Introdução à Álgebra Booleana (parte 1)

Alex Dias Gonsales – Notas de aula (23/05/2020)

A álgebra booleana recebe esse nome em homenagem ao matemático e filósofo britânico George Boole (1815 a 1864), que desenvolveu as bases para a criação dessa teoria, através das publicações "***The Mathematical Analysis of Logic***" (Análise Matemática da Lógica - 1847) e "***The Laws of Thought***" (As Leis do Pensamento – 1854).

Será resumida pelos axiomas mostrados a seguir.

> Um **axioma** é uma sentença ou proposição que não é provada ou demonstrada e é considerada como óbvia ou como um consenso inicial necessário para a construção ou aceitação de uma teoria. Por essa razão, é aceito como verdade e serve como ponto inicial para dedução e inferências de outras verdades (dependentes de teoria). Em muitos contextos, **axioma**, **postulado** e **hipótese** são usados como sinônimos.

Axiomas da Álgebra Booleana

- Dois valores
- Três operadores primitivos

Dois Valores

Existem apenas dois valores possíveis na álgebra booleana. Normalmente são usados os valores FALSO e VERDADEIRO. Não existe nada diferente desses dois valores. Uma variável só pode (e deve) assumir um desses dois valores. A álgebra também pode ser flexibilizada e utilizar outros valores além do FALSO e VERDADEIRO. Mas uma vez escolhido os dois valores a serem utilizados, eles serão os únicos valores que poderão ser utilizados.

Exemplos de valores utilizados na álgebra booleana:
- Falso / Verdadeiro
- Não / Sim
- Desligado / Ligado
- 0V / 5V
- 0 / 1

Neste documento serão utilizados os valores 0 (zero) e 1 (um).

Operadores Primitivos

A álgebra booleana pode ser definida com apenas 3 operadores (E, OU e NÃO), chamados de operadores primitivos, os quais são descritos a seguir:

Operador E

Também conhecido como operador de **conjunção** ou **multiplicação lógica**. Diversos símbolos são utilizados para representar esse operador, conforme o contexto em que se estiver trabalhando (. & ∧ ∩). Neste documento será utilizado o ponto (.). Esse operador é aplicado sobre dois operandos (valores). O resultado será VERDADEIRO (UM) somente se os dois operandos forem verdadeiros.

Definição do operador E:

```
0 . 0 = 0
0 . 1 = 0
1 . 0 = 0
1 . 1 = 1
```

Operador OU

Também conhecido como operador de **disjunção** ou **soma lógica**. Diversos símbolos são utilizados para representar esse operador, conforme o contexto em que se estiver trabalhando (+ | ∨ ∪). Neste documento será utilizado o sinal de soma (+). Esse operador é aplicado sobre dois operandos (valores). O resultado será VERDADEIRO (UM) se pelo menos um dos operandos for verdadeiro. Observe que apesar da operação ser chamada de soma lógica, ela não efetua uma soma aritmética binária.

Definição do operador OU:

```
0 + 0 = 0
0 + 1 = 1
1 + 0 = 1
1 + 1 = 1
```

Operador NÃO

Também conhecida como operação de **complementação** ou **negação**. Diversos símbolos são utilizados para representar esse operador, conforme o contexto em que se estiver trabalhando (¯ ~ ! ¬ ʻ). Neste documento será utilizada a barra horizontal superior (¯). Esse operador é aplicado sobre apenas um operando (valor). O resultado é a inversão do operando, ou seja, se o operando for VERDADEIRO então a operação resulta em FALSO e se o valor do operando for FALSO então a operação resulta em VERDADEIRO.

Definição do operador NÃO:

$\overline{0} = 1$

$\overline{1} = 0$

Tabela-Verdade

É uma tabela que relaciona todas as possíveis combinações dos operandos e o resultado da operação. Através de uma tabela-verdade conseguimos uma visualização do funcionamento (resultados) do operador. A baixo podemos ver as tabelas-verdade para os operadores E, OU, NÃO.

