Algebra di Boole

Il nome viene da George Boole, un matematico che definì delle strutture matematiche, dette appunto algebre di Boole

… è detta anche logica delle proposizioni ….

…. una frase può assumere solo uno dei due valori di *verità*:

… o è VERA o è FALSA

valori booleani (operandi)

solo due valori: **vero** (**1**) e **falso** (**0**)

Algebra di Boole

Operatori ed Operazioni

… connettivi logici

… permettono di formare frasi più complesse e valutarne la *verità*

operatori logici

not, ¬, $\overline{}$ (unario) - Negazione logica

and, ∧, • (binario) – Prodotto logico

or, ∨, + (binario) – Somma logica

Negazione (not) - Tavola di verità

Data una generica frase *x* : l'operatore di negazione ***not*** ne inverte il valore

x	not x
0	1
1	0

Esempi di not

x: "piove"

not x: "non piove"

Possibili notazioni:

not x oppure

¬x oppure

$\overline{\mathbf{x}}$

Somma (OR) - Tavola di verità

Data due (o più) generiche frasi *x* e *y* la frase risultante dalla loro somma logica è vera se almeno una è vera

x	y	x or y
0	0	0
0	1	1
1	0	1
1	1	1

Possibili notazioni: :

x or y oppure

$x \vee y$ oppure

$x + y$

Esempi di or

x: "Benevento è capoluogo di provincia" **VERO**
y: "il Napoli gioca in serie B" **FALSO**
z: "Benevento è capitale d'Italia" **FALSO**

x or y:
"Benevento è capoluogo di provincia oppure il NA gioca in serie B"
VERO

y or z:
"Il Napoli gioca in serie B oppure Benevento è capitale d'Italia"
FALSO

Prodotto (AND) - Tavola di verità

Data due (o più) generiche frasi *x* e *y* , la frase risultante dal loro prodotto logico è vera se tutte sono vere

x	y	x and y
0	0	0
0	1	0
1	0	0
1	1	1

Possibili notazioni: :

x and y oppure

x $\wedge$ **y** oppure

x • y

Esempi di and

x: "Benevento è capoluogo di provincia" **VERO**
y: "il Napoli gioca in serie B" **FALSO**

x and y:

"Benevento è capoluogo di provincia <u>e</u> il Napoli gioca in serie B"
FALSO

x: "la terra è un pianeta" **VERO**
y: "la luna è un satellite della terra" **VERO**

x and y:
"la terra è un pianeta <u>e</u> la luna è un satellite della terra"
VERO

Precedenza degli operatori logici

Operatore unario: **not**

Operatore binario: **and**

Operatore binario: **or**

Esempio

Supponiamo di avere tre informazioni: a, b, c

Sia:

a **VERA**

b **FALSA**

c **FALSA**

quanto vale l'espressione:

a or b and c ?

a or (b and c) ***VERA***

Tabella di verità di: a or (b and c)

a	b	c	b and c	a or (b and c)
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	0	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Teoremi fondamentali

• = and
+ = or
¯= not

Identità:	**1•x=x**	**0+x=x**
Nullo:	**0•x=0**	**1+x=1**
Idempotenza:	**x•x=x**	**x+x=x**
Inverso:	**(not x)•x=0**	**(not x)+x=1**
Commutativa:	**x•y=y•x**	**x+y=y+x**
Associativa:	**(x•y)•z=x•(y•z)**	**(x+y)+z=x+(y+z)**
Distributiva:	**x•(y+z)=x•y+x•z**	**x+(y •z)=(x+y)•(x+z)**

Principio di dualità:
derivabili l'una
dall'altra scambiando:
+ con **•**
0 con **1**

Valgono le proprietà distributive

x and (y or z) = (x and y) or (x and z)

“quella penna mi piace" e ("ho i soldi" oppure "la carta di credito")

=

("mi piace" e "ho i soldi") oppure
("mi piace" e "ho la carta di credito")

