AULA 9
BLOCOS LÓGICOS

Profª Letícia Rittner

Circuitos digitais

- Combinacionais
  - Composto por um conjunto de portas lógicas
  - O valor da saída é função apenas dos valores atuais das entradas
- Sequenciais
  - Composto por um circuito combinacional mais elementos de memória
  - O valor da saída é função dos valores atuais das entradas e do estado atual do circuito

Circuitos combinacionais

- Análise
  - Dado um circuito, descobrir qual a funcionalidade implementada por ele
- Síntese
  - Dada uma funcionalidade desejada, projetar um circuito digital que a implementa
- Ferramentas
  - Expressão booleana
  - Tabela verdade
  - Símbolos esquemáticos
  - Diagrama de tempo

Circuitos combinacionais

A) Descrição do problema

y é 1 se a é 1, ou se b e c são 1. z é 1 se b ou c é 1, mas não ambos, ou se todos são 1.

C) Expressões booleanas

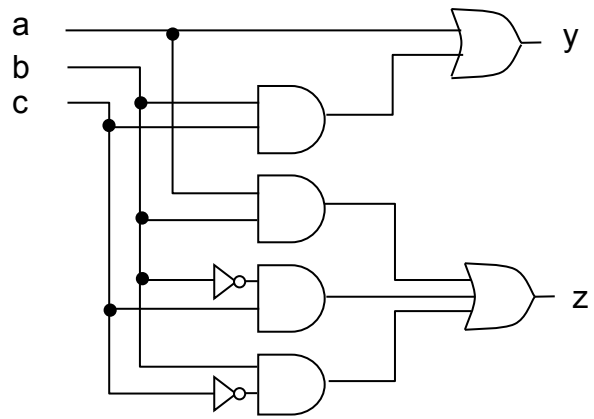
y = a'bc + ab'c' + ab'c + abc' + abc

z = a'b'c + a'bc' + ab'c + abc' + abc

B) Tabela verdade

Inputs			Outputs	
a	b	c	y	z
0	0	0	0	0
0	0	1	0	1
0	1	0	0	1
0	1	1	1	0
1	0	0	1	0
1	0	1	1	1
1	1	0	1	1
1	1	1	1	1

D) Portas lógicas



Adaptado da Profa. Alice

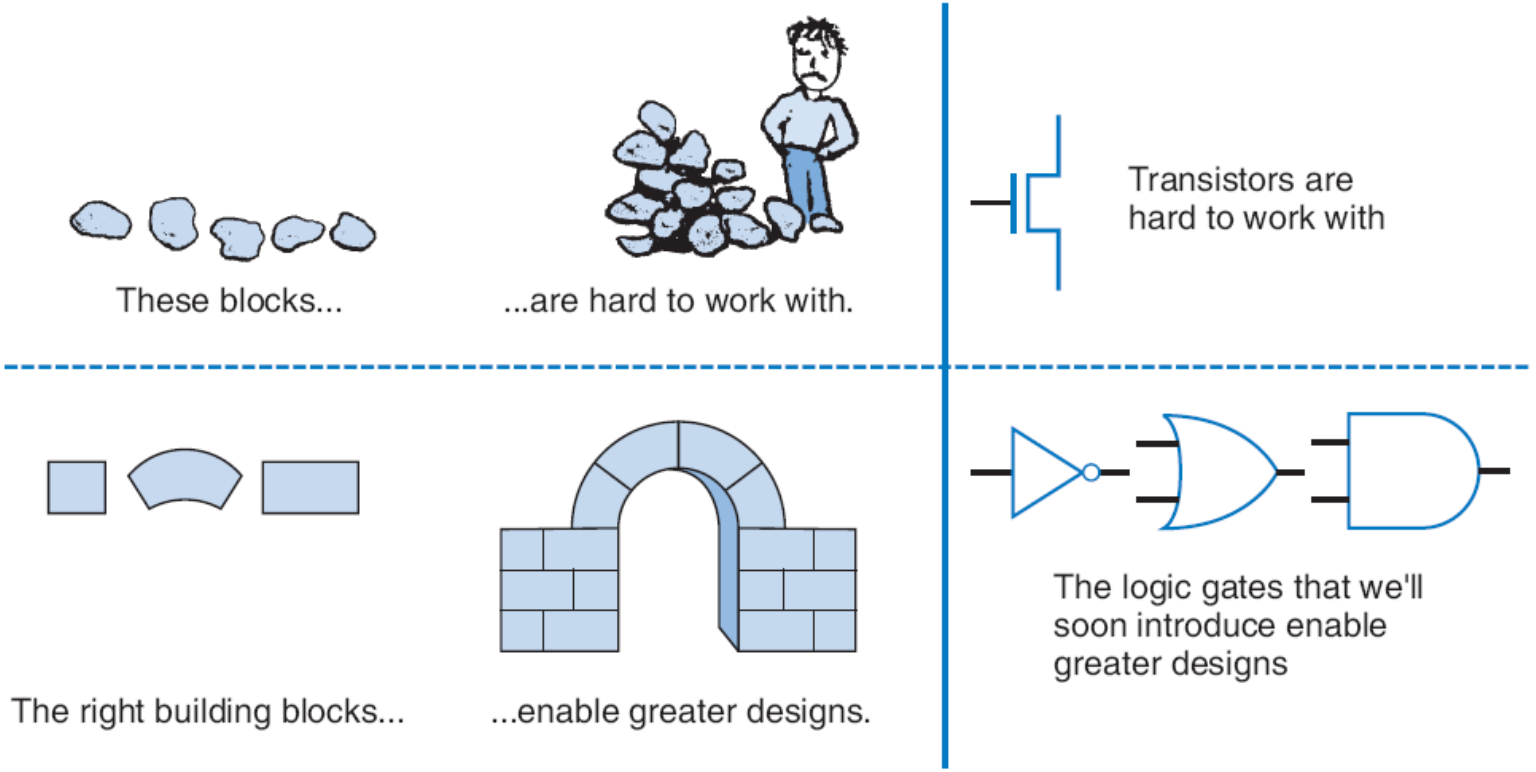
PORTAS LÓGICAS

NOT

AND e OR

NAND e NOR

Portas lógicas



- "Portas lógicas" são melhores elementos básicos do que transistores para a construção de circuitos digitais

Adaptado de Frank Vahid

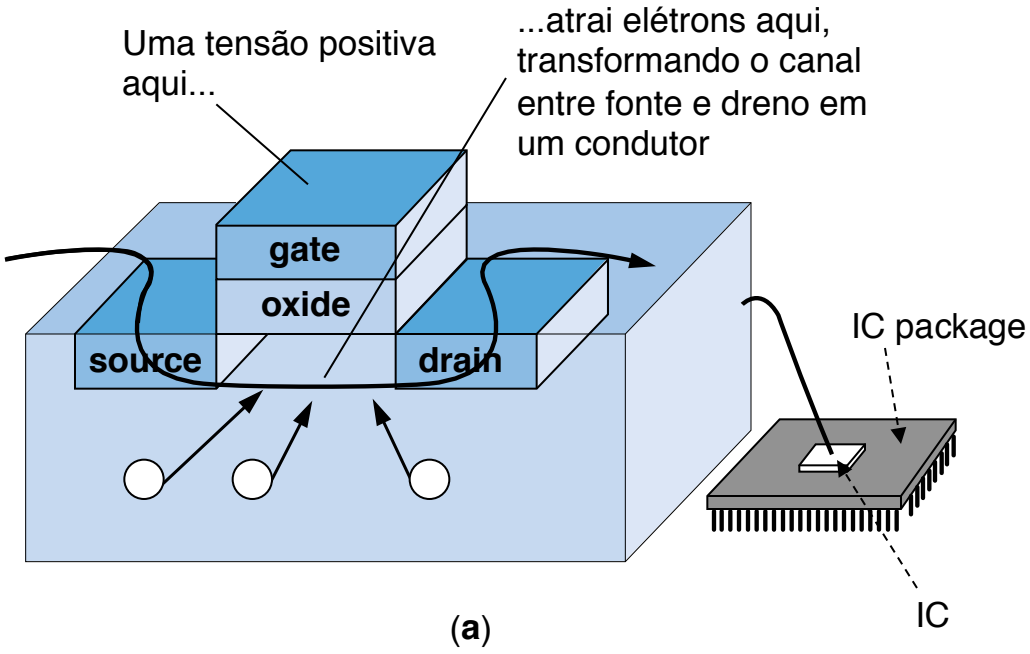
Significado dos valores verdade

- Níveis de tensão:
  - ALTO = 5 V = 1
  - BAIXO = 0 V = 0
- Capacitor carregado (1) ou descarregado (0)
- Chave fechada (1) ou aberta (0)
- Fusível intacto (1) ou queimado (0)

