1.6 Introduzione alle proprietà e ai teoremi dell'algebra booleana

Come noto, i contatti di pulsanti, interruttori, ecc. possono assumere solo due posizioni, cioè di **aperto** (non passa corrente) e **chiuso** (passa corrente); negli schemi con contatti, definirne il funzionamento significa determinare la loro combinazione affinché venga fornita corrente all'utilizzatore (es. relè, lampada).

Il contatto, come si è visto, è un organo operativo dei circuiti elettrici, capace di lasciar passare o di interrompere la corrente; può essere comandato manualmente (es. interruttore o pulsante) oppure con un comando a distanza (es. relè, contattore, ecc.).

Anche negli impianti pneumatici o meccanici è possibile trovare organi che funzionano come i contatti elettrici (es. valvole pneumatiche ad azionamento elettrico o manuale, leve, ecc.); si ricorda infine che anche i dispositivi a semiconduttore come i transistor, gli SCR e i TRIAC funzionanti in regime di commutazione si comportano nel medesimo modo; le regole precedentemente citate e quelle che saranno esposte di seguito valgono quindi anche per queste apparecchiature e dispositivi.

In questo capitolo, per semplicità, i contatti vengono identificati con una lettera maiuscola (es. A), essa indica quale è l'operatore che lo comanda come ad esempio una bobina o un comando manuale (normalmente i contatti vengono identificati mediante la siglatura secondo le norme CEI).

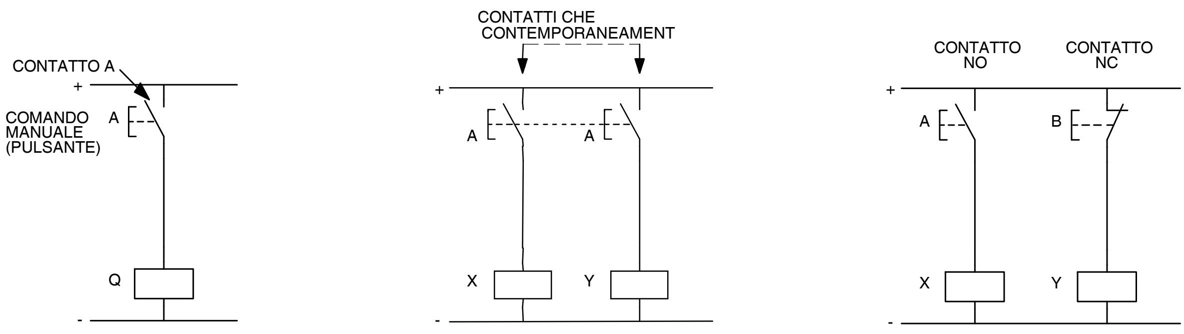
Ad un unico comando dell'operatore, tutti i contatti di uguale nome commutano contemporaneamente. I contatti a riposo possono essere normalmente aperti (NO) o normalmente chiusi (NC).

Quando si devono indicare, in un'equazione, i contatti NC, è convenzione indicarli con un trattino sopra la lettera; il contatto rappresentato nella fig. 1.39 viene indicato nelle equazioni booleane con $\overline{A}$ (si legge A negato), nella rappresentazione grafica non si segna la negazione sul simbolo, in quanto la forma grafica del contatto indica già il suo stato di riposo.

Si dicono negati o complementari i contatti NC, perché la funzione da loro risolta è esattamente "l'opposto" del contatto NO assunto come riferimento.

Secondo la matematica dei contatti, si indica convenzionalmente con "**livello logico 1**" il segnale (elettrico, pneumatico, meccanico) che fa cambiare di stato un contatto e con "**livello logico 0**" la mancanza di segnale oppure il segnale che non fa commutare il contatto.

