



Circuitos Digitais I - 6878

Nardênio Almeida Martins

Universidade Estadual de Maringá
Departamento de Informática

Bacharelado em Ciência da Computação

Aula de Hoje

Roteiro

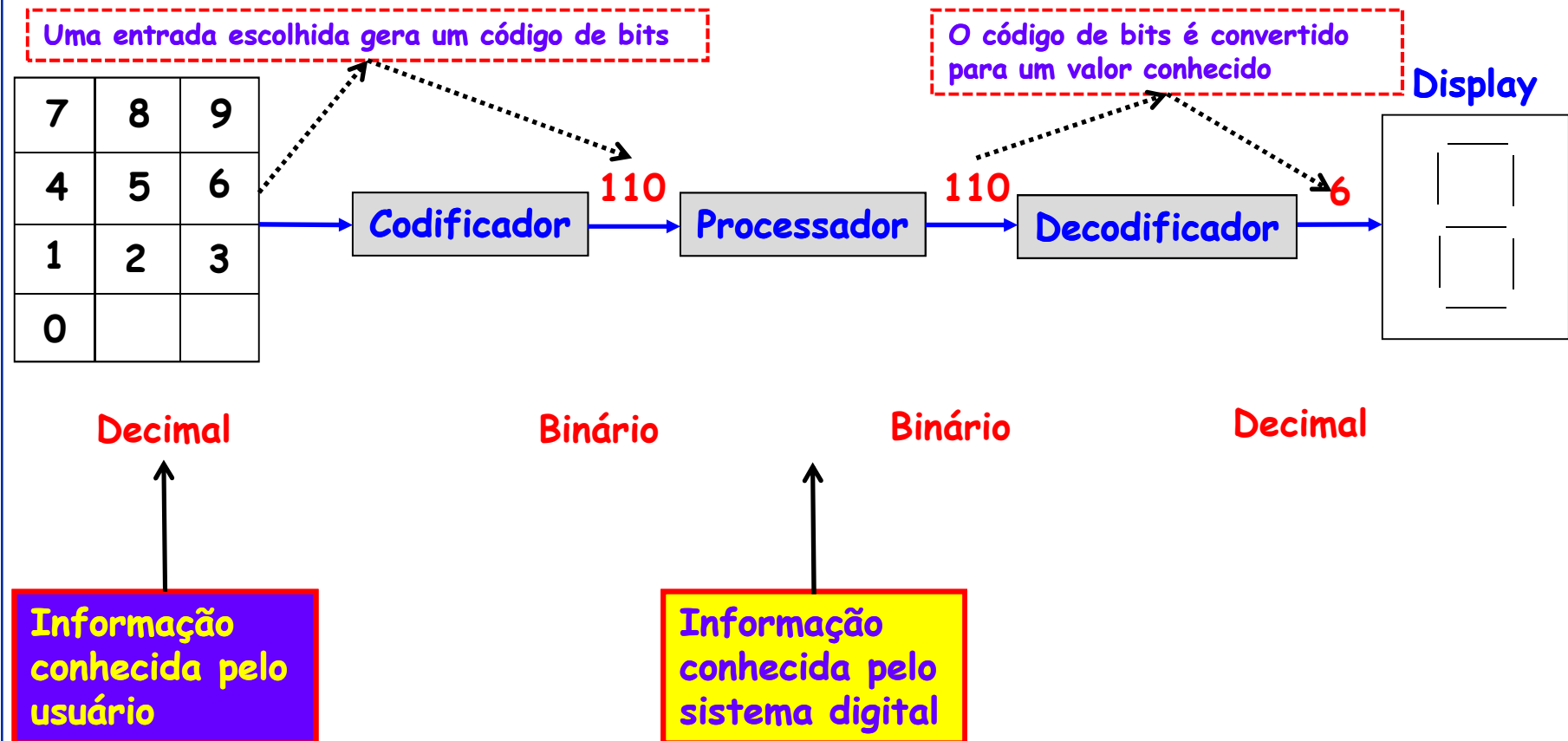
- **Revisão**
 - Circuitos Combinacionais - Circuitos de Apoio
 - Codificadores/Decodificadores
- **Circuitos Combinacionais - Circuitos de Apoio**
 - Multiplexadores/Demultiplexadores
 - Gerador de Paridade/Verificador de Paridade
- **Aritmética Computacional**
 - Somador

Revisão

Circuitos Combinacionais - Circuitos de Apoio

Circuitos de Apoio

Exemplo de Aplicação de Codificação e Decodificação



Circuitos de Apoio

Exemplo de Aplicação de Decodificação

Decodificador de Endereços para Memória



Decodificador de Endereços

Entradas		Saídas			
S_1	S_0	D_0	D_1	D_2	D_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1

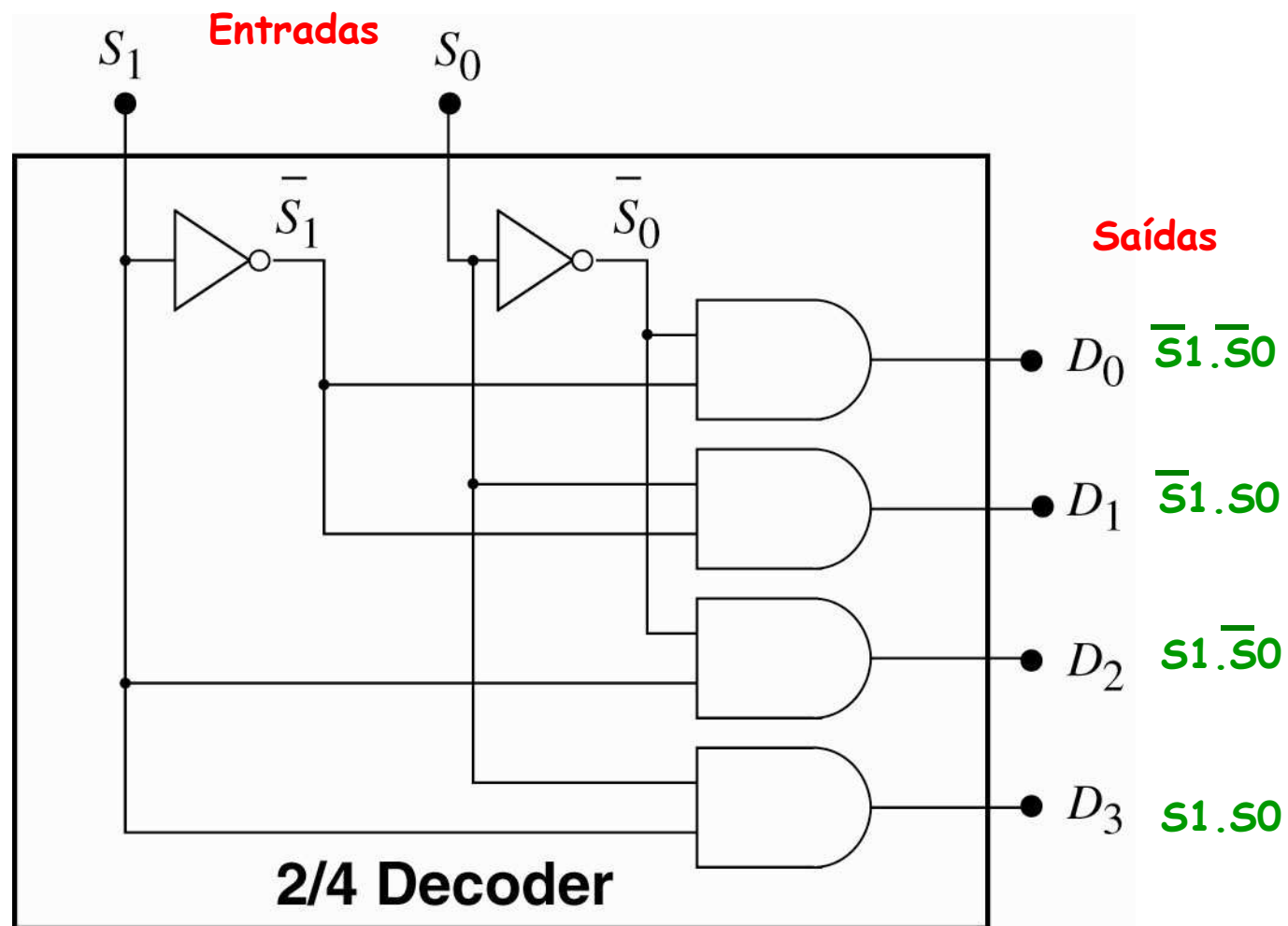
$D_0 = \overline{S_1} \cdot \overline{S_0}$

$D_1 = \overline{S_1} \cdot S_0$

$D_2 = S_1 \cdot \overline{S_0}$

$D_3 = S_1 \cdot S_0$

Decodificador de Endereços



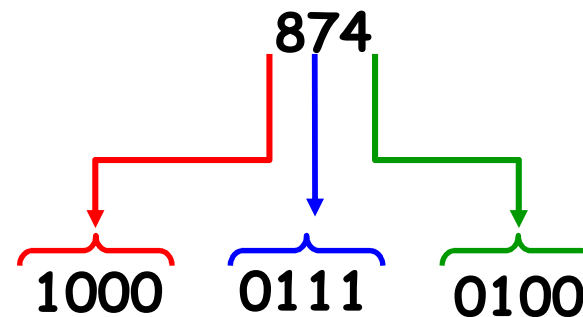
Circuitos de Apoio

Código BCD - Binary Coded Decimal

Decimal Codificado em Binário: Cada Dígito Decimal é representado por seu equivalente binário

Decimal	BCD
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001

Exemplo

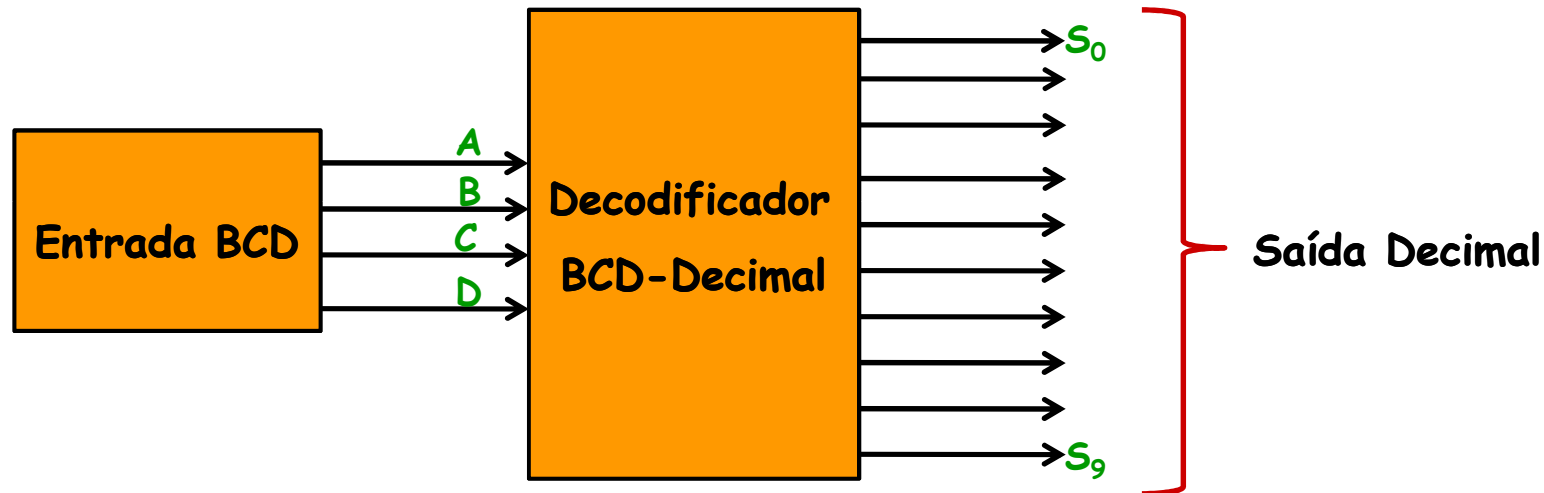


100001110100_{BCD}

Circuitos de Apoio

Código BCD - Binary Coded Decimal

Decimal Codificado em Binário: Cada Dígito Decimal é representado por seu equivalente binário



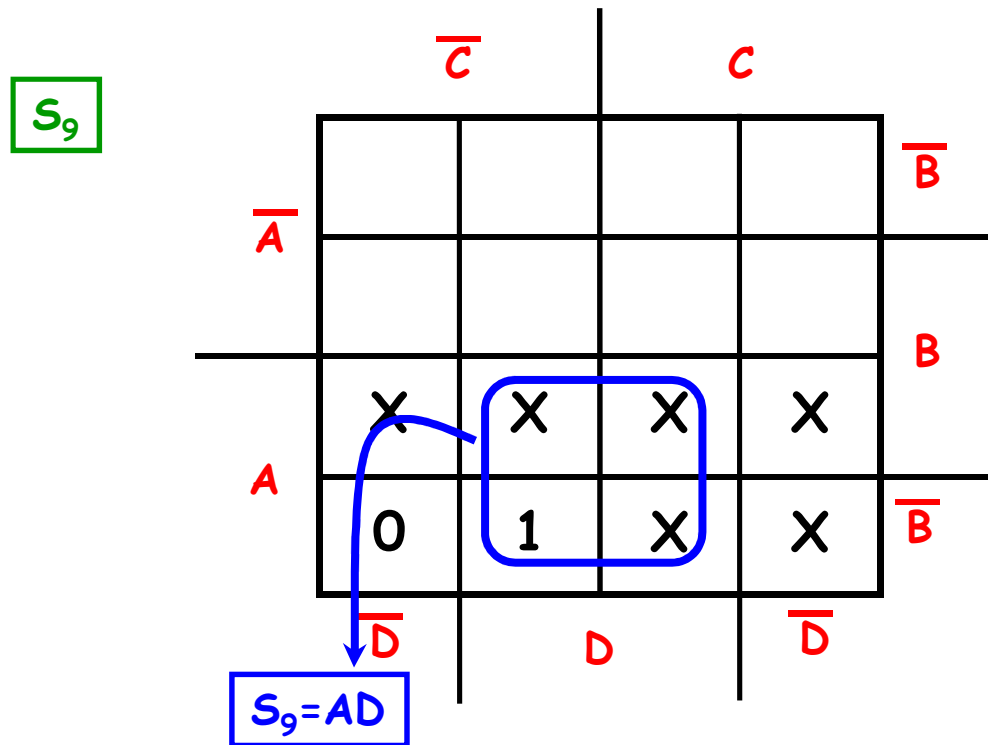
Circuitos de Apoio

Decodificador BCD para Decimal

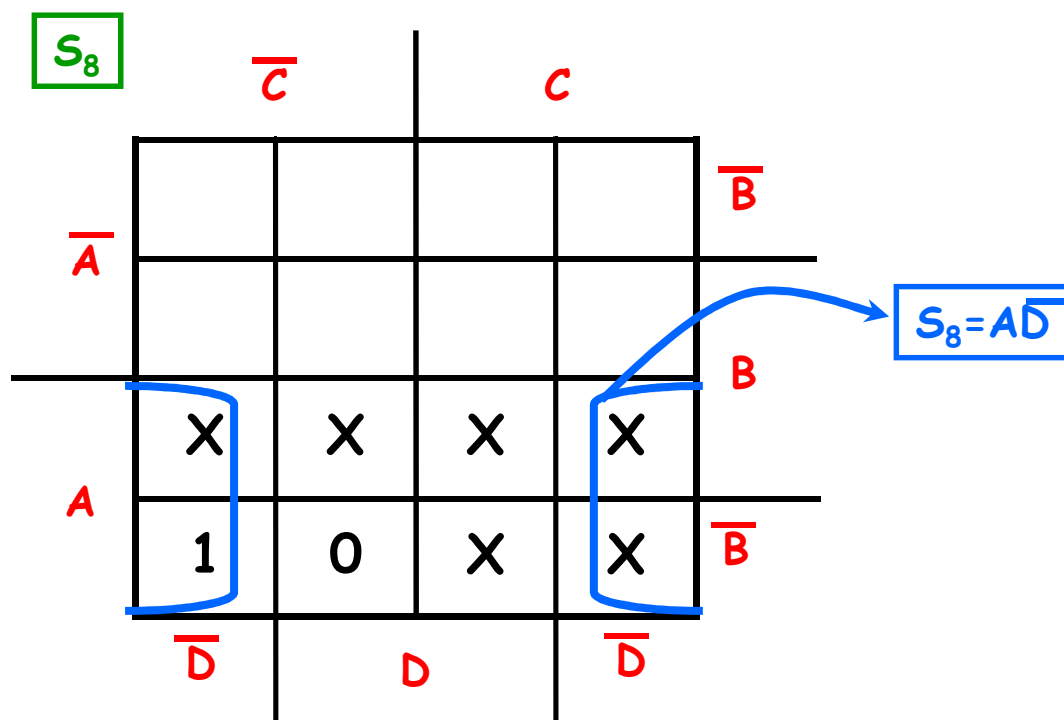
BCD	S ₉	S ₈	S ₇	S ₆	S ₅	S ₄	S ₃	S ₂	S ₁	S ₀
0000	0	0	0	0	0	0	0	0	0	1
0001	0	0	0	0	0	0	0	0	1	0
0010	0	0	0	0	0	0	0	1	0	0
0011	0	0	0	0	0	0	1	0	0	0
0100	0	0	0	0	0	1	0	0	0	0
0101	0	0	0	0	1	0	0	0	0	0
0110	0	0	0	1	0	0	0	0	0	0
0111	0	0	1	0	0	0	0	0	0	0
1000	0	1	0	0	0	0	0	0	0	0
1001	1	0	0	0	0	0	0	0	0	0
...	X	X	X	X	X	X	X	X	X	X
1111	X	X	X	X	X	X	X	X	X	X

