4. Derivação de Expressões Booleanas

Dada uma função Booleana, descrita por sua tabela verdade, derivar uma expressão Booleana para esta função é encontrar uma equação que a descreva. Logo, a derivação de expressões Booleanas é o problema inverso da avaliação de uma expressão Booleana, descrito na seção **Erro! Fonte de referência não encontrada.**.

Há basicamente duas maneiras de se definir (ou descrever) uma função Booleana:descrevendo-se todas as situações das variáveis de entrada para as quais a função vale **1** ou,alternativamente, todas as situações em que a função vale **0**. O primeiro método é conhecido por **soma de produtos** (SdP), enquanto que o segundo é chamado **produto de somas** (PdS).
Qualquer função Booleana pode ser descrita por meio de soma de produtos ou por meio de produto de somas. Como as funções Booleanas só podem assumir um dentre dois valores (0ou 1), basta usar-se um dos dois métodos para se encontrar uma equação para uma função.A seguir, são detalhados os métodos de derivação de expressões Booleanas.

4.1 Derivação de Expressões usando Soma de Produtos (SdP)

Dada uma função Booleana de n variáveis (ou seja, n entradas), haverá 2^n combinações possíveis de valores. Dizemos que esse conjunto de valores que as variáveis podem assumir, juntamente com os respectivos valores da função, constituem o espaço da função. A cada combinação de entradas podemos associar um **termo produto,** no qual todas as variáveis da função estão presentes, e que é construído da seguinte forma: se a variável correspondente vale ‘**0’**, ela deve aparecer negada; se a variável vale ‘**1’**, ela deve aparecer não negada. Por exemplo, a Tabela 1 a seguir lista os termos produto associados a cada combinação de entradas para uma função Booleana de três variáveis A, B e C.

Tabela 1. **Tabela com os mintermos para 3 variáveis**

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

Cada termo produto construído conforme a regra anteriormente descrita é denominado **mintermo** (ou minitermo). Note que, para um dado mintermo, se substituirmos os valores das variáveis associadas, obteremos 1. Porém, se substituirmos nesse mesmo mintermo quaisquer outras combinações de valores, obteremos 0. Dessa forma, se quisermos encontrar a equação para uma função a partir de sua tabela verdade, basta montarmos um **OU** entre os mintermos associados aos **1**s da função (também chamados **mintermos 1**).

Exemplo 4.1: encontrar a equação em soma de produtos (SdP) para a função F,descrita pela tabela verdade apresentada na Tabela 2:

Tabela 2. **Tabela verdade da função F**

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Na tabela acima, F é função das variáveis A, B e C e os valores de (A,B,C) para os quais F=1 são (0,1,0),(0,1,1), (1,0,0) e (1,1,1). Logo, a equação em soma de produtos para F será o **OU** entre estes produtos, conforme segue:

$$F = \overline{A} \cdot B \cdot \overline{C} + \overline{A} \cdot B \cdot C + A \cdot \overline{B} \cdot \overline{C} + A \cdot B \cdot C$$

A fim de simplificar a notação, o símbolo da operação **E** pode ser omitido. Desta forma, a equação anterior pode ser reescrita de maneira mais concisa:

$$F = \overline{A}\,B\,\overline{C} + \overline{A}\,B\,C + A\,\overline{B}\,\overline{C} + A\,B\,C$$

4.2 Derivação de Expressões usando Produto de Somas (PdS)

O método de derivação usando produto de somas é o **dual** (isto é, o oposto) do método de derivação em soma de produtos. A cada combinação das variáveis de entrada de uma função podemos associar um **termo soma,** no qual todas as variáveis da função estão presentes, e que é construído da seguinte forma: se a variável correspondente vale '**1**', ela deve aparecer negada; se a variável vale '**0**', ela deve aparecer não negada. A Tabela 3 a seguir lista os termos soma associados a cada combinação de entradas para uma função Booleana de três variáveis (A, B e C, por exemplo).

