



UNIVERSITA' DEGLI STUDI DI PARMA

DIPARTIMENTO DI INGEGNERIA DELL'INFORMAZIONE

# **Fondamenti di Informatica B**

---

Lezione n.4

# Fondamenti di Informatica B

---

## Lezione n.4

- Minimizzazione logica
- Mappe di Karnaugh
- Esercizi

In questa lezione saranno analizzate alcune tecniche di minimizzazione di circuiti logici combinatori

# Minimizzazione logica

---

- E' possibile sintetizzare la stessa funzione logica con circuiti diversi:
  - minimizzazione logica significa scegliere il circuito più semplice
- Per semplificare un circuito occorre:
  - esprimere la funzione realizzata dal circuito
  - semplificare l'espressione utilizzando le proprietà dell'algebra booleana

# Minimizzazione logica

---

- Semplificare le espressioni mediante le proprietà dell'algebra booleana non garantisce una soluzione ottima:
  - le proprietà applicate dipendono dall'intuizione del progettista
- In casi semplici, tuttavia, esistono algoritmi che consentono la progettazione del circuito ottimo

# Minimizzazione logica

---

- La traduzione in circuito delle forme canoniche non genera un progetto ottimo (a costo minimo)
- Costo minimo significa minimizzare:
  - numero di porte
  - numero di ingressi delle porte (fan-in)
- Per esempio, una forma SdP minima:
  - ha il numero minimo di termini prodotto
  - non è possibile eliminare variabili da alcun termine prodotto

# Minimizzazione logica

---

- In una forma SdP, un termine prodotto è detto implicante se vale 1 per configurazioni di valori delle variabili per cui la funzione vale 1
- Un implicante è principale quando contiene il minor numero di variabili possibile, cioè, eliminando una variabile, non è più un implicante
- Un implicante è essenziale se è l'unico a coprire un determinato minterm

# Minimizzazione logica

---

- Due implicantanti si dicono adiacenti se differiscono per una variabile
- es:  $f = AB + AC\bar{D}$ 
  - $ABC$  è un implicante, perché quando  $ABC=1$ , anche  $f=1$
  - $A$  non è un implicante, perché quando  $A=1$  non sempre  $f=1$
  - $AB$  è un implicante principale
  - $ABC$  e  $ABC\bar{C}$  sono implicantanti adiacenti

# Minimizzazione logica

---

- Per minimizzazione logica si intende:
  - il calcolo di tutti gli implicant principali di una funzione
  - la scelta dell'insieme minimo di implicant principali la cui somma logica corrisponde alla funzione da minimizzare
- La minimizzazione logica è un problema intrattabile:
  - fino a 4 o 5 variabili esistono tecniche manuali
  - fino a 10 variabili esistono tecniche al calcolatore che garantiscono la soluzione ottima
  - oltre 10 variabili è necessario ricorrere a tecniche euristiche sub-ottime



# Mappe di Karnaugh

---

- Permettono di identificare ad occhio gli implicantanti adiacenti
- Permettono di selezionare gli implicantanti essenziali e l'insieme minimo di implicantanti principali
- Sono difficilmente utilizzabili per funzioni con più di 5 variabili

# Mappe di Karnaugh

<i>AB</i>	00	01	11	10	
<i>CD</i>	00	0	0	0	0
01	0	0	0	0	
11	0	0	0	0	
10	0	1	1	0	

$$f = \overline{A}BC\overline{D} + A\overline{B}C\overline{D}$$

- Ogni casella rappresenta un valore della funzione
- Esempio: funzione che vale 1 solo per le combinazioni degli ingressi:
  - (A=0, B=1, C=1, D=0)
  - (A=1, B=1, C=1, D=0)

