadása is szükséges alsó indexben.A sor elején lévő nullák elhagyhatóak.]

$$n = (b_N b_{N-1} b_{N-2} \cdots b_1 b_0)_B$$

Boole-algebra
Alapok

Matematikai problémák megoldására szolgáló segédeszközök összefoglaló neve „algebra”. A digitális technika komplex problémáinak kezelésére jól használható

Boole-algebra.



A Bool-algebra felosztása

Az angol matematikus, George Boole (1815..1864) dolgozta ki a ma Boole-algebrának nevezett kételemes információs algebra-rendszer alapjait, melynek egy része használatos a digitális technikában.

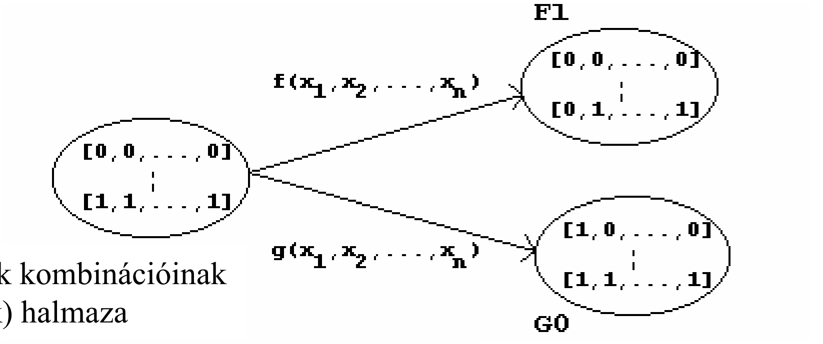
A Boole algebra szűkebb értelemben a bináris információkon végzett műveletekre jellemző szabályok rendszere. Ezen műveleteket digitális függvények, más néven **Boole-függvények** közvetítik. A függvényváltozók és a függvényérték is csak a két értéket, „0” vagy „1” vehetik fel.

A Bool-algebrában használatos a **halmazelmélet diagramm-ábrázolása** is. Definíció szerint halmaznak nevezzük bizonyos elemek összességét, ahol a halmazt a hozzá tartozó elemek felsorolásával adhatjuk meg, például:

•A tízes számrendszer számjegyeinek halmaza: A={0,1,2,3,4,5,6,7,8,9}
•A bináris számjegyek (bináris értékek) halmaza: B={0,1}
•A kétjegyű bináris szavak halmaza: C= {[0,0],[0,1],[1,0],[1,1]}
•A logikai értékek halmaza: D= {igaz,hamis}

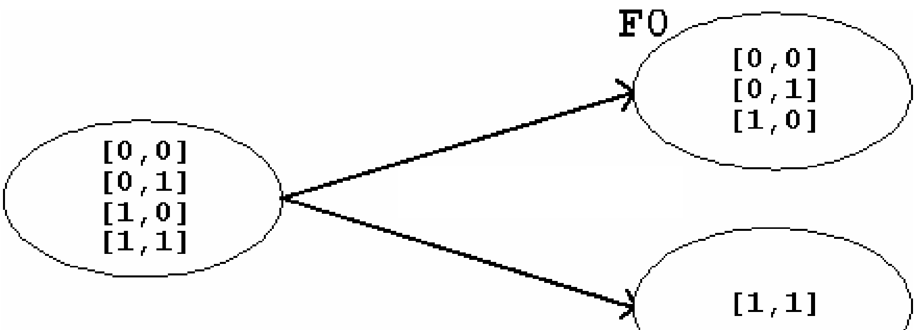
Boole-algebra
Logikai-függvények ábrázolási módjai

Ha n db halmazt egy függvény (művelet) segítségével egymással összekötünk, akkor az n db halmaz elemeinek minden egyes kombinációjához (n elemből álló sorhoz) a függvény egy-egy függvényértéket rendel.
A digitális technikában a halmazok két elemből állnak, azaz a független változók két értéket vehetnek fel, szintúgy a függvényérték.
n db bináris változó esetén 2^n db „variáció”, másnéven „n-sor” fordulhat elő. Ezen „n-sorok” halmazát egy adott $f(x_1,x_2,\ldots,x_n)$ függvény két részre bontja, F1 halmazra és F0 halmazra.



