

Capítulo 3

Álgebra Booleana

Definição 3.1. A $(\mathfrak{B} = \{0, 1\}, \cdot, +, \sim)$ especifica um alfabeto composto apenas pelos números zero - 0 e um - 1 e as operações binárias \cdot "e" lógico, $+$ "ou" lógico e a operação unária \sim negação.

Na álgebra Booleana, a exemplo da Euclideana¹, as operações válidas são definidas pela semântica dos operadores fundamentais previstas diretamente na definição da álgebra assim como pelos seus postulados, propriedades e teoremas.

3.1 Operações Fundamentais

Como descrito na definição anterior, a álgebra Booleana se baseia em apenas três, nomeadamente operação E, operação OU e operação NÃO. Note que há uma correspondência direta entre as operações fundamentais previstas pela álgebra booleana, pelas portas lógicas fundamentais vistas no capítulo 2 e os operadores da lógica proposicional.

3.2 Variáveis Booleanas

Variáveis Booleanas são receptáculos que podem assumir um dos dois números previstos pelo alfabeto $\mathfrak{B} = \{0, 1\}$.

Por convenção, utiliza-se letras maiúsculas (ex. A, B, C, \dots, X, Y, Z) para representar variáveis Booleanas. Eventualmente, também é costumeiro utilizar subscritos (ex. A_1, B_2) na representação de variáveis booleanas.

¹A Álgebra Euclideana é aquela aprendida no ensino fundamental e médio.

3.3 Funções e Constantes Booleanas

Funções Booleanas são definidas por uma ou múltiplas operações fundamentais e/ou compostas entre variáveis Booleanas. Sintaticamente elas são definidas de duas formas equivalentes entre si.

Na primeira forma utiliza-se uma letra em maiúscula, geralmente F, G, H, \dots para nomear a função. A seguir as variáveis que afiguram na função são listadas entre parênteses seguido do sinal "=" e a expressão booleana que a define propriamente dita.

Exemplo 3.1. $F(A, B, C) = (\overline{ABC} + \overline{AB}) \oplus (B \odot C)$

A álgebra Booleana permite apenas duas constantes que são efetivamente os símbolos previstos em seu alfabeto. Elas podem aparecer diretamente na expressão booleana tal como no exemplo a seguir:

Exemplo 3.2. $F(A, B, C, D) = (\overline{ABC} + \overline{C \oplus D}) \odot (D + 1)$

3.4 Operações Complementares

Similarmente ao que ocorre na álgebra Euclideana na qual uma série de operadores complementares são definidos (ex. a potencialização $x^2 = x \cdot x$) quatro operações complementares são definidas na álgebra Booleana. Tais complementares são efetivamente funções Booleanas que devido a recorrência de utilização recebem nomes e sintaxe próprias.

NÃO-E: A primeira operação complementar é chamada de NÃO-E. Como vimos no capítulo 2 ela refere-se a aplicação da operação NÃO imediatamente após a operação E.

Algebricamente ela é definida como:

$$\text{NÃO-E} = F(A, B) = \overline{A \cdot B}$$

NÃO-OU: Similarmente, a segunda complementar é chamada de NÃO-OU e composta por uma operação NÃO imediatamente após a operação OU.

Algebricamente ela é definida como:

$$\text{NÃO-OU} = F(A, B) = \overline{A + B}$$

OU-EXCLUSIVO: A terceira das operações complementares, chamada de OU-EXCLUSIVO, ou "operação diferença". Semanticamente ela produz saída "1" se uma e apenas uma das entradas for um mas não ambas.

Algebricamente ela é definida como:

$$\text{OU-EXCLUSIVO } F(A, B) = A \oplus B \equiv (\overline{AB} + A\overline{B})$$

NÃO-OU-EXCLUSIVO: A última das operações complementares, chamada de NÃO-OU-EXCLUSIVO, ou "operação igualdade". Semanticamente ela produz saída "1" se uma e apenas entradas forem iguais.

Algebricamente ela é definida como:

$$\text{NÃO-OU-EXCLUSIVO } F(A, B) = A \odot B \equiv (\overline{A} \cdot \overline{B} + A \cdot B)$$

3.5 Precedência entre Operadores

As regras de precedência de operações são simples:

1. A negação de uma variável é sempre avaliada primeiro;
2. Expressões Booleanas devem ser avaliadas preferencialmente da esquerda para a direita;
3. Negação de mais de uma variável é avaliada apenas após a operação ou operações que estão sendo avaliadas;
4. As operações OU e E possuem a mesma ordem de precedência;
5. A ordem de avaliação pode ser alterada por meio da utilização de parênteses.

Exemplo 3.3. $((ABC) + (\overline{A}(\overline{BC}))) \oplus (\overline{(\overline{AD})} + (\overline{D \oplus A}))$

1. $A \cdot B$
2. $(A \cdot B) \cdot C$
3. \overline{A}
4. $B \cdot C$
5. $\overline{A} \cdot (B \cdot C)$
6. $((A \cdot B) \cdot C) + (\overline{A} \cdot (\overline{B \cdot C}))$
7. $A \cdot D$
8. $(\overline{A \cdot D})$

9. $A \oplus D$
10. $\overline{(A \oplus D)}$
11. $\overline{(A \cdot D)} + \overline{(A \oplus D)}$
12. $\overline{\overline{(A \cdot D)} + \overline{(A \oplus D)}}$
13. $((ABC) + \overline{(A(BC))}) \oplus (\overline{(\overline{AD})} + \overline{(D \oplus A)})$

3.6 Universalidade das Portas NÃO-E e NÃO-OU

Como visto anteriormente, é possível representar todas as operações booleanas compostas a partir das três operações fundamentais. Também podemos, via manipulação algébrica, produzir expressões booleanas equivalentes às operações fundamentais a partir das operações compostas. Em especial, as operações NÃO-E e NÃO-OU são de especial interesse. Dizemos que as portas NÃO-E e NÃO-OU são autossuficientes pois, utilizando-as apenas, pode-se construir equivalências para todas as outras operações, fundamentais ou compostas.

A correspondência entre os arranjos de NÃO-Es e NÃO-OUs e as operações fundamentais serão derivadas diretamente a partir das tabelas verdade. Para fins de referência, deve-se consultar as tabelas verdade 2.4 e 2.5.

3.6.1 Operação NÃO a partir de NÃO-Es

Ao observarmos a tabela verdade da operação NÃO-E (2.4) observa-se que a primeira e última linhas ($0\ 0 \rightarrow 1$ e $1\ 1 \rightarrow 0$) apresentam um comportamento interessante. Caso ambas as entradas sejam iguais a zero, a saída será um e inversamente, caso ambas as entradas sejam um, a saída será zero.

Note ainda que a operação NÃO-E é binária, ou seja, prevê dois operandos e a operação NÃO é unária. Logicamente se conclui que para que seja construída, um dos operandos da NÃO-E deve ser atribuída diretamente a uma constante ou a mesma entrada.



Figura 3.1: Porta NÃO construída a partir de uma porta NÃO-E.

Algebricamente, a equivalência pode ser descrita como na equação 3.1.

$$\overline{A \cdot A} = \overline{A} \quad (3.1)$$

3.6.2 Operação E a partir de NÃO-Es

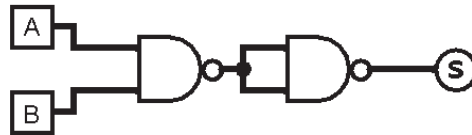


Figura 3.2: Porta E construída a partir de uma porta NÃO-E.

Algebricamente, a equivalência pode ser descrita como na equação 3.2.

$$\overline{(\overline{A \cdot B}) \cdot (\overline{A \cdot B})} = A \cdot B \quad (3.2)$$

3.6.3 Operação OU a partir de NÃO-Es

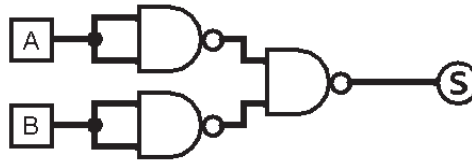


Figura 3.3: Porta OU construída a partir de uma porta NÃO-E.

Algebricamente, a equivalência pode ser descrita como na equação 3.3.

$$\overline{(\overline{A \cdot A}) \cdot (\overline{B \cdot B})} = A + B \quad (3.3)$$

3.6.4 Operação NÃO a partir de NÃO-OUs



Figura 3.4: Porta NÃO construída a partir de uma porta NÃO-OUs.

Algebricamente, a equivalência pode ser descrita como na equação 3.4.

$$\overline{A + A} = \overline{A} \quad (3.4)$$

3.6.5 Operação E a partir de NÃO-OUs

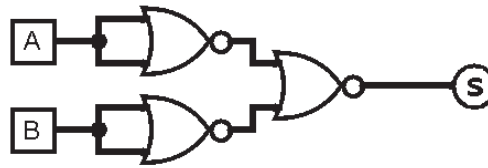


Figura 3.5: Porta OU construída a partir de uma porta NÃO-OUs.

Algebricamente, a equivalência pode ser descrita como na equação 3.5.

$$\overline{(\overline{A \cdot A}) \cdot (\overline{B \cdot B})} = A \cdot B \quad (3.5)$$

3.6.6 Operação OU a partir de NÃO-OUs

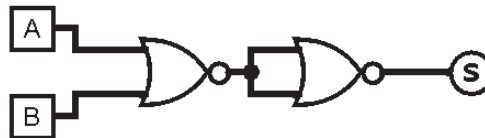


Figura 3.6: Porta OU construída a partir de uma porta NÃO-OUs.

Algebricamente, a equivalência pode ser descrita como na equação 3.6.

$$\overline{(\overline{A + B}) + (\overline{A + B})} = A + B \quad (3.6)$$

Embora pareça um exercício puramente teórico, a universalidade ou suficiência das portas NÃO-E e NÃO-OU são de importância relevada. Na prática, portas NÃO-E e NÃO-OU são fáceis de construir fisicamente utilizando apenas transistores (circuitos integrados do tipo TTL - Transistor Transistor Logic). Consequentemente, o custo associado a sua produção é inferior em comparação a outras portas lógicas, fato derivado diretamente do número de transistores necessários para sua construção.

3.7 Propriedades e Teoremas da Álgebra de Boole

1. Postulado da Complementação -

Considerando a variável A , um de dois possíveis casos pode ocorrer:

$$\text{Se } A = 0 \longrightarrow \bar{A} = 1$$

$$\text{Se } A = 1 \longrightarrow \bar{A} = 0$$

similarmente:

$$\text{Se } A = 0 \longrightarrow \overline{\bar{A}} = 0$$

$$\text{Se } \bar{A} = 1 \longrightarrow \overline{\bar{A}} = 1$$

$$\therefore \overline{\bar{A}} = A \text{ (P01)}$$

2. Postulado da "União" -

Considerando os quatro possíveis resultados produzidos pela operação binária OU:

$$\text{Se } 0 + 0 \longrightarrow 0 \text{ (Caso 1)}$$

$$\text{Se } 0 + 1 \longrightarrow 1 \text{ (Caso 2)}$$

$$\text{Se } 1 + 0 \longrightarrow 1 \text{ (Caso 3)}$$

$$\text{Se } 1 + 1 \longrightarrow 1 \text{ (Caso 4)}$$

Considerando a expressão $A + 0$ onde A é uma variável Booleana qualquer e 0 é a constante Booleana "0":

$$\text{Se } A = 0 \longrightarrow 0 + 0 = 0 \text{ (Caso 1)}$$

$$\text{Se } A = 1 \longrightarrow 1 + 0 = 1 \text{ (Caso 3)}$$

$$\therefore A + 0 = A \text{ (P02)}$$

Considerando a expressão $A + 1$:

$$\text{Se } A = 0 \longrightarrow 0 + 1 = 1 \text{ (Caso 2)}$$

$$\text{Se } A = 1 \longrightarrow 1 + 1 = 1 \text{ (Caso 4)}$$

$$\therefore A + 1 = 1 \text{ (P03)}$$

Considerando a expressão $A + A$:

$$\text{Se } A = 0 \longrightarrow 0 + 0 = 0 \text{ (Caso 1)}$$

$$\text{Se } A = 1 \longrightarrow 1 + 1 = 1 \text{ (Caso 4)}$$

$$\therefore A + A = A \text{ (P04)}$$

Considerando a expressão $A + \bar{A}$:

$$\text{Se } A = 0 \longrightarrow 0 + 1 = 1 \text{ (Caso 2)}$$

Se $A = 1 \longrightarrow 1 + 0 = 1$ (Caso 3)
 $\therefore \mathbf{A + \bar{A} = 1}$ (P05)