Circuitos de Apoio

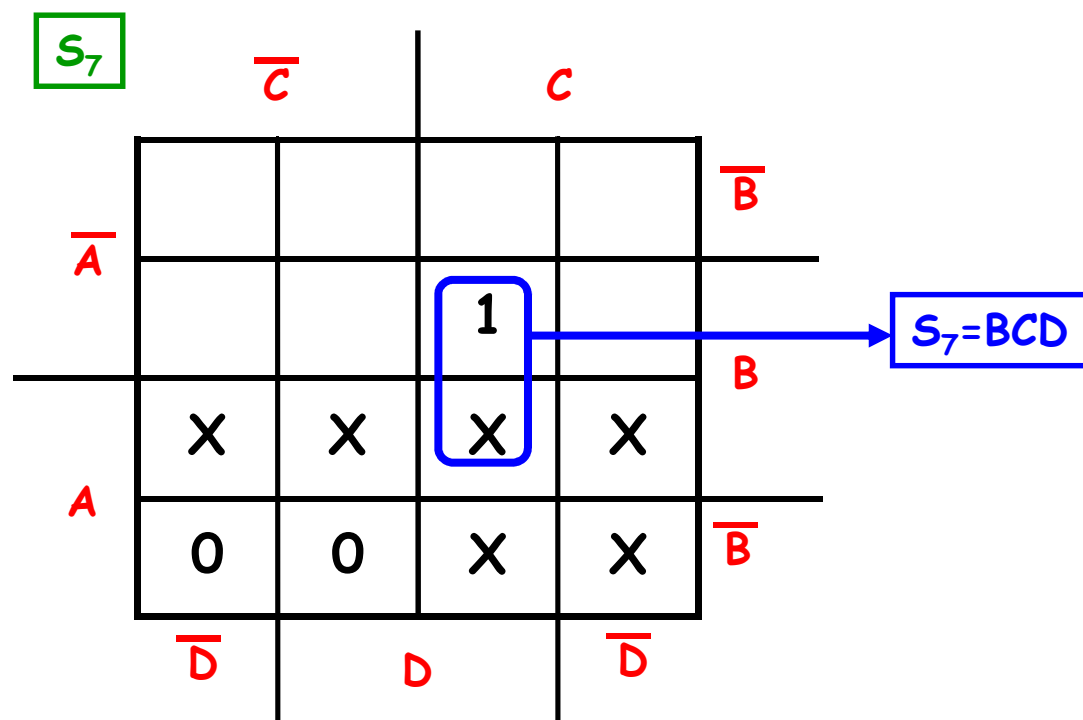
- Para definir o decodificador: simplificar expressões de S_0 a S_9 por Mapa de Karnaugh
- 9 Mapas: um para cada saída



