L’algebra booleana

L'algebra booleana è un ramo della matematica che si occupa di variabili che assumono due valori: vero o falso. È utilizzata principalmente per:

- la progettazione di circuiti digitali, microprocessori e memorie nei computer.
- la gestione delle espressioni logiche nei linguaggi di programmazione.
- la teoria degli automi, le macchine di Turing e la logica formale.

Nell’algebra di Bool abbiamo i possibili valori sui quali operare sono rappresentai in maniera equivalente da:

- **FALSO** ed i suoi sinonimi **0**, **F**, **false**
- **VERO** ed i suoi sinonimi **1**, **V**, **true**, **T**

L’algebra di Boole si basa su tre operatori base:

- NOT (negazione logica)
- AND (prodotto logico)
- OR (somma logica)

Gli operatori seguono le regole riportate nelle seguenti tabelle:

X	NOT X
0	1
1	0

X	Y	X AND Y
0	0	0
0	1	0
1	0	0
1	1	1

X	Y	X OR Y
0	0	0
0	1	1
1	0	1
1	1	1

Esistono altre funzioni booleane che possono essere espresse come una combinazione di questi tre operatori. Tramite le proprietà dell’algebra booleana è possibile semplificare espressione booleane complesse. Gli operatori booleani vengono indicati in diversi modi a seconda del campo di applicazione.

Oltre agli operatori già indicati esistono i seguenti operatori:

- XOR
- NAND
- NOR

Che seguono le regole riportate nelle seguenti tabelle:

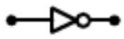
X	Y	X XOR Y
0	0	0
0	1	1
1	0	1
1	1	0

X	Y	X NAND Y
0	0	1
0	1	1
1	0	1
1	1	0

X	Y	X NOR Y
0	0	1
0	1	0
1	0	0
1	1	0

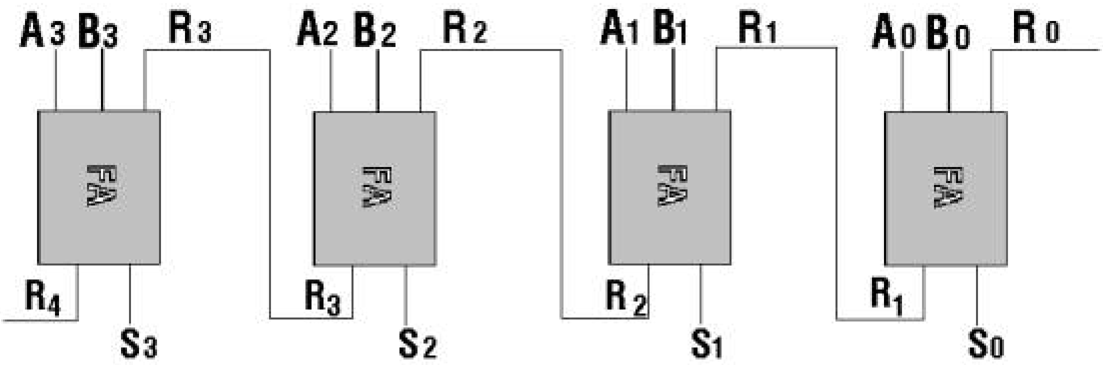
Vi sono diversi modi di indicare gli operatori a seconda delle diverse applicazioni dell’algebra booleana.

Vediamo quali sono alcuni di questi:

<table>
<tr><th></th><th>NOT</th><th>AND</th><th>OR</th></tr>
<tr><td>Elettronica</td><td></td><td></td><td></td></tr>
<tr><td>Logica elementare</td><td>Z=~X
Z=-X</td><td>$Z=X\wedge Y$
$Z=X\cap Y$</td><td>$Z=X\vee Y$
$Z=X\cup Y$</td></tr>
<tr><td>Espressione</td><td>$Z=\bar{X}$</td><td>$Z=X\bullet Y$</td><td>$Z=X+Y$</td></tr>
<tr><td>C++, Java</td><td>!</td><td>&&</td><td>||</td></tr>
</table>

Come possiamo vedere l'algebra booleana si applica a svariati usi in diversi campi del sapere. In particolare nel 1938 Shannon ha dimostrato come l'algebra booleana potesse essere presa a fondamento per la progettazione di circuiti logici digitali.

Vediamo come esempio il circuito logico digitale creato per realizzare la somma di numeri binari. Il circuito è stato pensato come formato da più blocchi ciascuno dei quali effettua la somma di una singola cifra dei numeri binari avendo come ingresso le cifre dei numeri ed il riporto e come uscita la somma ed il riporto per le cifre successive come si vede nella seguente figura:



Nella figura vengono sommati due numeri binari A, B di quattro cifre le cifre sono rispettivamente:

- primo numero A (A_3, A_2, A_1, A_0)
- secondo numero B (B_3, B_2, B_1, B_0)
- somma delle cifre S (S_3, S_2, S_1, S_0)

R_0 è il riporto iniziale che varrà, R_1, R_2, R_3, R_4 sono i riporti delle somme fatte sulle singole cifre per cui la somma sarà data dal numero che ha come cifre:

R_4, S_3, S_2, S_1, S_0

Eesaminiamo il circuito logico digitale creato per realizzare il singolo blocco indicato nella figura con la sigla FA.