3. Postulado da "Disjunção" -

Considerando os quatro possíveis resultados produzidos pela operação binária E:

Se $0 \cdot 0 \longrightarrow 0$ (Caso 5)

Se $0 \cdot 1 \longrightarrow 0$ (Caso 6)

Se $1 \cdot 0 \longrightarrow 0$ (Caso 7)

Se $1 \cdot 1 \longrightarrow 1$ (Caso 8)

Considerando a expressão $A \cdot 0$ onde A é uma variável Booleana qualquer e 0 e a constante Booleana "0":

Se $A = 0 \longrightarrow 0 \cdot 0 = 0$ (Caso 5)

Se $A = 1 \longrightarrow 1 \cdot 0 = 0$ (Caso 7)

$\therefore \mathbf{A \cdot 0 = 0}$ (P06)

Considerando a expressão $A \cdot 1$:

Se $A = 0 \longrightarrow 0 \cdot 1 = 0$ (Caso 6)

Se $A = 1 \longrightarrow 1 \cdot 1 = 1$ (Caso 8)

$\therefore \mathbf{A \cdot 1 = A}$ (P07)

Considerando a expressão $A \cdot A$:

Se $A = 0 \longrightarrow 0 \cdot 0 = 0$ (Caso 5)

Se $A = 1 \longrightarrow 1 \cdot 1 = 1$ (Caso 8)

$\therefore \mathbf{A \cdot A = A}$ (P08)

Considerando a expressão $A \cdot \bar{A}$:

Se $A = 0 \longrightarrow 0 \cdot 1 = 0$ (Caso 2)

Se $A = 1 \longrightarrow 1 \cdot 0 = 0$ (Caso 3)

$\therefore \mathbf{A \cdot \bar{A} = 0}$ (P09)

4. Propriedade Comutativa - dita que a ordem dos operandos das operações E e OU não altera o resultado.

Considerando a expressão (a) $A \cdot B$ em que a ordem dos operandos seja considerada e que A e B sejam variáveis Booleanas:

Se $A = 0$ e $B = 0 \longrightarrow 0 \cdot 0 = 0$ (Caso 1)

Se $A = 0$ e $B = 1 \longrightarrow 0 \cdot 1 = 0$ (Caso 2)

Se $A = 1$ e $B = 0 \longrightarrow 1 \cdot 0 = 0$ (Caso 3)

Se $A = 1$ e $B = 1 \longrightarrow 1 \cdot 1 = 1$ (Caso 4)

Considerando a expressão (b) $B \cdot A$ em que a ordem dos operandos seja considerada:

Se $B = 0$ e $A = 0 \longrightarrow 0 \cdot 0 = 0$ (Caso 1)

Se $B = 0$ e $A = 1 \longrightarrow 0 \cdot 1 = 0$ (Caso 2)

Se $B = 1$ e $A = 0 \longrightarrow 1 \cdot 0 = 0$ (Caso 3)

Se $B = 1$ e $A = 1 \longrightarrow 1 \cdot 1 = 1$ (Caso 4)

Observa-se por pareamento de casos que indiferentemente da ordem em que os operandos sejam postos na operação E o resultado produzido será o mesmo, $\therefore \mathbf{A \cdot B = B \cdot A}$ (P10)

Considerando a expressão (a) $A + B$ em que a ordem dos operandos seja considerada e que A e B sejam variáveis Booleanas:

Se $A = 0$ e $B = 0 \longrightarrow 0 + 0 = 0$ (Caso 5)

Se $A = 0$ e $B = 1 \longrightarrow 0 + 1 = 1$ (Caso 6)

Se $A = 1$ e $B = 0 \longrightarrow 1 + 0 = 1$ (Caso 7)

Se $A = 1$ e $B = 1 \longrightarrow 1 + 1 = 1$ (Caso 8)

Considerando a expressão (b) $B + A$ em que a ordem dos operandos seja considerada:

Se $B = 0$ e $A = 0 \longrightarrow 0 + 0 = 0$ (Caso 5)

Se $B = 0$ e $A = 1 \longrightarrow 0 + 1 = 1$ (Caso 6)

Se $B = 1$ e $A = 0 \longrightarrow 1 + 0 = 1$ (Caso 7)

Se $B = 1$ e $A = 1 \longrightarrow 1 + 1 = 1$ (Caso 8)

Observa-se por pareamento de casos que indiferentemente da ordem em que os operandos sejam postos na operação OU o resultado produzido será o mesmo, $\therefore \mathbf{A + B = B + A}$ (P11)

5. **Propriedade Associativa** - dita que a ordem de avaliação das operações E e OU em expressões onde múltiplas (duas ou mais operações de mesmo tipo) não influencia o resultado final da expressão: //

$$\mathbf{A \cdot (B \cdot C) \equiv (A \cdot B) \cdot C}$$
 (P12)

$$\mathbf{A + (B + C) \equiv (A + B) + C}$$
 (P13)

6. **Propriedade Distributiva** - dita que a disjunção de uma variável pode ser distribuída na conjunção de outras duas variáveis. Algebricamente:

$$A \cdot (B + C) \equiv (A \cdot B) + (A \cdot C) \text{ (P14)}$$

Similarmente, dita que a conjunção de uma variável pode ser distribuída na disjunção de outras duas variáveis. Algebricamente:

$$A + (B \cdot C) \equiv (A + B) \cdot (A + C) \text{ (P15)}$$

De forma compacta, a propriedade pode ser demonstrada através de uma tabela verdade comparativa, tal como apresentado na tabela 3.1.

Tabela 3.1: Tabela verdade que demonstra a equivalência da propriedade (P15) - $A + (B \cdot C) \equiv (A + B) \cdot (A + C)$

A	B	C	$B \cdot C$	$A + B$	$A + C$	$(A + B) \cdot (A + C)$	$A + (B \cdot C)$
0	0	0	0	0	0	0	0
0	0	1	0	0	1	0	0
0	1	0	0	1	0	0	0
0	1	1	1	1	1	1	1
1	0	0	0	1	1	1	1
1	0	1	0	1	1	1	1
1	1	0	0	1	1	1	1
1	1	1	1	1	1	1	1

A propriedade distributiva também pode ser estendida para a distribuição de um par de disjunções. Algebricamente:

$$(A \cdot B) + (C \cdot D) \equiv (A + C) \cdot (B + C) \cdot (A + D) \cdot (B + D) \text{ (P16)}$$

Algebricamente:

renomeia-se a termo $A \cdot B = X \longrightarrow X + (A \cdot B)$

aplicando a propriedade (P14) obtêm-se $(X + C) \cdot (X + D)$

substituindo-se de volta a variável X por $A \cdot B$ obtêm-se:

$$((A \cdot B) + C) \cdot ((A \cdot B) + D)$$

aplicando-se a propriedade (P14) a ambos os lados da expressão, obtêm-se:

$$(C + A) \cdot (C + B) \cdot (D + A) \cdot (D + B) \quad \square$$

De forma compacta, a propriedade pode ser demonstrada através de uma tabela verdade comparativa, tal como apresentado na tabela 3.2.

Tabela 3.2: Tabela verdade que demonstra a equivalência da propriedade (P16).

A	B	C	D	AB	CD	AB+CD	A+C	B+C	A+D	B+D	S
0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	1	1	0	0	0
0	0	1	1	0	1	1	1	1	1	1	1
0	1	0	0	0	0	0	0	1	0	1	0
0	1	0	1	0	0	0	0	1	1	1	0
0	1	1	0	0	0	0	1	1	0	1	0
0	1	1	1	0	1	1	1	1	1	1	1
1	0	0	0	0	0	0	1	0	1	0	0
1	0	0	1	0	0	0	1	0	1	1	0
1	0	1	0	0	0	0	1	1	1	0	0
1	0	1	1	0	1	1	1	1	1	1	1
1	1	0	0	0	0	1	1	1	1	1	1
1	1	0	1	1	0	1	1	1	1	1	1
1	1	1	0	1	0	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1

A expressão distribuída é renomeada renomeamos a expressão $S = (A+C) \cdot (B+C) \cdot (A+D) \cdot (B+D)$ para fins de formatação da tabela.

Uma consequência útil da propriedade distributiva e o fato de que ela pode ser aplicada inversamente gerando o fenômeno que costuma-se chamar evidenciação de uma ou mais variáveis.

Dois casos devem ser considerados quando se trata de evidenciação:

Primeiro Caso - Termos com múltiplas variáveis: Neste caso os termos a serem evidenciados possuem variáveis adicionais além da variável a ser evidenciada.

Exemplo 3.4. Considere a função Booleana $S = \overline{A}BCD + ABC\overline{D}$ composta por dois termos (em colchetes). Observa-se que ambos os termos possuem três variáveis em comum BCD . Sendo assim podemos colocá-los em evidência o que resultará na expressão $BCD(\overline{A} + A)$. Note que esta não é a única evidenciação possível. As evidenciações $B(\overline{A} \cdot \overline{C}D + A\overline{C}D)$ e $B\overline{C}(\overline{A}D + AD)$ também são válidas, porém menos relevantes pois não levam a uma simplificação direta da variável A .

Segundo Caso - Termo com única variável evidenciável Neste

caso a expressão possui ao menos um dos termos evidenciável que é composta por apenas a variável evidenciável.

Exemplo 3.5. Considere a função Booleana $\underbrace{A\bar{B}C} + \underbrace{A}$ composta por dois termos (em colchetes). Neste caso como o segundo termo é composto apenas pela variável A torna-se claro que este será a variável de interesse. A evidenciação possível seria $A(\bar{B}C + 1)$. Note que a constante 1 aparece devido a propriedade (P07) ou seja $X \cdot 1 = X$.

7. **Teorema de De Morgan** - também conhecida como a "regra da quebra de barra", dita que a conjunção negada é equivalente a disjunção das negações e similarmente que a disjunção negada é equivalente a conjunção das negações. Algebricamente:

$$\overline{A \cdot B} \equiv \bar{A} + \bar{B} \text{ (P17)}$$

$$\overline{A + B} \equiv \bar{A} \cdot \bar{B} \text{ (P18)}$$

8. **Identidades Auxiliares** - Aparte das propriedades, identidades e teoremas Booleanos apresentadas até este ponto, diversas outras expressões ocorrem frequentemente. Sendo assim elas podem ser tomadas como propriedades ou identidades auxiliares.

Considere a seguinte expressão Booleana: $A + AB \equiv A$ (P19)

Algebricamente:

Passo	Desenvolvimento	Propriedade
(0)	$A + A \cdot B = A$	(evidência)
(1)	$A \cdot (1 + B) = A$	$(X + 1 = X)$
(2)	$A \cdot 1 = A$	$(X \cdot 1 = X)$
(3)	$A = A$	□

A propriedade P18 também pode ser demonstrada através da listagem comparativa exaustiva, ou seja, pela comparação direta da tabela verdade das duas expressões a serem provadas equivalentes, tal como demonstrado na tabela 3.3.

Considere a seguinte expressão Booleana: $A + \bar{A} \cdot B \equiv A + B$ (P20)

Algebricamente:

Tabela 3.3: Tabela verdade que demonstra a equivalência da propriedade (P19).