Tabela 3. **Tabela com os maxtermos para 3 variáveis**

A	B	C	Maxtermos
0	0	0	$A + B + C$
0	0	1	$A + B + \overline{C}$
0	1	0	$A + \overline{B} + C$
0	1	1	$A + \overline{B} + \overline{C}$
1	0	0	$\overline{A} + B + C$
1	0	1	$\overline{A} + B + \overline{C}$
1	1	0	$\overline{A} + \overline{B} + C$
1	1	1	$\overline{A} + \overline{B} + \overline{C}$

Cada termo soma construído conforme a regra anteriormente descrita é denominado **maxtermo** (ou maxitermo). Note que, para um dado maxtermo, se substituirmos os valores das variáveis associadas, obteremos 0. Porém, se substituirmos nesse mesmo maxtermo quaisquer outras combinações de valores, obteremos 1. Dessa forma, se quisermos encontrar a equação para uma função a partir de sua tabela verdade, basta montarmos um **E** entre os maxtermos associados aos **0**s da função (também chamados **maxtermos 0**).

Exemplo 4.2: encontrar a equação em produto de somas (PdS) para a função F,descrita pela tabela verdade abaixo (Tabela 4):

Tabela 4. **Tabela verdade da função F exemplo**

A	B	C	F
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1

Foi escolhida a mesma função do exemplo anterior, para que se possa estabelecer comparações entre os dois métodos de derivação. Os valores das variáveis de entrada A,B e C para os quais F=0 são (0,0,0), (0,0,1), (1,0,1) e (1,1,0). Logo, a equação em produto de somas para F será o **E** entre estas somas:

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

Note que a ordem de precedência de uma expressão em produto de somas é “primeiro cada soma deve ser avaliada, para só então avaliar-se o produto”. Isto significa que os parêntesis em torno de cada termo soma **são obrigatórios**! Repare também que os símbolos referentes à operação **E** (entre os termos soma) podem ser omitidos.

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

4.3 Formas Canônicas, Padrão e Não-Padrão

As representações em soma de produtos e em produto de somas são denominadas formas **padrão**. A soma de produtos e o produto de somas descritos nas duas seções anteriores apresentam ainda uma característica bastante particular: em cada termo soma e em cada termo produto todas as variáveis da função estão presentes. Devido a essa característica,essas formas são chamadas **canônicas**. Neste curso adotaremos a forma de representação de mintermos como sendo a forma canônica adotada para representar as expressões.

Apesar da praticidade das representações canônicas, elas são pouco úteis para a implementação de circuitos digitais. O número de elementos (portas lógicas e conexões) de um circuito lógico depende diretamente do número de operações Booleanas (inversão, E e OU) contidas na expressão associada. Desta forma, é normal que se deseje reduzir o número de operações contidas numa função, de modo a poder-se implementá-la com circuitos lógicos mais simples, e portanto, de menor custo. A redução do número de operações é obtida mediante a eliminação de literais da expressão, aplicando-se as propriedades da álgebra Booleana descritas na seção **Erro! Fonte de referência não encontrada.**. Um literal é uma variável negada ou uma variável não negada. O processo de redução de literais (ou de redução de operações, equivalentemente) é denominado **simplificação**.

Para exemplificar os passos básicos para a simplificação algébrica (literal) de expressões Booleanas, tomemos a expressão canônica, em soma de produtos, para a função F:

$$F = \overline{A}\,B\,\overline{C} + \overline{A}\,B\,C + A\,\overline{B}\,C + A\,B\,\overline{C} \quad (e1)$$

O primeiro passo é identificar pares de mintermos que se diferenciam por apenas um literal, a fim de aplicar a propriedade (14 apresentada na seção **Erro! Fonte de referência não encontrada.**). Os mintermos ABC e ABC, por exemplo,possuem os mesmos literais, exceto pela variável C: no primeiro, o literal é C, enquanto no segundo, o literal é C. Então, com o uso da propriedade (14), pode-se fatorar esses dois mintermos, obtendo-se:

$$F = \overline{A}\,B\,(\overline{C} + C) + A\,\overline{B}\,C + A\,B\,\overline{C} \quad (e2)$$