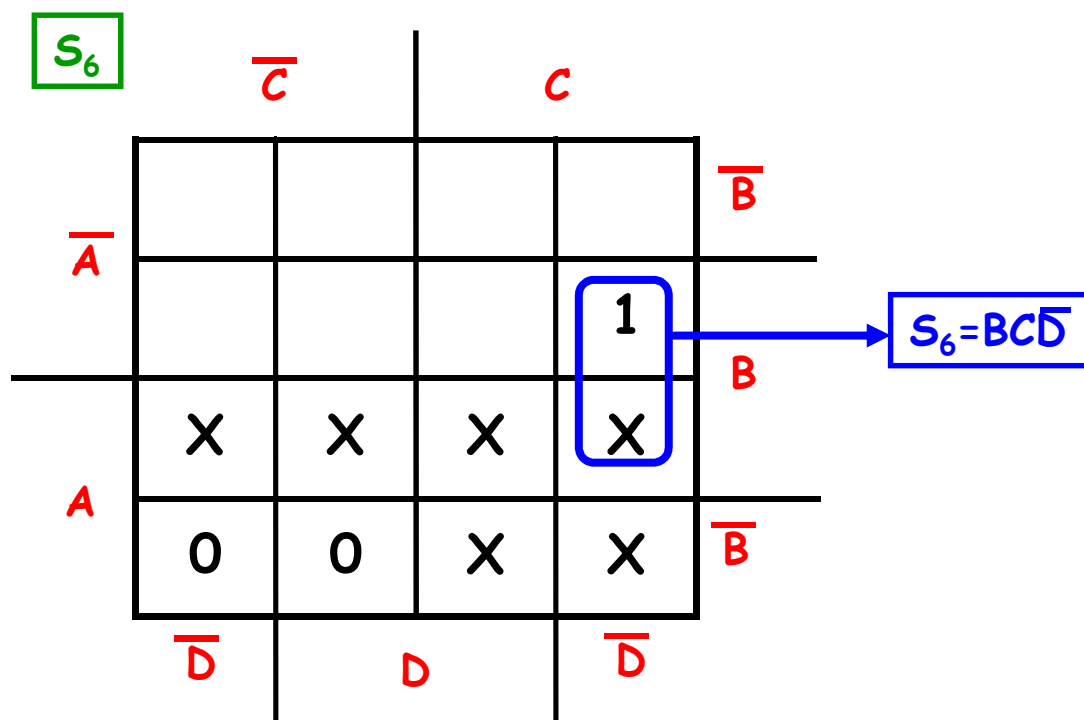
Circuitos de Apoio



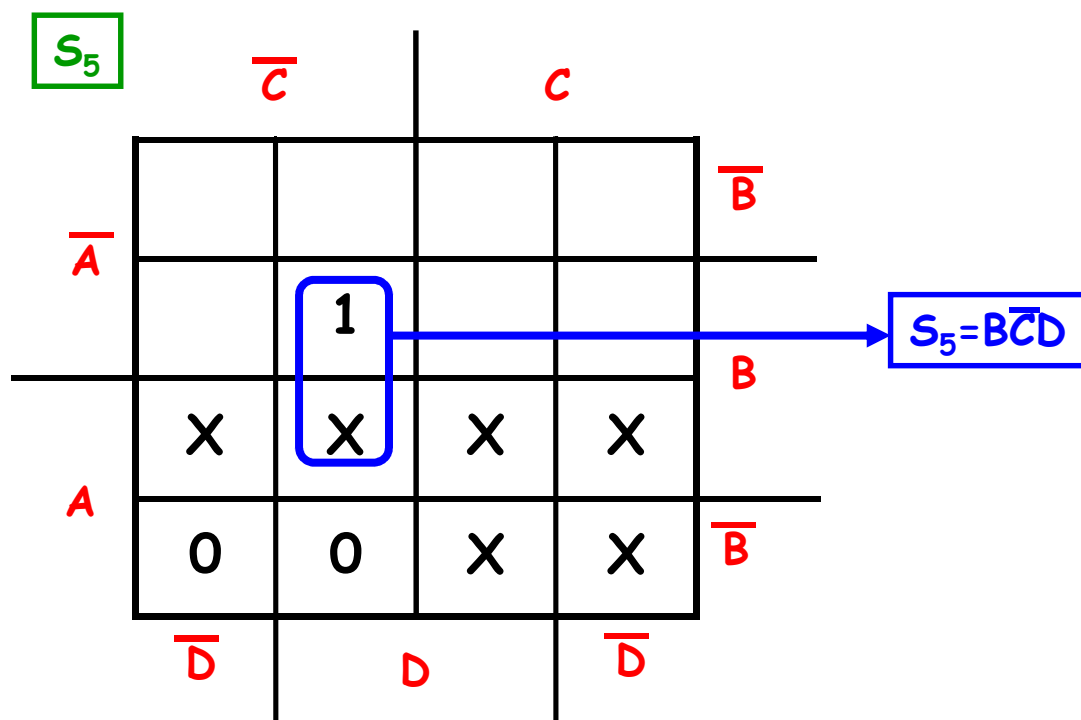
Circuitos de Apoio



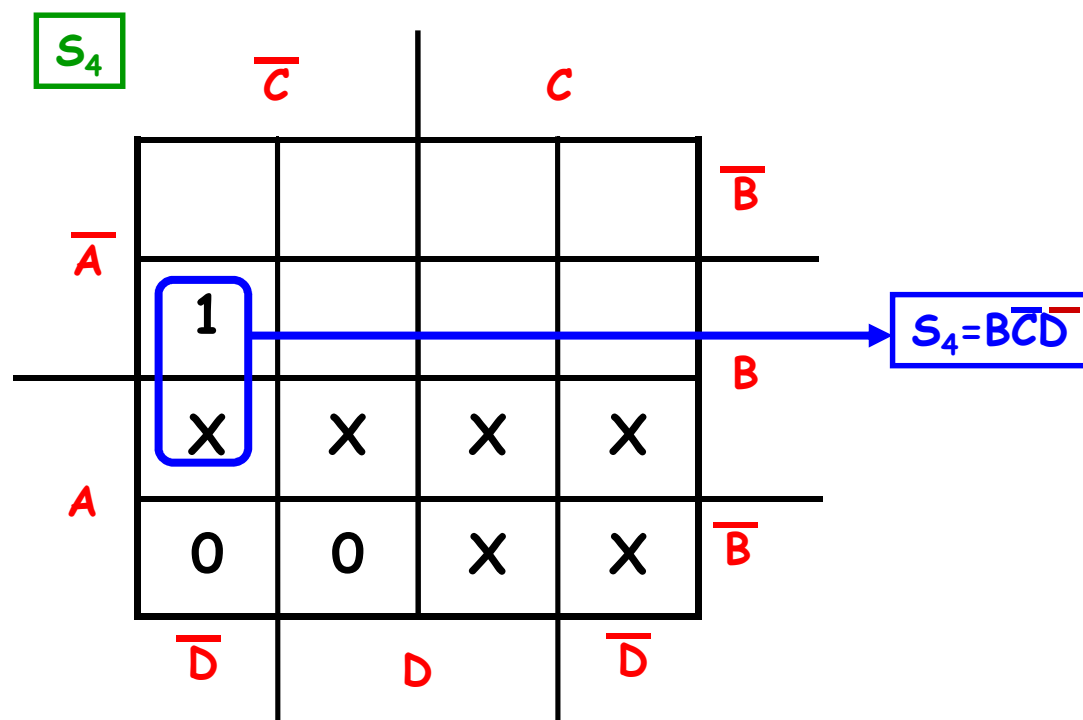
Circuitos de Apoio



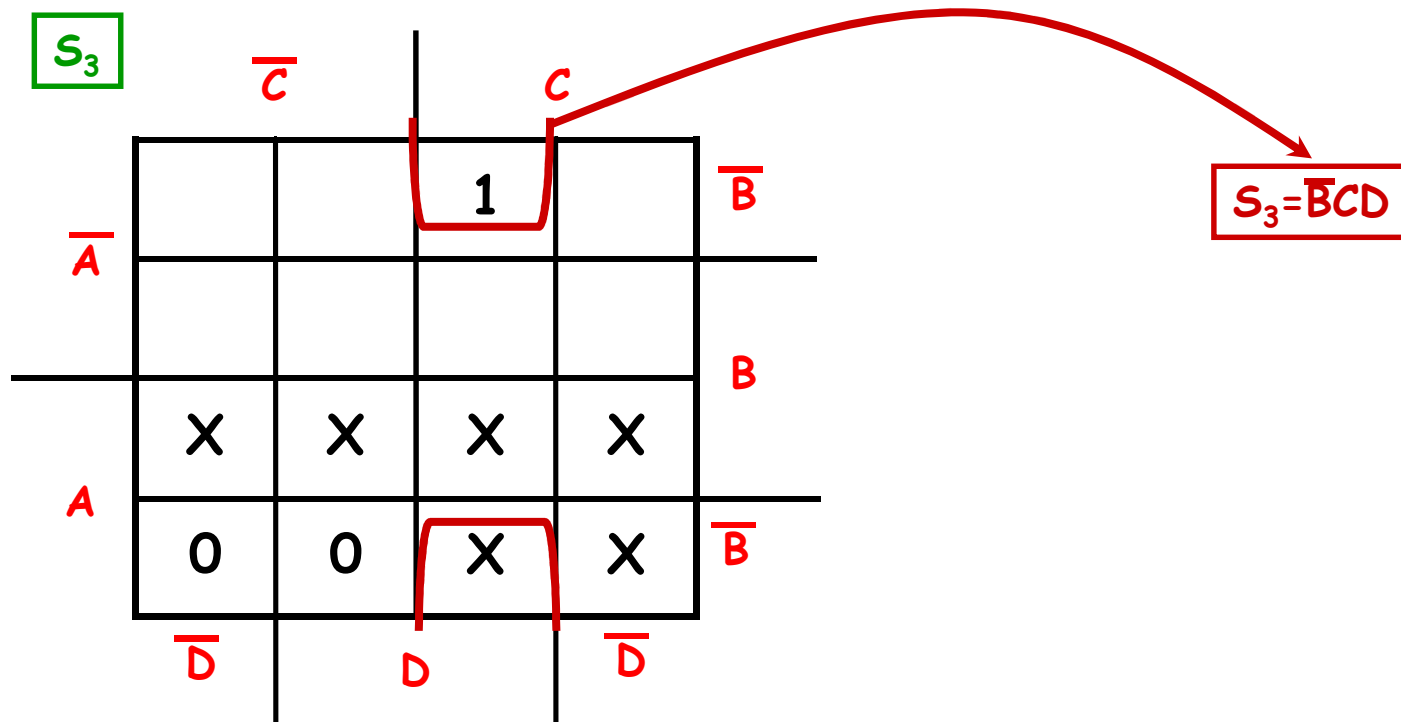
Circuitos de Apoio



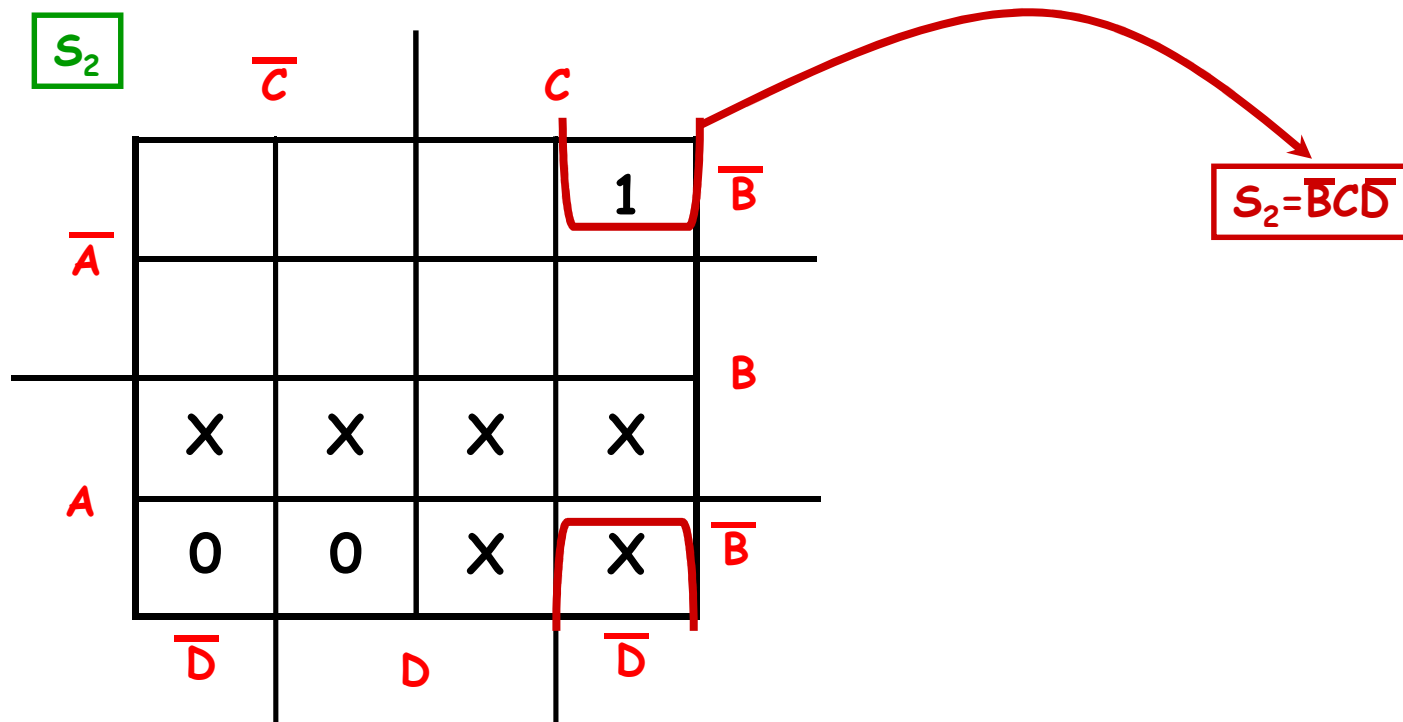
Circuitos de Apoio



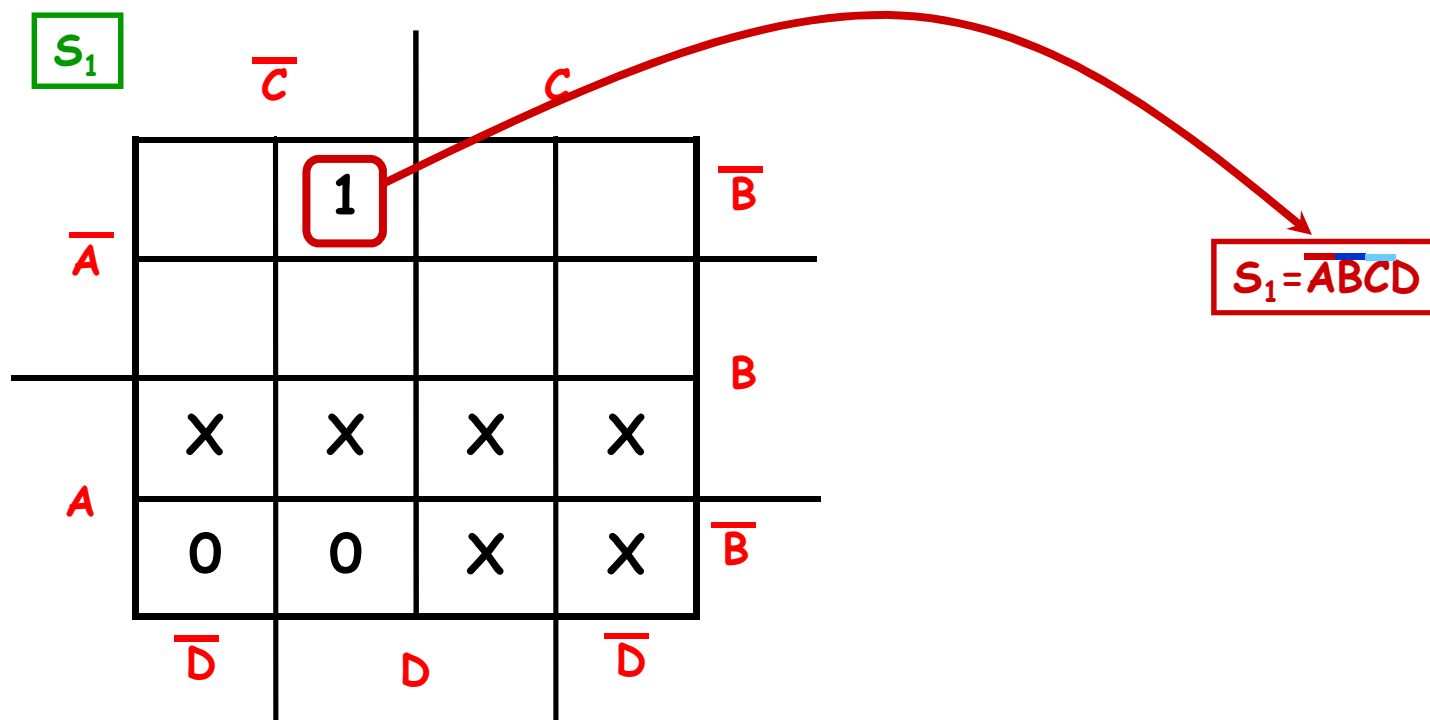
Circuitos de Apoio



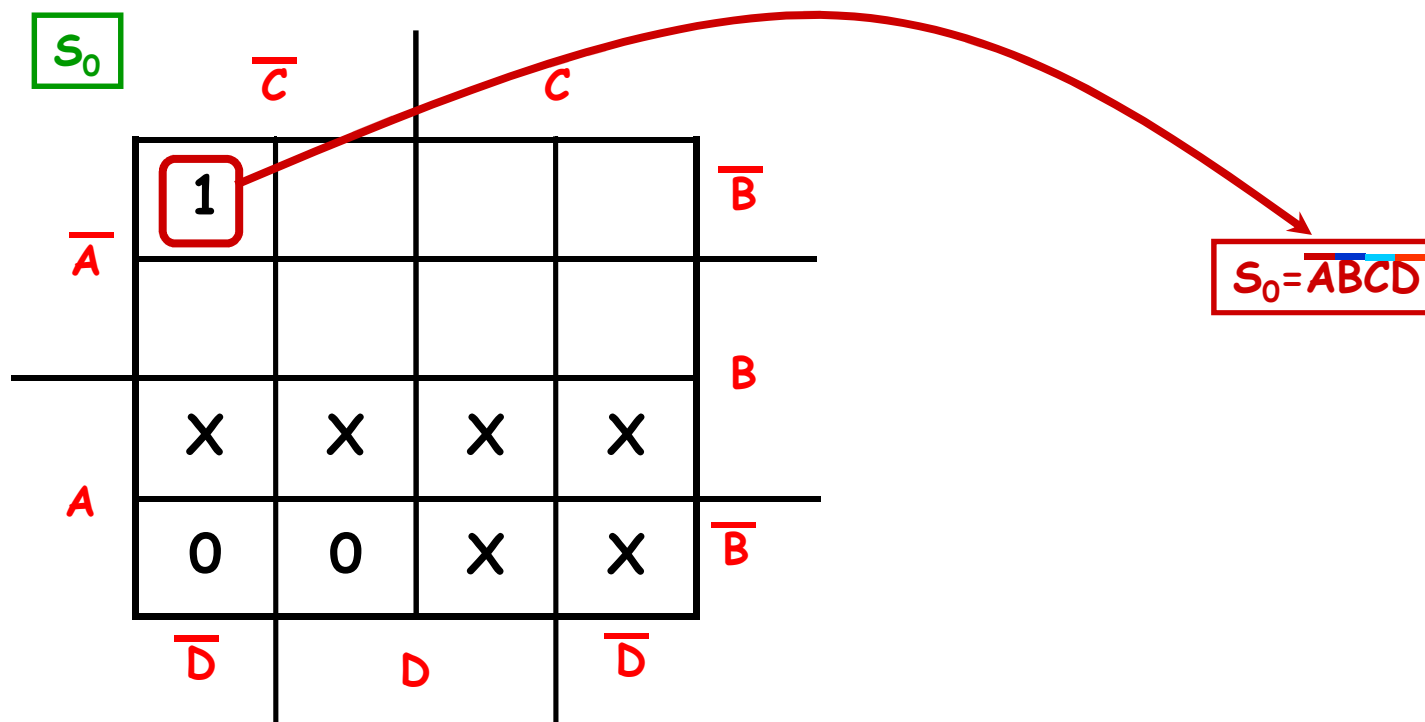
Circuitos de Apoio



Circuitos de Apoio



Circuitos de Apoio



Exercícios

1. Faça o diagrama de portas lógicas do circuito Decodificador BCD-Decimal

$$S_9 = AD$$

$$S_8 = A\overline{D}$$

$$S_7 = BCD$$

$$S_6 = B\overline{C}\overline{D}$$

$$S_5 = B\overline{C}D$$

$$S_4 = B\overline{C}\overline{\overline{D}}$$

$$S_3 = \overline{B}CD$$

$$S_2 = \overline{B}\overline{C}\overline{D}$$

$$S_1 = \overline{A}BCD$$

$$S_0 = \overline{A}\overline{B}\overline{C}\overline{D}$$

Soluções

1. Faça o diagrama de portas lógicas do circuito Decodificador BCD-Decimal

$$S_9 = AD$$

$$S_8 = A\bar{D}$$

$$S_7 = BCD$$

$$S_6 = BC\bar{D}$$

$$S_5 = B\bar{C}D$$

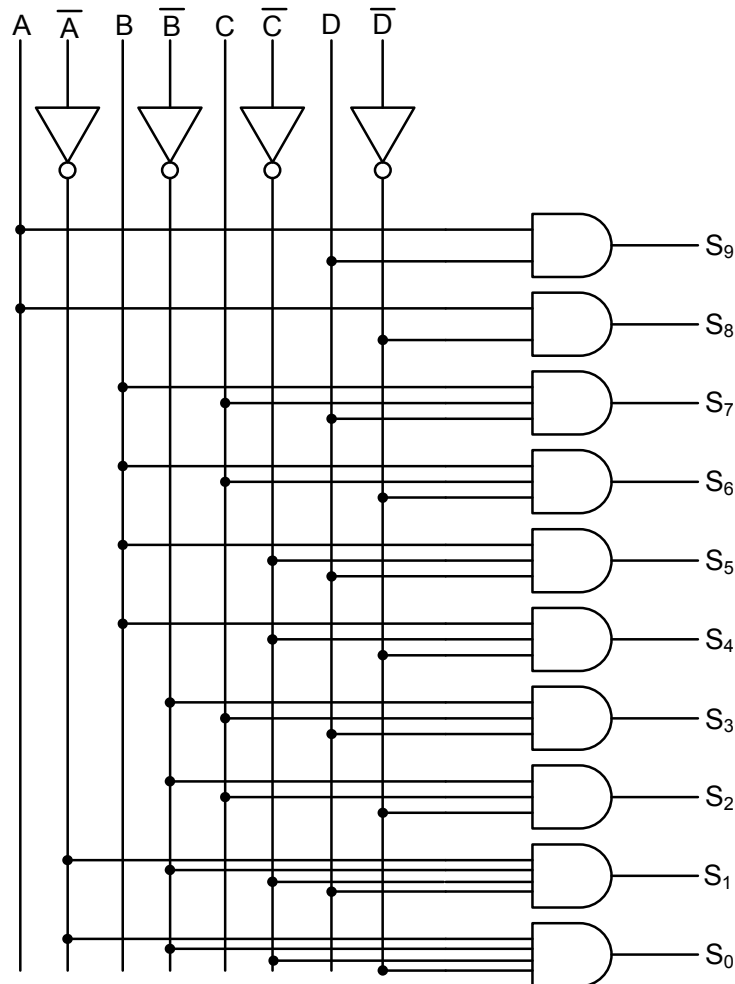
$$S_4 = B\bar{C}\bar{D}$$

$$S_3 = \bar{B}CD$$

$$S_2 = \bar{B}\bar{C}\bar{D}$$

$$S_1 = \bar{A}BCD$$

$$S_0 = \bar{A}\bar{B}\bar{C}\bar{D}$$



Exercícios

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

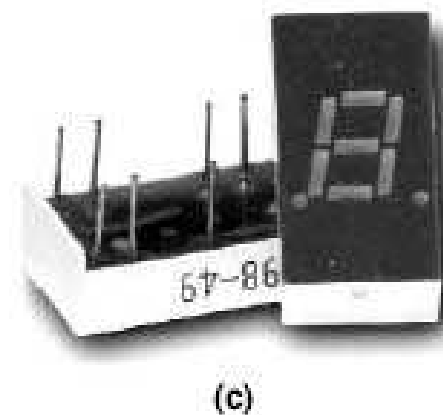
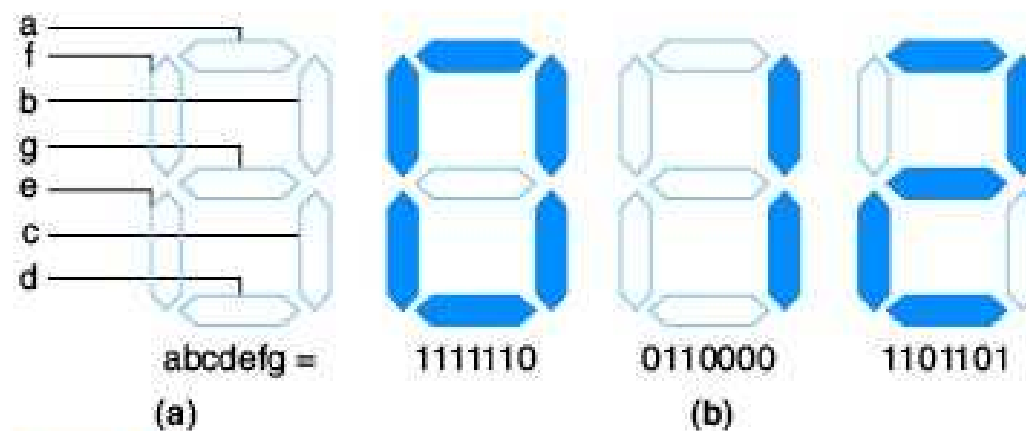
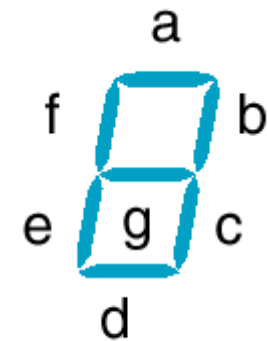
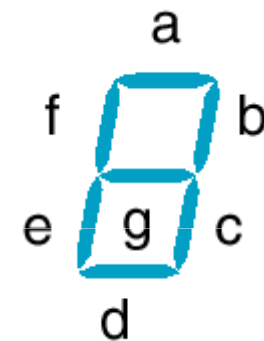


Figure 2.39 Seven-segment display: (a) connections of inputs to segments, (b) input values for numbers 0, 1, and 2, and (c) a pair of real seven-segment display components.

