11.4 Minimização de Circuitos

Introdução

A eficiência de um circuito combinatório depende do número e da disposição de suas portas. O processo de projetar um circuito combinatório começa com a tabela que especifica a saída para cada combinação de valores de entrada. Podemos sempre usar a expansão em soma de produtos de um circuito para encontrar um conjunto de portas lógicas que implementarão este circuito. Entretanto, a expansão em soma de produtos pode conter muito mais termos do que o necessário. Os termos em uma expansão em soma de produtos que diferem em apenas uma variável, de modo que em um termo esta variável ocorre e no outro termo ocorre o complemento desta variável, podem ser combinados. Por exemplo, considere o circuito que tem a saída 1 se e somente se $x = y = z = 1$ ou $x = z = 1$ e $y = 0$. A expansão em soma de produtos desse circuito é $xyz + x\bar{y}z$.

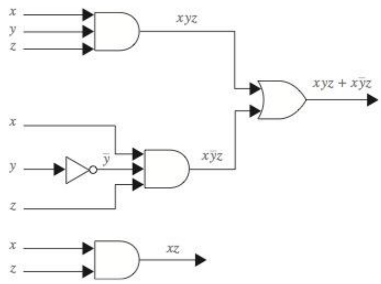

FIGURA 1 Dois Circuitos com a Mesma Saída.

Os dois produtos nesta expansão diferem em exatamente uma variável, a saber, y. Eles podem ser combinados como

$$\begin{aligned} xyz + x\bar{y}z &= (y + \bar{y})\,(xz) \\ &= 1 \cdot (xz) \\ &= xz. \end{aligned}$$

Logo, xz é uma expressão booleana com menos operadores que representa o circuito. Mostramos duas implementações diferentes desse circuito na Figura 1. O segundo circuito usa apenas uma porta, enquanto o primeiro circuito usa três portas e um inversor.

Esse exemplo mostra que combinar termos na expansão em soma de produtos de um circuito leva a uma expressão mais simples para o circuito. Descreveremos dois procedimentos que simplificam essas expansões.

O objetivo de ambos os procedimentos é produzir somas booleanas de produtos booleanos que representem uma função com o mínimo de produtos de literais, de modo que esses produtos contenham o menor número possível de literais entre todas as somas de produtos que representam a função booleana. Encontrar essa soma de produtos é chamado de **minimização de uma função booleana**, que torna possível construir um circuito para essa função que usa o menor número de portas e o menor número de entradas para as portas *E* e para as portas *OU* no circuito, entre todos os circuitos para a expressão booleana que está sendo minimizada.

Até o início de década de 1960, portas lógicas eram componentes individuais. Para reduzir custos, era importante usar o menor número possível de portas para produzir o resultado desejado. Entretanto, em meados da década de 1960, foi desenvolvida a tecnologia de circuitos integrados que tornou possível combinar portas em um único chip. Mesmo que atualmente seja possível construir circuitos integrados cada vez mais complexos em chips de baixo custo, a minimização de funções booleanas continua importante.

Reduzir o número de portas em um chip pode levar a circuitos mais confiáveis e reduzir o custo de produção do chip. Além disso, a minimização torna possível colocar mais circuitos no mesmo chip. Ela também reduz o número de entradas das portas de um circuito, que reduz o tempo usado por um circuito para computar sua saída. Além do mais, o número de entradas de uma porta pode ser limitado por causa da tecnologia particular usada para construir portas lógicas.

O primeiro procedimento que introduziremos, conhecido como mapas de Karnaugh (ou mapas K), foi desenvolvido na década de 1950 para ajudar a minimizar circuitos manualmente. Os mapas K são úteis na minimização de circuitos com até seis variáveis, embora eles se tornem bem complexos mesmo para cinco ou seis variáveis. O segundo procedimento que descreveremos, o método de Quine–McCluskey, foi inventado na década de 1960. Ele automatiza o processo de minimizar circuitos combinatórios e pode ser implementado como um programa de computador.

