Apostila de Circuitos Digitais

Julia Acras, Isabella Stuart, Nathalie Godoi e Rafaela L.

XX/XX/2025

Sumário

1 Terminologia e Símbolos

Neste seção, apresentamos os principais símbolo usados nas operações booleanas para garantir o entedimento dos conceitos apresentados no texto.

Símbolo	Operação / Descrição
$+$	OR (Disjunção Lógica)
$\cdot$	AND (Conjunção Lógica)
$\overline{X}$	NOT (Negação de X)
$\sum$	Soma de mintermos (forma canônica disjuntiva)
$\prod$	Produto de maxtermos (forma canônica conjuntiva)

2 Tabela Verdade

2.1 Introdução

A tabela verdade é um quadro que representa todas as possíveis combinações lógicas entre variáveis booleanas. Ela mostra como os valores de entrada (0 ou 1) influenciam a saída de um circuito.

Para construí-la, divide-se a tabela em duas partes: entradas (ou variáveis) e saídas (ou combinações resultantes). A cada nova variável adicionada, o número de combinações dobra, pois cada variável pode assumir dois valores: 0 ou 1.

Assim, para n variáveis, o número total de linhas na tabela será 2^n. Por exemplo, com 7 variáveis, temos $2^7 = 128$ combinações possíveis.

Observação: Neste material, utilizamos $\overline{A}$ para representar a negação da variável A.

2.2 Construção da Tabela Verdade

Na Tabela Verdade, as variáveis de entrada são colocadas do lado esquerdo, enquanto a função resultante que representa o circuito aparece à direita. Por exemplo, na função f definida a partir da variável A, colocamos na coluna da esquerda A e, na direita, o resultado, que depende do valor lógico de A. Assim, teremos:

Variável de entrada (A) $\longrightarrow$ | **A** | $\overline{A}$ | $\longleftarrow$ **Saída** ($\overline{A}$)

A	$\overline{A}$
1	0
0	1

$f = \overline{A}$

2.3 Operações Lógicas

- **Conjunção (AND) :** $A \cdot B$ - verdadeiro somente se A e B forem verdadeiros.
- **Disjunção (OR):** $A + B$ - falso somente se A e B forem falsos.
- **Negação (NOT):** $\overline{A}$ - inverte o valor lógico de A.
- **Implicação:** $A \to B$ - falso apenas se A for verdadeiro e B falso.
- **Bicondicional:** $A \leftrightarrow B$ - verdadeiro se A e B tiverem o mesmo valor lógico.

Exemplo de Negação (NOT):

$f = \overline{A}$

A	$\overline{A}$
1	0
0	1

$\overline{A}$ tem sempre o valor lógico oposto de A.

Exemplo de Conjunção (AND):

$s = A \cdot B$

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

A função só é verdadeira quando A e B são ambos 1.

Exemplo de Disjunção (OR):

$g = A + B$

A	B	$A + B$
0	0	0
0	1	1
1	0	1
1	1	1

A função só é verdadeira quando A ou B são 1.

Exemplo Composto: Disjunção, Conjunção e Negação:

Considere a função:

$$h = \overline{A \cdot B} \cdot C$$

Para construir a tabela verdade dessa função, primeiro calculamos$A \cdot B$, depois aplicamos a negação, e, por fim, combinamos esse resultado com C.

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

Note que a função h só é verdadeira quando C e $\overline{A \cdot B}$ são ambos verdadeiras.

Essa tabela permite acompanhar passo a passo o cálculo da função h para todas as combinações possíveis das variáveis A, B e C.

Observação: números de linhas da Tabela Verdade

A cada nova variável adicionada à tabela, o número de linhas dessa tabela verdade dobra. Dá para perceber isso comparando, por exemplo, os exemplos 1 e 2 - onde foram adicionadas 2 novas possibilidades - ou então do exemplo 3 para o 4, que aumentaram 4 possibilidade.

3 Álgebra de Boole

A Álgebra de Boole utiliza apenas dois valores:

- Verdadeiro: 1
- Falso: 0

3.1 Operações Fundamentais

As três operações fundamentais são:

1. **AND** (E lógico)
   A saída da operação **AND** é verdadeira somente quando todas as entradas são verdadeiras, ou seja, basta que apenas uma entrada seja falsa (0) para tornar todo o circuito falso.
2. **OR** (OU lógico)
   A saída é verdadeira se ao menos uma das entradas é verdadeira. Logo, a única situação em que o circuito tem seu valor lógico falso é quando todas as entradas são 0.
3. **NOT** (Negação)
   A saída é o inverso da entrada. Se a entrada for verdadeira (1), a saída será falsa (0) e vice-versa.