Teorema di De Morgan

$$\overline{(x \bullet y)} = \bar{x} + \bar{y}$$

$$\overline{(x + y)} = \bar{x} \bullet \bar{y}$$

Esempio:

non (ho i soldi o ho la carta di credito)

=

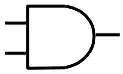
non ho i soldi e non ho la carta di credito

x	y	$\overline{(x+y)}$	$\bar{x} \bullet \bar{y}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

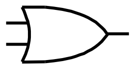
Rappresentazione circuitale delle operazioni logiche

Porte logiche

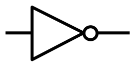
and



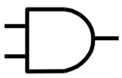
or



not



and realizzato con interruttori

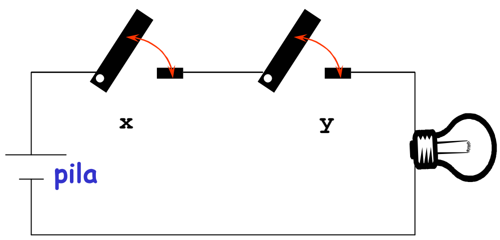


and

x	y	x and y
0	0	0
1	0	0
0	1	0
1	1	1

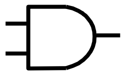
{aperto = 0, chiuso = 1}

Contatti in serie



x	y	x and y
aperto	aperto	aperto
chiuso	aperto	aperto
aperto	chiuso	aperto
chiuso	chiuso	chiuso

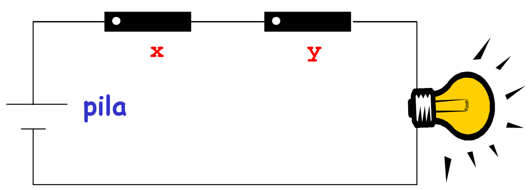
and realizzato con interruttori



x	y	x and y
0	0	0
1	0	0
0	1	0
1	1	1

{aperto = 0, chiuso = 1}

Contatti in serie

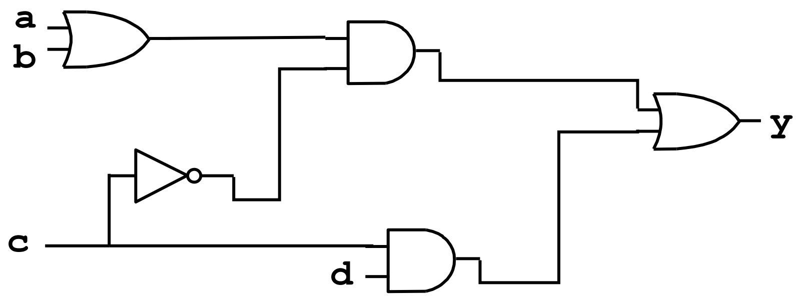


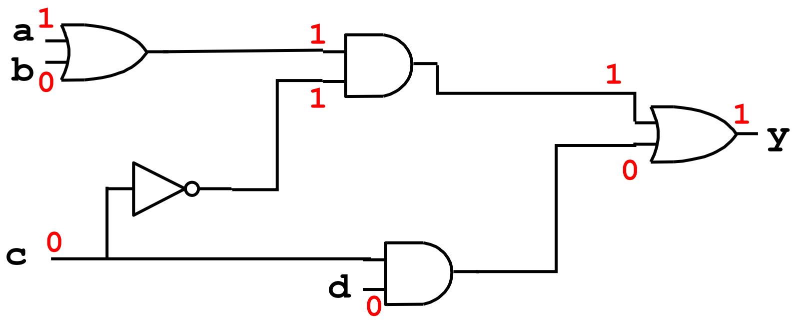
x	y	x and y
aperto	aperto	aperto
chiuso	aperto	aperto
aperto	chiuso	aperto
chiuso	chiuso	chiuso

