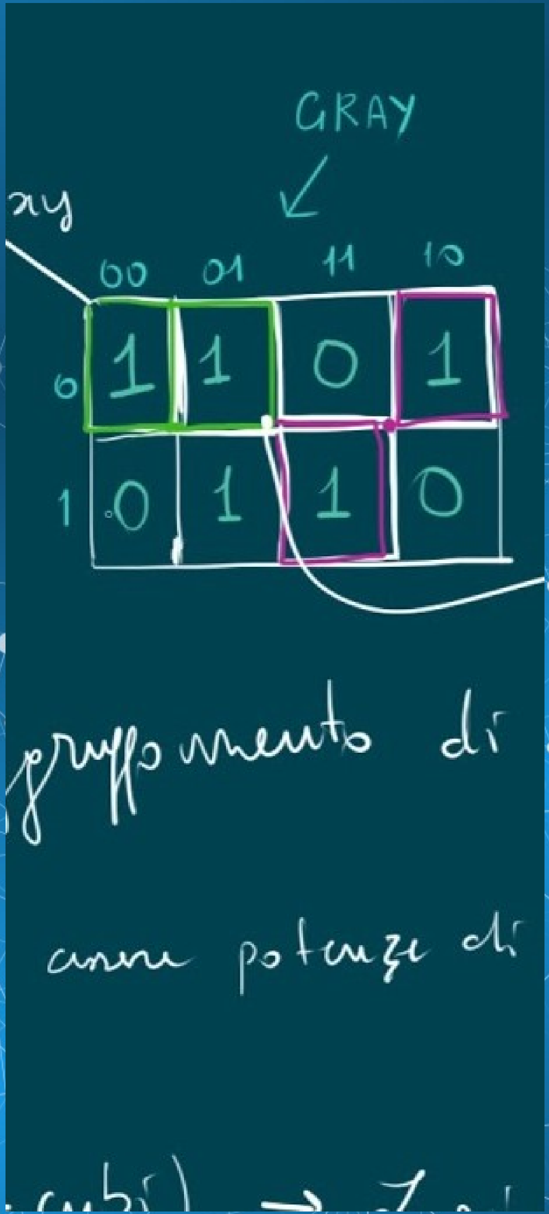
ISLab Intelligent Systems Laboratory

UNIVERSITÀ DEGLI STUDI DI BARI ALDO MORO

Espressioni logiche booleane e mappe di Karnaugh

Prof. Giuseppe Pirlo | Prof. Donato Impedovo | Dott. Stefano Galantucci

Architettura degli Elaboratori e Sistemi Operativi @ Corso di Laurea in Informatica



Espressioni logiche

Un’espressione logica (o funzione logica) è una qualsiasi espressione composta da variabili logiche, le quali possono assumere solo valori booleani (vero/falso) e da connettivi logici (operatori logici)

Un esempio di espressione logica è:

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

La quale è una funzione logica in 4 variabili (A, B, C e D)

Proprietà dell'algebra di Boole

Nome	Forma AND	Forma OR
Elemento neutro	$1A = A$	$0 + A = A$
Annullamento	$0A = 0$	$1 + A = 1$
Idempotenza	$AA = A$	$A + A = A$
Complementazione	$A\bar{A} = 0$	$A + \bar{A} = 1$
Proprietà commutativa	$AB = BA$	$A + B = B + A$
Proprietà associativa	$(AB)C = A(BC)$	$(A + B) + C = A + (B + C)$
Proprietà distributiva	$A + BC = (A + B)(A + C)$	$A(B + C) = AB + AC$
Legge di De Morgan	$\overline{AB} = \bar{A} + \bar{B}$	$\overline{A + B} = \bar{A}\bar{B}$
Prima legge di assorbimento	$A(A + B) = A$	$A + AB = A$
Seconda legge di assorbimento	$A(\bar{A} + B) = AB$	$A + \bar{A}B = A + B$

Espressioni logiche

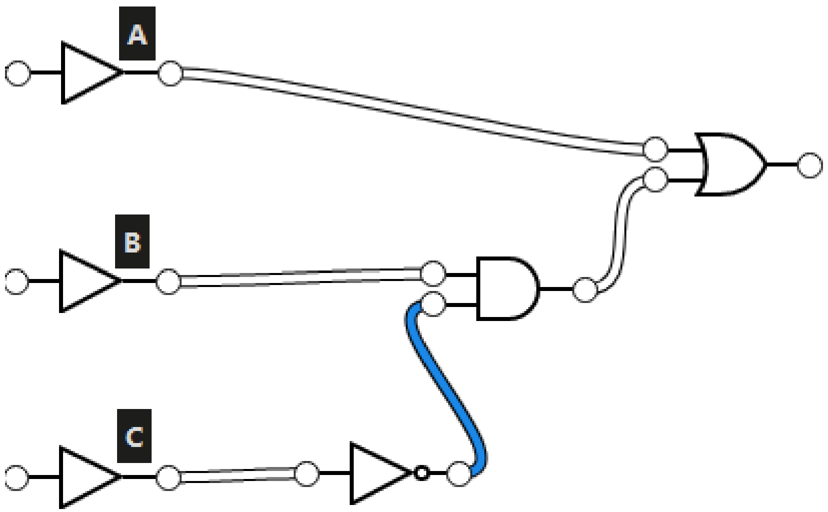
Un’espressione logica può essere rappresentata in modo esaustivo utilizzando la **tavola di verità**

Ciascuna espressione logica ha un suo equivalente **circuito logico**

Es.:

$A + B\bar{C}$

A	B	C	$\bar{C}$	$B\bar{C}$	$A + B\bar{C}$
0	0	0	1	0	**0**
0	0	1	0	0	**0**
0	1	0	1	1	**1**
0	1	1	0	0	**0**
1	0	0	1	0	**1**
1	0	1	0	0	**1**
1	1	0	1	1	**1**
1	1	1	0	0	**1**



Forma normale disgiuntiva e congiuntiva

Un'espressione logica è detta in **forma normale disgiuntiva** (DNF – *Disjunctive Normal Form*, detta anche *prima forma canonica*) se è una disgiunzione di clausole, dove ciascuna clausola è una congiunzione di letterali

Es:

$$AB + B\bar{C} + A\bar{B}C$$

Un'espressione logica è detta in **forma normale congiuntiva** (CNF – *Conjunctive Normal Form*, detta anche *seconda forma canonica*) se è una congiunzione di clausole, dove ciascuna clausola è una disgiunzione di letterali

Es:

$$(A + B)(B + \bar{C})(A + \bar{B} + C)$$

Una qualsiasi espressione logica può essere tradotta in DNF o in CNF

Ricostruire un’espressione logica

E’ immediato, avendo a disposizione una funzione logica arbitraria, ricavarne la tavola di verità e il circuito logico corrispondente

Partendo da un circuito logico è immediato ricostruire l’espressione corrispondente a tale circuito

Ma, avendo a disposizione la tavola di verità di una funzione logica, come posso ricostruire l’espressione logica corrispondente?