Operador E (.)		
A	B	A . B
0	0	0
0	1	0
1	0	0
1	1	1

Operador OU (+)		
A	B	A + B
0	0	0
0	1	1
1	0	1
1	1	1

Operador NÃO (‾)	
A	$\bar{A}$
0	1
1	0

Os operadores E e OU podem ser estendidos para 3 operandos (valores) ou mais. Abaixo vemos as tabelas-verdade para os operadores E e OU com 3 operandos:

Tabela Verdade
(Operadores primitivos estendidos para 3 variáveis)

Operador E (.)			
A	B	C	A . B . C
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

Operador OU (+)			
A	B	C	A + B + C
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	1
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Expressões e Equações Lógicas

Expressão Lógica

É uma sequência de duas ou mais operações lógicas. Exemplo:

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

Equação Lógica

É uma relação de igualdade entre expressões lógicas. Exemplo: relação de igualdade entre as expressões Y e $A + B \cdot \overline{C}$

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

Resolução de Equações e Expressões Lógicas

As operações lógicas devem ser realizadas respeitando uma determinada ordem (sequência), seguindo a regra de precedência dos operadores, conforme mostrado a seguir:

Precedência dos Operadores Lógicos

1º Parênteses (mais internos)
2º Operação NÃO
3º Operador E
4º Operador OU

Ou seja, para resolver uma expressão lógica devemos resolver primeiro os parênteses, depois o operador NÃO, depois o operador E e por último o operador OU.

Exemplo:
Resolver a equação $Y = A + B \cdot \overline{C}$ quando A=0, B=0 e C=0.
Antes de iniciar a resolução, devemos substituir os valores dos operandos na equação:

$Y = 0 + 0 \cdot \overline{0}$ (resolver o operador NÃO ¯)
$Y = 0 + 0 \cdot 1$ (resolver o operador E .)
$Y = 0 + 0$ (resolver o operador OU +)
$Y = 0$

Vamos agora resolver a equação para todos os demais 7 casos possíveis:

Resolução quando A=0, B=0 e C=1:

$Y = 0 + 0 . \overline{1}$ (resolver o operador NÃO ¯)
$Y = 0 + 0 . 0$ (resolver o operador E .)
$Y = 0 + 0$ (resolver o operador OU +)
$Y = 0$

Resolução quando A=0, B=1 e C=0:

$Y = 0 + 1 . \overline{0}$ (resolver o operador NÃO ¯)
$Y = 0 + 1 . 1$ (resolver o operador E .)
$Y = 0 + 1$ (resolver o operador OU +)
$Y = 1$

Resolução quando A=0, B=1 e C=1:

$Y = 0 + 1 . \overline{1}$ (resolver o operador NÃO ¯)
$Y = 0 + 1 . 0$ (resolver o operador E .)
$Y = 0 + 0$ (resolver o operador OU +)
$Y = 0$

Resolução quando A=1, B=0 e C=0:

$Y = 1 + 0 . \overline{0}$ (resolver o operador NÃO ¯)
$Y = 1 + 0 . 1$ (resolver o operador E .)
$Y = 1 + 0$ (resolver o operador OU +)
$Y = 0$

Resolução quando A=1, B=0 e C=1:

$Y = 1 + 0 . \overline{1}$ (resolver o operador NÃO ¯)
$Y = 1 + 0 . 0$ (resolver o operador E .)
$Y = 1 + 0$ (resolver o operador OU +)
$Y = 1$

Resolução quando A=1, B=1 e C=0:

$Y = 1 + 1 . \overline{0}$ (resolver o operador NÃO ¯)
$Y = 1 + 1 . 1$ (resolver o operador E .)
$Y = 1 + 1$ (resolver o operador OU +)
$Y = 1$

Resolução quando A=1, B=1 e C=1:

$Y = 1 + 1 . \overline{1}$ (resolver o operador NÃO ¯)
$Y = 1 + 1 . 0$ (resolver o operador E .)
$Y = 1 + 0$ (resolver o operador OU +)
$Y = 1$

Resumindo:

A=0;B=0;C=0	A=0;B=0;C=1	A=0;B=1;C=0	A=0;B=1;C=1
0+0.0'	0+0.1'	0+1.0'	0+1.1'
0+0.1	0+0.0	0+1.1	0+1.0
0+0	0+0	0+1	0+0
0	0	1	0

A=1;B=0;C=0	A=1;B=0;C=1	A=1;B=1;C=0	A=1;B=1;C=1
1+0.0'	1+0.1'	1+1.0'	1+1.1'
1+0.1	1+0.0	1+1.1	1+1.0
1+0	1+0	1+1	1+0
1	1	1	1

Agora podemos montar a tabela-verdade da equação $Y = A + B \cdot \overline{C}$

A	B	C	Y
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

Podemos utilizar outra estratégia para montar a tabela-verdade de uma equação, conforme mostrado a seguir para a equação $Y = A + B \cdot \overline{C}$

Montar a tabela verdade com colunas para calcular as expressões parciais, conforme a precedência dos operadores:

A	B	C	C'	B.C'	A+B.C'	Y
0	0	0				
0	0	1				
0	1	0				
0	1	1				
1	0	0				
1	0	1				
1	1	0				
1	1	1				

Resolver a primeira parcial:

A	B	C	C'	B.C'	A+B.C'	Y
0	0	0	1			
0	0	1	0			
0	1	0	1			
0	1	1	0			
1	0	0	1			
1	0	1	0			
1	1	0	1			
1	1	1	0			

Resolver a segunda parcial:

A	B	C	C'	B.C'	A+B.C'	Y
0	0	0	1	0		
0	0	1	0	0		
0	1	0	1	1		
0	1	1	0	0		
1	0	0	1	0		
1	0	1	0	0		
1	1	0	1	1		
1	1	1	0	0		

Resolver a terceira parcial:

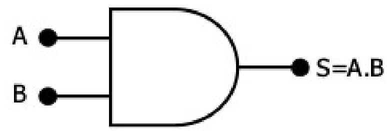
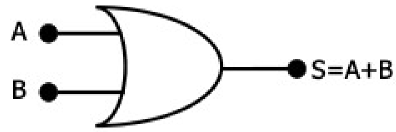
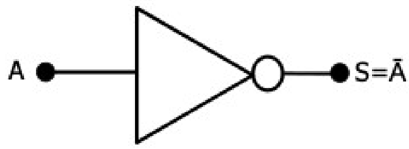
A	B	C	C'	B.C'	A+B.C'	Y
0	0	0	1	0	0	
0	0	1	0	0	0	
0	1	0	1	1	1	
0	1	1	0	0	0	
1	0	0	1	0	1	
1	0	1	0	0	1	
1	1	0	1	1	1	
1	1	1	0	0	1	

Portanto, os valores da coluna Y são os valores da última parcial resolvida:

A	B	C	C'	B.C'	A+B.C'	Y
0	0	0	1	0	0	0
0	0	1	0	0	0	0
0	1	0	1	1	1	1
0	1	1	0	0	0	0
1	0	0	1	0	1	1
1	0	1	0	0	1	1
1	1	0	1	1	1	1
1	1	1	0	0	1	1

Portas Lógicas

Portas lógicas são dispositivos eletrônicos que implementam as operações booleanas. As portas lógicas primitivas (que implementam as funções lógicas primitivas) são representadas graficamente pelos símbolos abaixo:

Operador Lógico	Expressão	Porta Lógica
E	S = A . B	
OU	S = A + B	
NÃO	S = $\overline{A}$	

Circuitos Lógicos

É o agrupamento de portas lógicas de forma a implementar funções lógicas (expressões) mais complexas. Exemplo: circuito lógico que implementa a função $Y = A + B \ . \ \overline{C}$