Calcolatori:
Algebra Booleana e Reti Logiche

Algebra Booleana e Variabili Logiche

- I fondamenti dell'Algebra Booleana (o Algebra di Boole) furono delineati dal matematico George Boole, in un lavoro pubblicato nel 1847.
- In pratica vengono descritte le operazioni che manipolano le cosiddette *Variabili Logiche.*
- Una Variabile Logica è una variabile che può assumere solo due valori:
  - True (vero, identificato con **1**)
  - False (falso, identificato con **0**)
- Tale teoria si presta bene quindi allo studio e progettazione dei Circuiti Elettronici.

Algebra Booleana e Connettivi Logici

- Le operazioni più elementari che possono essere svolte dai Calcolatori Elettronici, sono implementate direttamente al livello **Hardware** (si pensi alla somma di due numeri interi).
- Queste operazioni elementari vengono implementate mediante la combinazione delle operazioni booleane fondamentali, rappresentate dai *Connettivi Logici*.
- I principali Connettivi Logici sono:
  - Congiunzione (AND o $\wedge$)
  - Disgiunzione (OR o $\vee$)
  - Negazione (NOT o $\neg$)

Algebra Booleana e funzioni booleane

- Combinando variabili logiche mediante i connettivi logici si ottengono le *espressioni logiche*.
- Un’espressione logica può essere definita in maniera esaustiva tramite la sua Tabella di Verità, che riporta il suo valore per ogni possibile configurazione delle variabili in essa contenute (Vedi la sezione *Algoritmi*).
- La Tabella di Verità di un’espressione logica identifica la cosiddetta **funzione booleana.**

Algebra Booleana e funzioni booleane

- Una funzione booleana di *n* variabili booleane $X_1, . . . ,X_n$ è un'applicazione

$$f : \{0, 1\}^n \rightarrow \{0, 1\}$$

- A differenza di quanto accade ad esempio per le funzioni reali di variabile reale, il numero di funzioni booleane in *n* variabili è finito.
- Il numero delle funzioni booleane di un argomento (es: NOT), è riconducibile alle possibili combinazione di due valori booleani, quindi è $2^2 = 4$.
- Se gli argomenti di f sono due (AND, OR), allora il numero delle funzioni è $2^4 = 16$.
- In generale, se k è il numero degli argomenti, il numero delle funzioni è $2\uparrow 2^k$.

Algebra Booleana e funzioni booleane

X1	X2	X3	*f*
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

- Esempio di funzione booleana in tre argomenti
- La colonna *f* rappresenta una delle 256 possibili funzioni
- Infatti, il numero di funzioni booleane distinte corrisponde alle possibili configurazioni che si possono assegnare all'ultima colonna.

Assiomi dell'Algebra Booleana

- Proprietà commutativa

$$X \vee Y = Y \vee X$$

$$X \wedge Y = Y \wedge X$$

- Proprietà distributiva

$$X \wedge (Y \vee Z) = (X \wedge Y) \vee (X \wedge Z)$$

$$X \vee (Y \wedge Z) = (X \vee Y) \wedge (X \vee Z)$$

Assiomi dell'Algebra Booleana

- Elemento neutro

$$X \vee 0 = X$$

$$X \wedge 1 = X$$

- Complementarietà degli inversi

$$X \vee (\neg X) = 1$$

$$X \wedge (\neg X) = 0$$

Algebra Booleana

- Definizione generale: un insieme non vuoto V contenente almeno i due elementi **0** ed **1** è un'Algebra Booleana se tra i suoi elementi sono definite:
  - un'operazione binaria (da $V \times V \rightarrow V$) somma (indicata con **+**)
  - un'operazione binaria (da $V \times V \rightarrow V$) prodotto (indicata con **·**)
  - un'operazione unaria (da $V \rightarrow V$) (indicata con la soprassegnatura)

  che soddisfano gli assiomi 1 - 4 visti in precedenza.

NOTA: si può dimostrare che l'insieme $V = \{0,1\}$ con gli operatori AND, OR e NOT costituiscono un'Algebra Booleana

Ricavare l'Espressione Algebrica dalla Tavola di Verità

- Supponiamo di avere una Tavola di Verità rappresentante una determinata funzione booleana.
- Si considerano solo le righe per le quali la variabile di uscita è vera:
  - la forma algebrica rappresenta le condizioni che rendono vera l'uscita
  - la funzione sarà falsa in tutti gli altri casi
- Si mettono in AND tutte le variabili di ingresso di una riga, negando quelle che assumono valore falso.
- Si mettono in OR le espressioni trovate per le righe considerate.

