Architettura degli Elaboratori
5 - Sintesi di Reti Combinatorie

Ugo Dal Lago

Dipartimento di Scienze dell'Informazione
Università degli Studi di Bologna

Anno Accademico 2007/2008

Sommario

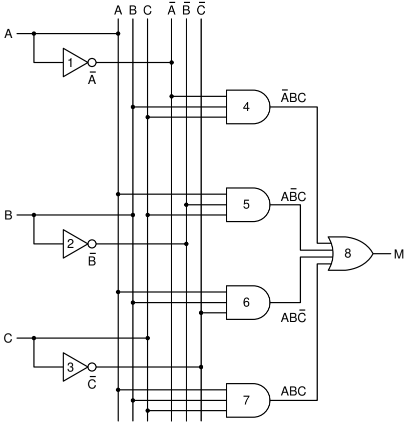
Reti Combinatorie: Sintesi

- Supponiamo di voler costruire una rete combinatoria che soddisfi certe proprietà.
- Prima di tutto: come possiamo **specificare** il comportanento che la rete combinatoria deve avere?
- Ciò che ci interessa è il suo comportamento funzionale.
  - La rete combinatoria dovrà avere un certo numero m di entrate.
  - La rete combinatoria dovrà avere un certo numero n di uscite.
  - Il suo comportamento atteso sarà poi specificato tramite n funzioni booleane in m variabili.
- Sappiamo poi che data una funzione booleana esiste sempre il modo di costruire un'espressione booleana ad essa corrispondente e, da quest'ultima, si passa facilmente ad una rete combinatoria.
- Non è però chiaro se il numero totale di porte logiche impiegate sia in qualche senso minimale.

Di Quante Porte Abbiamo Bisogno?

A	B	C	M
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)

$\Downarrow$ $\Uparrow$

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

Mintermini

- Fissato un insieme $\{A_1, \ldots, A_n\}$ di variabili booleane, un **mintermine** è un prodotto di letterali nel quale tutte le variabili appaiono esattamente una volta.
  - Ad esempio, se l'insieme di variabili di riferimento è $\{A, B, C, D\}$, allora $A\overline{B}C\overline{D}$ e $\overline{A}\,\overline{B}CD$ sono mintermini, mentre $A + BC\overline{D}$ e $\overline{A}C\overline{D}$ non lo sono.
- Ogni mintermine è in corrispondenza con un **assegnamento di verità** alle variabili.
  - Ogni mintermine ha valore 1 se si considera l'assegnamento ad esso corrispondente, mentre ha valore 0 in qualunque altro assegnamento di verità.
- Ciascun mintermine è solitamente indicato con m_j, dove j è il numero naturale rappresentato dall'assegnamento di verità corrispondente al mintermine.
- Date n variabili, esistono 2^n mintermini distinti.

Mintermini: Esempio

A	B	C	m_0	m_1	m_2	m_3	m_4	m_5	m_6	m_7		
0	0	0	1	0	0	0	0	0	0	0	$\overline{A}\,\overline{B}\,\overline{C}$	m_0
0	0	1	0	1	0	0	0	0	0	0	$\overline{A}\,\overline{B}C$	m_1
0	1	0	0	0	1	0	0	0	0	0	$\overline{A}\,B\,\overline{C}$	m_2
0	1	1	0	0	0	1	0	0	0	0	$\overline{A}\,B\,C$	m_3
1	0	0	0	0	0	0	1	0	0	0	$A\,\overline{B}\,\overline{C}$	m_4
1	0	1	0	0	0	0	0	1	0	0	$A\,\overline{B}\,C$	m_5
1	1	0	0	0	0	0	0	0	1	0	$A\,B\,\overline{C}$	m_6
1	1	1	0	0	0	0	0	0	0	1	$A\,B\,C$	m_7