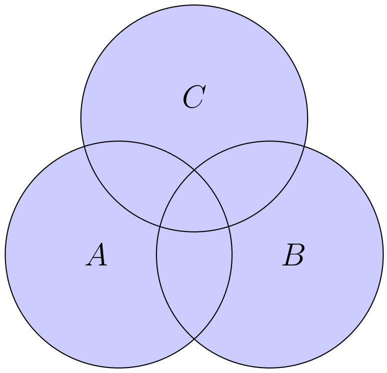
Observação: Relação com Diagrama de Venn

As operações booleanas também podem ser representadas por diagramas de Venn:

- **AND** – Interseção entre os conjuntos A e B
- **OR** – União dos conjuntos A e B
- **NOT** – Complemento do conjunto A

3.2 Representação na forma de Diagrama de Venn

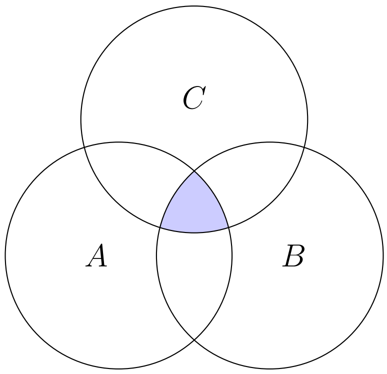
Diagrama de Venn da expressão $f = A + B + C$



Região destacada: $A \cup B \cup C$

Explicação: Nesse exemplo, para que a função f acima tenha valor lógico 1, é preciso que ao menos **UMA** das variáveis seja verdadeira.

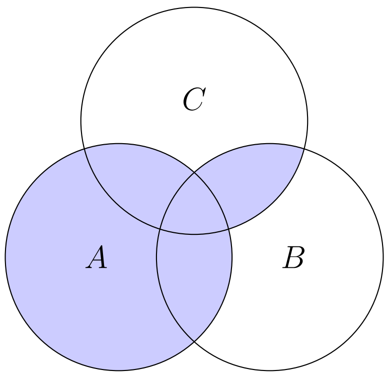
Diagrama de Venn da função: $f = A \cdot B \cdot C$



Região destacada: $A \cap B \cap C$

Explicação: Diferente do exemplo anterior, para que essa função f seja verdadeira, é preciso que **TODAS** as variáveis tenham valor lógico 1.

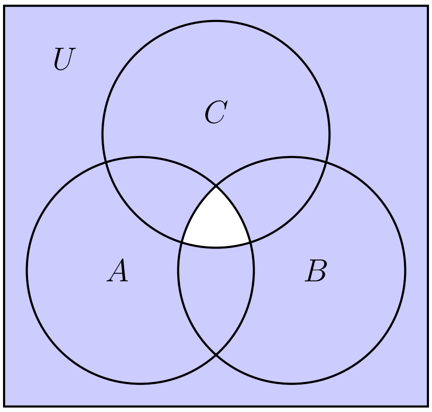
Diagrama de Venn da expressão: $f = A + (B \cdot C)$



Região destacada: $A \cup (B \cap C)$

Explicação:

Diagrama de Venn da expressão: $g = \overline{A \cdot B \cdot C}$



Região destacada: $\overline{A \cap B \cap C}$

Explicação: Esse exemplo é equivalente ao complemento da interseção de todas as variáveis. Por exemplo, se uma função f é determinada por essa interseção, e como visto anteriormente essa expressão só é verdadeira quando todas as variáveis são simultaneamente 1, temos que nossa função g é verdadeira sempre que f for falsa, ou seja, quando ao menos uma das variáveis não for 1.

4 Expressões, Axiomas e Simplificação

Sendo X, Y e Z variáveis que assumem os valores lógicos 0 e 1, e utilizando os símbolos:

- $+$ para disjunção (OU lógico)
- $\cdot$ para conjunção (E lógico)
- $\overline{X}$ para negação de X

As propriedades fundamentais da Álgebra de Boole serão representadas a seguir, separadas pelas operações OR e AND, para facilitar a compreensão.

