figura 2.3*d* è *a tre livelli*, poiché i segnali b e c incontrano tre blocchi, mentre il segnale a ne incontra due.

Spesso sono disponibili le variabili di ingresso in forma diretta e negata (vedi sotto), e i blocchi negatori per costruire le variabili negate non sono considerati nel computo dei livelli: per esempio la rete di figura 2.3*d* diviene a due livelli. Tale ipotesi è sempre ammessa per le reti SP e PS, che si considerano *a due livelli*. Per comodità di trattazione parleremo primariamente di queste reti, notando che le reti NAND e NOR che si ottengono per trasformazione diretta delle SP e PS (vedi Premessa) sono anch'esse a due livelli, e soggette a considerazioni perfettamente parallele a quelle che svolgeremo sotto.

Nel considerare la sintesi ottima di reti a due livelli supporremo che l'unico parametro da minimizzare sia il costo della rete, e che tale costo sia funzione crescente del numero N_b di blocchi AND e OR, o NAND, o NOR impiegati e, a pari N_b, del numero totale N_m dei morsetti di ingresso a tali blocchi. Come già detto, è lecito trascurare i negatori, o i NAND o NOR usati come negatori, perché nella pratica i segnali che alimentano le reti combinatorie provengono da "registri" che forniscono le variabili dirette e negate (cap. 4). Inoltre, se, come avviene nel progetto di reti complesse, è opportuno trasferire da un punto a un altro del sistema solo i segnali diretti, per ridurre il numero dei collegamenti, si ricostruiscono i segnali negati all'ingresso della rete da progettare con uno strato di negatori che nulla aggiunge allo svolgimento logico del processo di sintesi.

La sintesi ottima si ottiene lavorando sulla rappresentazione algebrica della rete, da cui i valori di N_b e N_m si ricavano con regole ovvie. Per esempio per la forma SP: $\overline{a}\overline{b}c + a\overline{b}c + ab\overline{c}$ abbiamo $N_b = 4$, $N_m = 12$; per la forma PS: $a \cdot (b + \overline{c})$ abbiamo $N_b = 2$, $N_m = 4$.

Non vi è naturalmente alcun motivo perché la rete ottima per un dato problema (cioè per una data descrizione ai morsetti) sia unica: in effetti esistono in genere più reti SP (o NAND) di costo minimo, più reti PS (o NOR) di costo minimo, e il costo delle une può essere uguale al costo delle altre. Se è assegnato il tipo di rete da realizzare (SP, PS, NAND, NOR), si cerca una rete ottima tra quelle di tale tipo; altrimenti si cercano reti ottime per ogni tipo, e si confrontano tra loro per determinarne una ottima in assoluto.

Vediamo ora come si ottiene una *forma SP minima*: una forma SP, cioè, per cui è minimo il valore di N_b e, a pari N_b, è minimo il valore di N_m. La proprietà 3 del paragrafo precedente si estende nel nuovo enunciato:

Proprietà 5 Ogni forma SP minima di una funzione f è una forma SP prima irridondante per f.

Che ogni forma SP minima sia una somma irridondante di implicanti primi si dimostra notando che: *a*) ogni forma SP di f è una somma di implicanti di f; *b*) se un implicante p_i della forma minima non fosse primo, esisterebbe un implicante primo p_i' tale che $p_i \rightarrow p_i'$ e $p_i' \rightarrow f$: poiché il prodotto che esprime p_i' conterrebbe meno variabili di quello che esprime p_i, p_i' potrebbe essere sostituito a p_i nella forma minima causando una riduzione del valore di N_m (contro l'ipotesi che detta forma sia minima); *c*) se la forma minima non fosse irridondante, si potrebbe escludere da essa almeno un prodotto, riducendo il valore di N_b (contro l'ipotesi che detta forma sia minima).

Dalla proprietà 4 risulta inoltre che ogni forma SP minima contiene tutti gli implicanti primi essenziali di f. Per esempio la prima delle due forme riportate nella figura 2.9*b* è una forma SP minima per la z (il lettore potrà convincersene con semplici ragionamenti); la seconda forma ivi riportata non è invece minima.

La ricerca della forma SP minima si suddivide in due fasi:

fase A: determinazione di tutti gli implicanti primi della f;

fase B: selezione di un insieme (irridondante) di implicanti primi la cui somma copra la f, e il cui costo complessivo sia minimo.

