Algebra booleana

Ver. 2

Algebra booleana

- George Boole, matematico e logico britannico del XIX secolo, è considerato il fondatore della logica matematica
- L'algebra (o logica) booleana opera su variabili *logiche* (o *booleane*) che possono assumere solo due valori: **Vero** o **Falso**
- Se si considerano le equivalenze Vero → 1 e Falso → 0 si può allora utilizzare l'algebra di Boole per descrivere il funzionamento e i dati di un sistema elettronico digitale

Algebra booleana

- L'algebra booleana rappresenta "*eventi binari*", ossia condizioni che possono assumere solo due valori
  - Esempio
    Una lampadina può essere o accesa o spenta
- Le *espressioni booleane o logiche* operano su variabili logiche e producono come risultato valori logici (Vero e Falso, ossia 1 e 0)

Tabelle della verità

- Una *funzione booleana* F ha la caratteristica di operare su variabili logiche v_1, v_2, …, v_n e produrre un valore logico, si indica genericamente con: $F(v_1,v_2, \ldots ,v_n)$
- Può essere espressa o come *espressione logica* o mediante la corrispondente **tabella della verità** (“truth table”)

Tabelle della verità

- La **tabella della verità** (“truth table”) è l’elenco *ordinato* di tutte le possibili combinazioni delle variabili logiche v_1, v_2, …, v_n con il il corrispondente risultato

Tabelle della verità

- Esempio
  La tabella della verità qui a fianco definisce la funzione $F(v_1,v_2,v_3)$
  - Ogni variabile booleana può assumere solo 2 valori per cui n variabili producono 2^n possibili combinazioni, ogni combinazione corrisponde a una riga della tabella

v_3	v_2	v_1	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Funzioni booleane

- Un’espressione verbale può essere descritta mediante una funzione logica, in particolare con una tabella di verità
- Ad esempio, si analizzi l’espressione verbale:

Per superare l’esame X, lo studente:

- *deve aver superato l’esonero e l’orale*

oppure

- *deve aver superato l’esame scritto e l’orale*

Funzioni booleane

- Bisogna innanzitutto identificare gli *eventi*
- Ogni evento può essere o *indipendente* o *dipendente* (ossia dipendente dagli eventi indipendenti) e si assegna una variabile logica a ciascun evento (indipendente e dipendente):

$a \rightarrow$ esonero (var. indip.), 1=superato

$b \rightarrow$ compito scritto (var. indip.), 1=superato

$c \rightarrow$ esame orale (var. indip.), 1=superato

$e \rightarrow$ risultato esame (var. dipend.), 1=superato

Funzioni booleane

- Essendoci 3 variabili indipendenti, si crea una tabella di 3 colonne (una per variabile) più una per il risultato e
- Le 2^3 righe sono tutte le combinazioni dei valori di a, b e c, queste devono essere elencate in ordine progressivo: 0, 1, 2, ..., 7 in binario, ossia 000, 001, 010, ..., 111

a	b	c	e
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Funzioni booleane

- La tabella della verità viene compilata considerando il significato delle variabili, ad es. la riga *101* significa:
  a=*1* → superato l'esonero
  b=*0* → non superato (o tentato) lo scritto
  c=*1* → superato l'orale
  quindi l'esame è superato e si pone e=1

a	b	c	e
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	0
1	1	1	1

Funzioni booleane

- Considerazione: si supponga che siano valide anche le seguenti condizioni:
  - si può sostenere l’orale solo **dopo** aver sostenuto l’esonero o lo scritto
  - se si supera l’esonero, **non** si può sostenere l’esame scritto

Con l’aggiunta di queste condizioni, alcune combinazioni dei valori degli ingressi non sono realistiche

Funzioni booleane

- Considerazione (seguito):
  - Con riferimento alla tabella della verità precedente, si ha che le combinazioni:

  a=0, b=0, c=1 (no eson, no scritto, sup orale)
  a=1, b=1, c=1 (sup eson, sup scritto, sup or.)

  non si verificheranno mai

Funzioni booleane

- Considerazione (seguito):
  - *La tabella deve essere sempre completa di tutte le righe*, quindi un valore 0 o 1 per e deve sempre essere espresso
  - Normalmente è ovvio quale debba essere:

    a=0, b=0, c=1 → *no superam. esame* → e=0

    a=1, b=1, c=1 → sì *superam. esame* → e=1

Funzioni booleane

- Considerazione (seguito):
  - Quando si deve semplificare l'espressione logica equivalente derivata dalla tabella o disegnarne il circuito logico, ci si può avvantaggiare del fatto che non si verifichino quelle combinazioni indicando che per esse un valore *qualsiasi* di e sia *accettabile*
  - Nella tabella della verità questo si esprime con il simbolo "**–**" detto "*don't care* "

Funzioni booleane

- Considerazione (seguito):
  - Ad esempio, per una successiva semplificazione, la tabella già vista si può scrivere come a lato
  - I *don't care* sono indipendenti, ciascuno potrà essere esplicitato a 0 o a 1 a seconda del maggior vantaggio che se ne ottiene in un successivo utilizzo (es. in un circuito)

a	b	c	e
0	0	0	0
0	0	1	–
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	–

Operatori logici/booleani

- Le variabili logiche possono essere combinate mediante gli operatori logici per formare espressioni logiche
- Il risultato di un'espressione logica è sempre un valore logico (Vero/Falso, 1/0)
- Gli operatori principali sono:
  - AND, OR, NOT
  - EX-OR (sebbene esprimibile utilizzando AND, OR e NOT)