A	B	$A \cdot B$	$A + A \cdot B$
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Passo	Desenvolvimento	Propriedade
(0)	$A + (\overline{A \cdot B}) = A + B$	$(\overline{\overline{X}} = X)$
(1)	$\overline{\overline{A + (\overline{A \cdot B})}} = A + B$	(De Morgan)
(2)	$\overline{\overline{A} \cdot (\overline{A \cdot B})} = A + B$	(De Morgan)
(3)	$\overline{\overline{A} \cdot (\overline{A + \overline{B}})} = A + B$	$(\overline{\overline{X}} = X)$
(4)	$\overline{\overline{A} \cdot (A + \overline{B})} = A + B$	(distributiva)
(5)	$\overline{(\overline{A \cdot A}) + (\overline{A \cdot \overline{B}})} = A + B$	$(X \cdot \overline{X} = 0)$
(6)	$\overline{0 + (\overline{A \cdot \overline{B}})} = A + B$	$(X + 0 = X)$
(7)	$\overline{(\overline{A \cdot \overline{B}})} = A + B$	(De Morgan)
(8)	$\overline{\overline{A \cdot \overline{B}}} = A + B$	$(\overline{\overline{X}} = X)$
(9)	$A + B = A + B$	□

3.8 Simplificação Algébrica

O processo de simplificação algébrica de funções Booleanas visa, por meio da aplicação das propriedades e teoremas Booleanos manipular a função dada de modo a produzir ao final do processo uma representação equivalente que utilize um menor número de operações fundamentais. Caso a função equivalente obtida não possa ser mais simplificada, diz-se que a expressão resultante seja mínima.

A importância da simplificação de funções Booleanas tem como motivação simples o fato de que em última instância funções Booleanas descrevem sistemas reais que potencialmente serão implementados sob a forma de eletrônica digital. A representação mínima de uma função traz vantagens reais no que tange ao número de portas lógicas e conseqüentemente, o número de transistores necessários para realizar fisicamente o sistema descrito.

A seguir alguns exemplos de simplificação algébrica são apresentados:

Exemplo 3.6. Exemplo de Simplificação:

Passo	Desenvolvimento	Propriedade
(0)	$(A + B) \cdot C + (D \cdot (B + C))$	De Morgan
(1)	$(\overline{A + B + C} + \overline{D} + (\overline{B + C}))$	De Morgan
(2)	$(\overline{A} \cdot \overline{B}) + \overline{C} + \overline{D} + (\overline{B} \cdot \overline{C})$	evidência \overline{C}
(3)	$\overline{C}(1 + \overline{B}) + \overline{A} \cdot \overline{B} + \overline{D}$	$1 + X = X$
(4)	$\overline{C} + \overline{A} \cdot \overline{B} + \overline{D}$	

Exemplo 3.7. Exemplo de Simplificação:

Passo	Desenvolvimento	Propriedade
(0)	$\overline{A} \cdot \overline{B} \cdot C + \overline{A} \cdot B \cdot C + A \cdot \overline{B} \cdot C + A \cdot B \cdot C$	evidência \overline{AC}/AC
(1)	$\overline{A} \cdot C(\overline{B} + B) + A \cdot C(\overline{B} + B)$	$X + \overline{X} = 1$
(2)	$\overline{A} \cdot C(1) + A \cdot C(1)$	$X \cdot 1 = X$
(3)	$\overline{A} \cdot C + A \cdot C$	evidência C
(4)	$C \cdot (\overline{A} + A)$	$X + \overline{X} = 1$
(5)	$C \cdot 1$	$x \cdot 1 = X$
(6)	C	

Exemplo 3.8. Exemplo de Simplificação:

Passo	Desenvolvimento	Propriedade
(0)	$AB(\overline{A} + BC)$	De Morgan
(1)	$AB(\overline{A} \cdot \overline{BC})$	De Morgan
(2)	$AB(\overline{A} \cdot (\overline{B} + \overline{C}))$	$\overline{\overline{X}} = X$
(3)	$AB(A \cdot (\overline{B} + \overline{C}))$	distributiva
(4)	$AB(A\overline{B} + A\overline{C})$	distributiva
(5)	$ABAB\overline{B} + ABAC\overline{C}$	$X \cdot X = X$
(6)	$AB\overline{B} + AB\overline{C}$	$X \cdot \overline{X} = 0$
(7)	$A0 + AB\overline{C}$	$X \cdot 0 = 0$
(8)	$0 + AB\overline{C}$	$X + 0 = X$
(9)	$AB\overline{C}$	

O processo de simplificação algébrica requer domínio das propriedades e teoremas. Durante o aprendizado, faz-se necessário ao menos o acesso pleno a todas as propriedades e teoremas. Sugere-se ao aluno que, como preparação para exercícios de simplificação algébrica, liste todas as propriedades e teoremas em uma folha de papel ou tela do computador para afigurar como instrumento de consulta.

No entanto, mesmo tal acesso pode não ser suficiente para o efetivo exercício da simplificação algébrica. Algumas dicas de como abordar o processo de simplificação podem ser úteis. São elas:

1. nos casos em que a expressão original esteja sob a forma de Soma de Produtos, vale buscar inicialmente a evidenciação de variáveis de modo a isolar possíveis situações de anulação de variáveis;
2. durante o processo de evidenciação vale buscar evidenciar o maior número de variáveis pois a chance de anulação de variáveis será maximizado;
3. expressões compostas por termos na forma soma de produtos e produto de somas concomitantemente devem ser normalizadas para uma ou outra forma. Isto pode ser conseguido através da aplicação da propriedade (P01) e do teorema de De Morgan;
4. Negações que contemplam várias variáveis devem ser estudadas quanto a aplicação do teorema de De Morgan.

As recomendações apresentadas são apenas um ponto de partida. A experiência mostra que a única forma efetiva para o desenvolvimento da habilidade de manipulação algébrica é o efetivo exercício.

3.9 Formas de Representação Booleana

Como visto no neste e no capítulo anterior, há três formas equivalentes de se representar uma expressão Booleana, nomeadamente, Tabelas Verdade, Funções Booleanas e Circuitos Lógicos. Elas são todas intercambiáveis entre si tal como esquematizado na figura 3.7.

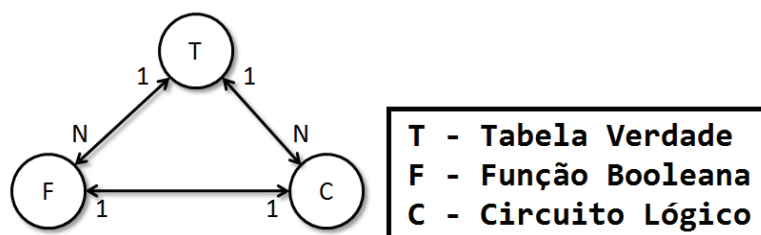


Figura 3.7: Diagrama da correspondência entre as diferentes formas de representação Booleana.

A figura ainda elucida a cardinalidade de possíveis representações equivalentes. Tome por exemplo a correspondência entre Tabelas Verdade e Funções Booleanas. Uma Tabela Verdade pode ser realizada por diversas funções Booleanas. No entanto, dada uma função Booleana específica ela produzirá

uma e apenas uma Tabela Verdade. Similarmente uma tabela produz diversos circuitos equivalentes mas um circuito produz apenas uma tabela. A única correspondência de 1 para 1 ocorre entre funções e circuitos.

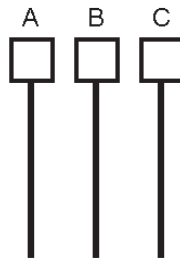
3.9.1 Função para Circuito

A conversão de uma função Booleana para um Circuito de portas lógicas requer o entendimento da precedência entre operadores. O circuito deve ser construído obedecendo a precedência de modo que eles sejam equivalentes.

1. Identifique todas as variáveis Booleanas existentes na função e desenhe uma linha vertical para cada uma delas;
2. Subdivida a função Booleana utilizando parênteses em expressões/operações Booleanas menores até o nível das operações fundamentais ou compostas;
3. Substitua as expressões Booleanas parciais por suas representações via portas lógicas;

Exemplo 3.9. Considere a expressão $F(A, B, C) = ((\bar{A} \cdot \bar{B}) + (A\bar{C})) \oplus ((\bar{A}B) + (A\bar{B}))$

Aplicando o passo 1 observa-se que a função é composta por três variáveis distintas. Desenha-se três linhas de conexão, uma para cada variável.



A seguir, a função deve ser desmembrada em seus elementos individuais. Para a função em questão seriam eles:

1. \bar{A}
2. \bar{B}
3. $(\bar{A} \cdot \bar{B})$

4. \bar{C}

5. $(A \cdot \bar{C})$

6. $((\bar{A} \cdot \bar{B}) + (A \cdot \bar{C}))$

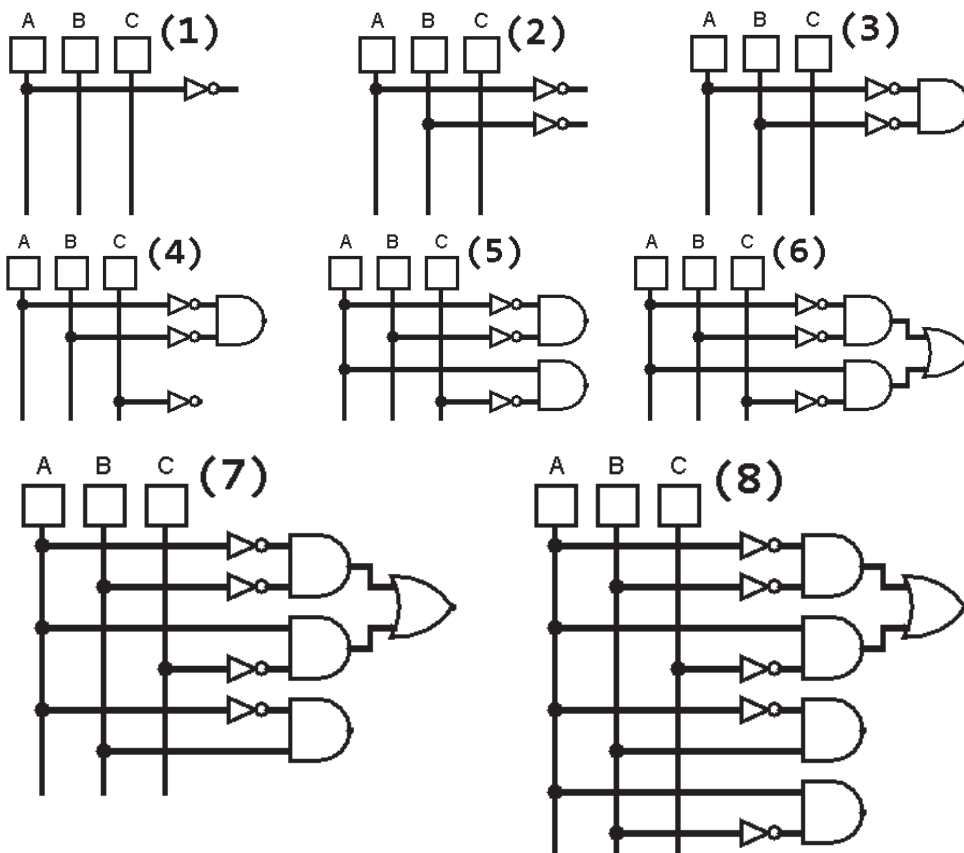
7. $(\bar{A} \cdot B)$

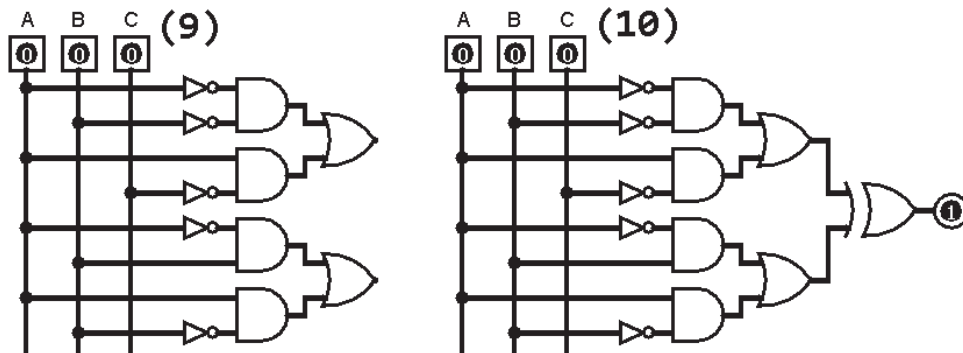
8. $(A \cdot \bar{B})$

9. $((\bar{A} \cdot B) + (A \cdot \bar{B}))$

10. $((\bar{A} \cdot \bar{B}) + (A \cdot \bar{C})) \oplus ((\bar{A} \cdot B) + (A \cdot \bar{B}))$