Maxtermini

- Il concetto di **maxtermine** è duale a quello di mintermine.
- Fissato un insieme $\{A_1, \ldots, A_n\}$ di variabili booleane, un **maxtermine** è una somma di letterali nel quale tutte le variabili appaiono esattamente una volta.
  - Ad esempio, se l'insieme di variabili di riferimento è $\{A, B, C\}$, allora $A + \overline{B} + C$ e $\overline{A} + \overline{B} + \overline{C}$ sono maxtermini, mentre $A + B + C + \overline{D}$ e $\overline{A}B + \overline{C}$ non lo sono.
- Ogni maxtermine è in corrispondenza con un assegnamento di verità alle variabili.
  - Dualmente a ciò che succede per i mintermini, ogni maxtermine ha valore 0 se si considera l'assegnamento ad esso corrispondente, mentre ha valore 1 in qualunque altro assegnamento di verità.
- Ciascun maxtermine è solitamente indicato con M_j, dove j è il numero naturale rappresentato dall'assegnamento di verità corrispondente al maxtermine.
- Analogamente al caso dei mintermini, date n variabili, esistono 2^n maxtermini distinti.

Maxtermini: Esempio

A	B	C	M_0	M_1	M_2	M_3	M_4	M_5	M_6	M_7		
0	0	0	0	1	1	1	1	1	1	1	$A+B+C$	M_0
0	0	1	1	0	1	1	1	1	1	1	$A+B+\overline{C}$	M_1
0	1	0	1	1	0	1	1	1	1	1	$A+\overline{B}+C$	M_2
0	1	1	1	1	1	0	1	1	1	1	$A+\overline{B}+\overline{C}$	M_3
1	0	0	1	1	1	1	0	1	1	1	$\overline{A}+B+C$	M_4
1	0	1	1	1	1	1	1	0	1	1	$\overline{A}+B+\overline{C}$	M_5
1	1	0	1	1	1	1	1	1	0	1	$\overline{A}+\overline{B}+C$	M_6
1	1	1	1	1	1	1	1	1	1	0	$\overline{A}+\overline{B}+\overline{C}$	M_7

Somme di Mintermini

- Sappiamo già che ogni funzione può essere espressa come **somma di mintermini** in modo univoco.
- Se un'espressione è esprimibile come una somma $m_{i_1} + m_{i_2} + \ldots + m_{i_n}$ di mintermini, indicheremo tale espressione con $\sum m(i_1, i_2, \ldots, i_n)$.
  - Per esempio, se lavoriamo con tre variabili A, B e C, la somma di mintermini $\overline{A}\ \overline{B}\ \overline{C} + \overline{A}\ B\ \overline{C} + A\overline{B}C$, ovvero $m_0 + m_2 + m_5$ si può scrivere anche $\sum m(0, 2, 5)$.
- Se una funzione booleana si può esprimere come somma di mintermini $\sum m(i_1, i_2, \ldots, i_n)$, allora il suo complemento si può esprimere anch'essa tramite la somma di mintermini $\sum m(j_1, j_2, \ldots, j_k)$, dove gli indici $j_1, \ldots, j_k$ sono tutti e soli gli indici che non compaiono nella sequenza $i_1, i_2, \ldots, i_n$.
  - Un'espressione per il complemento della funzione descritta nell'esempio precedente è $\sum m(1, 3, 4, 6, 7)$.
- Si osservi come la somma di tutti i possibili mintermini sia equivalente alla costante 1.

Prodotti di Maxtermini

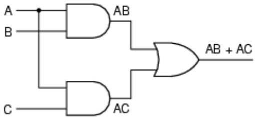
- Osserviamo innanzitutto come per ogni j l'espressione $\overline{m_j}$ sia equivalente a M_j e, viceversa, come $\overline{M_j}$ sia equivalente a m_j.
- Se una funzione booleana si può esprimere come somma di mintermini $\sum m(i_1, i_2, \dots, i_n)$, allora si può esprimere anche tremite l'espressione
$$\overline{\sum m(j_1, j_2, \dots, j_k)},$$
dove gli indici $j_1, \dots, j_k$ sono tutti e soli gli indici che non compaiono nella sequenza $i_1, i_2, \dots, i_n$.
- Ma a questo punto osserviamo come
$$\begin{aligned} \overline{\sum m(j_1, j_2, \dots, j_k)} &= \overline{m_{j_1} + m_{j_2} + \dots + m_{j_k}} \\ &= \overline{m_{j_1}} \cdot \overline{m_{j_2}} \cdot \dots \cdot \overline{m_{j_k}} \\ &= M_{j_1} \cdot M_{j_2} \cdot \dots \cdot M_{j_k}. \end{aligned}$$
Il prodotto ottenuto, detto **prodotto di maxtermini**, è indicato spesso com $\prod M(j_1, j_2, \dots, j_k)$ ed è una rappresentazione canonica della funzione di partenza.

