Álgebra Booleana
&
Portas Lógicas

Portas Lógicas: Circuitos Digitais

- Circuitos digitais podem ser construídos com pequeno número de elementos primitivos.
- Circuito digital é um componente que somente assume dois valores lógicos: 0 e 1.
- Circuito eletrônicos são pequenos dispositivos chamados de Portas Lógicas (Gates)

Portas Lógicas: Circuitos Digitais

- Representação de Transistores: Portas Lógicas

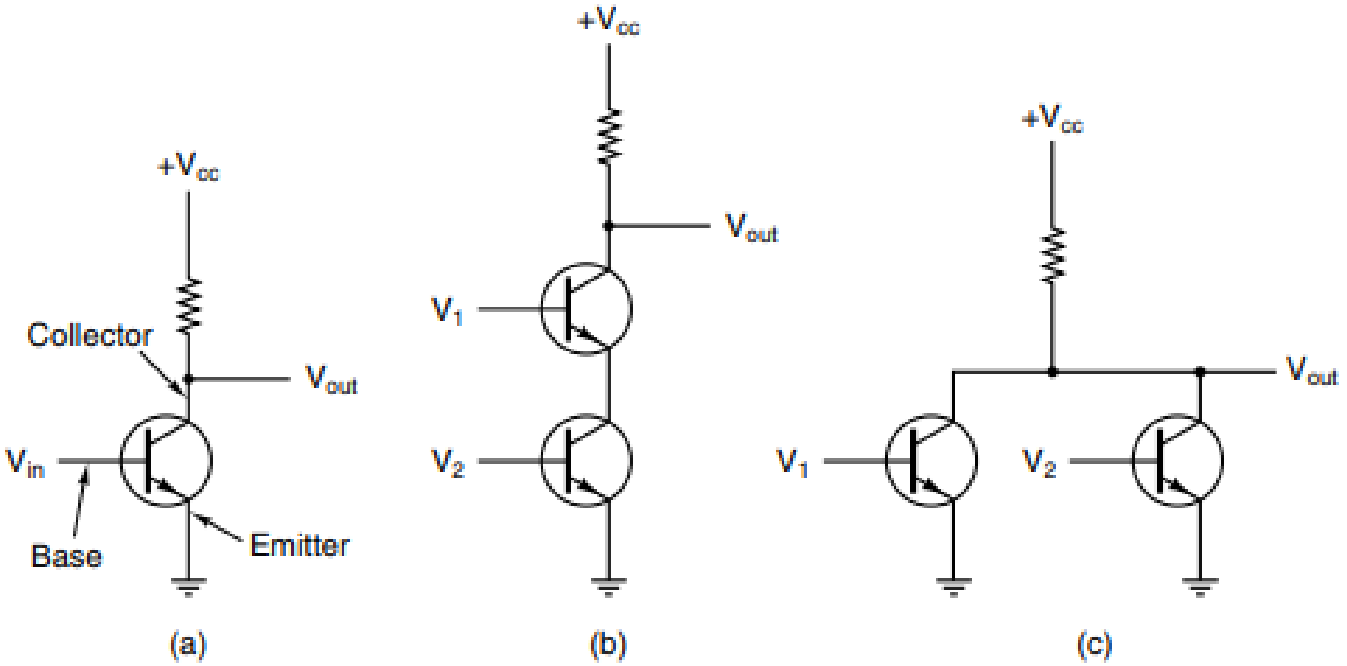


Figure 3-1. (a) A transistor inverter. (b) A NAND gate. (c) A NOR gate.

Portas Lógicas: Circuitos Digitais

- Esses três circuitos, ou seus equivalentes, formam as três portas mais simples. Elas são chamadas de portas NOT, NAND e NOR, respectivamente.

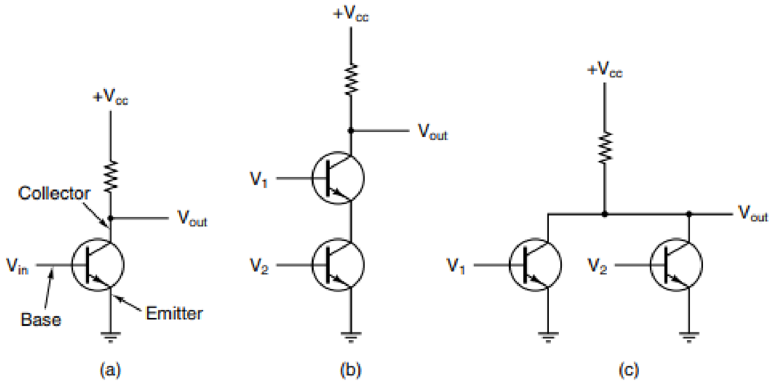


Figure 3-1. (a) A transistor inverter. (b) A NAND gate. (c) A NOR gate.

- As portas NOT são frequentemente chamadas de inversores.
- Se agora adotarmos a convenção de que "alto'' (Vcc volts) é um 1 lógico, e que "baixo'' (terra) é um 0 lógico, podemos expressar o valor de saída como uma função dos valores de entrada.