Per esempio il segnale logico 1 si identifica con il movimento del dito che preme il pulsante B; per il circuito della bobina, come mostrato nella fig. 1.40, il segnale logico 1 è dato dal polo positivo dell'alimentazione, che gli viene trasmesso dalla chiusura del pulsante; l'eccitazione della bobina Q fa cambiare di stato i suoi contatti; il contatto NO, che a riposo trasferisce in X il livello 0, si chiude e trasmette il livello 1, mentre il contatto Q negato, che è NC e che a riposo trasmette in Y il livello 1, si apre e trasmette ora il livello 0.



a b c

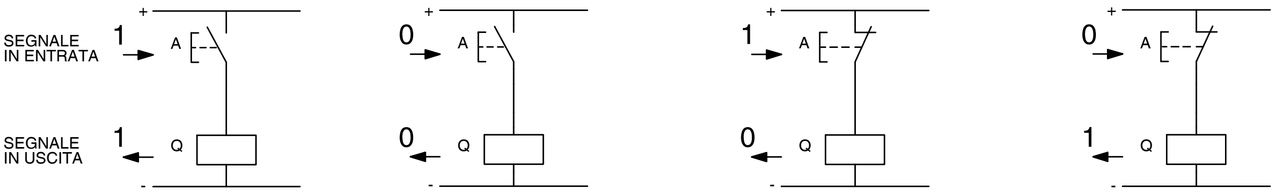
Fig. 1.39 - Esempi di funzionamento dei contatti a) Funzionamento di un contatto a comando manuale a pulsante - b) Funzionamento di contatti che commutano contemporaneamente - c) Esempio di contatto normalmente aperto e normalmente chiuso.

a b

***Fig. 1.40** - a) Esempio di contatto negato - b) Esempio di circuito con bobina e due contatti.*

In definitiva, come mostrato nella fig. 1.41, possiamo dire che:

1) applicare il segnale **1** ad un contatto **NO** vuol dire chiuderlo e determinare un segnale in uscita uguale a **1**, analogamente applicando il segnale **0** vuole dire determinare in uscita un segnale uguale a **0**;
2) applicare il segnale **1** ad un contatto **NC** vuol dire aprirlo e determinare un segnale in uscita uguale a **0**; viceversa applicando il segnale **0**, si determina in uscita il segnale **1**: questa operazione è detta di complementazione;
3) i simboli **0** e **1** non sono numeri in senso aritmetico, ma indicano lo stato logico dell'elemento operatore e/o dell'uscita.



Esempio con un pulsante con il contatto normalmente aperto (NO). Esempio con un pulsante con il contatto normalmente chiuso (NC).

***Fig. 1.41** - Esempi di segnali in entrata e in uscita in un circuito elettrico.*

Come mostrato in precedenza, si può utilizzare il segno di **moltiplicazione** per indicare l'operazione di AND, il segno di **addizione** per indicare una operazione di OR, una barra posta sopra alle lettere A, B, C, ecc., per indicare l'operazione di NOT.

Inoltre due o più contatti in serie eseguono un'operazione detta **prodotto logico** (AND), analogamente due o più contatti in parallelo eseguono una operazione detta **somma logica** (OR), infine l'operazione di **complementazione** od **inversione logica** (NOT) corrisponde alla trasformazione di contatti NO in contatti NC e viceversa.

Esistono nell'algebra di Boole delle proprietà che consentono di semplificare le espressioni booleane.

La proprietà commutativa (v. fig. 1.42) afferma che lo stato logico risultante è indipendente dall'ordine delle variabili sia che si effettui un'operazione logica di OR che una di AND.

La proprietà associativa (v. fig. 1.43) afferma che lo stato logico risultante è uguale a 0 solo se tutte le variabili sono a livello 0 nel caso di un'operazione logica di OR, mentre, nel caso di un'operazione logica di AND, lo stato logico risultante è uguale a 0 solo se almeno una variabile è a livello 0.