Infelizmente, minimizar funções booleanas com muitas variáveis é um problema computacionalmente intensivo. Foi mostrado que esse é um problema NP-completo (veja a Seção 3.3 e [Ka93]), de modo que a existência de algoritmos polinomiais no tempo para minimizar circuitos booleanos é pouco provável. O método de Quine–McCluskey tem complexidade exponencial. Na prática, ele pode ser usado apenas quando o número de literais não exceder 10. Desde a década de 1970, foram desenvolvidos diversos novos algoritmos para minimizar circuitos combinatórios (veja [Ha93] e [Ka93]). No entanto, com os melhores algoritmos já desenvolvidos, apenas circuitos com não mais de 25 variáveis podem ser minimizados. Além disso, métodos heurísticos (ou práticos) podem ser usados para simplificar substancialmente, mas não necessariamente minimizar, expressões booleanas com um número maior de literais.

Mapas de Karnaugh

Para reduzir o número de termos em uma expressão booleana que representa um circuito, é necessário encontrar termos para combinar. Existe um método gráfico, chamado de **mapa de Karnaugh** ou **mapa K**, para encontrar termos para combinar nas funções booleanas que envolvem um número relativamente pequeno de variáveis. O método que descreveremos foi introduzido por Maurice Karnaugh, em 1953. Seu método baseia-se em trabalhos anteriores de E. W. Veitch. (Este método é, em geral, aplicado apenas quando a função envolve seis variáveis ou menos.) Os mapas K nos dão um método visual para simplificar expansões em soma de produtos; eles não são adequados para mecanizar este processo. Ilustraremos primeiro como os mapas K podem ser usados para simplificar expansões de funções booleanas em duas variáveis. Continuaremos mostrando como eles podem ser usados para minimizar funções booleanas em três variáveis e, então, em quatro variáveis. A seguir, descreveremos os conceitos que podem ser usados para estendê-los para minimizar funções booleanas em mais de quatro variáveis.

	y	$\bar{y}$
x	xy	$x\bar{y}$
$\bar{x}$	$\bar{x}y$	$\bar{x}\bar{y}$

FIGURA 2
Mapas K em Duas Variáveis.

Existem quatro mintermos possíveis em expansões em soma de produtos de uma função booleana em duas variáveis x e y. Um mapa K para uma função booleana nessas duas variáveis consiste em quatro células, nas quais é colocado um 1 na célula que representa um mintermo se este mintermo está presente na expansão. As células são denominadas **adjacentes** se os mintermos que elas representam diferem em exatamente um literal. Por exemplo, a célula que representa $\bar{x}y$ é adjacente às células que representam xy e $\bar{x}\bar{y}$. As quatro células e os termos que elas representam estão mostrados na Figura 2.

EXEMPLO 1 Encontre mapas K para (a) $x\,y + \bar{x}\,y$, (b) $x\,\bar{y} + \bar{x}\,y$ e (c) $x\,\bar{y} + \bar{x}\,y + \bar{x}\,\bar{y}$.

Solução: Incluímos um 1 na célula quando o mintermo representado por ela está presente na expansão em soma de produtos. Os três mapas K estão mostrados na Figura 3. ◄

Podemos identificar mintermos que podem ser combinados a partir do mapa K. Sempre que existirem 1s em duas células adjacentes no mapa K, os mintermos representados por essas células podem ser combinados em um produto que envolve apenas uma das variáveis. Por exemplo, $x\,\bar{y}$ e $\bar{x}\,\bar{y}$ são representados por células adjacentes e podem ser combinados em $\bar{y}$, pois $x\,\bar{y} + \bar{x}\,\bar{y} = (x + \bar{x})\bar{y} = \bar{y}$. Além disso, se existirem 1s nas quatro células, os quatro mintermos podem ser combinados em um termo, a saber, a expressão booleana 1, que não envolve nenhuma das variáveis. Circulamos blocos de células no mapa K que representam mintermos que podem ser combinados e então encontramos a soma de produtos correspondente. O objetivo é identificar os