Exercícios

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

BCD	Display	a	b	c	d	e	f	g
0000		1	1	1	1	1	1	0
0001		0	1	1	0	0	0	0
0010		1	1	0	1	1	0	1
0011		1	1	1	1	0	0	1
0100		0	1	1	0	0	1	1
0101		1	0	1	1	0	1	1
0110		1	0	1	1	1	1	1
0111		1	1	1	0	0	0	0
1000		1	1	1	1	1	1	1
1001		1	1	1	1	0	1	1
...								
1111		X	X	X	X	X	X	X



Soluções

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

a

	\overline{C}		C		
	1	0	1	1	\overline{B}
\overline{A}	0	1	1	1	
	X	X	X	X	B
A	1	1	X	X	\overline{B}
	\overline{D}	D	\overline{D}		

$$a = A + C + BD + \overline{B}\overline{D}$$

Soluções

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

\overline{A}	\overline{C}	C	\overline{B}
1	1	1	1
1	0	1	0
X	X	X	X
A	\overline{D}	D	\overline{B}
1	1	X	X

$$b = \overline{B} + \overline{C}D + CD$$

Soluções

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

c

	\overline{c}		c	
	1	1	1	0
\overline{A}	1	1	1	1
	X	X	X	X
A	1	1	X	X
	\overline{D}	D	\overline{D}	
				\overline{B}
				B
				\overline{B}

$$c = B + \overline{C} + D$$

Soluções

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

d

	\overline{c}		c	
\overline{A}	1	0	1	\overline{B}
	0	1	0	1
	X	X	X	X
A	1	1	X	\overline{B}
	\overline{D}	D	\overline{D}	

The Karnaugh map shows the following groupings:

- Red group (A):** Cells (0,0), (0,1), (1,0), (1,1) where A=1.
- Blue group (B):** Cells (0,1), (1,1) where B=1.
- Green group (C):** Cells (0,0), (0,1), (1,0), (1,1) where C=1.
- Orange group (D):** Cells (0,0), (0,1), (1,0), (1,1) where D=1.

$$d = A + \overline{B}\overline{D} + \overline{B}C + C\overline{D} + B\overline{C}D$$

Soluções

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

A Karnaugh map for the 7-segment display output 'e'. The map is a 4x4 grid with variables A, B, C, and D. The top-left cell contains 'e' in a green box. The map is divided into four quadrants by a vertical line (C) and a horizontal line (B). The top-left quadrant is labeled with \overline{C} and \overline{A} . The top-right quadrant is labeled with C and \overline{B} . The bottom-left quadrant is labeled with A and D. The bottom-right quadrant is labeled with \overline{B} and \overline{D} . The map contains the following values: top-left (1, 0, 0, 1), top-right (0, 0, 0, 1), bottom-left (X, X, X, X), and bottom-right (1, 0, X, X). Red lines group the 1s in the top-left and top-right quadrants. An orange line groups the 1s in the top-right and bottom-right quadrants.

\overline{C}	\overline{A}	C	\overline{B}
1	0	0	1
0	0	0	1
X	X	X	X
A	D	\overline{B}	\overline{D}
1	0	X	X

$$e = \overline{B}\overline{D} + C\overline{D}$$

Soluções

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

\overline{f}	\overline{C}	C		
				\overline{B}
\overline{A}	1	0	0	0
	1	1	0	1
	X	X	X	X
A	1	1	X	X
	\overline{D}	D	\overline{D}	\overline{B}

$$f = A + \overline{C}D + BC + B\overline{D}$$

Soluções

2. Considere o display de 7 segmentos mostrado na figura e a tabela abaixo. Projete o decodificador do código BCD para o display de 7 segmentos.

g

	\overline{C}		C	
\overline{A}	0	0	1	1
	1	1	0	1
	X	X	X	X
A	1	1	X	X
	\overline{D}		D	
			\overline{D}	

Diagram illustrating the Karnaugh map for the 7-segment display output 'g'. The map is a 4x4 grid with inputs \overline{A} , A , \overline{C} , C , \overline{D} , D , \overline{B} , and B labeled around the perimeter. The output 'g' is shown in a green box. The map contains 1s and Xs. A green loop highlights the 1s in the first two rows, first two columns. A blue loop highlights the 1s in the first two rows, last two columns. A red loop highlights the 1s in the first two rows, last two columns. A yellow loop highlights the 1s in the first two rows, last two columns. A red line connects the 1s in the first two rows, last two columns to the output 'g'.

$$g = A + \overline{B}\overline{C} + \overline{B}C + C\overline{D}$$

Aula de Hoje

Circuitos Combinacionais Especiais:

- **Multiplexadores/Demultiplexadores**
- **Gerador de Paridade/Verificador de Paridade**

Aritmética Computacional:

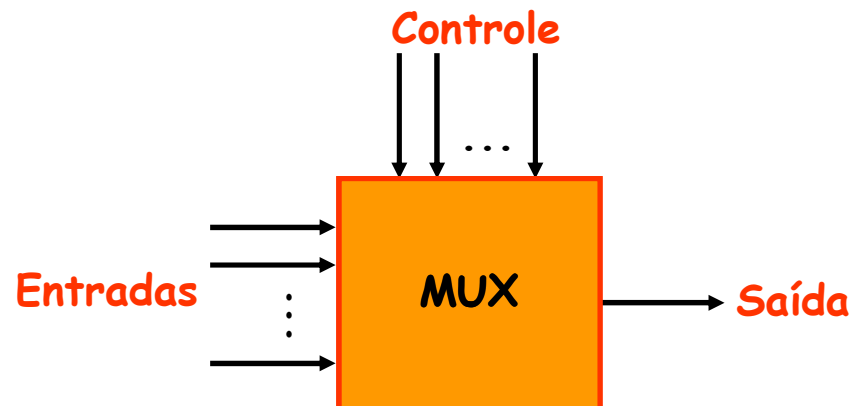
- **Somador**

Circuitos de Apoio

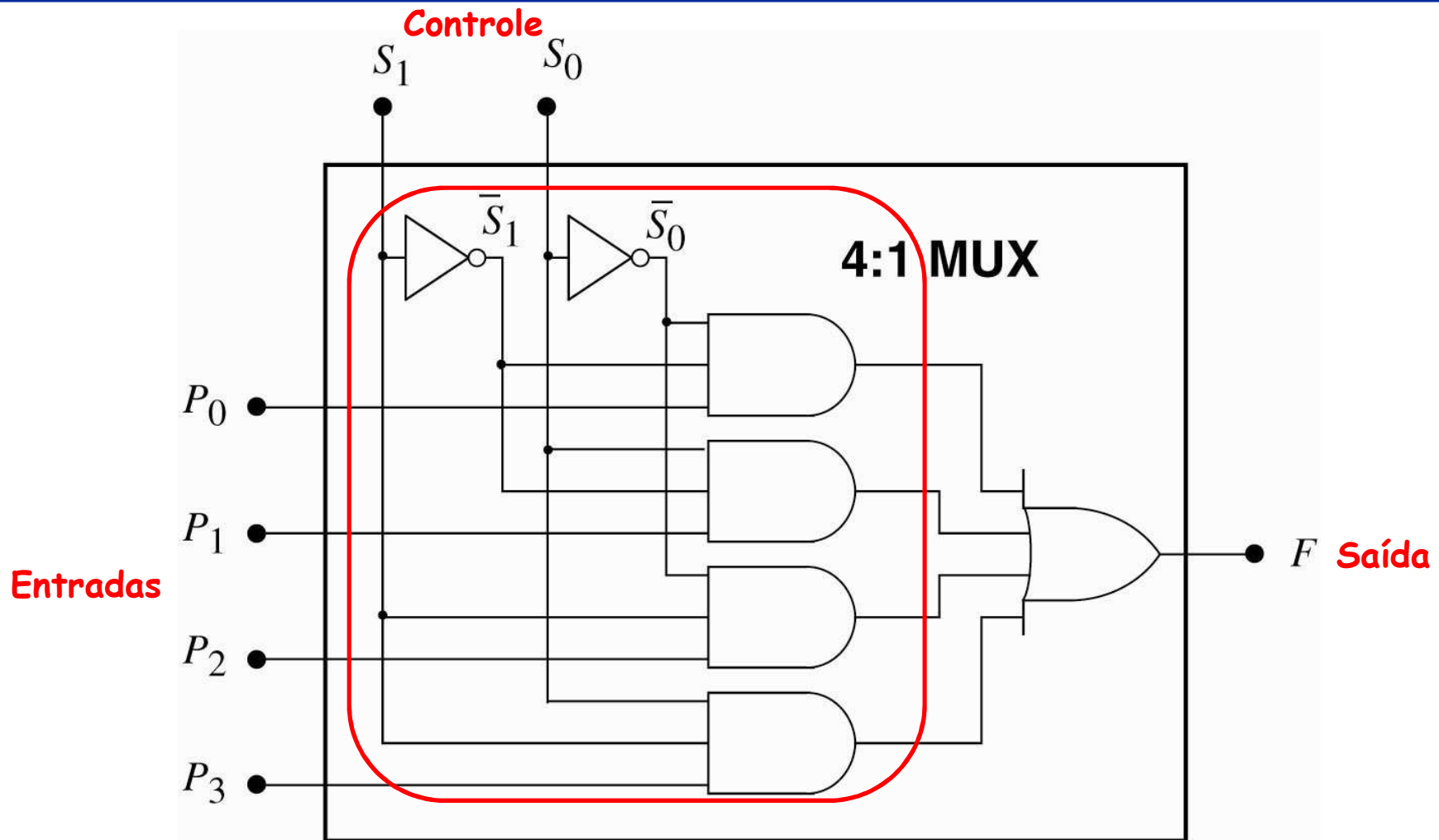
Multiplexador

Multiplexador ou Seletor de Dados: É um circuito lógico que tem diversas entradas e apenas uma saída. MUX seleciona uma única entrada para transmitir para a saída.

Entradas de Controle: permitem selecionar a entrada a ser transmitida.



Exemplo MUX 4x1

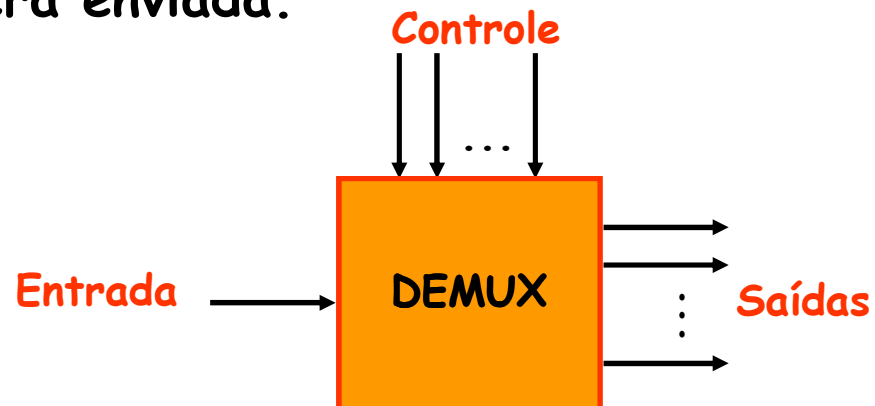


Circuitos de Apoio

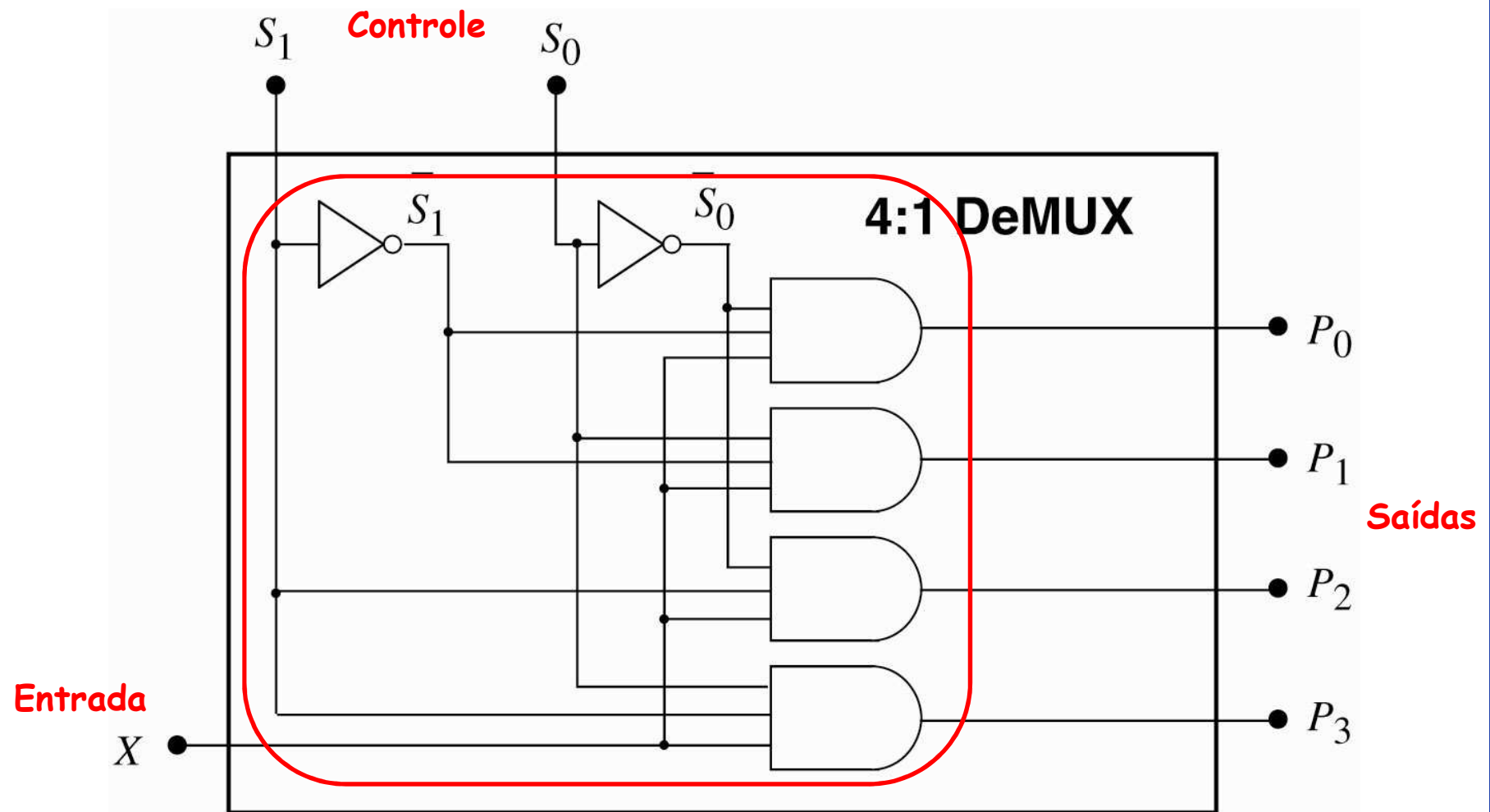
Demultiplexador

Demultiplexador: É um circuito lógico que realiza a função inversa à do MUX. Tem apenas uma única entrada que é enviada para uma de suas saídas.

Entradas de Controle: permitem selecionar para qual das saídas a entrada será enviada.

