

Circuitos Combinacionais

Instituto Federal de Educação, Ciência e
Tecnologia de Santa Catarina

Profa. Fernanda Argoud

Definição

Um circuito combinacional é aquele constituído por um conjunto de portas lógicas, as quais determinam (pelas EQUAÇÕES BOOLEANAS) os valores das saídas diretamente a partir dos valores atuais das entradas.

Ao contrário dos circuitos sequenciais, não há alteração dos estados do circuito, como função do tempo

Projeto de Circuitos Combinacionais

Passos para montagem de um circuito combinacional:

1. Determinar as representações (símbolo/letra) para cada variável de entrada e saída;
2. Identificação do problema (Tabela-Verdade);
3. Determinação das equações lógicas simplificadas;
4. Verificar quais componentes comerciais podem ser utilizados;
5. Desenhar o circuito final.

Projeto de Circuitos Combinacionais

Exemplo 1:

Projetar o circuito que recebe um inteiro binário de 3 bits e determina se este número é menor ou igual a 4.

O primeiro passo é pensar no que seria(m) a(s) entrada(s) e saída(s) do circuito!



Projeto de Circuitos Combinacionais

Se o circuito “recebe” um inteiro, então esta é a entrada esperada do circuito → um número inteiro!

O fato de ser um “número binário de 3 bits” nos indica:

- que a entrada será um inteiro binário (como deve ser, em sistemas digitais! ;));
- os números que podem entrar neste circuito variam entre 0 e 7 (3 bits);
- a entrada será composta por 3 pinos, cada um para um destes bits, que chamaremos de A, B e C.

Projeto de Circuitos Combinacionais



Projeto de Circuitos Combinacionais

Quanto à saída: ela deverá “indicar”, de alguma maneira, se o número que entrou em ABC é, ou não, menor ou igual à 4.

Em circuitos digitais, esta indicação pode ser dada, simplesmente, por um valor alto na saída (1), quando houver detecção, ou um valor baixo (0), quando não houver!

Em outras palavras, se entrar algum número entre 0 e 4, a saída deverá ser 1. Caso contrário, deverá ser 0!



Projeto de Circuitos Combinacionais

O segundo passo é montar a tabela-verdade, que vai se parecer com:

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	

Projeto de Circuitos Combinacionais

A saída esperada é que o circuito detecte valores de entradas menores ou iguais a 3. Então, na TV, as saídas Y deverão ser “1” apenas para os 4 primeiros valores:

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	

= 0

= 1

= 2

= 3

= 4

= 5

= 6

= 7



Projeto de Circuitos Combinacionais

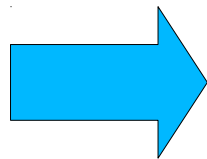
Desta forma:

A	B	C	Y	
0	0	0	1	= 0
0	0	1	1	= 1
0	1	0	1	= 2
0	1	1	1	= 3
1	0	0	1	= 4
1	0	1	0	= 5
1	1	0	0	= 6
1	1	1	0	= 7

Projeto de Circuitos Combinacionais

Para começar o 3o. Passo, você precisa agora, converter a TV em uma equação lógica booleana ou Mapa de Karnaugh e minimizar o número de componentes!

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

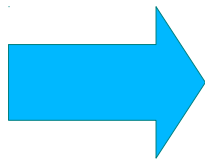


$$\begin{aligned} Y &= \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}C + \bar{A}B\bar{C} + \bar{A}BC + A\bar{B}\bar{C} \\ &= \bar{A}\bar{B}(\bar{C} + C) + \bar{A}B(\bar{C} + C) + A\bar{B}\bar{C} \\ &= \bar{A}(B + \bar{B}) + A\bar{B}\bar{C} \\ &= \bar{A} + A\bar{B}\bar{C} = A + \bar{B}\bar{C} \end{aligned}$$

Projeto de Circuitos Combinacionais

Ou:

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



AB \ C	00	01	11	10
0	1	1	0	1
1	1	1	0	0

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

Projeto de Circuitos Combinacionais

No quarto passo, verificaremos como podemos especificar o circuito que melhor se adequará à equação booleana resultante.

Verificamos que a saída desejada limita-se à entrada A invertida mais as entradas B e C, também invertidas, jogadas como entradas de uma porta E !!

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

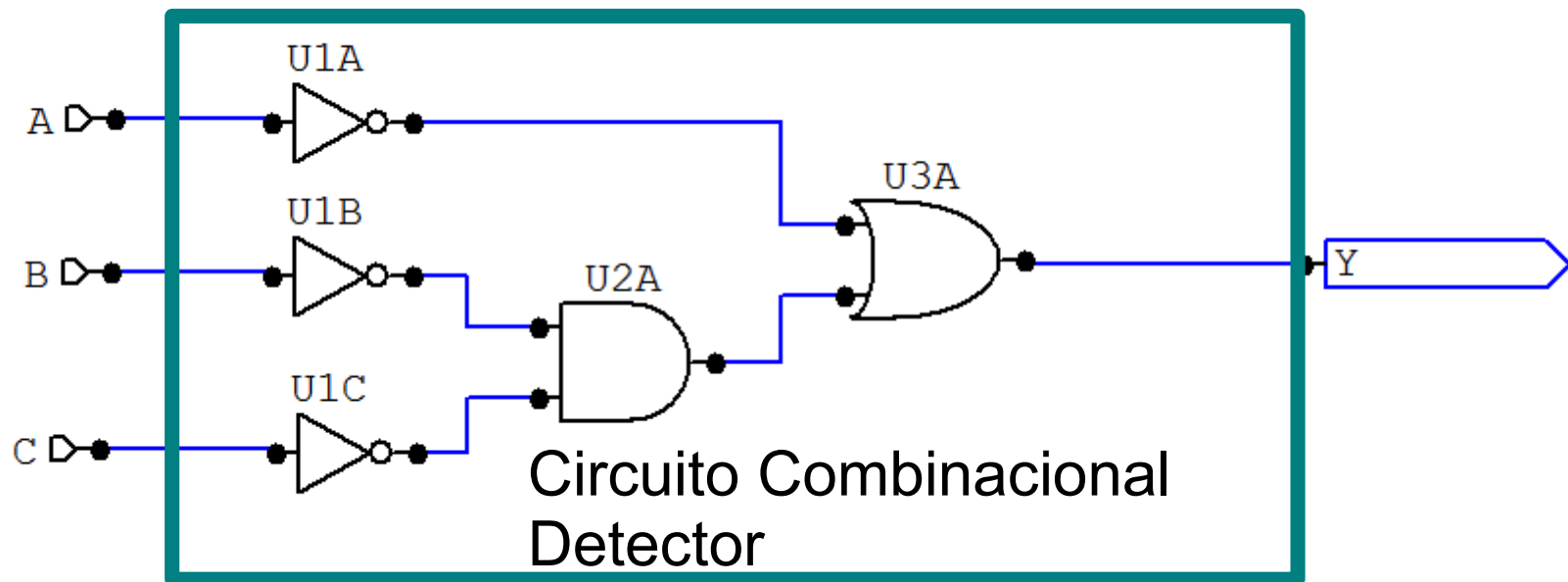
Inversoras

Operação
OR

Operação
AND

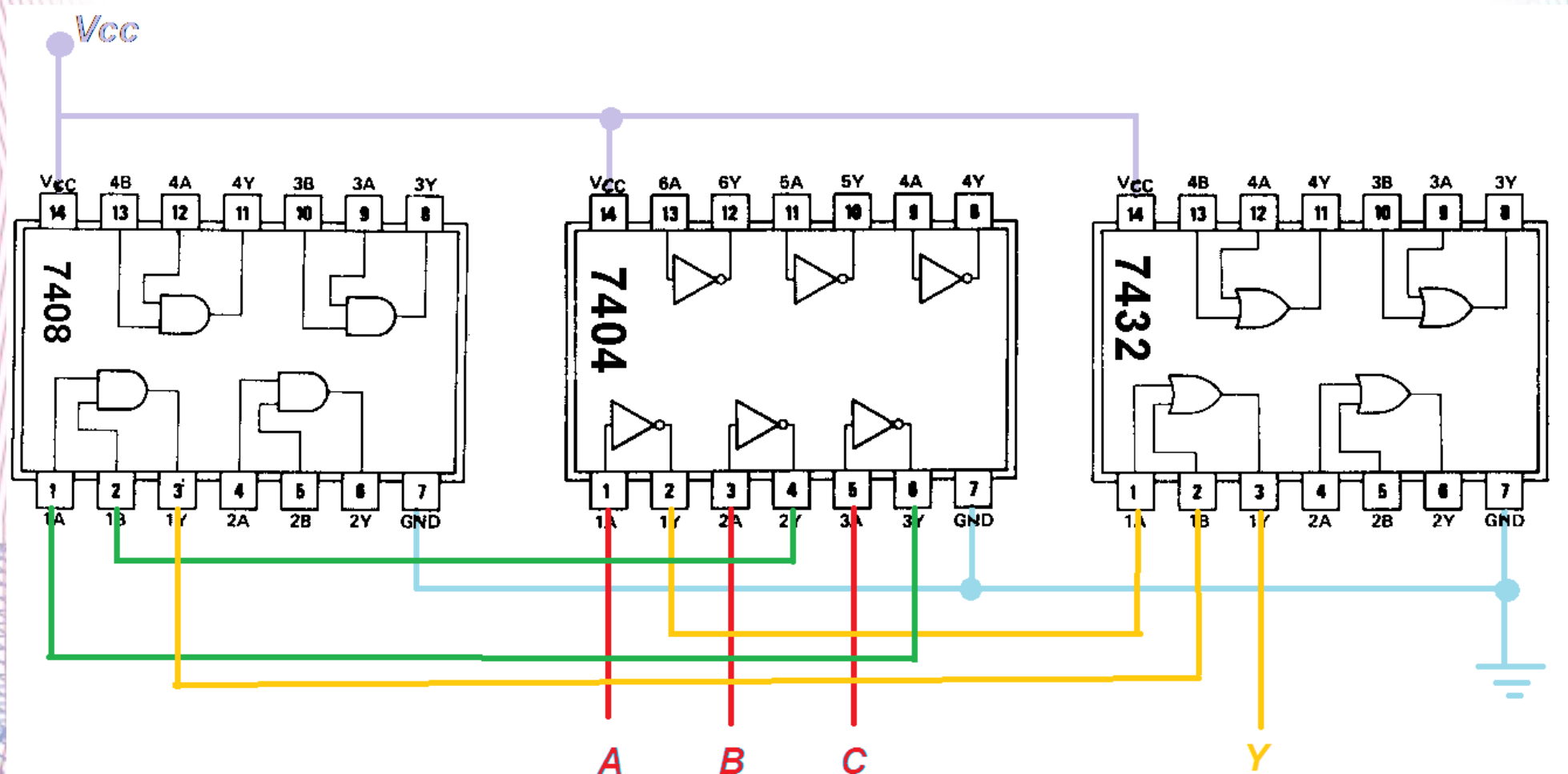
Projeto de Circuitos Combinacionais

Por fim, chegamos ao último passo:



Projeto de Circuitos Combinacionais

Esquema de interligação dos CIs:



Projeto de Circuitos Combinacionais

Exercícios:

- 1) Se o problema restringisse que só poderíamos usar portas NAND (NÃO-E) e OR (OU) na construção deste circuito, como você faria?
- 2) Se o problema pedisse que a saída Y também detectasse quando todas as entradas fossem 1, que alteração(ões) você faria no circuito?

Projeto de Circuitos Combinacionais

Exercícios Complementares:

Aulas Práticas 3 e 4, pgs. 133-137 do Capítulo 16

Apostila da Exsto

CODIFICADORES E DECODIFICADORES

- Como o próprio nome diz, são circuitos que “traduzem” o código binário utilizado nas portas lógicas, para outras “linguagens”, ou códigos.
- Os CODIFICADORES traduzem códigos para numeração binária.
- Os DECODIFICADORES fazem o inverso: “traduzem” o código binário para outro tipo de código.

CODIFICADORES E DECODIFICADORES

- Normalmente são utilizados para apresentar os valores (binários) em uma forma que O USUÁRIO possa compreender, e vice-versa.
- Também são usados para adaptar diferentes códigos usados em um circuito, para linguagens, protocolos ou códigos que outros circuitos e dispositivos utilizem.

