

# **Circuitos Lógicos Combinacionais**

## **(parte 3)**

*Sistemas de Informação*

*CPCX – UFMS*

*Slides: Prof. Renato F. dos Santos*

*Adaptação: Fernando Maia da Mota*

## 4.6 Circuitos Exclusive-OR e Exclusive-NOR

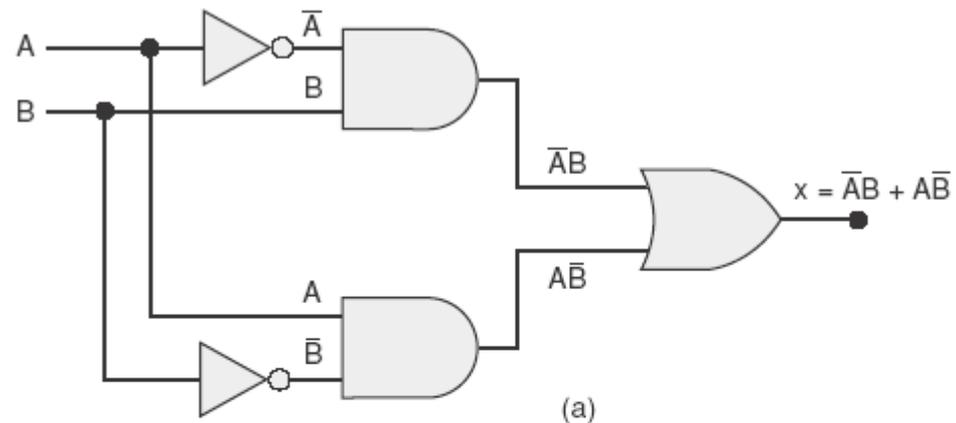
### – Exclusive-OR (OU-EXCLUSIVO)

- Este circuito produz uma saída em nível ALTO sempre que as duas entradas estiverem em níveis opostos
- Na Figura 4.20(a), a expressão de saída para esse circuito é  $x = \bar{A}B + A\bar{B}$
- $x = 1$  em dois casos:  $A = 0, B = 1$  e  $A = 1, B = 0$
- Esse é o circuito exclusive-OR (XOR)
- A Figura 4.20(b) mostra o símbolo do circuito XOR
- Normalmente é denominado porta XOR, consideraremos um outro tipo de porta lógica

## 4.6 Circuitos Exclusive-OR e Exclusive-NOR (Continuação)

- **Tem apenas duas entradas**
- **Uma forma abreviada usada para representar uma expressão XOR é  $x = A \oplus B$**
- **o símbolo  $\oplus$  representa a operação da porta XOR**

**FIGURA 4.20**  
 (a) Circuito XOR e  
 tabela-verdade;  
 (b) símbolo tradicional  
 para a porta XOR;



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



## 4.6 Circuitos Exclusive-OR e Exclusive-NOR (Continuação)

- **Exclusive-NOR (NOU-EXCLUSIVO)**
  - O circuito exclusive-NOR (XNOR), opera de forma completamente oposta ao circuito XOR
  - XNOR gerará uma saída em nível ALTO se as duas entradas estiverem no mesmo nível lógico
  - A Figura 4.21(a) mostra um circuito XNOR e sua expressão de saída é  $x = AB + \bar{A}\bar{B}$
  - $x$  é 1 para dois casos:  $A = B = 1$  e  $A = B = 0$
  - Deve ficar evidente que a saída de um circuito XNOR é exatamente o inverso da saída do circuito XOR
  - Seu símbolo é obtido acrescentando um pequeno círculo na saída do símbolo da porta XOR [Fig. 4.20(b)]

## 4.6 Circuitos Exclusive-OR e Exclusive-NOR (Continuação)

- Também tem apenas duas entradas
- Uma forma abreviada de representar uma expressão XNOR é  $x = \overline{A \oplus B}$

# Resumo

- **As características de uma porta XOR são:**
  - Ela tem apenas duas entradas e a expressão para sua saída é

$$x = \bar{A}B + A\bar{B} = A \oplus B$$

- Sua saída será nível ALTO apenas quando as duas entradas estiverem em níveis diferentes