Per raggiungere tale scopo si utilizzano diversi strumenti, tra cui la **riscrittura in somma di mintermini**, la **riscrittura in prodotto di maxtermini**, il **Teorema di Shannon** e le **mappe di Karnaugh**

Mintermini e maxtermini

La procedura di **riscrittura in somma di mintermini** (o *somma di prodotti*) consente di ricavare l’espressione logica di una funzione booleana qualsiasi. L’espressione risultante sarà espressa, appunto, come somma di mintermini, ovvero sarà in **forma normale disgiuntiva**

Un **mintermine** (o *prodotto canonico*) è una funzione booleana che assume il valore vero in corrispondenza di un’unica configurazione di variabili booleane indipendenti

In maniera perfettamente speculare, si definisce **maxtermine** (o *somma canonica*) una funzione booleana che assume il valore falso in corrispondenza di un’unica configurazione di variabili booleane indipendenti

Allo stesso modo, quindi, sarà possibile riscrivere una funzione booleana qualsiasi quale **prodotto di maxtermini** (o *prodotto di somme*), il cui risultato sarà in **forma normale congiuntiva**

Le due scritture si equivalgono

Mintermini e maxtermini

Per ciascuna combinazione di input della tavola di verità è quindi possibile ricavare un mintermine e un maxtermine

A	B	C	Mintermine	Maxtermine
0	0	0	$\bar{A}\bar{B}\bar{C}$	$A+B+C$
0	0	1	$\bar{A}\bar{B}C$	$A+B+\bar{C}$
0	1	0	$\bar{A}B\bar{C}$	$A+\bar{B}+C$
0	1	1	$\bar{A}BC$	$A+\bar{B}+\bar{C}$
1	0	0	$A\bar{B}\bar{C}$	$\bar{A}+B+C$
1	0	1	$A\bar{B}C$	$\bar{A}+B+\bar{C}$
1	1	0	$AB\bar{C}$	$\bar{A}+\bar{B}+C$
1	1	1	ABC	$\bar{A}+\bar{B}+\bar{C}$

Ciascun mintermine si ottiene come prodotto delle singole variabili booleane, associando:

- La variabile booleana quando il valore è 1
- La variabile booleana negata quando il valore è 0

Ciascun maxtermine si ottiene come somma delle singole variabili booleane, associando:

- La variabile booleana negata quando il valore è 1
- La variabile booleana quando il valore è 0

Ciascun mintermine è la negazione logica del relativo maxtermine (e viceversa) a seguito dell'applicazione delle leggi di De Morgan

Mintermini e maxtermini

È facile verificare che ciascun mintermine e ciascun maxtermine rispetta la definizione:

Mintermine = *assume il valore vero in corrispondenza di un'unica configurazione di variabili booleane indipendenti*

Maxtermine = *assume il valore falso in corrispondenza di un'unica configurazione di variabili booleane indipendenti*

Ad esempio, la configurazione A=0, B=1, C=0, D=1 ha:

- per mintermine $\bar{A}B\bar{C}D$
- per maxtermine $A + \bar{B} + C + \bar{D}$

Infatti, in corrispondenza della configurazione A=0, B=1, C=0, D=1, si ha:

- mintermine: $\bar{A}B\bar{C}D = \bar{0}1\bar{0}1 = 1111 = 1$; in tutte le altre configurazioni restituirà 0
- maxtermine: $A + \bar{B} + C + \bar{D} = 0 + \bar{1} + 0 + \bar{1} = 0 + 0 + 0 + 0 = 0$; in tutte le altre configurazioni restituirà 1

Somma di mintermini

Una qualsiasi funzione booleana può essere scritta in **somma di mintermini** sommando tutti i mintermini delle combinazioni dove la funzione assume valore 1

Ad esempio:

A	B	C	f	
0	0	0	**1**	$\longrightarrow \bar{A}\bar{B}\bar{C}$
0	0	1	0	
0	1	0	**1**	$\longrightarrow \bar{A}B\bar{C}$
0	1	1	**1**	$\longrightarrow \bar{A}BC$
1	0	0	0	
1	0	1	0	
1	1	0	**1**	$\longrightarrow AB\bar{C}$
1	1	1	**1**	$\longrightarrow ABC$

Quindi

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

Prodotto di maxtermini

Una qualsiasi funzione booleana può essere scritta in **prodotto di maxtermini** moltiplicando tutti i maxtermini delle combinazioni dove la funzione assume valore 0

Ad esempio:

A	B	C	f	
0	0	0	1	
0	0	1	**0**	$\rightarrow A + B + \bar{C}$
0	1	0	1	
0	1	1	1	
1	0	0	**0**	$\rightarrow \bar{A} + B + C$
1	0	1	**0**	$\rightarrow \bar{A} + B + \bar{C}$
1	1	0	1	
1	1	1	1	

Quindi

$$\boldsymbol{f = (A + B + \bar{C})(\bar{A} + B + C)(\bar{A} + B + \bar{C})}$$

Mintermini e maxtermini

La stessa funzione può essere quindi scritta, equivalentemente, come somma di mintermini o prodotto di maxtermini

Generalmente, **la scrittura in somma di mintermini è preferita**

L'espressione risultante, però, **non è detto che sia minima**, pertanto si deve provare a semplificarla attraverso l'uso delle proprietà dell'algebra di Boole, mediante, quindi, dei passaggi algebrici

Semplificare l'espressione consente dei vantaggi:

- gestione algebrica più semplice (meno porte = circuito più semplice)
- in termini di velocità (meno porte = circuito più veloce)
- in termini di costi (meno porte = meno spese)

