

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

ALTRI OPERATORI ELEMENTARI

E' possibile definire altre funzioni od operatori elementari.

A	B	$f(A,B)=A \oplus B$
0	0	0
0	1	1
1	0	1
1	1	0

EX-OR

A	B	$f(A,B)=\overline{A \cdot B}$
0	0	1
0	1	1
1	0	1
1	1	0

NAND

A	B	$f(A,B)=\overline{A+B}$
0	0	1
0	1	0
1	0	0
1	1	0

NOR

Fondamenti di Informatica B/2 Gianni CONTE 7/22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

FUNZIONI BOOLEANA

Oltre alle funzioni OR e AND esistono altre funzioni a due variabili.
Le funzioni a due variabili sono complessivamente 16

X	Y	F0	F1	F2	F3	F4	F5	F6	F7	F8	F9	F10	F11	F12	F13	F14	F15
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1

AND X + aritm o XOR OR NOR NAND

Fondamenti di Informatica B/2 Gianni CONTE 8/22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

ALGEBRA BOOLEANA

L'algebra booleana opera su variabili che possono solo assumere valore 0 o 1.
Una funzione logica:

$$z = f(x_1, x_2, \dots, x_n)$$

È una legge che fa corrispondere a ogni combinazione di valori delle variabili indipendenti x_1, x_2, \dots, x_n uno e un solo valore binario della variabile Z

Fondamenti di Informatica B/2 Gianni CONTE 9/22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

PROPRIETA'

N.	PROPRIETA'	NOME
1	Esistono gli elementi $0, 1 \in K$ tali che: $A + 0 = A$ $A \cdot 1 = A$	Esistenza elementi identità
2	$A + B = B + A$ $A \cdot B = B \cdot A$	Proprietà commutativa
3	$A \cdot (B + C) = A \cdot B + A \cdot C$ $A + (B \cdot C) = (A + B) \cdot (A + C)$	Proprietà distributiva
4	Per ogni $A \in K$, esiste \bar{A} tale che: $A \cdot \bar{A} = 0$ e $A + \bar{A} = 1$	Esistenza dell'inverso

segue

Fondamenti di Informatica B/2 Gianni CONTE 10/22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

PROPRIETA'

N.	PROPRIETA'	NOME
5	$A + (B + C) = (A + B) + C$ $A(BC) = (AB)C$	Proprietà associativa
6	$A + A = A$ $A \cdot A = A$	Proprietà dell'idempotenza
7	$\overline{A+B} = \bar{A} \cdot \bar{B}$ $\overline{A \cdot B} = \bar{A} + \bar{B}$	Legge di DeMorgan
8	$\overline{\bar{A}} = A$	Involuzione

Fondamenti di Informatica B/2 Gianni CONTE 11/22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

PROPRIETA' III

Le tabelle precedenti riportano le proprietà principali dell'algebra booleana. La verifica di tali proprietà può avvenire in tutti i casi mediante:

- analisi esaustiva
- utilizzando le proprietà riportate in precedenza.

Fondamenti di Informatica B/2 Gianni CONTE 12/22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

PRINCIPIO DI DUALITA'

- Tutte le proprietà precedenti sono espresse a coppie.
- Una proprietà è derivabile dall'altra con lo scambio di:
 - $0 \leftrightarrow 1$
 - $+ \leftrightarrow \cdot$

Fondamenti di Informatica B/2 Gianni CONTE 13 /22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

TEOREMA DI DE MORGAN*

E' possibile dimostrare il seguente teorema:

$$\overline{X_1 + X_2 + \dots + X_n} = \overline{X_1} \cdot \overline{X_2} \cdot \dots \cdot \overline{X_n}$$

Dimostrazione per induzione:

- $\overline{X_1 + X_2} = \overline{X_1} \cdot \overline{X_2}$ verifica tabellare
- $(X_1 + X_2 + \dots + X_n) + X_{n+1} = (X_1 + X_2 + \dots + X_n) \cdot \overline{X_{n+1}}$

$$= (\overline{X_1} \cdot \overline{X_2} \cdot \dots \cdot \overline{X_n}) \cdot \overline{X_{n+1}}$$

• Versione duale $\overline{X_1 \cdot X_2 \cdot \dots \cdot X_n} = \overline{X_1} + \overline{X_2} + \dots + \overline{X_n}$

*DeMorgan (1806 - 1871)

Fondamenti di Informatica B/2 Gianni CONTE 14 /22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

PORTE LOGICHE

- Espressione logica generale

$$z = E(x_1, x_2, \dots, x_n)$$

Fondamenti di Informatica B/2 Gianni CONTE 15 /22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

PORTE LOGICHE

- Una porta logica è un blocco elementare in cui la relazione fra variabili di ingresso e variabili di uscita è descritta da funzioni booleane.