4.1 Versão OR (Soma lógica)

Propriedade (OR)	**Expressão**
1 - Identidade	$X + 0 = X$
2 - Elemento Nulo	$X + 1 = 1$
3 - Idempotência	$X + X = X$
4 - Complemento	$X + \overline{X} = 1$
5 - Involução	$\overline{\overline{X}} = X$
6 - Comutativa	$X + Y = Y + X$
7 - Associativa	$(X + Y) + Z = X + (Y + Z)$
8 - Distributiva	$X + (Y \cdot Z) = (X + Y)(X + Z)$
9 - Absorção 1	$X + X \cdot Y = X$
10 - Absorção 2	$X + \overline{X} \cdot Y = X + Y$
11 - Consenso	$X \cdot Y + \overline{X} \cdot Z + Y \cdot Z = X \cdot Y + \overline{X} \cdot Z$
12 - De Morgan	$\overline{X + Y} = \overline{X} \cdot \overline{Y}$

4.2 Versão AND (Produto Lógico)

Propriedade (AND)	**Expressão**
1 - Identidade	$X \cdot 1 = X$
2 - Elemento Nulo	$X \cdot 0 = 0$
3 - Idempotência	$X \cdot X = X$
4 - Complemento	$X \cdot \overline{X} = 0$
5 - Involução	$\overline{\overline{X}} = X$
6 - Comutativa	$X \cdot Y = Y \cdot X$
7 - Associativa	$(X \cdot Y) \cdot Z = X \cdot (Y \cdot Z)$
8 - Distributiva	$X \cdot (Y + Z) = X \cdot Y + X \cdot Z$
9 - Absorção 1	$X \cdot (X + Y) = X$
10 - Absorção 2	$X \cdot (\overline{X} + Y) = X \cdot Y$
11 - Consenso	$(X + Y)(\overline{X} + Z)(Y + Z) = (X + Y)(\overline{X} + Z)$
12 - De Morgan	$\overline{X \cdot Y} = \overline{X} + \overline{Y}$

As propriedades 2, 8, 9 e 11 serão demonstradas com tabelas verdades. As outras ficam a cargo do leitor demonstar.

5 Demonstração das Propriedades

5.1 Propriedade 2: Elemento Nulo

Versão OR: Essa propriedade diz que:

$$X + 1 = 1$$

Veja a tabela verdade a seguir que comprova a igualdade

X	$X + 1$
0	1
1	1

Assim, independente do valor lógico de X a função sempre 1.
Versão AND:

X	$X \cdot 0$
0	0
1	0

Dessa forma, fica perceptível que o valor lógico de X na função não altera o valor da função, que será sempre 0.

5.2 Propriedade 8: Distributiva

Versão OR:

X	Y	Z	$X + (Y \cdot Z)$	$(X + Y) \cdot (X + Z)$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	1	1
1	0	0	1	1
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

Versão AND:

X	Y	Z	$X \cdot (Y + Z)$	$(X \cdot Y) + (X \cdot Z)$
0	0	0	0	0
0	0	1	0	0
0	1	0	0	0
0	1	1	0	0
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

5.3 Propriedade 9: Absorção 1

Versão OR:

X	Y	$X + (X \cdot Y)$
0	0	0
0	1	0
1	0	1
1	1	1

Versão AND:

X	Y	$X \cdot (X + Y)$
0	0	0
0	1	0
1	0	1
1	1	1

5.4 Propriedade 11: Consensus

Versão OR:

X	Y	Z	$X \cdot Y + \overline{X} \cdot Z + Y \cdot Z$	$X \cdot Y + \overline{X} \cdot Z$
0	0	0	0	0
0	0	1	1	1
0	1	0	0	0
0	1	1	1	1
1	0	0	0	0
1	0	1	0	0
1	1	0	1	1
1	1	1	1	1

Versão AND:

X	Y	Z	$(X + Y)(\overline{X} + Z)(Y + Z)$	$(X + Y)(\overline{X} + Z)$
0	0	0	0	0
0	0	1	0	0
0	1	0	1	1
0	1	1	1	1
1	0	0	0	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

6 MINTERMOS E MAXTERMOS

Mintermos e maxternos são formas canônicas de representar funções booleanas Toda expressão booleanas pode ser obtida através da sua tabela verdade

> **Observação:** Dizemos que uma expressão está na *forma canônica* quando ela é representada por uma **soma de mintermos** (forma canônica disjuntiva) ou um **produto de maxtermos** (forma canônica conjuntiva). Em ambas, todas as variáveis da função aparecem em cada termo, direta ou negada.

6.1 Mintermos

Mintermi: é a forma canônica de representar uma linha da tabela verdade utilizando o operador AND (E lógico) entre todas as variáveis da função. Cada variável pode aparecer de duas formas: direta ou negada, dependendo do valor que ela assume na linha da tabela.

