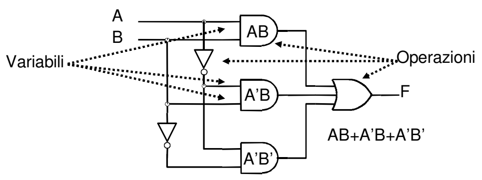
Richiami su Algebra di Boole

Espressioni Booleane

- Composte da
  - Variabili, costanti
  - Operatori logici
- Esempi
  - $F = A \cdot B' \cdot C + A' \cdot B \cdot C' + A \cdot B \cdot C + A' \cdot B' \cdot C'$
  - $F = (A+B+C') \cdot (A'+B'+C) \cdot (A+B+C)$
  - $F = A \cdot B' \cdot C' + A \cdot (B \cdot C' + B' \cdot C)$

Mapping sui circuiti

- Realizzate come **combinazione di gate logici**
- Ciascun gate implementa un'operazione presente nell'espressione
- Gli *input* dei gate sono le **variabili e costanti** coinvolte nelle espressioni logiche



Valutazione Espressioni

- Si ottiene sostituendo 0 e 1 per ogni variabile (**tutte le possibili combinazioni**)
- Uso di **tabelle di verità** per calcolare il valore dell'espressione
- Per un'espressione a **n** variabili la tabella avrà $\mathbf{2^n}$ righe

Tabella a 3 ingressi

#	A	B	C		F(A,B,C)
0	0	0	0		
1	0	0	1		
2	0	1	0		
3	0	1	1		
4	1	0	0		
5	1	0	1		
6	1	1	0		
7	1	1	1		

Tabella a 3 ingressi

#	A	B	C		F(A,B,C)
0	0	0	0		**1**
1	0	0	1		**0**
2	0	1	0		**1**
3	0	1	1		**1**
4	1	0	0		**0**
5	1	0	1		**0**
6	1	1	0		**1**
7	1	1	1		**1**

Tabella a 4 ingressi

#	A	B	C	D		F(A,B,C,D)
0	0	0	0	0		
1	0	0	0	1		
2	0	0	1	0		
3	0	0	1	1		
4	0	1	0	0		
5	0	1	0	1		
6	0	1	1	0		
…						
12	1	1	0	0		
13	1	1	0	1		
14	1	1	1	0		
15	1	1	1	1		

Esempio

Valutiamo l’espressione

$$F(A,B,C) = A'\cdot B'\cdot C + A\cdot B'\cdot C'$$

$F(A,B,C) = A'\cdot B'\cdot C + A\cdot B'\cdot C'$

#	A	B	C	A'B'C	AB'C'	A'B'C+AB'C'
0	0	0	0	**0**	**0**	**0**
1	0	0	1	**1**	**0**	**1**
2	0	1	0	**0**	**0**	**0**
3	0	1	1	**0**	**0**	**0**
4	1	0	0	**0**	**1**	**1**
5	1	0	1	**0**	**0**	**0**
6	1	1	0	**0**	**0**	**0**
7	1	1	1	**0**	**0**	**0**

Equivalenza di espressioni Booleane

- Due espressioni sono **equivalenti** se assumono lo stesso valore per tutte le combinazioni di valori delle variabili coinvolte
  - $F1 = (A + B)'$
  - $F2 = A'\cdot B'$
- Come posso verificarlo?
  - Tabelle di verità
  - Proprietà dell'algebra Booleana

Semplificazione di espressioni Booleane

- La **semplificazione** di espressioni Booleane consente di ottenere **espressioni equivalenti** che coinvolgono ***meno literal*** delle espressioni originali
  - **Literal** è una singola variabile in versione vera o negata
- Il circuito risultante sarà **più economico/efficiente** in quanto userà un **numero minore di porte logiche**
  - Si riducono anche le dimensioni del chip