Poiché il primo parametro da minimizzare è N_b, l'insieme di costo minimo deve avere minima cardinalità.

Per funzioni di $n \leqslant 5$ variabili la ricerca della soluzione si sviluppa sulle mappe di Karnaugh: ci si lascia aiutare largamente dall'intuito, generando diversi insiemi irridondanti di implicanti primi e confrontandoli tra loro solo quando la scelta di un insieme ottimo non sia immediatamente palese (come lo è nell'esempio di fig. 2.9). Questo procedimento si basa unicamente sull'esperienza: in esso le due fasi A e B possono non essere del tutto distinte, né occorrerà generare esplicitamente tutti gli implicanti primi quando la soluzione si intravede altrimenti. Ognuno si avvarrà dello schema mentale che preferisce: non cercheremo di insegnare qui ciò che più ragionevolmente segue linee personali, sottolineando però due punti:

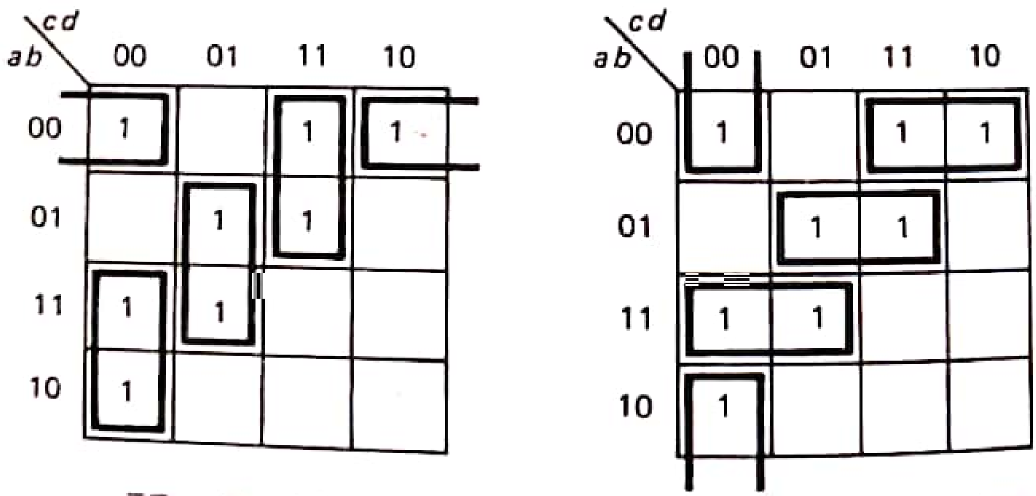
1) gli implicanti primi essenziali sono tutti contenuti nella forma minima, ed è quindi opportuno individuarli per primi: ciò può essere fatto esaminando ogni 1 della funzione e controllando se esiste un solo im-

plicante primo che copre tale 1 (in tal caso l'implicante è primo essenziale):

2) determinati gli implicanti primi essenziali, si cerca una ulteriore scelta di implicanti primi che copra gli 1 non coperti dai precedenti (è appena il caso di ricordare che un 1 della funzione f può essere coperto da più implicanti in una forma SP di f): in questa fase può essere necessario generare più scelte e confrontarle tra loro.

Esaminiamo nuovamente l'esempio di figura 2.9. L'1 nella cella 1000 è coperto dal solo implicante primo $a\overline{c}$, che risulta quindi essenziale e viene selezionato; l'esame degli 1 della f non coperti da $a\overline{c}$ non conduce alla determinazione di alcun altro implicante primo essenziale; tali 1 sono quattro, e a causa della loro posizione sulla mappa non possono essere tutti coperti da un solo implicante: occorreranno dunque almeno due altri implicanti primi per coprire la f, ed è immediato constatare che la prima scelta di figura 2.9*b* fornisce la soluzione ottima.

La figura 2.10*a* fornisce un semplice esempio, che il lettore potrà discutere per proprio conto. Più complesso è l'esempio di figura 2.10*b*, che riprendiamo nella figura 2.12. Dall'esame della mappa di figura 2.12 risulta chiaro che non esistono per la funzione implicanti primi essenziali, né implicanti primi che includono un maggior numero di 1 di altri (e che sarebbero quindi rappresentati da prodotti con minor numero di variabili). Scelto allora a caso un implicante primo: $a\overline{c}\overline{d}$,