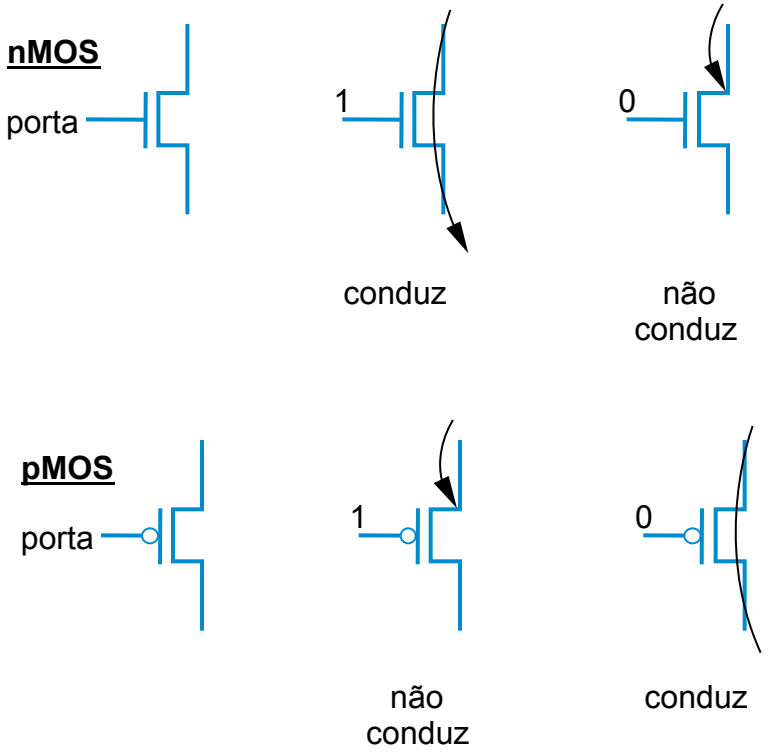
Convenção de ***Lógica Positiva***

Adaptado do Prof. Leonardo Abdala

Transistor CMOS

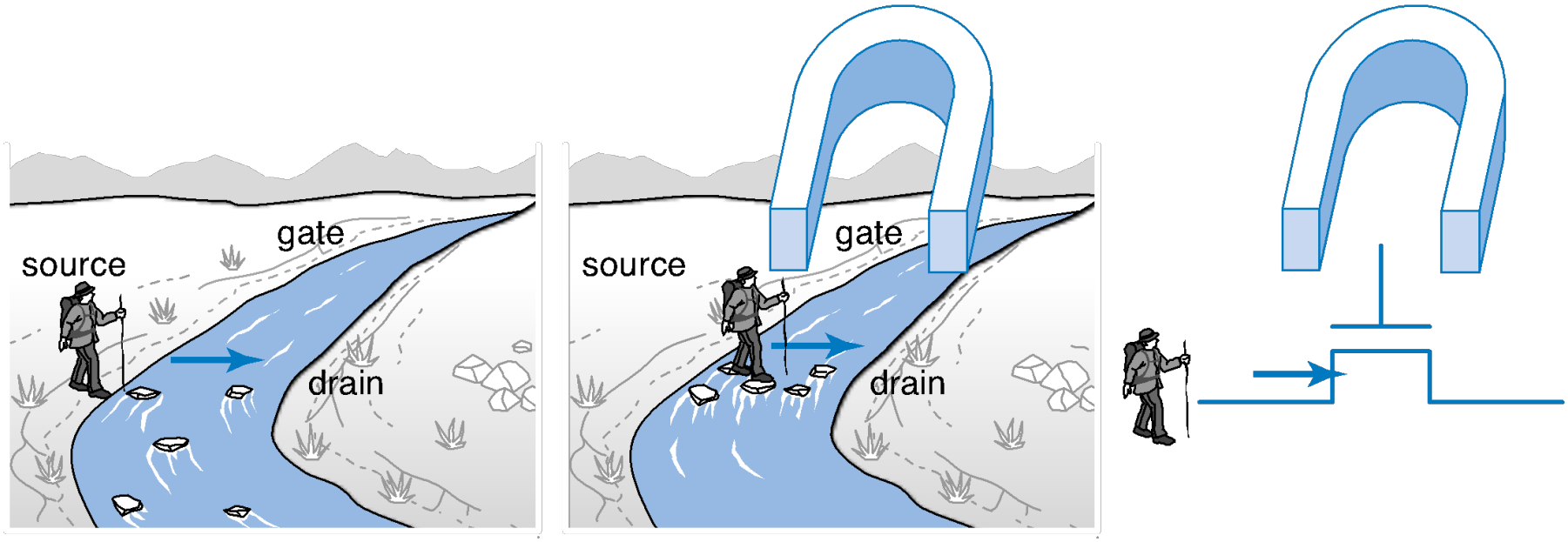


(**a**)



