Arquitetura de Computadores

Aula 3 – Lógica Digital

Prof. André Roberto Guerra

Organização da Aula

Lógica Digital

- Álgebra de Boole
- Funções
- Operadores e operações
- Portas
- Propriedades

Contextualização

Álgebra de Boole

- A Álgebra de Boole é aplicável ao projeto dos circuitos lógicos e funciona baseada em princípios da lógica formal, uma área de estudo da filosofia

- Um dos pioneiros no estudo da lógica formal foi Aristóteles (384-322 a.C.), que publicou um tratado sobre o tema denominado "De Interpretatione"

- Boole percebeu que poderia estabelecer um conjunto de símbolos matemáticos para substituir certas afirmativas da lógica formal. Publicou suas conclusões em 1854 no trabalho “Uma Análise Matemática da Lógica”

- Claude B. Shannon mostrou (em sua tese no MIT) que o trabalho de Boole poderia ser utilizado para descrever a operação de sistemas de comutação telefônica. As observações de Shannon foram divulgadas em 1938 no trabalho "Uma Análise Simbólica de Relés e Circuitos de Comutação"

- Desenvolvida pelo matemático britânico George Simon Boole (1815 – 1864) para estudo da lógica
- Definida sobre um conjunto de dois elementos:
  - (0, 1) (baixo, alto)
  - (falso, verdadeiro)

- Seus elementos, a princípio, não tem significado numérico

- Postulados: se x é uma variável booleana então:

  - Se $x \neq 0 \Rightarrow x = 1$

  - Se $x \neq 1 \Rightarrow x = 0$

Instrumentalização

Lógica Digital – Funções

- Uma variável booleana só pode assumir apenas um dos valores possíveis (0 e 1)
- Uma ou mais variáveis e operadores podem ser combinados formando uma função lógica

- $Z_1(A) = f(A) = \ldots$
  - ✓(expressão usando var. A)

- $Z_2(A,B) = f(A,B) = \ldots$
  - ✓(expressão usando var. A e B)

- Resultados de uma função lógica podem ser expressos numa tabela relacionando todas as combinações possíveis dos valores que suas variáveis podem assumir e seus resultados correspondentes:
  - a Tabela-Verdade

A Tabela Verdade

Variáveis

Função Lógica

Lista das combinações possíveis dos estados das variáveis de entrada.

A	B	Z=f(A,B)
0	0	0
0	1	1
1	0	1
1	1	1

Resultados da função lógica para cada combinação dos estados de entrada.

- Tabela Verdade relaciona os resultados (saída) de uma função lógica para todas as combinações possíveis de suas variáveis (entrada)
- Na Tabela Verdade apresentada a função lógica Z possui duas variáveis A e B, sendo
  $Z = f(A, B) = A + B$