MAURICE KARNAUGH (NASCIDO EM 1924) Maurice Karnaugh, nascido na cidade de Nova York, fez sua graduação na City College of New York e seu mestrado e doutorado na Universidade de Yale. Ele foi um membro da equipe tecnológica no Bell Laboratories, de 1952 até 1966, e Gerente de Pesquisa e Desenvolvimento na Federal Systems Division da AT&T, de 1966 a 1970. Em 1970, ingressou na IBM como membro da equipe de pesquisa. Karnaugh fez contribuições fundamentais às aplicações das técnicas digitais tanto na computação quanto nas telecomunicações. Seus interesses atuais incluem sistemas que se baseiam em conhecimento em computação e métodos de busca heurísticos.

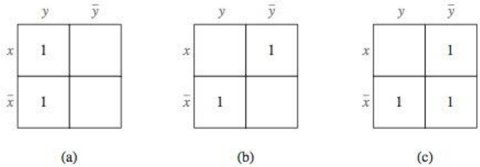

FIGURA 3 Mapas K para as Expansões em Soma de Produtos no Exemplo 1.

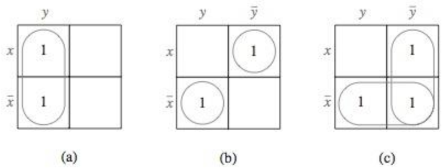

FIGURA 4 Simplificação das Expansões em Soma de Produtos do Exemplo 2.

os maiores blocos possíveis e cobrir todos os 1s com o menor número de blocos usando os blocos maiores primeiro e sempre usando os maiores blocos possíveis.

EXEMPLO 2 Simplifique as expansões em soma de produtos dadas no Exemplo1.

Solução: O agrupamento de mintermos é mostrado na Figura 4, usando os mapas K para essas expansões. As expansões mínimas para essas somas de produtos são (a) y, (b) $x\,\bar{y} + \bar{x}\,y$ e (c) $\bar{x} + \bar{y}$. ◀

Um mapa K em três variáveis é um retângulo dividido em oito células. Essas células representam os oito mintermos possíveis em três variáveis. Duas células são denominadas adjacentes se os mintermos que elas representam diferem em exatamente um literal. Uma das maneiras de formar um mapa K em três variáveis é mostrada na Figura 5(a). Este mapa K pode ser considerado como estando em um cilindro, conforme mostrado na Figura 5(b). No cilindro, duas células têm uma fronteira em comum se e somente se elas forem adjacentes.

Para simplificar uma expansão em soma de produtos em três variáveis, usamos o mapa K para identificar blocos de mintermos que podem ser combinados. Blocos de duas células adjacentes representam pares de mintermos que podem ser combinados em um produto de duas literais; blocos 2×2 e 4×1 de células representam mintermos que podem ser combinados em um único literal; e o bloco de todas as oito células representa um produto sem nenhum literal, ou

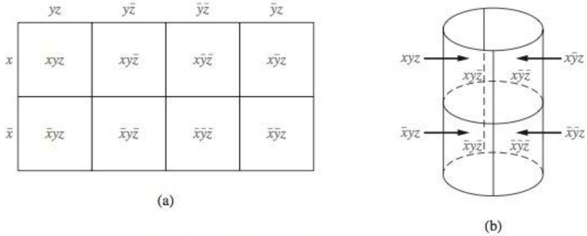

FIGURA 5 Mapas K em Três Variáveis.