Teoremi Fondamentali

Proprietà commutativa	$A+B= B+A$	$AB= BA$
Proprietà associativa	$A+(B+C) = (A+B)+C$	$A\cdot(BC)= (AB)\cdot C$
Proprietà distributiva	$A\cdot(B+C) = AB+AC$	$A+(BC)= (A+B)(A+C)$
Elemento nullo	$A+1 = 1$	$A\cdot 0 = 0$
Identità	$A+0 = A$	$A\cdot 1 = A$
Idempotenza	$A+A = A$	$A\cdot A = A$
Complemento	$A+A' = 1$	$A\cdot A' = 0$
Involuzione	$A'' = A$	
Assorbimento	$A+AB = A$	$A\cdot(A+B)= A$
Semplificazione	$A+A'B = A+B$	$A\cdot(A'+B)= AB$
De Morgan	$(A+B)' = A' B'$	$(AB)' = A'+B'$
Adiacenza Logica	$AB+AB' = A$	$(A+B)\cdot(A+B')= A$
Consenso	$AB+BC+A'C = AB+A'C$	$(A+B)(B+C)(A'+C)= (A+B)(A'+C)$

Teoremi fondamentali

versione inglese

Commutative Law	A + B = B + A	A.B = B.A
Associative Law	A + (B + C) = (A + B) + C	A . (B . C) = (A . B) . C
Distributive Law	A.(B + C) = AB + AC	A + (B . C) = (A + B) . (A + C)
Null Elements	A + 1 = 1	A . 0 = 0
Identity	A + 0 = A	A . 1 = A
Idempotence	A + A = A	A . A = A
Complement	A + A' = 1	A . A' = 0
Involution	A'' = A	
Absorption (Covering)	A + AB = A	A . (A + B) = A
Simplification	A + A'B = A + B	A . (A' + B) = A . B
DeMorgan's Rule	(A + B)' = A'.B'	(A . B)' = A' + B'
Logic Adjacency (Combining)	AB + AB' = A	(A + B) . (A + B') = A
Consensus	AB + BC + A'C = AB + A'C	(A + B) . (B + C) . (A' + C) = (A + B) . (A' + C)

Proprietà commutativa

L'ordine con cui le variabili sono poste in **OR** non cambia il risultato

$$A + B = B + A$$

L'ordine con cui le variabili sono poste in **AND** non cambia il risultato

$$A \cdot B = B \cdot A$$

Proprietà associativa

Se si pongono in <u>OR più di due variabili</u>, il <u>risultato è lo stesso</u> indipendentemente se le variabili sono raggruppate o meno

$$A + (B + C) = (A + B) + C$$

Se si pongono in AND più di due variabili<u>, il risultato è lo stesso</u> indipendentemente se le variabili sono raggruppate o meno

$$A \cdot (B \cdot C) = (A \cdot B) \cdot C$$

Teoremi di base

Elementi neutri e nulli

$$A + 0 = A \qquad A \cdot 1 = A$$
$$A + 1 = 1 \qquad A \cdot 0 = 0$$

Idempotenza

$$A + A = A \qquad A \cdot A = A$$

Involuzione

$$(A')' = A$$

Complementarità

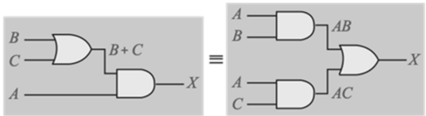
$$A + A' = 1 \qquad A \cdot A' = 0$$

Proprietà distributiva

Una variabile in comune tra due AND poste in OR (SOP) può essere messa a fattor comune

AB + AC = A(B+ C)

Risultato applicato sui circuiti: cosa cambia?



A(B+ C) **AB + AC**

Proprietà distributiva

Applicata a prodotti di somme (POS)

(A+B) (A+C) = A+BC

- Nota: vale soltanto per l'algebra Booleana!

Dim:

(A+B)(A+C)=

=AA+AC+BA+BC=

=A+AC+BA+BC=

=A(1+C+B)+BC=

=A+BC

Teoremi di De Morgan (I)

Il complemento del prodotto di due variabili è equivalente alla somma dei complementi

$$\overline{AB} = \overline{A} + \overline{B}$$

Applicazione a porte logiche

NAND Negative-OR

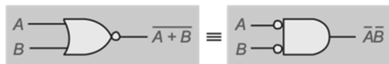
Inputs		Output	
A	B	$\overline{AB}$	$\overline{A} + \overline{B}$
0	0	1	1
0	1	1	1
1	0	1	1
1	1	0	0

Teoremi di De Morgan (II)

Il complemento della somma di due variabili è equivalente al prodotto dei complementi