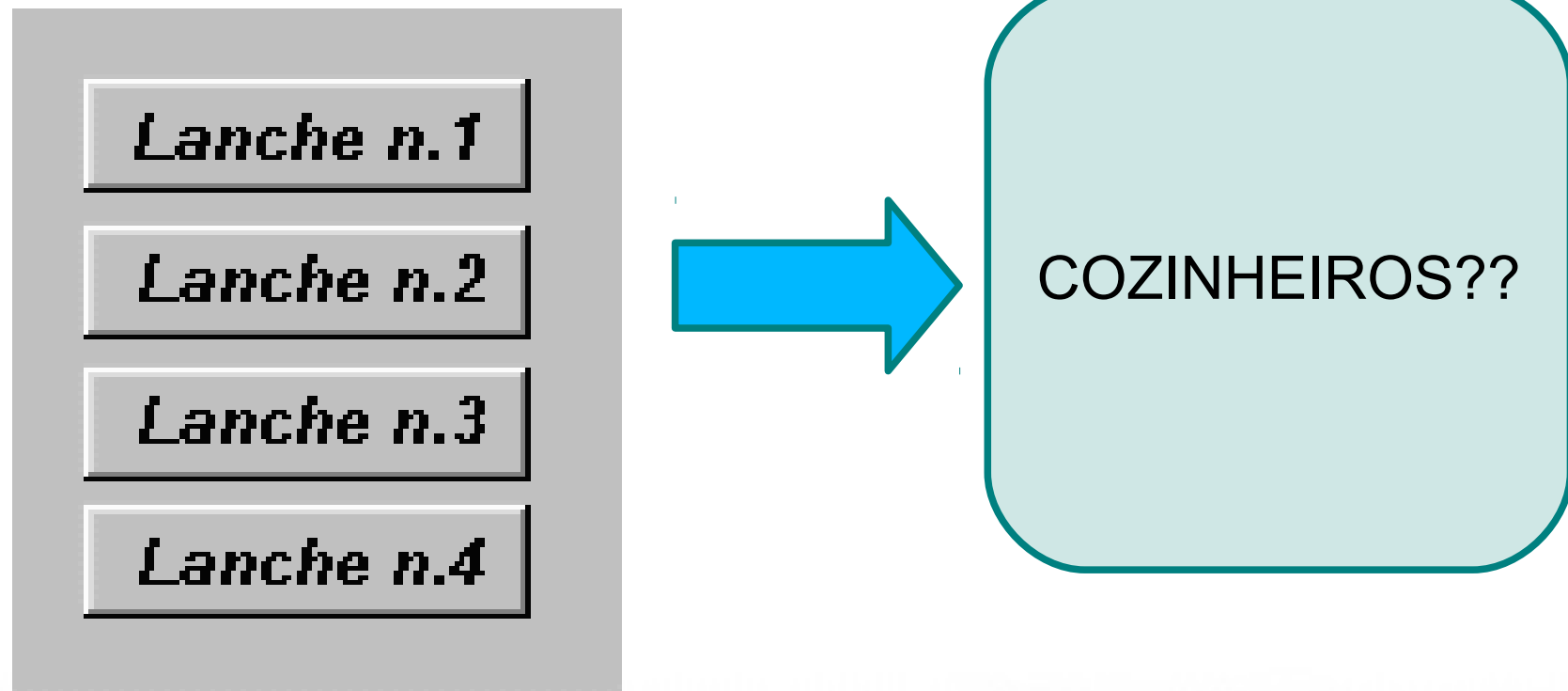


CODIFICADORES E DECODIFICADORES

- Existem dezenas de codificadores e decodificadores comerciais.
- Os mais comuns são:
 - ◆ BCD (8421)
 - ◆ Octal
 - ◆ Hexadecimal \Leftrightarrow Binário (BCH)
 - ◆ Decimal
 - ◆ ASCII
 - ◆ Paridade
 - ◆ etc

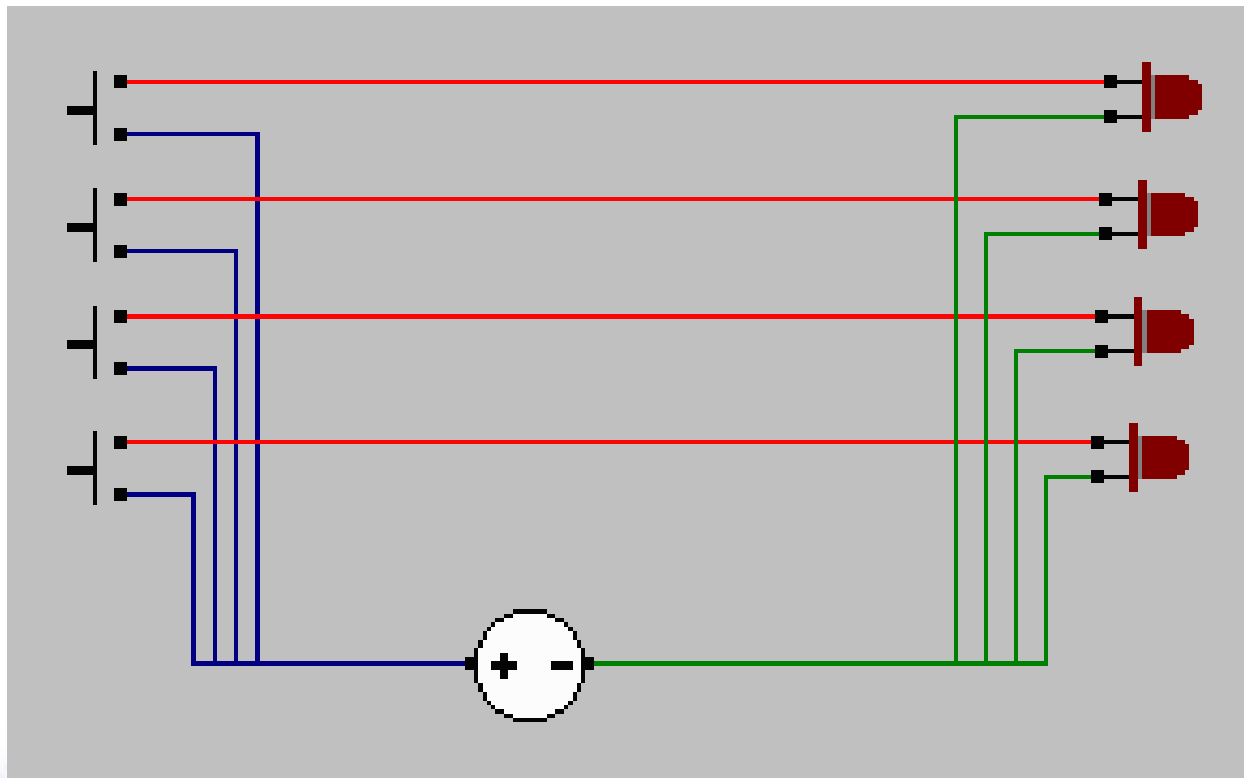
CODIFICADORES E DECODIFICADORES

- Analogia: o problema do lanche!



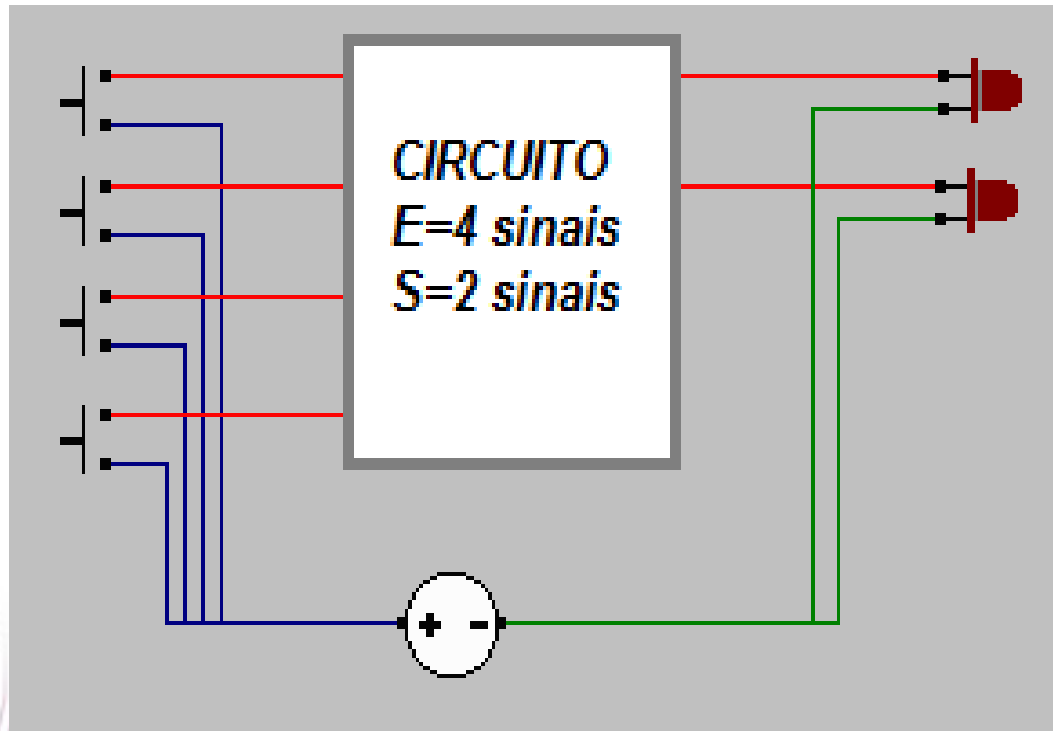
CODIFICADORES E DECODIFICADORES

- Analogia: o problema do lanche!



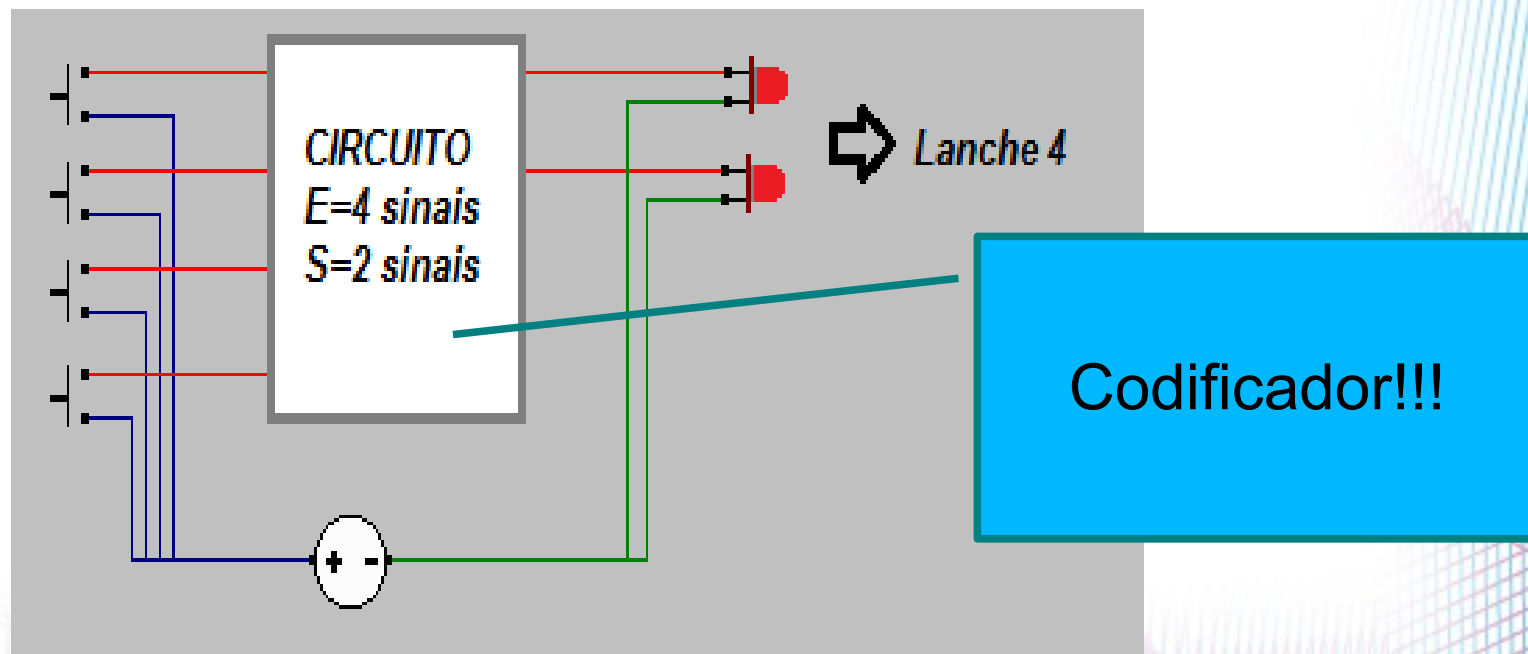
CODIFICADORES E DECODIFICADORES

- Mas não sabemos que quatro estados podem ser realizados utilizando apenas dois bits (ou chaves, ou par de fios, ou leds)???