Somme di Prodotti

- Le somme di mintermini si possono generalizzare a **somme di prodotti** di letterali che non siano necessariamente somme di mintermini.
- Una somma di prodotti "semplice" per una certa funzione può essere ottenuta manipolando algebricamente la relativa somma di mintermini:
  - Ad esempio:

$$
\begin{aligned}
A\overline{B}C + AB\overline{C} + ABC &= A\overline{B}C + AB(\overline{C} + C) \\
&= A\overline{B}C + AB \\
&= A(\overline{B}C + B) \\
&= A((\overline{B} + B)(C + B)) \\
&= A(C + B) = AC + AB
\end{aligned}
$$

- Il circuito logico relativo ad una somma di prodotti consiste in un gruppo di porte AND seguite da una porta OR.
  - Ad esempio:

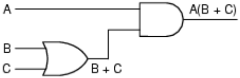


Prodotti Di Somme

- Dualmente alle somme di prodotti, possiamo costruire **prodotti di somme** di letterali che non siano necessariamente prodotti di maxtermini.
- Anche in questo caso la manipolazione algebrica può aiutarci a ottenere un prodotto di somme semplice a partire da un prodotto di maxtermini:
  - Ad esempio:

$$\begin{aligned}
& (A+B+C)(A+B+\overline{C})(A+\overline{B}+C)(A+\overline{B}+\overline{C})(\overline{A}+B+C) \\
= \; & (A+B)(C+\overline{C})(A+\overline{B})(C+\overline{C})(\overline{A}+B+C) \\
= \; & A(B+\overline{B})(\overline{A}+B+C) \\
= \; & A\overline{A}+A(B+C) \\
= \; & A(B+C)
\end{aligned}$$

- Il circuito logico relativo ad un prodotto di somme consiste in un gruppo di porte OR seguite da una porta AND.
  - Ad esempio:



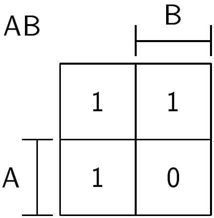
Implementazioni a Due Livelli

- Nel caso dei circuiti relativi a somme di prodotti e a prodotti di somme si parla di **implementazioni a due livelli**.
- In generale, potremmo avere a che fare con circuiti che non possono essere espressi su due livelli, ma è chiaro che le implementazioni a due livelli siano comunque **universiali**.
- Decidere se utilizzare un'implementazione a due livelli oppure multilivello (tre o più livelli) è una decisione critica a livello progettuale.
- Nel seguito descriveremo una tecnica di semplificazione per le implementazioni a due livelli.
  - Si può comunque utilizzare la semplificazione algebrica, come già anticipato.
  - Tale tecnica, però, risulta di difficile implementazione. In particolare, è difficile determinare se è stata ottenuta l'espressione più semplice.
  - Verranno invece utilizzate le **mappe di Karnaugh**, che offrono una procedura diretta per la semplificazione di funzioni dipendenti da al più quattro variabili.

Mappe di Karnaugh

- In una **mappa di Karnaugh** si può specificare la tabella di verità di una funzione booleana.
- Ad ogni casella della mappa di Karnaugh corrisponde un assegnamento di verità
- A differenza di ciò che succede con le tabelle di verità, nelle mappe di Karnaugh, l'adiacenza fisica tra le celle corrisponde all'adiacenza logica dei corrispondenti assegnamenti di verità
  - In altre parole, celle tra loro adiacenti corrispondono ad assegnamenti di verità alle variabili che differiscano nel valore assegnato ad una singola variabile.
- Esistono alcune piccole differenze tra i vari stili di rappresentazione delle mappe di Karnaugh.
  - In questo senso, è bene essere sufficientemente flessibili ed essere in grado di lavorare con qualunque stile.
  - Nei lucidi, utilizzeremo in particoare uno stile leggermente diverso da quello utilizzato nel libro.