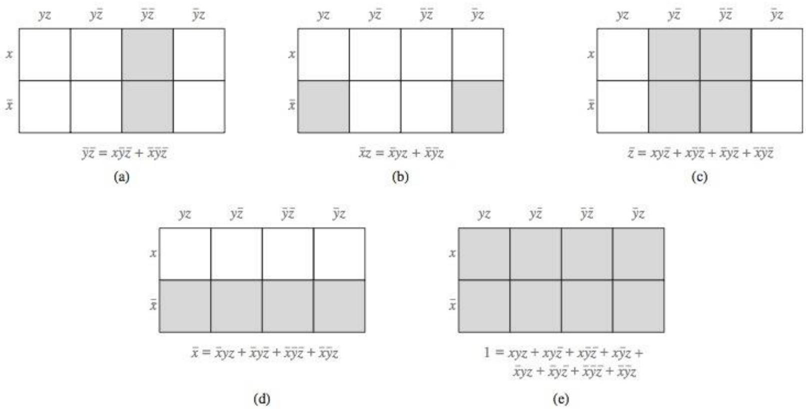

FIGURA 6 **Blocos em Mapas K em Três Variáveis.**

seja, a função 1. Na Figura 6, são mostrados os blocos 1×2, 2×1, 2×2, 4×1 e 4×2 e os produtos que eles representam.

O produto de literais que correspondem a um bloco só de 1s é chamado de **implicante** da função que está sendo minimizada. Ele é chamado de **implicante primo** se esse bloco de 1s não estiver contido em um bloco maior de 1s que representa o produto de menos literais que neste produto.

O objetivo é identificar os maiores blocos possíveis no mapa e cobrir todos os 1s no mapa com o menor número de blocos, usando os blocos maiores primeiro. Os maiores blocos possíveis são sempre escolhidos, mas podemos sempre escolher um bloco se ele for o único de 1s que cobre um 1 no mapa K. Esse bloco representa um **implicante primo essencial**. Cobrindo todos os 1s no mapa com blocos que correspondem a implicantes primos, podemos expressar a soma de produtos como uma soma de implicantes primos. Observe que pode existir mais de uma maneira de cobrir todos os 1s usando o menor número possível de blocos.

O Exemplo 3 ilustra como os mapas K em três variáveis são usados.

EXEMPLO 3 Use mapas K para minimizar estas expansões em soma de produtos.

(a) $xy\bar z + x\bar y\bar z + \bar x yz + \bar x\bar y\bar z$

(b) $x\bar y z + x\bar y\bar z + \bar x yz + \bar x\bar y z + \bar x\bar y\bar z$

(c) $xyz + xy\bar z + x\bar y z + x\bar y\bar z + \bar x yz + \bar x\bar y z + \bar x\bar y\bar z$

(d) $xy\bar z + x\bar y\bar z + \bar x\bar y z + \bar x\bar y\bar z$

Solução: Os mapas K para essas expansões em soma de produtos são mostrados na Figura 7. O agrupamento de blocos mostra que expansões mínimas em somas booleanas de produtos boolea-nos são (a) $x\bar z + \bar y\bar z + \bar x yz$, (b) $\bar y + \bar x z$, (c) $x + \bar y + z$ e (d) $x\bar z + \bar x\bar y$. Na parte (d), observe que os implicantes primos $x\bar z$ e $\bar x\bar y$ são implicantes primos essenciais, mas o implicante primo $\bar y\bar z$ é um implicante primo que não é essencial, pois as células que ele cobre são cobertas por outros dois implicantes primos. ◀

Um mapa K em quatro variáveis é um quadrado que é dividido em 16 células. As células representam os 16 mintermos possíveis em 4 variáveis. Uma das maneiras de formar um mapa K em quatro variáveis é mostrada na Figura 8.

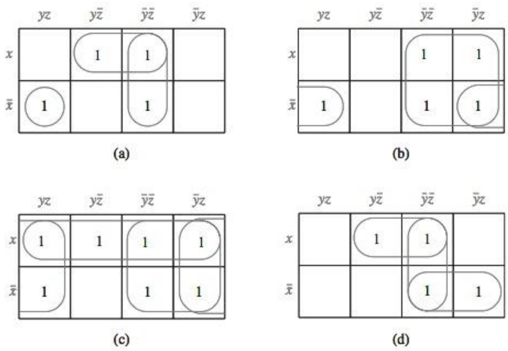

FIGURA 7 O Uso de Mapas K em Três Variáveis.