Scriviamo la tabella di verità con ingressi (input) ed uscite (output) del il singolo blocco.

X_0	Y_0	R_0	S_0	R_1
0	0	0	0	0
0	1	0	1	0
1	0	0	1	0
1	1	0	0	1
0	0	1	1	0
0	1	1	0	1
1	0	1	0	1
1	1	1	1	1

Scriviamo le espressioni logiche che servono per ottenere S_0:

$S_0 = \overline{X_0} \bullet Y_0 \bullet \overline{R_0} + X_0 \bullet \overline{Y_0} \bullet \overline{R_0} + \overline{X_0} \bullet \overline{Y_0} \bullet R_0 + X_0 \bullet Y_0 \bullet R_0$

$R_1 = X_0 \bullet Y_0 \bullet \overline{R_0} + X_0 \bullet \overline{Y_0} \bullet R_0 + \overline{X_0} \bullet Y_0 \bullet R_0 + X_0 \bullet Y_0 \bullet R_0$

Vi sono inoltre alcune proprietà fondamentali che ci consentono di semplificare espressioni booleane complesse in espressioni più semplici.

Esiste inoltre uno strumento grafico utilizzato per semplificare le espressioni booleane "Le mappe di Karnaugh". Questo strumento viene utilizzato per circuiti con numero di ingressi (input) limitato a non più di quattro o cinque ingressi. Le mappe di Karnaugh per 2, 3, 4 input sono:

	Y_0	0	1
X_0			
0		F_{00}	F_{01}
1		F_{10}	F_{11}

		V_0	0	1
X_0	Y_0			
0	0		F_{000}	F_{001}
0	1		F_{010}	F_{011}
1	1		F_{110}	F_{111}
1	0		F_{100}	F_{101}

	M_0N_0	00	01	11	10
X_0Y_0					
00		F_{0000}	F_{0001}	F_{0011}	F_{0010}
01		F_{0100}	F_{0101}	F_{0111}	F_{0110}
11		F_{1100}	F_{1101}	F_{1111}	F_{1110}
10		F_{1000}	F_{1001}	F_{1011}	F_{1010}

Queste mappe si utilizzano per ottenere velocemente le espressioni logiche vediamo il caso del sommatore:

S_0		R_0	0	1
X_0	Y_0			
0	0		0	1
0	1		1	0
1	1		0	1
1	0		1	0

R_1	R_0	0	1
X_0Y_0			
00		0	0
01		0	1
11		1	1
10		0	1

Tutte le celle individuate da 1 sono in AND tra loro inoltre le celle adiacenti sono indipendenti da una delle variabili la funzione si ottiene mettendo in OR tutte le espressioni così ricavate.

Per S_0 ed R_{01} avremo:

$S_0 = \overline{X_0} \bullet Y_0 \bullet \overline{R_0} + X_0 \bullet \overline{Y_0} \bullet \overline{R_0} + \overline{X_0} \bullet \overline{Y_0} \bullet R_0 + X_0 \bullet Y_0 \bullet R_0$

$R_1 = X_0 \bullet Y_0 + X_0 \bullet R_0 + Y_0 \bullet R_0$

mentre per S_0 non cambia molto vediamo come R_1 si è molto semplificata.

Per semplificare le espressioni booleani si utilizzano le proprietà fondamentali dell'algebra booleana che sono:

- Esistenza del complemento:
  - $x + \overline{x} = 1$
  - $x \bullet \overline{x} = 0$
- Idempotenza:
  - $x + x = x$
  - $x \bullet x = x$
- Elemento nullo (forcing function):
  - $x + 1 = 1$
  - $x \bullet 0 = 0$
- Elemento neutro:
  - $x + 0 = x$
  - $x \bullet 1 = x$
- Proprietà Commutativa:
  - $x + y = y + x$
  - $x \bullet y = y \bullet x$
- Proprietà Associativa:
  - $x + (y + z) = (x + y) + z = x + y + z$
  - $x \bullet (y \bullet z) = (x \bullet y) \bullet z = x \bullet y \bullet z$
- Proprietà Distributiva:
  - $(x \bullet y) + (x \bullet z) = x \bullet (y + z)$
  - $(x + y) \bullet (x + z) = x + (y \bullet z)$
- Proprietà Assorbimento:
  - $x + (x \bullet y) = x$
  - $x \bullet (x + y) = x$

Oltre a queste proprietà molto importanti sono le formule di De Morgan:

1) $\overline{x \bullet y} = \overline{x} + \overline{y}$

2) $\overline{x + y} = \overline{x} \bullet \overline{y}$