Mappa di Karnaugh a Due Variabili: Esempio



A	B	F
0	0	1
0	1	1
1	0	1
1	1	0

Mappa di Karnaugh a Tre Variabili: Esempio

ABC

		C	C, A	A
	1	1	1	1
B	0	0	1	0

A	B	C	F
0	0	0	1
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

Mappa di Karnaugh a Quattro Variabili: Esempio

ABCD

B

D

1	1	1	0
1	0	0	1
1	0	0	1
0	1	1	0

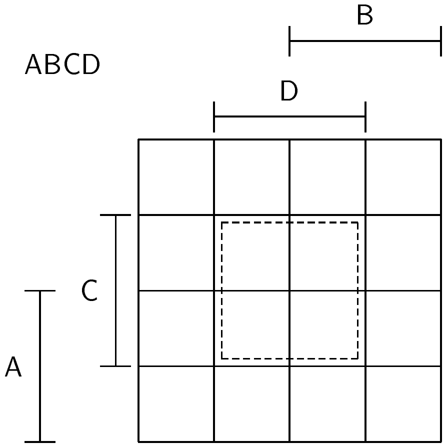
C

A

Raggruppamenti

- Sappiamo che ogni cella di una mappa di Karnaugh corrisponde ad un assegnamento di valore di verità alle variabili booleane.
  - ...e quindi a un mintermine e a un maxtermine.
- Consideriamo ora una mappa di Karnaugh a n variabili e scegliamo $k \leq n$ tra quelle in gioco.
- Supponiamo che le k variabili scelte assumano ognuna un valore binario fissato a priori, mentre le altre $n - k$ possano prendere qualunque valore binario.
- In questo modo abbiamo implicitamente definito un insieme di assegnamenti di verità, cui corrisponderà una regione (insieme di celle) della mappa di Karnaugh, che chiamiamo **raggruppamento** (o **rettangolo**, o **accoppiamento**).
- In analogia con quanto avviene nei mintermini e nei maxtermini, possiamo associare ad ogni raggruppamento:
  - Un **prodotto** di letterali che vale 1 solo negli assegnamenti del raggruppamento.
  - Una **somma** di letterali, che vale 0 solo negli assegnamenti del raggruppamento.

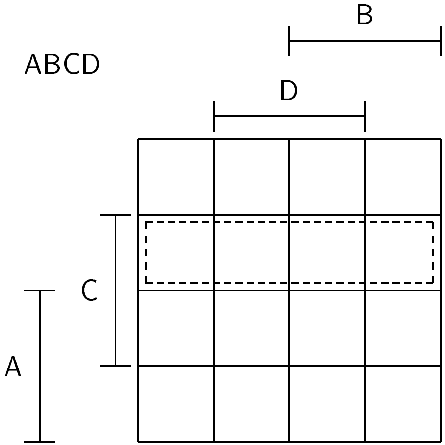
Esempio di Raggruppamento



Prodotto Corrispondente: CD

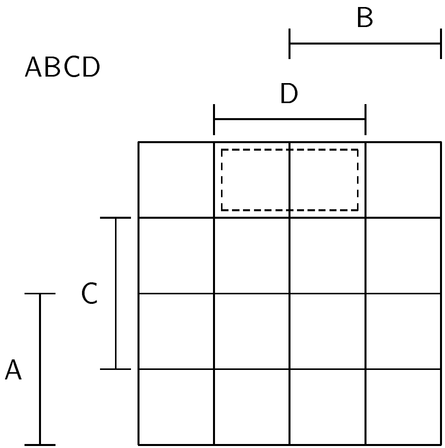
Somma Corrispondente: $\overline{C} + \overline{D}$

Esempio di Raggruppamento



Prodotto Corrispondente: $\overline{A}C$
Somma Corrispondente: $A + \overline{C}$