Notação:

- Valor 0: a variável aparece **negada** (exemplo: $A = 0 \Rightarrow \overline{A}$)
- Valor 1: a variável aparece **direta** (exemplo: $A = 1 \Rightarrow A$)

O mintermo da primeira linha da tabela verdade é o m_0 e o da útima será m_{2^n-1}, sendo n o número de variáveis da função. Para construir um mintermo, pega-se todas as variáveis da linha e usa-se a operação lógica **AND** entre elas, respeitando o uso da forma direta ou negada de acordo com os valores. Assim, cada mintermo representa exatamente uma linha onde a saída da função é igual a 1.

Notação – Forma Canônica por Mintermos

A notação $f(A,B,C) = \sum m(x,y,z)$ indica que a função f é expressa como uma **soma de mintermos**, ou seja, uma disjunção (OU) dos termos onde a saída é igual a 1 na tabela verdade. As letras "x", "y" e "z" indicam as **linhas** que a função é **1**.

6.2 Maxtermo

Maxtermo: é a forma canônica de representar uma linha da tabela verdade utilizando o operador OR (OU lógico) entre todas as variáveis da função. A contrução dos maxtermos é semlhante à dos mintemos, porém segue a lógica inversa:

Notação:

- Valor 0: a variável aparece **direta** (exemplo: $A = 0 \Rightarrow A$)
- Valor 1: a variável aparece **negada** (exemplo: $A = 1 \Rightarrow \overline{A}$)

Para cada linha da tabela verdade em que a saída é 0, pode-se formar um maxtermo. O maxtermo da primeira linha da tabela verdade é o M_0, e o último será M_{2^n-1}, com n sendo o número de variáveis.

Notação – Forma Canônica por Maxtermos

A notação $f(A,B,C) = \prod M(x,y,z)$ indica que a função f é expressa como uma **produto de maxtermos**, ou seja, uma conjunção (AND) dos termos onde a saída é igual a 0 na tabela verdade. As letras "x", "y" e "z" indicam as **linhas** que a função é **zero**.

Exemplo:

$$f = A + B$$

A	B	f	Mintermo	Maxtermo
0	0	0	$m_0 = \overline{A}\,\overline{B}$	$M_0 = A + B$
0	1	1	$m_1 = \overline{A}B$	$M_1 = A + \overline{B}$
1	0	1	$m_2 = A\overline{B}$	$M_2 = \overline{A} + B$
1	1	1	$m_3 = AB$	$M_3 = \overline{A} + \overline{B}$

- Os **mintermos** representam as linhas em que $f = 1$.
- Os **maxtermos** representam as linhas em que $f = 0$.
- A função $f = A + B$ tem como forma canônica:
  - **Soma de mintermos:** $f = m_1 + m_2 + m_3 = \overline{A}B + A\overline{B} + AB$
  - **Produto de maxtermos:** $f = M_0 = (A + B)$

Mintermos

Como cada mintermo é o complemento de seu maxtermo correspondente, podemos dizer que o complemento de uma função f é representado pela multiplicação (AND) dos maxtermos associados às linhas onde $f = 0$.
No exemplo acima, como $f = A + B$, temos que $f = 1$ nas linhas 1, 2 e 3, e $f = 0$ apenas na linha 0. Logo, o complemento de f é:

$$f' = M_0 = A + B \quad \Rightarrow \quad f = \overline{A + B}$$

Assim, mostramos que o maxtermo M_0 representa exatamente o valor de f' nesta função.

7 Circuito Combinacional

Circuitos combinacionais são aqueles que só dependem do estado atual da entrada, ou seja, não possuem **memória**. Um exemplo simples é um circuito onde a saída f é dada por $f = A \cdot B$, usando uma porta **AND** de duas entradas.

Esses circuitos incluem:

- Somadores
- Comparadores
- Codificadores
- Decodificadores
- Multiplexadores

8 Circuitos Sequenciais

Diferente dos circuitos combinacionais, os **circuitos sequenciais** dependem tanto das **entradas atuais** como do **estado anterior (memória)**.
Eles utilizam elementos de **armazenamento**, como flip-flops, para guardar o estado interno. Portanto, a saída é uma função das entradas e do estado armazenado.

São utilizados em:

- Regustradores
- Máquinas de Estados
- Contadores

Os circuitos sequenciais podem ser **síncronos** (controlados por um sinal de clock) ou **assíncronos** (controlados por variações nas entradas)