Nella fig. 1.44 viene mostrata la proprietà distributiva del prodotto rispetto alla somma e la stessa proprietà della somma rispetto al prodotto (duale rispetto alla precedente).

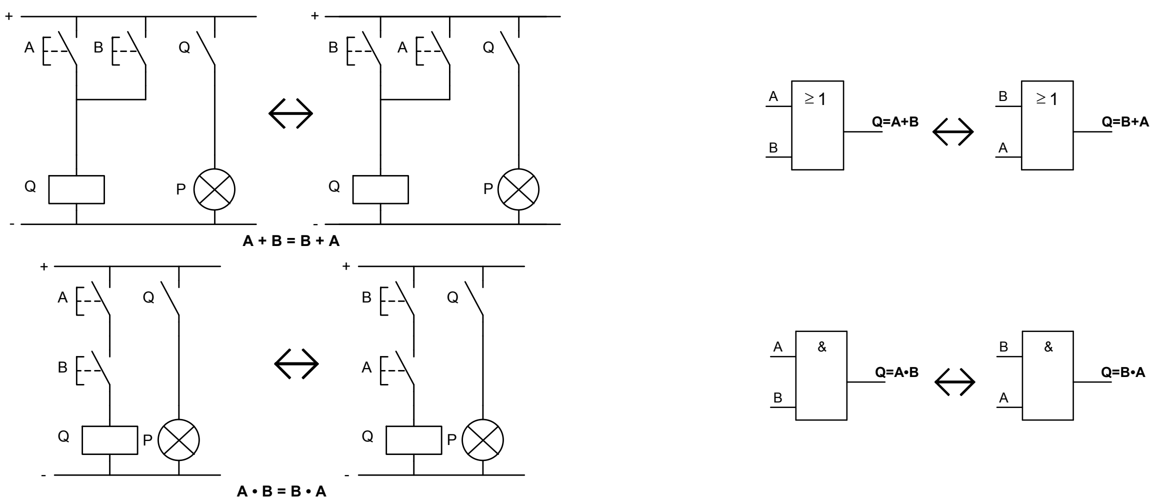


***Fig. 1.42** - Proprietà commutativa: Schemi elettrici a contatti e relativi circuiti logici. Simulazione LogicSim: 06)Proprietà_commutativa.*