Példa: AND (ÉS) bináris Boole-függvény (n=2)

A függvényérték akkor „1”, ha az összes független változó értéke $x_1=x_2=$„1”



1. Halmazelméleti grafikus ábrázolás

x_1	x_2	$f(x_1,x_2)$
0	0	0
0	1	0
1	0	0
1	1	1

2. Ábrázolás igazságtáblázattal

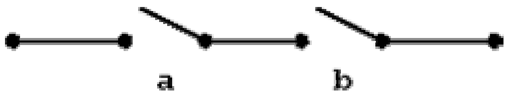
$$f(x_1,x_2) = x_1 \,\&\, x_2$$

$$f(x_1,x_2) = x_1 \, x_2$$

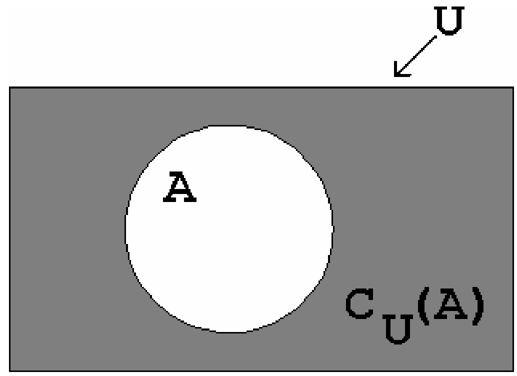
$$f(x_1,x_2) = x_1 \cdot x_2$$

3. Ábrázolás matematikai jelekkel

Boole-algebra
Boole-függvények ábrázolási módjai

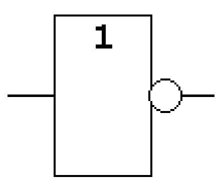
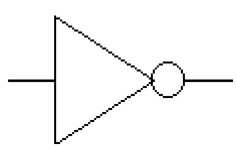
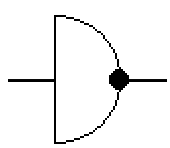


4. ÉS függvény ábrázolása kapcsolókkal



NOT függvény:
f(0)=1
f(1)=0

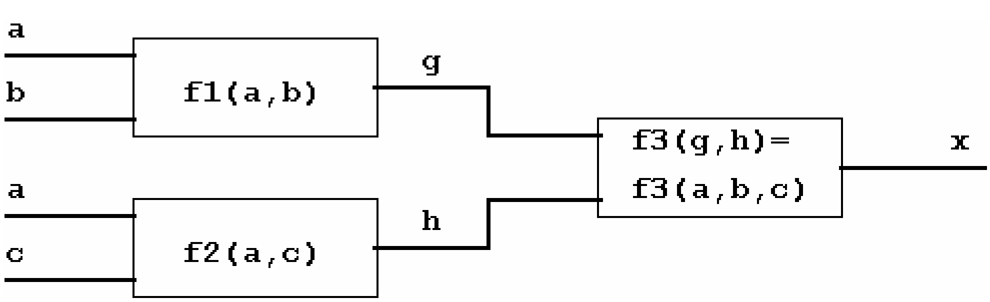
5. NOT függvény ábrázolása Venn-diagrammal
$C_U(A)$: A halmaz U-ra vonatkoztatott komplemenshalmaza

DIN/IEC	ANSI/IEEE	DIN (régi)
		

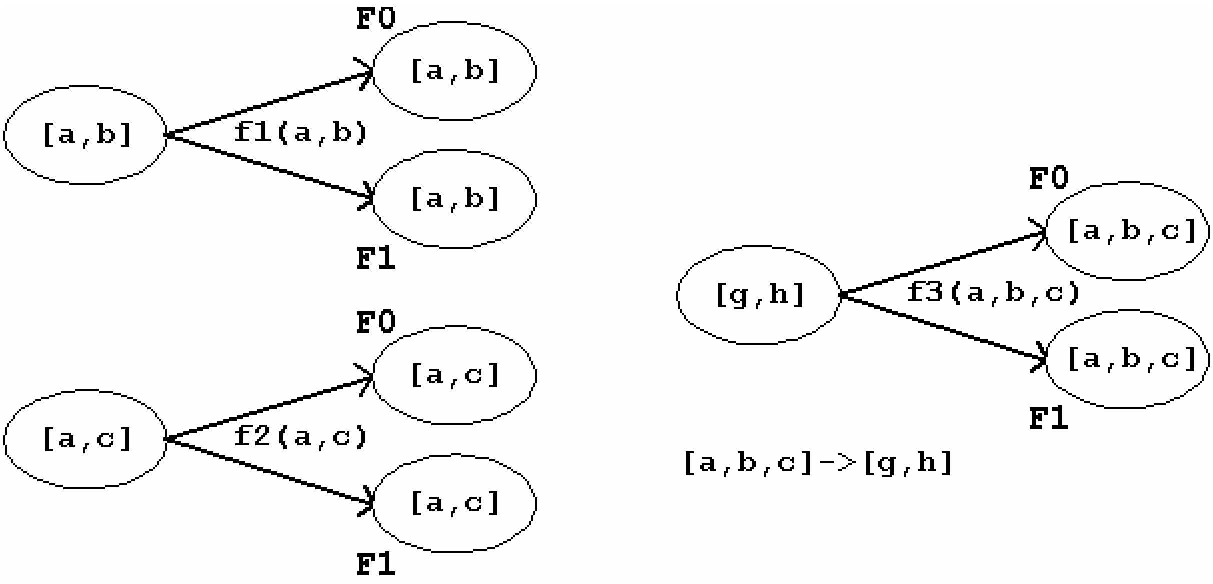
6. NOT függvény ábrázolása rajzi szimbolikus jelekkel