Operatori logici/booleani - AND

AND *(prodotto logico)*

- Simbolo: · (come il prod. algebrico), spesso omesso
- Combina 2 valori logici secondo la definizione:

  $0 \cdot 0 = 0$

  $0 \cdot 1 = 0$

  $1 \cdot 0 = 0$

  $1 \cdot 1 = 1$

- Il risultato è Vero (1) solo se <u>entrambi</u> gli operandi sono Veri (1) [ossia il risultato è Falso se <u>almeno un</u> operando è Falso]

Operatori logici/booleani - AND

- Esempio

"*Vado al mare se è soleggiato e fa caldo*"

X="*è soleggiato*"
Y="*fa caldo*"
Z="*vado al mare*"

$Z = X \cdot Y$

Entrambe le condizioni "*è soleggiato*" e "*fa caldo*" devono essere vere perché sia vera l'espressione "*vado al mare*"

Operatori logici/booleani - OR

OR *(OR inclusivo, somma logica)*

- Simbolo: + (come la somma algebrica)
- Combina 2 valori logici secondo la definizione:

  $0 + 0 = 0$

  $0 + 1 = 1$

  $1 + 0 = 1$

  $1 + 1 = 1$

- Il risultato è Vero (1) solo se <u>almeno un</u> operando è Vero (1), eventualmente tutti e due [ossia il risultato è Falso solo se <u>entrambi</u> sono Falsi]

Operatori logici/booleani - OR

- Esempio
  - "*Prendo l'auto se piove o fa freddo*"

X="*piove*"
Y="*fa freddo*"
Z="*prendo l'auto*"

Z = X + Y

Almeno una delle espressioni "*piove*" e "*fa freddo*" deve essere vera perché sia vera l'espressione "*prendo l'auto*", (anche entrambe)

Operatori logici/booleani - NOT

NOT (Negazione)

- Simbolo: un trattino al di sopra della variabile o dell'espressione da negare (es. $\overline{a}$) oppure un apice ' a destra (es. a', se è un'espressione, deve essere tra parentesi)
- Si applica a un'unica quantità secondo la def.:

  $\overline{0}$ = 1

  $\overline{1}$ = 0
- Il risultato è Vero (1) *solo se* l'operando è Falso (0) e viceversa

Operatori logici/booleani - NOT

- Esempio
  - "*Vengo a piedi se NON è lontano*"

    X="*è lontano*"

    Z="*vengo a piedi*"

    $$Z = \overline{X}$$

    Se non è vero che "*è lontano*" allora è vero che "*vengo a piedi*",

    se è vero che "*è lontano*" allora non è vero che "*vengo a piedi*"

Operatori logici/booleani - EXOR

EX-OR (EXclusive OR – OR esclusivo)

- Simbolo: $\oplus$
- Combina 2 valori logici secondo la definizione:

  $0 \oplus 0 = 0$

  $0 \oplus 1 = 1$

  $1 \oplus 0 = 1$

  $1 \oplus 1 = 0$
- Il risultato è Vero (1) solo se **uno solo** degli operandi è Vero (1) [ossia se gli op. sono diversi]
- Si può ricavare combinando AND, OR e NOT

Operatori logici/booleani - EXOR

- Esempio
  - “Si può sganciare il carrello se si inserisce una moneta o da 1 Euro o da 2 Euro”

    X=“si inserisce una moneta da 1 Euro”
    Y=“si inserisce una moneta da 2 Euro”
    Z=“si può sganciare il carrello”

    $Z = X \oplus Y$

    Solo una tra le espressioni X e Y può essere vera perché sia vera “si può sganciare il carrello”, non nessuna e neppure entrambe (si escludono)

Espressioni logiche

- Un'espressione logica (o booleana) è composta da:
  - variabili logiche
  - costanti (0 e 1, Vero/Falso)
  - operatori logici
  - parentesi

  Esempi
  - $\overline{a + b} \cdot c$
  - $a\overline{b} + c(d + \overline{a}\,\overline{e}) \cdot \overline{c \oplus e}$

Espressioni logiche

- Come nelle *espressioni algebriche* c'è una ***priorità*** tra le operazioni (ad es. in $-3+2\cdot 3$, il segno – viene valutato per primo, poi la moltiplicazione, poi la somma), così nelle *espressioni logiche* la priorità è (in ordine decrescente):
  - **NOT**
  - **AND**
  - **OR**

(l'EX-OR è un derivato)

Espressioni logiche

- Quindi $F = a + b \cdot c$ è valutata come se ci fossero le parentesi indicate: $F = a + (b \cdot c)$
- Per forzare una valutazione diversa degli operatori si possono usare le parentesi:
  $F = (a + b) \cdot (c + d)$
  senza parentesi sarebbe valutata:
  $F = a + (b \cdot c) + d$
- La barra di negazione rende *le* parentesi *superflue:* $\overline{a + b} \cdot c \rightarrow \overline{(a + b)} \cdot c$

Espressioni logiche equivalenti

- Due **espressioni** F_1 e F_2 si dicono **equivalenti** se, per gli stessi valori delle variabili indipendenti, entrambe producono lo stesso risultato, ossia hanno la stessa tabella della verità
- Esempio di espressioni equivalenti:
  - $F_1 = x + \bar{x}y$ e $F_2 = x + y$