Portas Lógicas: Circuitos Digitais

- Representação de Transistores: Portas Lógicas

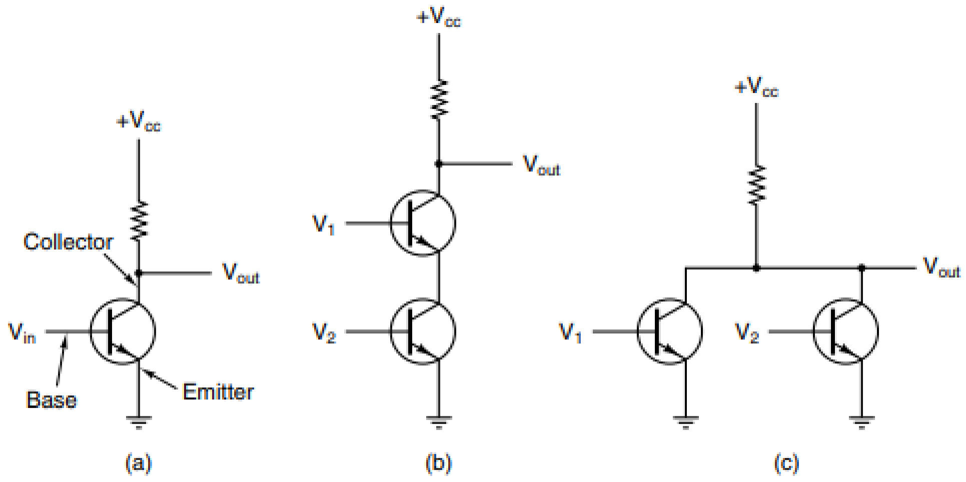
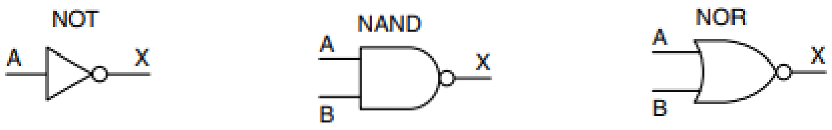


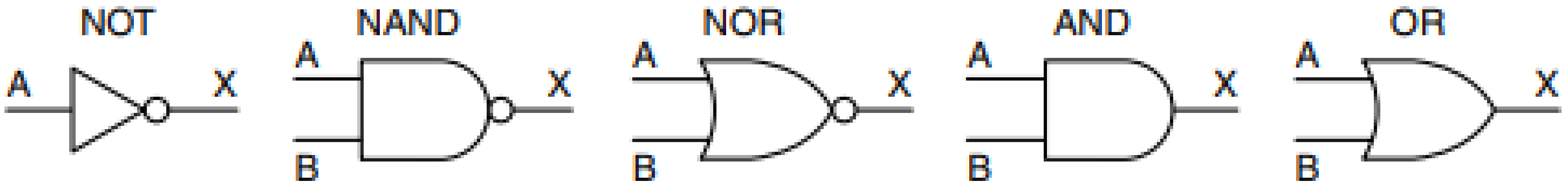
Figure 3-1. (a) A transistor inverter. (b) A NAND gate. (c) A NOR gate.



Representação adotada pela maioria dos autores

Portas Lógicas: Circuitos Digitais

- Os símbolos usados para representar cinco portas abaixo, juntamente com o **comportamento funcional** (tabela-verdade) de cada circuito.



A	X
0	1
1	0

(a)

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

(b)

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

(c)

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

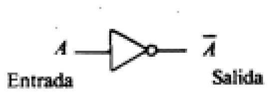
(d)

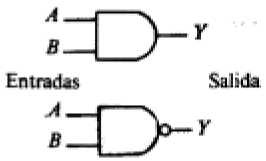
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

(e)

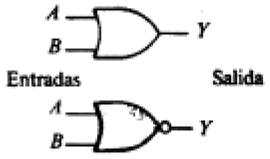
- Nessas figuras, **A e B são entradas** e **X é a saída**. Cada linha especifica a saída para uma combinação diferente das entradas.

Portas Lógicas: 7 funções lógicas

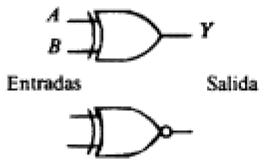
<table>
<tr><th>Función lógica</th><th>Símbolo lógico de puerta</th><th>Tabla de verdad</th><th>Expresión booleana</th></tr>
<tr><td>Inversor</td><td>
</td><td><table><tr><th>Entrada</th><th>Salida</th></tr><tr><td>A</td><td>$\overline{A}$</td></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table></td><td>$A = \overline{A}$</td></tr>
<tr><td>AND

NAND</td><td>
</td><td><table><tr><th colspan="2">Entradas</th><th colspan="2">Salidas</th></tr><tr><td>B</td><td>A</td><td>AND</td><td>NAND</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr></table></td><td>$A \cdot B = Y$

$\overline{A \cdot B} = Y$</td></tr>
<tr><td>OR

NOR</td><td>
</td><td><table><tr><th colspan="2">Entradas</th><th colspan="2">Salidas</th></tr><tr><td>B</td><td>A</td><td>OR</td><td>NOR</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>0</td></tr></table></td><td>$A + B = Y$