CODIFICADORES E DECODIFICADORES

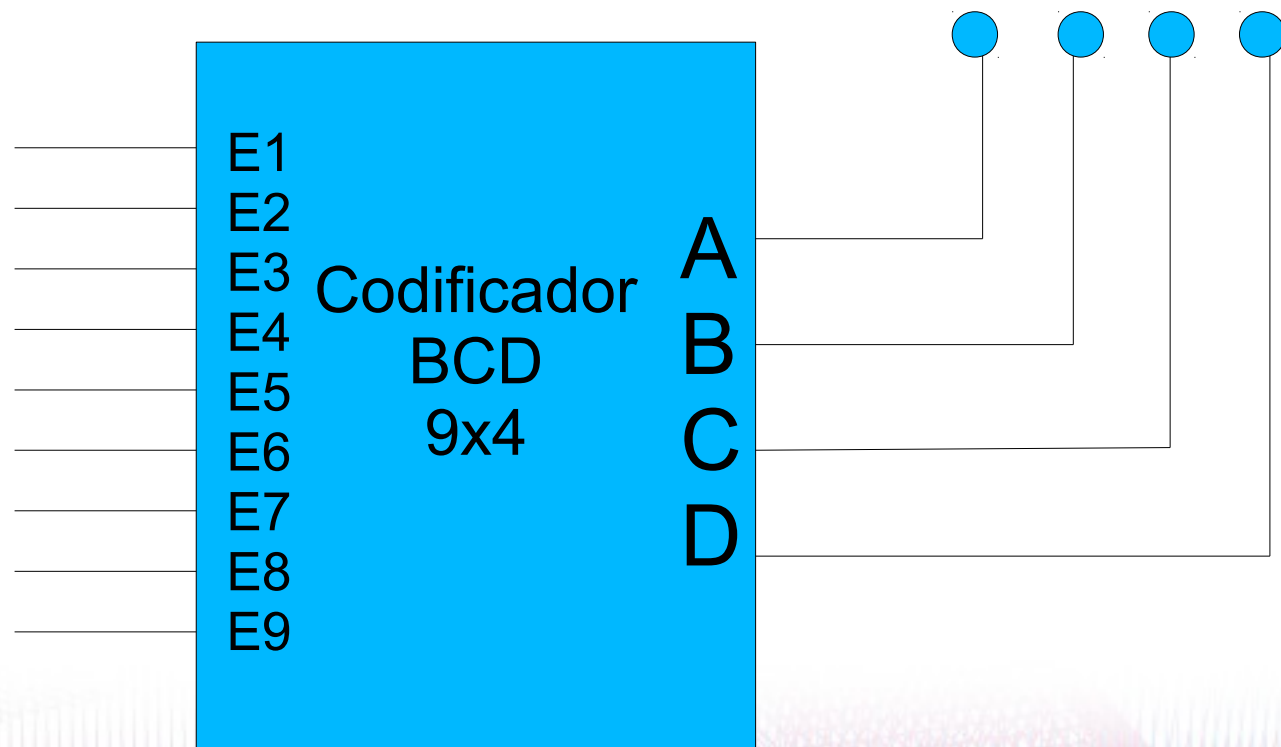
- Mas não sabemos que quatro estados podem ser realizados utilizando apenas dois bits (ou chaves, ou par de fios, ou leds)???



Exemplo de conversão decimal – BCD 8421:

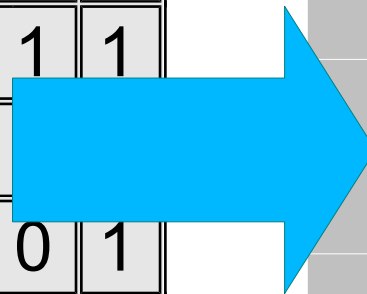
$$725_{10} \rightarrow 0111_0010_0101_{\text{BCD}}$$

- Código BCD (8421) – *binary coded decimal*, ou “binário codificado em decimal”:



CODIFICADOR BCD

9	8	7	6	5	4	3	2	1	A	B	C	D
0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	1	0	0	0	1	0
0	0	0	0	0	0	1	0	0	0	0	1	1
0	0	0	0	0	1	0	0	0	0	1	0	0
0	0	0	0	1	0	0	0	0	0	1	0	1
0	0	0	1	0	0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	0	0	0	1	1	1
0	1	0	0	0	0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0	0	1	0	0	1



Entrada ativada	A	B	C	D
-	0	0	0	0
E1	0	0	0	1
E2	0	0	1	0
E3	0	0	1	1
E4	0	1	0	0
E5	0	1	0	1
E6	0	1	1	0
E7	0	1	1	1
E8	1	0	0	0
E9	1	0	0	1

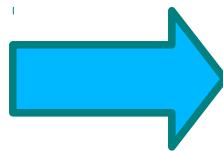
CODIFICADOR BCD

- Note que, apesar de termos 9 entradas, não temos as $2^9 = 512$ linhas na TV, porque cada uma das 9 entradas só pode ser ativada uma de cada vez = 9 estados + nenhuma entrada ativada = 10 linhas!
- As entradas inválidas, então, não vão resultar em saídas válidas => IRRELEVANTES (X)!
- Para efeitos de projeto de circuito, é necessário utilizar todas as linhas, então os estados irrelevantes são marcados como X na TV e no MK, e assumidos como “0”, ou “1”, o que for mais conveniente.

CODIFICADOR BCD

- Para exemplificar, vamos ver o caso do codificador BCD 3x2, cujo circuito é muito mais simples

E3	E2	E1	A	B
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
1	0	0	1	1



E3	E2	E1	A	B
0	0	0	0	0
0	0	1	0	1
0	1	0	1	0
0	1	1	x	x
1	0	0	1	1
1	0	1	x	x
1	1	0	x	x
1	1	1	x	x

Estados irrelevantes

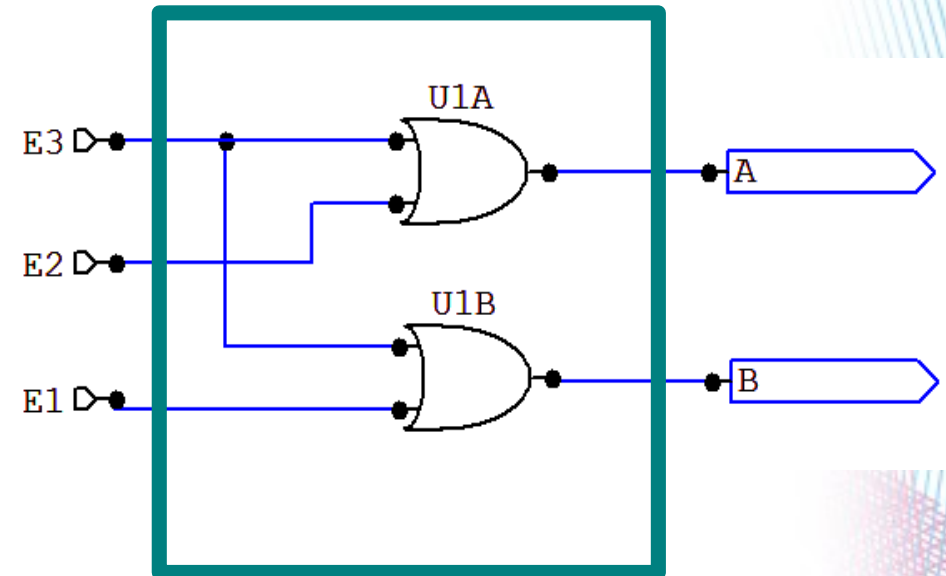
CODIFICADOR BCD

E3 E2	00	01	11	10
E1				
0	0	1	X	1
1	0	X	X	X

$$A = E3 + E2$$

E3 E2	00	01	11	10
E1				
0	0	0	X	1
1	1	X	X	X

$$B = E1 + E3$$

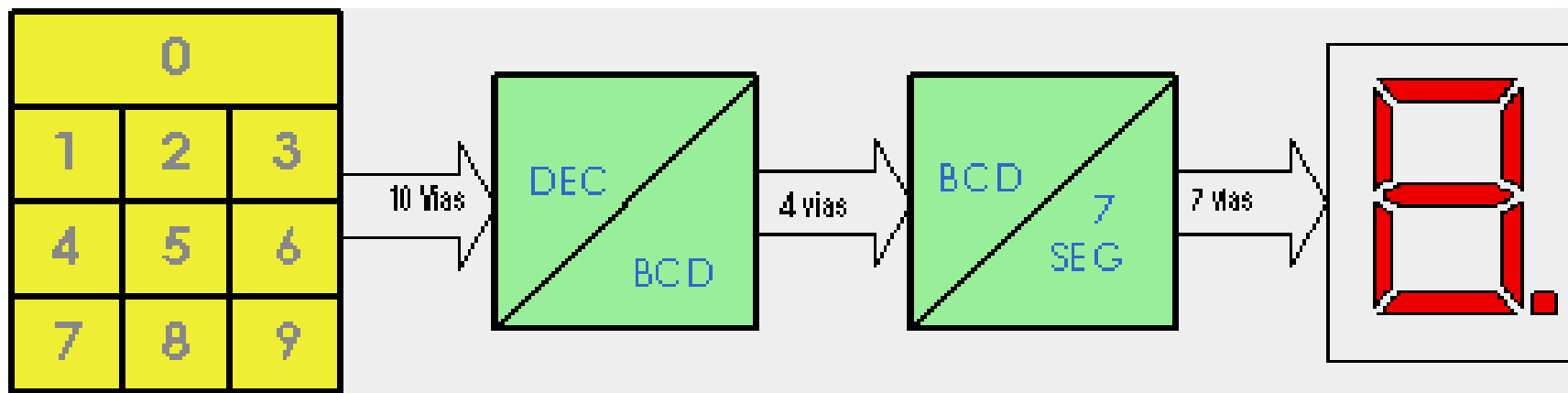


CODIFICADOR BCD

- Codificadores BCD 8421 comerciais mais utilizados:
74147 e **74148**
- Codificadores permitem chaveamento de muitas entradas, em pouquíssimas saídas.
- Ex: teclado de computador que tem 105 teclas > codificadas em apenas 7 fios/vias (permite até 128 entradas!!!)