I due termini sono implicanti adiacenti, così i corrispondenti 1 nella mappa risultano fisicamente adiacenti. Si raggruppano ad occhio i due implicanti in uno unico:

$$f = BCD$$

# Mappe di Karnaugh

- La struttura della mappa è toroidale

$CD \backslash AB$	00	01	11	10
00	0	0	1	1
01	0	0	0	0
11	0	0	0	0
10	0	0	1	1

$$f = AC\bar{D} + A\bar{C}\bar{D}$$

Implicanti adiacenti nella visione toroidale della mappa

$$f = A\bar{D}$$

# Mappe di Karnaugh

---

- I raggruppamenti possibili sono rettangolari, con lati pari a 1 o a un multiplo di due

<i>CD</i> \ <i>AB</i>	00	01	11	10
00	1	1	0	0
01	1	1	0	0
11	1	1	0	0
10	1	1	0	0

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

# Mappe di Karnaugh

	<i>AB</i>			
	00	01	11	10
<i>CD</i>				
00	1	1	1	1
01	1	1	1	1
11	1	1	1	1
10	1	0	0	1

- E' possibile agire in modo duale. Se la maggior parte dei valori assunti dalla funzione è 1, è conveniente usare la forma PdS
- Esempio: funzione che vale 0 solo per la combinazione degli ingressi:
  - (A=0, B=1, C=1, D=0)
  - (A=1, B=1, C=1, D=0)

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

# Esercizi

---

- Progettare un circuito logico che accetti in ingresso una cifra decimale (da 0 a 9) e restituisca in uscita la cifra incrementata di 1
  - se in ingresso si presenta 9, l'uscita deve essere posta a 0

# Esercizi

---

- Per rappresentare un numero decimale (0-9) sono necessari 4 bit
- Tuttavia, 4 bit possono codificare  $2^4=16$  cifre diverse, mentre ne servono solamente 10
  - alcune combinazioni delle variabili in ingresso non si possono verificare
- In corrispondenza di una combinazione di ingressi che non si può verificare, non interessa quale sia l'uscita del circuito
  - condizione di indifferenza (don't care)

# Esercizi

- Se  $x_1, x_2, x_3, x_4$  sono i bit della parola in ingresso, e  $z_1, z_2, z_3, z_4$  i bit di quella in uscita, le possibili configurazioni ingresso/uscita sono come in tabella:
  - si notino le condizioni di indifferenza

n°	INPUT				OUTPUT			
	$x_1$	$x_2$	$x_3$	$x_4$	$z_1$	$z_2$	$z_3$	$z_4$
0	0	0	0	0	0	0	0	1
1	0	0	0	1	0	0	1	0
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	1	0	0
4	0	1	0	0	0	1	0	1
5	0	1	0	1	0	1	1	0
6	0	1	1	0	0	1	1	1
7	0	1	1	1	1	0	0	0
8	1	0	0	0	1	0	0	1
9	1	0	0	1	0	0	0	0
10	1	0	1	0	-	-	-	-
11	1	0	1	1	-	-	-	-
12	1	1	0	0	-	-	-	-
13	1	1	0	1	-	-	-	-
14	1	1	1	0	-	-	-	-
15	1	1	1	1	-	-	-	-



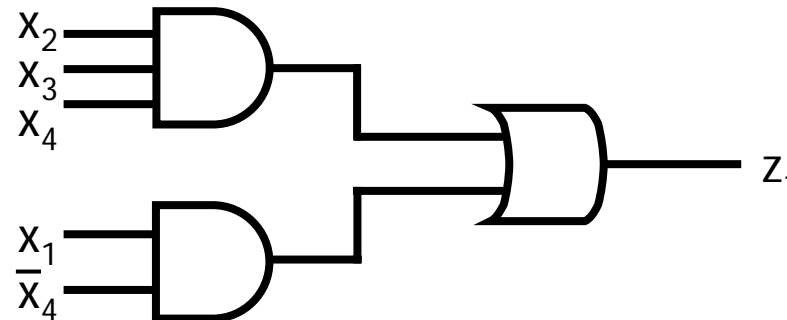
# Esercizi

- Per risolvere l'esercizio, bisogna sintetizzare quattro funzioni:  $z_1, z_2, z_3, z_4$  nelle variabili  $x_1, x_2, x_3, x_4$
- Sintesi mediante mappe di Karnaugh