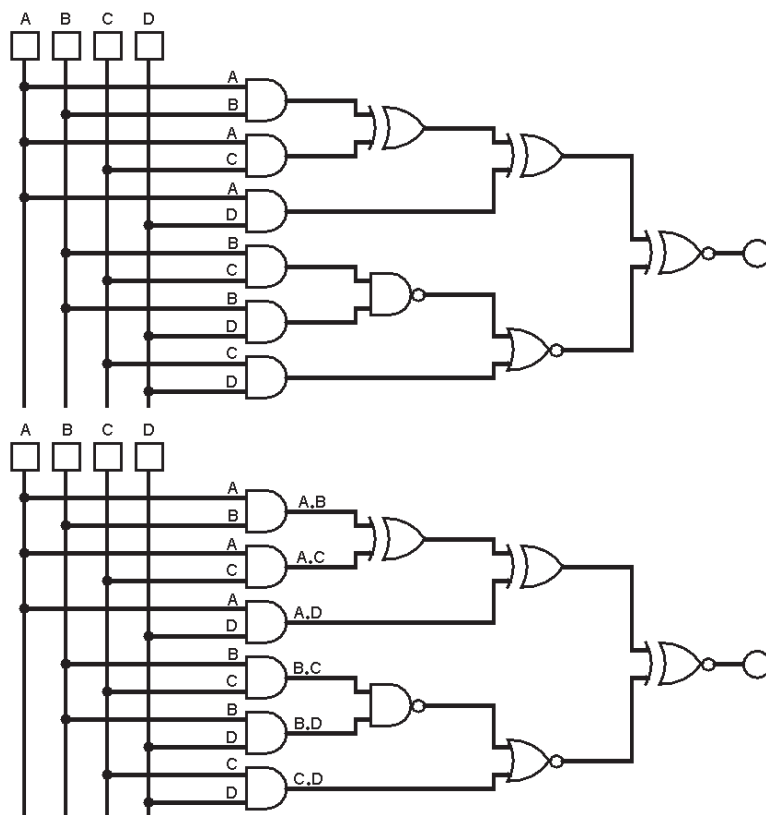
Uma vez desmembrada, a função pode ser transposta passo a passo para a forma de circuito lógico tal como apresentado a seguir.

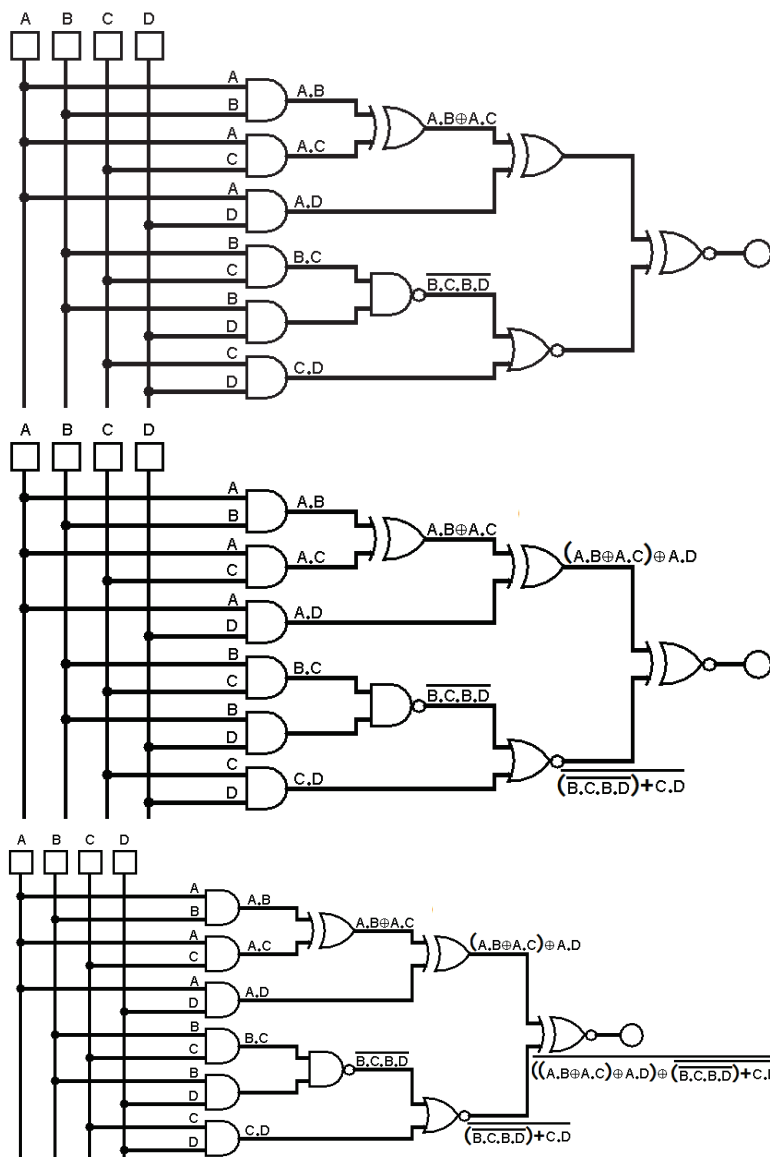




3.9.2 Circuito para Função

O procedimento para proceder a conversão de um circuito lógico para uma função Booleana dita que sigamos as linhas do circuito e escrevamos as expressões parciais diretamente no circuito. A expressão final será obtida na saída do sistema.





3.9.3 Função para Tabela

Caso a função Booleana esteja disponível e se deseje obter a tabela verdade deve-se primeiramente identificar o número de variáveis e montar uma tabela preliminar contendo todas as possíveis combinações Booleanas das entradas. A seguir, substitui-se sequencialmente, linha a linha, as variáveis por seus valores Booleanos e avalia-se a expressão.

Exemplo 3.10. Dada a função Booleana $F(A, B, C) = AB + AC + \overline{A}BC$

A	B	C	$AB + AC + \overline{ABC}$	S
0	0	0	$0 \cdot 0 + 0 \cdot 0 + 1 \cdot 0 \cdot 0$	0
0	0	1	$0 \cdot 0 + 0 \cdot 1 + 1 \cdot 0 \cdot 1$	0
0	1	0	$0 \cdot 1 + 0 \cdot 0 + 1 \cdot 1 \cdot 0$	0
0	1	1	$0 \cdot 1 + 0 \cdot 1 + 1 \cdot 1 \cdot 1$	1
1	0	0	$1 \cdot 0 + 1 \cdot 0 + 0 \cdot 0 \cdot 0$	0
1	0	1	$1 \cdot 0 + 1 \cdot 1 + 0 \cdot 0 \cdot 1$	1
1	1	0	$1 \cdot 1 + 1 \cdot 0 + 0 \cdot 1 \cdot 0$	1
1	1	1	$1 \cdot 1 + 1 \cdot 1 + 0 \cdot 1 \cdot 1$	1

3.9.4 Circuito para Tabela

A transposição de circuito lógico para tabela segue procedimento similar a conversão de função lógica para tabela. Primeiramente deve-se identificar quais as variáveis existentes no circuito e então montar uma tabela verdade do tamanho apropriado. A seguir atribui-se os valores das variáveis linha a linha e propaga-se os sinais ao longo do circuito. O resultado obtido é atribuído a saída da linha correspondente.

Para a transposição de tabela verdade para função Booleana ou circuito lógico faz-se necessário entender o que são produtos canônicos, assunto abordado a seguir.

3.10 Formas Canônicas

Qualquer função Booleana pode ser reescrita sob a norma canônica. Diz-se que uma função Booleana encontra-se em forma canônica se ela for composta por termos que sejam formados por todas as variáveis previstas pela função. Um termo é definido como $n - 1$ operações Booleanas E ou OU entre todas as variáveis previstas na função. A variável n refere-se ao número de variáveis Booleanas previstas na função.

Caso o termo for composto por operações E entre as variáveis dizemos que este é um mintermo. Similarmente, se o termo for composto por operações OU entre as variáveis, dizemos que este é um maxtermo.

Há duas formas canônicas:

1. Soma de Produtos - SdP (\sum) no qual todos os termos da função são mintermos;
2. Produto de Somas - PdS (\prod) no qual todos os termos da função são maxtermos.

Para que uma função Booleana esteja em na forma canônica ela deve ser composta apenas por mintermos ou maxtermos, mas não ambos. Também não encontra-se na forma canônica funções Booleanas que possuam termos incompletos.

Exemplo 3.11. Funções que estão ou não na forma canônica:

$$\begin{aligned} F(A, B, C) &= A\bar{B}C + ABC && \checkmark \text{ está na forma canônica} \\ F(A, B, C) &= ABC + AB + AC && X \text{ não está na forma canônica} \\ F(A, C, B) &= ABC + (A + B + C) && X \text{ não está na forma canônica} \\ f(A, B, C) &= (A + B + C) \cdot (A + B + \bar{C}) && \checkmark \text{ está na forma canônica} \end{aligned}$$

3.10.1 Soma de Produtos

Na forma da soma de produtos objetiva-se reescrever a função Booleana utilizando-se apenas mintermos. Há dois possíveis procedimentos: a) via manipulação algébrica; e b) via tabela verdade.

Via manipulação algébrica - O objetivo é aplicar sucessivamente as propriedades e teoremas Booleanos para transformar a expressão Booleana em mintermos.

Exemplo 3.12. Considere a função $F(A, B, C) = A \oplus B \oplus C$. Claramente ela não se encontra na forma de soma de produtos. O primeiro passo refere-se a substituir as operações \oplus em termos de Es e OUs. Como visto anteriormente, $X \oplus Y \equiv \bar{X}Y + X\bar{Y}$.

Passo	Expressão	Propriedade
(0)	$A \oplus B \oplus C$	$X \oplus Y = \bar{X}Y + X\bar{Y}$
(1)	$(\bar{A}B + A\bar{B}) \oplus C$	$X \oplus Y = \bar{X}Y + X\bar{Y}$
(2)	$\overline{(\bar{A}B + A\bar{B})}C + (\bar{A}B + A\bar{B})\bar{C}$	De Morgan
(3)	$(\bar{A}\bar{B} \cdot \overline{A\bar{B}})C + (\bar{A}B + A\bar{B})\bar{C}$	De Morgan
(4)	$((\bar{\bar{A}} + \bar{B}) \cdot (\bar{A} + \bar{\bar{B}})) + (\bar{A}B + A\bar{B})\bar{C}$	$\bar{\bar{X}} = X$
(5)	$((A + \bar{B}) \cdot (\bar{A} + B)) + (\bar{A}B + A\bar{B})\bar{C}$	distributiva
(6)	$(\bar{A}(A + \bar{B}) + B(A + \bar{A}))C + (\bar{A}B + A\bar{B})\bar{C}$	distributiva
(7)	$(\bar{A}A + \bar{A} \cdot \bar{B} + AB + B\bar{B})C + (\bar{A}B + A\bar{B})\bar{C}$	$X\bar{X} = 0$
(8)	$(0 + \bar{A} \cdot \bar{B} + AB + 0)C + (\bar{A}B + A\bar{B})\bar{C}$	$0 + X = X$
(9)	$(\bar{A} \cdot \bar{B} + AB)C + (\bar{A}B + A\bar{B})\bar{C}$	distributiva
(10)	$\bar{A} \cdot \bar{B}C + ABC + \bar{A}B\bar{C} + A\bar{B} \cdot \bar{C}$	\sum_s

Como podemos ver a partir do exemplo acima, a manipulação algébrica é suficiente para transformar qualquer expressão Booleana em sua forma de

soma de produtos. No entanto ela requer considerável habilidade. Considere o exemplo a seguir em que a expressão dada encontra-se em uma forma "quase" canônica:

Exemplo 3.13. Considere a função $F(A, B) = A + (\overline{AB})$. O segundo termo já é um mintermo. No entanto o primeiro termo contém apenas a variável A . Faz-se necessário que se "fabrique" a variável faltante, ou seja B . Isso pode ser alcançado pela aplicação sucessiva das propriedades $X = X \cdot 1$ e $1 = X + \overline{X}$ tal como apresentado no exemplo a seguir.