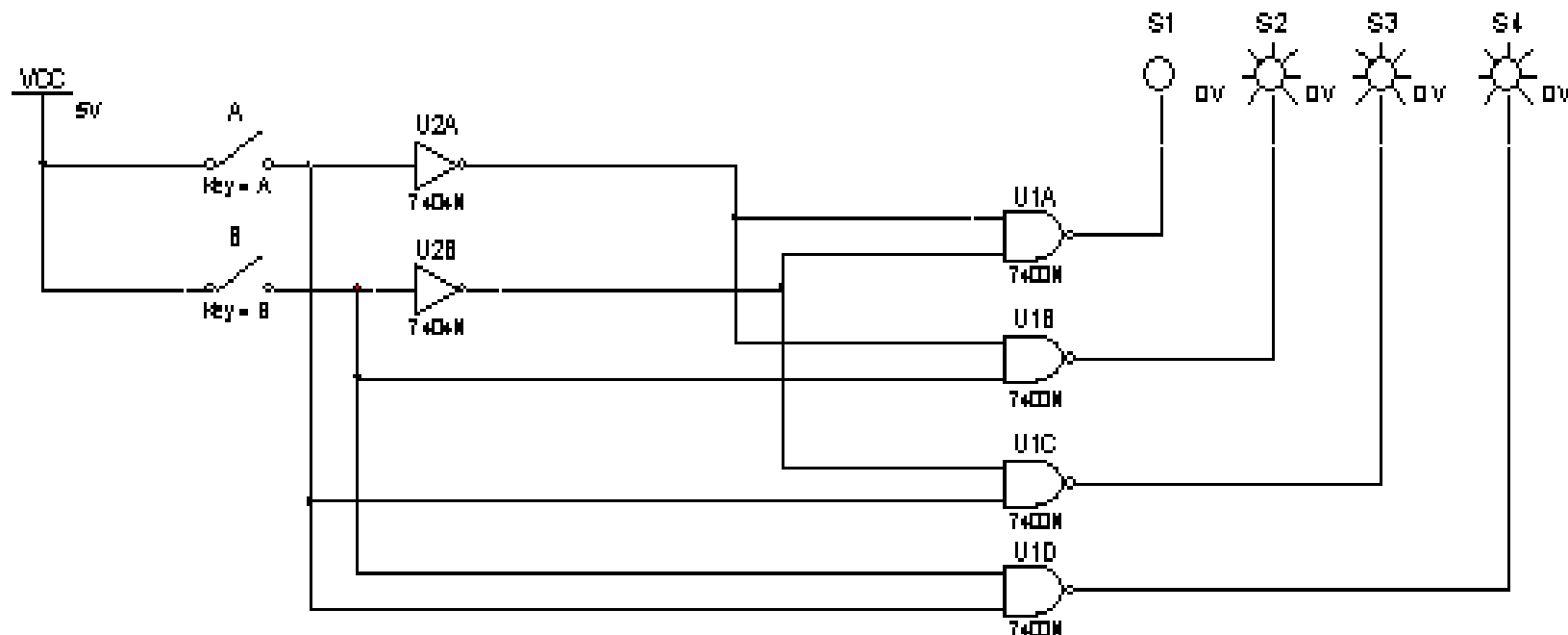
DECODIFICADORES

- Executa a função oposta a dos codificadores
- Permitem traduzir o código binário para outros códigos que homens ou máquinas compreendam
- Exemplo:



DECODIFICADORES DE n PARA 2^n LINHAS

- “Aciona”, “ativa”, apenas uma das saídas, dependendo da entrada binária (ativo baixo):



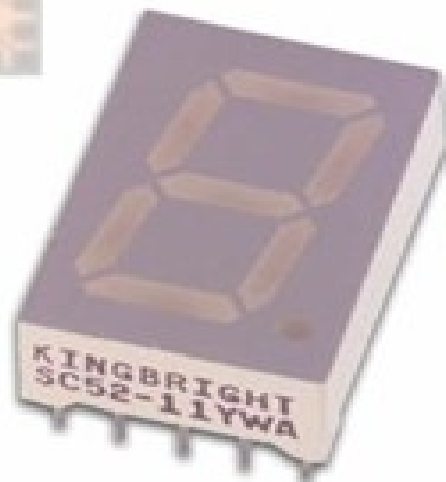
DECODIFICADORES DE n PARA 2^n LINHAS

- Valor binário de 2 bits na entrada coloca em nível baixo apenas uma das 4 saídas (74139):

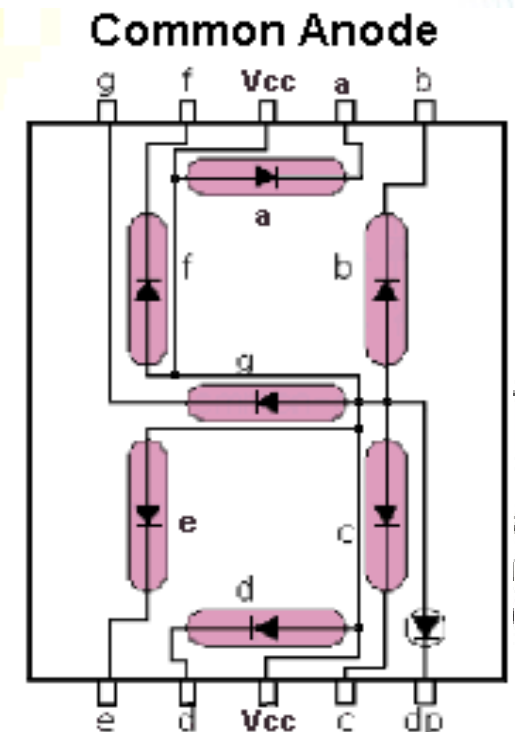
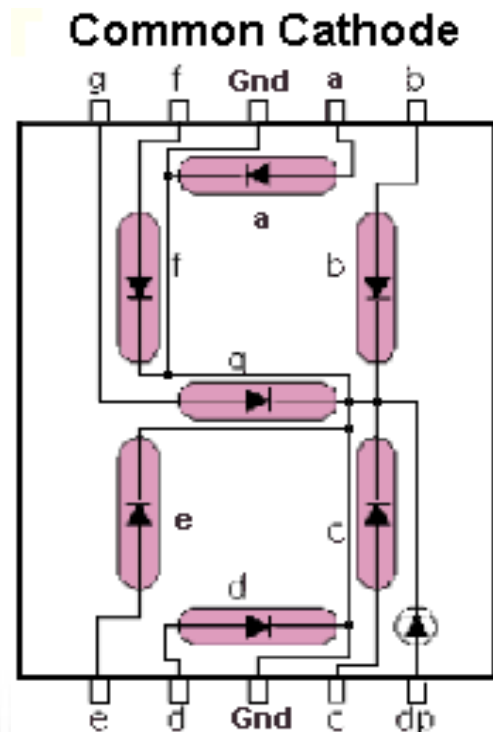
A	B	S1	S2	S3	S4
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

DECODIFICADOR BCD de 7 segmentos

- Converte entrada BCD para acionar mostrador de 7 segmentos (led's ou cristal líquido)



Kingbright



ado ou
todo
mum

Ponto decimal

DECCODIFICADOR BCD de 7

	a	b	c	d	e	f	g
0	1	1	1	1	1	1	0
1	0	1	1	0	0	0	0
2	1	1	0	1	1	0	1
3	1	1	1	1	0	0	1
4	0	1	1	0	0	1	1
5	1	0	1	1	0	1	1
6	1	0	1	1	1	1	1
7	1	1	1	0	0	0	0
8	1	1	1	1	1	1	1
9	1	1	1	1	0	1	1

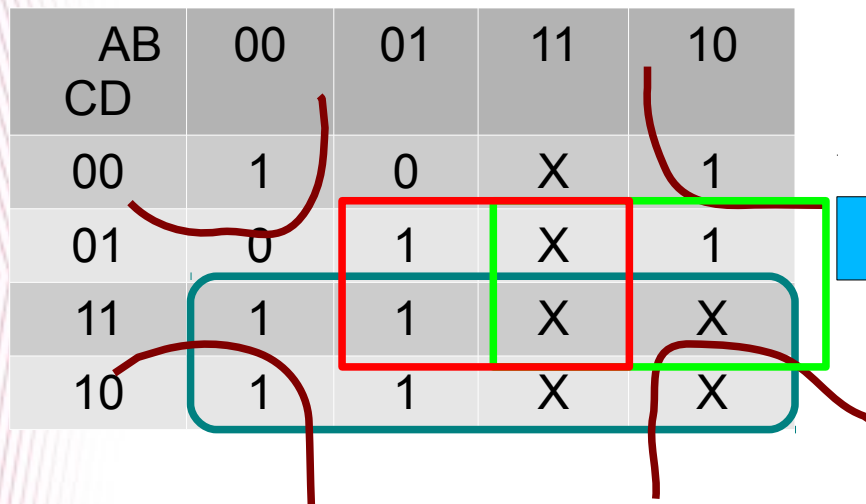
SAÍDA: **CÓDIGO BCD** **SEGMENTOS ACESOS**

	A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	0	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1

DECODIFICADOR BCD de 7 segmentos

Segmento "a":

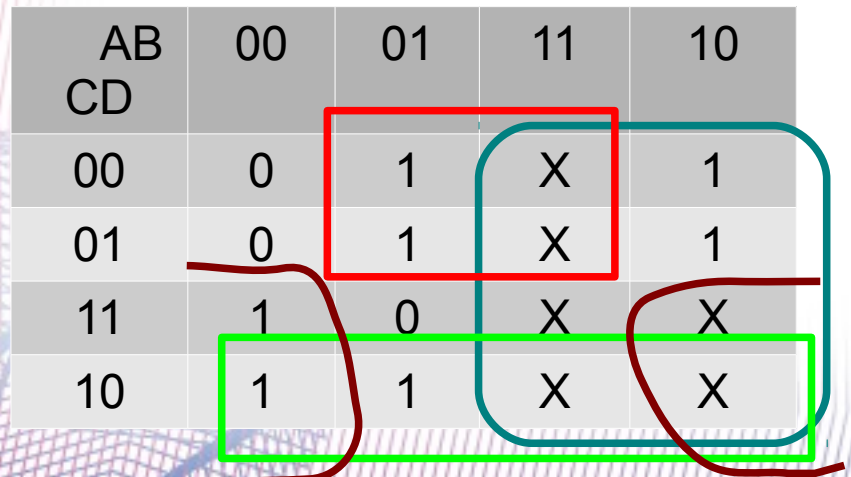
AB CD	00	01	11	10
00	1	0	X	1
01	0	1	X	1
11	1	1	X	X
10	1	1	X	X



$$a = C + \bar{B}\bar{D} + BD + AD$$

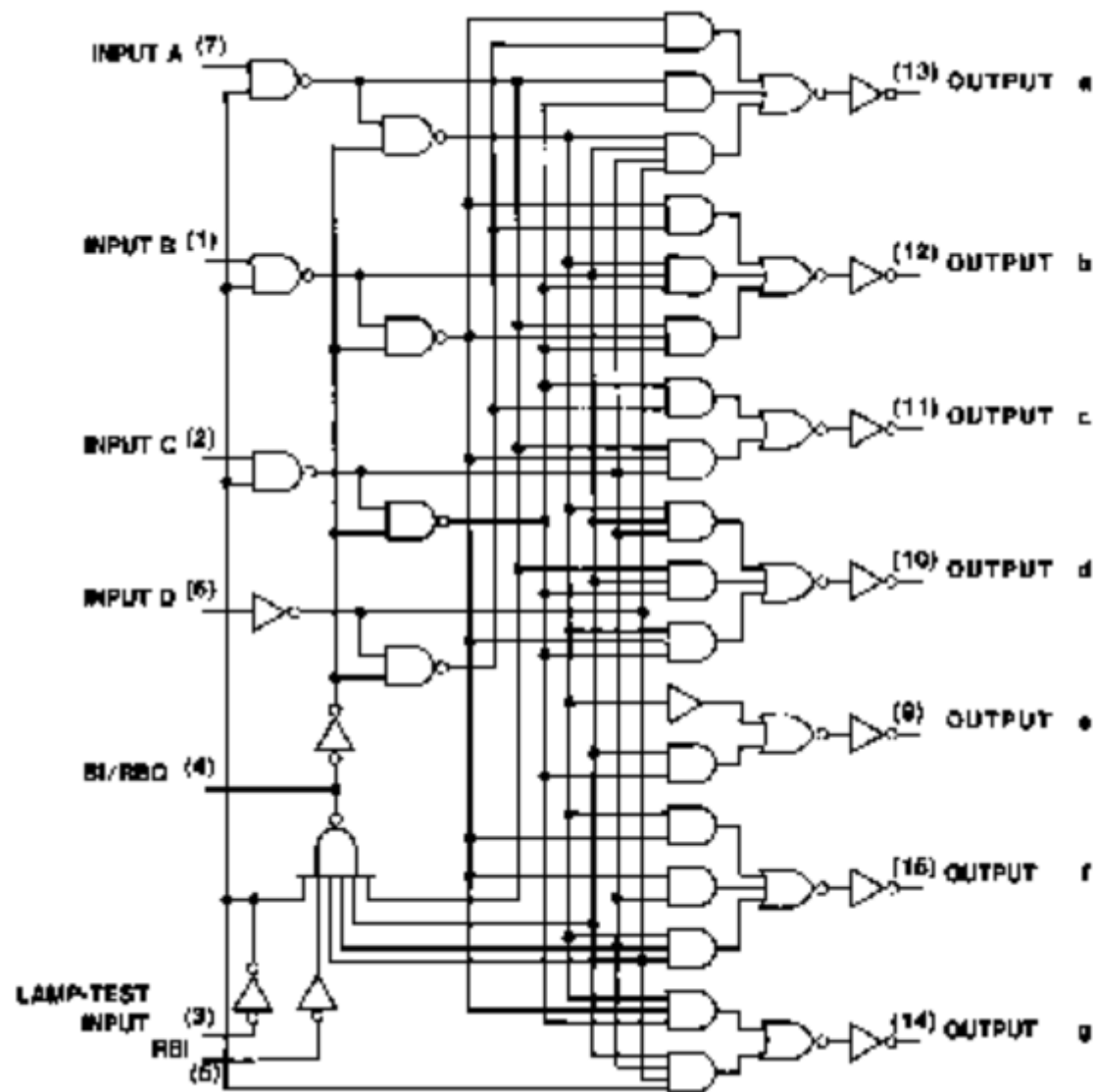
Segmento "g":

