

# Álgebra Booleana, Revisão Portas Lógicas e Mapas de Karnaugh

---

## 1) Álgebra Booleana:

Em 1854, George Boole introduziu o formalismo que até hoje se usa para o tratamento sistemático da lógica, que é a chamada **Álgebra Booleana**. Em 1938, C. E. Shannon aplicou esta álgebra para mostrar que as propriedades de circuitos elétricos de chaveamento podem ser representadas por uma álgebra Booleana com dois valores (estado “ligado” e “desligado” associado a 0 e 1)..

Em particular, na álgebra Booleana de dois valores, cada variável pode assumir um dentre dois valores possíveis, os quais podem ser denotados por [F,V] (falso ou verdadeiro), [H,L] (high and low) ou ainda [0,1]. Nesta disciplina, adotaremos a notação [0,1], a qual também é utilizada em eletrônica digital. Como o número de valores que cada variável pode assumir é finito (e pequeno), o número de estados que uma função Booleana pode assumir também será finito, o que significa que podemos descrever completamente as funções Booleanas utilizando tabelas. Devido a este fato, uma tabela que descreva uma função Booleana recebe o nome de **tabela verdade**, e nela são listadas todas as combinações de valores que as variáveis de entrada podem assumir e os correspondentes valores da função (saídas).

Na álgebra Booleana, existem três operações ou funções básicas. São elas, operação **OU**, operação **E** e **complementação**. Todas as funções Booleanas podem ser representadas em termos destas operações básicas.

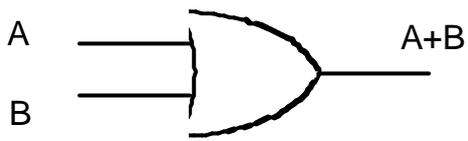
Axiomas e propriedades: comutatividade, associatividade, distributividade

### **FUNÇÃO OU / OR**

Tabela Verdade:

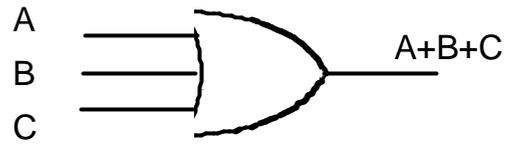
<b>A</b>	<b>B</b>	<b>S</b>
0	0	0
0	1	1
1	0	1
1	1	1

Representação da porta lógica



(a)

(a) Função OU de 2 entradas



(b)

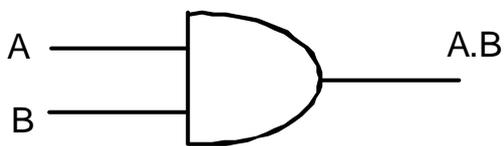
(b) Função OU de 3 entradas

### **FUNÇÃO E / AND**

Tabela Verdade:

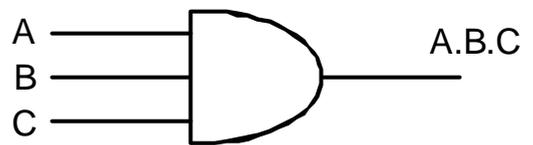
A	B	S
0	0	0
0	1	0
1	0	0
1	1	1

Representação da porta lógica



(a)

(a) Função E de 2 entradas



(b)

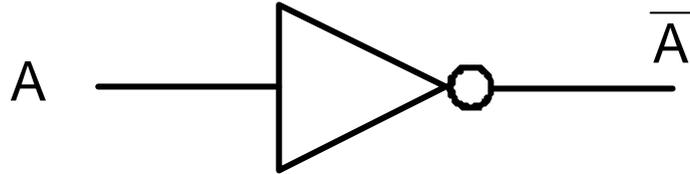
(b) Função E de 3 entradas

### **FUNÇÃO COMPLEMENTAR**

Tabela Verdade:

A	S
0	1
1	0

Representação da porta lógica



**Para desenhar um circuito a partir da equação Booleana**, seguir a ordem de avaliação de expressões:

1. Parêntesis (dos mais internos para os mais externos);
2. Operações E;
3. Operações OU

**Para criar Tabelas Verdades a partir de funções Booleanas:**

1. Criar colunas para as variáveis de entrada e listar todas as combinações possíveis, utilizando a fórmula:

$n^{\circ}$  de combinações =  $2^n$  (onde  $n$  é o número de variáveis de entrada);

2. Criar uma coluna para cada variável de entrada que apareça complementada na equação e anotar os valores resultantes;

3. Avaliar a equação seguindo a ordem de precedência, a partir do nível de parênteses mais interno:

1<sup>o</sup> negação de variáveis, i.e., operador complementação

2<sup>o</sup> multiplicação lógica, i.e., operador E

3<sup>o</sup> adição lógica, i.e., operador OU

Exercício:

Dada a função Booleana  $F = (A+B).C./A.D$ , determinar a tabela verdade e o circuito lógico associado a equação.

A	B	C	D	/A	C./A.D	A+B	F= (A+B).C./A.D
0	0	0	0	1	0	0	0
0	0	0	1	1	0	0	0
0	0	1	0	1	0	0	0
0	0	1	1	1	1	0	0
0	1	0	0	1	0	1	0
0	1	0	1	1	0	1	0
0	1	1	0	1	0	1	0
0	1	1	1	1	1	1	1
1	0	0	0	0	0	1	0
1	0	0	1	0	0	1	0
1	0	1	0	0	0	1	0

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

$$F = (A+B).C./A.D = A.C./A.D + /A.B.C.D = /A.B.C.D$$

$$F = /A.B.C.D$$

Representação gráfica: 1 porta E de 4 entradas onde a entrada A é invertida por um inversor.

## 2) Leis Fundamentais da Álgebra Booleana:

Seja A uma variável Booleana.

Então:

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

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

(definição do espaço de uma variável Booleana)

Propriedades da Operação OU

$$(1) \quad A + 0 = A \quad , \quad A \cdot 0 = 0$$

$$(2) \quad A + 1 = 1 \quad , \quad A \cdot 1 = A$$

$$(3) \quad A + A = A \quad , \quad A \cdot A = A$$

$$(4) \quad A + \bar{A} = 1 \quad , \quad A \cdot \bar{A} = 0$$

Propriedade da Complementação

$$(9) \quad \overline{\bar{A}} = A$$

Propriedade Comutativa

$$(10) \quad A + B = B + A$$

$$(11) \quad A \cdot B = B \cdot A$$

Propriedade Associativa

$$(12) \quad A + (B + C) = (A + B) + C = (A + C) + B$$

$$(13) \quad A \cdot (B \cdot C) = (A \cdot B) \cdot C = (A \cdot C) \cdot B$$

Propriedade Distributiva

$$(14) \quad A \cdot (B + C) = A \cdot B + A \cdot C$$

**Teoremas de Morgan**

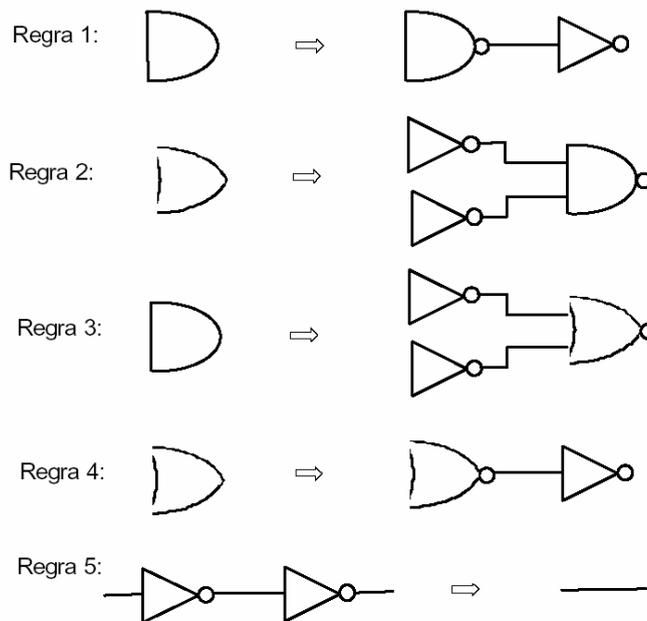
$$\overline{A \cdot B \cdot C \cdot \dots} = \bar{A} + \bar{B} + \bar{C} + \dots \quad (2.1)$$