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

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

Fondamenti di Informatica B/2 Gianni CONTE 16 /22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

ELEMENTI PRIMITIVI

- Una qualunque funzione logica può essere realizzata utilizzando un numero limitato di funzioni elementari (insieme completo).
- Sono insiemi completi: NAND, NOR, (AND, NOT), (OR, NOT)
- Questa proprietà garantisce che l'uso di un limitato numero di funzioni consente di rappresentare una qualunque funzione logica.
- Utilizzando un numero limitato di tipi di porte logiche è possibile realizzare un circuito che realizza una qualsiasi funzione logica.

Fondamenti di Informatica B/2 Gianni CONTE 17 /22

UNIVERSITA' DEGLI STUDI DI PARMA
Dipartimento di Ingegneria dell'Informazione

COMPUTER ENGINEERING

FONDAMENTI DI INFORMATICA

Esercizio n. 1

La polizza n. 22 può essere stipulata se l'assicurando:

- Ha stipulato la polizza n. 19 ed è un maschio sposato
- Ha stipulato la polizza n. 19 ed è sposato e ha meno di 25 anni.
- Non ha stipulato la polizza n. 19 ed è una donna sposata.
- E' un maschio con meno di 25 anni.
- È sposato con 25 o più anni.

E' possibile semplificare questa sequenza di regole ?

Fondamenti di Informatica B/2 Gianni CONTE 18 /22



Esercizio n. 1

Le frasi precedenti possono essere trasformate:

La polizza n. 22 può essere stipulata se l'assicurando:

1. Ha stipulato la polizza n. 19 AND è un maschio AND è sposato
- OR 2. Ha stipulato la polizza n. 19 AND è sposato AND ha meno di 25 anni.
- OR 3. Non ha stipulato la polizza n. 19 AND è una donna AND è sposata.
- OR 4. E' un maschio AND ha meno di 25 anni.
- OR 5. È sposato AND ha 25 o più anni.



SOLUZIONE ESERCIZIO N. 1

A = l'assicurando è coperto dalla polizza n. 19

B = l'assicurando è sposato

C = l'assicurando è maschio

D = l'assicurando ha meno di 25 anni.

1. $A \cdot B \cdot C$
2. $A \cdot B \cdot D$
3. $\bar{A} \cdot B \cdot \bar{C}$
4. $C \cdot D$
5. $B \cdot \bar{D}$



SOLUZIONE ESERCIZIO N. 1

$$A \cdot B \cdot C + A \cdot B \cdot D + \bar{A} \cdot B \cdot \bar{C} + C \cdot D + B \cdot \bar{D} =$$

$$A \cdot B \cdot C + A \cdot B \cdot D + B \cdot \bar{D} + \bar{A} \cdot B \cdot \bar{C} + C \cdot D =$$

$$A \cdot B \cdot C + A \cdot B + B \cdot \bar{D} + \bar{A} \cdot B \cdot \bar{C} + C \cdot D =$$

$$A \cdot B + A \cdot B \cdot \bar{C} + B \cdot \bar{D} + \bar{A} \cdot B \cdot \bar{C} + C \cdot D =$$

$$A \cdot B + B \cdot \bar{C} + B \cdot \bar{D} + C \cdot D =$$

$$A \cdot B + B \cdot \bar{C} + B \cdot \bar{D} + B \cdot C \cdot D + C \cdot D =$$

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

$$A \cdot B + B + C \cdot D = B + C \cdot D$$



SOLUZIONE ESERCIZIO N. 1

A = l'assicurando ha stipulato la polizza n. 19

B = l'assicurando è sposato

C = l'assicurando è maschio

D = l'assicurando ha meno di 25 anni.

Soluzione $\Leftrightarrow B + C \cdot D$

La polizza n. 22 può essere stipulata se l'assicurando:

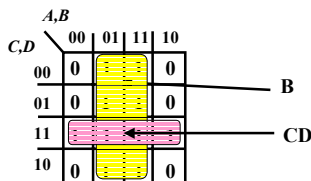
1. E' sposato
- oppure 2. E' un maschio con meno di 25 anni.



SOLUZIONE ESERCIZIO N. 1

$$A \cdot B \cdot C + A \cdot B \cdot D + \bar{A} \cdot B \cdot \bar{C} + C \cdot D + B \cdot \bar{D} =$$

$$= B + CD$$



L'uso delle mappe consente di trovare facilmente la soluzione ottima.