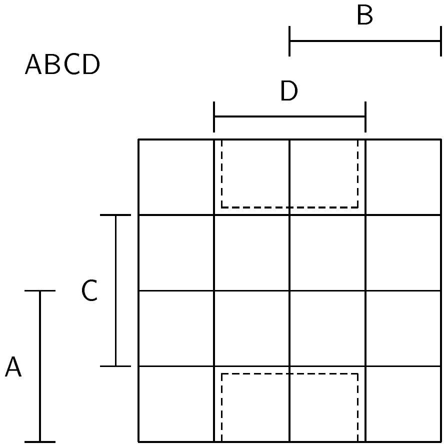
Esempio di Raggruppamento



Prodotto Corrispondente: $\overline{A}\,\overline{C}D$

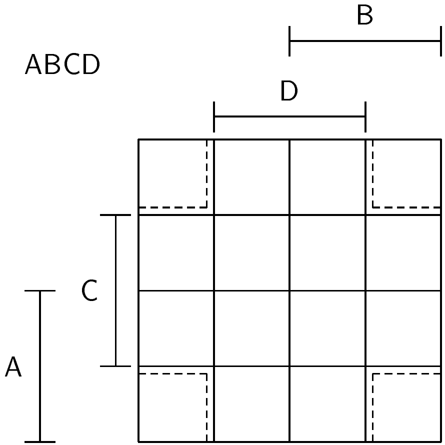
Somma Corrispondente: $A + C + \overline{D}$

Esempio di Raggruppamento



Prodotto Corrispondente: $\overline{C}D$
Somma Corrispondente: $C + \overline{D}$

Esempio di Raggruppamento



Prodotto Corrispondente: $\overline{C}\ \overline{D}$

Somma Corrispondente: $C + D$

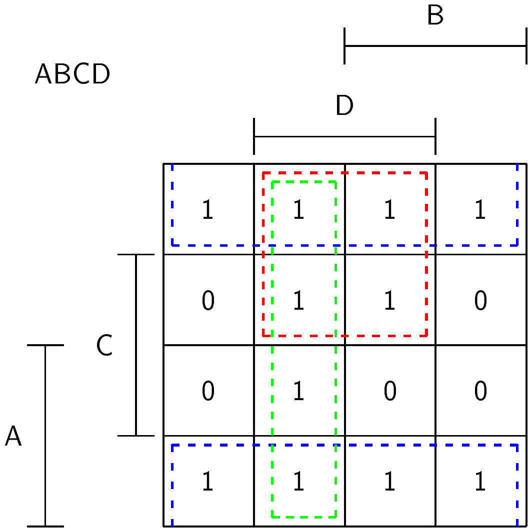
Semplificazione: Procedura Informale

- Supponiamo di avere a disposizione una mappa di Karnaugh relativa ad una funzione booleana.
- Supponiamo di riuscire a trovare una **copertura** per tutti gli 1 presenti nella mappa, ossia un insieme di raggruppamenti tali che:
  - Tutti gli 1 presenti nella mappa compaiano all'interno di almeno un raggruppamento nell'insieme.
  - Nessuno raggruppamento contenga una cella valorizzata a 0.
- In tal modo, abbiamo implicitamente trovato una somma di prodotti per la funzione di partenza: basterà considerare la somma dei prodotti corrispondenti a tutti i raggruppamenti dell'insieme.
- Non è però detto che la copertura che abbiamo costruito sia **ottimale**, ossia contenga un numero minimo di raggruppamenti.
- Una copertura per tutti gli 1 si può però sempre determinare:
  - Basta considerare ogni cella valorizzata a 1 come un raggruppamento, ottenendo così una somma di mintermini.