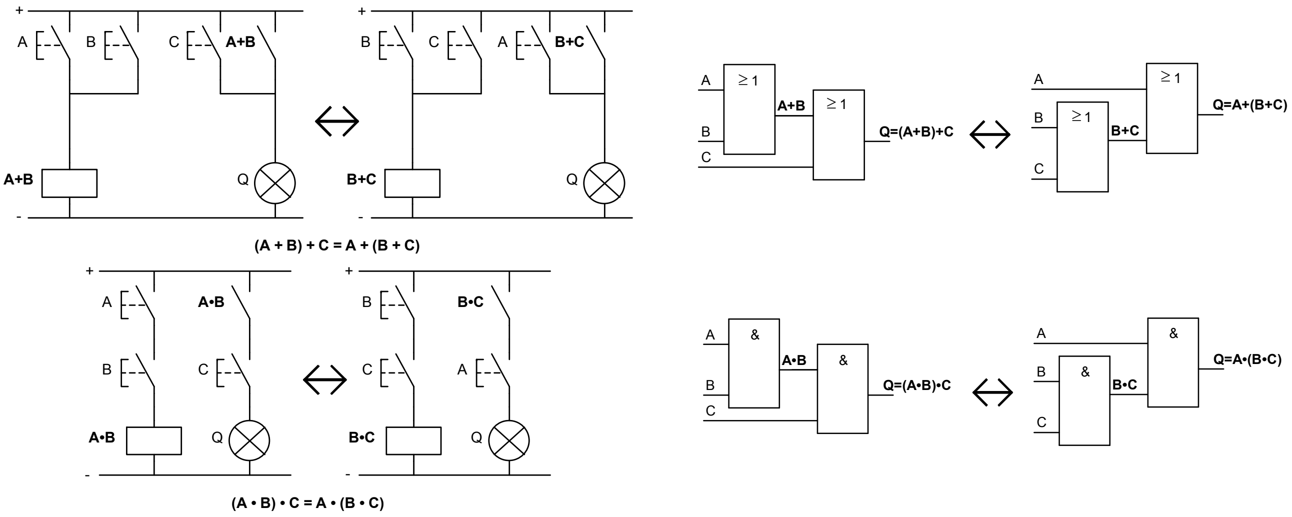


Fig. 1.43 *- Proprietà associativa: Schemi elettrici a contatti e relativi circuiti logici. Simulazione LogicSim: 07)Proprietà_associativa.*

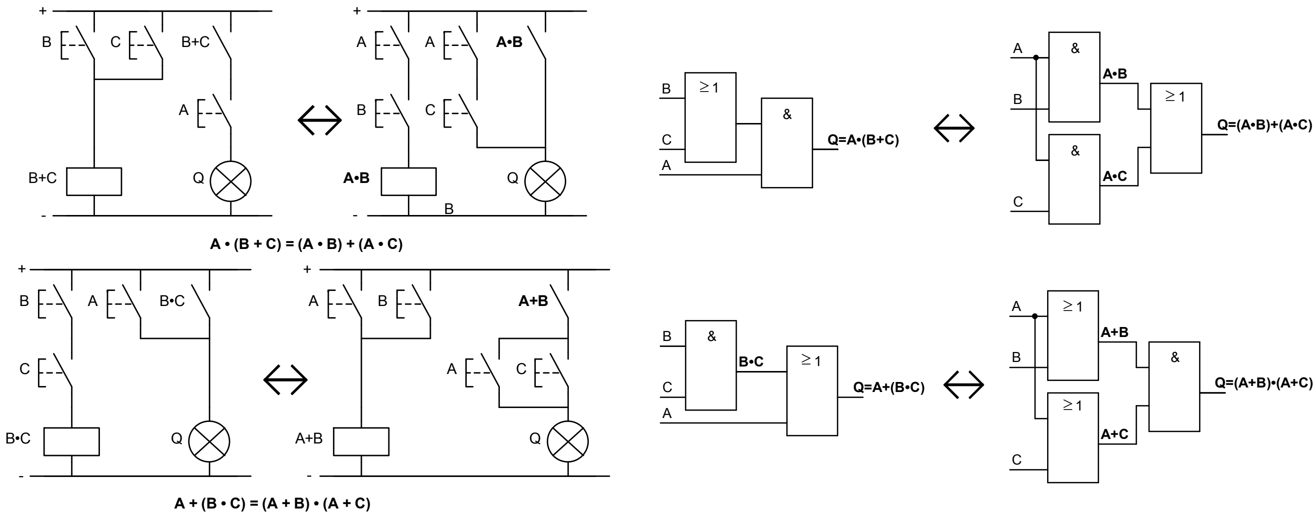
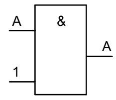
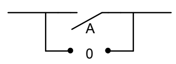
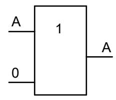
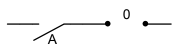
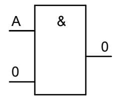
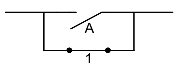
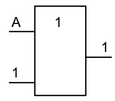
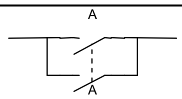
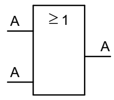
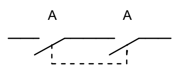
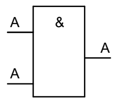
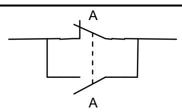
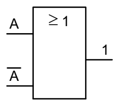
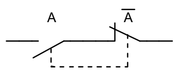
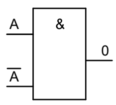
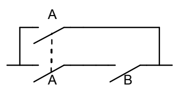
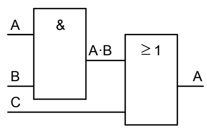
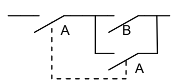
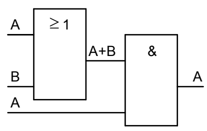