Pela propriedade (4), tem-se que $C + \overline{C} = 1$. Então, obtém-se:

$$F = \overline{A}\,B \cdot (1) + A\,\overline{B}\,C + A\,B\,\overline{C} \quad (e3)$$

E pela propriedade (6), $\overline{A}\,B \cdot 1 = \overline{A}\,B$. Então, obtém-se:

$$F = \overline{A}\,B + A\,\overline{B}\,C + A\,B\,\overline{C} \quad (e4)$$

Assim, pela manipulação algébrica, obtivemos uma expressão em soma de produtos que é simplificada em relação a sua expressão em soma de produtos na forma canônica, pois o número de operações e também de literais foram reduzidos.

Entretanto, na equação (e1), o mintermo $\overline{A}B\overline{C}$ também poderia ter sido agrupado como mintermo $AB\overline{C}$, pois ambos possuem os mesmos literais, exceto pela variável A ($\overline{A}$ no primeiro e A no segundo). Naturalmente, os passos a serem seguidos seriam os mesmos descritos anteriormente. E a equação resultante seria um pouco diferente, mas com o mesmo número de operações sendo, portanto, de mesma complexidade. Na verdade, o melhor seria se pudéssemos agrupar o mintermo $\overline{A}B\overline{C}$ com o mintermo $\overline{A}BC$ e ao mesmo tempo com o mintermo $AB\overline{C}$. Felizmente, a propriedade (3) da álgebra Booleana diz que o **OU** entre duas ou mais variáveis Booleanas iguais é igual a própria variável Booleana em questão.

Estendendo esta propriedade, pode-se dizer que o **OU** entre duas ou mais funções (inclusive produtos) Booleanas iguais equivale à própria função Booleana em questão. Desta forma,pode-se expandir o mintermo $\overline{A}B\overline{C}$ para:

$$\overline{A}B\overline{C} = \overline{A}B\overline{C} + \overline{A}B\overline{C} \quad (e5)$$

Retomando a equação (e1) e utilizando (e5) , segue que:

$$F = \overline{A}\,B\,\overline{C} + \overline{A}\,B\,C + A\,\overline{B}\,C + A\,B\,\overline{C} + \overline{A}\,B\,\overline{C} \quad \text{(e6)}$$

Então, a propriedade (3) garante que as expressões (e1) e (e6) são equivalentes,embora o mintermo $\overline{A}B\overline{C}$ apareça duplicado. E pelo fato de aparecer duas vezes, pode-se usar uma cópia de $\overline{A}B\overline{C}$ para simplificar com $\overline{A}BC$ e outra para simplificar com $AB\overline{C}$. Os passos da simplificação são os mesmos já descritos: pela propriedade (14), segue:

$$F = \overline{A}\,B \cdot (\overline{C} + C) + A\,\overline{B}\,C + (\overline{A} + A)\cdot B\,\overline{C} \quad \text{(e7)}$$

E pela propriedade (6), vem:

$$F = \overline{A}\,B \cdot (1) + A\,\overline{B}\,C + (1)\cdot B\,\overline{C} \quad \text{(e8)}$$

Finalmente, pela propriedade (4), tem-se:

$$F = \overline{A}\,B + A\,\overline{B}\,C + B\,\overline{C} \quad \text{(e9)}$$

Repare que o mintermo $A\overline{B}C$ não pôde ser agrupado com nenhum outro mintermo.Note também que foram feitas todas as simplificações possíveis, uma vez que foram agrupados e simplificados todos os pares de mintermos que se diferenciam de somente uma variável. Logo, a expressão (e9) representa a máxima simplificação possível sob a forma de soma de produtos. E por esse motivo, ela é dita equação **mínima** em **soma de produtos** da função F. Quanto a expressão (e4), diz-se ser uma equação em **soma de produtos simplificada** (porém, não-mínima). Logo, toda equação mínima é simplificada, porém, nem toda equação que foi simplificada é necessariamente mínima.