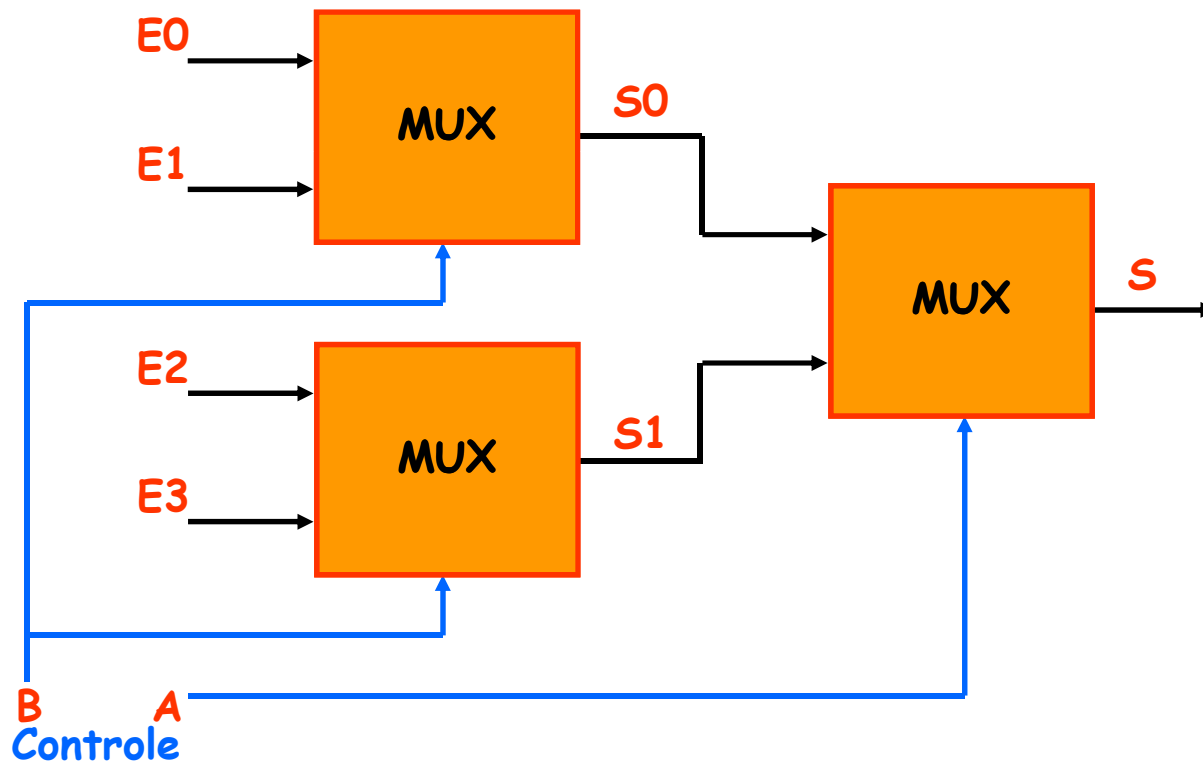


Exemplo DEMUX 1x4



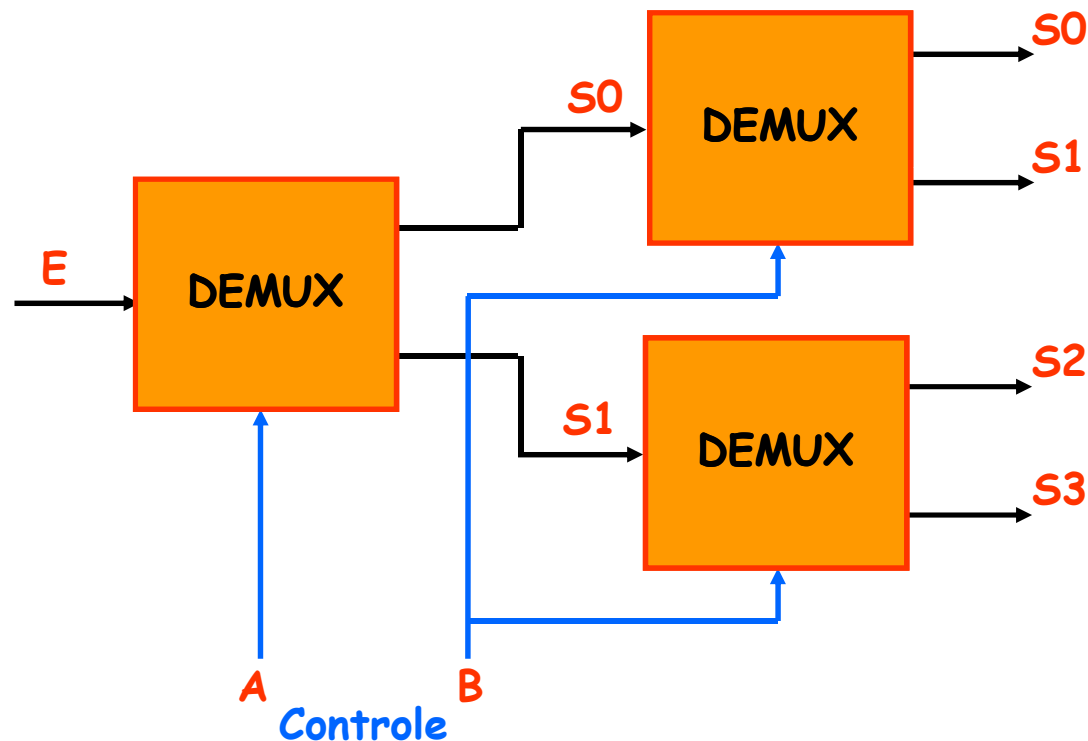
Expansão da Capacidade

- A partir de circuitos multiplexadores de baixa capacidade pode-se formar um MUX de maior capacidade
- Exemplo: projetar um MUX 4x1 a partir de MUXes 2x1



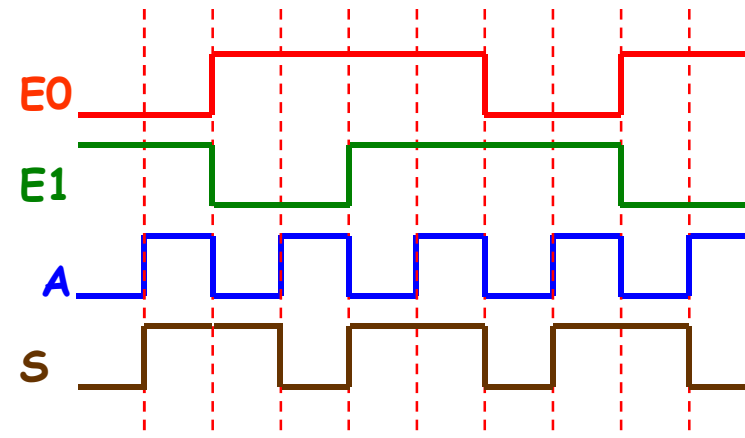
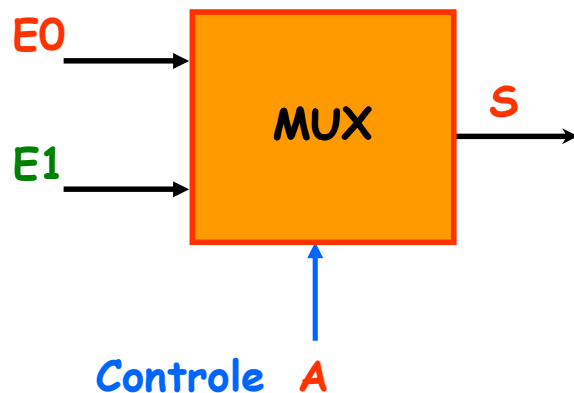
Expansão da Capacidade

- A partir de circuitos demultiplexadores de baixa capacidade pode-se formar um DEMUX de maior capacidade
- Exemplo: projetar um DEMUX 1x4 a partir de DEMUXes 1x2



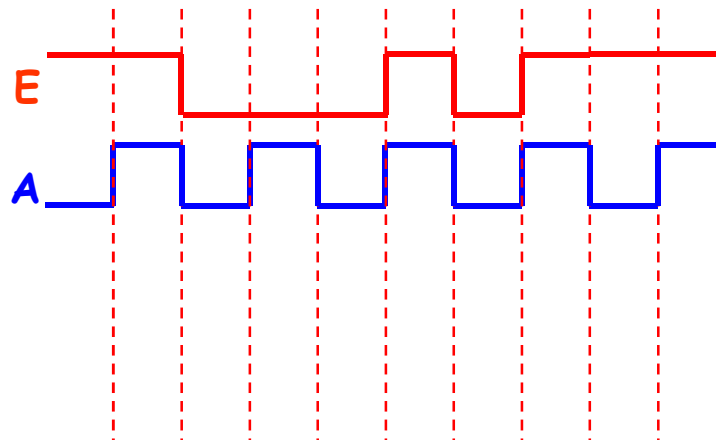
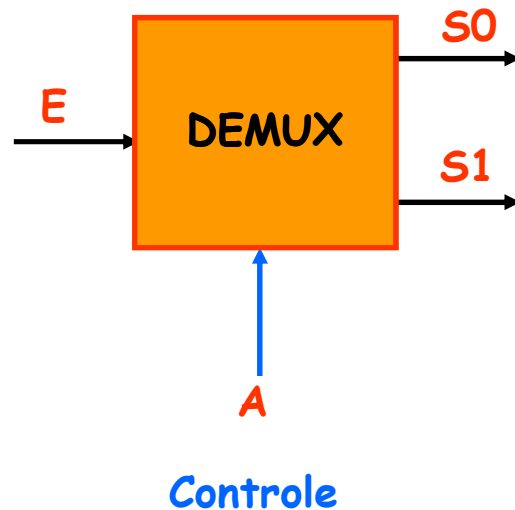
Formas de Onda

Exemplo: A partir dos sinais de entrada e de controle abaixo, desenhe o sinal multiplexado na saída do MUX.



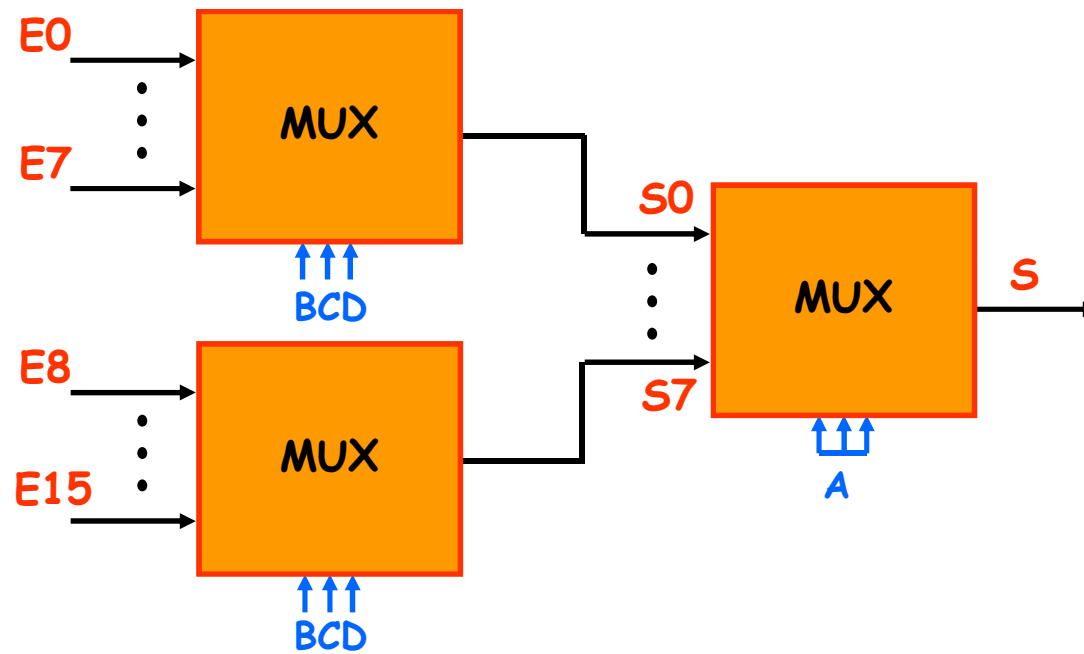
Exercícios

1. Projete um circuito multiplexador de 16x1 utilizando circuitos MUXes 8x1.
2. A partir do DEMUX 1x2 e dos sinais de entrada (E) e de controle (A) desenhe os sinais de saída demultiplexados.



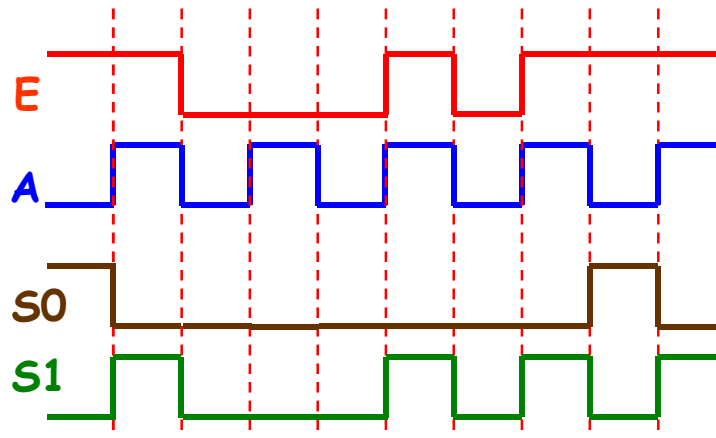
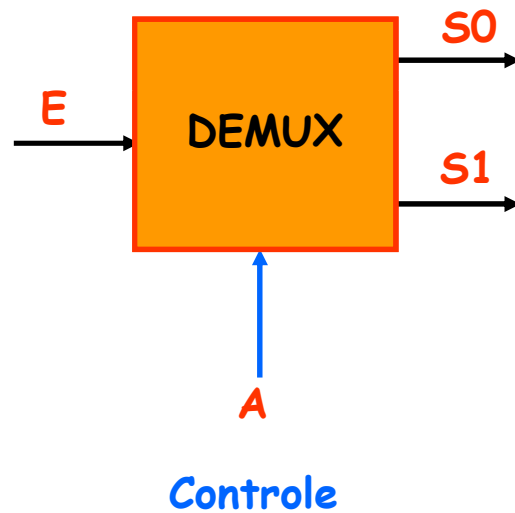
Soluções

1. Projete um circuito multiplexador de 16x1 utilizando circuitos MUXes 8x1.



Soluções

2. A partir do DEMUX 1x2 e dos sinais de entrada (E) e de controle (A) desenhe os sinais de saída demultiplexados.



Circuitos de Apoio

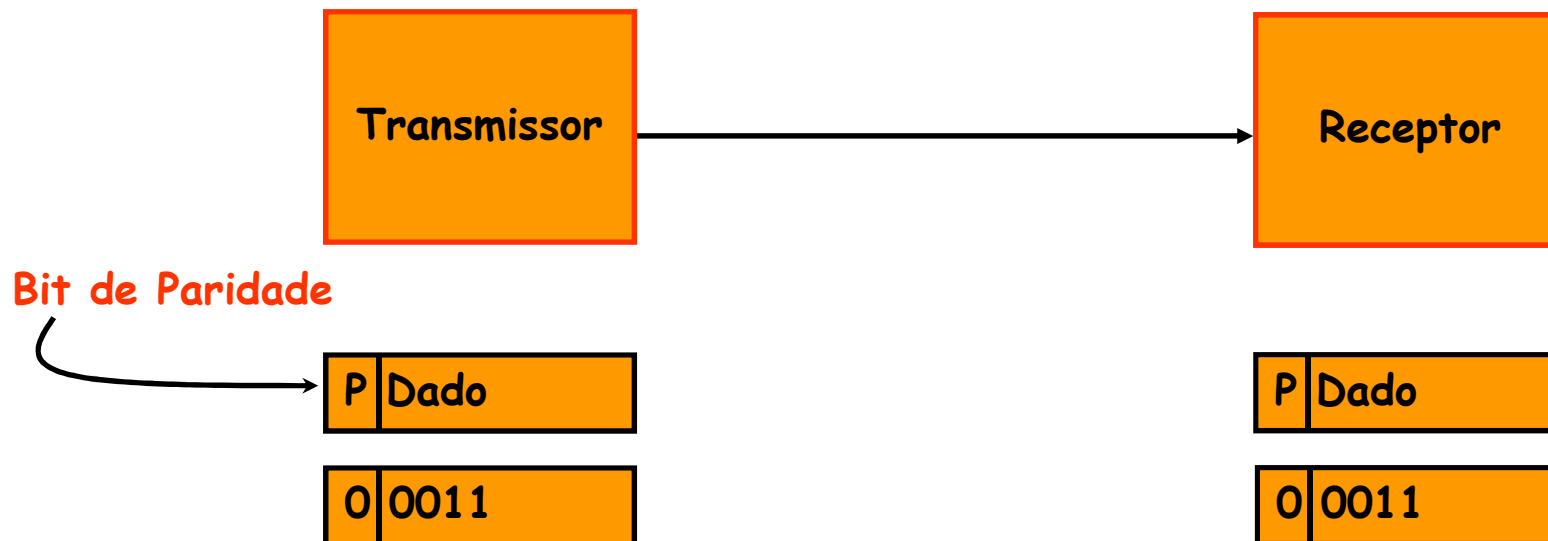
Paridade

- Paridade: indica se a quantidade de dígitos "1" num número binário é par ou ímpar
- Paridade Par: indica que a quantidade de dígitos "1" do número binário é par
- Paridade Ímpar: indica que a quantidade de dígitos "1" do número binário é ímpar
- Exemplo: 11001100 tem 4 dígitos "1" \Rightarrow paridade Par
11101100 tem 5 dígitos "1" \Rightarrow paridade Ímpar