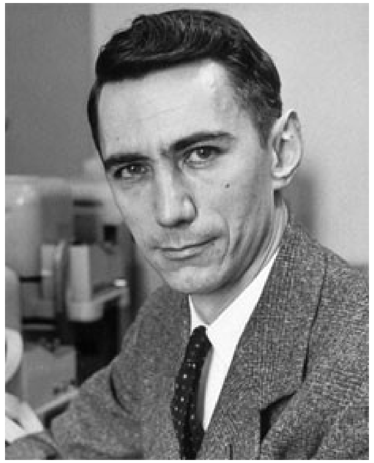
Ad esempio, l'espressione di prima $f = \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + \bar{A}BC + AB\bar{C} + ABC$ richiede 5 porte AND a tre vie, 1 porta OR a 5 vie e 7 porte NOT

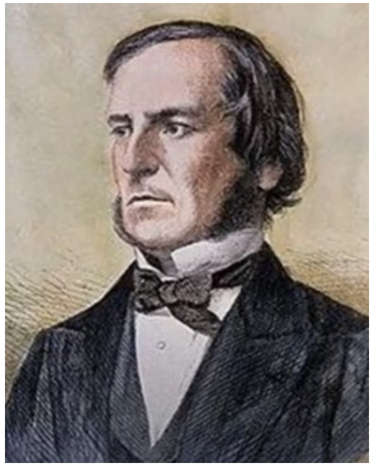
Se fossero disponibili solo porte a due vie, servirebbero allora 10 porte AND, 4 porte OR e 7 porte NOT

Semplificazione dell'espressione

$f = \bar{A}\bar{B}\bar{C} + \underbrace{\bar{A}B\bar{C} + \bar{A}BC} + \underbrace{AB\bar{C} + ABC} =$	Applico proprietà distributiva forma OR (al contrario, ovvero il cosiddetto *mettere a fattor comune*)
$= \bar{A}\bar{B}\bar{C} + \bar{A}B(\bar{C} + C) + AB(\bar{C} + C) =$ $= \bar{A}\bar{B}\bar{C} + \bar{A}B\underbrace{(\bar{C} + C)} + AB\underbrace{(\bar{C} + C)} =$	Applico complementazione forma OR
$= \bar{A}\bar{B}\bar{C} + \bar{A}B1 + AB1 =$ $= \bar{A}\bar{B}\bar{C} + \underbrace{\bar{A}B1} + \underbrace{AB1} =$	1 è elemento neutro per l'AND
$= \bar{A}\bar{B}\bar{C} + \bar{A}B + AB =$ $= \bar{A}\bar{B}\bar{C} + \underbrace{\bar{A}B + AB} =$	Applico proprietà distributiva forma OR (al contrario)
$= \bar{A}\bar{B}\bar{C} + B(A + \bar{A}) =$ $= \bar{A}\bar{B}\bar{C} + B\underbrace{(A + \bar{A})} =$	Applico complementazione forma OR
$= \bar{A}\bar{B}\bar{C} + B1 =$ $= \bar{A}\bar{B}\bar{C} + \underbrace{B1} =$	1 è elemento neutro per l'AND
$= \bar{A}\bar{B}\bar{C} + B =$ $= \underbrace{\bar{A}\bar{B}\bar{C} + B} =$	Applico seconda legge di assorbimento forma OR
$= \bar{A}\bar{C} + B$	

Teorema di Shannon


Claude Shannon


George Boole

Il **teorema di Shannon*** consente di scomporre una funzione booleana complessa in funzioni più semplici

Data una funzione booleana f in n variabili $v_1, v_2, \ldots, v_n$, si ha che:

$$f(v_1, v_2, \ldots, v_n) = v_1 \cdot f(1, v_2, \ldots, v_n) + \overline{v_1} \cdot f(0, v_2, \ldots, v_n)$$

Nel caso sopracitato il teorema è applicato rispetto alla variabile v_1, ma può essere applicato a una qualsiasi delle variabili $v_1, v_2, \ldots, v_n$

La funzione complessa di partenza in n variabili viene trasformata in una somma di due prodotti, aventi ciascuno una funzione in $n-1$ variabili (in quanto la variabile su cui è applicato il Teorema di Shannon è già avvalorata)

** è attribuito a Shannon nonostante sia stato enunciato prima da George Boole*

Teorema di Shannon

Un esempio pratico di applicazione del teorema di Shannon è il seguente.

Sia f una funzione delle variabili A, B, C, avente la seguente tavola di verità:

A	B	C	f
0	0	0	**1**
0	0	1	**0**
0	1	0	**1**
0	1	1	**0**
1	0	0	**0**
1	0	1	**1**
1	1	0	**1**
1	1	1	**1**

essa può essere riscritta in due funzioni più semplici, separando sulla base di una variabile

Teorema di Shannon

Ad esempio, scegliendo di separare sulla base della variabile B:

A	B	C	f
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$f(A, B, C)$

A	B	C	f
0	**0**	0	1
0	**0**	1	0
1	**0**	0	0
1	**0**	1	1

$f(A, 0, C)$

A	B	C	f
0	1	0	1
0	1	1	0
1	1	0	1
1	1	1	1

$f(A, 1, C)$

Teorema di Shannon

ISLab

Si avrà allora che

A	B	C	f
0	0	0	1
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

$f(A, B, C)$

$= B$

A	B	C	f
0	1	0	1
0	1	1	0
1	1	0	1
1	1	1	1

$f(A, 1, C)$

$+\bar{B}$

A	B	C	f
0	0	0	1
0	0	1	0
1	0	0	0
1	0	1	1

$f(A, 0, C)$

$$f(A, B, C) = B \cdot f(A, 1, C) + \bar{B} \cdot f(A, 0, C)$$

Teorema di Shannon

Il suo funzionamento è banale:

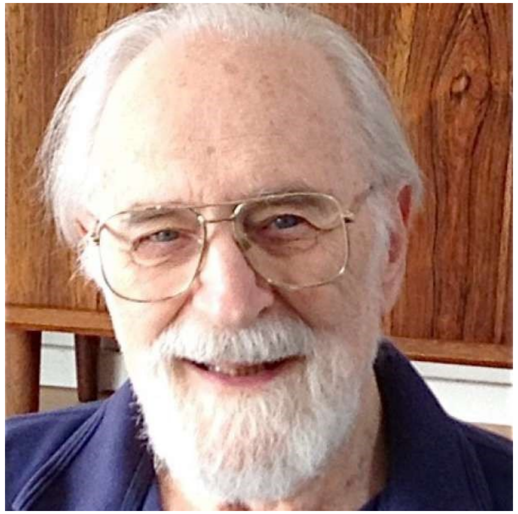
se $B = 0$

$$\begin{aligned} f(A,B,C) &= B \cdot f(A,1,C) + \bar{B} \cdot f(A,0,C) = \\ &= 0 \cdot f(A,1,C) + \bar{0} \cdot f(A,0,C) = \\ &= 0 \cdot f(A,1,C) + 1 \cdot f(A,0,C) = \\ &= 0 \qquad + f(A,0,C) = \\ &= \boldsymbol{f(A,0,C)} \end{aligned}$$

se $B = 1$

$$\begin{aligned} f(A,B,C) &= B \cdot f(A,1,C) + \bar{B} \cdot f(A,0,C) = \\ &= 1 \cdot f(A,1,C) + \bar{1} \cdot f(A,0,C) = \\ &= 1 \cdot f(A,1,C) + 0 \cdot f(A,0,C) = \\ &= f(A,1,C) \qquad + 0 = \\ &= \boldsymbol{f(A,1,C)} \end{aligned}$$

Mappa di Karnaugh



Maurice Karnaugh

La **mappa di Karnaugh** è uno strumento che consente di ricavare, partendo dalla tavola di verità di una funzione logica, la sua **espressione minima**

Il metodo della mappa di Karnaugh si basa sul metodo della somma di mintermini, ma le espressioni vengono semplificate direttamente. Se il metodo viene applicato al massimo delle sue potenzialità, ovvero scegliendo i raggruppamenti migliori, si ricaverà l'espressione minima, altrimenti si ricaveranno delle espressioni semplificabili

Tale metodo è particolarmente efficace nei casi di funzioni fino a 4 variabili, gestibile nei casi di 5 e 6 variabili, difficilmente applicabile nei casi di funzioni a più di 6 variabili

Per le funzioni con più di 6 variabili si utilizzano altri metodi che non saranno trattati in questo corso

Mappa di Karnaugh

La mappa di Karnaugh può essere quindi utilizzata per ottimizzare un’espressione logica o un circuito logico

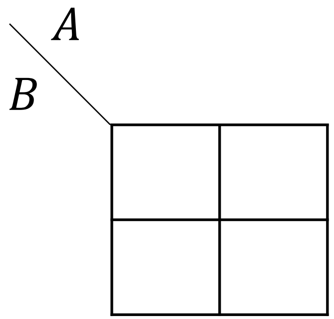
La mappa di Karnaugh opera in 3 fasi:

1. Costruzione della/e mappe
2. Scelta dei raggruppamenti
3. Generazione degli implicanti e dell’espressione finale

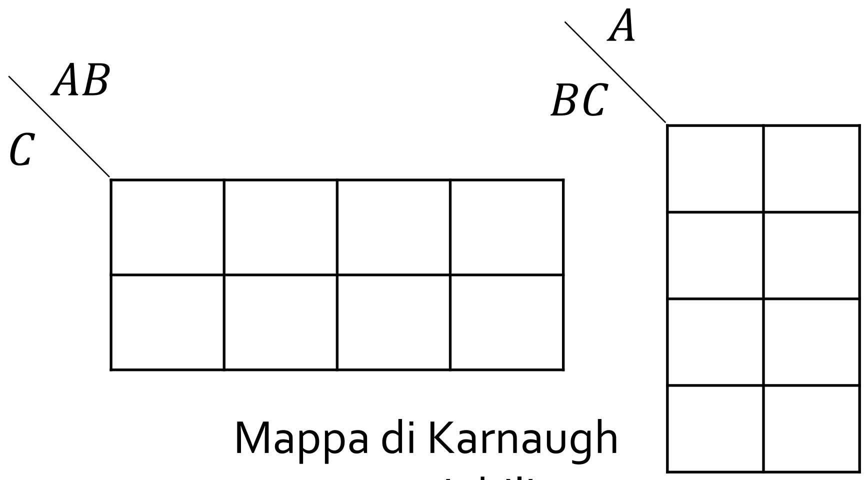
Mappa di Karnaugh – Costruzione della mappa

Una singola mappa di Karnaugh può essere realizzata per 2, 3 o 4 variabili

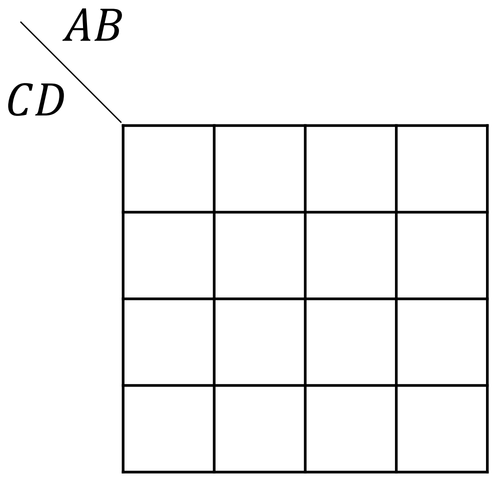
La costruzione di tale mappa avviene disponendo le variabili lungo il lato sinistro e il lato superiore di una tabella (massimo due variabili per lato). I lati dove sono disposte due variabili verranno divisi in 4 parti, i lati dove è disposta una sola variabile verranno divisi in 2 parti. La scelta di come disporre le variabili e in quale ordine sta all'utilizzatore della mappa ed è indifferente per il risultato



Mappa di Karnaugh
a 2 variabili



Mappa di Karnaugh
a 3 variabili
(due esempi)



Mappa di Karnaugh
a 4 variabili

Mappa di Karnaugh – Costruzione della mappa

Lungo i lati della mappa di Karnaugh si riportano le configurazioni:

- Per i lati a singola variabile si usa la configurazione 0-1
- Per i lati a doppia variabile si usa la configurazione 00-01-11-10

I valori riportati rappresentano le possibili configurazioni delle variabili. Tali disposizioni possono essere anche scritte al contrario (1-0; 10-11-01-00)

Attenzione: la configurazione per la doppia variabile NON è 00-01-10-11, ma è 00-01-11-10. Tale codifica prende il nome di **codifica di Gray** ed è utilizzata in questo caso in maniera tale che nel passare da una casella all'altra cambi sempre il valore di una sola variabile