Összetett függvények grafikus ábrázolása:

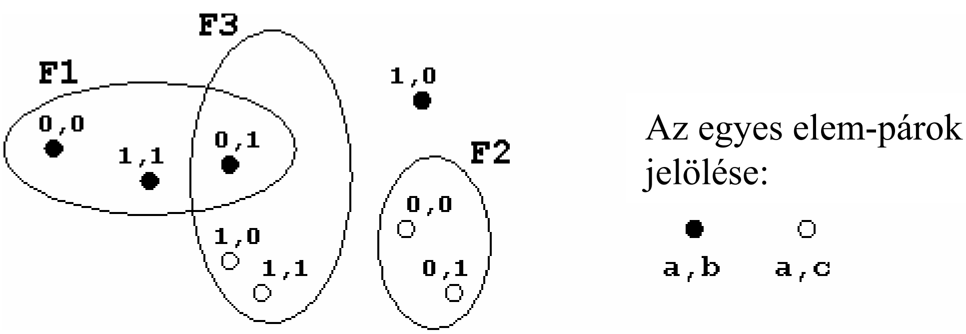
Példa: f1(a,b) és f2(a,c) függvények kimenetei képezik F3 bemeneti változóit.



Összetett függvények grafikus ábrázolása:



Halmazok leképezésének ábrázolása általános formában



Példa Venn-diagram alkalmazására összetett függvényeknél

$f_1(a,b) = 1$ ha $(a,b) \in \{[0,0],[0,1],[1,1]\}$ $\quad f_1(1,0) = 0$
$f_2(a,c) = 1$ ha $(a,c) \in \{[0,0],[0,1]\}$ $\quad f_2(a,c) = 0$ ha $(a,c) \in \{[1,0],[1,1]$

A példában grafikusan megadott f_1 és f_2 egyértelműen definiálva van, viszont az f_3-ra ez nem érvényes, lásd például az $f_3(1,0,0)$ függvényértéket!

Boole-algebra
Logikai-függvények osztályozása

Az n-változós logikai függvény igazságtáblázatának általános formája:

x_n	...	x_2	x_1	$f(x_1,x_2,...,x_n)$
0	...	0	0	f_1
0	...	0	1	f_2
0	...	1	0	f_3
0	...	1	1	f_4
:	...	:	:	:
:	...	:	:	:
1	...	1	1	f_{2^n}

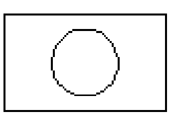
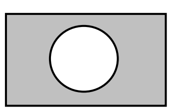
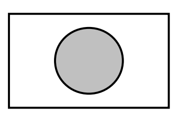
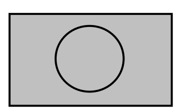
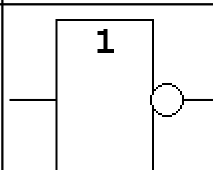
x_i - az n-változós $f(x_1,x_2,\ldots x_n)$ függvény i-dik változója

f_i - függvény-érték $\{0,1\}$

Az egymástól különböző összes lehetséges függvény száma: $m = 2^{(2^n)}$

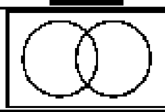
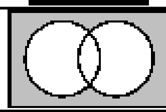
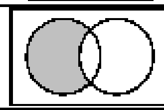
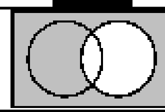
Ezen belül csak néhány olyan függvény van, melyek nem triviálisak és a gyakorlatban jelentőséggel bírnak.

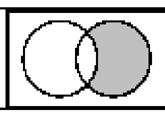
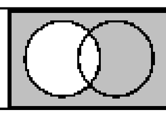
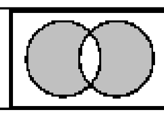
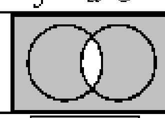
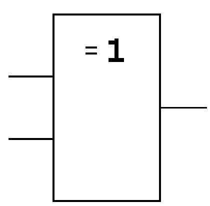
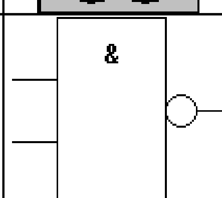
n=1, Egyváltozós függvény

Név		**0-függvény**	**Negáció**	**Azonosság**	**1-függvény**
Igazságtáblázat	**a**	**f(a)**	**f(a)**	**f(a)**	**f(a)**
	0	**0**	**1**	**0**	**1**
	1	**0**	**0**	**1**	**1**
Mat. kifejezés		$f = 0$	$f = \bar{a}$	$f = a$	$f = 1$
Venn-Diagramm					
Kapcsolási jel **(DIN/IEC)**					