Circuitos de Apoio

Paridade

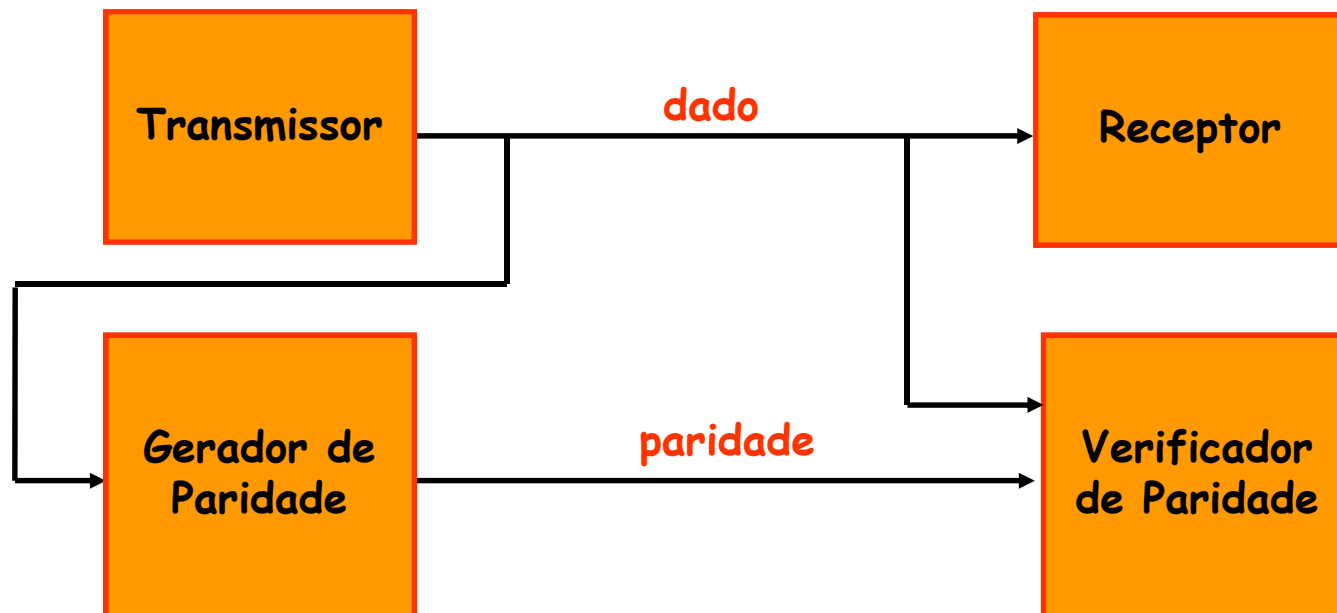
Aplicação: Detecção de erro na transmissão de dados



Circuitos de Apoio

Gerador e Verificador de Paridade

Gera a paridade do dado a ser transmitido e verifica a paridade na recepção do dado

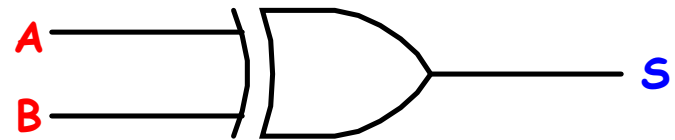


Circuitos de Apoio

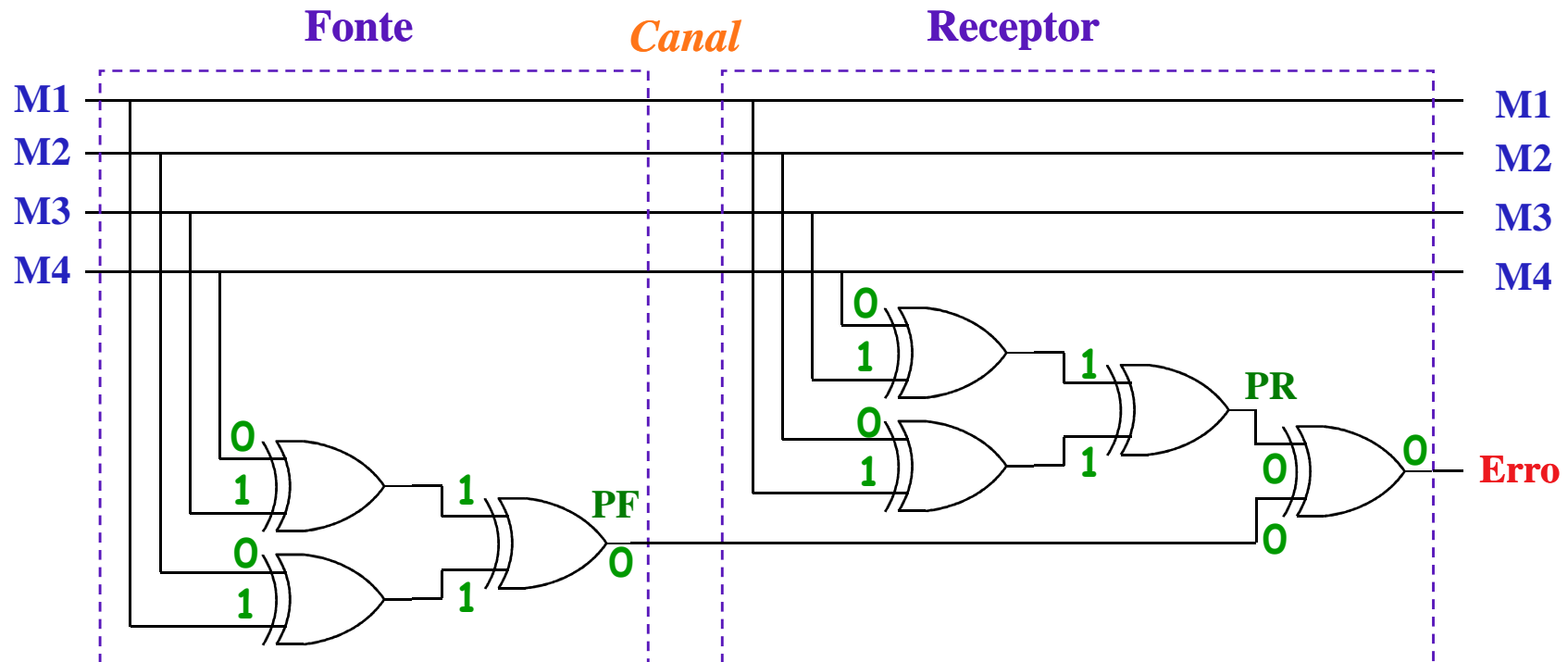
TV da Porta XOR

Entradas		Saída
A	B	S
0	0	0
0	1	1
1	0	1
1	1	0

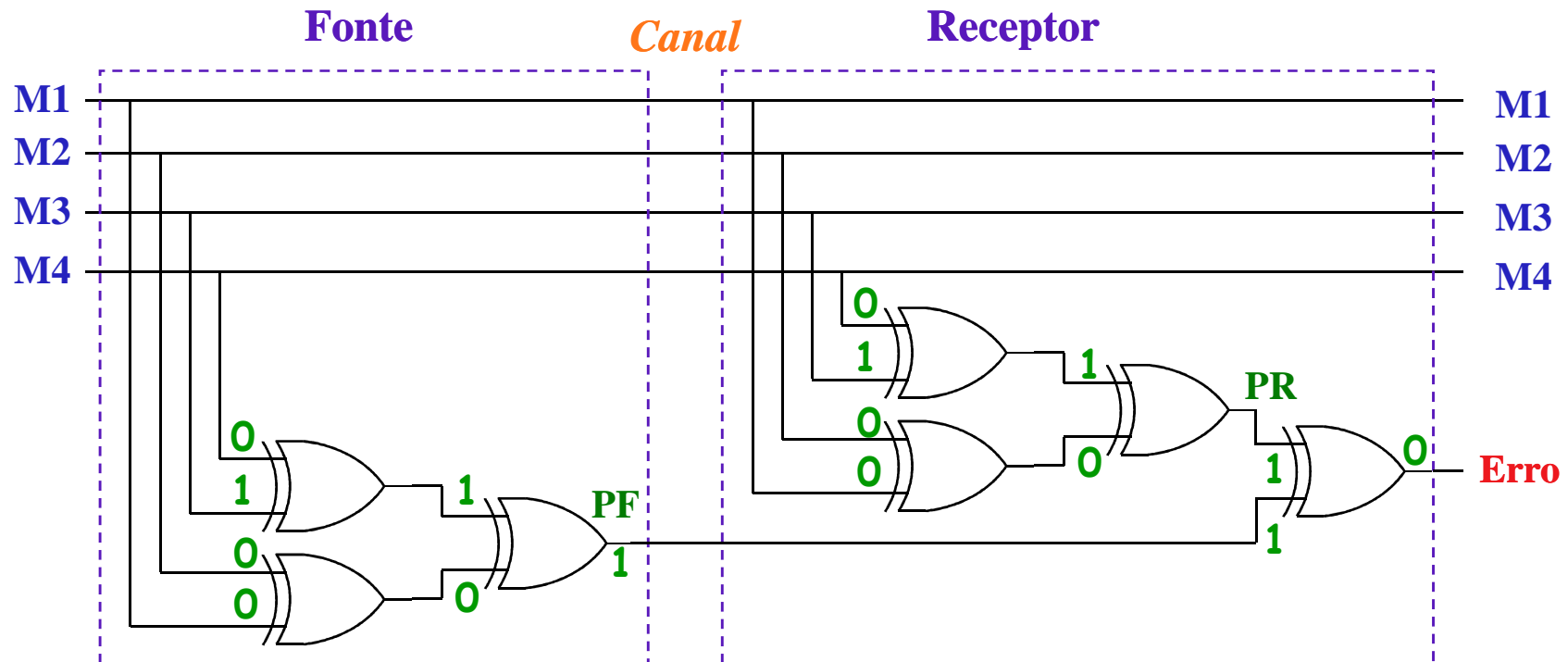
Símbolo da Porta XOR



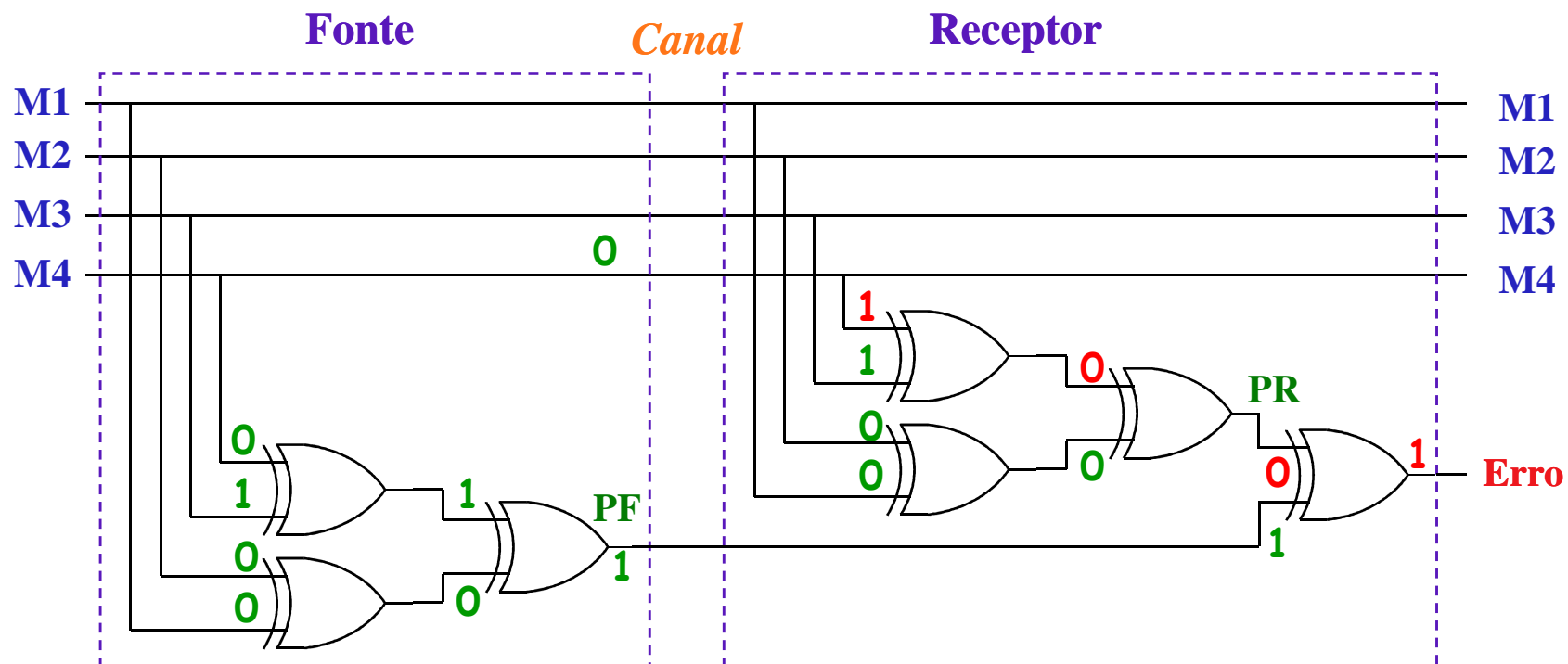
Gerador-Verificador de Paridade Par



Gerador-Verificador de Paridade Par



Gerador-Verificador de Paridade Par



Aritmética Computacional

Aritmética Computacional



Aritmética Computacional

Aritmética Computacional

Circuitos Aritméticos: circuitos utilizados para construir a ULA (Unidade Lógica e Aritmética)

Adição

Exemplo de adição em decimal (dígitos de 0 a 9):

3 7 6	3 7 6	3 7 6	3 7 6
+ 4 6 1	+ 4 6 1	+ 4 6 1	+ 4 6 1
<hr/>	<hr/>	<hr/>	<hr/>
	7	3 7	8 3 7

The diagram shows four addition problems. The first three are 376 + 461. The first has a red underline under the bottom row. The second has a red underline under the bottom row and a red '7' below it. The third has a red underline under the bottom row, a red '3 7' below it, and a red curved arrow pointing from the '1' in the units column to the '3' in the tens column. The fourth has a red underline under the bottom row and a red '8 3 7' below it.

Cada posição só pode representar um dígito, por isso, gera um carry (vai um)

Aritmética Computacional

Aritmética Computacional

Adição em Binário :

Exemplo

a) $\begin{array}{r} 0 \\ + 0 \\ \hline 0 \end{array}$ b) $\begin{array}{r} 0 \\ + 1 \\ \hline 1 \end{array}$ c) $\begin{array}{r} 1 \\ + 0 \\ \hline 1 \end{array}$ d) $\begin{array}{r} 1 \\ + 1 \\ \hline 10 \end{array}$

Cada posição só pode representar um dígito, por isso, gera um carry

Diagram illustrating the binary addition of 10101_2 and 00111_2 using a ripple-carry method:

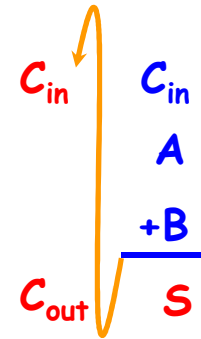
The addition is performed from right to left, with carries propagating to the left:

- Step 1: $10101 + 00111 = 10101$ (Carry out = 0)
- Step 2: $10101 + 00111 = 10101$ (Carry in = 1, Carry out = 0)
- Step 3: $10101 + 00111 = 10101$ (Carry in = 1, Carry out = 0)
- Step 4: $10101 + 00111 = 10101$ (Carry in = 1, Carry out = 1)
- Step 5: $10101 + 00111 = 10101$ (Carry in = 0, Carry out = 1)
- Step 6: $10101 + 00111 = 10101$ (Carry in = 0, Carry out = 0)

The final result is 11100_2 , labeled "Soma".

