Informatica per l’Ingegneria

Corsi M – N

A.A. 2023/2024

Angelo Cardellicchio

04 – Algebra booleana

Outline

- Circuiti elettronici digitali
- Algebra di Boole
- L'operazione *AND*
- L'operazione *OR*
- L'operazione *NOT*
- *AND* e *NOT* con variabili binarie
- Altre proprietà
- Funzioni logiche e tabelle di verità
- Leggi notevoli

Circuiti elettronici digitali

- I circuiti elettronici digitali sono costruiti con elementi caratterizzati da ***due soli possibili stati di funzionamento***.
- I dispositivi elettronici a due stati di funzionamento sono giustificati da:
  - *facilità di realizzazione;*
  - *possibilità di rappresentare le informazioni come successioni di* 1 *e di* 0.
- I segnali binari, corrispondenti ai due livelli di funzionamento degli elementi costitutivi del calcolatori, sono trattati dall'***algebra di Boole***.

Algebra di Boole (1)

- L'algebra di Boole venne introdotta da George Boole nel XIX secolo per analizzare algebricamente problemi di calcolo proposizionale.
  - Fondata su un insieme di teoremi e regole che governano le operazioni logiche, consentendone la rappresentazione matematica.
  - Fa da base per lo sviluppo dell'elettronica digitale.
- Questo tipo di algebra contempla due unici stati: 0, equivalente a falso, ed 1, equivalente a vero.
  - I due stati sono ***mutualmente esclusivi***.
  - Possono descrivere lo stato di apertura o chiusura di un generico contatto.
- Sui valori booleani si definiscono le operazioni AND, OR, NOT.

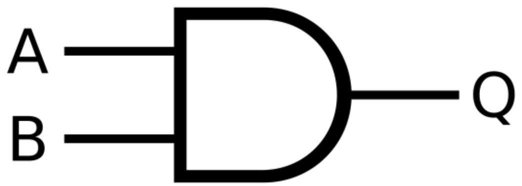
Algebra di Boole (2)

- Le operazioni AND ed OR sono operazioni binarie, mentre la NOT è unaria.
- Nella valutazione delle espressioni booleane vi è una relazione di precedenza tra gli operatori NOT, AND ed OR, nell'ordine in cui sono stati elencati.
- Gli operatori possono essere rappresentati in vari modi, anche usando operatori matematici o porte logiche.
- Il comportamento di ciascuno di questi operatori è definito usando una **tabella delle verità**.

L'operazione AND

- L'operazione AND è definita come ***prodotto logico***.
- In particolare, il valore del prodotto logico è il simbolo 1 se e solo se il valore dei due operandi è pari ad 1.

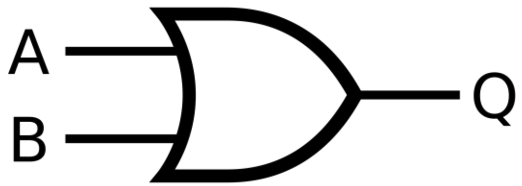
A	B	$\times$
0	0	0
0	1	0
1	0	0
1	1	1



L’operazione OR

- L’operazione OR è definita come ***somma logica***.
- In particolare, il valore della somma logica è il simbolo 1 se e solo se almeno uno dei due operandi ha valore 1.

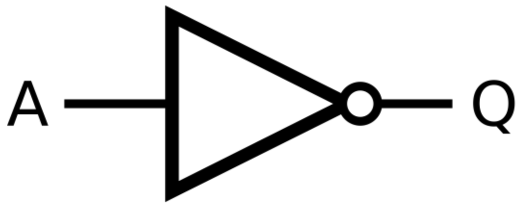
A	B	$+$
0	0	0
0	1	1
1	0	1
1	1	1



L’operazione NOT

- L’operazione NOT è definita come ***negazione logica***.
- In particolare, l’operatore inverte il valore della costante su cui opera.