Semplificazione: Procedura Informale

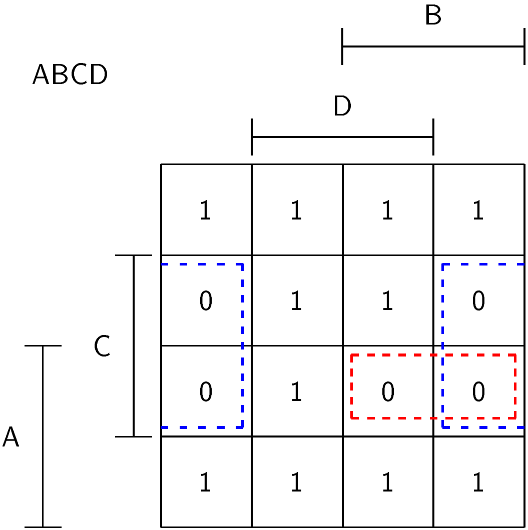
- La procedura che abbiamo appena descritto può essere facilmente generalizzata al caso duale, quello in cui l'espressione prodotta è un **prodotto di somme** di letterali.
- Siamo in questo caso interessati ad una copertura per tutti gli 0 presenti nella mappa, ossia un insieme di raggruppamenti tali che:
  - Tutti gli 0 presenti nella mappa compaiano all'interno di almeno un raggruppamento
  - Nessuno raggruppamento contenga una cella valorizzata a 1.
- Per ottenere un prodotto di somme corrispondente alla funzione basterà in questo caso considerare il prodotto corrispondente a tutti i raggruppamenti dell'insieme (ognuno dei quali, sappiamo, corrisponde ad una somma).
- Anche in questo caso la copertura prodotta può non essere ottimale, ma si può sempre costruire una copertura corrispondente ad un prodotto di maxtermini.

Esempio di Semplificazione



Somma di Prodotti: $\overline{C} + \overline{B}D + \overline{A}D$

Esempio di Semplificazione



Prodotto di Somme: $(\overline{C} + D)(\overline{A} + \overline{B} + \overline{C})$

Implicanti

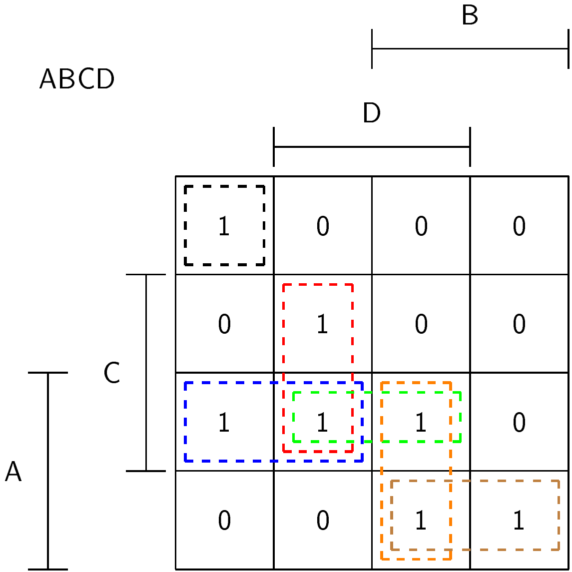
- A questo punto, abbiamo bisogno di dare un certo numero di nozioni. Supponiamo data una funzione booleana F.
- Un **implicante** è un prodotto di letterali E tale che la funzione F vale 1 ogniqualvolta E vale 1.
  - Possiamo pensare ad un implicante come ad un raggruppamento della mappa di Karnaugh di F che contenenta solo 1.
- Un **primo implicante** è un implicante E tale che la rimozione di un letterale da E dà luogo ad un prodotto che non è un implicante.
  - Dal punto di vista delle mappe di Karnaugh, un primo implicante è un implicante che non è interamente contenuto in nessun altro implicante.
- Se un certo assegnamento di verità rende vera F ed esiste un unico primo implicante E che è vero in corrispondenza di tale assegnamento, allora E è un **primo implicante essenziale**.
  - Se esiste una cella della mappa di Karnaugh di F che è valorizzata a 1 e che è inclusa in un solo primo implicante, tale raggruppamento è un primo implicante essenziale.