$f = a\overline{c}\overline{d} + b\overline{c}d + \overline{a}cd + \overline{a}\overline{b}\overline{d}$ $\qquad$ $f = \overline{b}\overline{c}\overline{d} + ab\overline{c} + \overline{a}bd + \overline{a}\overline{b}c$

(*a*) $\qquad$ (*b*)

Figura 2.12
Due forme SP minime per la funzione di figura 2.10*b*.

esaminiamo in alternativa il caso che tale implicante sia contenuto nella forma minima cercata, e il caso che non vi sia. Se $a\bar{c}\bar{d}$ è contenuto nella soluzione (fig. 2.12*a*), l'1 nella cella 1100 è coperto da esso e non deve più necessariamente essere coperto dall'implicante $ab\bar{c}$: poiché tale implicante copre anche l'1 nella cella 1101, che a sua volta è coperto dall'altro implicante $b\bar{c}d$, concludiamo che $ab\bar{c}$ può essere scartato, ma si deve scegliere $b\bar{c}d$. Continuando nello stesso modo si ottiene la forma irridondante di figura 2.12*a*. Per converso, se $a\bar{c}\bar{d}$ non è contenuto nella soluzione, questa deve necessariamente contenere $\bar{b}\bar{c}\bar{d}$ e $ab\bar{c}$: procedendo con il meccanismo precedente si ottiene la forma irridondante di figura 2.12*b*. Le due forme ottenute sono (le uniche) forme SP minime per la funzione.

La più volte citata trasformazione algebrica consente di passare da una forma SP minima a una *forma NAND minima*. Omettiamo la dimostrazione della minimalità di tale seconda forma, ricordando però che essa può essere ottenuta direttamente dagli implicanti individuati sulla mappa; per esempio per la selezione di figura 2.12*a* avremo la forma NAND minima:

$$f = (a|\bar{c}|\bar{d})|(b|\bar{c}|d)|(\bar{a}|c|d)|(\bar{a}|\bar{b}|\bar{d}).$$

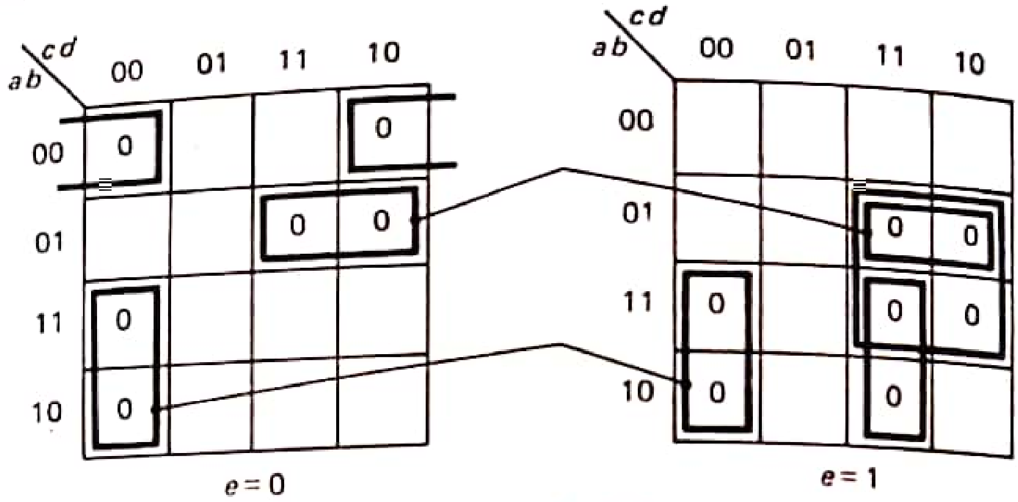
(Si ricordi che nella derivazione diretta di forme NAND e NOR, due casi limite, non presenti nell'espressione appena generata, possono causare qualche confusione; si veda la Premessa.)

Una trattazione duale a quella svolta riguarda le forme PS. La proprietà 3′ del paragrafo 2.3 si estende nel nuovo enunciato:

Proprietà 5′ Ogni forma PS minima di una funzione f è una forma PS prima irridondante per f.

Una forma PS minima è dunque un prodotto irridondante di implicati primi di f, i cui 0 coprono gli 0 di f, e che contiene il minimo numero di implicati primi (in particolare deve contenere tutti gli implicati primi essenziali).

La ricerca di una forma PS minima viene condotta, similmente alle forme SP, attraverso la determinazione degli implicati primi essenziali, e la successiva selezione di un insieme minimo di altri implicati primi che coprano gli 0 non coperti da quelli essenziali. Per esempio la funzione di figura 2.11 ammette le forme PS minima, e corrispondente *NOR minima*, indicate in figura 2.13.