Embora a equação mínima em soma de produtos apresente menor número de operações Booleanas que a representação na forma canônica, as vezes pode ser possível reduzir-se ainda mais o número de operações, fatorando-se literais. Por exemplo, na expressão (e9) pode-se fatorar o primeiro e o terceiro mintermos como segue:

$$F = B \cdot (A + \overline{C}) + A\,\overline{B}\,C \quad \text{(e10)}$$

A expressão (e10), obtida pela fatoração de (e9), não é nem do tipo soma de produtos,nem produto de somas, pois há um termo que não é nem produto, nem soma. Diz-se que a expressão está na **forma fatorada**. No caso de (e10), a fatoração não resultou em redução do número de operações.

No que se refere a terminologia, as formas soma de produtos e produto de somas são ditas **formas padrão** (formas *standard*). A forma fatorada é dita **não-padrão**. As formas **canônicas** são, pois, casos especiais de formas padrão, nas quais os termos são mintermos ou maxtermos. A fim de diferenciar somas de produtos canônicas de somas de produtos simplificadas, usaremos a expressão "**soma de**

mintermos”. De maneira similar, usaremos a expressão “**produto de maxtermos”** para diferenciar produtos de somas canônicos de produtos de somas simplificados.

4.4 Circuitos Lógicos para Formas Padrão e Não-Padrão

As regras gerais para se realizar o desenho de circuitos lógicos já foram apresentadas na seção **Erro! Fonte de referência não encontrada.**. No caso de equações na forma soma de produtos (canônica ou simplificada), há um primeiro nível (desconsiderando-se possíveis inversores), constituído somente por portas **E**,onde cada porta **E** implementa um dos produtos da equação. Há ainda um segundo nível,constituído por uma porta **OU**, responsável pela “soma” lógica dos produtos. A figura abaixo (Figura 1) mostra um possível circuito lógico para a equação (e1).

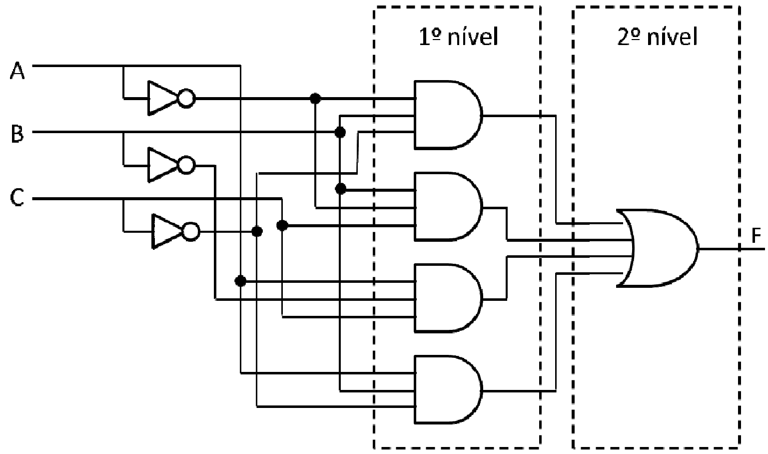

Figura 1. Possível circuito lógico que implementa equação (e1).

A fim de facilitar a compreensão do circuito acima, as seguintes regras devem ser observadas:

- as variáveis de entrada devem ser identificadas preferencialmente à esquerda, junto aos respectivos fios;
- inversores devem ser providos para as variáveis que aparecem negadas na equação;
- as portas que implementam as operações Booleanas que aparecem na equação normalmente são posicionadas da esquerda para a direita, seguindo a ordem de avaliação dos operadores.