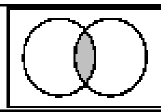
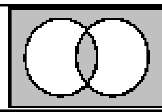
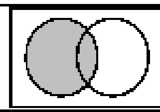
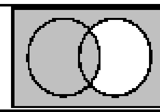
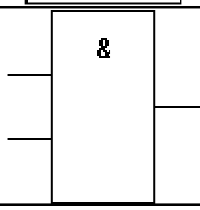
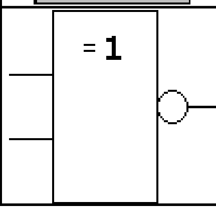
Boole-algebra
Logikai-függvények osztályozása

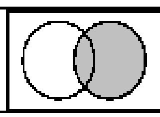
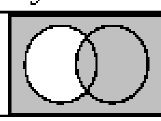
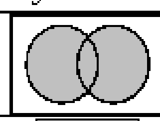
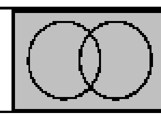
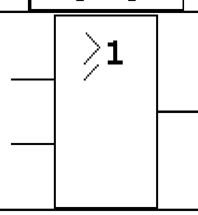
n=2, Kétváltozós függvény

Név			0-függvény	NOR		$\overline{b}$
	b	**a**	**f(a,b)**	**f(a,b)**	**f(a,b)**	**f(a,b)**
	0	0	0	1	0	1
Igazságtáblázat	0	1	0	0	1	1
	1	0	0	0	0	0
	1	1	0	0	0	0
Matematikai képlet			[redacted]	[redacted]	[redacted]	[redacted]
Venn-Diagram						
Kapcsolási rajzjel (DIN/IEC)				[redacted]		

Név				$\overline{a}$	XOR	NAND
	b	**a**	**f(a,b)**	**f(a,b)**	**f(a,b)**	**f(a,b)**
	0	0	0	1	0	1
Igazságtáblázat	0	1	0	0	1	1
	1	0	1	1	1	1
	1	1	0	0	0	0
Matematikai képlet			[redacted]	$f=\overline{a}$	$f=a\oplus b$	$f=\overline{a\wedge b}$ $f=\overline{a\cdot b}$
Venn-Diagram						
Kapcsolási rajzjel (DIN/IEC)						

n=2, Kétváltozós függvény

Név			AND	XNOR	a	
Igazságtáblázat	b	a	f(a,b)	f(a,b)	f(a,b)	f(a,b)
	0	0	0	1	0	1
	0	1	0	0	1	1
	1	0	0	0	0	0
	1	1	1	1	1	1
Matematikai képlet			$f = a \wedge b$ $f = a \cdot b$	$f = \overline{a \oplus b}$	$f = a$	$f = a \vee \overline{b}$ $f = a + \overline{b}$
Venn-Diagram						
Kapcsolási rajzjel (DIN/IEC)						

Név			b		OR ODER	1-függvény
Igazságtáblázat	b	a	f(a,b)	f(a,b)	f(a,b)	f(a,b)
	0	0	0	1	0	1
	0	1	0	0	1	1
	1	0	1	1	1	1
	1	1	1	1	1	1
Matematikai képlet			$f = b$	$f = \overline{a} \vee b$ $f = \overline{a} + b$	$f = a \vee b$ $f = a + b$	$f = 1$
Venn-Diagram						
Kapcsolási rajzjel (DIN/IEC)						

Boole-algebra
Alapszabályai

Boole-posztulátumok (axiómák):

A szűkebb értelemben vett Boole-algebra nem más, mit egy halmaz A, melynek elemei a „0” és az „1” és amelyikben definiálva vannak az „ÉS” és a „VAGY” (ill. „+” és „·”) műveletek úgy, hogy az alábbiak érvényesek:

$a=0$ vagy $a=1$, $a \in A$
$0 \cdot 0 = 0$
$1+1 = 1$
$0+0 = 0$
$1 \cdot 1 = 1$
$1 \cdot 0 = 0 \cdot 1 = 0$
$1+0 = 0+1 = 1$

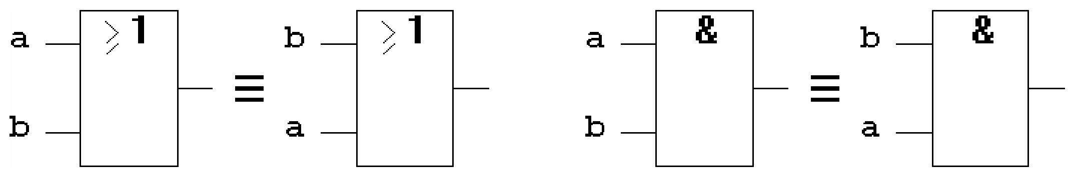
Boole-tételek:

A Boole-tételek a posztulátumokból bármikor levezethetők, például az igazságtáblázatok segítségével. A tételek olyan szabályok, melyek a Bool-változók közötti alapösszefüggéseket fejezik ki. Segítségükkel bonyolult kapcsolások manipulálhatók, függvények egyszerűsíthetők.