Semplificazione in Somma di Prodotti: Procedura Formale

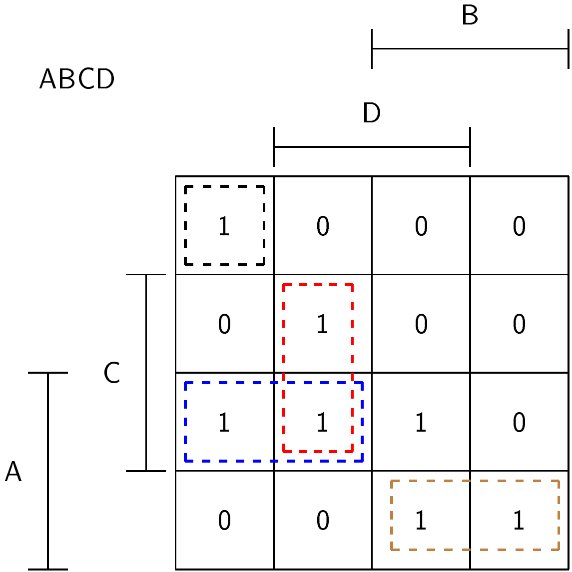
- Esiste una procedura formale per la semplificazione di una mappa di Karnaugh in una somma di prodotti:

1. Prima di tutto, occorre determinare **tutti i primi implicanti**.
   - Nella mappa di Karnaugh, basterà determinare tutti i raggruppamenti contenenti soltanto celle valorizzate con 1 e che non sono inclusi in altri raggruppamenti dello stesso tipo.
2. Occorre poi **isolare i primi implicanti essenziali**.
   - Nella mappa di Karnaugh, bisognerà capire quali tra i primi implicanti contengono soltanto celle contentute in altri primi implicanti. Tali implicanti non sono essenziali e vanno scartati.
3. L'**espressione semplificata** si ottiene come somma logica dei prodotti corrispondenti ai primi implicanti essenziali e ad eventuali primi implicanti non essenziali, ma necessari ad ottenere una copertura.
   - Occorre assicurarsi che ogni primo implicante non essenziale incluso nella somma contenga una cella valorizzata a 1 non contenuta in nessun altro primo implicante incluso nella somma.

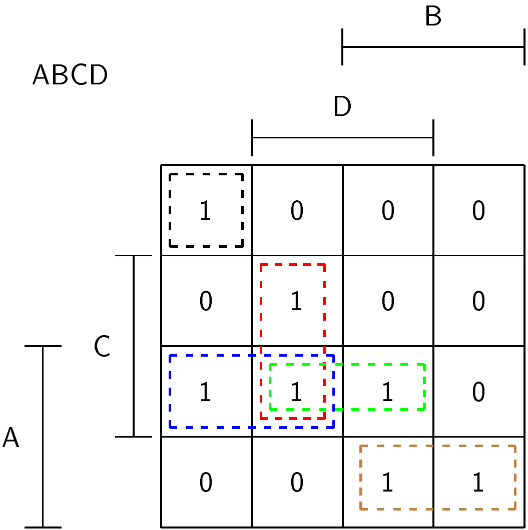
Esempio: Primi Implicanti



Esempio: Primi Implicanti Essenziali

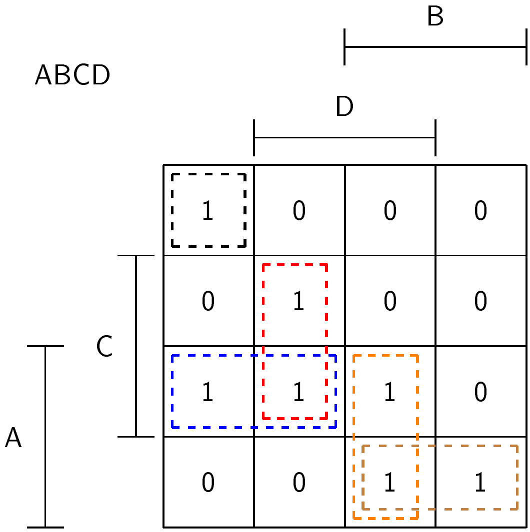


Esempio: Prima Possibilità



$$\overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + A\overline{B}C + \overline{B}CD + AB\overline{C} + ACD$$

Esempio: Seconda Possibilità



$$\overline{A}\,\overline{B}\,\overline{C}\,\overline{D} + A\overline{B}C + \overline{B}CD + AB\overline{C} + ABD$$