Espressioni logiche equivalenti

- Per verificare se due espressioni sono equivalenti si procede o con opportuni calcoli logici o confrontando le tabelle della verità
- Ad esempio per $F_1 = x + \bar{x}y$ e $F_2 = x + y$:

x	y	F_1
0	0	0
0	1	1
1	0	1
1	1	1

x	y	F_2
0	0	0
0	1	1
1	0	1
1	1	1

Espressioni logiche complementari

- Due **espressioni** F_1 e F_2 si dicono **complementari** se, per gli stessi valori delle variabili indipendenti, producono sempre risultati complementari (0→1 e 1→0), ossia hanno le tabelle della verità complementari (ossia <u>nella colonna dei risultati</u> gli 0 e gli 1 sono invertiti)
- Esempio di espressioni complementari
  - $F_1 = x \cdot \bar{y} + \bar{x} \cdot y$ e $F_2 = x \cdot y + \bar{x} \cdot \bar{y}$

Espressioni logiche complementari

- Per verificare se due espressioni sono complementari si procede o con opportuni calcoli logici o confrontando le tabelle della verità, ad esempio per:
  $F_1 = x \cdot \bar{y} + \bar{x} \cdot y$ e $F_2 = x \cdot y + \bar{x} \cdot \bar{y}$:

x	y	F_1
0	0	0
0	1	1
1	0	1
1	1	0

x	y	F_2
0	0	1
0	1	0
1	0	0
1	1	1

Tabelle complementari

Espressioni logiche duali

- Due **espressioni** F_1 e F_2 si dicono **duali** se:
  - ogni OR di F_1 corrisponde a un AND di F_2 e viceversa
  - ogni 1 di F_1 corrisponde a uno 0 di F_2 e vicev.
  - l'ordine di valutazione delle espressioni è uguale
- Esempio $F_1 = a + b \cdot (\overline{c} + 1)$ e
  $F_2 = a \cdot (b + \overline{c} \cdot 0)$

Espressioni logiche duali

- Esempio: calcolare la duale dell'espressione

$$F_1 = a + b \cdot (\overline{c} + 1)$$

Sostituiti $+$ con $\cdot$ e 1 con 0 (e viceversa), l'ordine di valutazione deve essere preservato, questo è:

$$F_1 = \underbrace{a + \underbrace{b \cdot \underbrace{(\overline{c} + 1)}}}$$

per mantenerlo si devono aggiungere delle parentesi (eliminando quelle sovrabbondanti):

$$F_2 = \underbrace{a \cdot \underbrace{(b + \underbrace{\overline{c} \cdot 0})}}$$

altrimenti si calcolerebbe prima l'AND di a e b

Semplificazione delle espr. logiche

- Un’espressione semplificata è più veloce da calcolare (e corrisponde a un circuito elettronico più piccolo, quindi meno costoso e con minor consumo energetico)
- La semplificazione può essere effettuata in diversi modi, tra questi si hanno quelli che utilizzano:
  - gli assiomi e i teoremi dell’algebra di Boole
  - le mappe di Karnaugh (trattate in altri corsi)

Assiomi e teoremi dell’algebra b.

- **Teorema della dualità**:
  se un’equivalenza è vera, anche la duale è vera
- **Assiomi e Teoremi** (duali sulla stessa riga)

1.	$x \cdot 0 = 0$	$x + 1 = 1$
2.	$x \cdot 1 = x$	$x + 0 = x$
3.	$x \cdot \bar{x} = 0$	$x + \bar{x} = 1$
4.	$x \cdot x = x$	$x + x = x$
5.	$x \cdot y = y \cdot x$	$x + y = y + x$

Assiomi e teoremi dell'algebra b.

6. $x \cdot y \cdot z = (x \cdot y) \cdot z = x \cdot (y \cdot z)$	$x+y+z = (x+y) + z = x + (y+z)$

7. Teorema di De Morgan

$\boxed{\overline{x \cdot y \cdot z \cdot \ldots} = \overline{x}+\overline{y}+\overline{z}+\ldots}$	$\boxed{\overline{x+y+z+\ldots} = \overline{x} \cdot \overline{y} \cdot \overline{z} \cdot \ldots}$
8. $x \cdot y+x \cdot z = x \cdot (y+z)$	$\boxed{(x+y) \cdot (x+z) = x+y \cdot z}$
9. $x+x \cdot y = x$	$x \cdot (x+y) = x$
10. $x \cdot y + x \cdot \overline{y} = x$	$(x+y) \cdot (x+\overline{y}) = x$
11. $\boxed{x + \bar{x} \cdot y = x + y}$	$x \cdot (\bar{x} + y) = x \cdot y$

12. $x \oplus y = x \cdot \overline{y}+\overline{x} \cdot y = \overline{x} \oplus \overline{y}$ $\quad$ $\overline{x \oplus y}=\overline{x} \oplus y=x \oplus \overline{y}=xy+\overline{x}\overline{y}$

N.B. I meno intuitivi sono incorniciati

Assiomi e teoremi dell’algebra b.

- È IMPORTANTE NOTARE CHE
nelle precedenti eguaglianze, le variabili x, y e z possono essere tanto singole variabili quanto intere espressioni

  Ad es. in $(a \cdot b + c \cdot d) + 1$
  il contenuto delle parentesi corrisponde alla x della regola $x+1=1$ (1*bis*) e quindi si ha:
  $(a \cdot b + c \cdot d) + 1 = 1$

Assiomi e teoremi dell'algebra b.