Passo	Expressão	Propriedade
(0)	$A + (\overline{AB})$	$X = X \cdot 1$
(1)	$A \cdot 1 + (\overline{AB})$	$1 = X + \overline{X}$
(2)	$A \cdot (B + \overline{B}) + (\overline{AB})$	distributiva
(3)	$AB + A\overline{B} + \overline{AB}$	\sum_S

Considere a função $F(A, B, C) = AB + C$. No primeiro termo a variável C não está presente e no segundo faltam as variáveis A e B para que a função esteja na forma de soma de produtos. A ausência de uma variável em um termo significa que ela não depende da variável faltante, ou seja, é equivalente a ter ou não ter a variável no termo. Sendo assim, o termo AB é equivalente aos termos $ABC + AB\overline{C}$. O mesmo se aplica ao termo formado apenas pela variável C . Ela é equivalente a ter ou não ter todas as possíveis combinações das variáveis faltantes, ou seja, A e B . Consequentemente, o termo C é equivalente aos mintermos $\overline{A} \cdot \overline{B}C + \overline{A}BC + A\overline{B}C + ABC$. A expressão combinada se configura $\overline{A} \cdot \overline{B}C + \overline{A}BC + A\overline{B}C + ABC + ABC + ABC + ABC$. Note que o mintermo ABC aparece duas vezes, o que pela propriedade $X + X = X$ nos permite seguramente remover um deles.

Via tabela verdade - Esta é uma forma mecânica para a obtenção da forma canônica. O processo requer que a tabela verdade seja conhecida e segue os seguintes passos:

1. Dado um circuito lógico ou uma função Booleana levante sua tabela verdade;
2. Identifique as linhas da tabela nas quais a saída seja igual a "1";
3. Escreva um mintermo para cada linha que a saída seja igual a "1" e una os mintermos por operações OU;
4. Consulte a linha correspondente a cada mintermo. Caso a variável na linha seja igual a "0" coloque uma barra sobre esta variável no mintermo correspondente.

Exemplo 3.14. Coincide novamente a função $F(A, B, C) = AB + C$. Construindo sua tabela verdade obtêm-se:

Linha	A	B	C	$A \cdot B$	$A \cdot B + C$
0	0	0	0	0	0
1	0	0	1	0	1
2	0	1	0	0	0
3	0	1	1	0	1
4	1	0	0	0	0
5	1	0	1	0	1
6	1	1	0	1	1
7	1	1	1	1	1

Aplicando o passo 2, identifica-se que as linhas 1, 3, 5, 6 e 7 possuem saída igual a "1", ou seja, cinco linhas da tabela.

Aplicando-se o passo 3, obtêm-se a expressão $ABC + ABC + ABC + ABC + ABC$

Finalmente aplicando o passo 4, nega-se todas as variáveis dos respectivos mintermos cujo valor seja igual a "0", obtendo-se $\sum_S = \bar{A} \cdot \bar{B}C + \bar{A}BC + A\bar{B}C + ABC + ABC$.

3.10.2 Produto de Somas

Na forma do Produto de Somas objetiva-se reescrever a função Booleana utilizando-se apenas maxtermos. Como visto anteriormente, um maxtermo é um termo composto por todas as variáveis previstas na função unidas por operações OU. Também é possível obter a forma canônica do produto de somas via manipulação algébrica ou por meio da tabela verdade. Apenas algumas alterações se fazem necessárias aos passos para obtenção mecânica da forma canônica. Os passos revisados são os seguintes:

1. Dado um circuito lógico ou uma função Booleana levante sua tabela verdade;
2. Identifique as linhas da tabela nas quais a saída seja igual a "0";
3. Escreva um maxtermo para cada linha que a saída seja igual a "0" e una os maxtermos por operações E;
4. Consulte a linha correspondente a cada maxtermo. Caso a variável na linha seja igual a "1" coloque uma barra sobre esta variável no maxtermo correspondente.

Exemplo 3.15. Considere a seguinte tabela verdade:

Linha	A	B	C	S
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	1
5	1	0	1	0
6	1	1	0	0
7	1	1	1	1

Aplicando o passo 2, identifica-se que as linhas 0,3,5 e 6 possuem saída igual a "0", ou seja, quatro linhas da tabela.

Aplicando-se o passo 3, obtêm-se a expressão $(A + B + C) \cdot (A + B + C) \cdot (A + B + C) \cdot (A + B + C)$

Finalmente aplicando o passo 4, nega-se todas as variáveis dos respectivos maxtermos cujo valor seja igual a "1", obtendo-se $\prod_S = (A + B + C) \cdot (A + \bar{B} + \bar{C}) \cdot (\bar{A} + B + \bar{C}) \cdot (\bar{A} + \bar{B} + C)$.

3.11 Simplificação via Mapas de Karnaugh

O método de simplificação por meio dos mapas de Veich-Karnaugh visa sistematizar o processo de simplificação de funções algébricas. Ele funciona muito bem para funções que possuem de duas a cinco variáveis. A partir de 6 variáveis, embora aplicável, torna-se muito complexo identificar os conjuntos de casos a serem simplificados. O método requer que a função algébrica seja fornecida sob a forma de soma de produtos ou que a tabela verdade seja conhecida.

O processo de simplificação segue os passos a seguir:

1. Colocar a função na forma de Soma de Produtos;
2. Desenhar o mapa de Karnaugh apropriado para o número de variáveis;
3. Mapear os termos da \sum_S que possuem saída "1" para o mapa;
4. Agrupar os "1"s do mapa de modo a utilizar todos eles;
 - (a) agrupe os 1's isolados ou seja, não adjacente a nenhum outro 1;
 - (b) agrupe os 1's adjacente a apenas um outro 1 formando assim pares;

- (c) agrupe os 1's em grupos de 4, 8, 16, ... a medida do possível, até que todos os 1's do mapa estejam em pelo menos um grupo.
5. Para cada grupo, manter apenas as variáveis que não variam para nenhum dos "1"s.

Os mapas de Karnaugh automatizam a aplicação das propriedades: a) distributiva (P14) mais especificamente sua aplicação inversa $((XY + XZ) = X(Y + Z))$; b) (P05) ou seja $X + \bar{X} = 1$; e c) (P07), ou seja $X \cdot 1 = X$.

A geometria dos mapas de Karnaugh dependem do número de variáveis distintas na função a ser simplificada. O mapa é construído de modo a que todas as possíveis combinações de variáveis seja contemplada. Na verdade, o mapa de Karnaugh nada mais é que uma forma conveniente de reescrever a tabela verdade.

3.11.1 Mapas de Duas Variáveis

No caso mais simples, a função a ser simplificada é composta por apenas duas variáveis. Em tal caso, o mapa de Karnaugh será composto por uma matriz de quatro células (2×2).

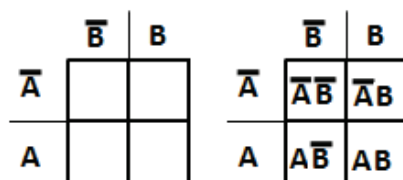
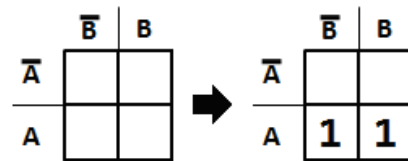


Figura 3.8: Representação do mapa de Veich-Karnaugh para duas variáveis. A direita os produtos canônicos correspondentes a cada uma das células.

A figura 3.8 apresenta o mapa e o mapeamento de todos os produtos canônicos para as suas respectivas células. Todos os possíveis produtos canônicos para funções de duas variáveis estão listados a seguir. Note que cada um deles corresponde a uma das células do mapa.

A	B	mintermo
0	0	$\bar{A} \cdot \bar{B}$
0	1	$\bar{A} \cdot B$
1	0	$A \cdot \bar{B}$
1	1	$A \cdot B$

Exemplo 3.16. Considere a função $AB + A\bar{B}$. Ela já se encontra na forma de soma de produtos, então pode ser mapeada diretamente para o mapa. Os produtos correspondem à célula inferior direita e a inferior esquerda, respectivamente. Aplicando o passo 2, desenha-se o mapa-K. Subsequentemente aplica-se o passo 3 mapeando os produtos canônicos para as células respectivas por meio da marcação de um "1".



O próximo passo (4) refere-se a agrupar os "1" no mapa-K em grupos retangulares em potências de 2, ou seja, para o caso de duas variáveis $2^0 = 1$ célula, $2^1 = 2$ células e $2^2 = 4$ células. As formas válidas de agrupamento dos "1" para o mapa-K de duas variáveis é apresentada na figura 3.9.

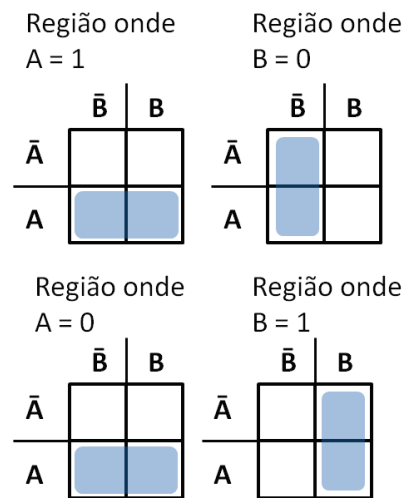


Figura 3.9:

Note que as regiões de agrupamento visam identificar regiões do mapa em que uma ou mais variáveis não variam, ou seja, que a variável ou ela negada se mantém inalteradas. Aplicando o passo 5 observamos que ambos os "1" encontram-se na região que a variável A se mantém inalterada. Quanto a variável B um dos "1"s encontra-se na região \bar{B} e o outro na região B . Consequentemente a variável B pode ser simplificada (suprimida) obtendo-se a expressão simplificada $S = A$.

3.11.2 Mapas de Três Variáveis

No caso em que a função seja composta por três variáveis distintas o mapa de Karnaugh será realizado por uma matriz de oito células (2×4).

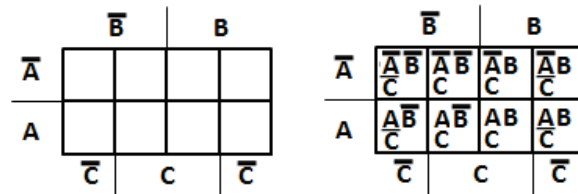


Figura 3.10: Representação do mapa de Veich-Karnaugh para três variáveis

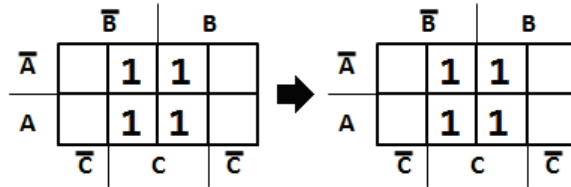
A figura 3.10 apresenta o mapa e o mapeamento de todos os produtos canônicos para as suas respectivas células. Todos os possíveis produtos canônicos para funções de duas variáveis estão listados a seguir. Note que cada um deles corresponde a uma das células do mapa.

A	B	C	mintermo
0	0	0	$\bar{A} \cdot \bar{B} \cdot \bar{C}$
0	0	1	$\bar{A} \cdot \bar{B} \cdot C$
0	1	0	$\bar{A} \cdot B \cdot \bar{C}$
0	1	1	$\bar{A} \cdot B \cdot C$
1	0	0	$A \cdot \bar{B} \cdot \bar{C}$
1	0	1	$A \cdot \bar{B} \cdot C$
1	1	0	$A \cdot B \cdot \bar{C}$
1	1	1	$A \cdot B \cdot C$

Exemplo 3.17. Considere a função $\bar{A}C + A\bar{B}C + ABC$. Ela não se encontra na forma de soma de produtos. Consequentemente devemos aplicar o passo 1 para transformá-la na forma apropriada.