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

Applicazione a porte logiche

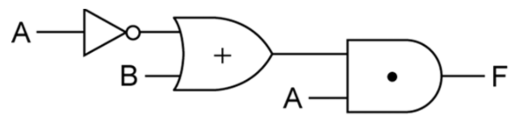


NOR Negative-AND

Inputs		Output	
A	B	$\overline{A + B}$	$\overline{A}\overline{B}$
0	0	1	1
0	1	0	0
1	0	0	0
1	1	0	0

Semplificazione di espressioni

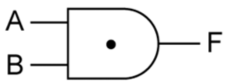
Semplificazione di circuiti



- Consideriamo il circuito sopra:
- Corrisponde a F=(A'+B)·A
- Come possiamo semplificarlo?

Semplificazione…

F=(A'+B)·A = A'A+AB = AB



Teorema dell'adiacenza logica

AB+AB'=A

- Dim: AB+AB'= A(B+B')=A

(A+B)(A+B')=A

- Dim: A+AB'+BA+BB'=A(1+B'+B)+0=A

Teorema dell'Assorbimento

A+AB=A

- Dim: A+AB=A(1+B)=A

A(A+B)=A

- Dim: A(A+B)=AA+AB=A+AB=A

Semplificazione di espressioni Booleane

(A+B')B = AB

- Dim: (A+B')B=

  =AB+B'B=AB+0=AB

AB'+B=A+B

Semplificazione di espressioni Booleane

(A+B')B = AB

- Dim: (A+B')B=
  =AB+B'B=AB+0=AB

AB'+B=A+B

- Dim: AB'+B= (B+A)(B+B') (prop. distributiva)
  = (B+A)1 = B+A

Teoremi per la semplificazione: sommario

Adiacenza logica:	AB+AB'=A	(A+B)(A+B')=A
Assorbimento:	A+AB=A	A(A+B)=A
Semplificazione:	(A+B')B = AB	AB'+B=A+B

Esercizio

Semplificare la seguente espressione:

$F = A' \cdot B \cdot C + A'$

Esercizio

Semplificare la seguente espressione:

$F = (A + B'C + D + EF)(A + B'C + (D + EF)')$

Esercizio

Semplificare la seguente espressione:

$F = (AB + C)(B'D + C'E') + (AB + C)'$

Esercizio

Semplificare la seguente espressione:

$F(A,B,C) = A' \cdot B \cdot C + A \cdot B' \cdot C + A \cdot B \cdot C$

Verificare l’equivalenza dell’espressione originale e di quella semplificata mediante tabelle di verità

Disegnare il circuito più economico

Teorema del consenso

AB + BC + A'C = AB + A'C
(A + B)(B + C)(A' + C) = (A + B)(A' + C)

- Utile per la semplificazione di espressioni Booleane
- Il termine che viene eliminato è detto **termine di consenso**.
- Dati 2 termini tali che una variabile appare in un termine e il suo complemento nell'altro termine, il termine di consenso è ottenuto moltiplicando (POS) o sommando (SOP) le restanti parti dei 2 termini

- il consenso di AB e A'C è dato da BC
- il consenso di (A+B) e (A'+C) è dato da (B+C)

Dim:

AB+A'C+BC=

AB+A'C + (A+A')BC =

=AB+A'C+ABC+A'BC=

=AB+ABC + (A'C+A'BC)=

AB(1+C)+A'C(1+B)=

=AB+A'C

Semplificazione mediante teorema del consenso

- Data l'espressione:

a'b'+ac+bc'+b'c+ab

- Identifichiamo i termini di consenso:

a'b'+ac+bc'+b'c+ab

- e quindi eliminiamoli. L'espressione diventa:

a'b'+ac+bc'

Attenzione!