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

Mappa di Karnaugh – Costruzione della mappa

$B \backslash A$	0	1
0		
1		

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

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

Mappa di Karnaugh – Costruzione della mappa

A	B	C	D	f
0	0	0	0	0
0	0	0	1	**1**
0	0	1	0	**1**
0	0	1	1	**1**
0	1	0	0	0
0	1	0	1	**1**
0	1	1	0	0
0	1	1	1	**1**
1	0	0	0	0
1	0	0	1	0
1	0	1	0	**1**
1	0	1	1	**1**
1	1	0	0	**1**
1	1	0	1	0
1	1	1	0	0
1	1	1	1	**1**

Prendendo la tavola di verità, si riportano nella mappa di Karnaugh i valori (è sufficiente riportare i valori per i quali la funzione assume il valore 1) della funzione per ciascuna configurazione:

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

Le due rappresentazioni si equivalgono

Mappa di Karnaugh – Scelta dei raggruppamenti

Dalla mappa di Karnaugh devono essere effettuati dei raggruppamenti delle celle dove è contenuto il valore 1

Tali raggruppamenti devono rispondere a tutte le seguenti condizioni:

1. Si possono raggruppare solo elementi tra celle adiacenti orizzontalmente o verticalmente
2. I raggruppamenti non devono contenere celle con il valore 0
3. Si possono raggruppare solo elementi in rettangoli o quadrati
4. Il numero di celle di ciascun raggruppamento deve essere una potenza di 2
5. Tutte le celle con il valore 1 devono essere contenute almeno in un raggruppamento

I raggruppamenti di 2^k elementi prendono anche il nome di **k-cubi**

Mappa di Karnaugh – Scelta dei raggruppamenti

Esempi di singoli raggruppamenti validi:

		1	
1	1	1	
1	1	1	1
1			1

		1	
1	1	1	
1	1	1	1
1			1

		1	
1	1	1	
1	1	1	1
1			1

		1	
1	1	1	
1	1	1	1
1			1

Esempi di singoli raggruppamenti NON validi:

		1	
1	1	1	
1	1	1	1
1			1

Il numero di elementi non è una potenza di 2

		1	
1	1	1	
1	1	1	1
1			1

Sono prese celle non aventi il valore 1

		1	
1	1	1	
1	1	1	1
1			1

Il raggruppamento non è rettangolare/quadrato

		1	
1	1	1	
1	1	1	1
1			1

Sono prese celle non adiacenti

Mappa di Karnaugh – Scelta dei raggruppamenti

NB: La griglia non è da considerarsi come un piano, bensì come un toro, pertanto i raggruppamenti possono attraversare i bordi della mappa, sia verticalmente che orizzontalmente, passando da un lato a quello opposto

Sono quindi dei raggruppamenti validi:

1		1	1
1			1
1		1	1

Oltrepassando il limite orizzontalmente

1		1	1
1			1
1		1	1

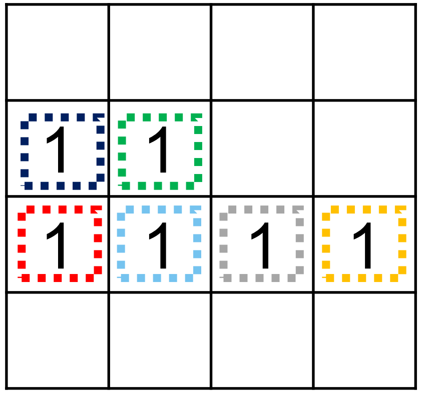
Oltrepassando il limite verticalmente

1		1	1
1			1
1		1	1

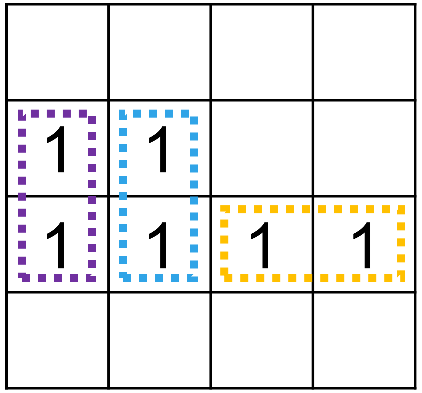
Oltrepassando il limite in entrambe le direzioni

Mappa di Karnaugh – Scelta dei raggruppamenti

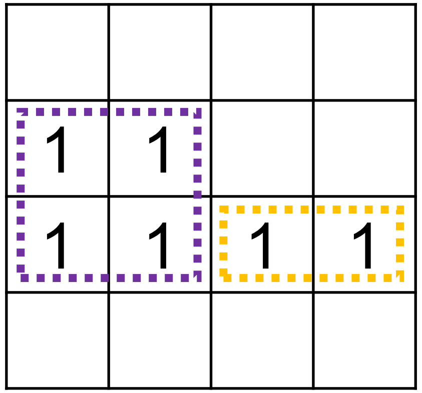
Nell'ambito della scelta dei raggruppamenti, è sempre preferibile scegliere i raggruppamenti in maniera tale che siano sempre i più grandi possibile, anche selezionando lo stesso valore più volte (ridondanza). La ragione verrà spiegata in seguito