A	B	C	$\bar{A}C$	$A\bar{B}C$	ABC	
0	0	0	0	0	0	
0	0	1	1	0	1	$\rightarrow \bar{A} \cdot \bar{B}C$
0	1	0	0	0	0	
0	1	1	1	0	1	$\rightarrow \bar{A}BC$
1	0	0	0	0	0	
1	0	1	0	1	1	$\rightarrow A\bar{B}C$
1	1	0	0	0	0	
1	1	1	0	0	1	$\rightarrow ABC$

Aplicando o passo 2, desenha-se o mapa-K. Subsequentemente aplica-se o passo 3 mapeando os produtos canônicos para as células respectivas por meio da marcação de um "1".



O próximo passo (4) refere-se a agrupar os "1" no mapa-K em grupos retangulares em potências de 2, ou seja, para o caso de duas variáveis $s2^0 = 1$ célula, $2^1 = 2$, $2^2 = 4$, e $2^3 = 8$ células. As formas válidas de agrupamento dos "1" para o mapa-K de duas variáveis é apresentada na figura 3.11.

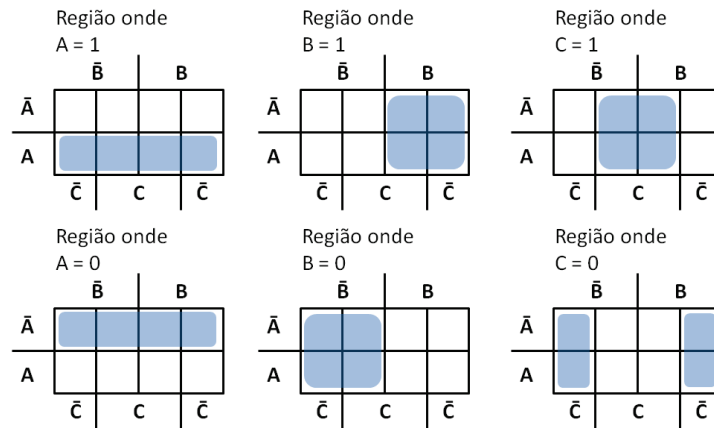


Figura 3.11:

è possível agrupar todos os "1"s do mapa em apenas um grupo. Observa-se que a variável C é a única comum a todos os mintermos, consequentemente as variáveis A e B são simplificadas resultando em $S = C$.

3.11.3 Mapas de Quatro Variáveis

No caso em que a função seja composta por quatro variáveis distintas o mapa de Karnaugh será realizado por uma matriz de dezesseis células (4×4).

A figura 3.12 apresenta o mapa e o mapeamento de todos os produtos canônicos para as suas respectivas células. Todos os possíveis produtos canônicos para funções de quatro variáveis estão listados a seguir. Note que cada um deles corresponde a uma das células do mapa.

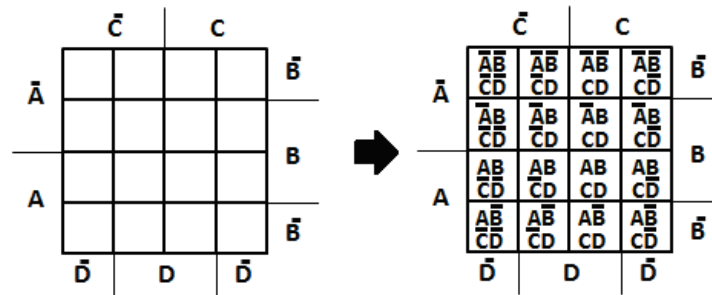


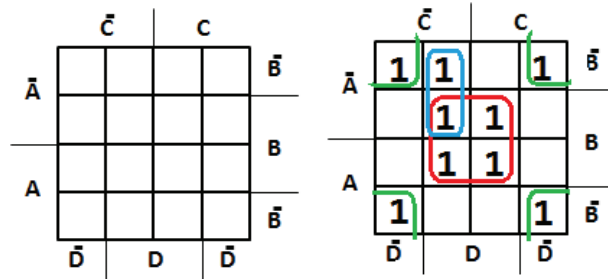
Figura 3.12: Representação do mapa de Veich-Karnaugh para quatro variáveis. A direita os produtos canônicos correspondentes a cada uma das células.

A	B	C	D	mintermo	A	B	C	D	mintermo
0	0	0	0	$\overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D}$	0	0	0	1	$\overline{A} \cdot \overline{B} \cdot \overline{C} \cdot D$
0	0	1	0	$\overline{A} \cdot \overline{B} \cdot C \cdot \overline{D}$	0	0	1	1	$\overline{A} \cdot \overline{B} \cdot C \cdot D$
0	1	0	0	$\overline{A} \cdot B \cdot \overline{C} \cdot \overline{D}$	0	1	0	1	$\overline{A} \cdot B \cdot \overline{C} \cdot D$
0	1	1	0	$\overline{A} \cdot B \cdot C \cdot \overline{D}$	0	1	1	1	$\overline{A} \cdot B \cdot C \cdot D$
1	0	0	0	$A \cdot \overline{B} \cdot \overline{C} \cdot \overline{D}$	1	0	0	1	$A \cdot \overline{B} \cdot \overline{C} \cdot D$
1	0	1	0	$A \cdot \overline{B} \cdot C \cdot \overline{D}$	1	0	1	1	$A \cdot \overline{B} \cdot C \cdot D$
1	1	0	0	$A \cdot B \cdot \overline{C} \cdot \overline{D}$	1	1	0	1	$A \cdot B \cdot \overline{C} \cdot D$
1	1	1	0	$A \cdot B \cdot C \cdot \overline{D}$	1	1	1	1	$A \cdot B \cdot C \cdot D$

Exemplo 3.18. Considere a tabela verdade dada a seguir:

A	B	C	D	S		A	B	C	D	S	
0	0	0	0	1	$\rightarrow \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D}$	0	0	0	1	1	$\rightarrow \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot D$
0	0	1	0	1	$\rightarrow \overline{A} \cdot \overline{B} \cdot C \cdot \overline{D}$	0	0	1	1	0	
0	1	0	0	0		0	1	0	1	1	$\rightarrow \overline{A} \cdot B \cdot \overline{C} \cdot D$
0	1	1	0	0		0	1	1	1	1	$\rightarrow \overline{A} \cdot B \cdot C \cdot \overline{D}$
1	0	0	0	1	$\rightarrow \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D}$	1	0	0	1	0	
1	0	1	0	1	$\rightarrow \overline{A} \cdot \overline{B} \cdot C \cdot \overline{D}$	1	0	1	1	0	
1	1	0	0	0		1	1	0	1	1	$\rightarrow \overline{A} \cdot B \cdot \overline{C} \cdot \overline{D}$
1	1	1	0	0		1	1	1	1	1	$\rightarrow \overline{A} \cdot B \cdot C \cdot \overline{D}$

Aplicando o passo 2, desenha-se o mapa-K. Subsequentemente aplica-se o passo 3 mapeando os produtos canônicos para as células respectivas por meio da marcação de um "1".



O próximo passo (4) refere-se a agrupar os "1" no mapa-K em grupos retangulares em potências de 2, ou seja, para o caso de duas variáveis $s2^0 = 1$ célula, $2^1 = 2$, $2^2 = 4$, $2^3 = 8$, e $2^4 = 16$ células. As principais formas válidas de agrupamento dos "1" para o mapa-K de quatro variáveis é apresentada na figura 3.13.

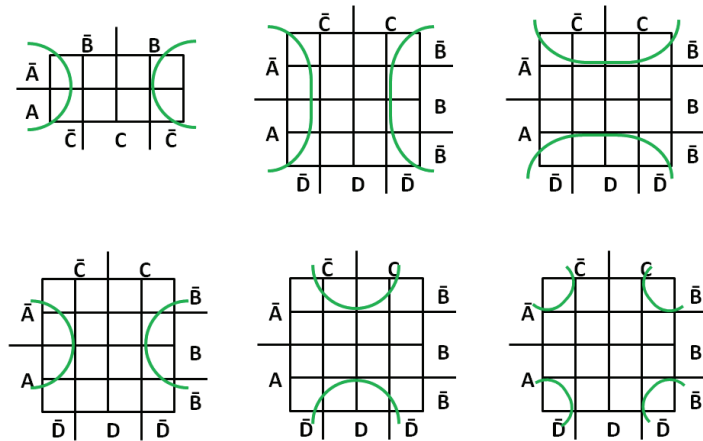


Figura 3.13:

Note que qualquer agrupamento válido em mapas de menor ordem (Mapa-K3 e Mapa-K2) também serão válidos em um mapa de mais alta ordem. Identifica-se assim três possíveis grupos, o grupo verde, vermelho e azul.

Na aplicação do passo 5 para obter-se a expressão minimizada deve-se avaliar cada um dos agrupamentos separadamente, e ao final uni-los todos por meio de operações OUs. No grupo azul as variáveis \bar{A} , \bar{C} e D se mantêm inalteradas. Consequentemente apenas a variável B/\bar{B} pode ser simplificada resultando em $\bar{A} \cdot \bar{C} D$. No segundo grupo (verde) as variáveis \bar{B} e \bar{D} se mantêm inalteradas resultando no termo simplificado $\bar{B} \cdot \bar{D}$. Por último o grupo vermelho apresenta as variáveis B e D como inalteradas resultando no termo minimizado BD . A expressão simplificada final é $\bar{A} \cdot \bar{C} D + \bar{B} \cdot \bar{D} + BD$

Casos Especiais

Há casos em que o mapa de Karnaugh não admite simplificação. Tome por exemplo os mapas apresentados na figura 3.14. Nota-se que em ambos os casos a) e b) não é possível proceder com o agrupamento de 1's adjacentes gerando assim oito grupos unitários. Note também que o padrão de localização dos 1's forma uma espécie de tabuleiro de xadrez.

a)		\bar{B}		B		
		1		1		\bar{C}
\bar{A}	1			1		
		1		1		C
A	1			1		\bar{C}
		\bar{D}		D		\bar{D}

b)		\bar{B}		B		
		1		1		\bar{C}
\bar{A}				1		C
		1		1		\bar{C}
A				1		C
		\bar{D}		D		\bar{D}

Figura 3.14: Casos notáveis em que o mapa de Karnaugh não admite simplificação.

Nestes casos obtêm-se as seguintes expressões:

$$\sum_a = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot D + \bar{A} \cdot B \cdot \bar{C} \cdot \bar{D} + \bar{A} \cdot \bar{B} \cdot C \cdot \bar{D} + \bar{A} \cdot B \cdot C \cdot D + A \cdot \bar{B} \cdot C \cdot D + A \cdot B \cdot C \cdot \bar{D} + A \cdot \bar{B} \cdot \bar{C} \cdot \bar{D} + A \cdot B \cdot \bar{C} \cdot D \quad (3.7)$$

$$\sum_a = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot \bar{D} + \bar{A} \cdot B \cdot \bar{C} \cdot D + \bar{A} \cdot \bar{B} \cdot C \cdot D + \bar{A} \cdot B \cdot C \cdot \bar{D} + A \cdot \bar{B} \cdot C \cdot \bar{D} + A \cdot B \cdot C \cdot D + A \cdot \bar{B} \cdot \bar{C} \cdot D + A \cdot B \cdot \bar{C} \cdot \bar{D} \quad (3.8)$$

Que são equivalentes a:

$$S_a = A \oplus B \oplus C \oplus D$$

$$S_b = A \odot B \odot C \odot D$$

Outro caso interessante refere-se ao mapa-K apresentado na figura 3.15. Note que se procedermos com o agrupamento dos 1's adjacentes utilizando o maior grupo possível a seleção resultante será a apresentada na figura 3.16 (direita).

	\bar{B}	B	
\bar{A}		1	\bar{C}
	1	1	C
A		1	\bar{C}
	1	1	C
	\bar{D}	D	\bar{D}

Figura 3.15: Mapa-K apresentando uma situação interessante.