Duas células são adjacentes se e somente se os mintermos que elas representam diferem apenas em um literal. Conseqüentemente, cada célula é adjacente a quatro outras células. O mapa K de uma expansão em soma de produtos em quatro variáveis pode ser considerado como estando em um toro, de modo que células adjacentes tenham uma fronteira comum (veja o Exercício 28). A simplificação de uma expansão em soma de produtos em quatro variáveis é feita identificando aqueles blocos de 2, 4, 8 ou 16 células que representam mintermos que podem ser combinados. Cada célula que representa um mintermo deve ou ser utilizada para formar um produto usando menos literais ou ser incluída na expansão. Na Figura 9, são mostrados alguns exemplos de blocos que representam produtos de três literais, produtos de dois literais e um único literal.

Como no caso dos mapas K em duas e três variáveis, o objetivo é identificar os maiores blocos de 1s no mapa, que correspondem aos primos implicantes, e cobrir todos os 1s com o mínimo necessário de blocos, usando sempre os maiores blocos possíveis primeiro. O Exemplo 4 ilustra como os mapas K em quatro variáveis são usados.

EXEMPLO 4 Use mapas K para simplificar estas expansões em soma de produtos.

(a) $wxyz + wxy\bar{z} + wx\bar{y}\bar{z} + w\bar{x}yz + w\bar{x}\bar{y}z + w\bar{x}\bar{y}\bar{z} + \bar{w}x\bar{y}z +$
$\bar{w}\bar{x}yz + \bar{w}\bar{x}y\bar{z}$
(b) $wx\bar{y}\bar{z} + w\bar{x}yz + w\bar{x}y\bar{z} + w\bar{x}\bar{y}\bar{z} + \bar{w}x\bar{y}\bar{z} + \bar{w}\bar{x}y\bar{z} + \bar{w}\bar{x}\bar{y}\bar{z}$
(c) $wxy\bar{z} + wx\bar{y}\bar{z} + w\bar{x}yz + w\bar{x}y\bar{z} + w\bar{x}\bar{y}\bar{z} + \bar{w}xyz + \bar{w}xy\bar{z} + \bar{w}x\bar{y}\bar{z} +$
$\bar{w}x\bar{y}z + \bar{w}\bar{x}y\bar{z} + \bar{w}\bar{x}\bar{y}\bar{z}$

	yz	$y\bar{z}$	$\bar{y}\bar{z}$	$\bar{y}z$
wx	$wxyz$	$wxy\bar{z}$	$wx\bar{y}\bar{z}$	$wx\bar{y}z$
$w\bar{x}$	$w\bar{x}yz$	$w\bar{x}y\bar{z}$	$w\bar{x}\bar{y}\bar{z}$	$w\bar{x}\bar{y}z$
$\bar{w}\bar{x}$	$\bar{w}\bar{x}yz$	$\bar{w}\bar{x}y\bar{z}$	$\bar{w}\bar{x}\bar{y}\bar{z}$	$\bar{w}\bar{x}\bar{y}z$
$\bar{w}x$	$\bar{w}xyz$	$\bar{w}xy\bar{z}$	$\bar{w}x\bar{y}\bar{z}$	$\bar{w}x\bar{y}z$

FIGURA 8 Mapas K em Quatro Variáveis.