$\overline{A + B} = Y$</td></tr>
<tr><td>OR exclusiva

NOR exclusiva</td><td>
</td><td><table><tr><th colspan="2">Entradas</th><th colspan="2">Salidas</th></tr><tr><td>B</td><td>A</td><td>XOR</td><td>XNOR</td></tr><tr><td>0</td><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td><td>1</td></tr></table></td><td>$A \oplus B = Y$

$\overline{A \oplus B} = Y$</td></tr>
</table>

Portas Lógicas: Tecnologias

- No período anterior a 2013, as duas principais tecnologias CIs (Circuitos Integrados) são:
- a)Portas Tipos Bipolares: TTL (*Transistor-Transistor Logic*) e ECL (*Emitter-Coupled Logic- alta velocidade*)
- b)Portas MOS (*Metal Oxide Semiconductor*): MOS são **mais lentas** do que TTL e ECL. MOS foram mais adotadas porque requerem **menos energia e ocupam menos espaço**. O MOS tem variedades (PMOS, NMOS e CMOS).

Álgebra Booleana

- Para descrever os CIs que podem ser construídos combinando portas, um novo tipo de álgebra é necessário, um em que variáveis e funções podem assumir apenas os valores 0 e 1.
- Tal álgebra é chamada de álgebra booleana, em homenagem ao seu descobridor, o matemático inglês George Boole (1815–1864).

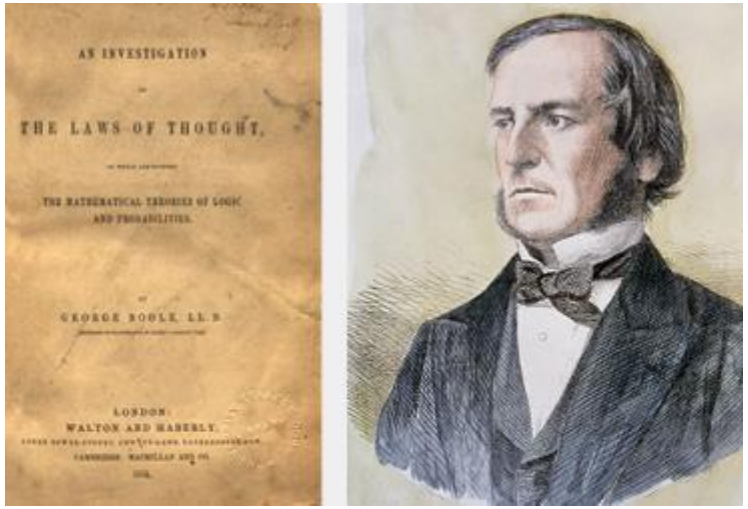
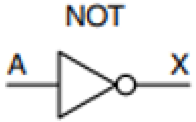


Imagem de: https://georgeboole.com/media/central-media/it-mag-2015/LoT-GB200-news-story-mag.jpg

Álgebra Booleana

- Álgebra Booleana(álgebra de comutação) é um tipo específico de álgebra
- Uma função booleana tem **uma ou mais variáveis de entrada** e **produz um único resultado.**
- Uma função simples, f, pode ser definida dizendo que f (A) é 1 se A é 0 e f (A) é 0 se A é 1. Esta função é a função NOT da abaixo:



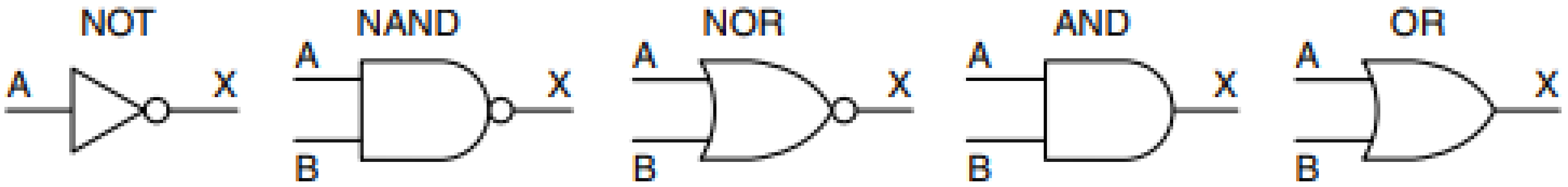
$x = f(A)$, *se A = 0 então x = 1; se A = 1 então x = 0*

A	X
0	1
1	0

Representação com Álgebra Booleana

Álgebra Booleana

- Como uma **função booleana** de n variáveis tem 2^n combinações possíveis de valores de entrada.
- A função é descrita como uma tabela com 2^n linhas, cada linha informando o valor da função para uma combinação diferente de valores de entrada.
- Tal tabela é chamada de **tabela verdade**. As tabelas abaixo são exemplos de tabelas verdade.



2^1

A	X
0	1
1	0

(a)

2^2

A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

(b)

2^2

A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

(c)

2^2

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

(d)

2^2

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

(e)

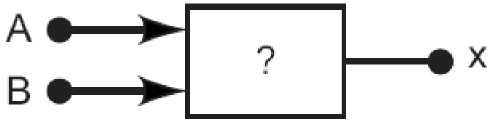
Álgebra Booleana

- Exemplo de tabela verdade com: duas, três e quatro entradas.