AB CD	00	01	11	10
00	0	1	X	1
01	0	1	X	1
11	1	0	X	X
10	1	1	X	X



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

DECODIFICADOR BCD de 7 segmentos



background image

DECODIFICADOR BCD de 7 segmentos

- As correntes nos segmentos variam tipicamente entre 10 e 50 mA → consumo máximo com o dígito oito (todos os segmentos acesos) \cong 400 mA por dígito.
- Displays LCD são muito mais caros, porém nestes o consumo é muito menor.
- Modelos comerciais:
 - ◆ TTL: 7446, 7447, 7448, 7449, 74142, etc
 - ◆ CMOS: 4511

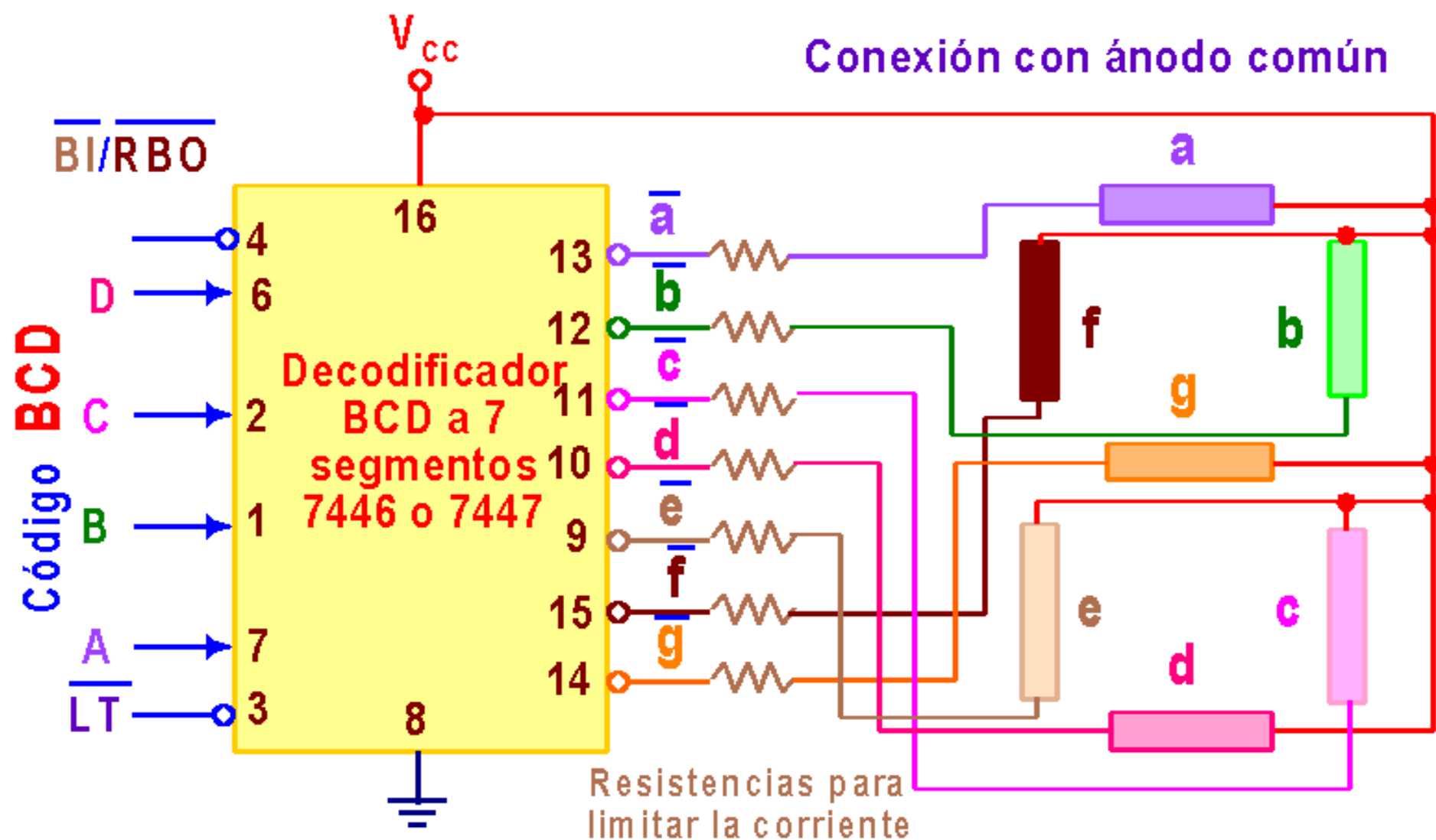


FIGURA 6.1 (a). Decodificador/manejador de BCD a 7 segmentos que maneja una presentación visual de 7 segmentos con ánodo común

MULTIPLEXADORES E DEMULTIPLEXADORES

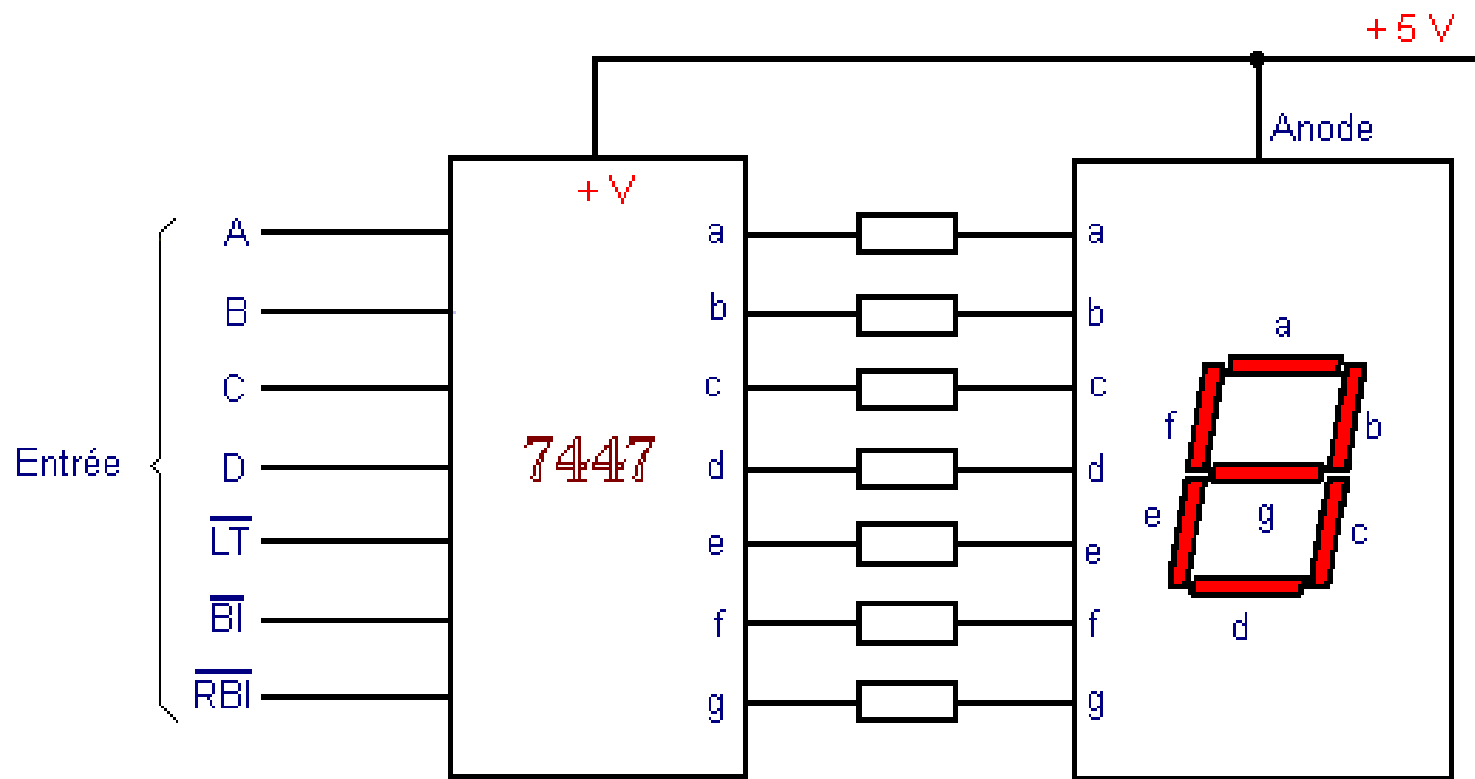
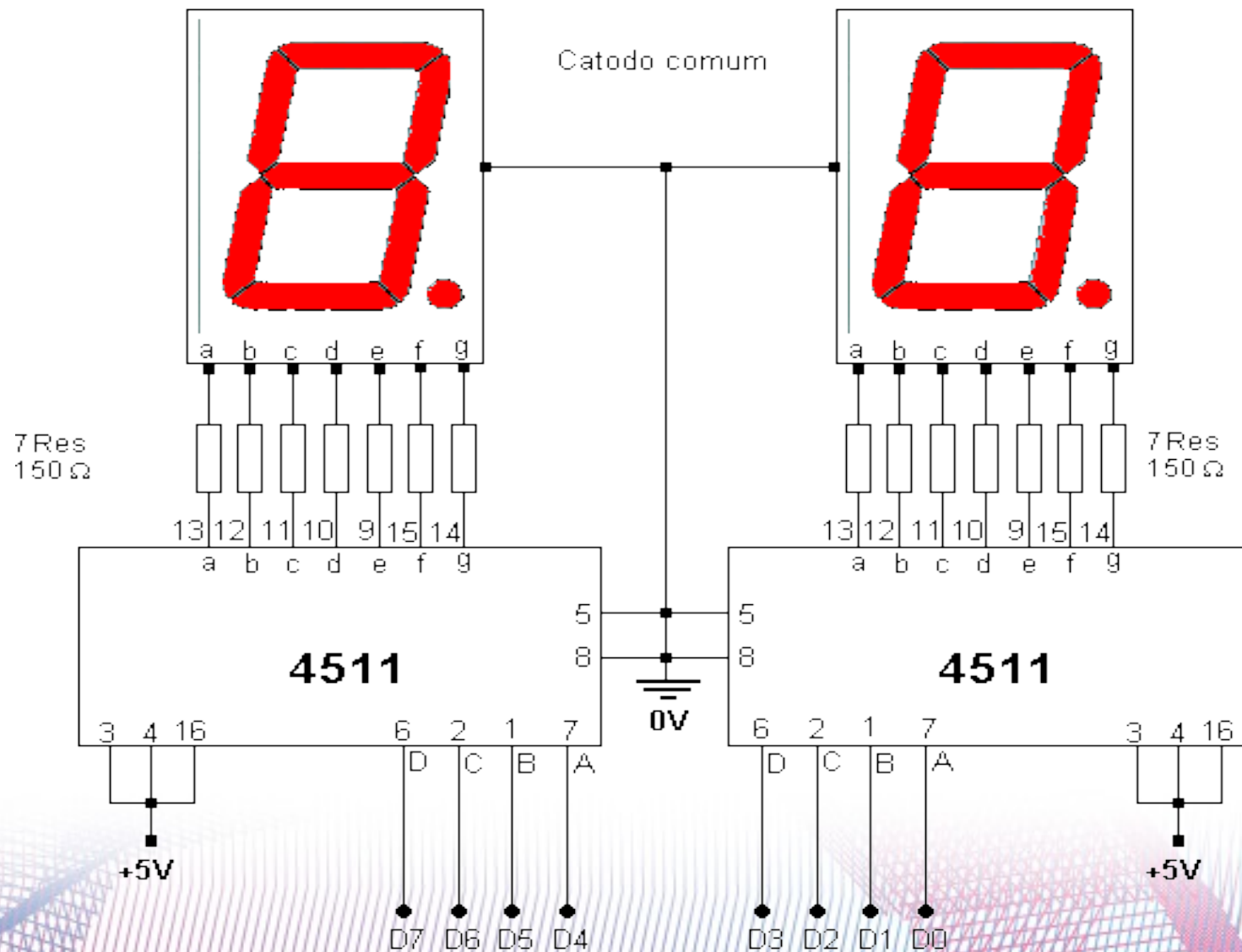
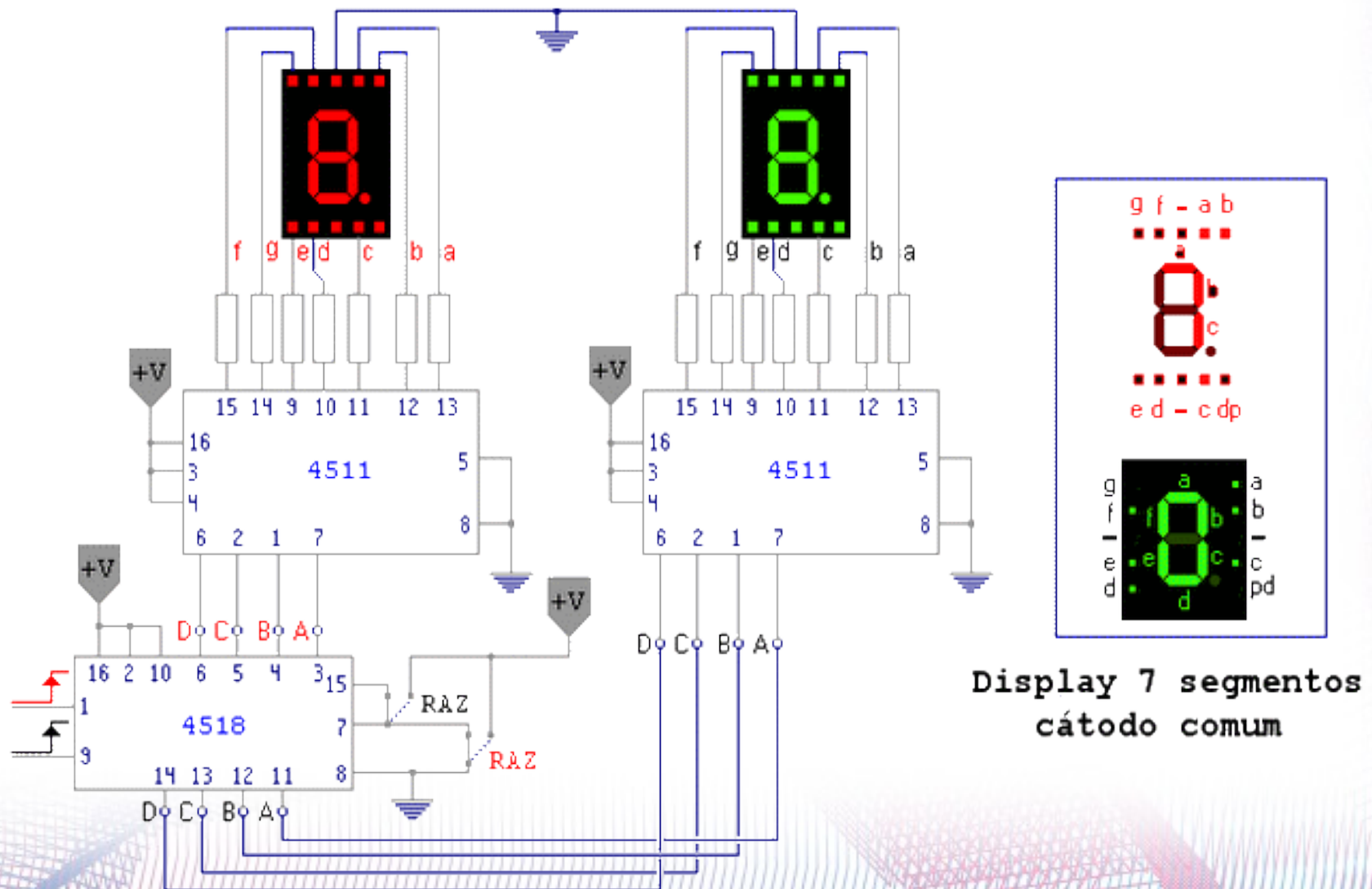