Esercizio n. 2

Cinque astronauti A,B,C,D,E sono pronti per una missione spaziale. Per definire la lista di coloro che partiranno è necessario soddisfare tutte le seguenti relazioni:

- a) A o B devono necessariamente essere inclusi nella lista, ma non contemporaneamente;
- b) C o E devono essere inclusi nella lista, anche contemporaneamente;
- c) qualora D sia incluso nella lista anche B deve essere incluso nella lista;
- d) A e C possono essere entrambi inclusi o entrambi esclusi dalla lista;
- e) qualora E sia incluso anche C e D lo devono essere.



SOLUZIONE ESERCIZIO N. 2

A, B, C, D, E sono variabili indipendenti che assumono valore 1 se l'astronauta corrispondente viene prescelto e valore 0 in caso opposto.

Ognuna delle 5 condizioni a, b, c, d, e deve essere verificata indipendentemente e assume valore 1 se i valori assegnati alle variabili indipendenti soddisfano ai vincoli imposti e valore 0 se non soddisfano.

- a) A o B devono necessariamente essere inclusi nella lista, ma non contemporaneamente;
 $a = a(A,B) = \overline{AB} + A\overline{B}$ (a=1 se A=1 e B=0 oppure A=0 e B=1)
- b) C o E devono essere inclusi nella lista, anche contemporaneamente;
 $b = b(C,E) = C + E$ (b=1 se C=1 oppure E = 1)



SOLUZIONE ESERCIZIO N. 2

c) qualora D sia incluso nella lista anche B deve essere incluso nella lista;
 $c = c(D,B) = DB + \overline{D}$ (c=1 se D=0 oppure D=1 e B=1)

d) A e C possono essere entrambi inclusi o entrambi esclusi dalla lista;
 $d = d(A,C) = AC + \overline{AC}$ (d=1 se A=1 e C=1 oppure A=0 e C=0)

e) qualora E sia incluso anche C e D lo devono essere.
 $e = e(E,C,D) = ECD + \overline{E}$ (e=1 se E=0 oppure E=1, C=1, D=1)



SOLUZIONE ESERCIZIO N. 2

$$\begin{aligned} Z &= (A'B + AB')(C+E)(DB+D')(AC+A'C')(ECD+E') = \\ &= (A'BC' + AB'C)(C+E)(DB+D')(ECD+E') = \\ &= (A'BC' + AB'C)(CDE + CE')(DB+B') = \\ &= (A'BC'D + AB'C)(CDE + CE') = AB'CDE + AB'CE' = \\ &= AB'C(DE+E') \end{aligned}$$



Esercizio n. 3

Progettare un circuito logico a 3 ingressi (A,B,C) e una uscita U che assuma valore 1 quando, all'ingresso, il numero degli 1 supera il numero degli 0. (Circuito di rilevazione di maggioranza)

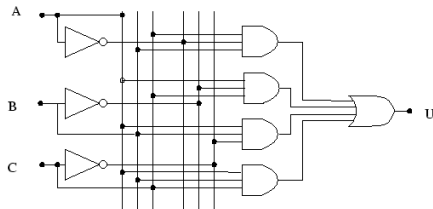
A	B	C	U
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$U = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$

$\overline{A}BC$ (row 4)
 $A\overline{B}C$ (row 6)
 $AB\overline{C}$ (row 7)
 ABC (row 8)



SOLUZIONE ESERCIZIO 3



$$U = \overline{A}BC + A\overline{B}C + AB\overline{C} + ABC$$



SOLUZIONE ESERCIZIO 3

Soluzione alternativa:

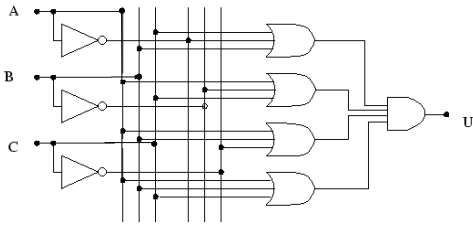
A	B	C	U
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$A+B+C$ (row 4)
 $A+B+\overline{C}$ (row 6)
 $A+\overline{B}+C$ (row 7)
 $\overline{A}+B+C$ (row 8)

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



SOLUZIONE ESERCIZIO 3



$$U = (\bar{A} + B + C) \cdot (A + \bar{B} + C) \cdot (A + B + \bar{C}) \cdot (A + B + C)$$



Esercizio n. 4

Progettare un circuito che accetti in ingresso numeri da 0 a 15.

L'uscita deve assumere valore 1 quando il numero è primo.

Soluzione:

- Rappresentare i numeri secondo una codifica binaria (4 bit).
- Scrivere la tabella della funzione.
- Inserire tante porte AND quanti sono gli 1 della funzione.
- Inserire una porta OR per generare l'uscita.