- **Porta XNOR:**

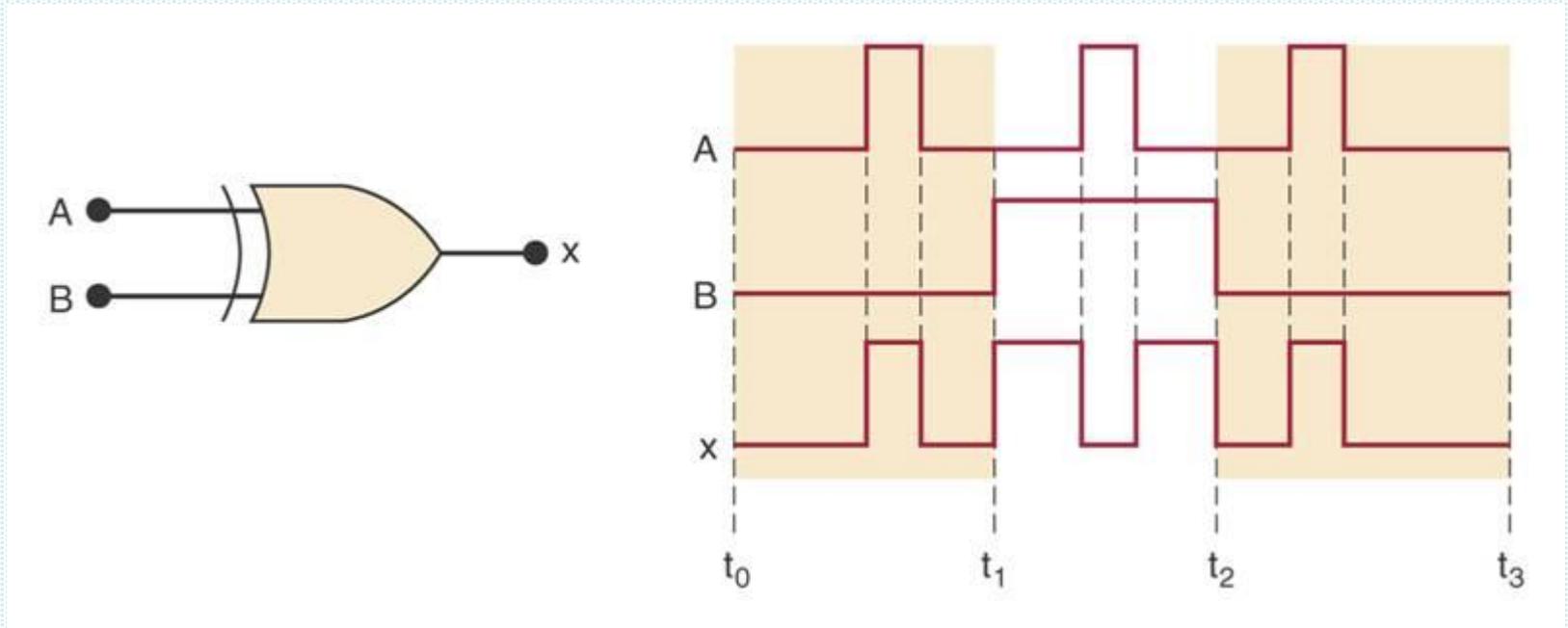
- Também tem apenas duas entradas e a expressão para sua saída é

$$x = AB + \bar{A}\bar{B} = \overline{A \oplus B}$$

- Sua saída será nível ALTO apenas quando as duas entradas estiverem no mesmo nível lógico

# Exemplo 4.16

- **Determine a forma de onda de saída para as formas de onda de entrada mostradas na Figura 4.22.**



# Exemplo 4.17

- A notação  $x_1 x_2$  representa um número binário que pode ter qualquer valor (00, 01, 10 ou 11); por exemplo, quando  $x_1 = 1$  e  $x_2 = 0$ , o número é 10, e assim por diante. De forma similar, a notação  $y_1 y_2$  representa um outro número binário de dois bits. Projete um circuito lógico usando as entradas  $x_1, x_2$  e  $y_1, y_2$  cuja saída será nível ALTO apenas quando os dois números binários,  $x_1 x_2$  e  $y_1 y_2$ , forem iguais.