Aritmética Computacional

Entradas			Saídas	
A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Aritmética Computacional

Entradas			Saídas	
A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$\overline{A} \overline{B} C_{in}$$

$$\overline{A} B \overline{C_{in}}$$

$$A \overline{B} \overline{C_{in}}$$

$$A B C_{in}$$

$$S = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C_{in}} + A \overline{B} \overline{C_{in}} + A B C_{in}$$

Aritmética Computacional

Entradas			Saídas	
A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$\bar{A} B C_{in}$

$A \bar{B} C_{in}$

$A B \bar{C}_{in}$

$A B C_{in}$

$$C_{out} = \bar{A} B C_{in} + A \bar{B} C_{in} + A B \bar{C}_{in} + A B C_{in}$$

Exercício

1. Simplifique as expressões de S e C_{out}
2. Desenhe o circuito para S e C_{out}

$$S = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C_{in}} + A \overline{B} \overline{C_{in}} + A B C_{in}$$

$$C_{out} = \overline{A} B C_{in} + A \overline{B} C_{in} + A B \overline{C_{in}} + A B C_{in}$$

Solução

Aritmética Computacional

Simplificando as expressões

$$S = \overline{A} \overline{B} C_{in} + \overline{A} B \overline{C_{in}} + A \overline{B} \overline{C_{in}} + A B C_{in}$$

$$S = \overline{A} (\overline{B} C_{in} + B \overline{C_{in}}) + A (\overline{B} \overline{C_{in}} + B C_{in}) \quad \leftarrow A \text{ e } \overline{A} \text{ em evidência}$$

$$\text{Como } B \oplus C_{in} = \overline{B} C_{in} + B \overline{C_{in}} \quad \text{e} \quad B \odot C_{in} = \overline{B} \overline{C_{in}} + B C_{in}$$

$$S = \overline{A} (B \oplus C_{in}) + A (B \odot C_{in})$$

$$\text{Fazendo } X = B \oplus C_{in} \text{ e } \overline{X} = B \odot C_{in}$$

$$S = \overline{A} X + A \overline{X}$$

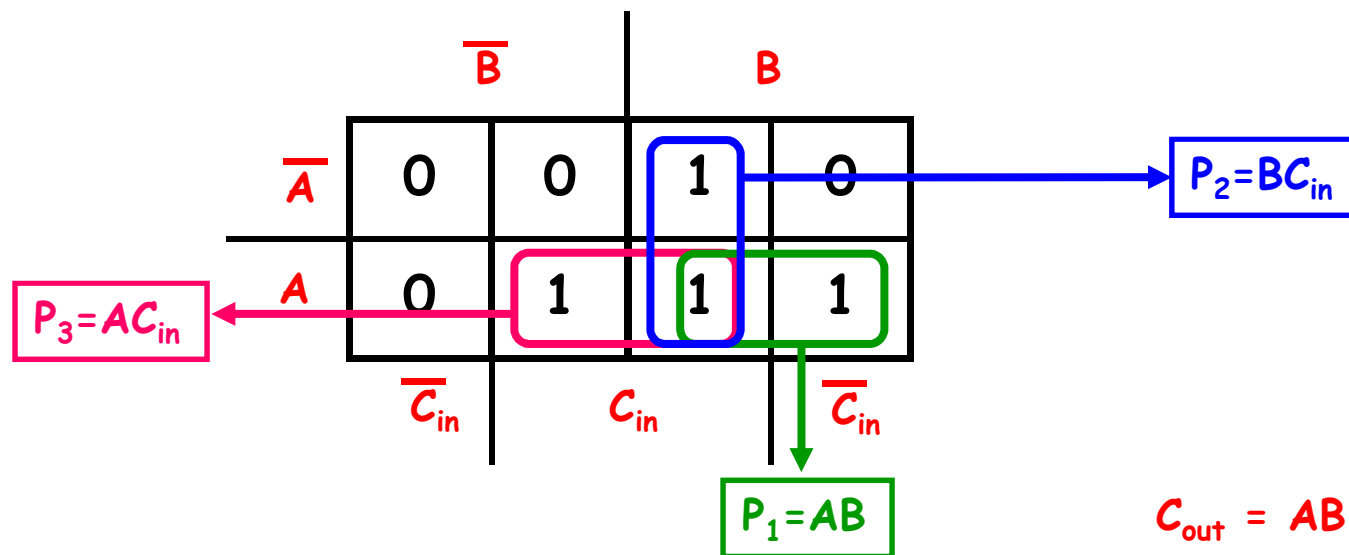
$$S = A \oplus X$$

$$S = A \oplus B \oplus C_{in}$$

Aritmética Computacional

Simplificando as expressões

$$C_{out} = \bar{A} B C_{in} + A \bar{B} C_{in} + A B \bar{C}_{in} + A B C_{in}$$



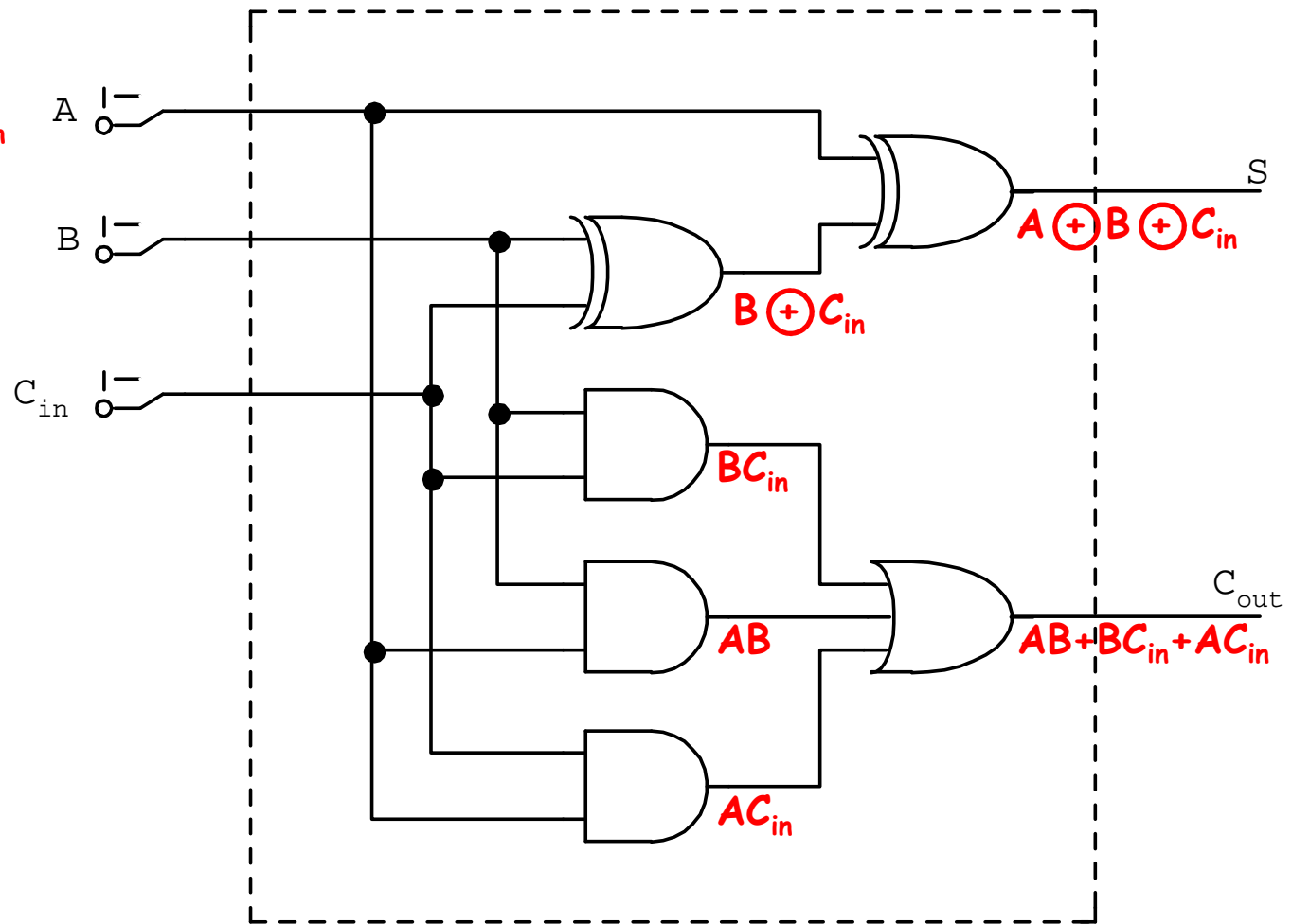
$$C_{out} = AB + BC_{in} + AC_{in}$$

Aritmética Computacional

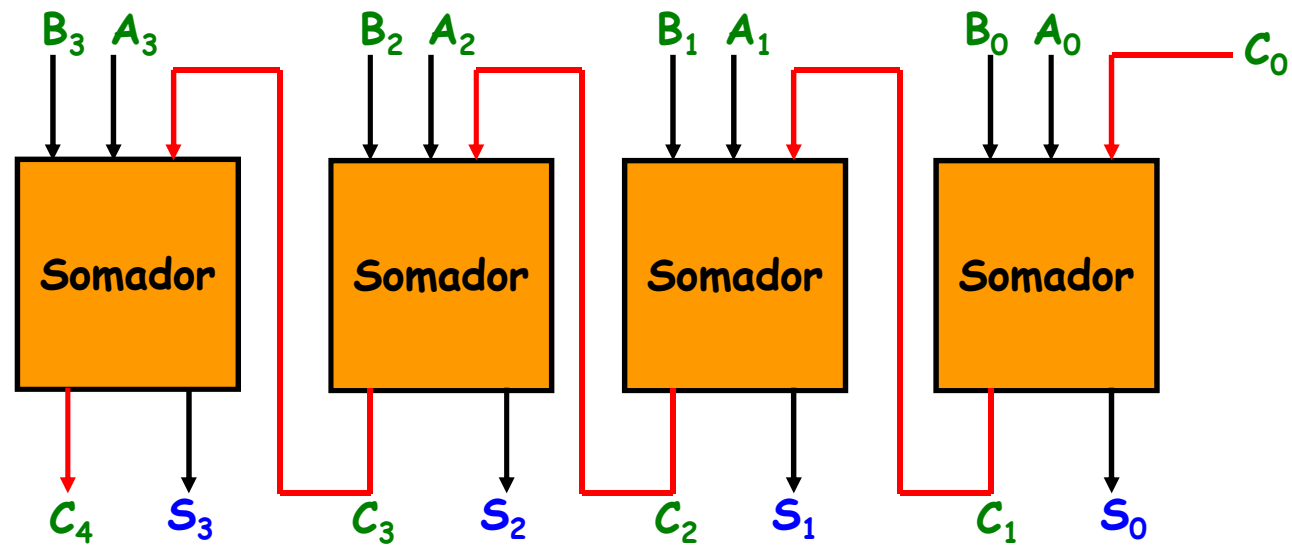
Circuito Somador

$$S = A \oplus B \oplus C_{in}$$

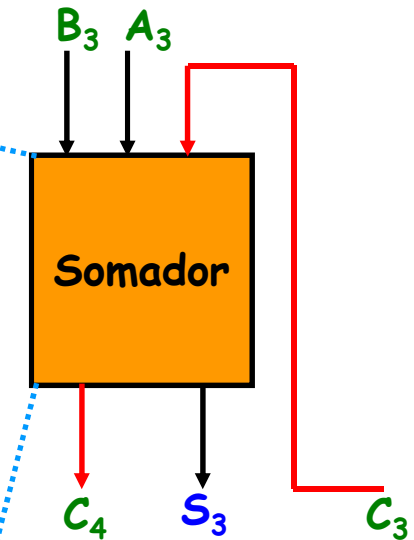
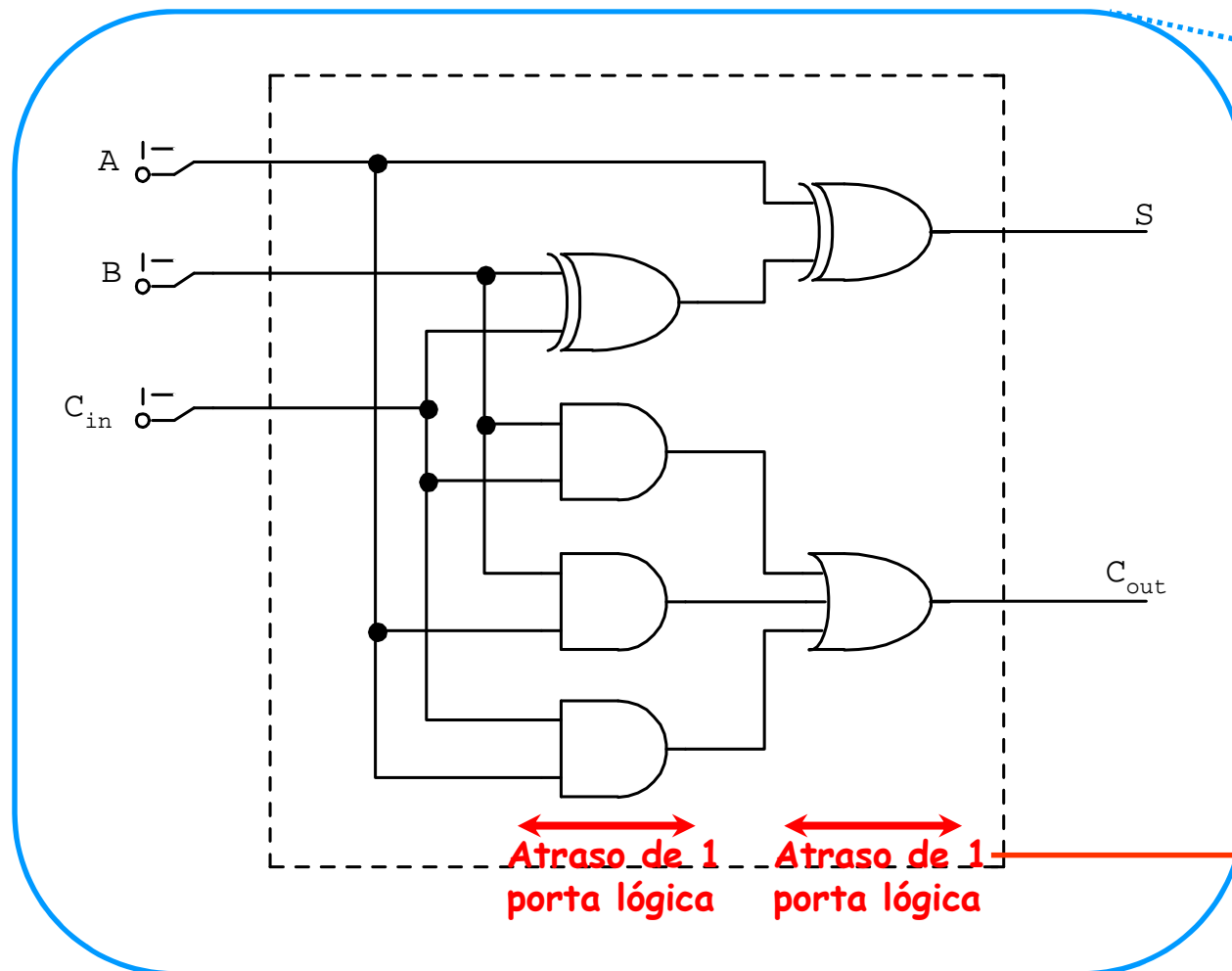
$$C_{out} = AB + BC_{in} + AC_{in}$$