Fig. 37. - Afficheur 7 segments commandé par un décodeur 7447.

DECODIFICADOR BCD de 7 segmentos



DECODIFICADOR BCD de 7 segmentos

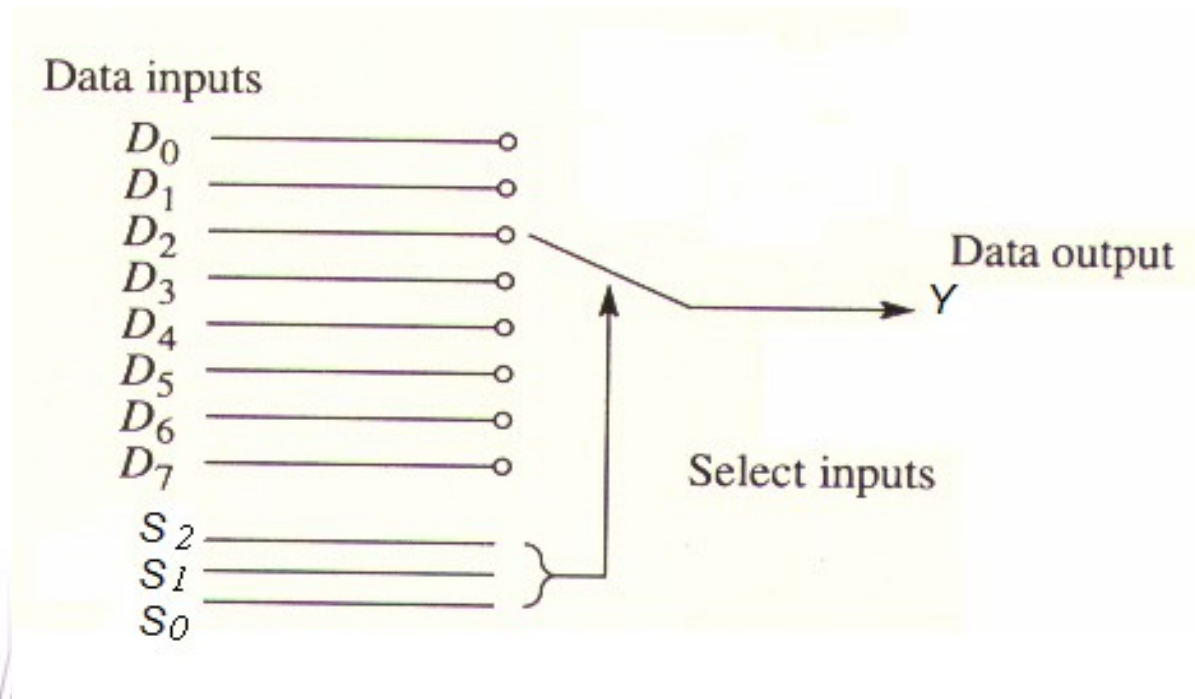


MULTIPLEXADORES E DEMULTIPLEXADORES

- Também chamados de “Seletores de Dados”
- Direcionam, “chaveiam” dados num circuito

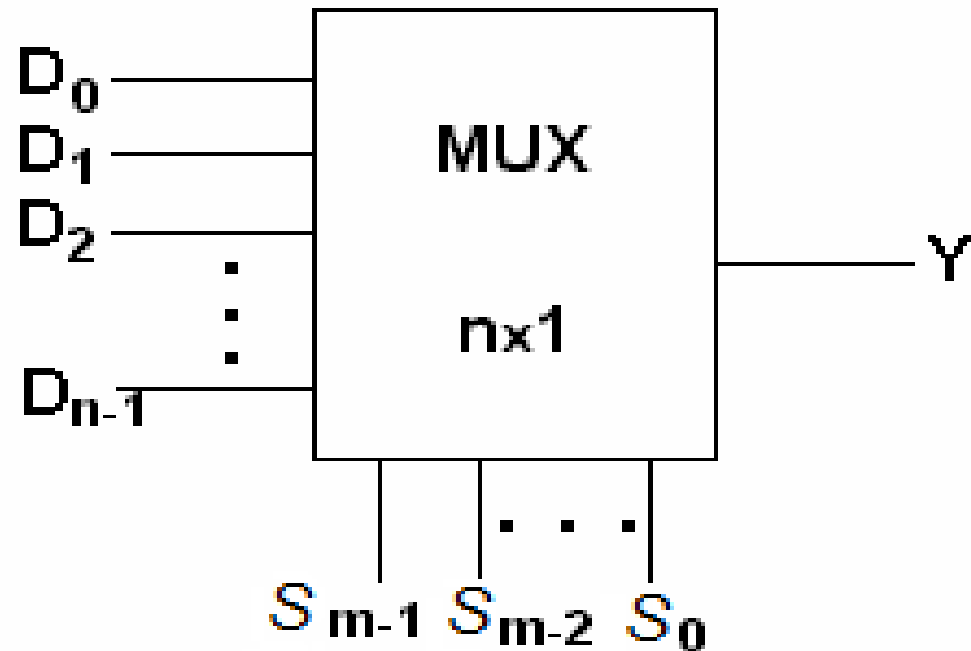
MULTIPLEXADORES

- As entradas de controle ou seleção dos multiplexadores (ou “MUX”) decidem qual das múltiplas entradas será jogada na única saída.



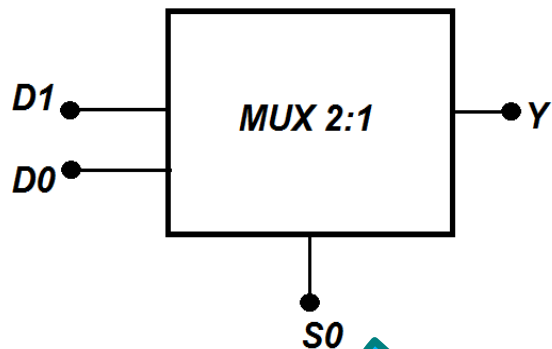
MULTIPLEXADORES

- Entradas de seleção: S_i
- Entradas de dados: D_i
- Saída: Y



MULTIPLEXADORES

- Exemplo: MUX de 2 entradas (1 de seleção)