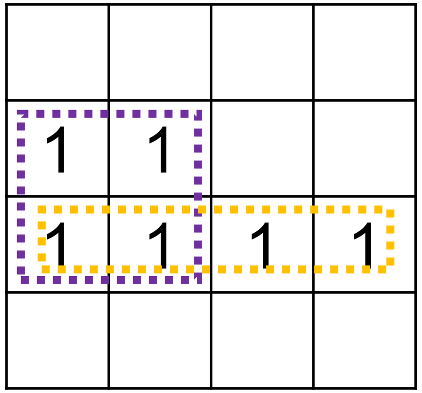
Raggr. Es. A



Raggr. Es. B



Raggr. Es. C

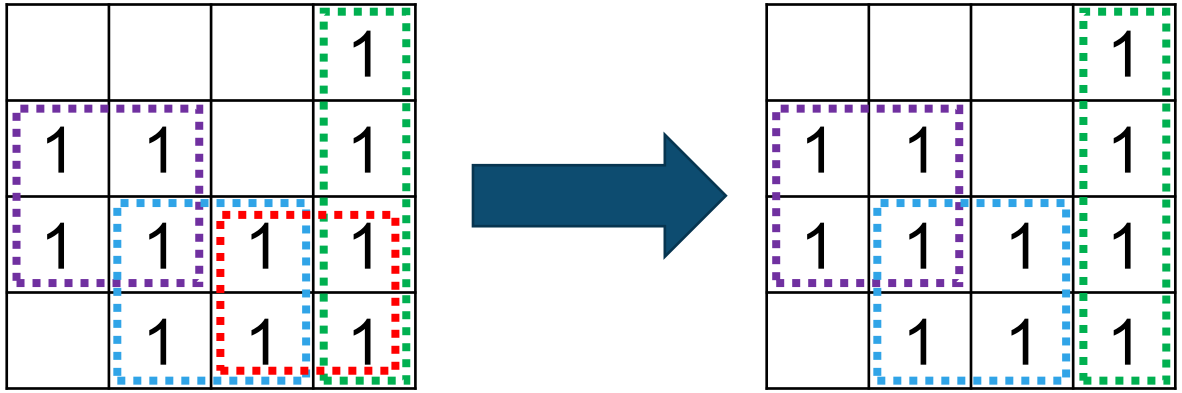


Raggr. Es. D

La qualità dei raggruppamenti migliora progressivamente dall'esempio A all'esempio D. L'esempio D è infatti quello che ha i raggruppamenti di dimensione più grande che coprono l'intera superfice delle celle con valore 1

Mappa di Karnaugh – Scelta dei raggruppamenti

Non ha invece senso creare un raggruppamento ridondante se tutte le celle interessate sono già state selezionate



Il raggruppamento in rosso è superfluo (quindi inutile): tutte le celle del raggruppamento sono già coperte dal raggruppamento azzurro e dal raggruppamento in verde

Mappa di Karnaugh – Generazione degli implicanti

L'espressione finale risultante dalla mappa di Karnaugh sarà una **somma di implicanti**:

$$\boldsymbol{f = I_1 + I_2 + \cdots + I_n}$$

Ciascun raggruppamento individuato darà luogo a un singolo implicante

Il singolo termine I_i, corrispondente a un k-cubo, è detto **implicante** in quanto implica la funzione f: se I_i è vero allora f è vero

«se I_i è vero allora f è vero» è corretto in virtù delle proprietà dell'OR (somma logica), cioè che una qualsiasi somma logica è vera se almeno uno degli elementi che la compone è vero

Mappa di Karnaugh – Generazione degli implicanti

Vediamo ora come si generano gli implicanti a partire dai raggruppamenti

L'implicante di un raggruppamento composto da una singola cella è equivalente al mintermine di tale cella

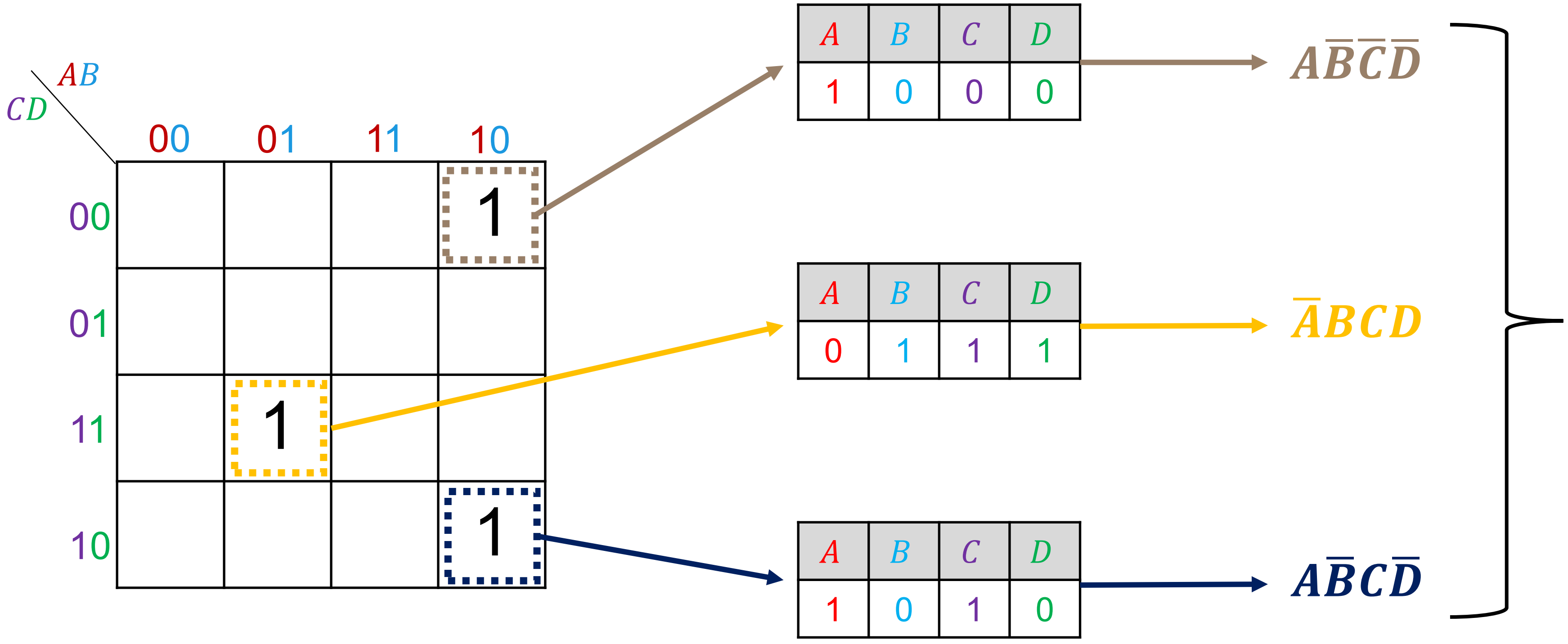
$CD \backslash AB$	00	01	11	10
00				
01				
11		$\boxed{1}$		
10				

A	B	C	D
0	1	1	1

$\bar{A}BCD$

Mappa di Karnaugh – Generazione degli implicanti

La mappa di Karnaugh, quindi, nel caso in cui vengano scelti tutti i raggruppamenti con celle singole, corrisponde perfettamente alla somma di mintermini:



$$f = A\bar{B}\bar{C}\bar{D} + \bar{A}BCD + A\bar{B}C\bar{D}$$

Mappa di Karnaugh – Generazione degli implicanti

L'implicante di un raggruppamento composto da più celle è calcolato in maniera analoga al mintermine ma **utilizzando solo le variabili che mantengono un valore costante**:

CD \ AB	00	01	11	10
00				
01				
11		1	1	
10				

A	B	C	D
0	1	1	1
1	1	1	1

$\rightarrow \boldsymbol{BCD}$

A varia (0-1), quindi viene esclusa
B è costante (1), quindi viene mantenuta
C è costante (1), quindi viene mantenuta
D è costante (1), quindi viene mantenuta

Mappa di Karnaugh – Generazione degli implicanti

Altri due esempi:

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

A	B	C	D
0	0	0	0
0	1	0	0
1	1	0	0
1	0	0	0
0	0	0	1
0	1	0	1
1	1	0	1
1	0	0	1

A varia (0-1), quindi viene esclusa
B varia (0-1), quindi viene esclusa
C è costante (0), quindi viene mantenuta
D varia (0-1), quindi viene esclusa

$\rightarrow \overline{C}$

A	B	C	D
1	1	1	1
1	0	1	1
1	1	1	0
1	0	1	0

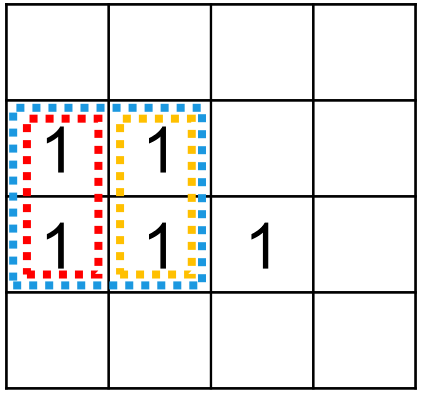
A è costante (1), quindi viene mantenuta
B varia (0-1), quindi viene esclusa
C è costante (1), quindi viene mantenuta
D varia (0-1), quindi viene esclusa

$\rightarrow AC$

Mappa di Karnaugh – Generazione degli implicanti

Un implicante è detto **implicante primo** di f se non implica un altro implicante di f

Un implicante primo corrisponde a un k-cubo di dimensione massima, ovvero un k-cubo che non è contenuto (per intero) in un x-cubo di dimensioni maggiori:



Gli implicanti prodotti dai raggruppamenti rosso e giallo NON sono primi, in quanto sono contenuti interamente all'interno del raggruppamento azzurro. Il raggruppamento azzurro è un 2-cubo, mentre invece il raggruppamento rosso e quello giallo sono 1-cubi

Il raggruppamento azzurro genera un implicante primo, in quanto k-cubo dimensioni maggiori

Mappa di Karnaugh – Generazione degli implicanti

E' semplice dimostrare che gli implicanti corrispondenti ai raggruppamenti rosso e giallo implicano quello corrispondente al raggruppamento azzurro (pertanto non sono primi):

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

$\bar{A}D$ $\quad\quad$ $\bar{A}\bar{B}D$ $\quad\quad$ $\bar{A}BD$

Th: $\bar{A}\bar{B}D \Longrightarrow \bar{A}D$ ovvero che se $\bar{A}\bar{B}D$ è vero allora $\bar{A}D$ è vero
Per proprietà associativa, $\bar{A}\bar{B}D = (\bar{A}D)\bar{B}$. Sappiamo dalle proprietà dell'AND che se $(\bar{A}D)\bar{B}$ è vero, allora lo sono sia $\bar{A}D$ che $\bar{B}$. Quindi $\bar{A}D$ è vero

Th: $\bar{A}BD \Longrightarrow \bar{A}D$ ovvero che se $\bar{A}BD$ è vero allora $\bar{A}D$ è vero
Per proprietà associativa, $\bar{A}BD = (\bar{A}D)B$. Sappiamo dalle proprietà dell'AND che se $(\bar{A}D)B$ è vero, allora lo sono sia $\bar{A}D$ che B. Quindi $\bar{A}D$ è vero

Infatti gli implicanti dei raggruppamenti rosso e giallo possono essere semplificati algebricamente nell'implicante primo del raggruppamento azzurro:

$\bar{A}\bar{B}D + \bar{A}BD = \bar{A}D(\bar{B}+B) = \bar{A}D(1) = \bar{A}D$

Mappa di Karnaugh – Generazione degli implicanti

Come già affermato in precedenza, i raggruppamenti di destra nell'esempio di sotto sono preferibili a quelli di sinistra. Ciò perché gli implicanti dei raggruppamenti di destra sono più semplici...

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

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

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

$f' = \bar{A}D + BCD$

...e sono ricavabili algebricamente dagli implicanti dei raggruppamenti di sinistra, scomponendo il raggruppamento azzurro nelle sue parti, applicando le proprietà e ricomponendo il raggruppamento azzurro:

$$f = \bar{A}D + ABCD = (\bar{A}\bar{B}\bar{C}D + \bar{A}B\bar{C}D + \bar{A}\bar{B}CD + \bar{A}BCD) + ABCD =$$

Scompongo raggruppamento azzurro

$$= \bar{A}\bar{B}\bar{C}D + \bar{A}B\bar{C}D + \bar{A}\bar{B}CD + \bar{A}BCD + ABCD =$$

$$= \bar{A}\bar{B}\bar{C}D + \bar{A}B\bar{C}D + \bar{A}\bar{B}CD + \bar{A}BCD + ABCD + \bar{A}BCD =$$

Applico idempotenza OR aggiungendo un altro $\bar{A}BCD$

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

Metto a fattor comune BCD e applico complementazione OR

$$= \bar{A}\bar{B}\bar{C}D + \bar{A}B\bar{C}D + \bar{A}\bar{B}CD + \bar{A}BCD + BCD(1) =$$

Semplifico elemento neutro AND

$$= \bar{A}\bar{B}\bar{C}D + \bar{A}B\bar{C}D + \bar{A}\bar{B}CD + \bar{A}BCD + BCD =$$

Ricompongo il raggruppamento azzurro

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

$$= \bar{A}D + BCD = f'$$

Mappa di Karnaugh – Generazione degli implicanti

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

$\bar{A}D$ e BCD sono quindi ambedue implicanti primi (anche se hanno $\bar{A}BCD$ in comune)

Infatti, l'uno non implica l'altro. Provando a dimostrare che l'uno implichi l'altro si ha:

Th: 1. $\bar{A}D \Longrightarrow BCD$ ovvero che se $\bar{A}D$ è vero allora BCD è vero
oppure 2. $BCD \Longrightarrow \bar{A}D$ ovvero che se BCD è vero allora $\bar{A}D$ è vero

1. Per proprietà associativa $\bar{A}D = (\bar{A})(D)$. Sapendo che $\bar{A}D$ è vero, allora sia $\bar{A}$ che D sono vere. Pertanto $BCD = BC1 = BC$. Si vuole quindi ora dimostrare che $\bar{A}D \Longrightarrow BC$, ovvero che $1 \Longrightarrow BC$, ma non vi sono elementi sulla veridicità di B e C, pertanto l'uno non implica l'altro

2. Per proprietà associativa $BCD = (B)(C)(D)$. Sapendo che BCD è vero, allora sia B che C che D sono vere. Pertanto $\bar{A}D = \bar{A}1 = \bar{A}$. Si vuole quindi ora dimostrare che $BCD \Longrightarrow \bar{A}$, ovvero che $1 \Longrightarrow \bar{A}$, ma non vi sono elementi sulla veridicità di A, pertanto l'uno non implica l'altro

Mappa di Karnaugh – Condizioni di indifferenza

È possibile avere delle funzioni booleane nelle quali l'output della funzione per una certa configurazione, per diverse ragioni possibili (ad es. è una configurazione non ammessa), non ci interessa. Tale condizione prende il nome di **condizione di indifferenza** per la specifica configurazione

Le condizioni di indifferenza sono generalmente indicate con la lettera greca phi ϕ (che idealmente rappresenta uno zero e un uno sovrapposti), ma anche con X, Δ, *…

Le condizioni di indifferenza rappresentano quindi un jolly: verranno considerate se aiutano (in verde) a costituire un raggruppamento o a costituire un raggruppamento migliore, non verranno considerate se non forniscono un contributo (in rosso)

Non hanno senso raggruppamenti di sole condizioni di indifferenza

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

BD

Mappa di Karnaugh – Dividere in più mappe

Utilizzando il Teorema di Shannon, si può dividere una qualsiasi mappa di Karnaugh a n variabili in due mappe di Karnaugh a $n-1$ variabili

Ad esempio, dividendo una mappa a 4 variabili A, B, C, D sulla base della variabile C:

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

(con $C = 0$)

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

$\bar{C}$

$+$

(con $C = 1$)

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

C

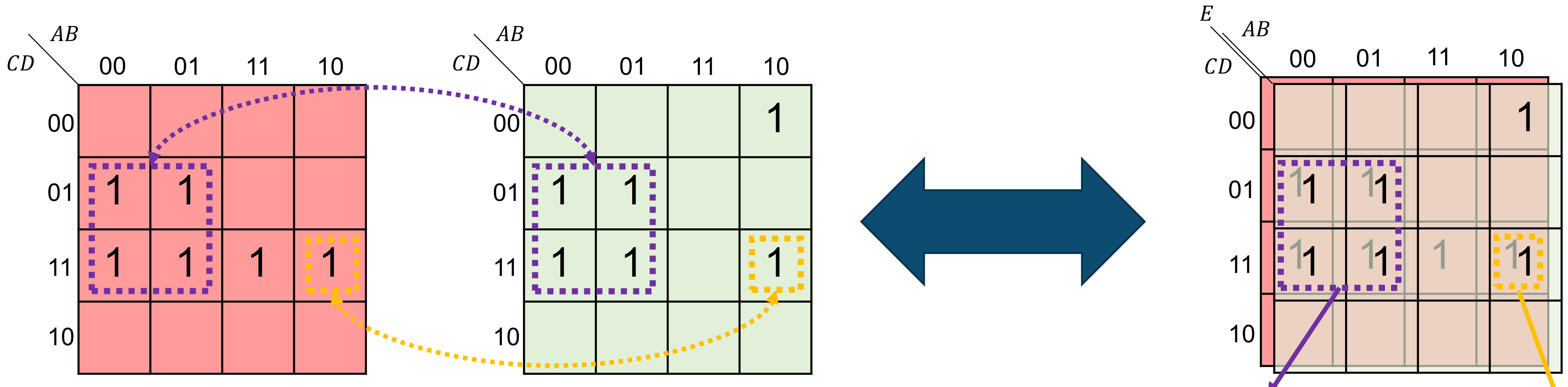
Agli implicanti ottenuti dalle singole mappe sarà quindi necessario moltiplicare il termine C, sulla base della mappa:

- C per la mappa con $C = 1$
- $\bar{C}$ per la mappa con $C = 0$

Tale approccio viene applicato per utilizzare le mappe di Karnaugh con 5 variabili (due mappe da 4 variabili) e con 6 variabili (quattro mappe da 4 variabili)

Mappa di Karnaugh – Dividere in più mappe

Le due mappe separate possono essere viste in maniera sovrapposta, consentendo dei raggruppamenti cubici: abbiamo parlato, infatti, di k-cubi (in realtà sono parallelepipedi)



(con $E = 0$)

(con $E = 1$)

A	B	C	D	E
0	0	0	1	0
0	1	0	1	0
0	0	1	1	0
0	1	1	1	0
0	0	0	1	1
0	1	0	1	1
0	0	1	1	1
0	1	1	1	1

$\bar{A}D$

A	B	C	D	E
1	0	1	1	0
1	0	1	1	1

$A\bar{B}CD$

NB: quelli proposti non sono i raggruppamenti migliori, in quanto si sarebbe potuto operare un raggruppamento di 4 elementi (il raggruppamento giallo con $\bar{A}\bar{B}CD\bar{E}$ e $\bar{A}\bar{B}CDE$)

Mappa di Karnaugh – Maxtermini

È possibile, in maniera speculare, utilizzare le mappe di Karnaugh con gli zeri, individuando un corrispettivo speculare degli implicanti analogo ai maxtermini

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

A	B	C	D
0	0	0	0
0	1	0	0
0	0	0	1
0	1	0	1

$\rightarrow A + C$

A	B	C	D
0	1	1	1
1	1	1	1

$\rightarrow \bar{B} + \bar{C} + \bar{D}$

A	B	C	D
0	0	0	0
0	0	0	1
0	0	1	1
0	0	1	0

$\rightarrow A + B$

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