Ricavare l'Espressione Algebrica dalla Tavola di Verità

- Esempio: data la seguente Tavola di Verità

X1	X2	X3	f
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	0

- Si considerano solo le righe che rendono vera la variabile di uscita
- Si mettono in AND tutte le variabili di ingresso di ciascuna riga, negando quelle che assumono valore falso:

$\neg X1 \wedge X2 \wedge \neg X3$

$\neg X1 \wedge X2 \wedge X3$

$X1 \wedge \neg X2 \wedge \neg X3$

Ricavare l'Espressione Algebrica dalla Tavola di Verità

- La variabile di uscita è vera quando almeno una delle tre condizioni trovate è vera,
- Quindi, si pongono in OR le tre condizioni per ricavare l'espressione algebrica finale:

$\neg X1 \wedge X2 \wedge \neg X3 \vee \neg X1 \wedge X2 \wedge X3 \vee X1 \wedge \neg X2 \wedge \neg X3$

<u>NOTA</u>: tra gli operatori logici NOT, AND e OR esiste un ordine di *precedenza,* come elencato, da tener presente per la valutazione delle espressioni.

Espressioni logiche Equivalenti

- Due espressioni sono uguali (o *logicamente equivalenti*) se il loro valore di verità è identico per ogni assegnazione delle variabili.
- L'equivalenza può essere verificata mediante le tabelle di verità.
- Esempio: $X + X \cdot Y = X$

X	Y	$X \cdot Y$	$X+X \cdot Y$
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Riduzione delle funzioni booleane

- Una funzione booleana f può essere *ridotta* in una funzione f' se f' è *logicamente equivalente* ad f e la sua espressione algebrica contiene meno operatori e variabili.
- La riduzione di funzioni booleane è estremamente importante per garantire la massima efficienza possibile nel calcolarne i valori.

Metodi di Riduzione

- La riduzione di una funzione booleana f si ricava mediante manipolazione delle espressioni algebriche tramite gli assiomi dell'Algebra di Boole, con l'aggiunta di teoremi e proprietà, quali:

  - Teoremi di De Morgan

$$\neg(A \vee B) = \neg A \wedge \neg B$$
$$\neg(A \wedge B) = \neg A \vee \neg B$$

  - Proprietà dell'Idempotenza

$$A \vee A = A$$
$$A \wedge A = A$$

  - Proprietà del Limite superiore e inferiore

$$A \vee 1 = 1$$
$$A \wedge 0 = 0$$

Metodi di Riduzione: esempio

- Semplificare la seguente espressione:

$$(\neg X \wedge Y) \vee (X \wedge Y) \vee (X \wedge \neg Y)$$

- Dato che $Z \vee Z = Z$ (idempotenza) possiamo aggiungere uno dei termini presenti nell'espressione senza cambiarla.
- Aggiungendo $X \wedge Y$ si ottiene:

$$(\neg X \wedge Y) \vee (X \wedge Y) \vee (X \wedge \neg Y) \vee \color{red}{(X \wedge Y)}$$

- Utilizzando la proprietà distributiva dell'operatore OR

$$(\neg X \vee X) \wedge Y \vee X \wedge (\neg Y \vee Y)$$

- E infine dato che $Z \vee \neg Z = 1$ (complementarietà degli inversi) e $1 \wedge Z = Z$ (elemento neutro del prodotto)

$$(1 \wedge Y) \vee (X \wedge 1) = X \vee Y$$

Metodi di Riduzione: esempio

- Proprietà di Assorbimento

$$
\begin{aligned}
A + A \cdot B &= A \cdot 1 + A \cdot B \text{ (elemento neutro)} \\
&= A \cdot (1 + B) \text{ (proprietà distributiva)} \\
&= A \cdot 1 \text{ (limite superiore)} \\
&= A \text{ (elemento neutro)}
\end{aligned}
$$

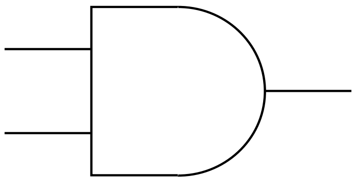
$$
\begin{aligned}
A \cdot (A + B) &= A \cdot A + A \cdot B \text{ (proprietà distributiva)} \\
&= A + A \cdot B \text{ (idempotenza)} \\
&= A \text{ (proprietà di assorbimento)}
\end{aligned}
$$