1. Kommutativitás (felcserélhetőség)

$a + b = b + a$
$a \cdot b = b \cdot a$

A diszjunkció („+”) és a konjunkció („·”) műveletek operandusai felcserélhetőek

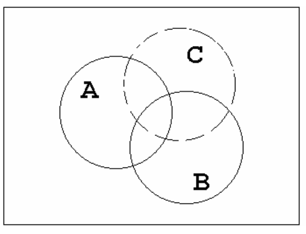


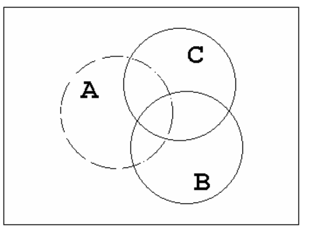
Boole-algebra
Alapszabályai

2. Asszociativitás (csoportosíthatóság)

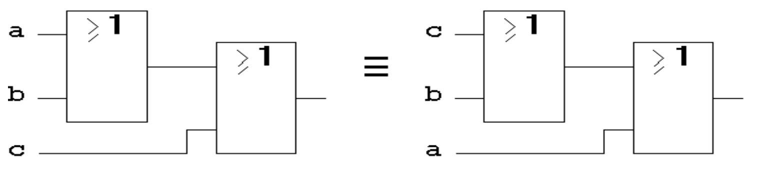
a) $(a + b) + c = a + (b + c)$
b) $(a \cdot b) \cdot c = a \cdot (b \cdot c)$

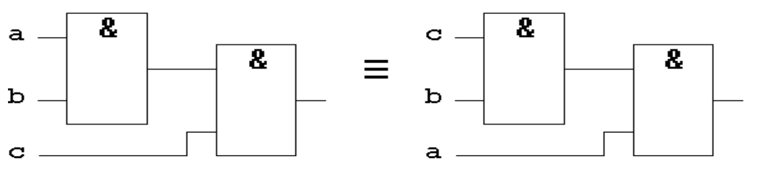
A diszjunkció („+”) és a konjunkció („·”) műveletek operandusai tetszőlegesen csoportosíthatóak





Nyilvánvalóan teljesen mindegy, hogy az A, B és C halmazokat milyen sorrendben kapcsoljuk össze egymással.





Az algebrai alak zárójele a kapcsolásban egy kapuáramkörnek felel meg.

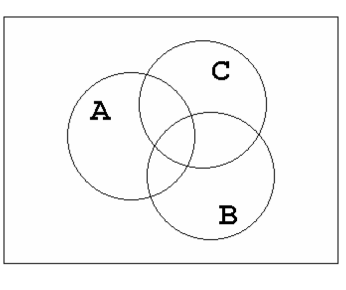
3. Disztributivitás (eloszthatóság)

a) $a + (b \cdot c) = (a + b) \cdot (a + c)$

b) $a \cdot (b + c) = (a \cdot b) + (a \cdot c)$

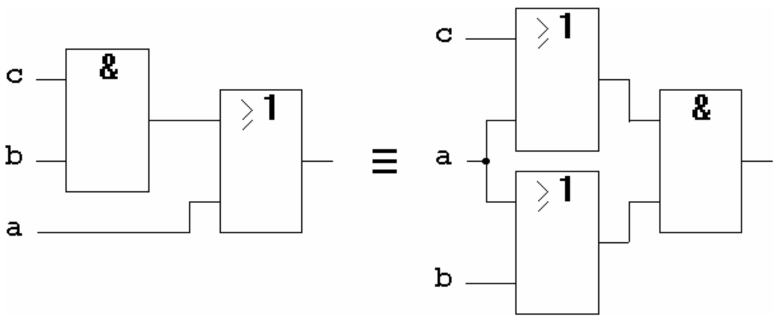
$= a \cdot b + a \cdot c$

A diszjunkció és a konjunkció műveletek „eloszthatóak”, azaz a „VAGY” elosztható az „ÉS”-re, és az „ÉS” elosztható a „VAGY”-ra.



A disztributivitás törvénye lépésről lépésre végigvezetve a halmaz-diagramon is igazolható.

Disztributivitás, folyt.