L'ordine con cui si applica il teorema del consenso potrebbe variare l'esito della semplificazione finale

- Esempio:
  a'b'+ac+b'c+bd+cd
- Prima possibilità: eliminiamo **b'c** come consenso di a'b' e ac (come fatto prima):
  a'b'+ac+bd+cd =a'b'+ac+d(b+c)

Oppure…

a’b’+ac+b’c+bd+cd

- Eliminiamo **cd** come consenso di b’c e bd:
  a’b’+ac+b’c+bd
- A questo punto, posso ancora eliminare **b’c**:
  a’b’+ac+bd

Esercizio

Dimostrare che:
(A + B)(B + C)(A' + C) = (A + B)(A' + C)

Semplificazione di espressioni Booleane

Sommario

Semplificazione di espressioni Boooleane

1· Combinazione di termini, usando il teorema **AB + AB′ = A**

2· Eliminazione di termini. Usare il teorema **A + AB = A** per eliminare termini ridondanti;
Quindi, applicare il teorema del consenso
(AB + A′C + BC = AB + A′C) per eliminare termini di consenso;

3· Eliminazione di literal. Usare il teorema **A + A’B = A + B** per eliminare literal ridondanti·

4· Aggiunta di termini ridondanti: aggiunta di **AA′**, moltiplicare per **(A + A’)**, aggiungere **BC** a **AB + A’C**, aggiungere **AB** a **A.** L’idea è di aggiungere termini ridondanti in maniera tale che si possano ricombinare con altri termini

Proprietà XOR

Proprietà XOR

Elemento Neutro	**$X \oplus 0 = X$**
Involuzione	**$X \oplus 1 = X'$**
Elemento nullo	**$X \oplus X = 0$**
Complemento	**$X \oplus X' = 1$**
Commutativa	**$X \oplus Y = Y \oplus X$**
Associativa	**$(X \oplus Y) \oplus Z = X \oplus (Y \oplus Z)$**
Distributiva	**$X \cdot (Y \oplus Z) = X \cdot Y \oplus X \cdot Z$**
Negazione	**$(X \oplus Y)' = X \oplus Y' = X' \oplus Y = XY + X'Y'$**

Forme SOP e POS

Forme SOP e POS

- Le espressioni Booleane possono essere scritte come somme di prodotti (Sum of Products - SOP) o prodotti di somme (Products of Sums - POS)
- Utili per semplificare l'implementazione dei circuiti
- Esempi di SOP:

$\overline{A}\,\overline{B}\,\overline{C} + A\,B$ $\qquad A\,B\,\overline{C} + \overline{C}\,\overline{D}$ $\qquad C\,D + \overline{E}$

- Esempi di POS:

$(A + B)(\overline{A} + C)$ $\qquad (A + B + C)(\overline{B} + D)$ $\qquad (A + \overline{B})C$

Forma canonica SOP

- Nelle SOP in **<u>forma standard</u>**, ciascuna variabile deve apparire <u>in ciascun termine</u>
- ABC+A’BC è una forma SOP **<u>standard (CANONICA)</u>**
- AB’+BC è una forma SOP, ma **<u>non canonica</u>**
- Utile dal punto di vista implementativo
- Quando una variabile manca in un termine, <u>la si può aggiungere come somma della variabile stessa e del suo complemento</u>

Esempio

Convertiamo $F = \overline{A}\,\overline{B} + A\,B\,C$ in forma standard:

Notiamo che il primo termine non include la variabile C. Quindi, moltiplichiamo tale termine per **$(C + \overline{C})$ (uguale a 1)**

$$F = \overline{A}\,\overline{B}\,(C + \overline{C}) + A\,B\,C$$
$$= \overline{A}\,\overline{B}\,C + \overline{A}\,\overline{B}\,\overline{C} + A\,B\,C$$

Forma canonica POS

- Nella forma standard POS, **<u>ciascuna variabile deve apparire in ciascun termine somma</u>**
- Espressioni non-standard possono essere ricondotte a forme standard <u>aggiungendo il prodotto della variabile mancante e del suo complemento</u>, e infine applicando la <u>proprietà distributiva della somma sul prodotto</u> come segue:

  $A + BC = (A + B)(A + C)$

Esempio

Convertiamo $F = (\overline{A} + \overline{B})(A + B + C)$ in forma standard

La prima somma non include la variabile C. Quindi, aggiungiamo $C\,\overline{C}$

$$F = (\overline{A} + \overline{B} + C\,\overline{C})(A + B + C)$$

A questo punto applichiamo la proprietà
$A + BC = (A + B)(A + C)$

$$F = (\overline{A} + \overline{B} + C\,\overline{C})(A + B + C)$$
$$= (\overline{A} + \overline{B} + C)(\overline{A} + \overline{B} + \overline{C})(A + B + C)$$