Fig. 1.44 *- Proprietà distributiva: Schemi elettrici a contatti e relativi circuiti logici. Simulazione LogicSim: 08)Proprietà_distributiva.*

Oltre le proprietà viste precedentemente, valgono i teoremi che vengono riportate in modo sintetico nella tab. 1.8, questi teoremi insieme alle proprietà viste precedentemente consentono di semplificare le espressioni booleane.

Teorema	Equazione	Schema elettrico	Schema logico
Teorema di identità	$A \cdot 1 = A$		
	$A + 0 = A$		
Teorema di annullamento	$A \cdot 0 = 0$		
	$A + 1 = 1$		
Teorema di idempotenza	$A + A = A$		
	$A \cdot A = A$		
Teorema dei complementi	$A + \overline{A} = 1$		
	$A \cdot \overline{A} = 0$		
Teorema dell'assorbimento	$A + A \cdot B = A$		
	$A \cdot (A + B) = A$		

Tab. 1.8 *- Teoremi dell'algebra booleana.*

Analizzando gli schemi elettromeccanici a contatti del tipo serie/parallelo, dopo aver determinato la loro equazione booleana, è interessante osservare che in molti casi è possibile scrivere delle espressioni e quindi degli schemi semplificati e quindi dal minor costo, che però hanno un funzionamento del tutto equivalente a quelli di partenza.

Analogo ragionamento vale se invece che dei contatti elettromeccanici vengono utilizzati, per realizzare una certa funzione logica, circuiti integrati digitali o un controllore logico programmabile (PLC) di cui verranno descritte le caratteristiche successivamente, nel primo caso si potrà minimizzare il numero dei componenti elettronici (minor costo dell'apparecchiatura), mentre nel secondo caso ciò che interessa è ridurre al minimo il numero di passi del programma che in altre parole vuol dire un programma di dimensioni più ridotte ed un conseguente minore uso della memoria.

Dovendo trasformare un'espressione booleana in circuito logico è necessario rispettare il seguente ordine:

Fase 1	Fase 2	Fase 3
NEGAZIONE (NOT)	***PRODOTTO (AND)***	***SOMMA (OR)***

Se nell'espressione vi sono parentesi o trattini di complementazione estesi su più variabili, si deve risolvere prima la parte di espressione tra parentesi o sottostante il trattino, seguendo comunque l'ordine precedentemente citato. Di seguito, nella fig. 1.45, fig. 1.46, fig. 1.47 vengono riportati alcuni esempi di trasformazioni da equazioni booleane ai corrispondenti schemi logici.

Esempio 1:

$$Q = A \cdot \overline{C} + \overline{A} \cdot B \cdot C$$

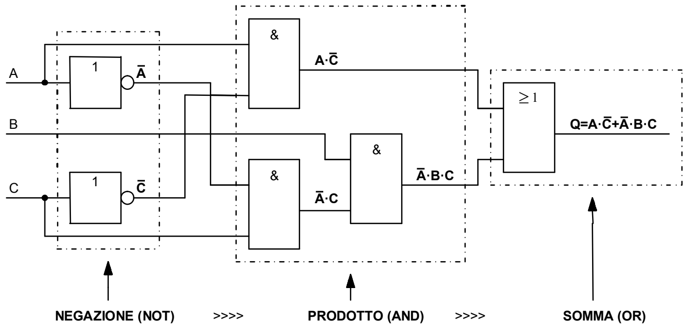


Fig. 1.45 *- Esempio di trasformazione di una espressione booleana in circuito logico.*