A	$\bar{A}$
0	1
1	0



- Per definizione, $\bar{\bar{A}} = A$.
  - *In altre parole, negando la negazione si ritorna allo stato originario.*

AND e NOT con variabili binarie

- Una variabile binaria indipendente può assumere un valore tra 0 ed 1.
- Date n variabili binarie indipendenti, la loro somma logica (OR) è:

$$x_1 + x_2 + \cdots + x_n = \begin{cases} 1, & sse \ \exists \ x_i = 1, i \in 1, \dots, n \\ 0, & sse \ \nexists \ x_i = 1, i \in 1, \dots, n \end{cases}$$

- Date n variabili binarie indipendenti, il loro prodotto logico (AND) è:

$$x_1 \times x_2 \times \cdots \times x_n = \begin{cases} 0, & sse \ \exists \ x_i = 0, i \in 1, \dots, n \\ 1, & sse \ x_1 = x_2 = \cdots = x_{n_i} \end{cases}$$

Altre proprietà (1)

- È facile verificare le seguenti identità:

$$
\begin{aligned}
x + 1 &= 1 \qquad & x \times 1 &= x \\
x + 0 &= x \qquad & x \times 0 &= 0 \\
x + x &= x \qquad & x \times x &= x
\end{aligned}
$$

- Le ultime due espressioni sono definite come **legge dell'idempotenza**.
- In particolare, 0 è l'elemento neutro per l'operazione di OR, mentre 1 è l'elemento neutro per l'operazione di AND.

Altre proprietà (2)

- Per gli operatori di AND ed OR valgono le seguenti proprietà:

Proprietà	OR	AND
Commutativa	$x_1 + x_2 = x_2 + x_1$	$x_1 \times x_2 = x_2 \times x_1$
Associativa	$x_1 + x_2 + x_3 = x_1 + (x_2 + x_3)$	$x_1 \times x_2 \times x_3 = x_1 \times (x_2 \times x_3)$
Distributiva del prodotto rispetto alla somma	$x_1 \times x_2 + x_1 \times x_3 = x_1 \times (x_2 + x_3)$	

- Per l'operatore NOT valgono le seguenti identità:

$$x + \bar{x} = 1$$

$$x \times \bar{x} = 0$$

$$\bar{\bar{x}} = x$$

Funzioni logiche e tabelle di verità (1)

- Date n variabili binarie indipendenti $x_1, x_2, \dots, x_n$, queste possono assumere 2^n configurazioni distinte.
- Una variabile y è funzione delle n variabili indipendenti $x_1, x_2, \dots, x_n$ se esiste un criterio che fa corrispondere in modo univoco ad ognuna delle 2^n configurazioni delle x_i un valore di y.

$$y = F(x_1, x_2, \dots, x_n)$$

- Una rappresentazione esplicita di una funzione è la ***tabella di verità***, in cui si elencano tutte le possibili combinazioni di $x_1, x_2, \dots, x_n$ con associato il valore di y.

Funzioni logiche e tabelle di verità (2)

- Facciamo un esempio.
- Immaginiamo tre variabili A, B, C associate alle seguenti condizioni:
  - A *è vera se è disponibile un buon posto per seguire la lezione in aula;*
  - B *è vera se il docente rende interessante e semplice seguire la lezione;*
  - C *è vera se lo studente è interessato alla materia*.
- Vogliamo caratterizzare una funzione F che assume valore vero quando *almeno due* delle condizioni modellate da A, B, C sono vere.

Funzioni logiche e tabelle di verità (2)

- Allora:

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Potremo scrivere F come *somma logica* delle combinazioni corrispondenti ad 1:

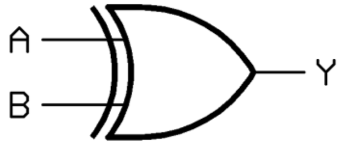
$$F(A,B,C) = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

Funzioni logiche e tabelle di verità (3)

- Un altro esempio è la funzione XOR.
- Questa funzione rappresenta un OR esclusivo. In altre parole, è pari ad 1 se e solo se una sola variabile è pari ad 1.

A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

$$XOR(A,B) = \bar{A}B + A\bar{B}$$



Leggi notevoli (1)

- La **legge dell'assorbimento** afferma che la somma di una variabile booleana x_1 con il prodotto tra la stessa ed un'altra variabile x_2 è uguale alla variabile x_1.

$$x_1 + x_1 \times x_2 = x_1$$

- Le **leggi di De Morgan** sono formulate nel seguente modo:

$$\overline{(x_1 + x_2)} = \overline{x_1} \times \overline{x_2}$$

$$\overline{(x_1 \times x_2)} = \overline{x_1} + \overline{x_2}$$

Leggi notevoli (2)

- Dalle leggi di De Morgan si evince che la scelta delle funzioni OR, AND e NOT come funzioni primitive è ridondante.
- In pratica, l’operazione logica AND può essere espressa in funzione delle operazioni OR e NOT; in modo analogo, l’operazione OR può essere espressa tramite AND e NOT.
- Le relazioni stabilite sono generalmente applicate nelle trasformazioni di funzioni booleane in altre equivalenti, ma di più facile realizzazione circuitale.