- Il teorema di De Morgan si può facilmente ricordare come:
  - la negazione del prodotto è pari alla somma dei negati
  - la negazione della somma è pari al prodotto dei negati
- Il complementare di $x \cdot y$ non è $\bar{x} \cdot \bar{y}$, ma $\bar{x} + \bar{y}$
- Il complementare di $x + y$ non è $\bar{x} + \bar{y}$, ma $\bar{x} \cdot \bar{y}$

Esempi di semplificazioni

1. $\boxed{a} + \bar{b} + \boxed{\bar{a}b} + (a + \bar{b})\bar{a}b =$ r.11: $x + \bar{x} \cdot y = x + y$

$= \boxed{a + b} + \bar{b} + (a + \bar{b})\bar{a}b =$ r.3b: $x + \bar{x} = 1$

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

$= 1 + \textit{qualsiasi espress.} = 1$ r.1b: $x + 1 = 1$

2. $\boxed{(a + \bar{b})\bar{a}}\boxed{b} + \bar{c} =$ r.11b: $x \cdot (\bar{x} + y) = x \cdot y$

$= \boxed{ba} \cdot \bar{a} + \bar{c} =$ r.3: $x \cdot \bar{x} = 0$

$= b \cdot 0 + \bar{c} = \bar{c}$ r.1: $x \cdot 0 = 0$

Esempi di semplificazioni

- Alcune dimostrazioni di regole

9. $x+x\cdot y = x\cdot 1+x\cdot y = x\cdot\cancel{(1+y)} = x\cdot 1 = x$

8. $(x+y)\cdot(x+z) \overset{th.\ 8}{=} (x+y)\cdot x + (x+y)\cdot z = \cancel{x}x+xy+xz+yz =$

$= x+xy+xz+yz = x\cdot 1 +xy+xz+yz =$

$= x\cdot\cancel{(1+y+z)}+yz = x\cdot 1+yz = x+yz$

10. $x\cdot y+x\cdot\overline{y} = x\cdot\cancel{(y+\overline{y})} = x\cdot 1 = x$

11. $x+\overline{x}\cdot y = x\cdot 1+\overline{x}y = x\cdot(1+y)+\overline{x}y = x+\overbrace{xy+\overline{x}y}^{\text{teorema 10}} = x+y$

- Inoltre: $xy + \overline{x}y + x\overline{y} + \overline{x}\,\overline{y} = 1$

infatti $y\cancel{(x+\overline{x})} + \overline{y}\cancel{(x+\overline{x})} = y + \overline{y} = 1$

Esempi di semplificazioni

- Altra dimostrazione della regola 11
  - $x+\bar{x}\cdot y \;= x\cdot 1+\bar{x}y = x\cdot(y+\bar{y})+\bar{x}y =$
    $= xy+x\bar{y}+\bar{x}y =$ [*regola 4b*] $= \boxed{xy}+xy+\boxed{x\bar{y}}+\bar{x}y =$
    $= x(\cancel{y+\bar{y}})+y(\cancel{x+\bar{x}}) = x+y$
    - Qui si è utilizzata la regola 4b a rovescio: questa afferma che un termine può essere sostituito da un OR di se stesso con altre sue copie: $x=x+x+x+\ldots$
    - Analogamente la 4a afferma che un termine può essere sostituito da un AND di se stesso con altre sue copie: $x=x\cdot x\cdot x\cdot\ldots$

Esempio

- Scrivere l'espressione logica che identifica se un anno è bisestile. Un anno è bisestile se è divisibile per 4 e:
  1. o non è divisibile per 100
  2. o è divisibile per 400
- Siano D4 = "divisibile per 4"
  D100 = "divisibile per 100"
  D400 = "divisibile per 400"

Allora $\text{Bisestile} = d4 \cdot (\overline{d100} + d400)$

Esempio

- Può essere utile, quando possibile, considerare anche l'aspetto algebrico del problema: "divisibile per 400" implica "divisibile per 4", quindi in questo caso si può avere una forma alternativa equivalente:

$$\text{Bisestile} = d4 \cdot (\overline{d100} + d400) =$$
$$= d4 \cdot \overline{d100} + d4 \cdot d400 =$$
$$= d4 \cdot \overline{d100} + d400$$

Passaggio funzione → tabella

- Per ricavare la tabella di verità corrispondente ad una funzione logica, devono essere calcolati i risultati della funzione per tutte le combinazioni delle variabili indipendenti
- Può essere d'aiuto creare colonne della tabella della verità contenenti i calcoli intermedi

Passaggio funzione → tabella

- Esempio
  Calcolare la tabella della verità della funzione
  $F(a,b,c) = (ab + \overline{b}) \cdot \overline{c}$
  - si calcola facilmente ab in una colonna (vale 1 solo se a=1 e b=1)
  - si calcola $\overline{b}$
  - si calcola $ab+\overline{b}$ come OR delle colonne precedenti (vale 0 solo se entrambe sono 0)
  - etc. la tabella della verità è la parte indicata i grassetto, le altre colonne sono solo per i calcoli

Passaggio funzione → tabella

abc	ab	$\overline{b}$	$ab + \overline{b}$	$\overline{c}$	$\mathbf{(ab+\overline{b}) \cdot \overline{c}}$
000	0	1	1	1	**1**
001	0	1	1	0	**0**
010	0	0	0	1	**0**
011	0	0	0	0	**0**
100	0	1	1	1	**1**
101	0	1	1	0	**0**
110	1	0	1	1	**1**
111	1	0	1	0	**0**

Espressioni in forma canonica

- **Forma canonica SP**
  - **S**omma (logica) di **P**rodotti (logici): SP
  - Ogni termine si chiama *minterm* ed è il prodotto di tutte le variabili (alcune affermate altre negate)