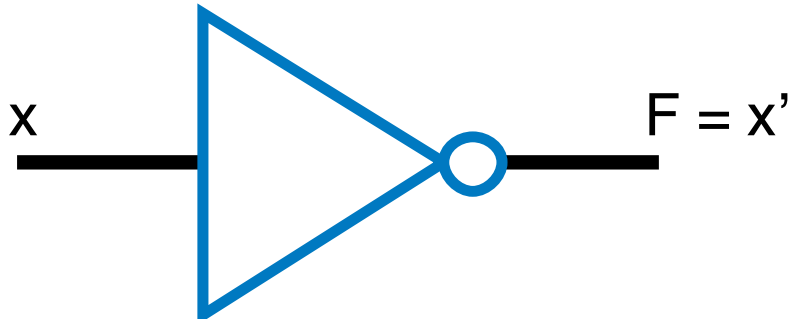
Adaptado de Frank Vahid

Transistor CMOS: analogia

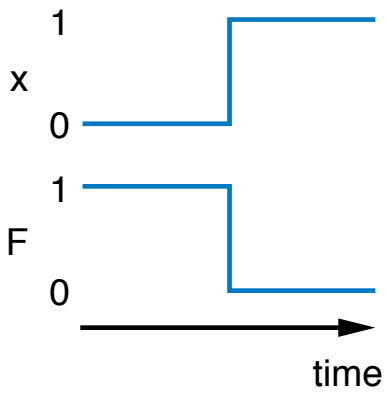


Adaptado de Frank Vahid

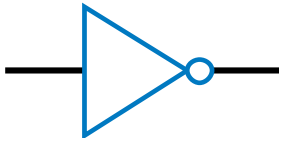
Porta NOT



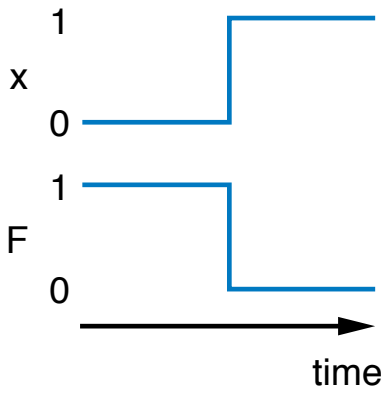
x	F
0	1
1	0

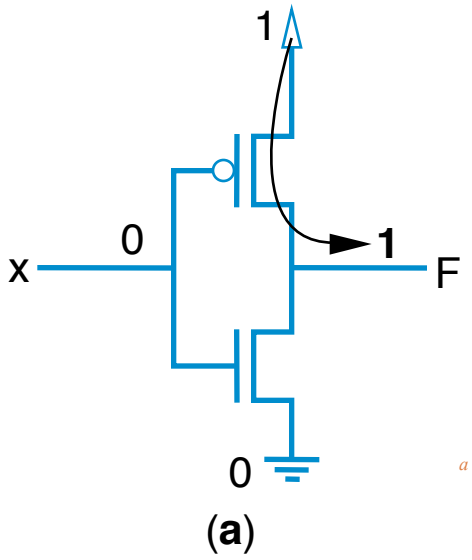


Porta NOT

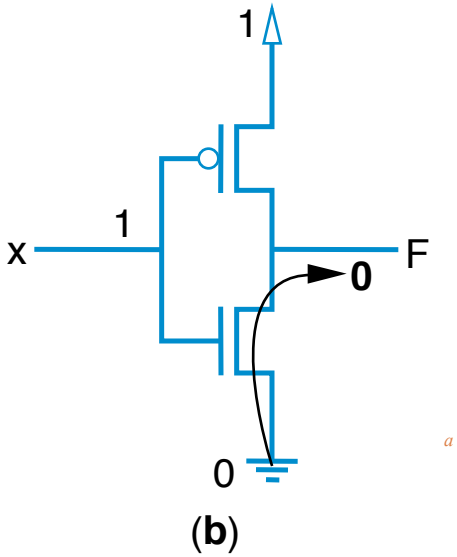


x	F
0	1
1	0



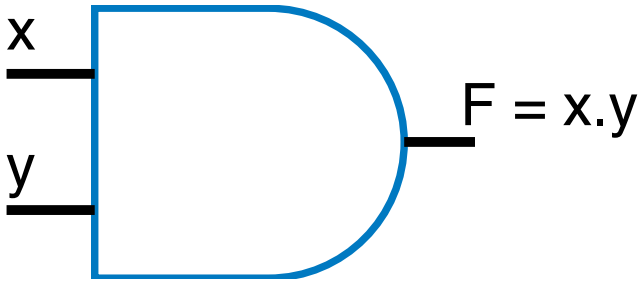


Quando a entrada é 0

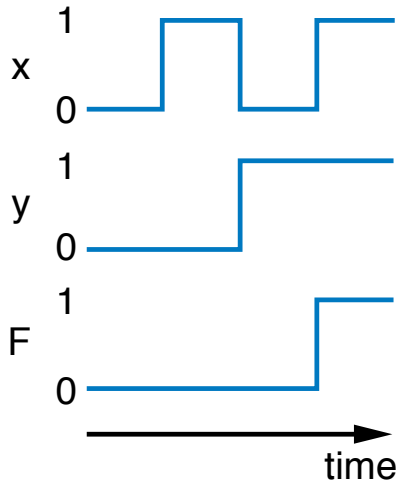


Quando a entrada é 1

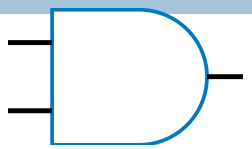
Porta AND



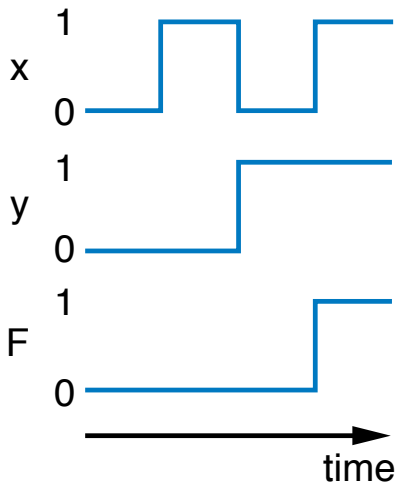
x	y	F
0	0	0
0	1	0
1	0	0
1	1	1

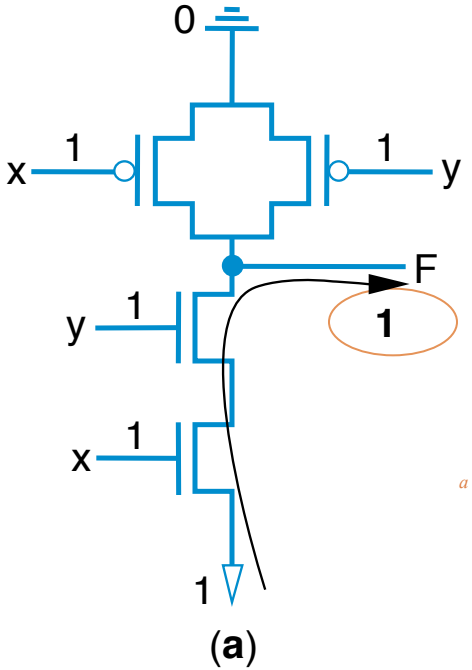