Para duas variáveis

$$\overline{A \cdot B} = \bar{A} + \bar{B} \quad (2.3)$$

$$\overline{A + B + C + \dots} = \bar{A} \cdot \bar{B} \cdot \bar{C} \cdot \dots \quad (2.2)$$

$$\overline{A + B} = \bar{A} \cdot \bar{B} \quad (2.4)$$



Derivação das Expressões Booleanas: Tabela Verdade, Simplificação, Equação Lógica através da soma de produtos (Lista todas as combinações das variáveis de entrada para os quais a função vale 1) ou produto de somas (Lista todas as combinações das variáveis de entrada para os quais a função vale 0). Construção do circuito lógico.

### 3) Simplificação de Funções Booleanas usando Mapas de Karnaugh

O método de simplificação apresentado na seção 2.6 é de aplicabilidade limitada, uma vez que é bastante difícil certificar-se que todos os pares de mintermos que podiam ser simplificados foram determinados. Alternativamente àquele método, há outro

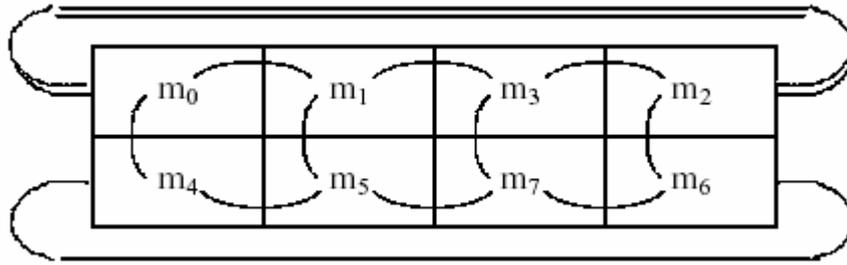
método de simplificação baseado na identificação visual de grupos de mintermos passíveis de serem simplificados. No entanto, para que se possa identificar tais grupos, é necessário que os mintermos sejam dispostos de maneira mais conveniente, o que será explicado a seguir. Na descrição do método de simplificação literal, foi mencionado que os pares de mintermos que se diferenciam de somente uma variável são passíveis de simplificação. Observando a ordem com que os mintermos de uma função Booleana qualquer (com, por exemplo, 3 variáveis) aparecem na tabela verdade, vemos que, se trocarmos o 3º mintermo com o 4º e trocarmos também o 7º mintermo com o 8º, obteremos uma nova ordem, na qual quaisquer dois mintermos adjacentes são passíveis de simplificação. É interessante notar também que o 1º mintermo pode ser simplificado com o 5º, o 2º mintermo pode ser simplificado com o 6º e assim por diante.

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	$A \cdot B \cdot C$



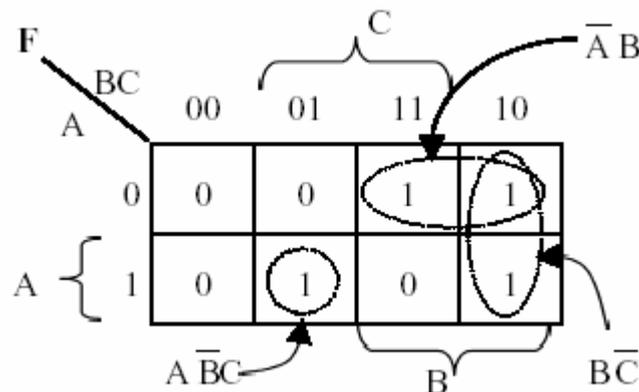
$\bar{A} \cdot \bar{B} \cdot \bar{C}$	$\bar{A} \cdot \bar{B} \cdot C$	$\bar{A} \cdot B \cdot C$	$\bar{A} \cdot B \cdot \bar{C}$
$A \cdot \bar{B} \cdot \bar{C}$	$A \cdot \bar{B} \cdot C$	$A \cdot B \cdot C$	$A \cdot B \cdot \bar{C}$