Entradas → A, B; Saída → x

A	B	x
0	0	1
0	1	0
1	0	1
1	1	0

Tabela duas entradas: 2^2



(a)

A	B	C	x
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

(b)

Tabela três entradas: 2^3

A	B	C	D	x
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

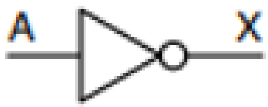
(c)

Tabela de quatro entradas: 2^4

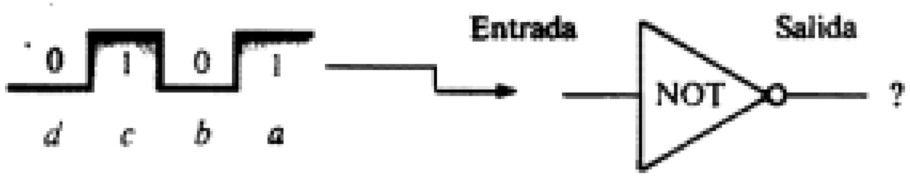
2^2

Álgebra Booleana

- 1) Quais são as saídas do inversor (às vezes chamado de porta NOT) na abaixo com o pulso (onda) na entrada?

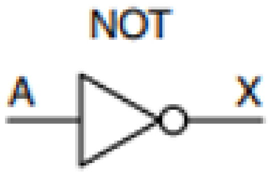


A	X
0	1
1	0

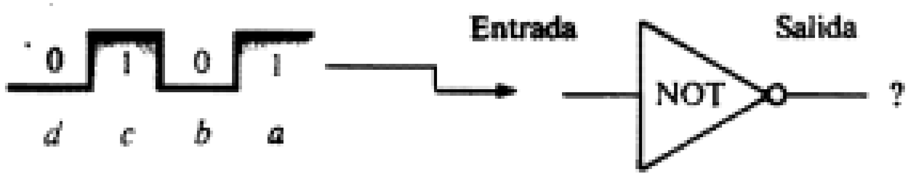


Álgebra Booleana

- 1) Quais são as saídas do inversor (às vezes chamado de porta NOT) na abaixo com o pulso (onda) na entrada?



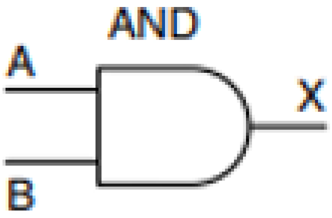
A	X
0	1
1	0



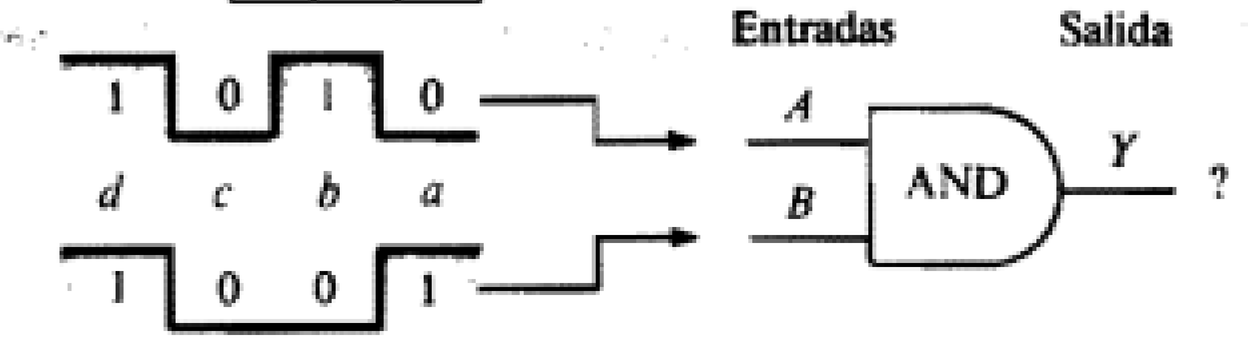
Resposta da Saída: 1010

Álgebra Booleana

- 2) Quais são as saídas da função AND com a onda digital de entrada?

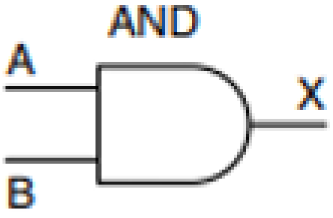


A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

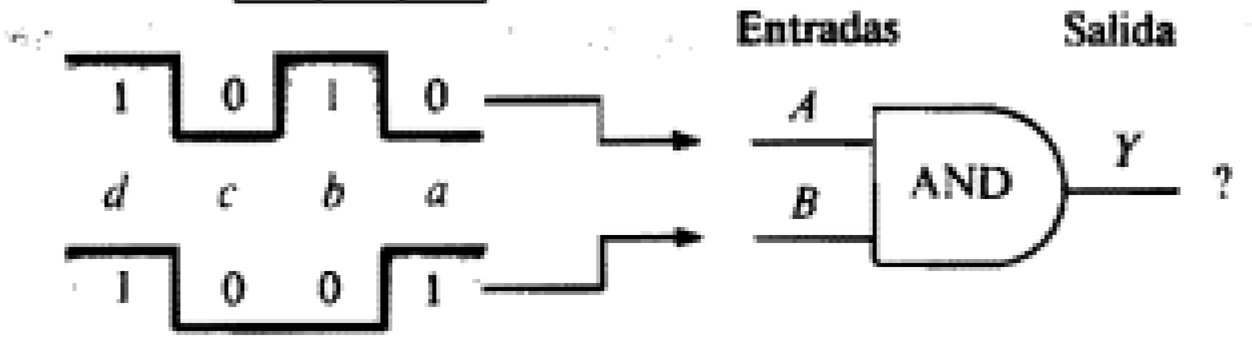


Álgebra Booleana

- 2) Quais são as saídas da função AND com a onda digital de entrada?