Repare que em todas as interseções de fios em que há conexão física, **deve haver um ponto** (suficientemente grande), como se fora uma “solda”. Logo, quando não há o referido ponto na interseção de fios, significa que tais fios estão “eletricamente isolados”.

O circuito da equação (e1) ainda pode ser desenhado utilizando-se uma notação simplificada para os inversores das entradas. Ao invés de se desenhar um inversor para cada variável que aparece negada na equação, coloca-se um círculo junto a cada entrada de cada porta na qual há uma variável negada. A aplicação desse procedimento para o circuito da equação (e1) resulta no circuito apresentado na Figura 2:

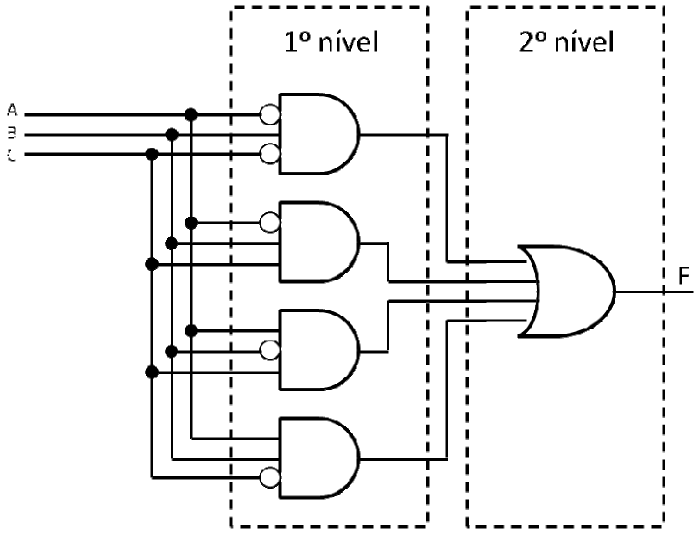

Figura 2. Um circuito lógico para soma de produtos.

No caso de equações na forma produto de somas (canônica ou simplificada), o primeiro nível é constituído por portas **OU**, sendo cada uma responsável por uma das "somas"lógicas da equação. O segundo nível, por sua vez, é constituído por uma porta **E**, que realiza o produto lógico das parcelas. A Figura 3 mostra um possível circuito lógico para a equação (e1).

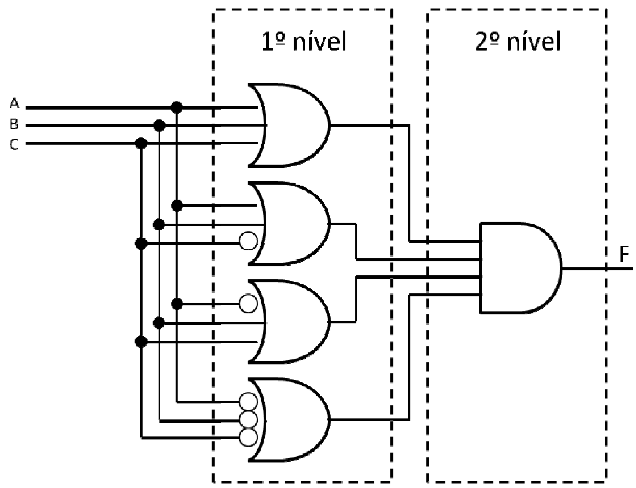

Figura 3. Um circuito lógico para produto de somas.

Pelo fato de apresentarem apenas dois níveis de portas (dois níveis lógicos), circuitos para equações representadas nas formas padrão, canônicas ou simplificadas, são ditos **circuitos em dois níveis** (ou **lógica a dois níveis**).

A Figura 4 mostra o circuito lógico para a equação (e9), que é a **forma mínima** para a função da equação (e1). Note que este circuito é de menor complexidade que o circuito da Figura 2 e Figura 3.