Esempio di espressione SP composta da 4 minterm

$$F(a,b,c) = \bar{a}bc+ab\bar{c}+\bar{a}b\bar{c}+abc$$

Espressioni in forma canonica

- **Forma canonica PS**
  - **P**rodotto (logico) di **S**omme (logiche): PS
  - Ogni termine si chiama *maxterm* ed è la somma di tutte le variabili (alcune affermate altre negate)

Esempio di espressione PS composta da 3 maxterm

$$F(a,b,c) = (a+b+\overline{c}) \cdot (\overline{a}+b+c) \cdot (\overline{a}+b+\overline{c})$$

Passaggio tabella → funzione SP

- La funzione deve dare 1 per specifiche combinazioni delle var. indip, quindi:
  - per ciascuna riga che ha come risultato 1 scrivere il prodotto di tutte le variabili indip. (minterm), ciascuna riga corrisp. a un minterm
- Ogni minterm deve dare 1 per quella specifica combinazione delle var. indip. quindi:
  - per ciascuna riga con risultato 1, nel minterm corrispondente negare le variabili con valore 0
- Sommare i minterm

Passaggio tabella → funzione SP

- Esempio

a	b	c	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

$$\bar{a}\bar{b}c$$

$$F(a,b,c) = \bar{a}\bar{b}c+\bar{a}b\bar{c}+a\bar{b}\bar{c}+a\bar{b}c$$

Passaggio tabella → funzione PS

- La funzione deve dare 0 per specifiche combinazioni delle var. indip, quindi:
  - per ciascuna riga che ha come risultato 0 scrivere la somma di tutte le variabili indipend. (maxterm), ciascuna riga corrisp. a un maxterm
- Ogni maxterm deve dare 0 per quella specifica combinazione delle var. indip. quindi:
  - per ciascuna riga con risultato 0, nel maxterm corrispondente negare le variabili con valore 1
- Moltiplicare i maxterm

Passaggio tabella → funzione PS

- Esempio

a	b	c	F
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

$$(\overline{a}+\overline{b}+c)$$

$$F(a,b,c)=(a+b+c)\cdot(a+\overline{b}+\overline{c})\cdot(\overline{a}+\overline{b}+c)\cdot(\overline{a}+\overline{b}+\overline{c})$$

Trasformazione in forma canonica

- **Espressione SP**
  - Se in un prodotto manca una variabile (es. x), moltiplicare il prodotto per $(x+\bar{x})$ e risolvere eliminando i termini duplicati

Esempio

$$
\begin{aligned}
F(x,y,z) &= xyz+yz+\bar{z} \\
&= xyz+(x+\bar{x})yz+(x+\bar{x})(y+\bar{y})\bar{z} \\
&= xyz+\cancel{xyz}+\bar{x}yz+xy\bar{z}+x\bar{y}\bar{z}+\bar{x}y\bar{z}+\bar{x}\bar{y}\bar{z}
\end{aligned}
$$

  - È immediato passare da una forma canonica SP alla tabella corrispondente

Trasformazione in forma canonica

- **Espressione PS**
  - Se in una somma manca una variabile (es. x), sommare il prodotto $x \cdot \bar{x}$ e risolvere eliminando i termini duplicati ricordando la regola 8b: $(x+y)\cdot(x+z) = x+y\cdot z$

Esempio

$$F(x,y,z) = (x+\bar{y}+z) \cdot (x+\bar{z}) \cdot x$$
$$= (x+\bar{y}+z) \cdot (x+y\bar{y}+\bar{z}) \cdot (x+y\bar{y}+z\bar{z})$$
$$= (x+\bar{y}+z) \cdot (x+y+\bar{z}) \cdot (x+\bar{y}+\bar{z}) \cdot \text{etc.}$$

  - È immediato da forma canonica SP → tabella

Circuiti logici

- Una funzione logica può essere realizzata mediante un circuito elettronico digitale (o circuito logico)
- Le variabili indipendenti corrispondono ai valori dei segnali elettrici che entrano (input) nel circuito
- Le variabili dipendenti corrispondono ai segnali elettrici che escono (output) dal circuito

Circuiti logici

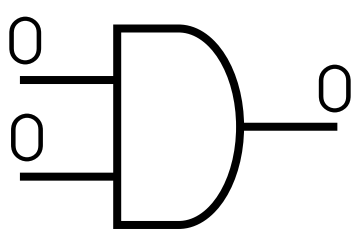
- I segnali hanno solo due livelli elettrici (in genere sono tensioni), corrispondenti ai valori 0 e 1 logici
- Esempio
  - 5 V → 1 logico
  - 0 V → 0 logico

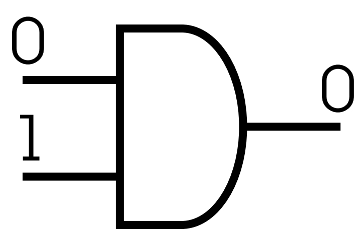
Porte logiche - Gates

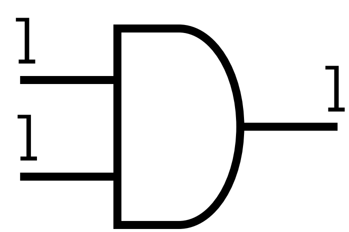
- Gli operatori logici hanno corrispondenza con componenti elettronici detti *porte logiche*, in Inglese [*logic] gates*
- Le porte sono collegate tra loro mediante fili che negli schemi sono rappresentati da linee
- In genere negli schemi si considera che il segnale si propaghi da sinistra a destra