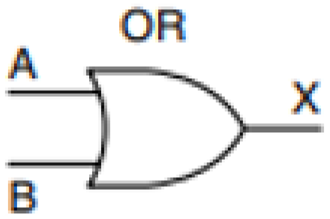
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1



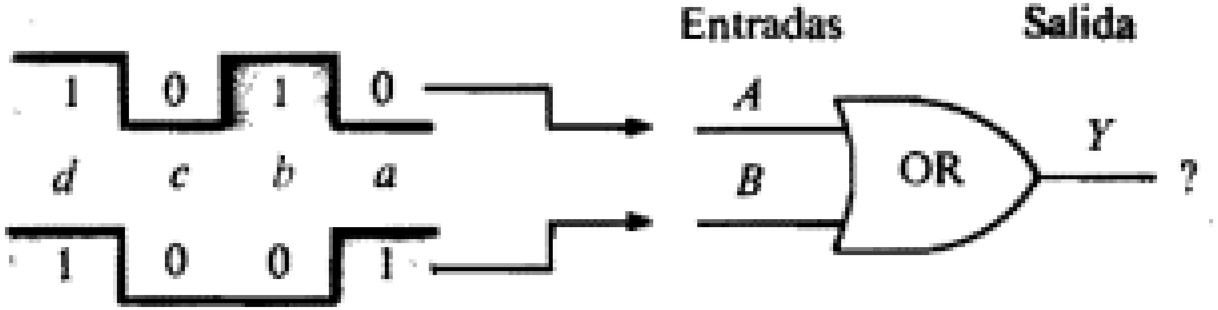
Resposta da Saída: 1000

Álgebra Booleana

- 3) Quais são as saídas da função OR com a onda digital de entrada?

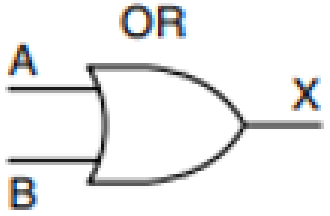


A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

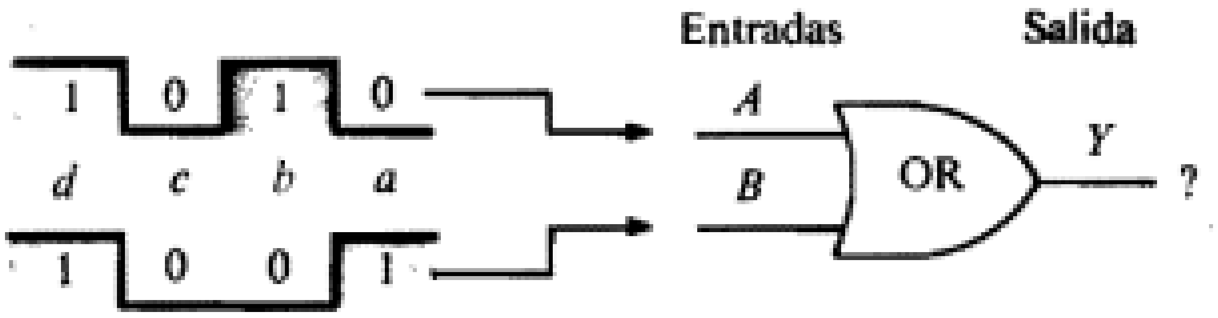


Álgebra Booleana

- 3) Quais são as saídas da função OR com a onda digital de entrada?



A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

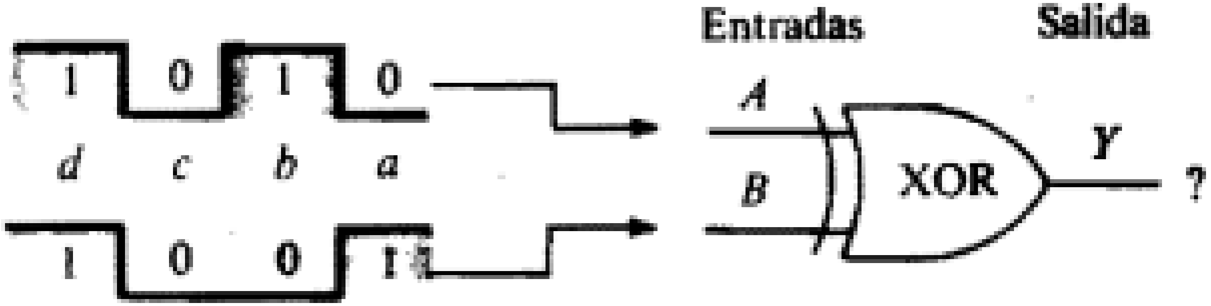


Resposta da Saída: 1011

Álgebra Booleana

- 4) Quais são as saídas da função OR com a onda digital de entrada?