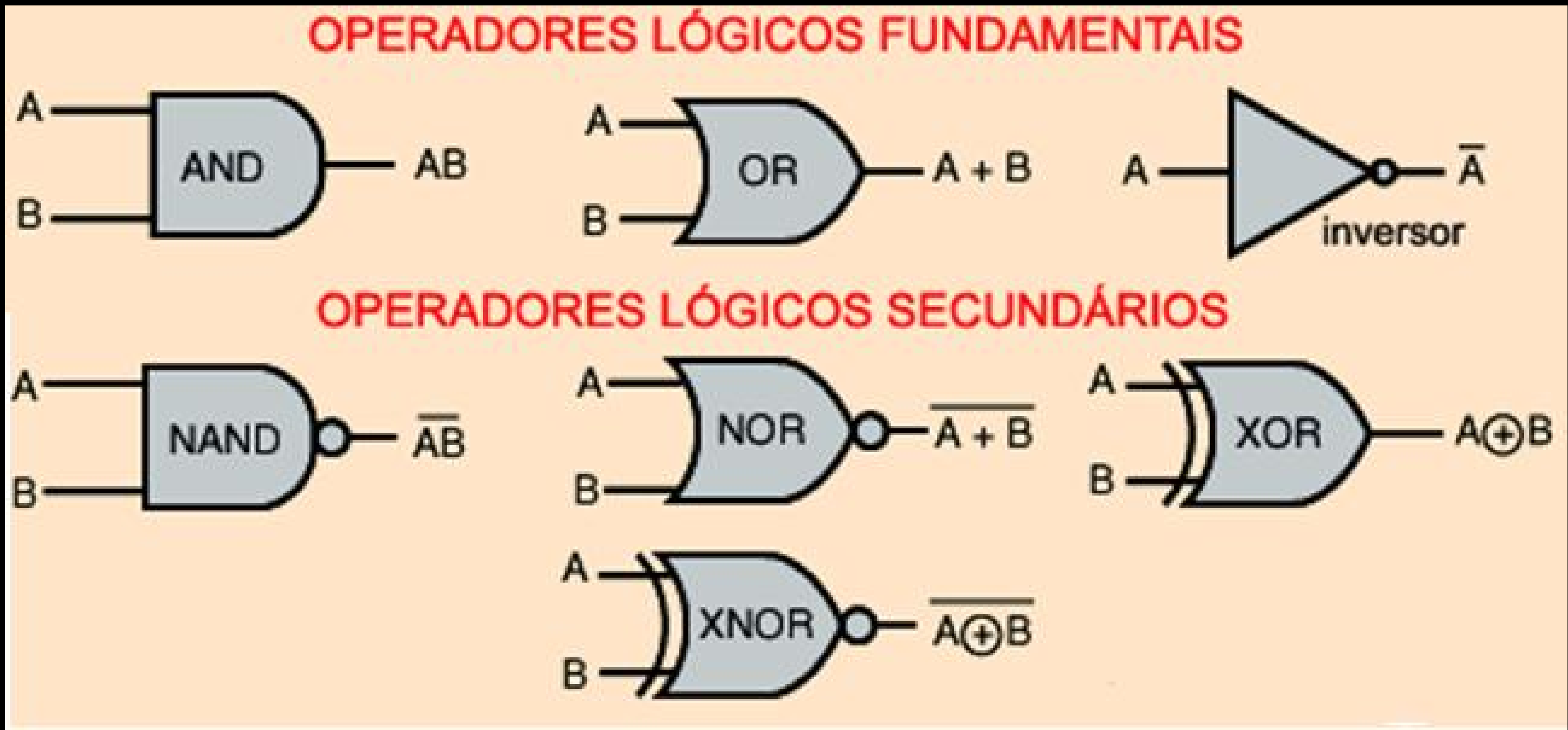
Operações e Operadores

- São definidas algumas operações elementares (básicas) na álgebra booleana:

  - Operação “Não” (NOT)
  - Operação “E” (AND)
  - Operação “Ou” (OR)

- Definidas também algumas operações complementares na álgebra booleana:
  - Operação NAND
  - Operação NOR
  - Operação “Ou-Exclusivo” (Exclusive-Or ou XOR)
  - Operação XNOR

- As variáveis booleanas são representadas por letras maiúsculas, A, B, C,... e as funções pela notação f(A,B,C,D,...)



Portas Lógicas

- Precedência das Operações

  1. ( ) - “Parêntesis”
  2. ‘ _ - “Negação”
  3. . - “E”
  4. \+ - “OU”, “OU Exclusivo”,…

- O uso de parêntesis altera a precedência “normal” dos operadores, como na álgebra comum

Portas Lógicas Fundamentais

Porta Lógica NOT

- É a porta Inversora
- Operador: Barra, Apóstrofe $\overline{A}$, A'

Símbolo

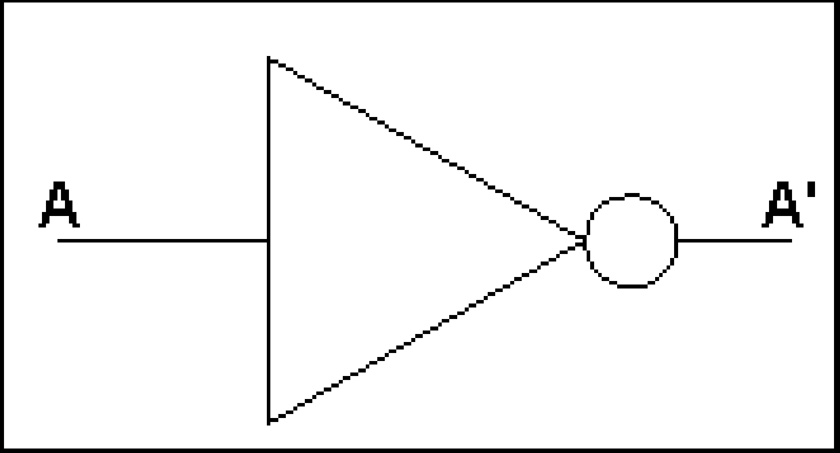


Tabela Verdade

A	F = A'
0	1
1	0

Porta Lógica AND

- Requer duas ou mais entradas
- Operador: (F = A . B)