No entanto, se aplicarmos a lista de passos apresentada anteriormente obter-se-á a seleção de grupos apresentada na figura 3.16(esquerda). Ao simplificarmos o mapa da direita obtemos a expressão $S1_{min} = CD + \bar{A}BD + \bar{A} \cdot \bar{B}C + \bar{A}\bar{B}D + ABC$. A simplificação do mapa da esquerda produz $S2_{min} = \bar{A}BD + \bar{A} \cdot \bar{B}C + \bar{A}\bar{B}D + ABC$ que obviamente é mínima em relação a $S1_{min}$ pois esta possui o termo CD em adição aos termos presentes em $S2_{min}$.

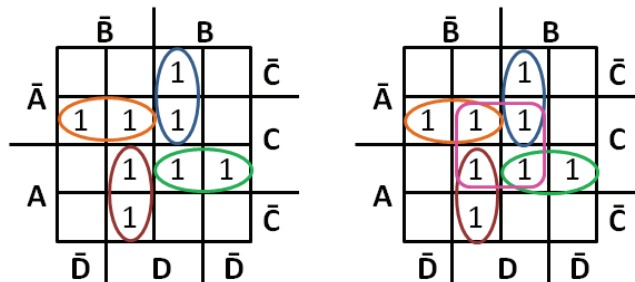


Figura 3.16: Possíveis agrupamentos dos conjuntos adjacentes de 1's.

O que se observa do exemplo acima é que em geral queremos selecionar grupos contendo o maior número de 1's possível. No entanto 1's vizinhos a apenas um outro 1 podem impor uma situação na qual todos os 1's acabam sendo selecionados duas vezes. Nestes casos chegaremos a uma função simplificada que não é mínima advogando assim para a aplicação sistemática dos passos para simplificação via mapas de Karnaugh apresentados anteriormente.

Note que a expressão $S1_{min}$ pode ser subsequentemente simplificada utilizando para tal manipulação algébrica. No entanto esta situação contradiz a premissa dos mapas-K na qual o processo de simplificação deve ser sistemático e mecânico.

3.11.4 Mapas de Cinco ou Mais Variáveis

Acima de quatro variáveis a aplicação do mapa de Karnaugh se torna complicada. A mecânica do método dita que para cinco variáveis um mapa de 32 células se faz necessário. Tal mapa é alcançado pela duplicação do mapa-K de quatro variáveis como visto na figura 3.17. O processo de duplicação pode ser continuado indefinidamente. No entanto a aplicação do passo 4, ou seja o agrupamento dos "1"s mapeados no mapa torna-se incrementalmente mais complexo. A chance de não encontrar grupos que contenham o maior número de "1"s passa a ser grande.

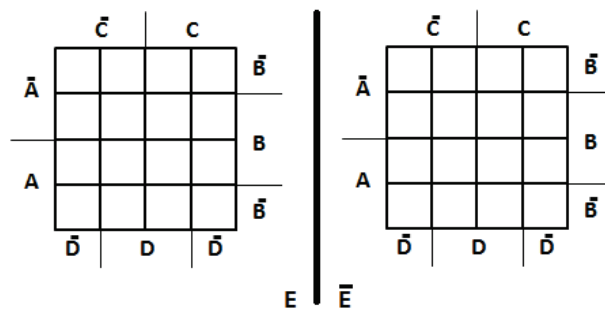
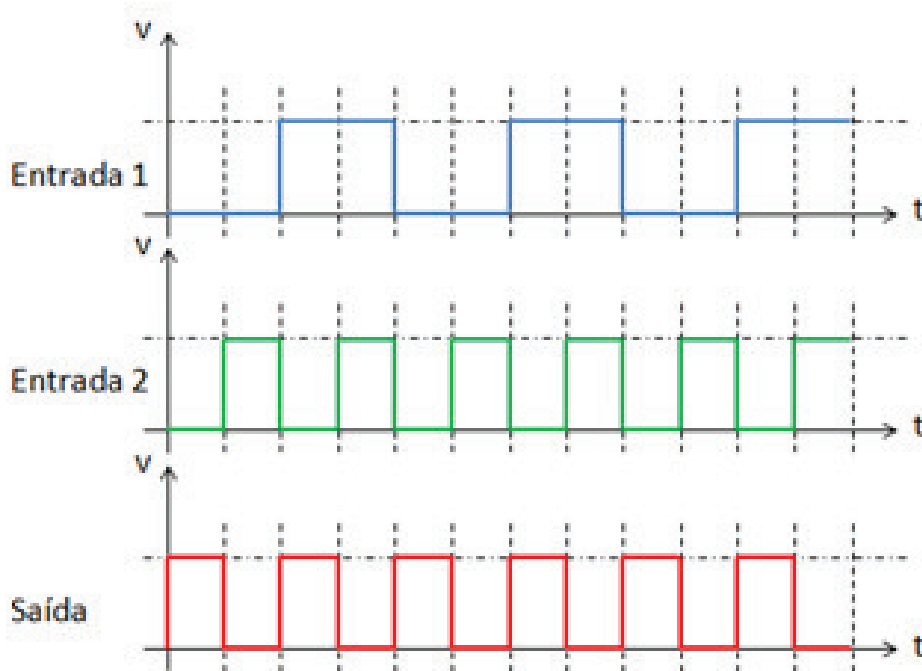


Figura 3.17: Representação do mapa de Veich-Karnaugh para cinco variáveis

Exercícios

1. As variáveis Booleanas são normalmente representadas por letras maiúsculas do alfabeto. Que valores tais variáveis podem assumir?
2. Existem três operações básicas a partir das quais todas as outras funções lógicas podem ser sintetizadas. Quais são elas? Forneça os diversos símbolos utilizados para cada uma delas, suas tabelas verdade para duas variáveis e o desenho lógico de cada uma delas.
3. Forneça a tabela verdade dado o diagrama de tempo abaixo:



4. Com relação a tabela verdade do exercício anterior, forneça um circuito de portas lógicas que a implementa;
5. Uma expressão lógica é uma função que aceita apenas variáveis Booleanas e produz como saída um valor verdadeiro ou falso dependendo dos valores de suas variáveis de entrada. Forneça a tabela verdade para as seguintes funções Booleanas:

(a) $F(A, B, C) = \overline{(A \cdot B)} + C$

(b) $F(A, B, C) = (A + B) \cdot \overline{C}$

(c) $F(A, B, C) = (A + B).(A + C)$

(d) $F(A, B, C, D) = \overline{(A + B).(\overline{C + D})}$

(e) $F(A, B, C) = AB + AC + BC$

6. Existe, tal como na álgebra Euclideana, muitas propriedades da álgebra Booleana. Complete as seguintes igualdades das propriedades listadas abaixo:

(a) $X + 0 =$

(b) $X.1 =$

(c) $X.X =$

(d) $X + \overline{X} =$

(e) $X.\overline{X} =$

(f) $X + 1 =$

(g) $X.0 =$

(h) $X + \overline{X} =$

(i) $X.\overline{X} =$

(j) $\overline{\overline{X}} =$

(k) $X.(1 + Y) =$

(l) $X + \overline{X}.Y =$

7. Explique e dê exemplos de como a propriedade comutativa se processa.
 8. Explique e dê exemplos de como a propriedade associativa se processa.
 9. Explique e dê exemplos de como a propriedade distributiva se processa.
 10. Dadas as expressões Booleanas abaixo, projete os circuitos digitais:

(a) $\overline{A.B} = \overline{A} + \overline{B}$

(b) $\overline{A + B} = \overline{A}.\overline{B}$

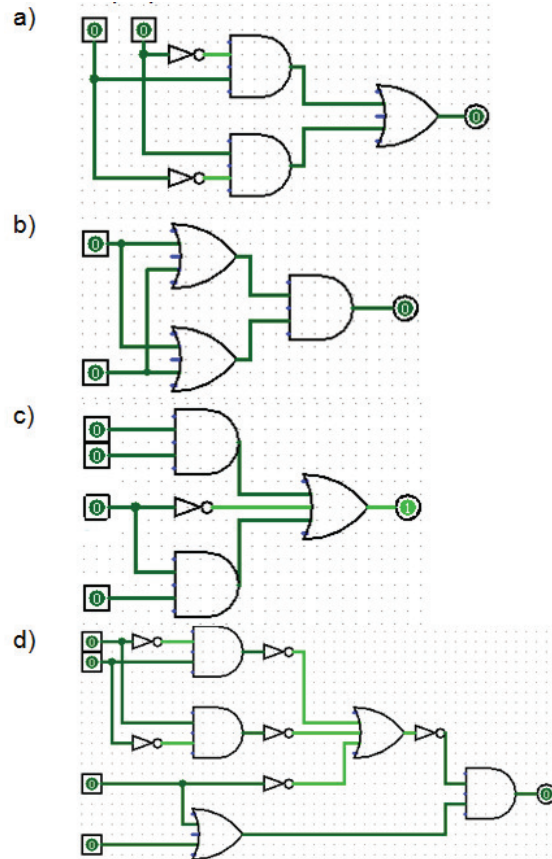
(c) $\overline{A}.B.\overline{C}D + AC\overline{D} + \overline{B}.\overline{C}.\overline{D} + \overline{A}.\overline{D}$

(d) $A.B + A.C + A.D + B.C + B.D + C.D$

(e) $\overline{A}.B + \overline{C}.D$

11. Prove via tabela verdade que $A + (BC) \equiv (A + B) \cdot (A + C)$

12. Prove via tabela verdade e manipulação algébrica que $AB + CD \equiv (A + C) \cdot (B + C) \cdot (A + D) \cdot (B + D)$
13. Levante a expressão booleana a partir dos circuitos propostos.



14. Construa a tabela verdade para todos os circuitos do exercício anterior.
15. O Teorema de DeMorgan é dado como segue: “O complemento do produto é igual à soma dos complementos.” Prove via tabela verdade que:

$$(a) (A.B) + \overline{C} + \overline{C.D}$$

$$(b) (A + B + C).\overline{C} + B.C + \overline{A.C}$$

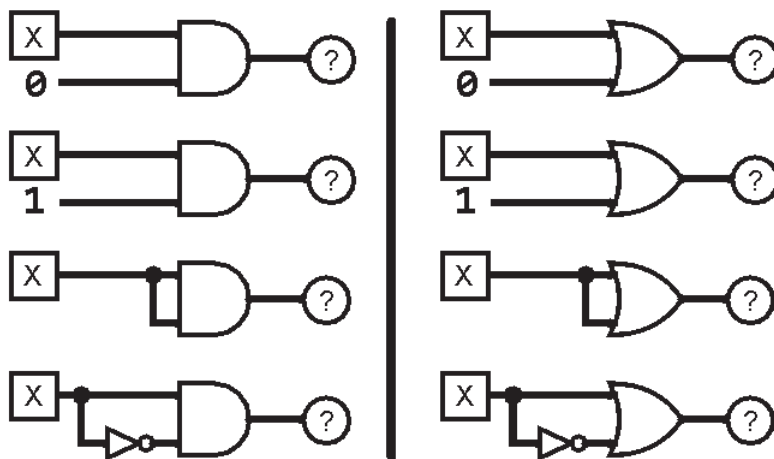
16. Simplifique as seguintes expressões algébricas via manipulação algébrica. Liste na coluna da esquerda qual propriedade está sendo usada para cada passo da evolução e construa o circuito correspondente antes e depois da simplificação:

$$(a) S = (\overline{A} + \overline{B} + \overline{C}).(A + B + \overline{C})$$

- (b) $S = (\bar{A} \cdot \bar{B}C) + (\bar{A}BC) + (\bar{A}\bar{B}\bar{C}) + (ABC) + (A\bar{B}\bar{C})$
 (c) $S = \overline{ABCD} + (\bar{A} \cdot \bar{B} \cdot \bar{C}D) + (\bar{A}\bar{B}\bar{C}D) + (\bar{A}\bar{B}\bar{C}\bar{D})$
 (d) $S = (A\bar{C}) + (A\bar{B}\bar{C})$
 (e) $S = (\bar{A} \cdot \bar{B} \cdot \bar{C} \cdot \bar{D}) + (\bar{A}\bar{B}\bar{C} \cdot \bar{D})$
 (f) $S = \overline{A + \bar{B}C}$
 (g) $S = \overline{(A + BC) \cdot (D + \bar{A}\bar{B})}$
 (h) $S = \overline{AB\bar{C}D} + \overline{CD} + \overline{AB}$
 (i) $S = \overline{ABCDE} + \overline{ABC} + \bar{D}E + \bar{C}D + \bar{A}D + \bar{A}\bar{B}$
 (j) $S = (\bar{A} + \bar{B} + \bar{C}) \cdot (A + B + \bar{C})$
 (k) $S = (\bar{A} \cdot \bar{B}C) + (\bar{A}BC) + (\bar{A}\bar{B}\bar{C}) + (ABC) + (A\bar{B}\bar{C})$
 (l) $S = \overline{A + (\bar{B} + \bar{C})D} \cdot (A + \bar{B})$
 (m) $S = A(\bar{A} \odot \bar{B}) \oplus (ABC + (\bar{A} \odot \bar{C}))$
 (n) $S = (\bar{A} \cdot \bar{B} \cdot \bar{C}) + (\bar{A} \cdot \bar{B}C) + (\bar{A}\bar{B}\bar{C}) + (\bar{A}\bar{B}C) + (\bar{A}\bar{B} \cdot \bar{C}) + (\bar{A}\bar{B}C) + (A\bar{B}\bar{C})$

17. Construa o circuito de portas lógicas para cada uma das funções Booleanas apresentadas no exercício anterior. Forneça também seu circuito simplificado, e a tabela verdade.