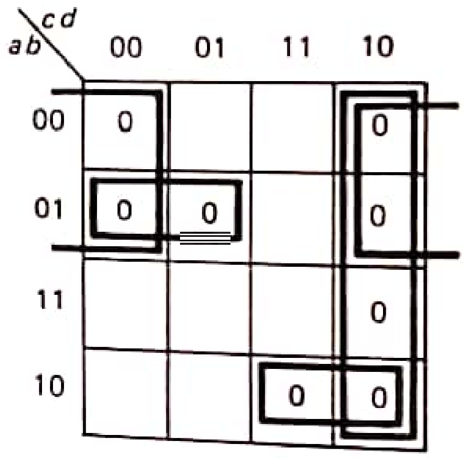
$$z = (\bar{a} + c + d)\cdot(a + \bar{b} + \bar{c})\cdot(\bar{a} + \bar{c} + \bar{d} + \bar{e})\cdot(a + b + d + e)\cdot(\bar{b} + \bar{c} + \bar{e})$$

$$z = (\bar{a} \downarrow c \downarrow d) \downarrow (a \downarrow \bar{b} \downarrow \bar{c}) \downarrow (\bar{a} \downarrow \bar{c} \downarrow \bar{d} \downarrow \bar{e}) \downarrow (a \downarrow b \downarrow d \downarrow e) \downarrow (\bar{b} \downarrow \bar{c} \downarrow \bar{e})$$

Figura 2.13
Forme PS e NOR minima per la funzione di figura 2.11.

Studiamo infine la funzione della figura 2.9 in forma PS, riportando su una mappa gli 0 della funzione (fig. 2.14): tutti gli implicati primi sono in questo caso essenziali, e il loro prodotto costituisce quindi l'unica forma PS minima. Il costo di questa forma è maggiore di quello della forma SP minima per la stessa funzione (fig. 2.9*b*, prima forma indicata).

Nel prossimo paragrafo studieremo la sintesi ottima a due livelli per funzioni di un numero qualsiasi di variabili.



$$z = (a + d)\cdot(\bar{c} + d)\cdot(a + \bar{b} + c)\cdot(\bar{a} + b + \bar{c})$$

Figura 2.14
Forma PS minima per la funzione di figura 2.9.

Esercizi

2.12 Ricavare una forma SP minima e una forma PS minima per ognuna delle funzioni dell'esercizio 2.7. Ricavare quindi le forme minime NAND e NOR.

2.13 Ricavare una forma SP minima per la funzione di figura 2.13, e una forma PS minima per la funzione di figura 2.12.

2.14 Per le funzioni:

$a \backslash bc$	00	01	11	10
0	1	1		
1				

f_1

$a \backslash bc$	00	01	11	10
0	1	1	1	
1	1	1		

f_2

ricavare la forma SP minima e trasformarla algebricamente nella forma NAND minima; quindi ricavare la forma NAND minima dalla mappa: se le due forme così ottenute dovessero differire, studiare attentamente quale errore è stato commesso (alla luce di quanto esposto nella Premessa).

Ripetere l'esercizio per le forme NOR minime delle funzioni: $f_3 = \Pi_3(6, 7)$, $f_4 = \Pi_3(2, 3, 5, 6, 7)$.

2.15 Dimostrare che con operatori AND, OR, NOT, o NAND, o NOR, non si possono in genere realizzare reti con meno di due livelli, a eccezione di quelle che corrispondono a funzioni banali; e che non è ragionevole pensare di evitare questa limitazione cambiando la scelta degli operatori.

2.16 Un *confrontatore* è una rete a $2n$ ingressi e una uscita, che accetta in ingresso due numeri di n bit e produce uscita 1 se il primo numero è maggiore o uguale al secondo, uscita 0 altrimenti. Progettare un confrontatore per numeri di due bit.

2.17 Per una configurazione di n bit, il *bit di parità* è un bit addizionale che vale 1 se il numero di 1 della configurazione è dispari, vale 0 se tale numero è pari. (Il bit di parità si introduce per poter controllare successivamente se la configurazione subisce errori, come sarà diffusamente spiegato nel cap. 9.) Progettare una rete per la generazione del bit di parità per configurazioni di cinque bit. Qual è la forma della rete per configurazioni di n bit?