# Exemplo 4.17 (Continuação)

## – Solução

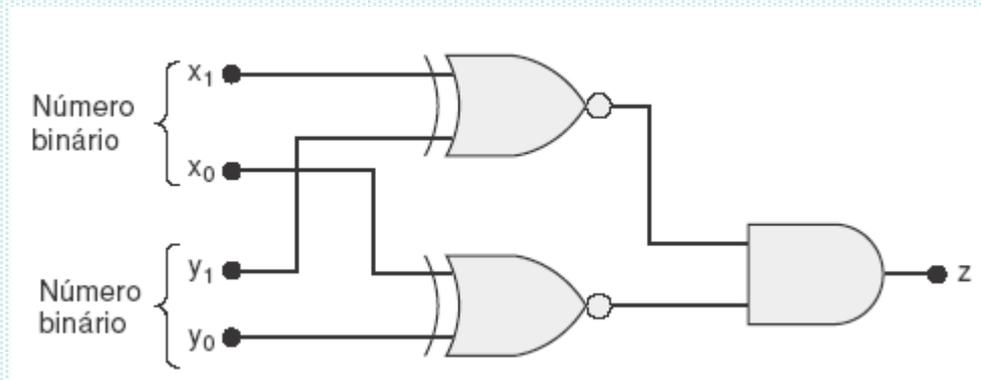
- O primeiro passo é construir a tabela-verdade. A saída  $z$  é nível ALTO sempre que  $x1 = y1$  e  $x2 = y2$

$x1$	$x2$	$y1$	$y2$	$Z$
0	0	0	0	1
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
0	1	1	0	0
0	1	1	1	0
1	0	0	0	0
1	0	0	1	0
1	0	1	0	1
1	0	1	1	0
1	1	0	0	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

# Exemplo 4.17 (Continuação)

- O próximo passo é obter a expressão na forma de soma-de-produtos e então implementar o resultado

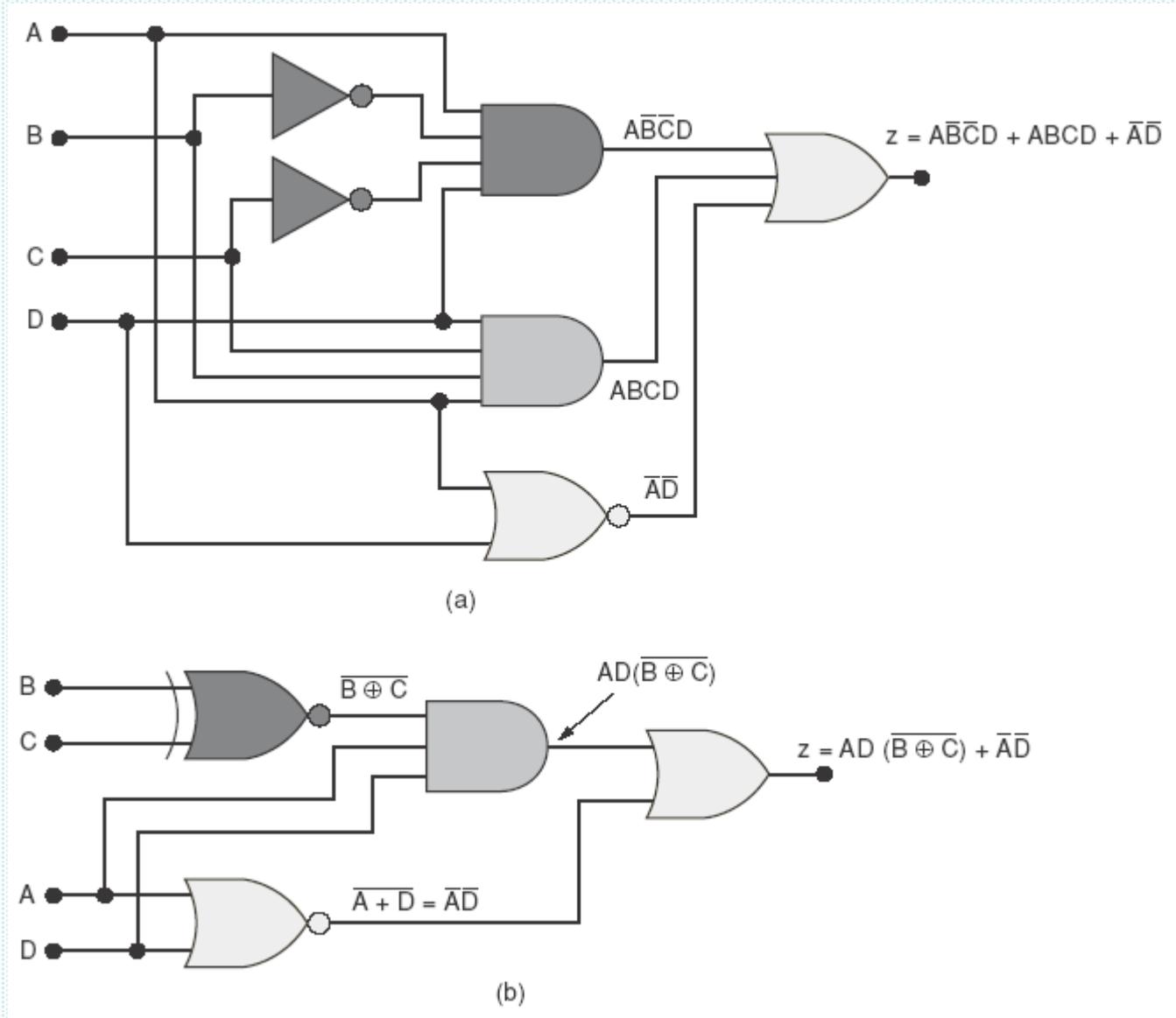
**FIGURA 4.23**  
Circuito para detectar a  
igualdade de dois números  
binários de dois bits.