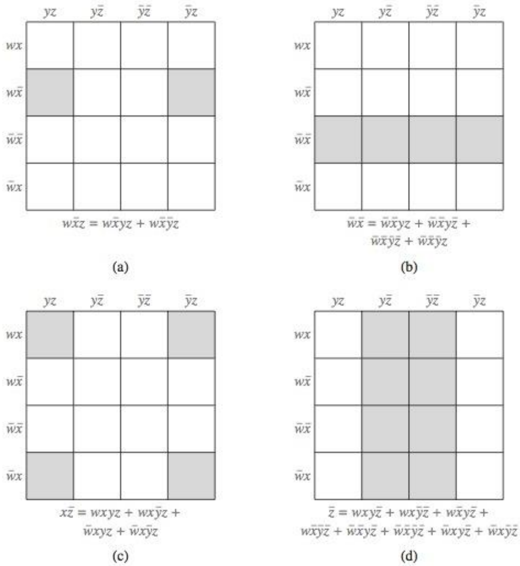

FIGURA 9 **Blocos em Mapas K em Quatro Variáveis.**

Solução: Os mapas K para essas expansões são exibidos na Figura 10. Usar os blocos mostrados leva à soma de produtos (a) $wyz + wx\bar{z} + w\bar{x}\bar{y} + \bar{w}\bar{x}y + \bar{w}x\bar{y}z$, (b) $\bar{y}\,\bar{z} + w\bar{x}y + \bar{x}\bar{z}$ e (c) $\bar{z} + \bar{w}x + w\bar{x}y$. O leitor deve determinar se existem outras escolhas de blocos em cada parte, que levem a somas de produtos diferentes que representem essas funções booleanas. ◀

Os mapas K podem ser usados realisticamente para minimizar funções booleanas com cinco ou seis variáveis, mas, além disso, eles raramente são usados, pois se tornam extremamente complicados. Entretanto, os conceitos usados nos mapas K desempenham papel importante em algoritmos mais novos. Além disso, dominar esses conceitos o ajudará a entender esses algoritmos mais novos e programas para projetos, com o auxílio do computador (CAD), que os implementam. À

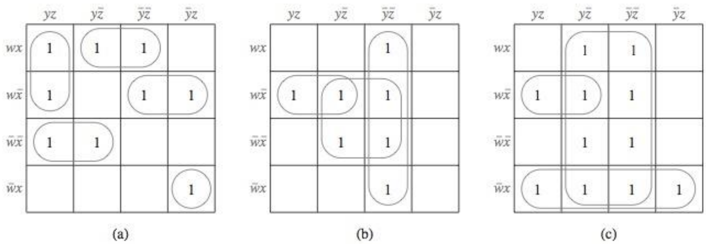

FIGURA 10 **O Uso de Mapas K em Quatro Variáveis.**

medida que desenvolvermos esses conceitos, seremos capazes de ilustrá-los referindo-nos novamente a nossa discussão de minimização de funções booleanas em três e quatro variáveis.

Os mapas K que usamos para minimizar as funções booleanas em duas, três e quatro variáveis são construídos com retângulos 2×2, 2×4 e 4×4, respectivamente. Além disso, células correspondentes na linha de cima e na linha de baixo e na coluna mais à esquerda e na coluna mais à direita em cada um desses casos são consideradas adjacentes, pois representam mintermos que diferem em apenas um literal. Podemos construir mapas K para minimizar funções booleanas em mais de quatro variáveis de modo parecido. Usamos um retângulo com $2^{\lfloor n/2 \rfloor}$ linhas e $2^{\lceil n/2 \rceil}$ colunas. (Esses mapas K contêm 2^n células, porque $\lceil n/2 \rceil + \lfloor n/2 \rfloor = n$.) As linhas e as colunas precisam ser posicionadas de modo que as células que representam mintermos que difiram em apenas um literal sejam adjacentes ou sejam consideradas adjacentes pela especificação de adjacências adicionais para linhas e colunas. Para ajudar (mas não completamente) a alcançar isso, as linhas e as colunas de um mapa K são dispostas usando códigos de Gray (veja a Seção 9.5), em que associamos seqüências de bits e produtos, especificando que um 1 corresponde ao aparecimento de uma variável e um 0, ao aparecimento de seu complemento. Por exemplo, em um mapa K de dimensão 10, o código de Gray 01110 usado para rotular uma linha corresponde ao produto $\bar{x}_1 x_2 x_3 x_4 \bar{x}_5$.