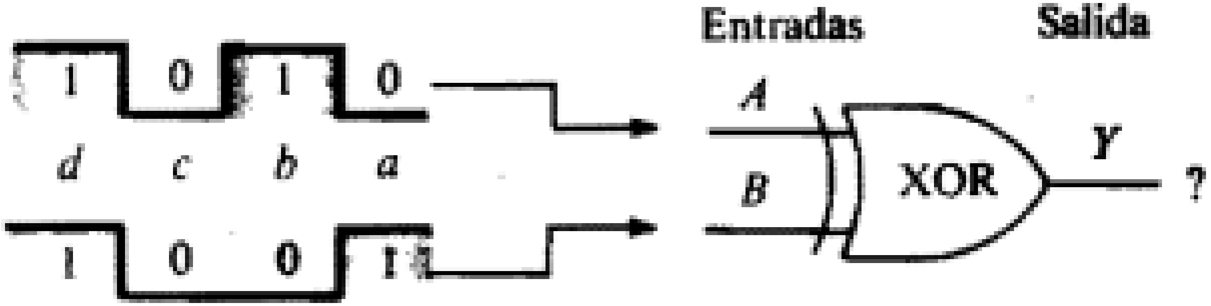
Entradas		Salidas	
B	*A*	XOR	XNOR
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1



Álgebra Booleana

- 4) Quais são as saídas da função OR com a onda digital de entrada?

Entradas		Salidas	
B	*A*	XOR	XNOR
0	0	0	1
0	1	1	0
1	0	1	0
1	1	0	1



Resposta da Saída: 0011

Converter: TABELA VERDADE em FUNÇÃO BOOLEANA

- a) **Saída 1:** a **função booleana** é em linhas com combinações de variáveis de entrada que dão um valor de saída de 1.
- b) Multiplicação implícita ou um ponto para significar a função booleana AND
- c) + significar a função booleana OR
- d) Por convenção, colocaremos uma barra sobre uma variável de entrada para indicar que seu valor é invertido.
- e) A função booleana, M, é verdadeira (i.e., 1) se qualquer uma dessas quatro condições (mostradas abaixo) forem verdadeiras. $M = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$

A	B	C	M	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\bar{A}BC$
1	0	0	0	
1	0	1	1	$A\bar{B}C$
1	1	0	1	$AB\bar{C}$
1	1	1	1	ABC

$$M = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

Converter: TABELA VERDADE em FUNÇÃO BOOLEANA

- a) **Saída 1:** a **função booleana** é em linhas com combinações de variáveis de entrada que dão um valor de saída de 1.
- b) Multiplicação implícita ou um ponto para significar a função booleana AND
- c) + significar a função booleana OR
- d) Por convenção, colocaremos uma barra sobre uma variável de entrada para indicar que seu valor é invertido.
- e) A função booleana, M, é verdadeira (i.e., 1) se qualquer uma dessas quatro condições (mostradas abaixo) forem verdadeiras.

A	B	C	M	
0	0	0	0	
0	0	1	0	
0	1	0	0	
0	1	1	1	$\bar{A}BC$
1	0	0	0	
1	0	1	1	$A\bar{B}C$
1	1	0	1	$AB\bar{C}$
1	1	1	1	ABC

$\bar{A}BC$: assume valor 1 quando A=0 e B =1 e C= 1
$A\bar{B}C$: assume valor 1 quando A=1 e B =0 e C= 1
$AB\bar{C}$:assume valor 1 quando A=1 e B =1 e C= 0
ABC : assume valor 1 quando A=1 e B =1 e C= 1

$$M = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

A **função booleana**, **M**, é verdadeira (i.e., 1) se qualquer uma dessas quatro condições forem verdadeiras.