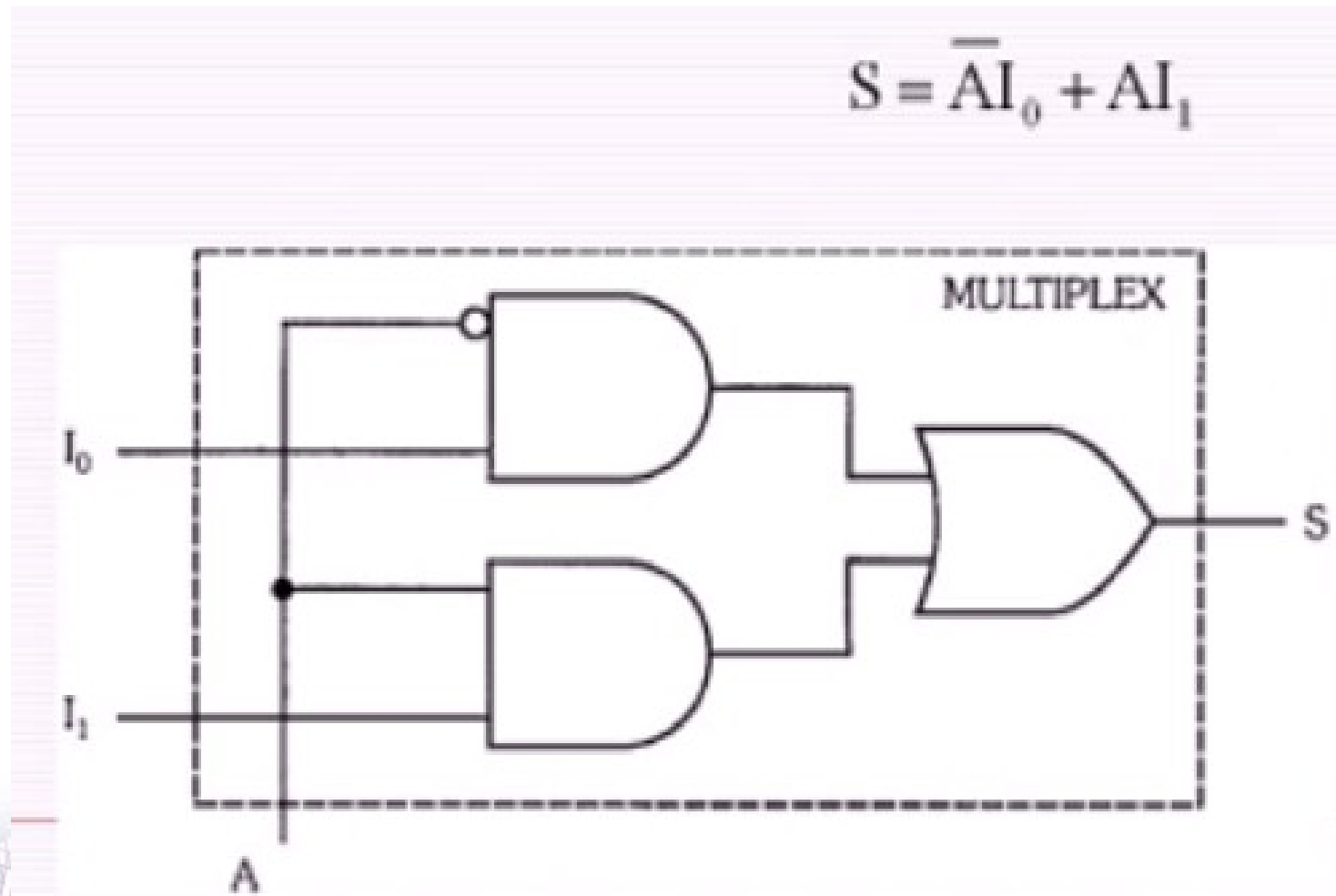
D_1	D_0	S_0	Y
0	0	0	0 (D_1)
0	0	1	0 (D_0)
0	1	0	0 (D_1)
0	1	1	1 (D_0)
1	0	0	1 (D_1)
1	0	1	0 (D_0)
1	1	0	1 (D_1)
1	1	1	1 (D_0)

$D_1 D_0$	00	01	11	10
S_0				
0	0	0	1	1
1	0	1	1	0

$$Y = D_1 \cdot \overline{S_0} + D_0 \cdot S_0$$

MULTIPLEXADORES

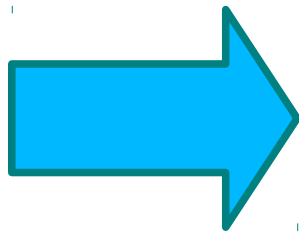
- Exemplo: MUX de 2 entradas (1 de seleção)



MULTIPLEXADORES

- Exemplo2: MUX 4:1 - 4 entradas (2 de seleção) e 1 saída

$$Y = D_0 \cdot \overline{S_1} \cdot \overline{S_0} + D_1 \cdot \overline{S_1} \cdot S_0 + D_2 \cdot S_1 \cdot \overline{S_0} + D_3 \cdot S_1 \cdot S_0$$



S_1	S_0	Y
0	0	D_0
0	1	D_1
1	0	D_2
1	1	D_3

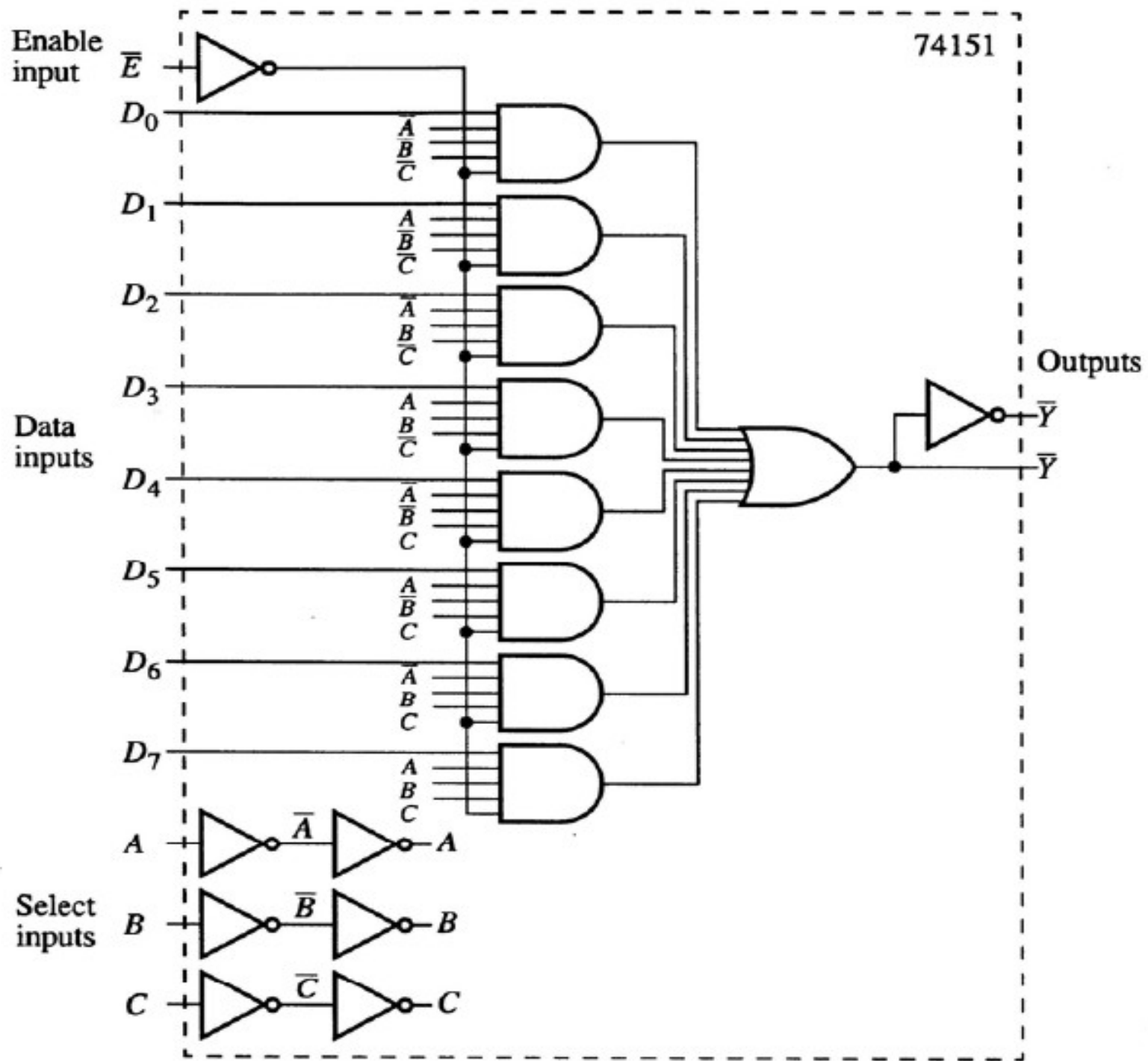


Figure 3.37 8-line-to-1-line data selector/multiplexer

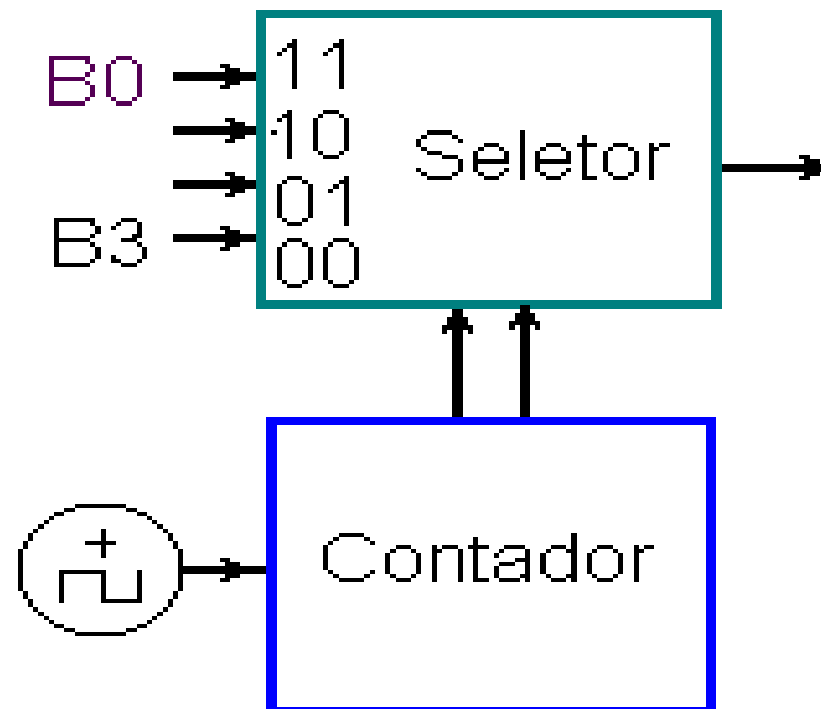
Tabela-Verdade

CBA	\overline{E}	Y	\overline{Y}
XXX	1	0	1
000	0	D_0	$\overline{D_0}$
001	0	D_1	$\overline{D_1}$
010	0	D_2	$\overline{D_2}$
011	0	D_3	$\overline{D_3}$
100	0	D_4	$\overline{D_4}$
101	0	D_5	$\overline{D_5}$
110	0	D_6	$\overline{D_6}$
111	0	D_7	$\overline{D_7}$

- CI 74151: Mux 8x1 com habilitação e saída dupla.

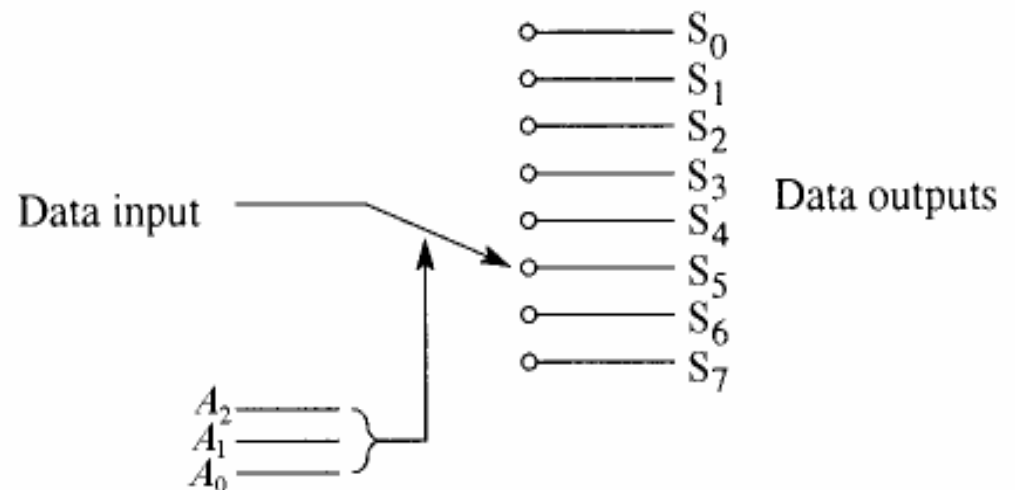
MULTIPLEXADORES

- Exemplo de aplicação: conversão Paralelo \Leftrightarrow Serial



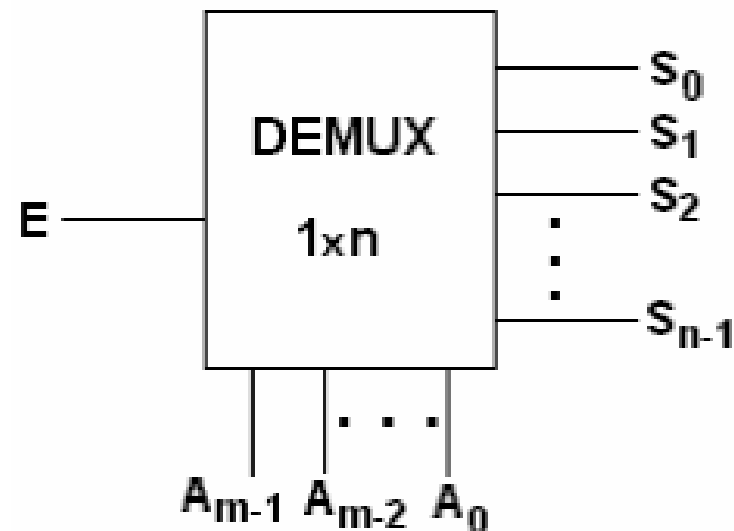
DEMULTIPLEXADORES

- Realizam a função inversa dos MUX.
- As entradas de controle dos demultiplexadores (ou “DEMUX”) decidem para qual das múltiplas saídas será jogada na única entrada.