Porte logiche - Gates

- Ad esempio, la porta [logica] AND dà in output un segnale corrispondente all'1 logico (Vero) solo se i suoi due segnali di ingresso corrispondono a un 1 logico, quindi si comporta come un operatore logico AND: $a \cdot b$

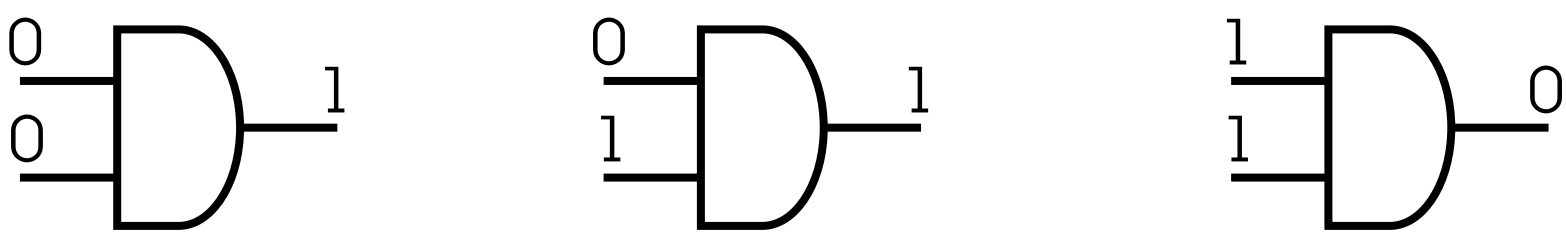






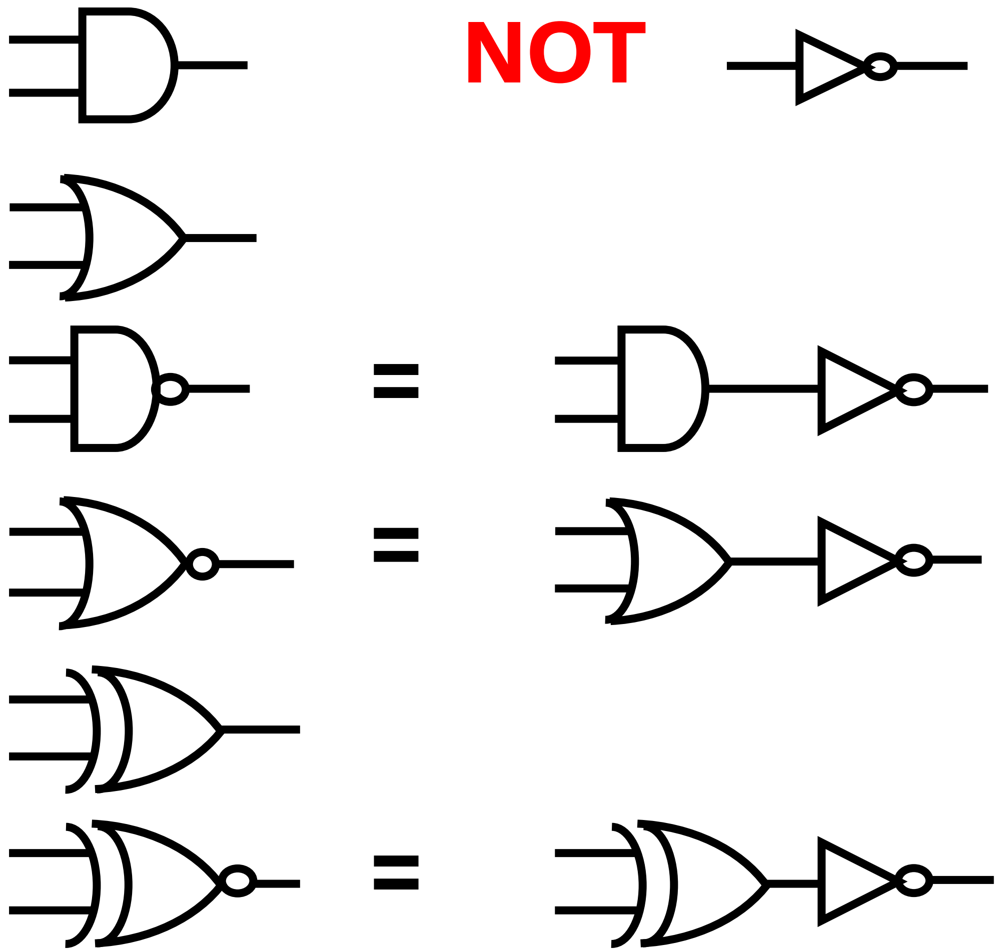
Porte logiche - Gates

- Esistono porte logiche che danno un'uscita negata rispetto all'operatore logico da cui derivano (es. la porta NAND dà uscita 0 solo quando entrambi gli ingressi sono a 1, equivale a una porta AND la cui uscita viene negata da una porta NOT: $\overline{a \cdot b}$