Tabela Verdade vs Função Booleana vs CI

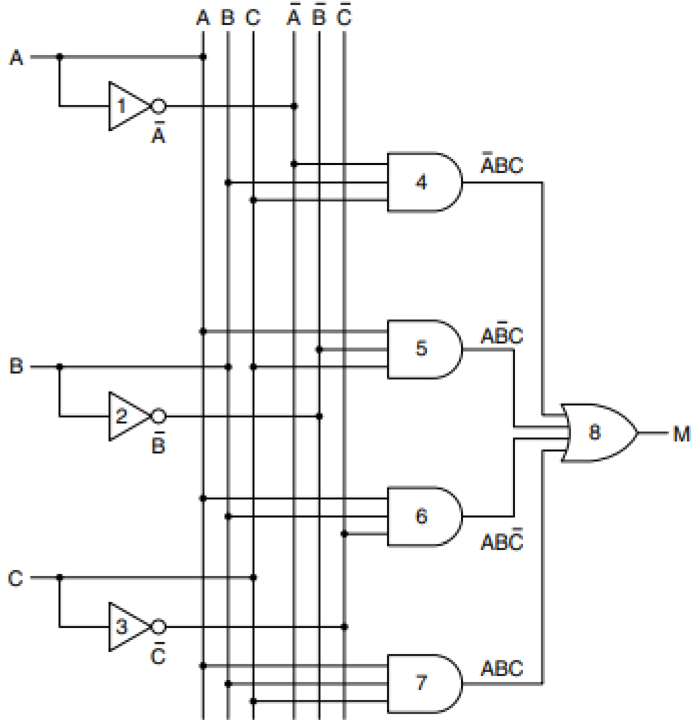
Tabela Verdade

A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Função booleana: M

$$M = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

Circuito Integrado



Método para implementar um circuito para uma função booleana

- **Método geral para implementar um circuito para qualquer função booleana:**
  - 1. Escreva a tabela verdade para a função.
  - 2. Forneça inversores para gerar o complemento de cada entrada.
  - 3. Desenhe uma porta AND para cada termo com um 1 na coluna de resultado.
  - 4. Conecte as portas AND às entradas apropriadas.
  - 5. Alimente a saída de todas as portas AND em uma porta OR

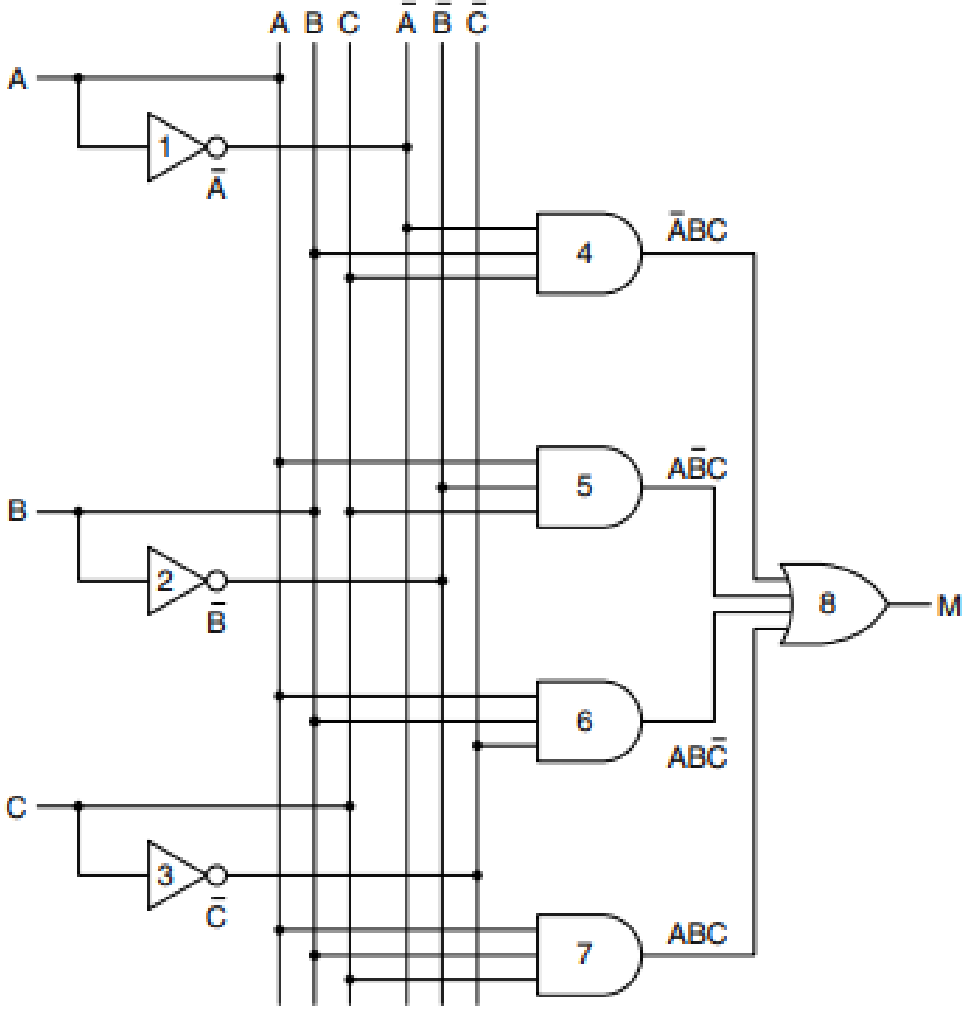
Tabela Verdade vs Função Booleana vs CI

$$M = \bar{A}BC + A\bar{B}C + AB\bar{C} + ABC$$

Função Booleana: M

Tabela Verdade

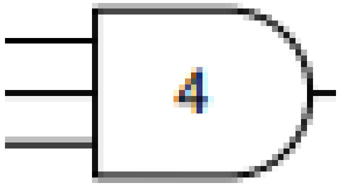
A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



CI: Circuito Integrado

Tabelas da Verdade: AND

- **A^B^C: Apresentar todas as possibilidades para as entradas A, B e C**



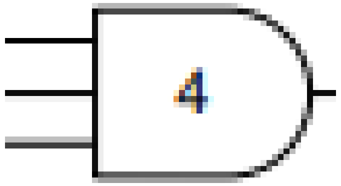
A	B	C	A^B^C

- O número de colunas corresponde ao número de entradas.