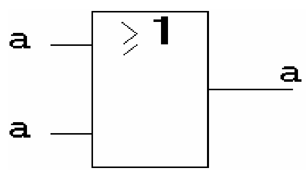


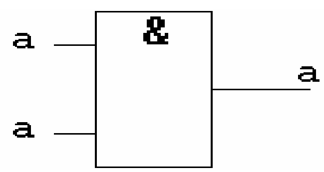
Ennek a tételnek a felhasználásával az egyes kapuáramkörök bemeneteinek a számát (ang. *pin count*) csökkenteni lehet.

4. Azonosság-tétel

a) $a + a = a$

b) $a \cdot a = a$

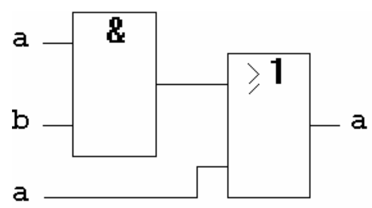




A kapcsolás logikailag felesleges, redundáns, viszont gyakorlati jelentősége, hogy késlelteti az „a” jelet.

5. Abszorpciós tétel (Beolvadási-, Redundancia-tétel)

a) $a + (a \cdot b) = a$
b) $a \cdot (a + b) = a$



Az "a + (a · b)” kifejezésre érvényes, hogy ha „a” igaz, akkor a kifejezés értéke „b”-től független. Amennyiben viszont „a” hamis, akkor az (a · b) sem lehet igaz. A kifejezés értékét tehát egyesegyedül „a” határozza meg, „b” elnyelődik. Ez analóg módon érvényes b) esetre is.

6. Nulla- és Egy-tétel

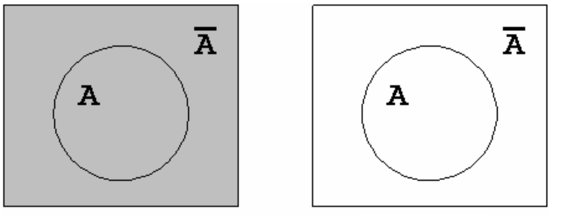
a) $a + 0 = a \qquad a + 1 = 1$
b) $a \cdot 0 = 0 \qquad a \cdot 1 = a$

7. Komplemens-tétel

a) $a + \bar{a} = 1$

b) $a \cdot \bar{a} = 0$

A halmaz minden „a" eleméhez létezik egy „ā"komplemens elem.



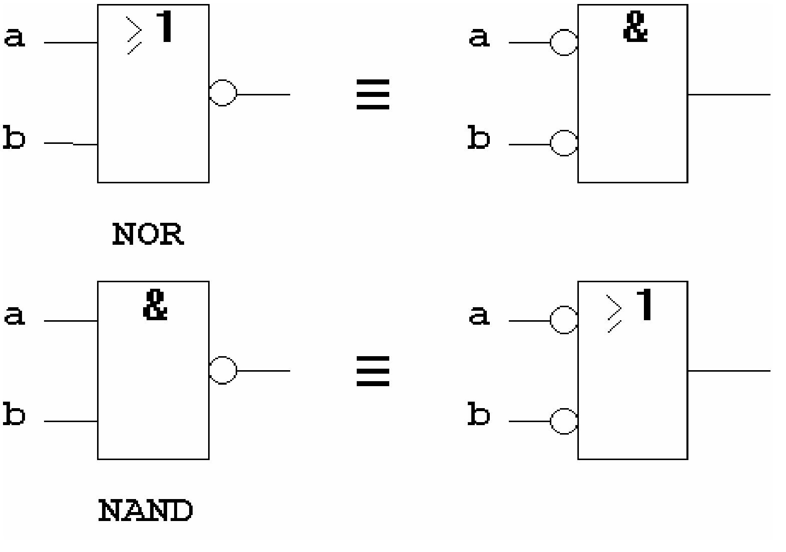
$A + \overline{A} = 1$ $A \cdot \overline{A} = 0$

8. De Morgan-tétel

a) $\overline{(a + b)} = \bar{a} \cdot \bar{b}$

b) $\overline{(a \cdot b)} = \bar{a} + \bar{b}$

De Morgan tétele lehetővé teszi az egymástól eltérő log. függvények közötti átalakítást, ezért nagy a gyakorlati jelentősége.
NOR → AND
NAND → OR árváltás közvetlenül lehetséges.



Boole-algebra
Dualitás, NAND és NOR rendszerek

A **dualitás elve** a Boole-algebra posztulátumaiból következik:

Minden állításnak, mely a Boole-posztulátumokból következik, létezik egy duális megfelelője, mely a műveletek („+” és „·”) és az operandusok („0” és „1”) egyidejű felcserélésével keletkezik.

De Morgan-tétel általános alakban:

$$\overline{f_1(x_1, x_2, x_3, \ldots, x_n ;+,\cdot)} = f_1(\overline{x}_1, \overline{x}_2, \overline{x}_3, \ldots, \overline{x}_n ;\cdot,+) = f_2$$

f_1 függvény duáltjának (komplemensének) nevezzük azt az f_2 függvényt, amely az f_1-től csak abban különbözik, hogy a függvényértéke minden esetben az f_1 aktuális függvényértékének negáltja.