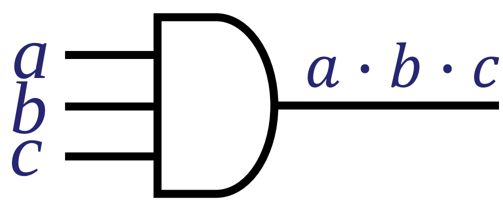
Simboli delle porte logiche

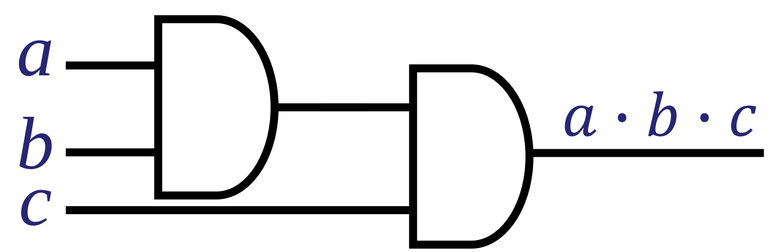
- **AND**
- **NOT**
- **OR**
- **NAND**
- **NOR**
- **EXOR**
- **EX-NOR**



Porte logiche multi-ingresso

- Equivalgono a porte logiche connesse in cascata, ad es. La AND a 3 ingressi dà risultato 1 solo se tutti i 3 ingressi sono a 1



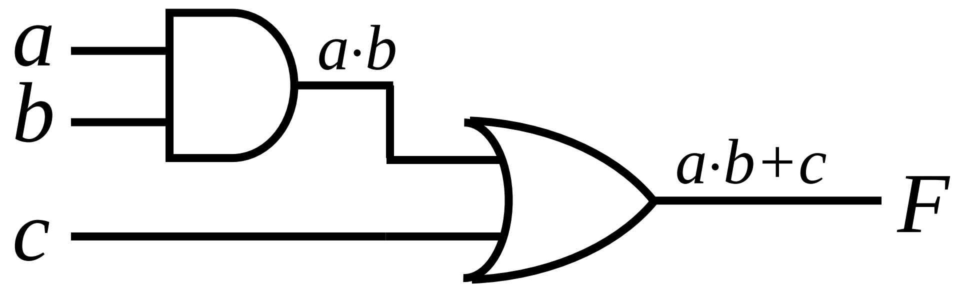


- La EX-OR multi-ingresso viene chiamata anche *disparità* in quanto dà 1 se il numero di ingressi a 1 è dispari

Circuiti logici

- **Esempio di circuito logico equivalente a un'espressione logica**

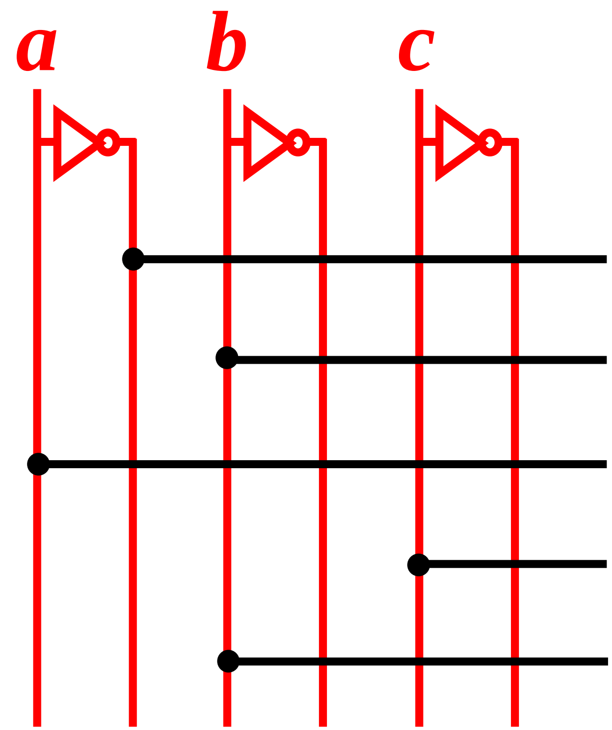
$$F = a \cdot b + c$$



La priorità degli operatori viene ottenuta grazie alla posizione relativa delle porte (qui la AND viene eseguita prima perché il segnale passa prima dalla AND e poi dalla OR)