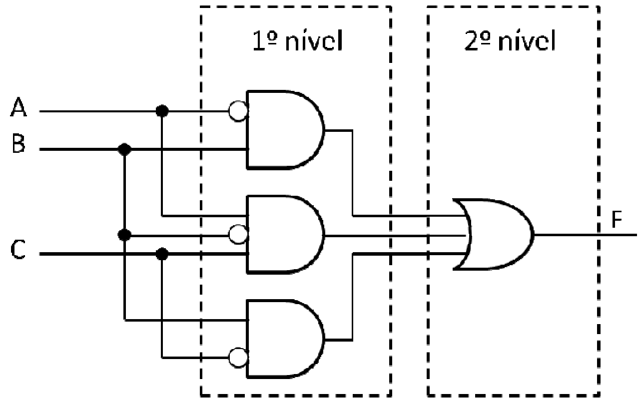

Figura 4. Circuito lógico para a equação (e9).

Dada uma equação canônica de uma função qualquer, o circuito para uma equação simplificada a partir da canônica possui menos portas e/ou portas de menor complexidade. (A complexidade relativa de uma porta lógica pode ser medida pelo número de entradas que ela possui). A complexidade relativa de um circuito lógico pode ser calculada somando-se o número de entradas das portas do circuito. No circuito da Figura 2 e da Figura 3 há 4 portas de 3 entradas e 1 porta de 4 entradas. Então, a complexidade relativa será 4x3+1x4=16. No circuito da figura2.8 há 2 portas de 2 entradas e 2 portas de 3 entradas. Sua complexidade relativa será2x2+2x3=10. Claramente, o circuito da Figura 4 é de menor complexidade que os circuitos da Figura 2 e da Figura 3. Estes dois são de mesma complexidade relativa. No cálculo da complexidade relativa, as inversões normalmente não são levadas em conta.Circuitos para formas fatoradas podem ser vistos como o caso mais genérico. Em geral, as formas fatoradas conduzem a circuitos cujo número de níveis lógicos é maior do que dois. Por isso, circuitos lógicos para formas fatoradas são denominados **circuitos multinível** (**lógica multinível**). Como dito na seção 4.4, as vezes uma forma fatorada pode apresentar menor número de operações do que a respectiva forma padrão. Quando isso ocorre, o circuito associado à forma fatorada também será de menor complexidade relativa. Entretanto, se não ocorrer redução no número de operações, mesmo assim é possível que o circuito para a forma fatorada seja de menor complexidade relativa, pois o conceito de complexidade relativa também inclui o número de entradas de cada porta. Então, a maneira mais segura de saber se o circuito associado à forma fatorada é de menor complexidade ou não é desenhá-lo e somar o número de entradas. A Figura 5 mostra o circuito para a equação 2.21, obtida a partir da equação (e9) fatorando-se o literal B.

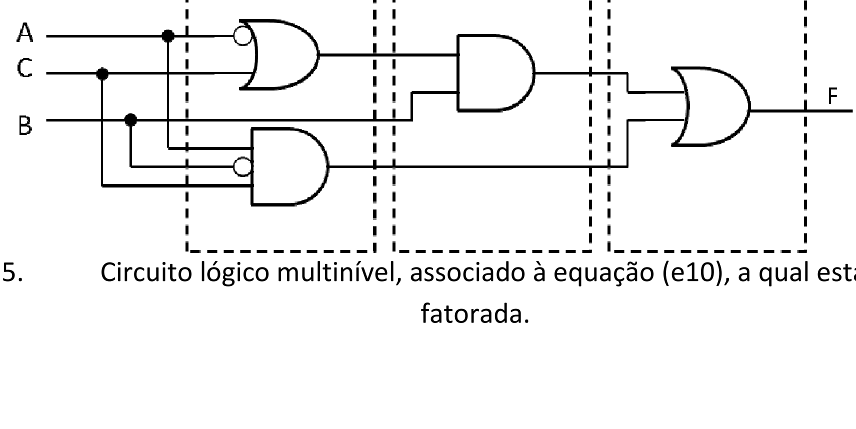

Figura 5. Circuito lógico multinível, associado à equação (e10), a qual está na forma fatorada.