Esempio

y = ((a or b) and (not c)) or (c and d)

circuito





y = ((a or b) and (not c)) or (c and d)

a	b	c	d	a or b	not c	c and d	(a or b) and (not c)	y
0	0	0	0	0	1	0	0	0
0	0	0	1	0	1	0	0	0
0	0	1	0	0	0	0	0	0
0	0	1	1	0	0	1	0	1
0	1	0	0	1	1	0	1	1
0	1	0	1	1	1	0	1	1
0	1	1	0	1	0	0	0	0
0	1	1	1	1	0	1	0	1

`y = ((a or b) and (not c)) or (c and d)`

a	b	c	d	a or b	not c	c and d	(a or b) and (not c)	y
1	0	0	0	1	1	0	1	1
1	0	0	1	1	1	0	1	1
1	0	1	0	1	0	0	0	0
1	0	1	1	1	0	1	0	1
1	1	0	0	1	1	0	1	1
1	1	0	1	1	1	0	1	1
1	1	1	0	1	0	0	0	0
1	1	1	1	1	0	1	0	1

Semplificazione

Il vantaggio dell'algebra di Boole sta nel fatto che permette la semplificazione dei circuiti

Esempio

$$F = \bar{x} \bullet y \bullet z + \bar{x} \bullet y \bullet \bar{z} + x \bullet z \qquad \textit{(distributiva)}$$

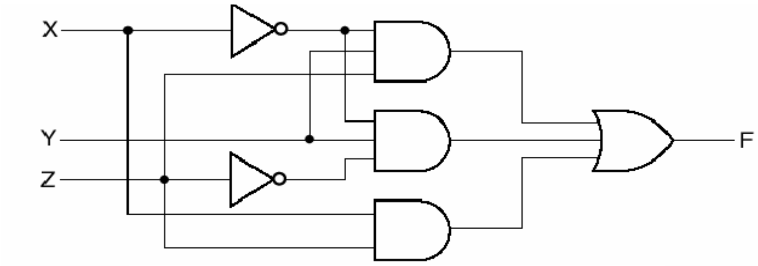
$$= \bar{x} \bullet y \bullet (z + \bar{z}) + x \bullet z \qquad \textit{(inverso)}$$

$$= \bar{x} \bullet y \bullet 1 + x \bullet z \qquad \textit{(identità)}$$

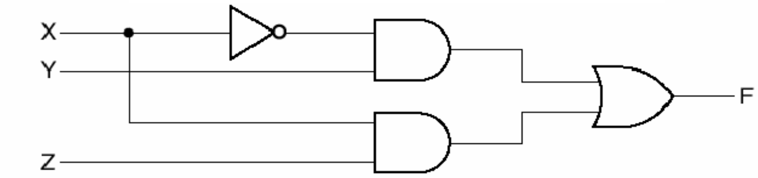
$$= \bar{x} \bullet y + x \bullet z$$

Le due funzioni sono equivalenti:

hanno la stessa tabella di verità ma la seconda funzione è realizzabile con un circuito più semplice



$$F = \bar{x} \bullet y \bullet z + \bar{x} \bullet y \bullet \bar{z} + x \bullet z$$



$$F = \bar{x} \bullet y + x \bullet z$$

Esercizio

Scrivere un'espressione booleana che determina se:
il valore di una variabile **i** è nell'intervallo da **1** a **100**, estremi inclusi

```
1. (i>=1)
2. (i<=100)
      (i>=1) and (i<=100)
```

Esercizio

Scrivere un'espressione booleana che determina se:
il valore di una delle due variabili intere **j** e **k** è multiplo dell'altro

```
(j % k = 0)
(k % j = 0)

(j % k = 0) or (k % j = 0)
```

Corretto??

… imporre con **j** e **k** diversi da **0**

```
(j ≠ 0) and (k ≠ 0)
```

Esercizio

Valutare e disegnare il circuito

F = (not a) and b

G = (not a) or b

Soluzione: F

a	b	not a	F = (not a) and b
0	0	1	0
0	1	1	1
1	0	0	0
1	1	0	0