EXEMPLO 5 Os mapas K que usamos para minimizar funções booleanas com quatro variáveis têm duas linhas e duas colunas. Ambas as linhas e as colunas são dispostas usando o código de Gray 11,10,00,01. As linhas representam produtos $w\,x$, $w\,\bar{x}$, $\bar{w}\,\bar{x}$ e $\bar{w}\,x$, respectivamente, e as colunas correspondem aos produtos yz, $y\bar{z}$, $\bar{y}\bar{z}$ e $\bar{y}z$, respectivamente. Usando códigos de Gray e as células adjacentes na primeira e na última linhas e na primeira e na última colunas, garantimos que os mintermos que diferem em apenas uma variável sejam sempre adjacentes. ◀

EXEMPLO 6 Para minimizar funções booleanas em cinco variáveis, usamos mapas K com $2^3 = 8$ colunas e $2^2 = 4$ linhas. Rotulamos as quatro linhas usando o código de Gray 11,10,00,01, que corresponde a x_1x_2, $x_1\bar{x}_2$, $\bar{x}_1\bar{x}_2$ e $\bar{x}_1x_2$, respectivamente. Rotulamos as oito colunas usando o código de Gray 111,110,100,101,001,000,010,011, que corresponde aos termos $x_3x_4x_5$, $x_3x_4\bar{x}_5$, $x_3\bar{x}_4\bar{x}_5$, $x_3\bar{x}_4x_5$, $\bar{x}_3\bar{x}_4x_5$, $\bar{x}_3\bar{x}_4\bar{x}_5$, $\bar{x}_3x_4\bar{x}_5$ e $\bar{x}_3x_4x_5$, respectivamente. Usar códigos de Gray para rotular colunas e linhas garante que os mintermos representados por células adjacentes difiram apenas em uma variável. Entretanto, para garantir que todas as células que representam produtos que diferem em apenas uma variável sejam consideradas adjacentes, consideramos células nas linhas de cima e de baixo como sendo adjacentes, bem como as células na primeira e na oitava colunas, na primeira e na quarta colunas, na segunda e na sétima colunas, na terceira e na sexta colunas e na quinta e na oitava colunas (como o leitor deve verificar). ◀

Para usar um mapa K para minimizar uma função booleana em n variáveis, primeiro desenhamos um mapa do tamanho apropriado. Colocamos 1s em todas as células que correspondam aos mintermos na expansão em soma de produtos dessa função. A seguir, identificamos todos os implicantes primos da função booleana. Para fazer isso, procuramos blocos que consistam em 2^k células agrupadas, todas com um 1, em que $1 \leq k \leq n$. Esses blocos correspondem ao produto de $n - k$ literais. (O Exercício 33 pede que o leitor verifique isso.) Além disso, um bloco de 2^k células cada uma com um 1 que não esteja contido em um bloco de 2^{k+1} células, cada uma das quais com um 1, representa um implicante primo. A razão de esse implicante ser um implicante primo é que nenhum produto obtido removendo um literal também é representado por um bloco de células, todas com 1s.

EXEMPLO 7 Um bloco de oito células que representa o produto de dois literais em um mapa K para minimizar funções booleanas em cinco variáveis, todas com 1s, é um implicante primo se ele não estiver contido em um bloco maior de 16 células, todas com 1s que representam um único literal. ◀

Uma vez que todos os implicantes primos tenham sido identificados, o objetivo é encontrar o menor subconjunto possível desses implicantes primos com a propriedade de que as células que os representam cobrem todas as células que contêm um 1 no mapa K. Começamos escolhendo implicantes primos essenciais, pois cada um destes é representado por um bloco que cobre uma célula que contém um 1 e que não está coberta por nenhum outro implicante primo. Adicionamos implicantes primos para garantir que todos os 1s no mapa K sejam cobertos. Quando o número de variáveis é grande, este último passo pode se tornar extremamente complicado.