Semplificazione in Prodotto di Somme: Procedura Formale

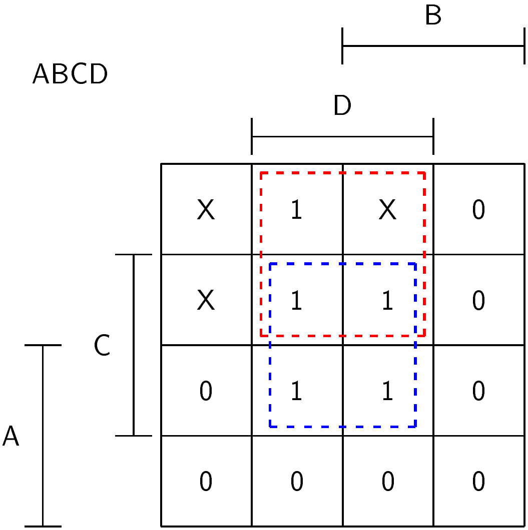
- Se occorre costruire un prodotto di somme anziché una somma di prodotti, si possono sfruttare al solito le proprietà dell'algebra di Boole.
- Data una funzione booleana F, si considera il suo complemento $\overline{F}$.
  - Una mappa di Karnaugh per $\overline{F}$ è facilmente ottenibile da una mappa per la funzione di partenza: basta invertire ciascun bit.
- Si applica poi alla mappa di Karnaugh per $\overline{F}$ la procedura di semplificazione, ottenendo una somma di prodotti.
  - Non è necessario modificare la mappa di Karnaugh di F.
- Applicando poi le proprietà di De Morgan e di doppia negazione, sarà semplice trasformare l'espressione ottenuta in un prodotto di somme per $\overline{F}$.
  - Ad esempio, se $\overline{F} = AB + \overline{B}C\overline{D}$, allora

$$
\begin{aligned}
F &= \overline{\overline{F}} = \overline{AB + \overline{B}C\overline{D}} \\
&= \overline{AB} \cdot \overline{\overline{B}C\overline{D}} \\
&= (\overline{A} + \overline{B})(\overline{\overline{B}} + \overline{C} + \overline{\overline{D}}) \\
&= (\overline{A} + \overline{B})(B + \overline{C} + D)
\end{aligned}
$$

Condizioni di Non-Specificazione

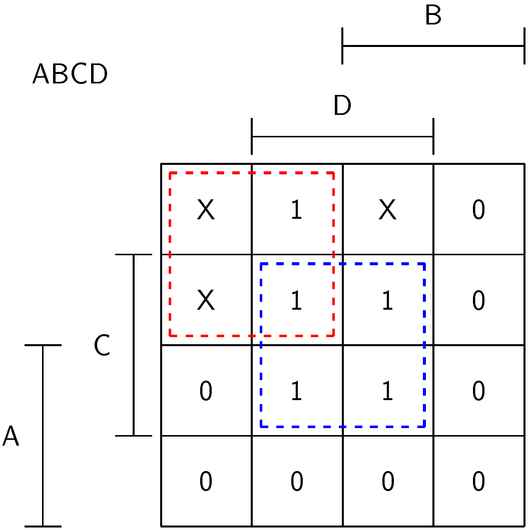
- Spesso succede che una funzione booleana, vista come specifica di una rete combinatoria, **non sia completamente specificata**, ossia che ad alcuni assegnamenti di valori di verità alle variabili non corrisponda nessun valore di verità.
- Ciò si può verificare in almeno due situazioni:
  - Alcuni possibili input **non si presentano mai in pratica** e, di conseguenza, non è rilevante il valore dell'output.
  - La rete combinatoria su alcuni input **può produrre indifferentemente** uno dei due valori in output.
- Il fatto che una funzione non è specificata in corrispondenza di un assegnamento di valori di verità alle variabili è indicato da una X nella corrispondente cella della relativa mappa di Karnaugh.
- Nella fase di semplificazione, poi, le X possono assumere indifferentemente i valori 0 e 1. La scelta dipende da quale delle due possibilità genera l'espressione più semplice.
  - L'espressione ottenuta assumerà valori ben precisi anche in corrispondenza agli assegnamenti cui la funzione di partenza non faceva corrispondere un valore di verità.

Condizioni di Non-Specificazione: Esempio



Somma di Prodotti: $\overline{A}D + CD$

Condizioni di Non-Specificazione: Esempio



Somma di Prodotti: $\overline{A}\,\overline{B} + CD$