Símbolo

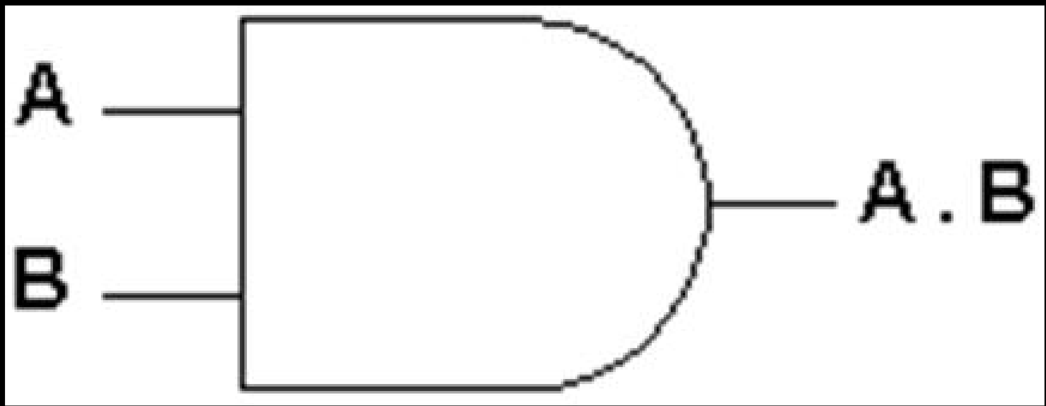


Tabela Verdade

A	B	F = (A.B)
0	0	0
0	1	0
1	0	0
1	1	1

Porta Lógica OR

- Requer duas ou mais entradas
- Operador: + (F = A + B)