# Exemplo 4.18

- Ao simplificar a expressão para a saída de um circuito lógico combinacional, você pode encontrar operações XOR ou XNOR durante a fatoração. Isso permite muitas vezes o uso de portas XOR ou XNOR na implementação do circuito final. Para ilustrar, simplifique o circuito da Figura 4.24(a).
- Solução

**FIGURA 4.24**  
 O Exemplo 4.18 mostra como uma porta XNOR pode ser usada para simplificar a implementação de um circuito.



## 4.7 Circuitos gerador e verificador de paridade

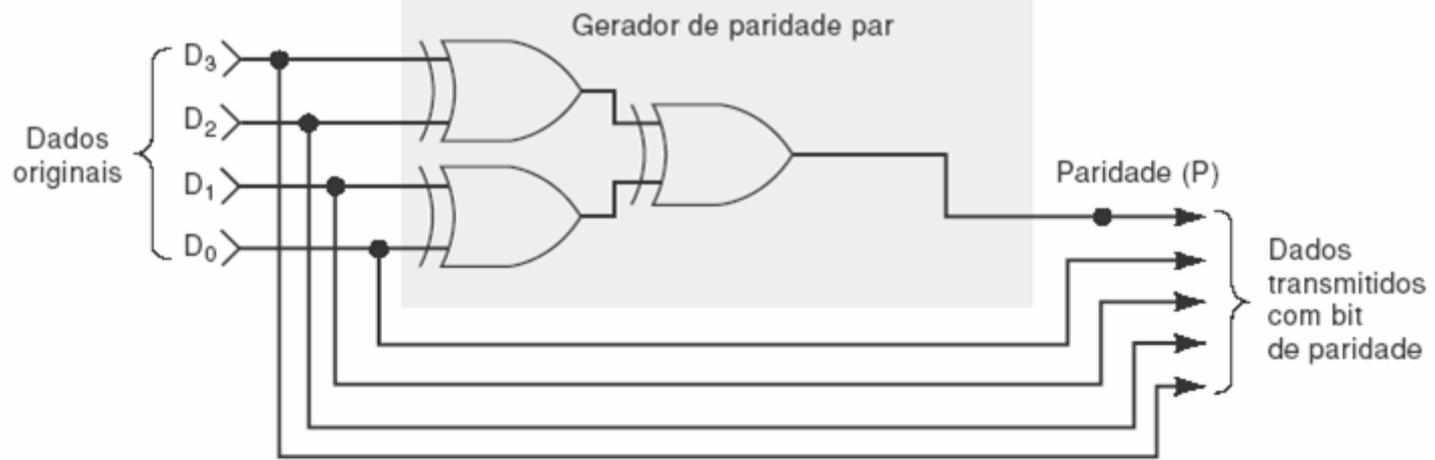
- **A Figura 4.25 mostra um exemplo de um tipo de circuito lógico que é usado para geração de paridade e verificação de paridade**
- **Esse exemplo particular usa um grupo de quatro bits como os dados a serem transmitidos**
- **Faz uso de paridade par (pode ser facilmente adaptado para paridade ímpar e um número qualquer de bits)**