A konjunkció (ÉS) és a diszjunkció(VAGY) egymásnak duáltjai.

Teljes logikai rendszernek nevezzük azokat az elméleti logikai rendszereket, melyekkel teljes egészében leírható és megoldható az összes logikai probléma. Ilyen a Bool-algebra is, melyet a „0” és „1” operandusok (elemek) és a három alapművelet - az ÉS, a VAGY és a negálás (konjunkció, diszjunkció, komplementáció) - alkotnak. Az összes létező logikai áramkör ezen alapműveletekből felépíthető.

Létezik azonban még két másik teljes logikájú rendszer is, melyek szintén két elemből, viszont a három alapművelet helyett csak egyből állnak. Ez az alapfüggvény (művelet) az alábbi lehet.

a) NAND (Not-AND)

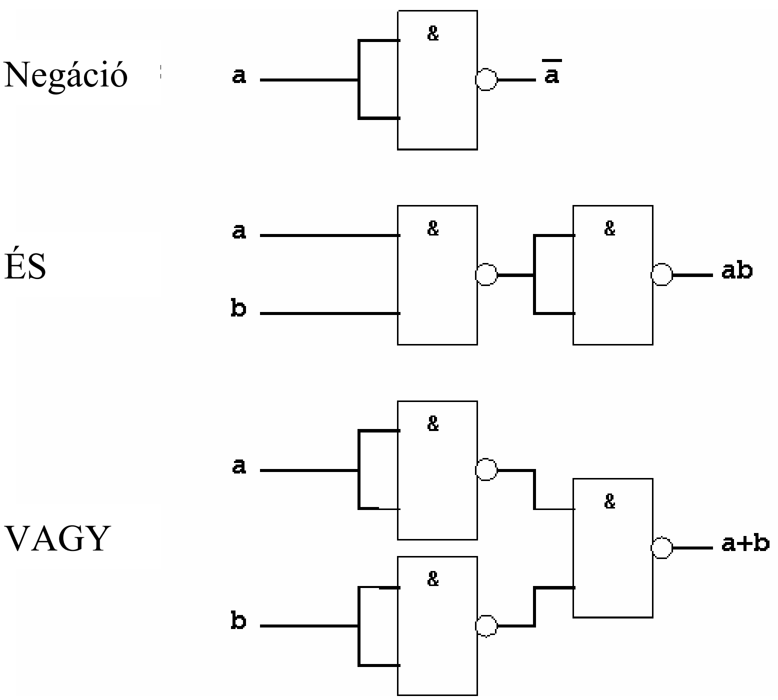
b) NOR (Not-OR)

Mindkettő önállóan teljes rendszert alkot, ami azt jelenti, hogy az összes logikai függvény megvalósítható kizárólag ezen alapművelet segítségével, annak kombinációiból.

Boole-algebra
Dualitás, NAND és NOR rendszerek

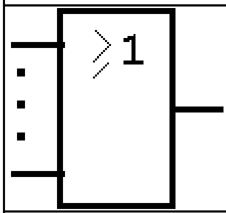
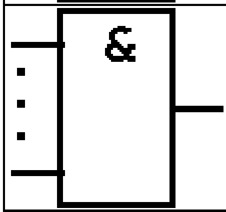
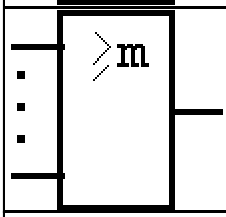
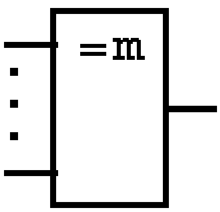
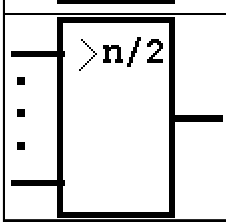
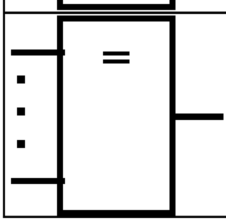
A három Bool-alapfüggvény felépítése pl. NAND kapukból:

Negáció:	$f(a) = \overline{a} = \overline{(a \wedge a)}$
ÉS:	$f(a,b) = a \wedge b = (a \wedge b) \vee (a \wedge b) = \overline{\overline{(a \wedge b) \vee (a \wedge b)}}$ $f(a,b) = \overline{\overline{(a \wedge b)} \wedge \overline{(a \wedge b)}}$
VAGY:	$f(a,b) = a \vee b = \overline{\overline{a \vee b}} = \overline{\overline{a} \wedge \overline{b}}$ $f(a,b) = \overline{\overline{(a \wedge a)} \wedge \overline{(b \wedge b)}}$