Somador de 4 bits

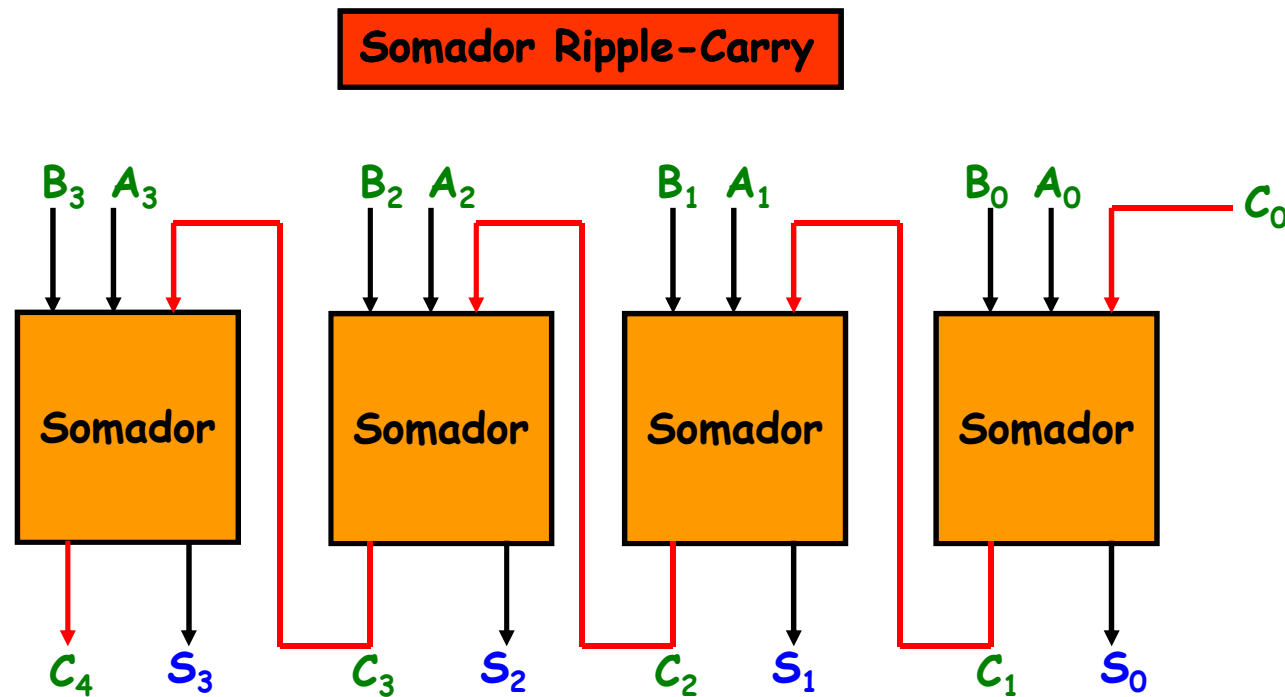


Somador Bit Slice



Para gerar o carry seguinte há 2 estágios de portas lógicas

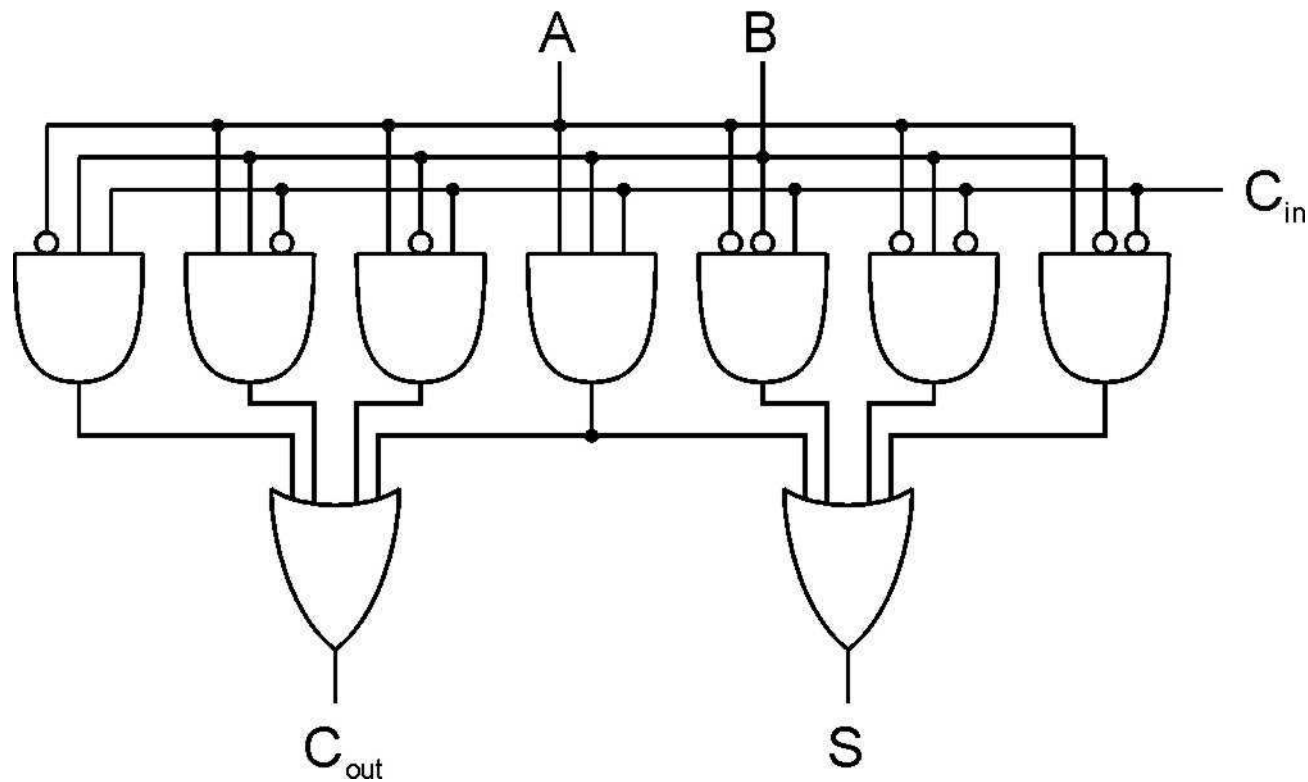
Somador de 4 bits



- Ripple-Carry: Ondulação ou Propagação do Carry. Carry-Out de um estágio se transforma no Carry-In do estágio seguinte.
- A_i e B_i "alimentam" os somadores em paralelo, mas o circuito deve esperar a propagação dos Carries para concluir a operação.

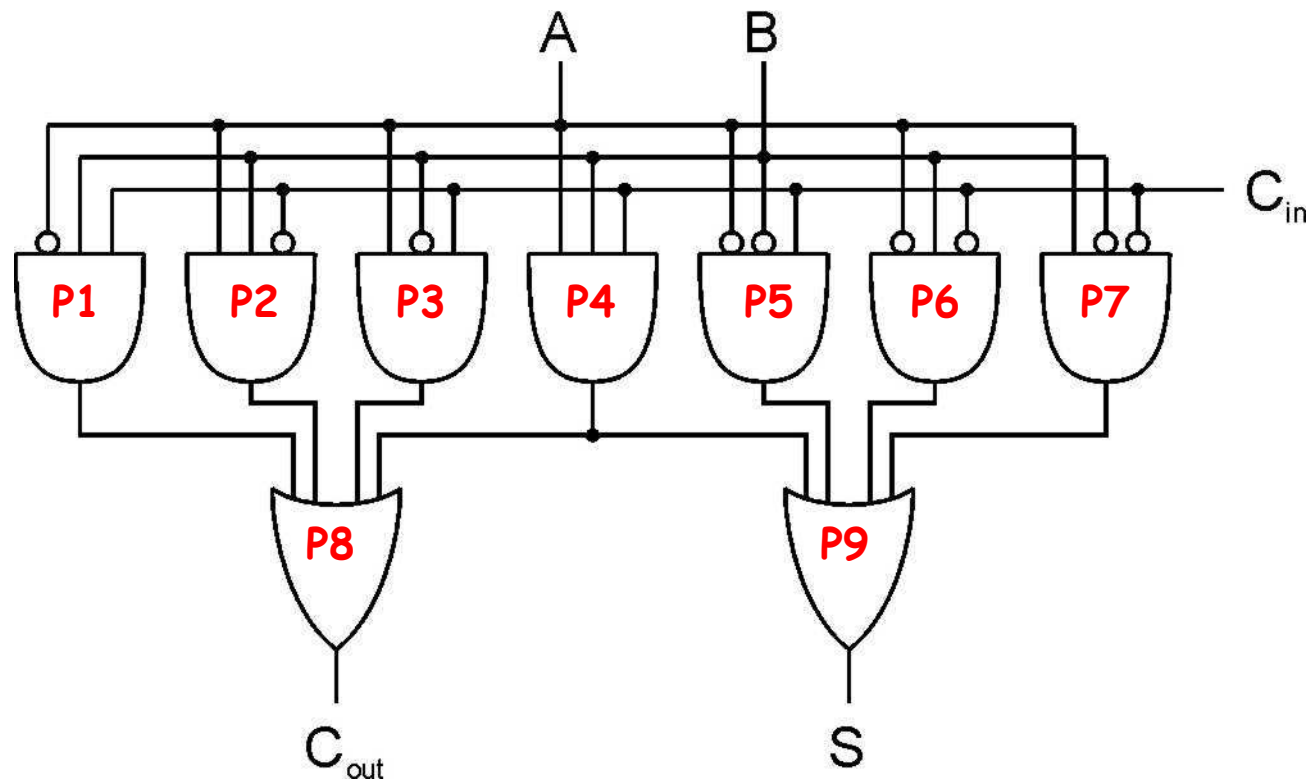
Exercícios

1. Verifique se o circuito abaixo executa a função de um somador



Soluções

1



$$C_{out} = P1 + P2 + P3 + P4$$

$$C_{out} = \bar{A}BC_{in} + A\bar{B}C_{in} + A\bar{B}C_{in} + ABC_{in}$$

$$S = P4 + P5 + P6 + P7$$

$$S = ABC_{in} + \bar{A}BC_{in} + A\bar{B}C_{in} + A\bar{B}C_{in}$$

Soluções

1

$$S = ABC_{in} + \overline{A}BC_{in} + \overline{A}\overline{B}C_{in} + \overline{A}\overline{B}\overline{C}_{in}$$

Tabela Verdade S

A	B	C _{in}	S
0	0	0	
0	0	1	1
0	1	0	1
0	1	1	
1	0	0	1
1	0	1	
1	1	0	
1	1	1	1

Tabela para S é igual à Tabela do S do slide 54, que é a TV do Somador

Soluções

1

$$C_{out} = \overline{A}BC_{in} + A\overline{B}C_{in} + \overline{A}B\overline{C}_{in} + AB\overline{C}_{in}$$

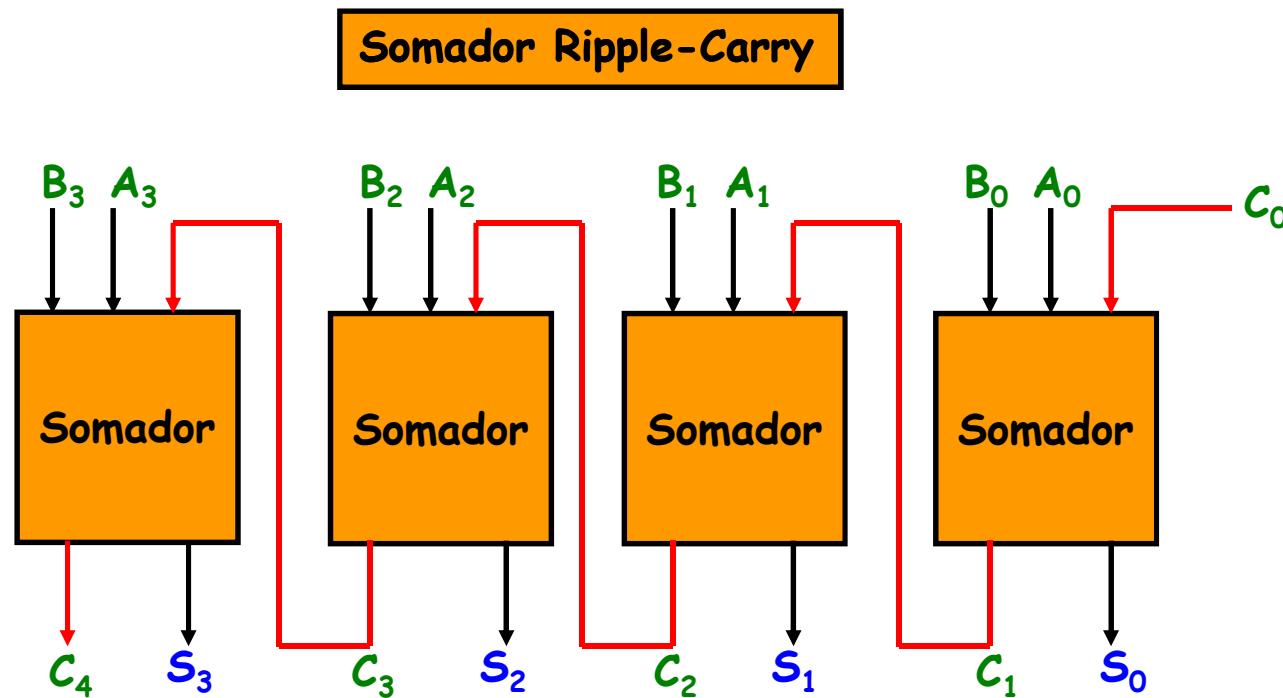
Tabela Verdade C_{out}

A	B	C_{in}	C_{out}
0	0	0	
0	0	1	
0	1	0	
0	1	1	1
1	0	0	
1	0	1	1
1	1	0	1
1	1	1	1

Tabela para C_{out} é igual à Tabela do C_{out} do slide 55, que é a TV do Somador

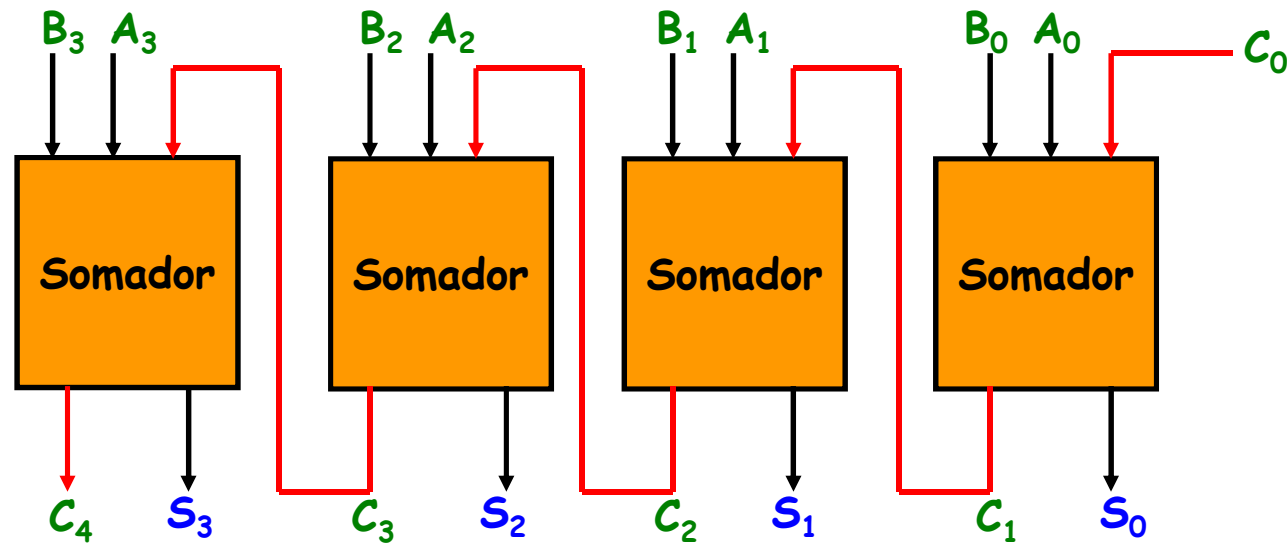
Exercícios

2. Considere um somador ripple-carry de 4 bits. Considere que as portas lógicas têm um atraso de 1ns. Qual é o atraso causado pelo somador ripple-carry para propagar o carry por todos os somadores?



Soluções

2



Para gerar:

$\left\{ \begin{array}{l} C_1 \text{ consome-se } 2\text{ns} \\ C_2 \text{ consome-se } 4\text{ns} \\ C_3 \text{ consome-se } 6\text{ns} \\ C_4 \text{ consome-se } 8\text{ns} \end{array} \right.$

Quanto maior o número de bits do somador,
maior o atraso para gerar o carry final

Resumo da Aula de Hoje

Tópicos mais importantes:

- **Circuitos Combinacionais Especiais**
 - **Multiplexadores/Demultiplexadores**
 - **Gerador de Paridade/Verificador de Paridade**
 - **Circuitos Aritméticos**
 - Somador