Circuiti logici

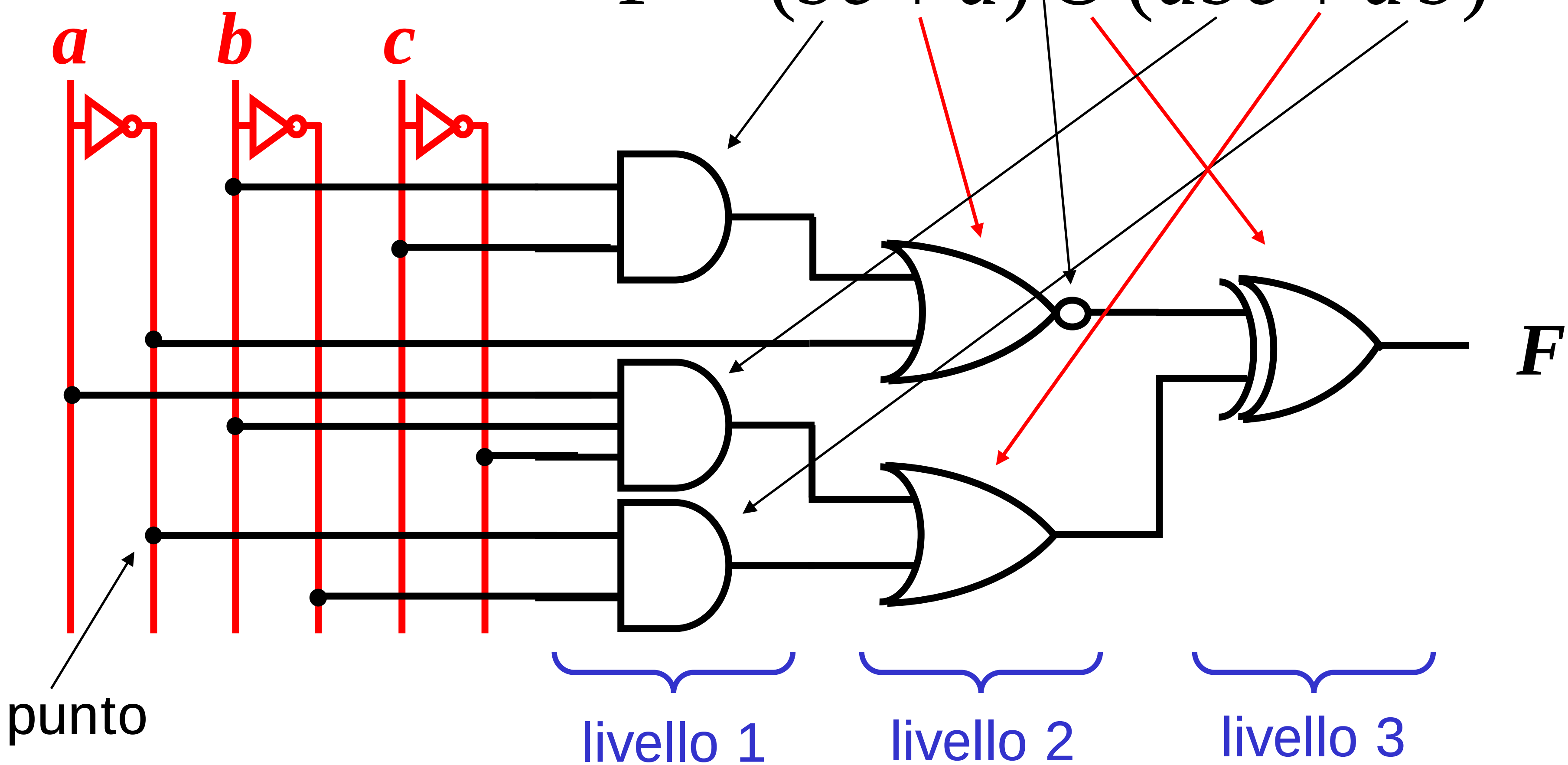
- Per disegnare un circuito più chiaro, è consigliabile fornire ingressi affermati e negati insieme nella parte sinistra del circuito, con lo schema qui indicato
  - I collegamenti tra fili sono rappresentati da un punto di medie dimensioni. Due line che si incrociano senza punto non sono connesse



Circuiti logici

- **Esempio**

$$F=\overline{(bc+\overline{a})}\oplus(ab\overline{c}+\overline{a}\,\overline{b})$$



Circuiti logici

- Ogni porta logica introduce un ritardo di propagazione (delay), es. 2 ns
- Se ogni porta ha un ritardo, quando i segnali in input vengono cambiati è necessario che passi un certo tempo prima che le uscite siano stabili
- Si deve considerare il caso peggiore (maggior numero di porte attraversate), nell’es. con 3 livelli il delay è 3×2 ns $= 6$ ns

Esercizi risolti

1. Quattro porte (A, B, C e D) separano due stanze. Le porte sono comandate da 3 interruttori (X, Y e Z) che, quando premuti, chiudono alcune delle porte: X chiude A e C, Y chiude B e D, Z chiude B e C. Ricavare la tabella di verità che descrive lo stato di separazione delle due stanze (1= separate, ossia tutte le porte sono chiuse, 0 = almeno una porta è aperta). Ricavarne la funzione e semplificarla.

Esercizi risolti

XYZ	ABCD	D
000	0000	**0**
001	0110	**0**
010	0101	**0**
011	0111	**0**
100	1010	**0**
101	1110	**0**
110	1111	**1**
111	1111	**1**

Le var. indip. sono X, Y e Z, 1=interruttore premuto A, B, C e D dipendono da X, Y e Z (1=porta chiusa)

La tabella di verità è la parte in grassetto, le colonne ABCD sono solo valori intermedi)

$D = XY\overline{Z} + XYZ = XY$

Esercizi risolti

2. Disegnare un circuito logico con input:
   1. un valore `D` in Complemento a 2 composto dai bit a e b
   2. una linea selettore `s` di 1 bit

   e output:
   1. un valore `U` in CA2 composto dai bit `x` e `y`
   2. un indicatore di overflow `w` di 1 bit

   Il circuito funzioni nel seguente modo:
   se `s`=0 → `U = D`,   se `s`=1 → `U =–D`
   `w` = 1 se l'operazione in CA2 è errata

Esercizi risolti

sab	D	U	xy	w
000	0	0	**00**	**0**
001	+1	+1	**01**	**0**
010	-2	-2	**10**	**0**
011	-1	-1	**11**	**0**
100	0	0	**00**	**0**
101	+1	-1	**11**	**0**
110	-2	*	**--**	**1**
111	-1	+1	**01**	**0**

$x=\bar{s}a\bar{b}+\bar{s}ab+s\bar{a}b=$

$\quad\bar{s}a+s\bar{a}b$

$y=\bar{s}\bar{a}b+\bar{s}ab+s\bar{a}b+sab=$

$\quad=b\cdot(\bar{s}\bar{a}+\bar{s}a+s\bar{a}+sa)=$

$\quad=b\cdot 1=b$

$w=sa\bar{b}$

con i *don’t care* = 0

Circuiti logici