Porta AND

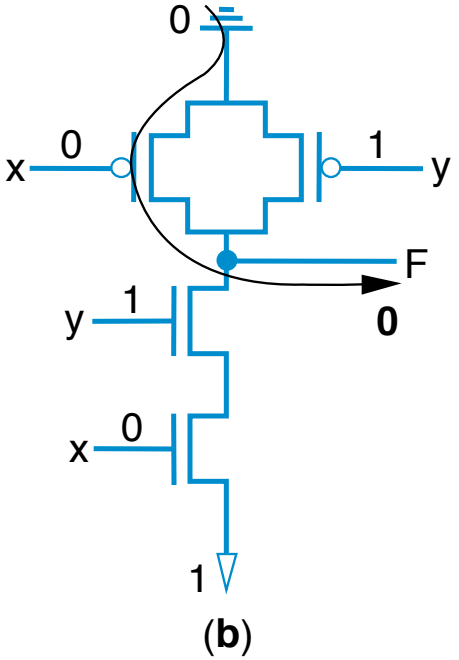


x	y	F
0	0	0
0	1	0
1	0	0
1	1	1



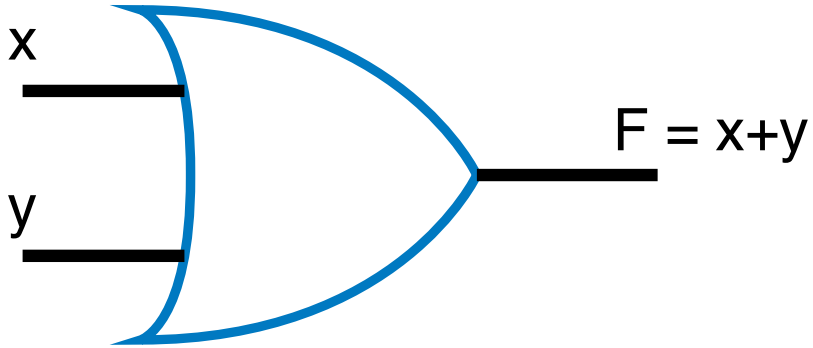


Quando 2 entradas são 1

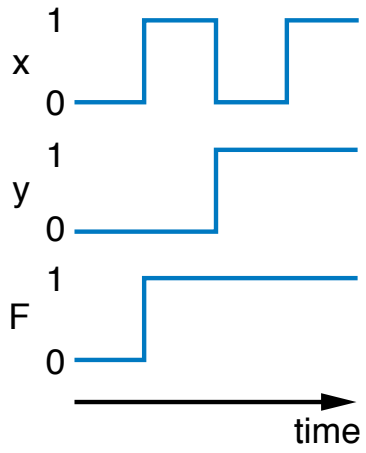


Quando 1 entrada é 0

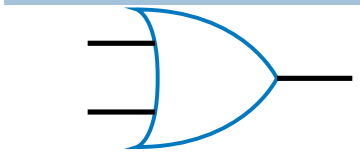
Porta OR



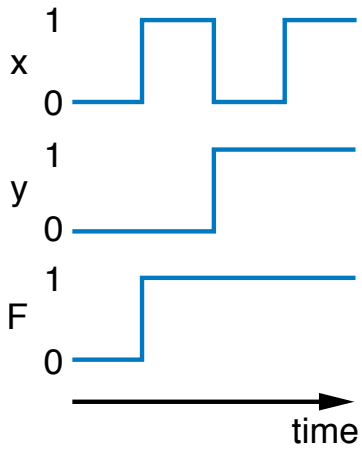
x	y	F
0	0	0
0	1	1
1	0	1
1	1	1

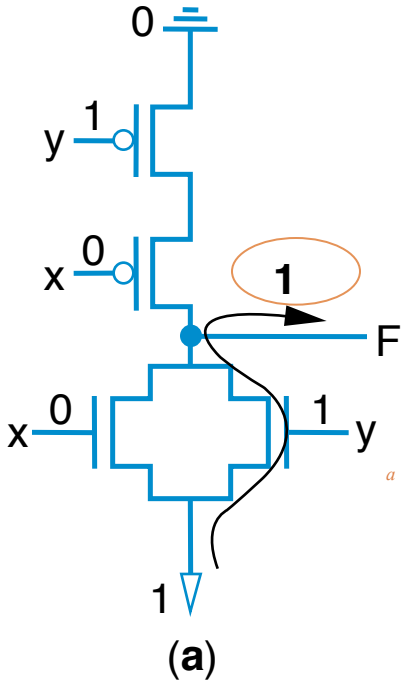


Porta OR

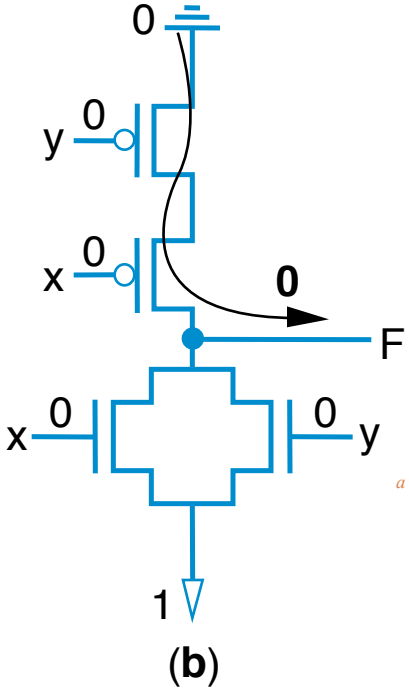


x	y	F
0	0	0
0	1	1
1	0	1
1	1	1





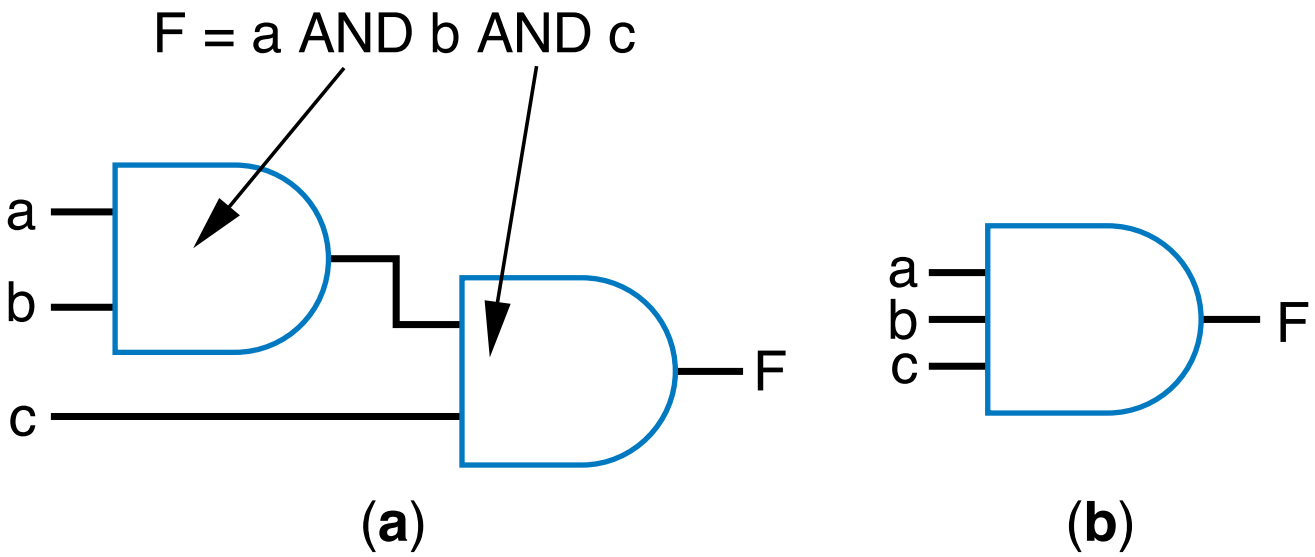
Quando 1 entrada é 0



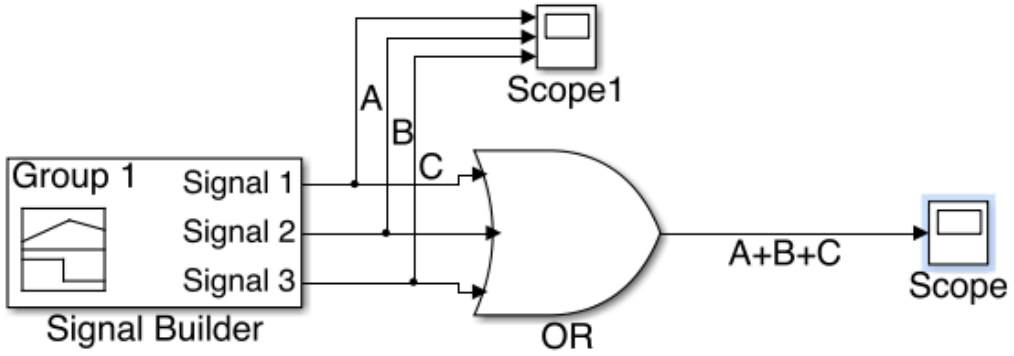
Quando 2 entradas são 0

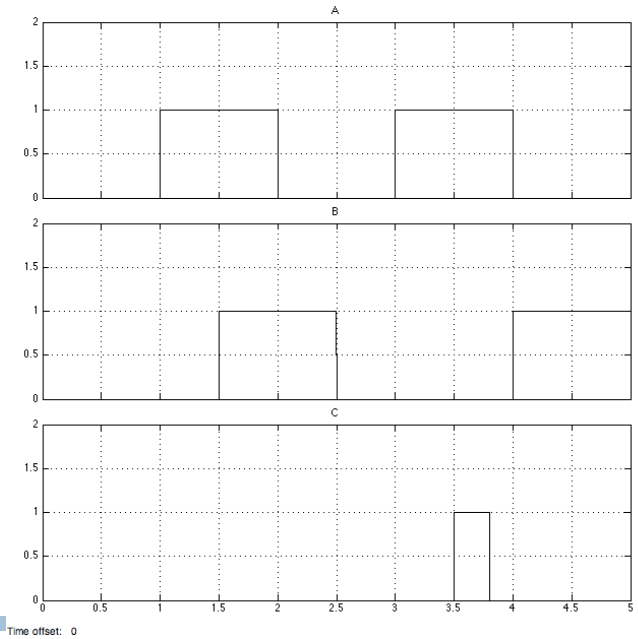