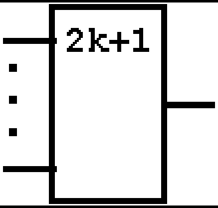
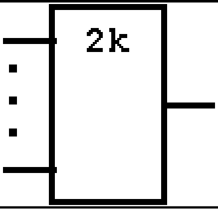
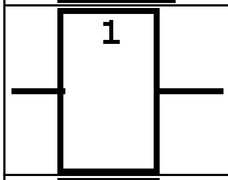
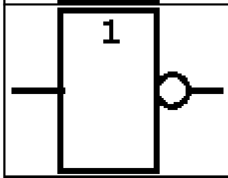
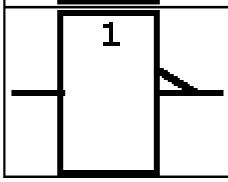


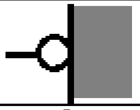
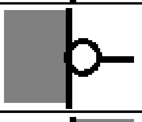
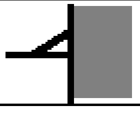
Mivel a három alapfüggvényen kívül a két konstans elem , a „0” és „1” is realizálható NAND kapuáramkörrel, ezért a NAND művelettel az összes logikai probléma megoldható.
Ugyanez érvényes a NOR műveletre és gyakorlati megvalósításaira is.

Néhány fontosabb, DIN 40900 szabvány szerinti kapcsolási jel

Rajzjel	Leírás
	**VAGY kapu (OR)** A kimenet csak akkor veszi fel az "1" értéket, ha egy vagy több bemenet "1" állapotban van.
	**ÉS kapu (AND)** A kimenet csak akkor van az "1" állapotban, ha mindegyik bemenet "1" állapotban van.
	**Logikai küszöbérték-kapu (*logic threshold element*)** A kimenet csak akkor veszi fel az "1" értéket, ha az "1"-es állapotban lévő bemenetek száma egyenlő vagy nagyobb mint m. **Megjegyzés 1:** m-nek mindig kisebbnek kell lenni, mint a bemenetek száma **Megjegyzés 2:** Ha m=1, akkor az egy VAGY kapu.
	**n-ből m kapu (*m and only m element*)** A kimenet csak akkor "1"-es, ha, az "1"-es állapotban lévő bemenetek száma megegyezik m-el (a kapcsolási jel-be beírt számmal). **Megjegyzés 1:** m-nek mindig kisebbnek kell lenni, mint a bemenetek száma **Megjegyzés 2:** Az a kapu, melynek két bemenete van és m=1, az egy kizáró VAGY kapu.
	**Majoritás(többség)-kapu (*majority element*)** A kimenet csak akkor "1"-es, ha a bemenetek több mint a fele "1"-es.
	**Ekvivalencia (azonosság)-kapu (*logic identity element*)** A kimenet csak akkor "1"-es, ha az összes bemenet ugyanabban az állapotban van.

Boole-algebra
Kapcsolási rajzjelek (folytatás)

Rajzjel	Leírás
	**PÁRATLAN-kapu, Összeadás-modulo-2-kapu** **(*ODD element, ODD-parity element, addition modulo 2 element*)** A kimenet csak akkor veszi fel az "1"-es állapotot, ha a bemenetek közül páratlan számú (1,3,5, stb.) van "1" állapotban.
	**PÁROS-kapu, (EVEN *element, EVEN-parity element*)** A kimenet csak akkor veszi fel az "1"-es állapotot, ha a bemenetek közül páros számú (0,2,4, stb.) van "1" állapotban.
	**Buffer, a kimeneten nincs erősítés** **(buffer without specially amplified output)** A kimenet csak akkor veszi fel az "1"-es állapotot, ha a bemenet az "1"-es állapotban van.
	**Negáló kapu, Inverter (az egységes, egyeztetett logikai szimbólumrendszer szerinti kapcsolásokban) (*Negator, Inverter*)** A kimenet csak akkor veszi fel a "0" állapotot, ha a bemenet az "1"-es állapotban van.
	**Inverter (*Inverter*)** **(azokban a kapcsolásokban, ahol a logikai polaritás jelölése használatos)** A kimenet csak akkor veszi fel az "L" szintet, ha a bemenet "H" szinten van.

	Negálás, a bemenetre vonatkozóan (*logic negation, shown at an input*)
	Negálás, a kimenetre vonatkozóan (*logic negation, shown at an output*)
	Logikai polaritás a bemeneten (*logic polarity, polarity indicator shown at an input*)
	Logikai polaritás a kimeneten (*logic polarity, polarity indicator shown at an output*)
	Logikai polaritás a bemeneten, jelhaladási irány jobbról balra
	Logikai polaritás a kimeneten, jelhaladási irány jobbról balra