Repare que nessa nova tabela, quaisquer dois mintermos adjacentes (na horizontal ou na vertical) são passíveis de serem simplificados, pois só se diferenciam de uma variável. É importante ressaltar que esse conceito de adjacência **não está restrito aos limites da tabela**, uma vez que os elementos extremos de uma mesma linha (ou de uma mesma coluna) também são simplificáveis. Isto implica que a tabela de adjacências de mintermos da figura 2.11 pode e deve ser encarada como uma figura geométrica tridimensional do tipo “toróide” (ou uma “rosquinha”). A figura a seguir explicita as relações de adjacência dos mintermos para uma função de três variáveis.



É importante ressaltar que o conceito de adjacência é aplicável na horizontal e na vertical, **mas nunca na diagonal**. Por exemplo, observe que  $m_0$  e  $m_5$  não são adjacentes, pois não estão nem na mesma linha, nem na mesma coluna. O processo de simplificação usando a nova disposição inicia pela construção da tabela em si: os valores da função que se quer simplificar devem ser preenchidos conforme a nova ordem. Após, identificam-se todos os grupos de mintermos-1 adjacentes entre si. Cada grupo origina um termo produto, no qual somente as variáveis comuns a todos os mintermos-1 permanecem. Desde que os grupos de mintermos-1 adjacentes tenham sido corretamente identificados, a simplificação se torna trivial.

**Exemplo 2.3:** simplificar a função  $F$ , cuja tabela verdade encontra-se no item 2.5.2. O primeiro passo é construir uma tabela para  $F$ , usando a nova disposição dos mintermos.



Após, deve-se identificar todos os grupos de mintermos-1 adjacentes entre si. Cada grupo de mintermos-1 originará um produto, conforme indicado na figura 213. A equação em soma de produtos simplificada será o **OU** entre os produtos encontrados:

$$F = \bar{A} B + A \bar{B} C + B \bar{C}$$

### Mapas de Karnaugh e Subcubos

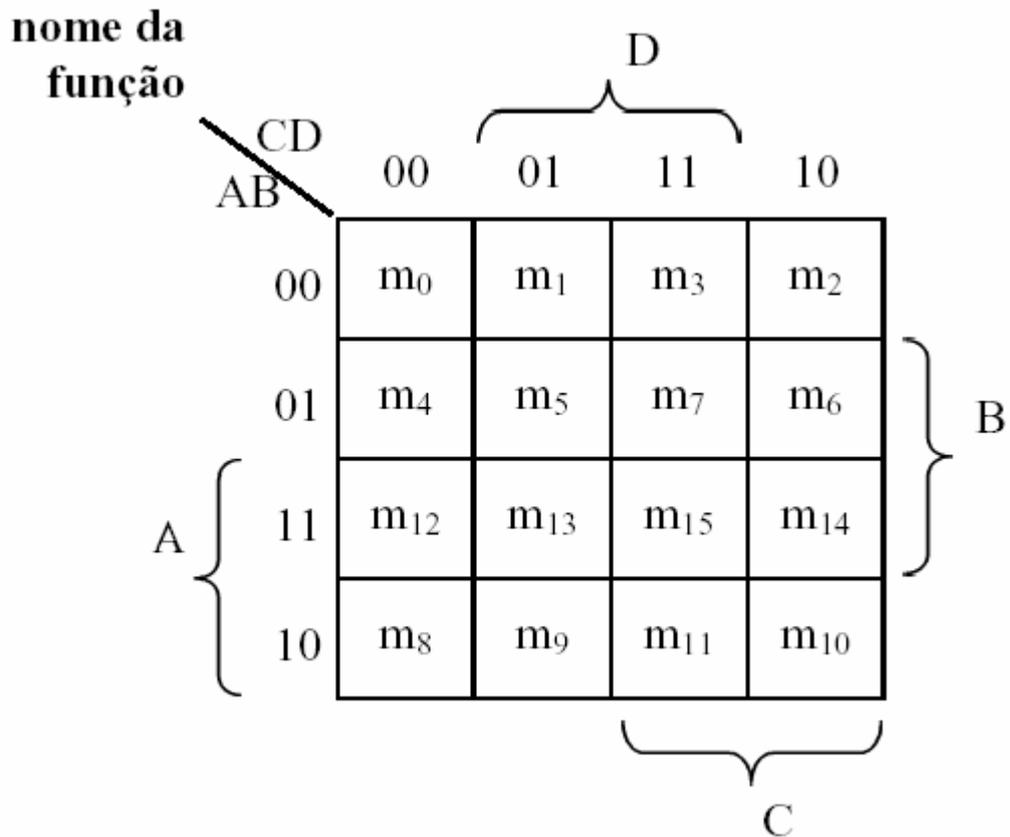
A tabela modificada usada para a identificação dos mintermos-1 adjacentes no exemplo anterior é denominada **mapa de Karnaugh** (no caso, para 3 variáveis). Na figura 2.14 são mostrados os mapas de Karnaugh para funções de 2, 3 e 4 variáveis.