Porta AND: 3 entradas

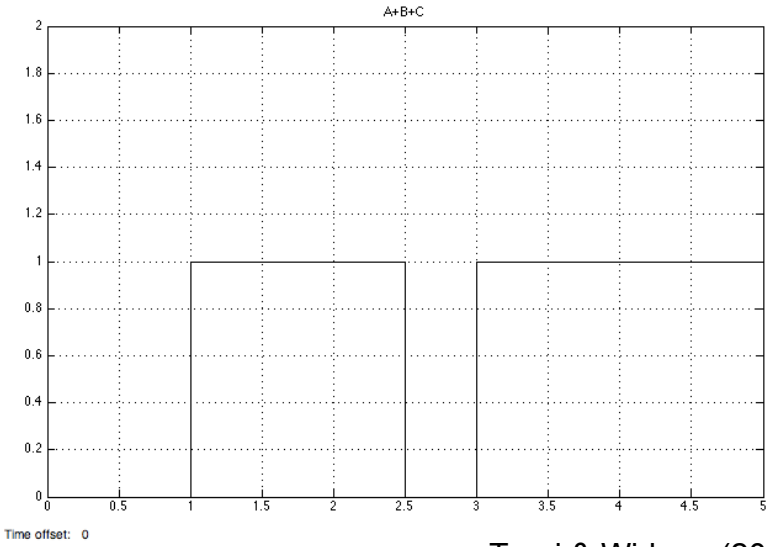
- **F = a.b.c**



Exemplo: diagrama de tempo

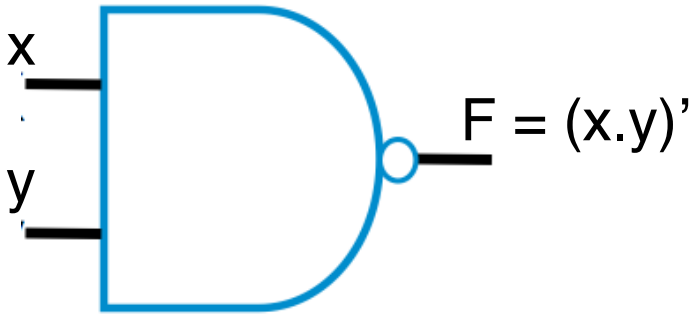


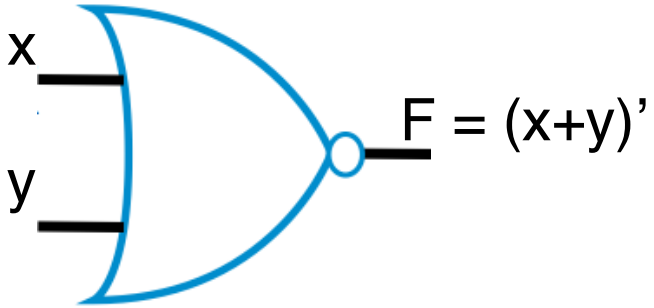




Tocci & Widmer (2011)

Portas NAND e NOR





x	y	F
0	0	1
0	1	1
1	0	1
1	1	0

x	y	F
0	0	1
0	1	0
1	0	0
1	1	0

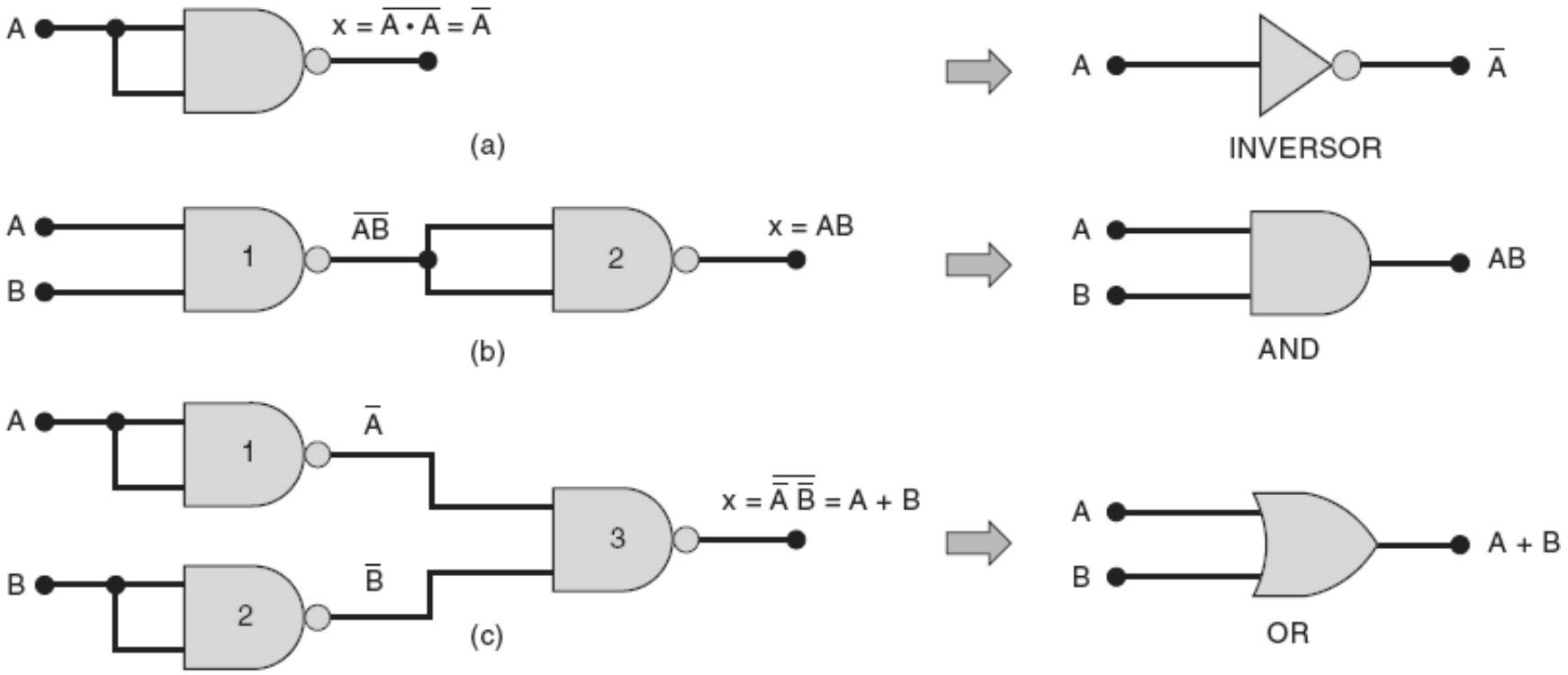
Porta NAND: universalidade

- Qualquer função boleana pode ser implementada usando apenas portas NAND
- Dizemos então que NAND é uma porta universal
- Como conseguimos isso?
  - Precisamos de AND, OR, e NOT para implementar qualquer função
  - Bastar implementar AND, OR e NOT a partir de portas NAND