–  $Z_1$ :

	$x_3x_4$				
	00	01	11	10	
$x_1x_2$	00	0	0	0	0
	01	0	0	1	0
	11	-	-	-	-
	10	1	0	-	-

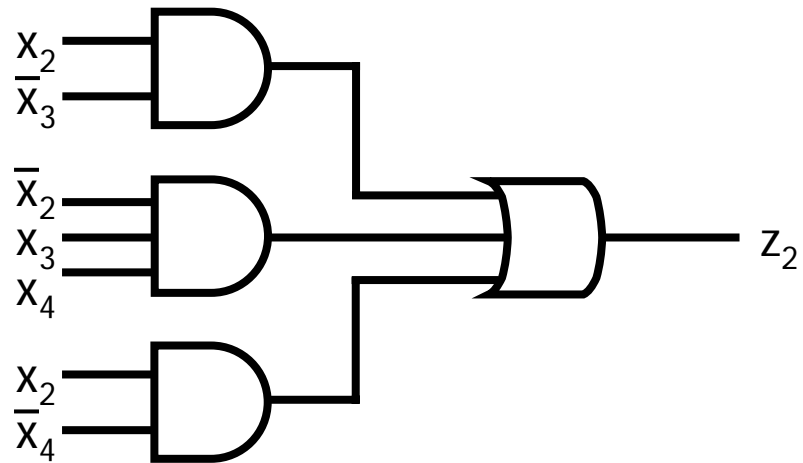
$$Z_1 = \underbrace{x_1 \overline{x_4}} + \underbrace{x_2 x_3 x_4}$$



# Esercizi

- Mappa e circuito per  $z_2$ :

$X_3X_4$	00	01	11	10
$X_1X_2$				
00	0	0	1	0
01	1	1	0	1
11	-	-	-	-
10	0	0	-	-

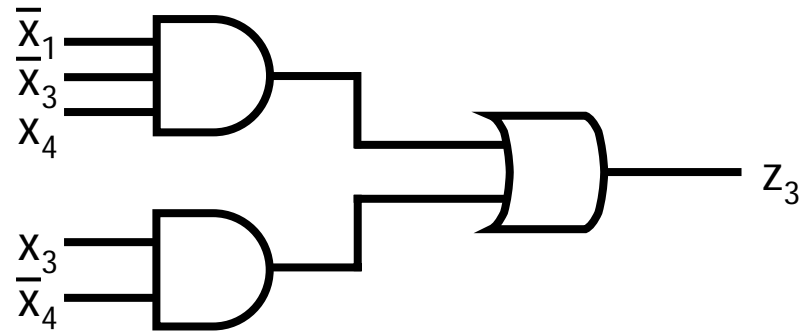


$$z_2 = \underbrace{X_2 \overline{X_4}} + \underbrace{X_2 \overline{X_3}} + \underbrace{\overline{X_2} X_3 X_4}$$

# Esercizi

- Mappa e circuito per  $z_3$ :

$X_3X_4$	00	01	11	10
$X_1X_2$ 00	0	1	0	1
01	0	1	0	1
11	-	-	-	-
10	0	0	-	-



$$z_3 = \overline{x_1} \overline{x_3} x_4 + x_3 \overline{x_4}$$

# Esercizi

---

## ■ Mappa per $z_4$ :

$X_3X_4$	00	01	11	10
$X_1X_2$ 00	1	0	0	1
01	1	0	0	1
11	-	-	-	-
10	1	0	-	-

$$z_4 = \overline{x_4}$$

- Il circuito che realizza il sistema è costituito dall'insieme dei circuiti visti
- Soluzione non necessariamente ottima
  - possono esistere dei termini comuni, che potrebbero portare ad ulteriori ottimizzazioni

# Esercizi

---

- Progettare un circuito logico che generi i comandi per un display a 7 segmenti
  - 4 variabili in ingresso
  - 7 uscite