Símbolo

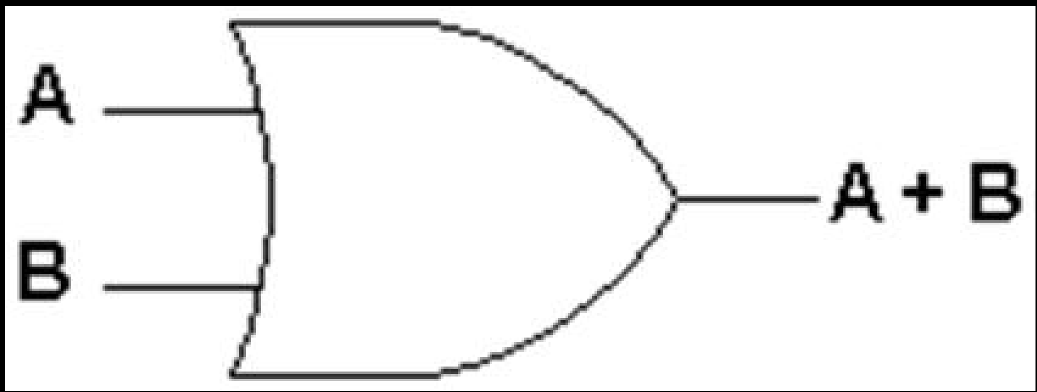


Tabela Verdade

A	B	F = (A+B)
0	0	0
0	1	1
1	0	1
1	1	1

Porta Lógica NAND

- Equivalente a porta AND seguida de uma NOT
- Operador: . (F = A . B) ‘

Símbolo

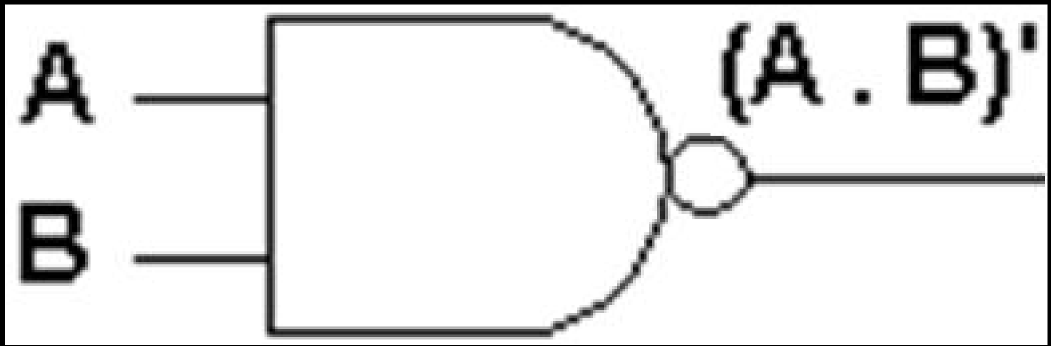


Tabela Verdade

A	B	F = (A.B)‘
0	0	1
0	1	1
1	0	1
1	1	0

Porta Lógica NOR

- Equivalente a porta OR seguida de uma NOT
- Operador: (F = A + B) ‘

Símbolo

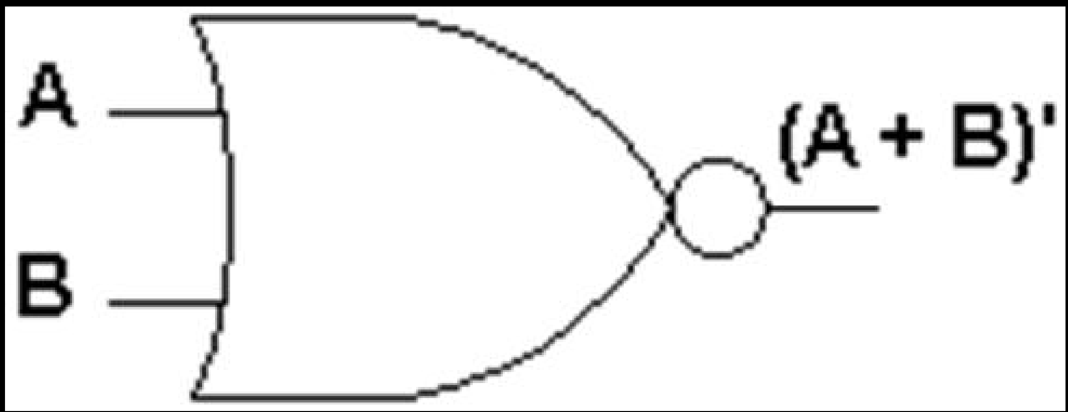


Tabela Verdade

A	B	F = (A+B)‘
0	0	1
0	1	0
1	0	0
1	1	0

Porta Lógica XOR

- É o OR (OU) Exclusivo
- Operador: (F = A ⊕ B)