Porta NAND: universalidade

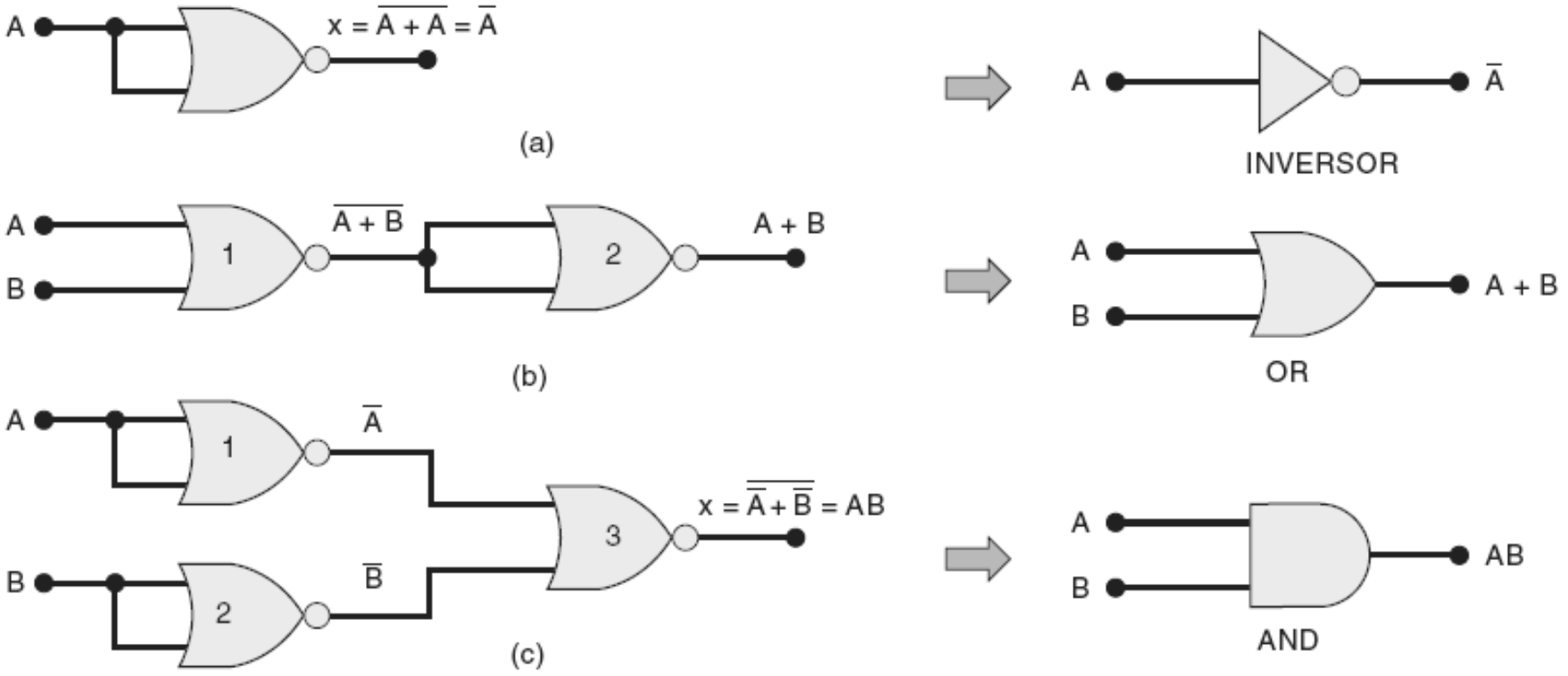
- Precisamos de AND, OR, e NOT
  - NOT: NAND de 2-entradas (entradas iguais)
  - AND: NAND seguido de NOT
  - OR: NAND precedido de NOTs
- Dizemos então que NAND é uma porta universal

Porta NAND



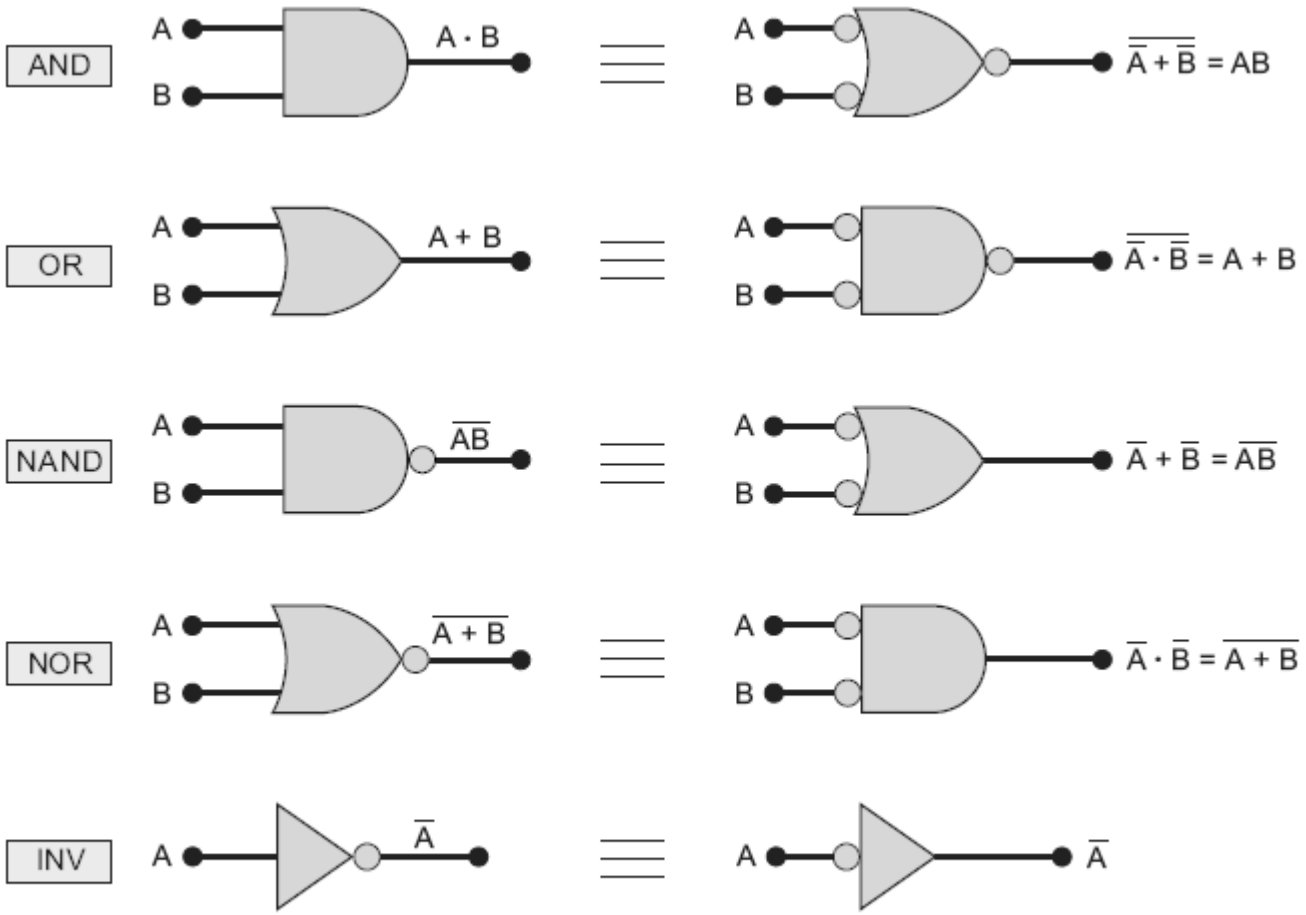
Tocci & Widmer (2011)

Porta NOR



Tocci & Widmer (2011)

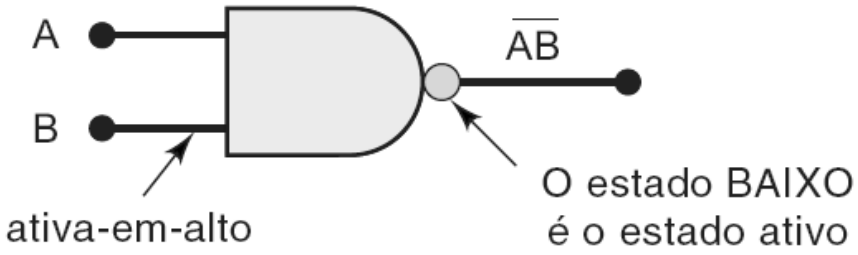
Simbologia Alternativa



Tocci & Widmer (2011)

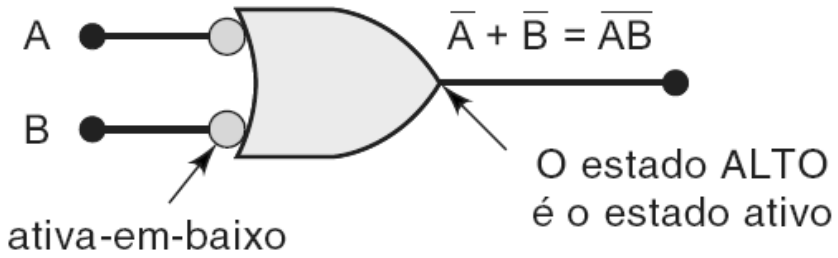
Simbologia Alternativa

- Interpretação dos dois símbolos da porta NAND.