nome da  
função

		B	
		0	1
A	0	m <sub>0</sub>	m <sub>1</sub>
	1	m <sub>2</sub>	m <sub>3</sub>

nome da  
função

		C			
		00	01	11	10
A	0	m <sub>0</sub>	m <sub>1</sub>	m <sub>3</sub>	m <sub>2</sub>
	1	m <sub>4</sub>	m <sub>5</sub>	m <sub>7</sub>	m <sub>6</sub>



O primeiro passo para simplificar-se uma função usando mapa de Karnaugh é escolher o mapa conforme o número de variáveis da função e preencher os valores dos mintermos conforme a tabela verdade fornecida, ou conforme a equação fornecida. O segundo passo é identificar grupos de mintermos adjacentes que formem grupos de  $2^m$  elementos adjacentes entre si, com  $0 \leq m \leq n$ , onde  $n$  é o número de variáveis da função. Estes grupos são denominados **subcubos**.

No caso de se querer encontrar uma expressão em soma de produtos, estaremos interessados nos subcubos de mintermos-1. Então, cada subcubo contendo mintermos-1 irá originar um produto, no qual uma ou mais variáveis poderão estar ausentes devido à simplificação que é obtida. Os produtos associados aos subcubos de mintermos-1, simplificados ou não, são denominados **implicantes**. É importante ressaltar que quanto maior o número de elementos do subcubo, maior será a simplificação obtida.

Exemplo:

Determinar os implicantes do Mapa a seguir:

		D				
		⏟				
F0	CD	00	01	11	10	
	AB					
	00	0	0	0	0	} B
	01	1	1	1	1	
	11	0	0	0	0	
10	0	0	0	0		
		⏟				C
						} A

No caso de se querer encontrar uma expressão em produtos de somas, estaremos interessados nos subcubos de mintermos-0. Então, cada subcubo contendo mintermos-0 irá originar uma soma, no qual uma ou mais variáveis poderão estar ausentes devido à simplificação que é obtida. As somas associadas aos subcubos de mintermos-0, simplificadas ou não, são denominadas **implicados**. Também neste caso, quanto maior o número de elementos do subcubo, maior será a simplificação obtida.