Reti Logiche

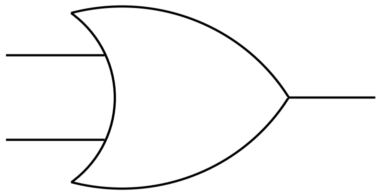
- La tecnologia a semiconduttori consente di realizzare su un’unica piastrina (*chip*) complesse strutture di componenti microscopici (*circuiti integrati*) che nel complesso realizzano <u>Reti Logiche</u> di dimensioni molto grandi.
- I componenti base di tali *circuiti* sono detti porte logiche (*logic gates*) e realizzano le funzioni logiche elementari AND,OR e NOT.
- Le porte logiche si basano su due valori di tensione dei segnali elettrici, alto (**H**) e basso (**L**), associati ai due valori di verità (alto=true=1, basso=false=0).

Rappresentazione Grafica delle Porte Logiche

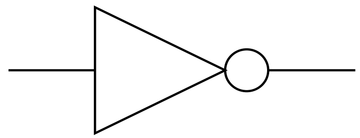
AND



OR



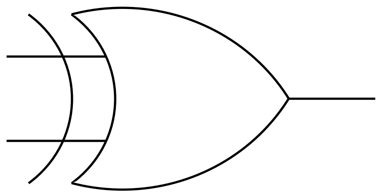
NOT



OR Esclusivo:

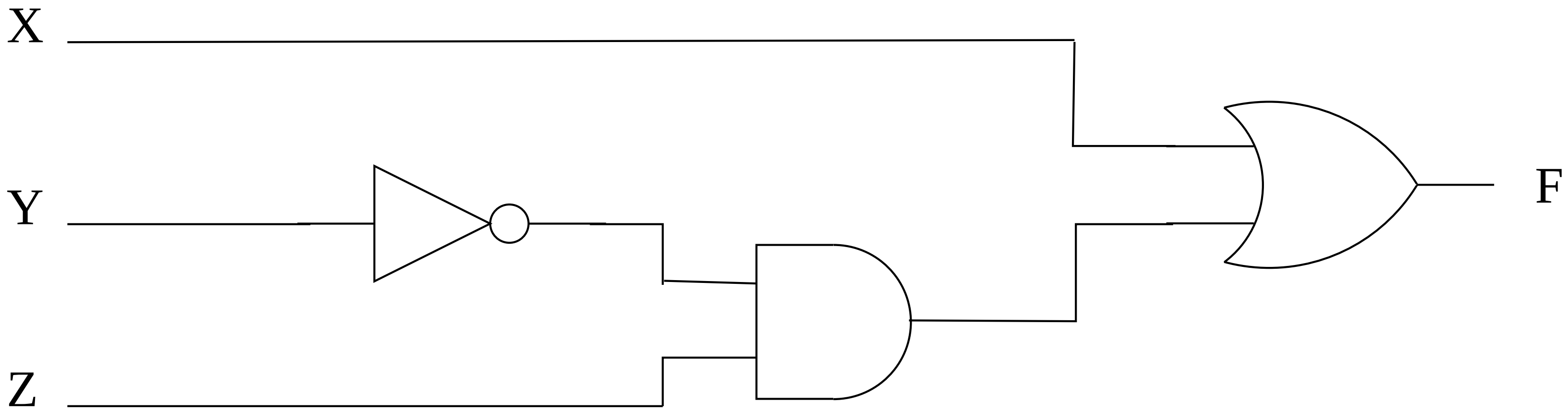
X	Y	X **XOR** Y ($\neg X \wedge Y \vee X \wedge \neg Y$)
0	0	0
0	1	1
1	0	1
1	1	0

XOR



Esempio di Rete Logica

- $F = X \vee \neg Y \wedge Z$



Reti Logiche: il Sommatore

- Supponiamo di voler progettare un Circuito per effettuare la somma di due numeri rappresentati da due sequenze di bit lunghe n.
- Per prima cosa consideriamo la Tavola di Verità rappresentante la situazione relativa al generico bit k:

c_k	a_k	b_k	s_k	c_{k+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

- c_k rappresenta l'eventuale riporto proveniente dalla *k-1-esima* somma, mentre a_k e b_k sono i bit *correnti* dei due addendi.
- s_k è il risultato della somma, mentre c_{k+1} è il riporto per la *k+1-esima* somma.
- In pratica ho una funzione booleana per la Somma e una per il Riporto.