(a)

Saída vai para o nível BAIXO somente quando todas as entradas forem ALTAS



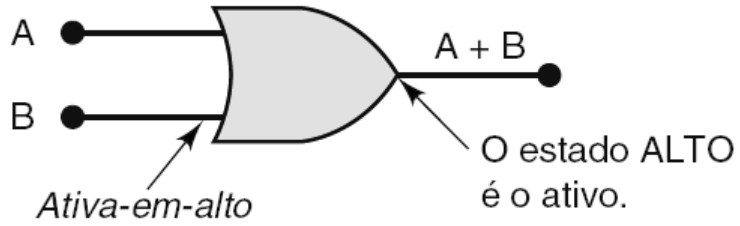
(b)

Saída é ALTA somente quando qualquer entrada é BAIXA

Tocci & Widmer (2011)

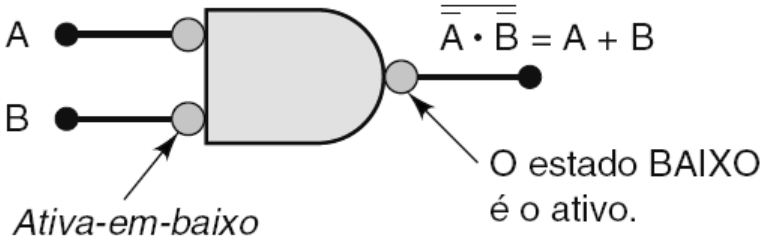
Simbologia Alternativa

- Interpretação dos dois símbolos da porta OR



(a)

A saída vai para o nível ALTO quando *qualquer* entrada for para o nível ALTO.



A saída vai para o nível BAIXO somente quando *todas* as entradas forem para o nível BAIXO.

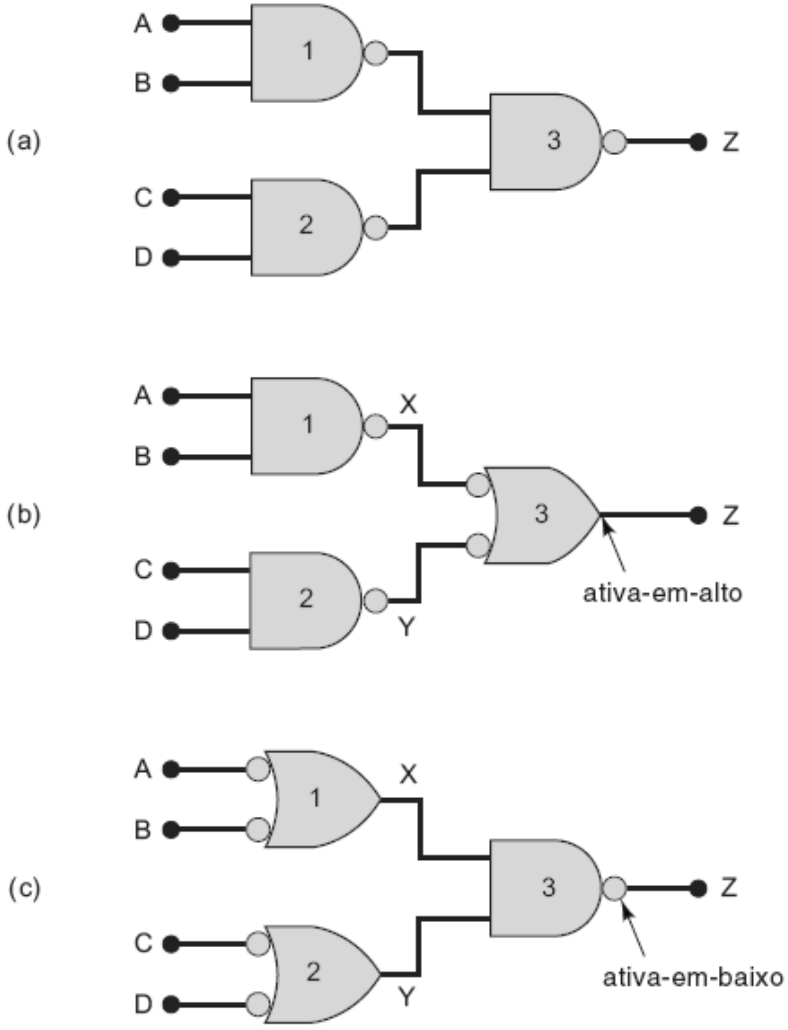
Tocci & Widmer (2011)

Simbologia Alternativa

Aspectos sobre as equivalências de símbolos lógicos:

- As equivalências podem ser estendidas para portas com qualquer número de entradas.
- Nenhum dos símbolos-padrão tem bolhas em suas entradas, e todos os símbolos alternativos os têm.
- NAND e NOR são portas inversoras
  - Os símbolos padrão e os símbolos alternativos para cada um terão uma bolha sobre a entrada ou a saída.
- Portas AND e OR são portas não inversoras.
  - Os símbolos alternativos para cada um terá bolhas em ambas as entradas e as saídas.

Exemplo



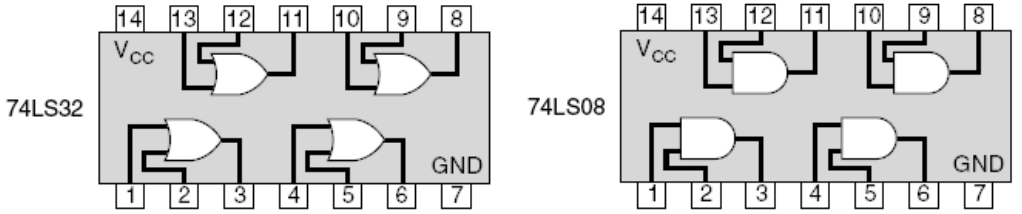
A	B	C	D	Z
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	0
0	1	0	1	0
0	1	1	0	0
0	1	1	1	1
1	0	0	0	0
1	0	0	1	0
1	0	1	0	0
1	0	1	1	1
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

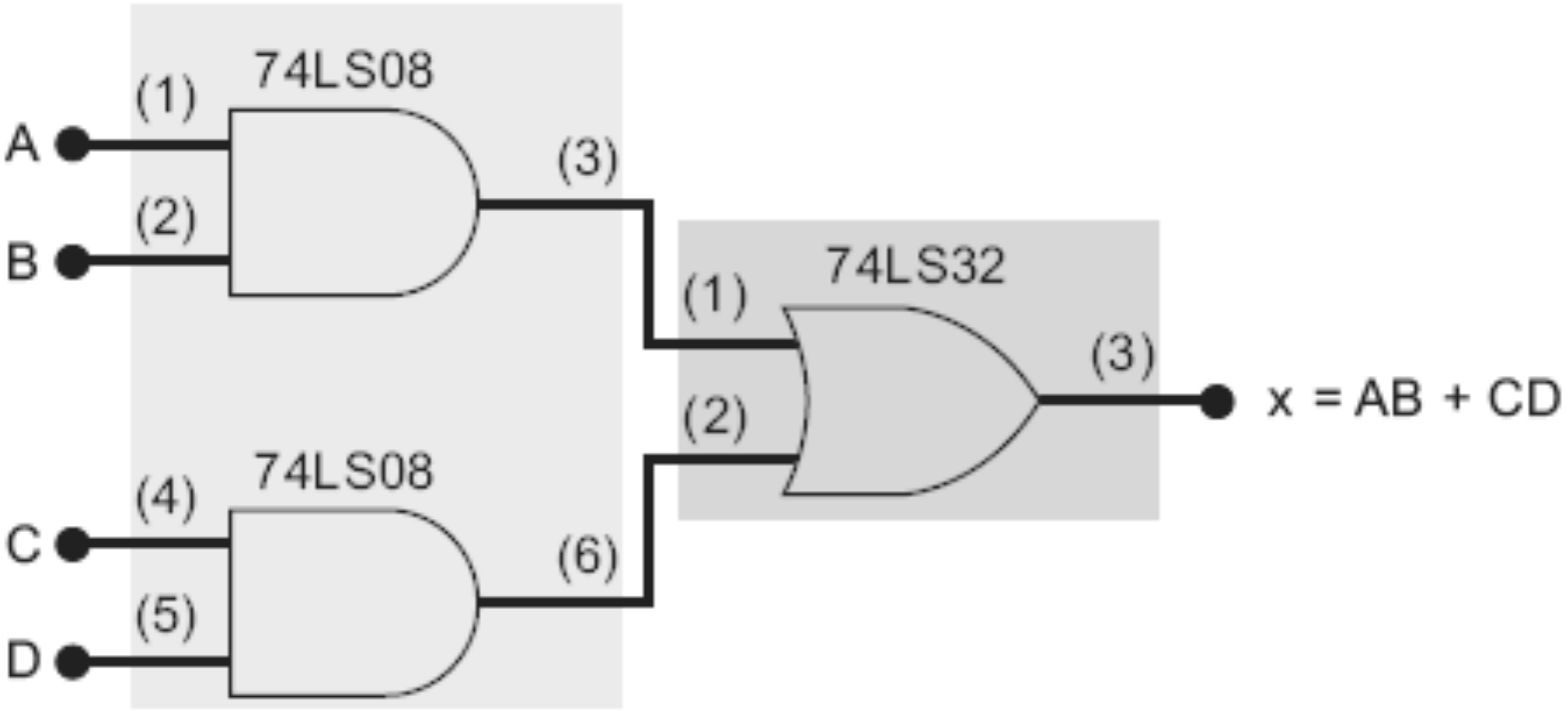
(d)