Símbolo

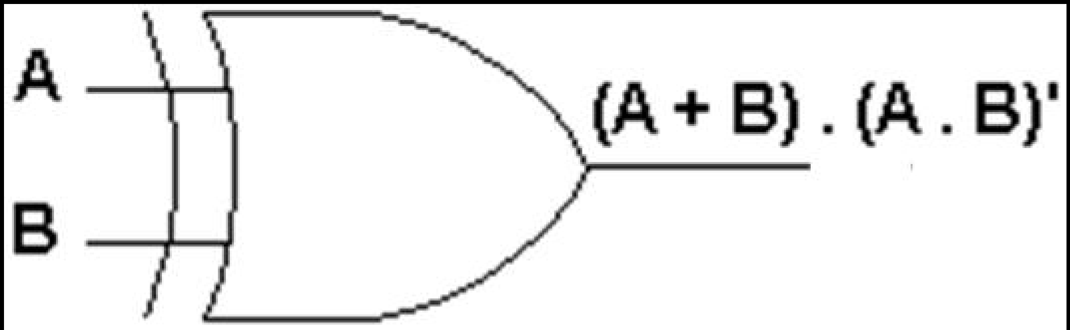


Tabela Verdade

A	B	F = (A⊕B)
0	0	0
0	1	1
1	0	1
1	1	0

Porta Lógica XNOR

- É o complemento da porta XOR
- Operador: (F = A ⊕ B) ‘

Símbolo

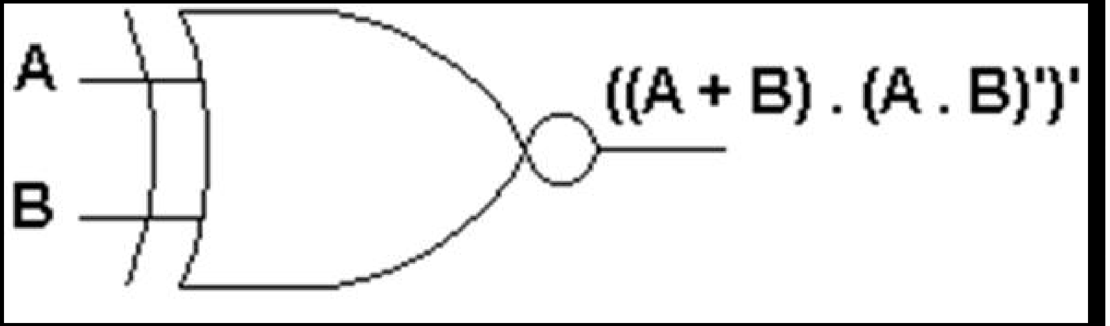


Tabela Verdade

A	B	F = (A⊕B)‘
0	0	1
0	1	0
1	0	0
1	1	1

Lógica Digital - Propriedades

- Sendo A, B e C variáveis booleanas
  - Propriedade comutativa
    - ✓ A . B = B . A
    - ✓ A + B = B + A
    - ✓ A ă B = B ă A
  - Propriedade distributiva
    - ✓ A . (B + C) = A . B + A . C
    - ✓ A + B . C = (A + B) . (A + C)

- Propriedade Associativa
  - ✓ (A . B) . C = A . (B . C) =
  - ✓ A . B . C
  - ✓ (A + B) + C = A + (B + C) = A + B + C
  - ✓ (A ă B) ă C = A ă (B ă C) =
  - ✓ A ă B ă C

- Propriedades (Leis) de Absorção

  - ✓ $A + A.B = A$
  - ✓ $A + A'.B = A + B$
  - ✓ $(A + B').B = A.B$

- Identidades importantes
  - ✓ $A.B + A.B' = A$
  - ✓ $(A + B) . (A + B') = A$
  - ✓ $A.(A + B) = A$
  - ✓ $A.(A' + B) = AB$
  - ✓ $A.B + A'.C = (A + C) . (A' + B)$

Identidades

NOT	AND	OR
$0' = 1$	$A \cdot 1 = A$	$A + 1 = 1$
$1' = 0$	$A \cdot 0 = 0$	$A + 0 = A$
$(A')' = A$	$A \cdot A = A$	$A + A = A$
	$A \cdot A' = 0$	$A + A' = 1$

Dualidades

- Princípio Especial (álgebra booleana)
  - Para uma equação booleana qualquer, ao trocar as operações E (.) e operações OU (+) entre si assim como valores 0 e 1 entre si, obtém-se uma equação igualmente válida

Dualidades

$$A + 0 = A \qquad A \cdot 1 = A$$

$$A + 1 = 1 \qquad A \cdot 0 = 0$$

$$A + A = A \qquad A \cdot A = A$$

$$A + \overline{A} = 1 \qquad A \cdot \overline{A} = 0$$

Equivalência de operações

- Qualquer função lógica pode ser expressa em termos das operações AND, OR e NOT

$$A \oplus E = \bar{A}E + A\bar{E}$$

Síntese

Lógica Digital

- Álgebra de Boole
- Funções
- Operadores e operações
- Portas
- Propriedades

- Computadores são construídos com base em chips de circuito integrado que contêm minúsculos elementos comutadores denominados portas

- As portas mais comuns são AND, OR, NAND, NOR, e NOT. Circuitos Simples podem ser montados ao se combinar diretamente portas individuais

- Circuitos mais complexos são multiplexadores, demultiplexadores, codificadores, decodificadores, deslocadores e ULA

- As leis da álgebra booleana podem ser usadas para transformar circuitos de uma forma para outra. Em muitos casos é possível produzir circuitos mais econômicos dessa maneira

Referências de Apoio

- TANENBAUM, A. S. Organização Estruturada de Computadores. 6. ed. São Paulo: Prentice-Hall, 2013.