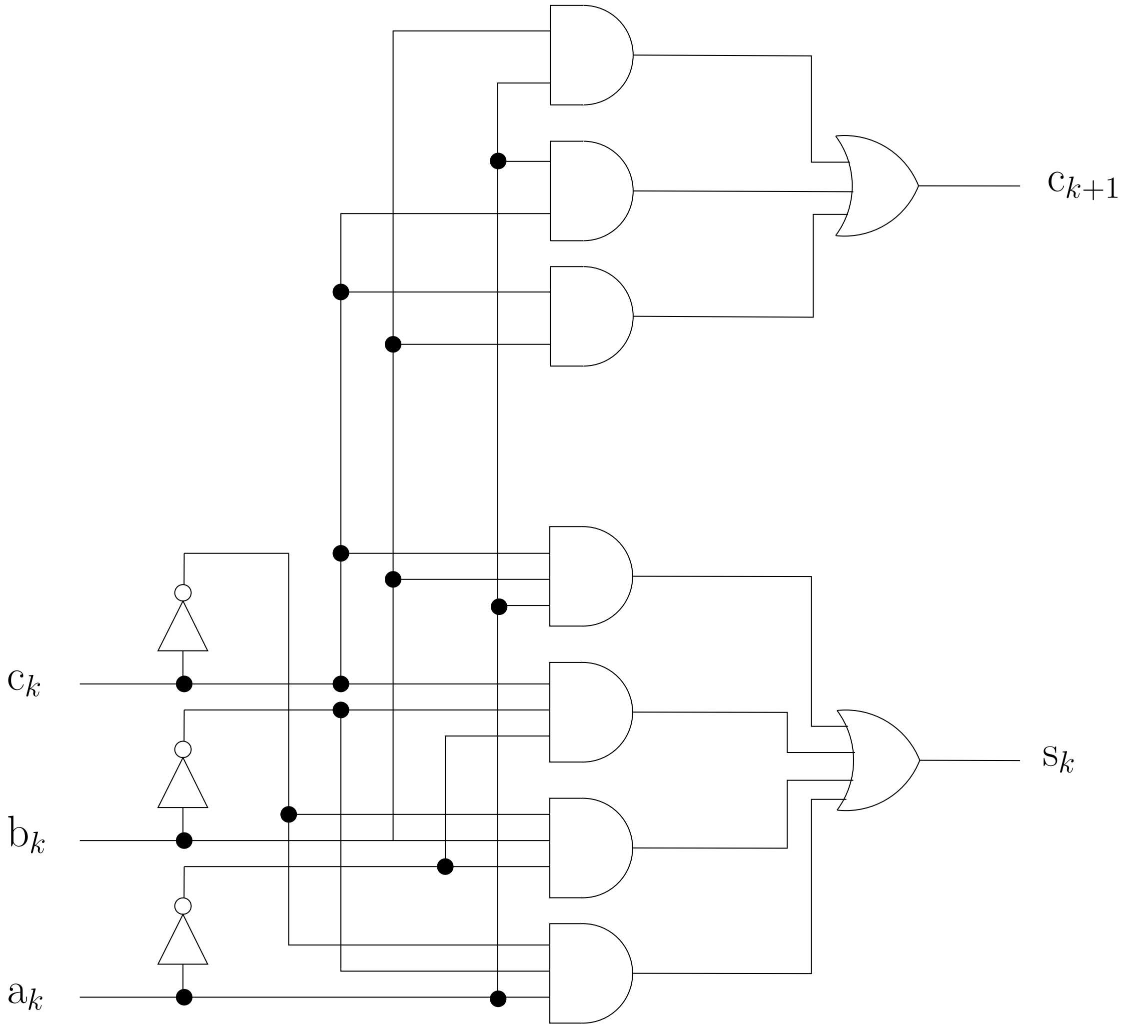
Reti Logiche: il Sommatore

- Una Rete che implementa le due funzioni prima descritte viene chiamata **full-adder**, per via della presenza del Riporto tra i valori in ingresso.
- Se il bit del Riporto non compare tra i valori in ingresso si ha un Semisommatore (**half-adder**); è il caso del Circuito che si occuperà della somma relativa al bit meno significativo dei due addendi.
- L'espressione algebrica semplificata per le due funzioni è la seguente:

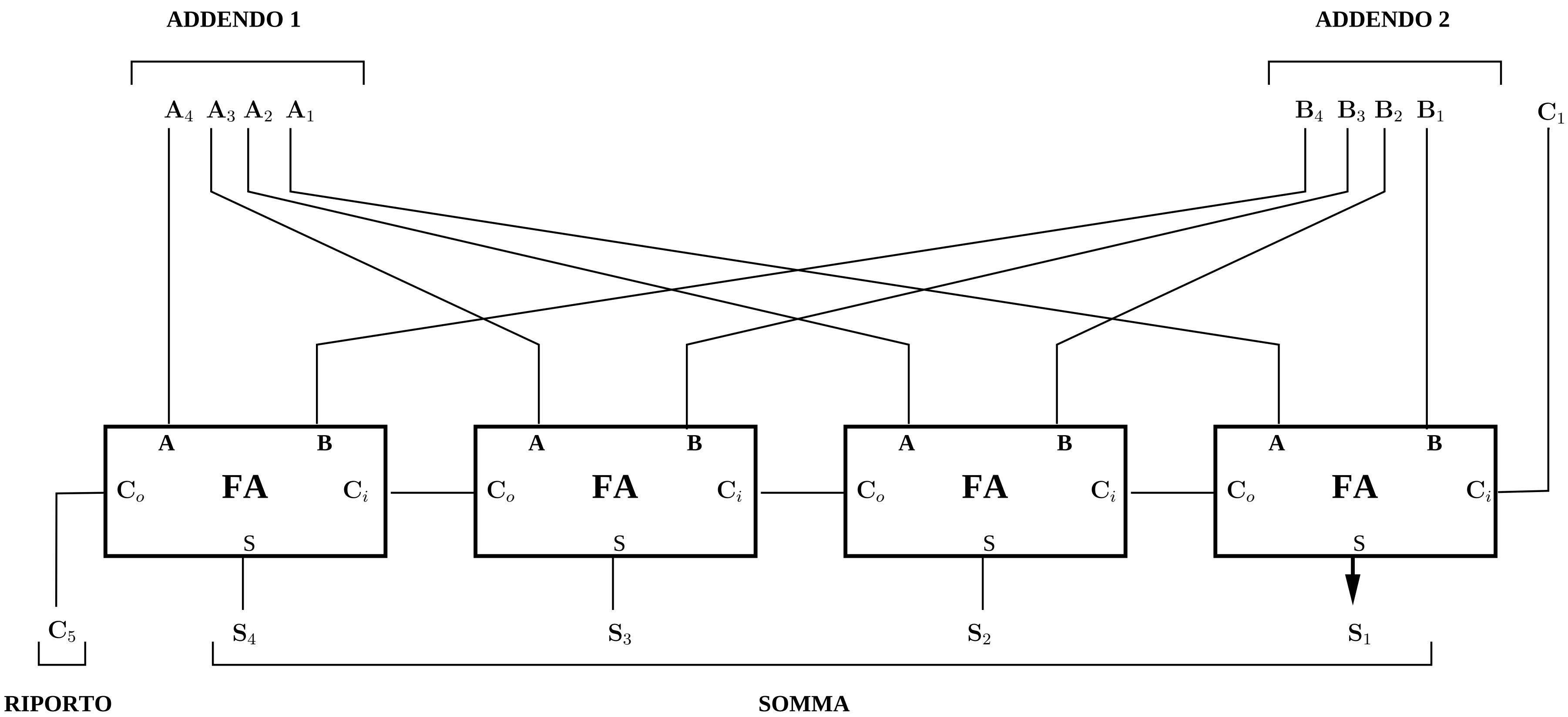
$$s_k = (\neg c_k \wedge \neg a_k \wedge b_k) \vee (\neg c_k \wedge a_k \wedge \neg b_k) \vee (c_k \wedge \neg a_k \wedge \neg b_k) \vee (c_k \wedge a_k \wedge b_k)$$

$$c_{k+1} = (a_k \wedge b_k) \vee (b_k \wedge c_k) \vee (c_k \wedge a_k)$$

Reti Logiche: il Sommatore



Reti Logiche: il Sommatore



Esempio di Sommatore a 4 bit ottenuto collegando in cascata 4 Full-Adders.