## 4.7 Circuitos gerador e verificador de paridade (Continuação)

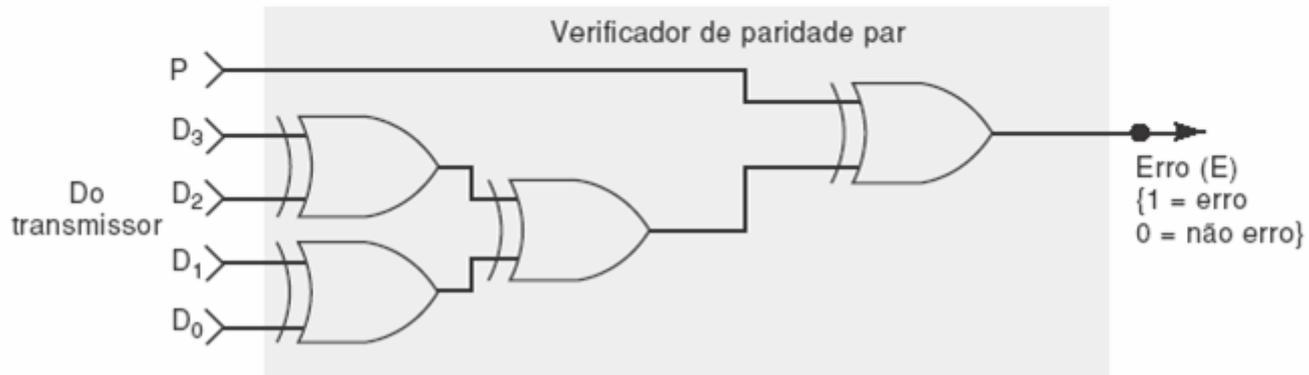
- **Na Figura 4.25(a), o conjunto de dados a serem transmitidos são aplicados ao circuito gerador de paridade**
- **O circuito produz um bit de paridade P em sua saída**
- **O bit P é transmitido juntamente com os demais bits**

## 4.7 Circuitos gerador e verificador de paridade (Continuação)

- **Na Figura 4.25(b), esses cinco bits (dado + paridade) entram no circuito verificador de paridade do receptor**
- **O circuito gera uma saída de erro, E, indicando se ocorreu ou não, um erro em um único bit**



(a)



(b)

**FIGURA 4.25**

Portas XOR utilizadas para implementar um gerador de paridade e um verificador de paridade para um sistema que usa paridade par.

# Exemplo 4.19

- **Determine a saída do gerador de paridade para cada um dos seguintes conjuntos de dados de entrada,  $D3 D2 D1 D0$ : (a) 0111; (b) 1001; (c) 0000; (d) 0100. Veja a Figura 4.25(a).**
- **Solução**
  - **Os resultados são: (a) 1; (b) 0; (c) 0 e (d) 1.**