Tocci & Widmer (2011)

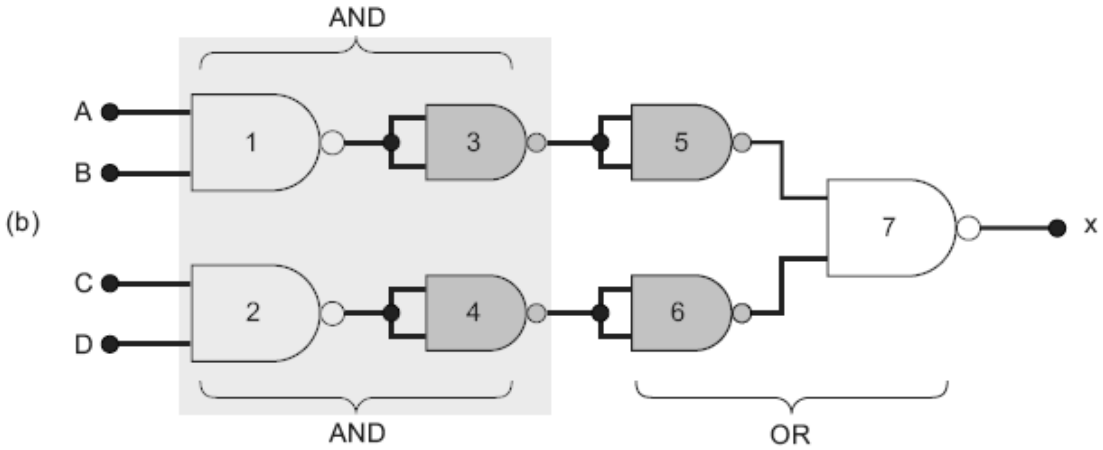
Exemplo

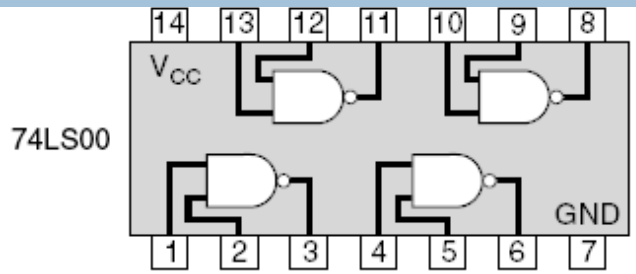
Tocci & Widmer (2011)



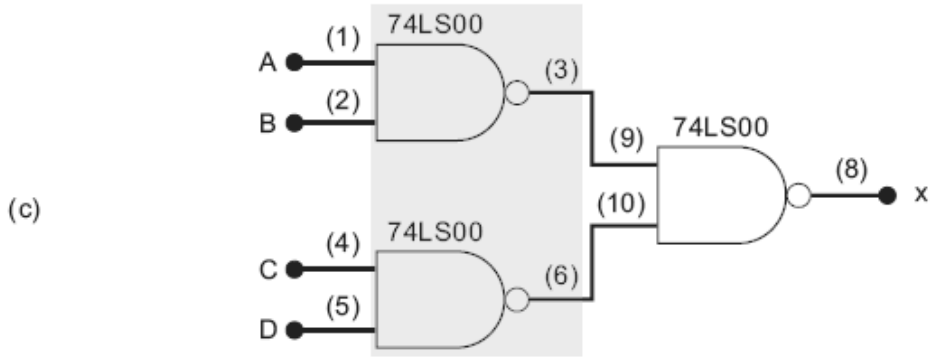


Exemplo





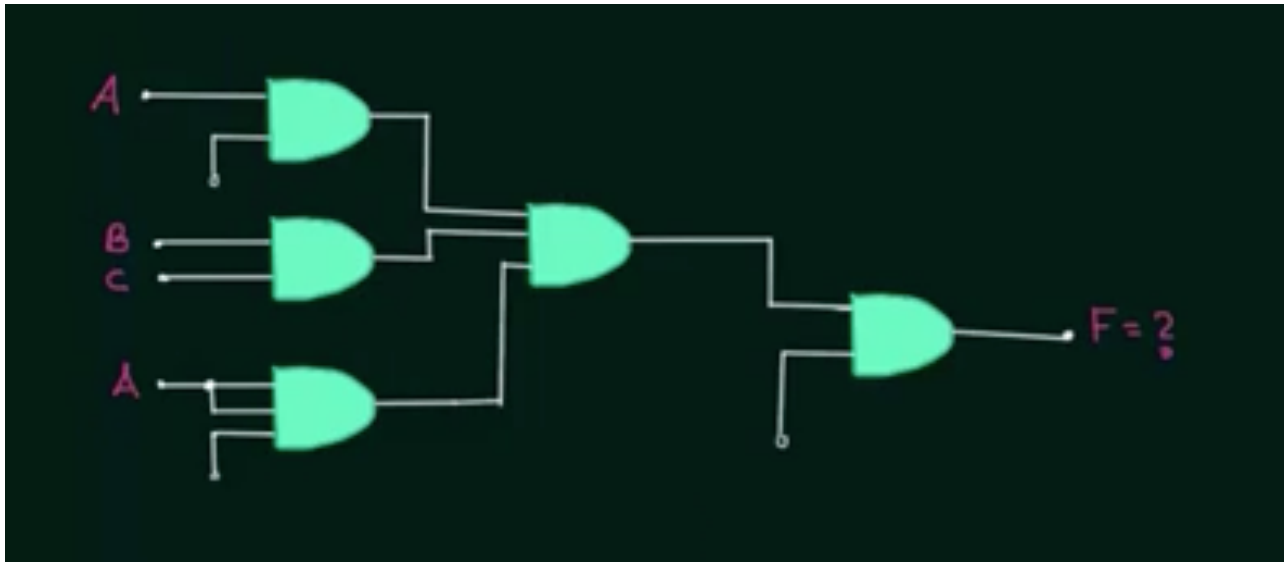
Após eliminar as duplas inversões



Tocci & Widmer (2011)

Para casa

- No circuito escreva a saída F em função das entradas A, B e C, simplificando quando possível.



Observação: considere lógica TTL, onde entradas "abertas" se comportam como "1"

Para casa

- Encontre o número mínimo de portas NAND de 2 entradas para implementar as expressões a seguir. Desenhe o circuito resultante em cada caso:
  - Y = A'.B
  - Z = A'+B

Para casa

- Implemente uma porta NOR usando apenas portas NAND