Condições Indiferentes (Don't care)

Links

Em determinados circuitos, nos importamos apenas com a saída para algumas combinações de valores de entrada, pois as outras combinações de valores de entrada não são possíveis ou nunca ocorrem. Isso nos dá liberdade na produção de um circuito simples com a saída desejada, pois os valores de saída para todas as combinações que nunca ocorrem podem ser escolhidos arbitrariamente. Os valores da função para estas combinações são chamados de **condições indiferentes** (ou ***don't care***). Um d é usado em um mapa K para marcar aquelas combinações de valores das variáveis para as quais a função pode ser arbitrariamente definida. No processo de minimização, podemos associar 1s como valores àquelas combinações de valores de entrada que levem a blocos maiores no mapa K. Isso é ilustrado no Exemplo 8.

EXEMPLO 8 Uma maneira de codificar expansões decimais usando bits é usar os quatro bits da expansão binária de cada algarismo na expansão decimal. Por exemplo, 873 é codificado como 100001110011. Essa codificação de uma expansão decimal é chamada de **expansão decimal codificada binária**. Como existem 16 blocos de quatro bits e apenas 10 algarismos, existem seis combinações de quatro bits que não são usadas para codificar algarismos. Suponha que seja construído um circuito que produza uma saída de 1 se o algarismo decimal for maior que ou igual a 5 e uma saída 0 se o algarismo decimal for menor que 5. Como esse circuito pode ser construído de modo simples usando portas *OU*, portas *E* e inversores?

Solução: Indique por $F(w, x, y, z)$ a saída do circuito, em que $wxyz$ é uma expansão binária de um algarismo decimal. Os valores de F estão mostrados na Tabela 1. O mapa K para F, com ds nas posições *don't care*, é exibido na Figura 11(a). Podemos ou incluir ou excluir quadrados com ds dos blocos. Isso nos dá muitas escolhas possíveis para os blocos. Por exemplo, excluir todos os quadrados com ds e formar blocos, como mostrado na Figura 11(b), produz a expressão $w\overline{x}\overline{y} + \overline{w}xy + \overline{w}xz$. Incluir alguns dos ds, excluir outros e formar blocos, como mostrado na Figura 11(c), produz a expressão $w\overline{x} + \overline{w}xy + x\overline{y}z$. Finalmente, incluir todos os ds e usar os blocos mostrados na Figura 11(d) produz a expansão em soma de produtos mais simples possível, a saber, $F(x, y, z) = w + xy + xz$. ◀

TABELA 1

Algarismo	*w*	*x*	*y*	*z*	*F*
0	0	0	0	0	0
1	0	0	0	1	0
2	0	0	1	0	0
3	0	0	1	1	0
4	0	1	0	0	0
5	0	1	0	1	1
6	0	1	1	0	1
7	0	1	1	1	1
8	1	0	0	0	1
9	1	0	0	1	1

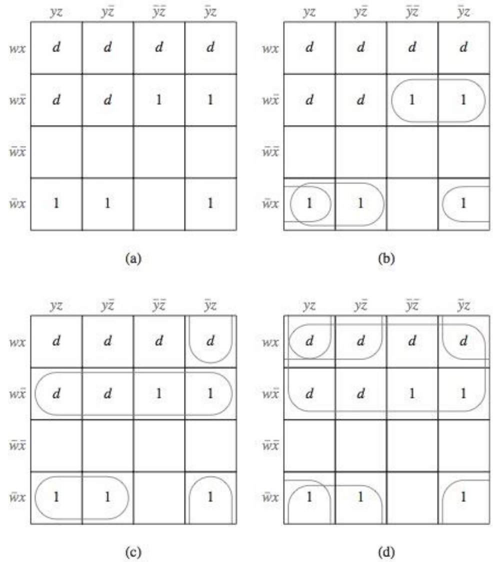

FIGURA 11 O Mapa K para *F* Mostrando Suas Posições *Don't Care*.