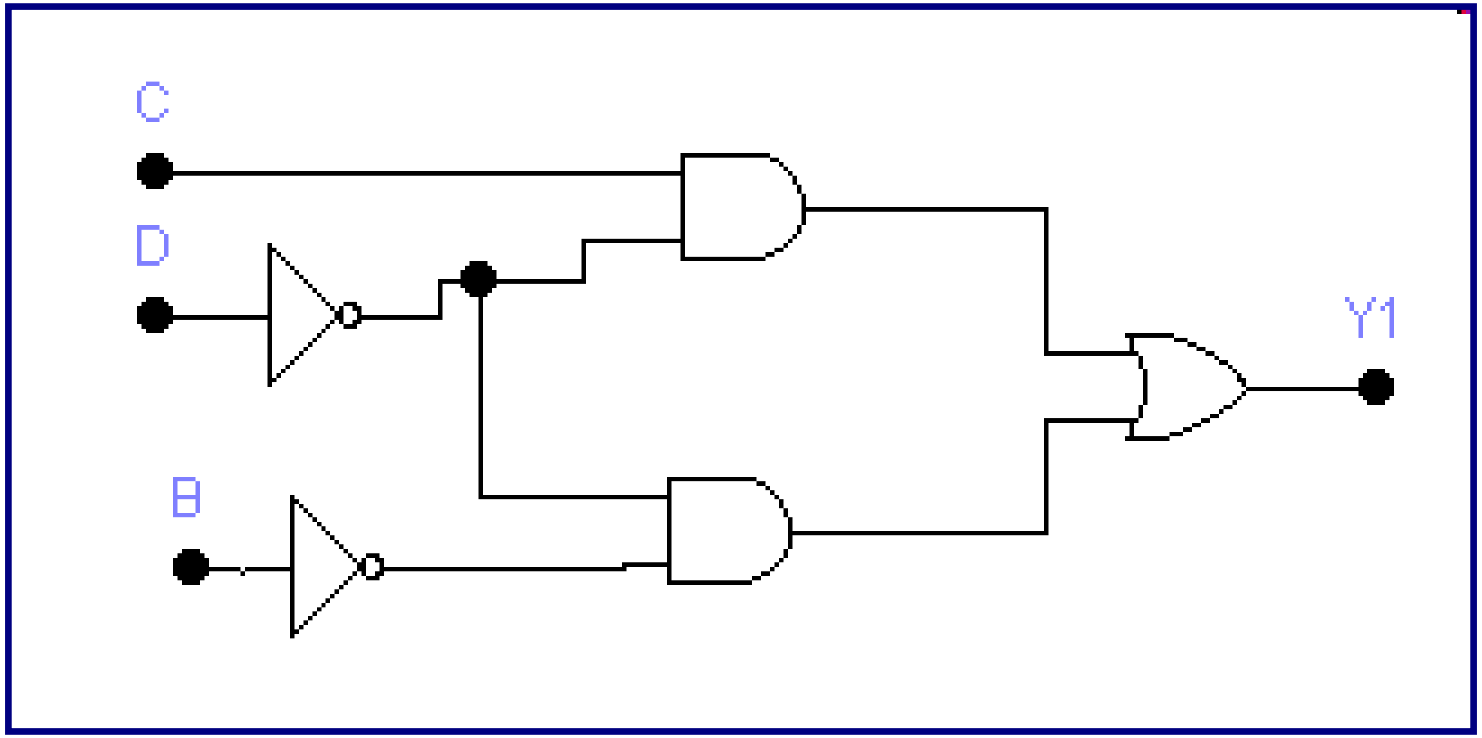
Dimostrazione del teorema di DeMorgan

- Dimostriamo le leggi di De Morgan usando le tabelle della verità.
- Siano $X = x_1, Y = x_2$. Allora:

X	Y	$X+Y$	$\overline{X+Y}$	$\bar{X}$	$\bar{Y}$	$\bar{X}\times\bar{Y}$
0	0	0	1	1	1	1
0	1	1	0	1	0	0
1	0	1	0	0	1	0
1	1	1	0	0	0	0