# Exemplo 4.20

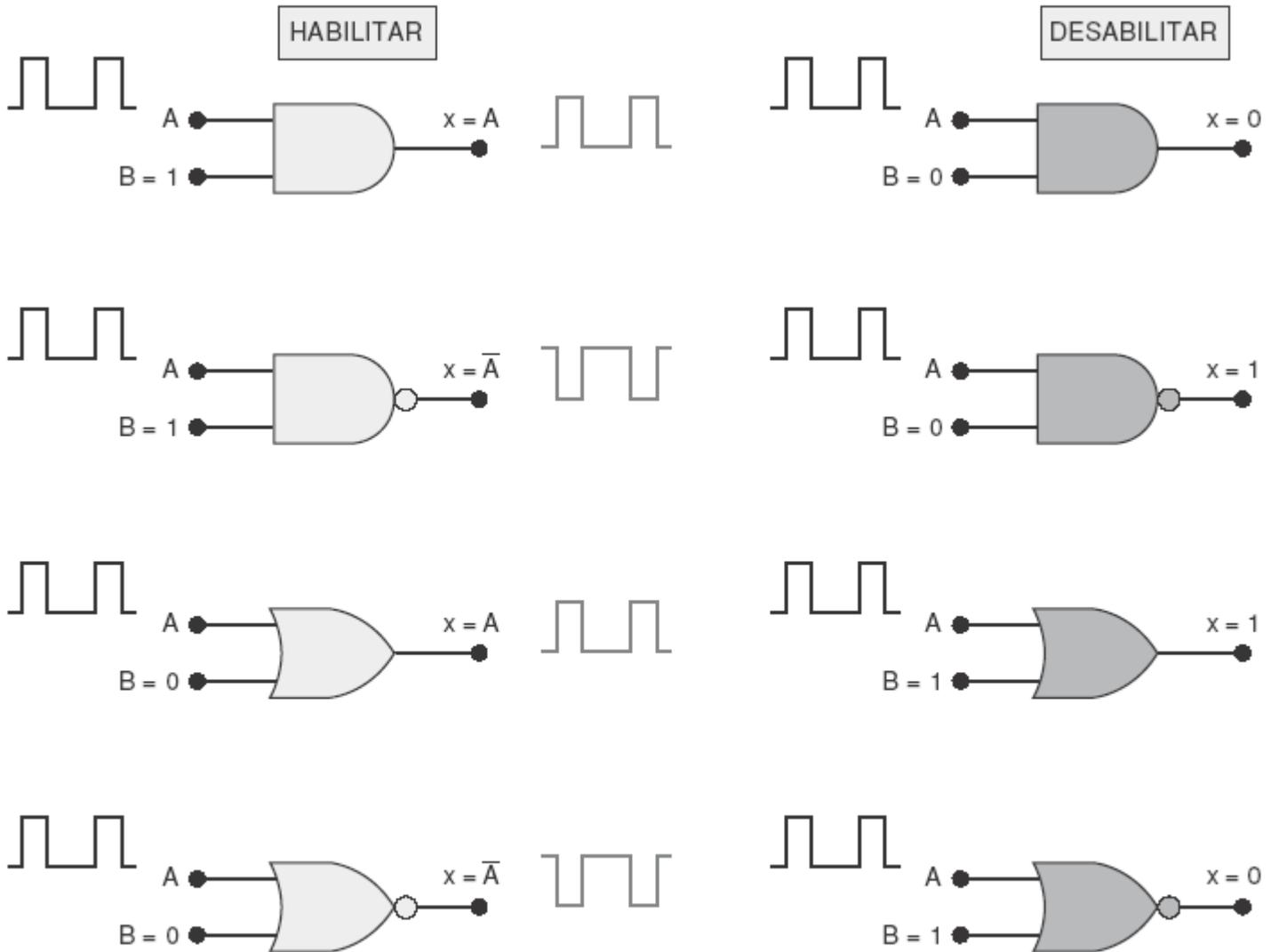
- **Determine a saída do verificador de paridade [veja a Figura 4.25(b)] para cada um dos conjuntos de dados enviados pelo transmissor:**

	<i>P</i>	<i>D3</i>	<i>D2</i>	<i>D1</i>	<i>D0</i>
<b>(a)</b>	0	1	0	1	0
<b>(b)</b>	1	1	1	1	0
<b>(c)</b>	1	1	1	1	1
<b>(d)</b>	1	0	0	0	0

- **Os resultados são: (a) 0; (b) 0; (c) 1 e (d) 1.**

## 4.8 Circuitos para habilitar / desabilitar

- Cada uma das portas lógicas básicas pode ser usada para controlar a passagem de um sinal lógico da entrada para a saída
- Um sinal lógico A é aplicado em uma das entradas de cada porta lógica
- A outra entrada de cada porta lógica é a entrada de controle B
- O nível lógico na entrada de controle determina se o sinal de entrada está habilitado a alcançar a saída ou impedido (desabilitado)



**FIGURA 4.26**

As quatro portas básicas podem habilitar ou desabilitar a passagem de um sinal de entrada A, sob o controle de um nível lógico na entrada de controle B.

## 4.8 Circuitos para habilitar / desabilitar (Continuação)

- **Observe que quando portas não-inversoras (AND e OR) são habilitadas a saída segue exatamente o sinal de A**
- **Ao contrário, quando portas inversoras (NAND e NOR) são habilitadas, a saída será exatamente o inverso do sinal A**
- **As portas AND e NOR geram uma saída constante em nível BAIXO quando estão desabilitadas**
- **Ao contrário as portas NAND e OR geram uma saída constante em nível ALTO quando estão desabilitadas**