**Exemplo 2.5:** determinar os implicados das funções dadas a seguir.

		D				
		⏟				
F7	CD	00	01	11	10	
	AB					
	00	0	1	1	1	} B
	01	1	1	1	1	
	11	1	1	1	1	
10	1	1	0	0		
		⏟				C
						} A

## Cobertura dos Mapas de Karnaugh

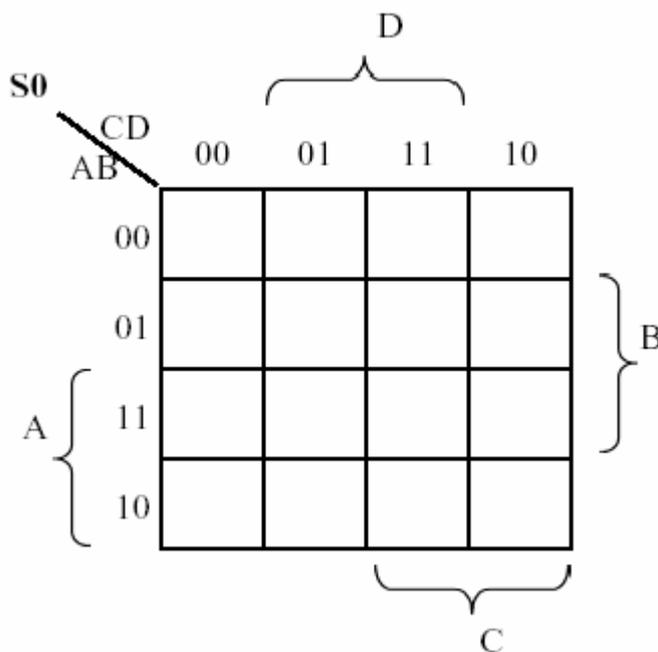
Normalmente, é possível identificar-se numa mesma função Booleana mais de um implicante (ou mais de um implicado). Neste caso, é necessário determinar o conjunto de implicantes (ou implicados) que melhor “cobrem” a função, onde a melhor cobertura significa necessariamente a expressão mais simplificada possível, a qual é denominada expressão mínima.

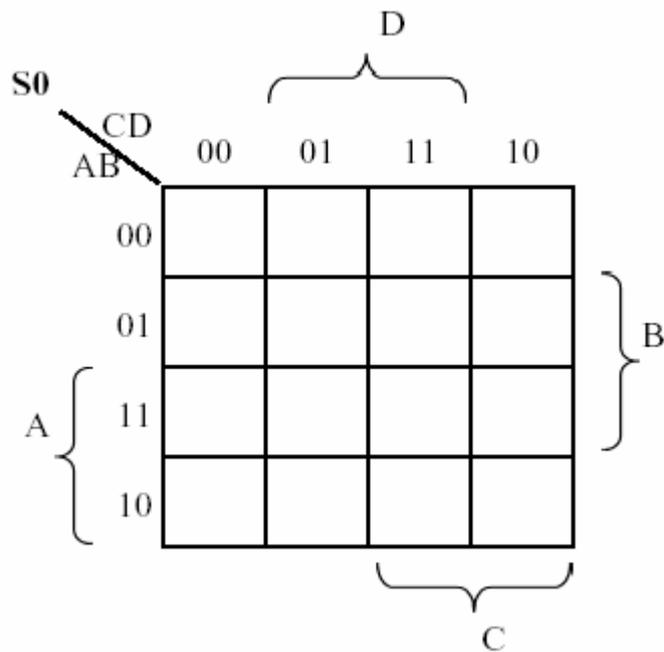
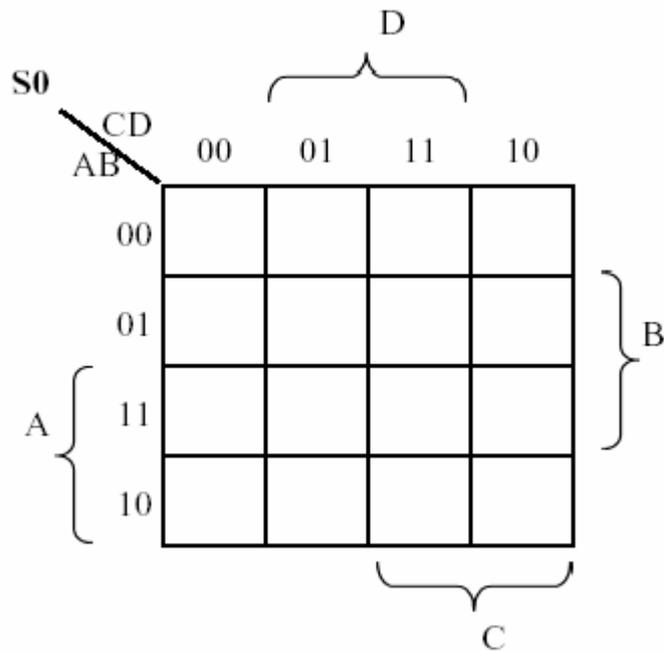
O procedimento básico para se determinar a melhor cobertura (também chamada cobertura mínima) para uma expressão em soma de produtos é o seguinte:

- Identificar os subcubos de mintermos-1 com maior número de elementos possível, iniciando do tamanho  $2^n$ , onde  $n$  é o número de variáveis da função. Caso algum mintermo-1 fique isolado (isto é, não há nenhum outro mintermo-1 adjacente a ele), então ele constituirá um subcubo de um elemento;
- Identificar o menor conjunto de subcubos de modo que cada mintermo-1 pertença a pelo menos um subcubo (= seja coberto pelo menos uma vez).

Observações:

1. Cada mintermo-1 pode ser coberto por mais de um subcubo, caso isso resulte numa simplificação maior;
2. Um último teste para verificar se a expressão obtida é realmente a mínima consiste em verificar se algum subcubo pode ser removido, sem deixar algum mintermo-1 descoberto. Um subcubo que poder ser removido sem descobrir mintermos é dito subcubo **não-essencial**. Logo, todo o subcubo que não pode ser removido é dito **essencial**;
3. Pode haver mais de uma expressão mínima para uma mesma função Booleana;
4. A expressão mínima é aquela de menor complexidade. E a complexidade será medida pelo número de literais de uma função.





Até agora, tem-se visto apenas duas funções booleanas de duas variáveis, OU e E. Mas, existem  $2^{2^2}$  funções Booleanas com n variáveis binárias. Assim, existem 16 funções Booleanas de duas variáveis e as funções E e OU são apenas duas dessas 16 funções. A tabela a seguir lista todas as 16 funções Booleanas de duas variáveis, x e y.

Nome	Símbolo	Valores para x,y= 00 01 10 11	Expressão algébrica	Comentário
Zero	<b>0</b>	0 0 0 0	$F_0 = 0$	Constante 0
E	x.y	0 0 0 1	$F_1 = xy$	x e y
Inibição	x/y	0 0 1 0	$F_2 = x\bar{y}$	x mas não y
Transferência		0 0 1 1	$F_3 = x$	x
Inibição	y/x	0 1 0 0	$F_4 = \bar{x}y$	y mas não x
Transferência		0 1 0 1	$F_5 = y$	y
XOR	$x \oplus y$	0 1 1 0	$F_6 = x\bar{y} + \bar{x}y$	x ou y, mas não ambos
OU	x+y	0 1 1 1	$F_7 = x + y$	x ou y
NOR	$x \downarrow y$	1 0 0 0	$F_8 = (x + y)'$	Não-OU
Equivalência		1 0 0 1	$F_9 = xy + x' y'$	x igual a y
Complemento	y'	1 0 1 0	$F_{10} = y'$	Não y
Implicação	$x \subset y$	1 0 1 1	$F_{11} = x + \bar{y}$	Se y, então x
Complemento	x'	1 1 0 0	$F_{12} = \bar{x}$	Não x
Implicação	$x \supset y$	1 1 0 1	$F_{13} = \bar{x} + y$	Se x, então y
NAND	$x \uparrow y$	1 1 1 0	$F_{14} = (xy)'$	Não E
Um	<b>1</b>	1 1 1 1	$F_{15} = 1$	Constante 1