Esempio 2:

$$Q = \overline{A \cdot B} \cdot (C + \overline{D}) + E$$

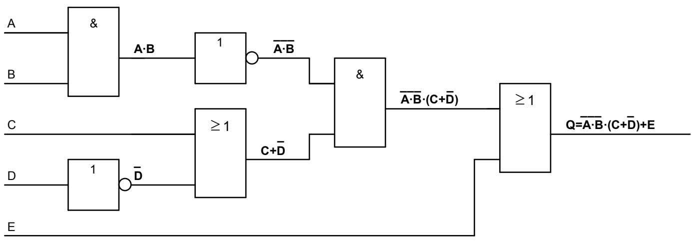


Fig. 1.46 *- Esempio di trasformazione di una espressione booleana in circuito logico.*

Esempio 3:

$$Q = A \cdot (\overline{C} \cdot \overline{D} + B \cdot D)$$

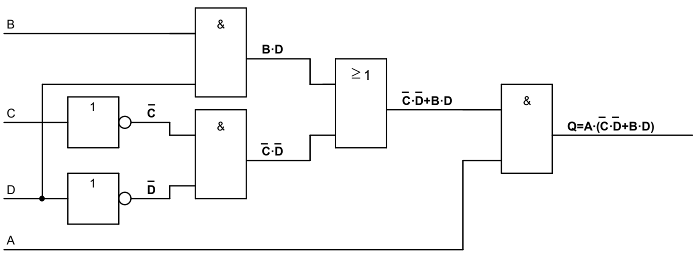
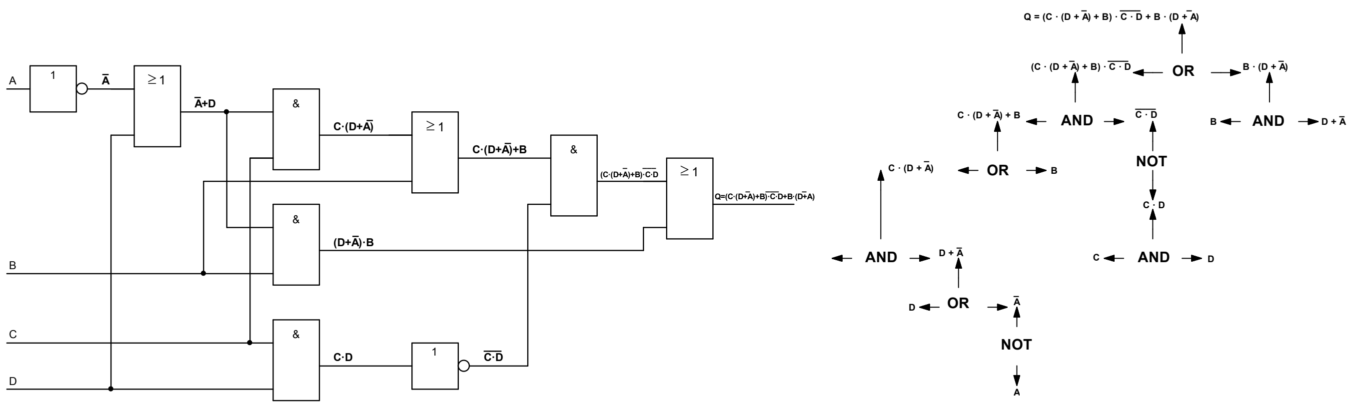


Fig. 1.47 *- Esempio di trasformazione di una espressione booleana in circuito logico.*

Qualora si debba invece trasformare un circuito logico in funzione booleana, si deve suddividere l'espressione secondo l'ordine inverso a quello con cui vengono effettuate le operazioni, come mostrato nella fig. 1.48.



Schema logico — Scomposizione della funzione booleana

Fig. 1.48 *- Esempio di trasformazione di uno schema logico in una funzione booleana.*