18. Indique a saída de cada um dos circuitos abaixo:

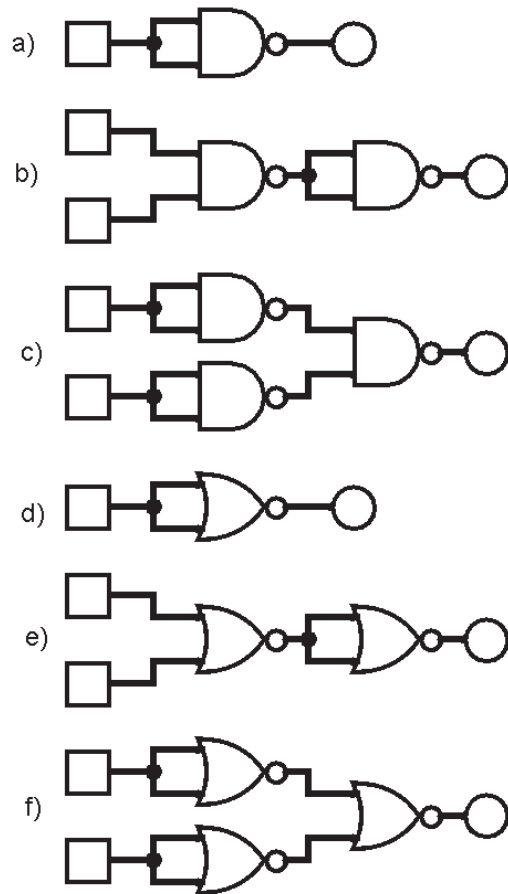


19. Dada as tabelas verdade abaixo:

- (a) Construa o circuito que a implementa;
 (b) Levante a expressão booleana correspondente;
 (c) Simplifique a expressão via manipulação algébrica.

A	B	C	S	A	B	C	S	A	B	C	S
0	0	0	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	1	1	0	0	1	1
0	1	0	0	0	1	0	1	0	1	0	1
0	1	1	1	0	1	1	0	0	1	1	0
1	0	0	0	1	0	0	1	1	0	0	1
1	0	1	1	1	0	1	0	1	0	1	0
1	1	0	1	1	1	0	0	1	1	0	0
1	1	1	0	1	1	1	0	1	1	1	0

20. Levante a tabela verdade dos seguintes circuitos lógicos e então identifique para cada um deles a operação lógica que eles simulam:



21. coloque as seguintes funções Booleanas em sua forma canônica via manipulação algébrica:

(a) $S = A + (\overline{AB})$

(b) $S = A \oplus B \oplus C$

(c) $S = \overline{A} + BC$

(d) $S = \overline{A} \cdot \overline{B} \cdot \overline{C} + \overline{A} \cdot \overline{B}C + \overline{A}BC + AC + AB + BC$

(e) $S = \overline{\overline{A \oplus B \oplus C} \cdot D + E}$

22. Construa o circuito das seguintes funções Booleanas utilizando apenas portas NÃO-E. A seguir construa os mesmos circuitos utilizando apenas portas NÃO-OU:

(a) $F(A, B) = A \oplus B$

(b) $F(A, B) = A \odot B$

(c) $F(A, B, C, D) = \overline{A} + \overline{B} + \overline{C} + \overline{D}$

(d) $F(A, B, C) = \overline{AB} \oplus \overline{AC} \oplus \overline{BC}$

23. Prove via manipulação algébrica que as expressões são equivalentes:

(a) $A + AB \equiv A$

(b) $(A + B) \cdot (A + C) \equiv A + BC$

(c) $ABC + A\overline{C} + A\overline{B} \equiv A$

(d) $(\overline{A} \cdot \overline{B}) + (A\overline{B}) \equiv \overline{A}$

(e) $A + (\overline{AB}) \equiv A + B$

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

24. Dadas as funções Booleanas abaixo na forma de soma de produtos, forneça sua equivalente na forma de produto de somas:

(a) $\sum_S = \overline{AB} + A\overline{B}$

(b) $\sum_S = \overline{ABC} + A\overline{BC} + \overline{ABC} + ABC$

(c) $\sum_S = \overline{ABCD} + \overline{AB}CD + A\overline{BC}D + ABC\overline{D} + ABCD$

25. Explique passo a passo como funciona o processo de simplificação pelos mapas de Veich-Karnaugh e correlacione quais são as propriedades Booleanas sua mecânica automatiza.

26. Dada as funções Booleanas abaixo, simplifique-as utilizando a técnica de Mapas de Veich-Karnaugh:

(a) $F(A, B) = A \oplus B \oplus C$

(b) $\overline{A + BC} \cdot \overline{D} + \overline{AB}$

(c) $\overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} + \overline{A} \cdot \overline{B} \cdot C \overline{D} + \overline{A} B \overline{C} D + A \overline{B} \cdot \overline{C} \cdot \overline{D} \cdot + A \overline{B} \cdot \overline{C} \cdot D$

(d) $\overline{A} \cdot \overline{BC} + \overline{A} B C + \overline{A} B \overline{C} + A B \overline{C} + A B C$

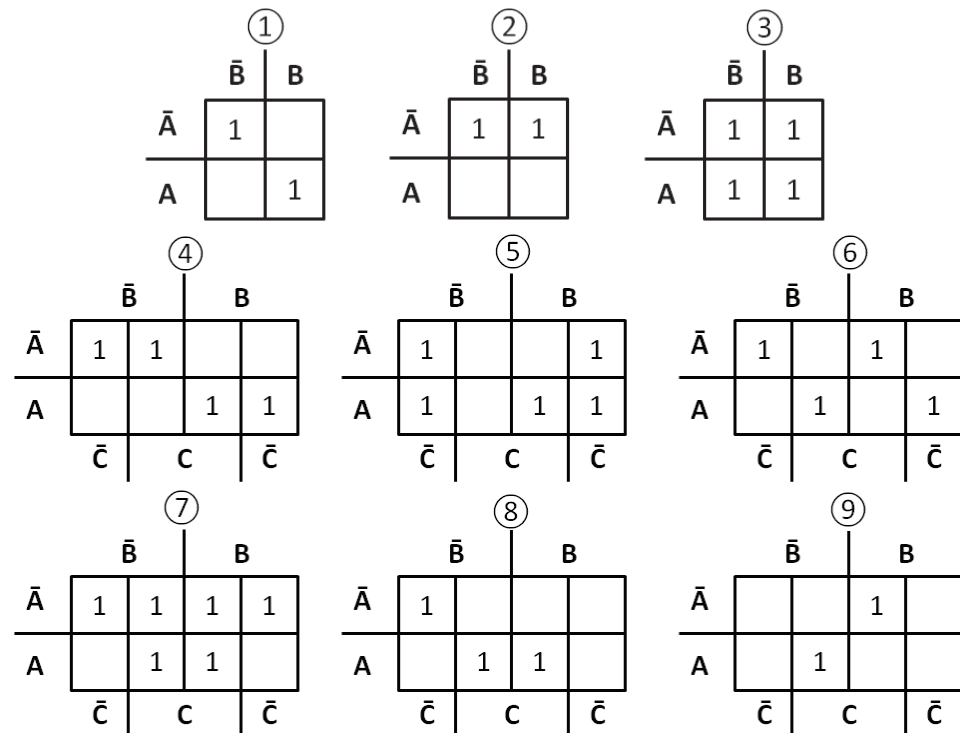
(e) $\overline{A} B C \overline{D} + \overline{A} \cdot \overline{B} \cdot \overline{C} D + \overline{A} B \overline{C} D + \overline{A} B \overline{C} D$

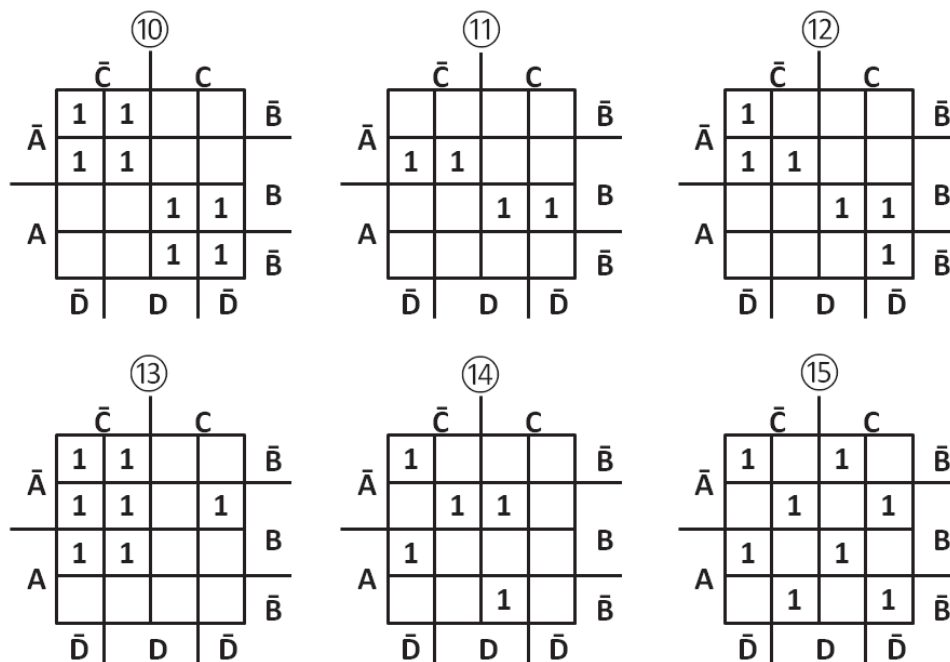
(f) $\overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D} + \overline{A} B \overline{C} \cdot \overline{D}$

(g) $\overline{A} B C D \overline{E} + \overline{A} B \overline{C} + \overline{D} E + \overline{C} D + \overline{A} D + \overline{A} B$

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

27. dado os mapas Karnaugh a seguir agrupe os conjuntos máximos de "1"s e forneça a expressão mínima.





28. O mecanismo de simplificação via mapas de Veich-Karnaugh requer que a função Booleana esteja na forma canônica de Soma de Produtos. Seria possível aplicar a mesma técnica utilizando a forma de Produto de Somas? Em caso afirmativo, quais seriam as alterações necessárias na mecânica do método?

EXTRA Construa um circuito capaz de somar dois números de 4 bits cada. Dica, comece levantando a tabela verdade da soma. Considere o “vai um” como um bit de entrada extra.