X	Y	$X\times Y$	$\overline{X\times Y}$	$\bar{X}$	$\bar{Y}$	$\bar{X}+\bar{Y}$
0	0	0	1	1	1	1
0	1	0	1	1	0	1
1	0	0	1	0	1	1
1	1	1	0	0	0	0

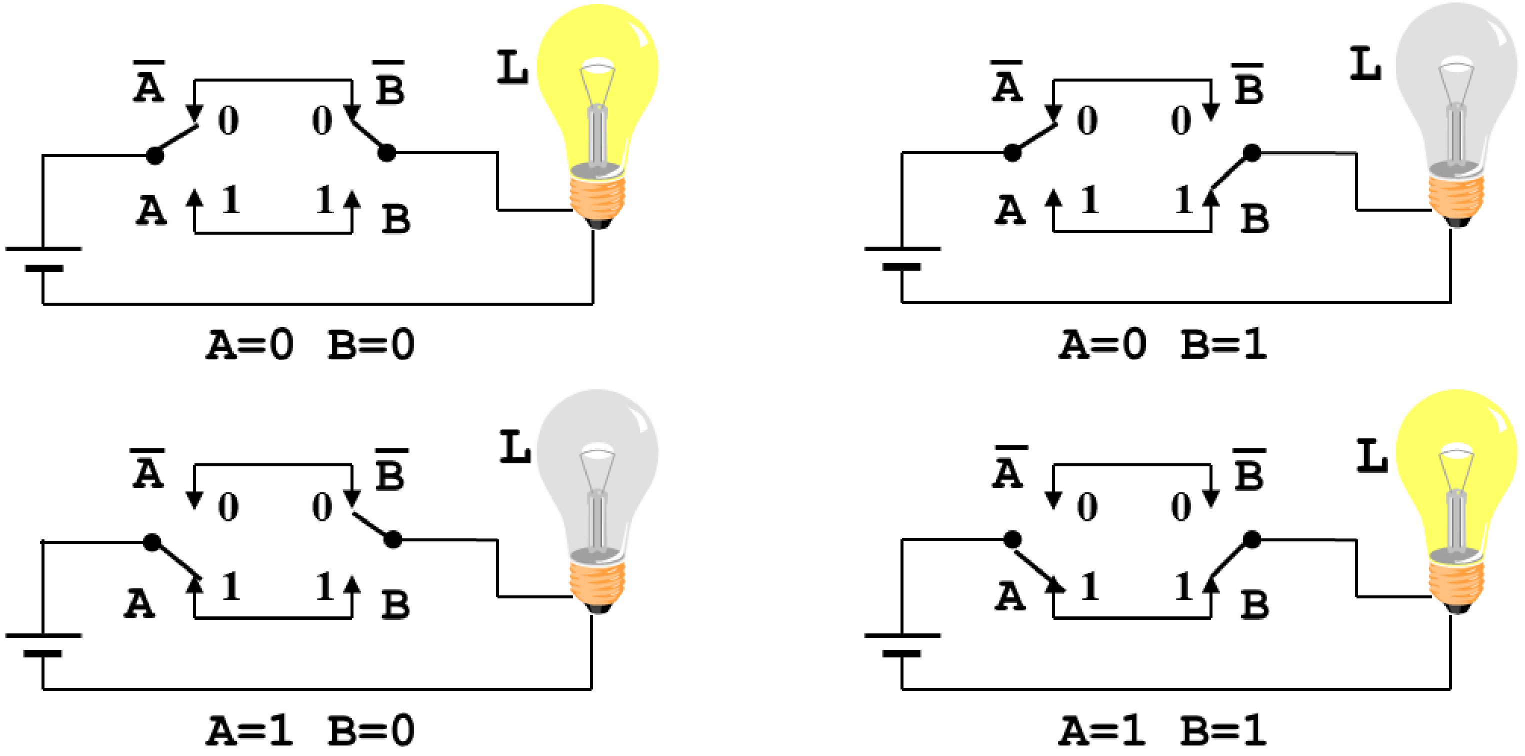
Un esempio di circuito logico



$$Y_1 = F(B, C, D) = (\bar{B} + C)\bar{D}$$

Un circuito con due interruttori

- I due interruttori corrispondono a due variabili (A, B) a valori booleani.
- Le variabili assumono i due valori 0 ed 1 che corrispondono alle due posizioni dell’interruttore.



A	B	L
0	0	1
0	1	0
1	0	0
1	1	1

$$L = \bar{A}\times\bar{B}+A\times B$$

Domande?

42