Uma tabela de três entradas teria 2^3 = oito linhas.

Tabelas da Verdade: AND

- **A^B^C: Apresentar todas as possibilidades para as entradas A, B e C**



A	B	C	A^B^C
V	V	V	V
V	V	F	F
V	F	V	F
V	F	F	F
F	V	V	F
F	V	F	F
F	F	V	F
F	F	F	F

- O número de colunas corresponde ao número de entradas.

Uma tabela de três entradas teria 2^3 = oito linhas.

Conversão: Tabela Verdade vs Função Booleana vs CI

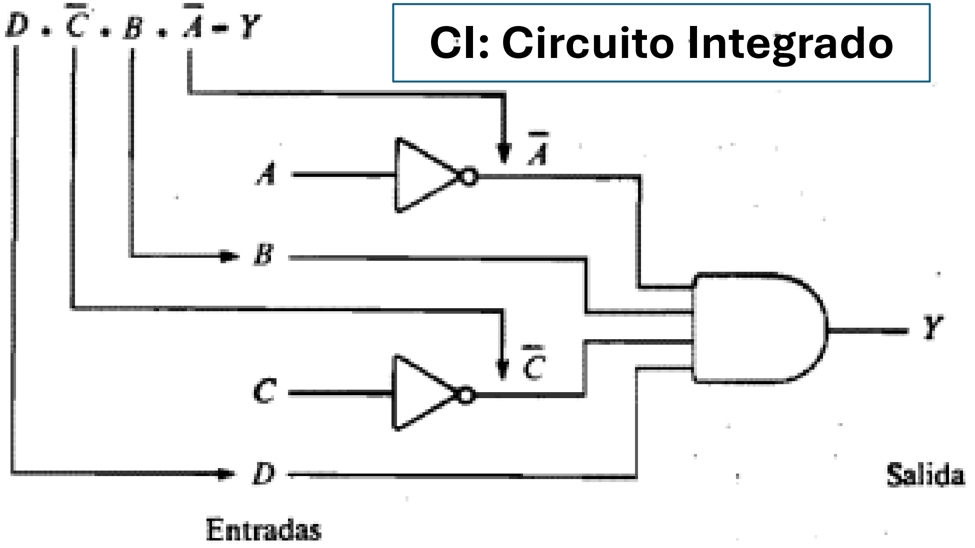
Tabela Verdade

Entradas				Salida	Entradas				Salida
D	*C*	*B*	*A*	*Y*	*D*	*C*	*B*	*A*	*Y*
0	0	0	0	0	1	0	0	0	0
0	0	0	1	0	1	0	0	1	0
0	0	1	0	0	1	0	1	0	1
0	0	1	1	0	1	0	1	1	0
0	1	0	0	0	1	1	0	0	0
0	1	0	1	0	1	1	0	1	0
0	1	1	0	0	1	1	1	0	0
0	1	1	1	0	1	1	1	1	0

$\rightarrow D \cdot \overline{C} \cdot B \cdot \overline{A} = Y$

Função Booleana: M

CI: Circuito Integrado



1. Exercício: Tabela Verdade de "A E B“

- Construa a tabela verdade para a expressão booleana (A AND B).

1. Exercício: Tabela Verdade de "A E B“

- Construa a tabela verdade para a expressão booleana (A AND B).
- Solução:

A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

2. Exercício: Tabela Verdade de "A OU B“

- Construa a tabela verdade para a expressão booleana (A OR B).

2. Exercício: Tabela Verdade de "A OU B“

- Construa a tabela verdade para a expressão booleana (A OR B).
- Solução:

A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

3. Exercício: Tabela Verdade de “NOT A“

- Construa a tabela verdade para a expressão booleana (NOT A).

3. Exercício: Tabela Verdade de “NOT A“

- Construa a tabela verdade para a expressão booleana (NOT A).
- Solução:

A	X
0	1
1	0

4. Exercício: Tabela Verdade de $A \oplus B$ (XOR)

- Construa a tabela verdade para a expressão booleana $A \oplus B$ (XOR)

4. Exercício: Tabela Verdade de $A \oplus B$ (XOR)

- Construa a tabela verdade para a expressão booleana $A \oplus B$ (XOR)
- Solução:

A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

5. **Exercício: Tabela Verdade de** $(A \vee B) \wedge (A \wedge \neg B)$

5. Exercício: Tabela Verdade de $(A \vee B) \wedge (A \wedge \neg B)$

Construa a tabela verdade para $(A \vee B) \wedge (A \wedge \neg B)$.

Solução:

A	B	$A \vee B$	$A \wedge \neg B$	$(A \vee B) \wedge (A \wedge \neg B)$
0	0	0	0	0
0	1	1	0	0
1	0	1	1	1
1	1	1	0	0

Referências:

- FUNDAMENTOS DE LOS MICROPROCESADORES – Autores: ROGER L. TOKHEIM, Henry Sibley, High School Mendota Heights, Minnesota, Segunda edição.
- Structured computer organization – Autores: Andrew S. Tanenbaum, Todd Austin. -- 6th ed