Note que o número de operações Booleanas destas equações é o mesmo: 4. No entanto, a complexidade do circuito da forma fatorada é 3x2+1x3=9, portanto menor do que a complexidade do circuito da Figura 4.

5. LEIS FUNDAMENTAIS E PROPRIEDADES DA ÁLGEBRA BOOLEANA

As leis da álgebra Booleana dizem respeito ao espaço Booleano (isto é., valores que uma variável pode assumir) e operações elementares deste espaço. Já as propriedades podem ser deduzidas a partir das definições das operações.

Sejam A e B duas variáveis Booleanas. Então, o espaço Booleano é definido:

se $A \neq 0$, então $A=1$;
se $A \neq 1$, então $A=0$.

As operações elementares deste espaço são operações **OU**, operações **E** e **complementação**, cujas definições foram dadas nas seções anteriores.

As propriedades da álgebra Booleana são as seguintes.

Da adição lógica:

$$A+0 = A \quad (1)$$
$$A+1= 1 \quad (2)$$
$$A+ A= A \quad (3)$$
$$A+ \overline{A}= 1 \quad (4)$$

Da multiplicação lógica:

$$A \cdot 0 = 0 \quad (5)$$
$$A \cdot 1 = A \quad (6)$$
$$A \cdot A = A \quad (7)$$
$$A \cdot \overline{A} = 0 \quad (8)$$

Da complementação:

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

Comutatividade:

$$A + B = B + A \quad (10)$$
$$A \cdot B = B \cdot A \quad (11)$$

Associatividade:

$$A + (B + C) = (A + B) + C = (A + C) + B \quad (12)$$
$$A \cdot (B \cdot C) = (A \cdot B) \cdot C = (A \cdot C) \cdot B \quad (13)$$

Distributiva (da multiplicação em relação à adição):

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

5.1 Teoremas de De Morgan

O primeiro teorema de De Morgan diz que a complementação de um produto lógico equivale à soma lógica das negações de cada variável do referido produto. A equação abaixo generaliza este teorema:

$$\overline{A \cdot B \cdot C \cdot \ldots} = \overline{A} + \overline{B} + \overline{C} + \ldots$$

O segundo teorema é o contraponto do primeiro, ou seja, a complementação de uma soma lógica equivale ao produto lógico das negações individuais das variáveis:

$$\overline{A + B + C + \ldots} = \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \ldots$$

5.2 Propriedade Especial

$$A + \bar{A}.B = A + B$$

1 Complete cada expressão

a) $A + 1 = 1$
b) $A . A = A$
c) $B . \overline{B} = 0$
d) $C + C = C$
e) $x . 0 = 0$
f) $D . 1 = D$
g) $D + 0 = D$
h) $C + \overline{C} = 1$
i) $G + GF = G . (1 + F) = G . (1) = G$
j) $y + \overline{w}y = y . (1 + /w) = y . (1) = y$

2 Simplifique a seguinte expressão

$X = (M + N)(\overline{M} + P)(\overline{N} + \overline{P})$

3 Simplifique as seguintes expressões utilizando álgebra booleana.

a) $x = ABC + \overline{A}C$
b) $y = (Q + R)(\overline{Q} + \overline{R})$
c) $w = ABC + A\overline{B}C + \overline{A}$
d) $q = \overline{RST}\ (\overline{R + S + T})$
e) $x = \overline{A}\overline{B}\overline{C} + \overline{A}BC + ABC + A\overline{B}\overline{C} + A\overline{B}C$
f) $z = (B + \overline{C})(\overline{B} + C) + \overline{A} + B + \overline{C}$
g) $y = (\overline{C + D}) + \overline{A}C\overline{D} + A\overline{B}\overline{C} + \overline{A}\overline{B}CD + AC\overline{D}$
h) $x = AB(CD) + ABD + BCD$