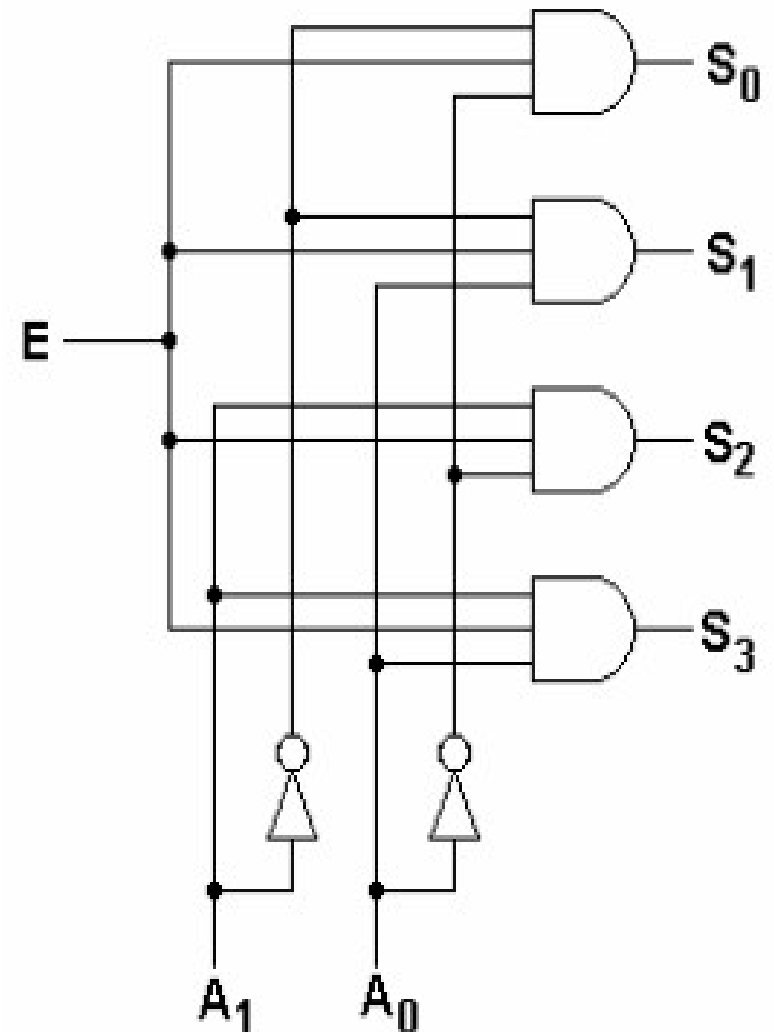
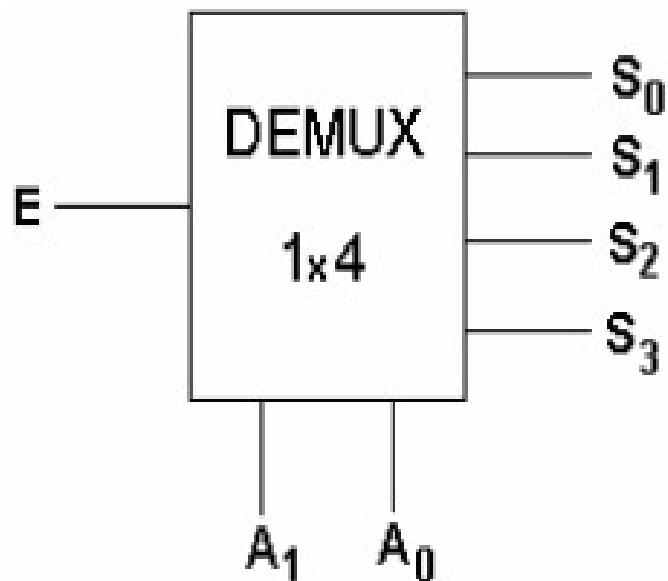


DEMULTIPLEXADORES

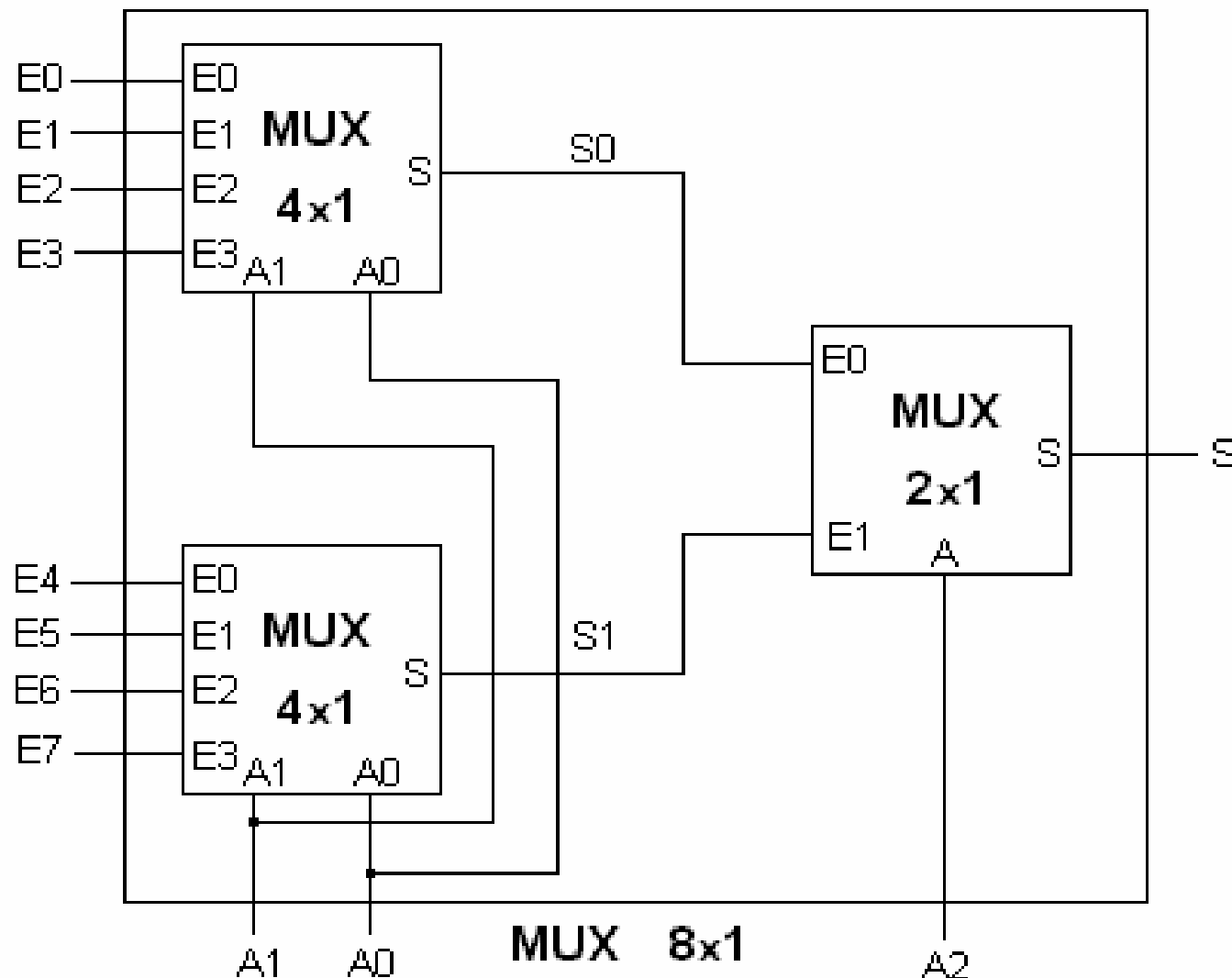
- Realizam a função inversa dos MUX.
- As entradas de controle dos demultiplexadores (ou “DEMUX”) decidem para qual das múltiplas saídas será jogada na única entrada.



DEMULTIPLEXADORES



MODULARIDADE DE MUX E DEMUX

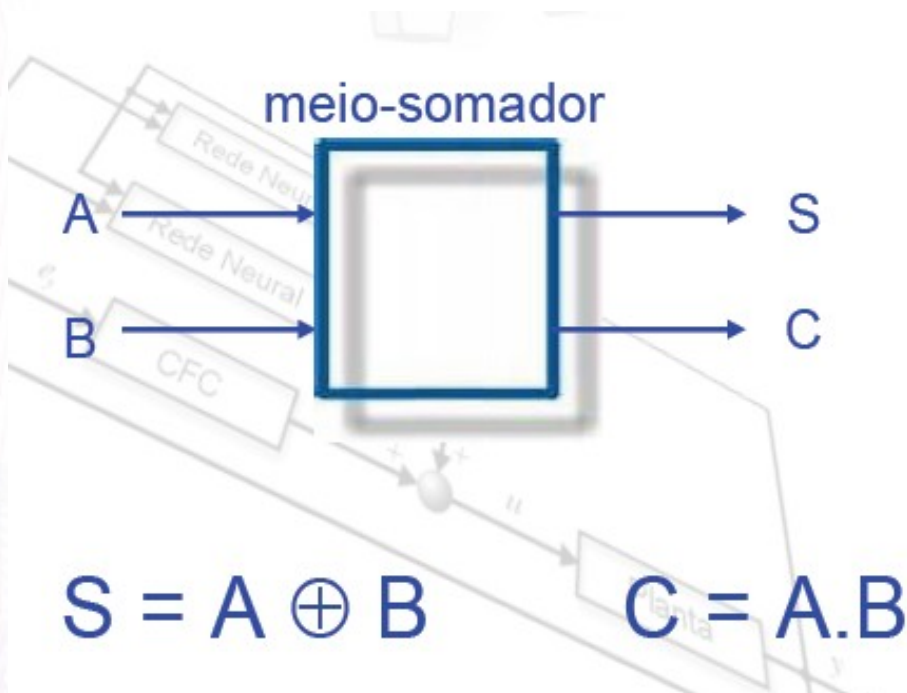


CIRCUITOS MEIO-SOMADOR E SOMADOR COMPLETO

- Realizam a soma binária inteira de duas palavras, **A** e **B** e devolvem a saída **S**.
- Lembrando que, em binários:
 - ◆ $0 + 0 = 0$
 - ◆ $0 + 1 = 1$
 - ◆ $1 + 0 = 1$
 - ◆ $1 + 1 = 0$ e “vai-um”

CIRCUITO MEIO-SOMADOR

- Para somar palavras de 1 bit, o circuito (chamado de “Meio-Somador”) seria:

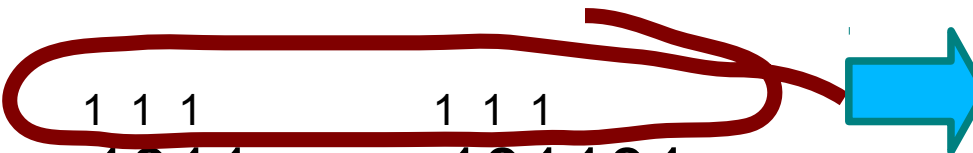


A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

O bit de
“Carry” C
é o
“vai-um”!

CIRCUITO SOMADOR COMPLETO

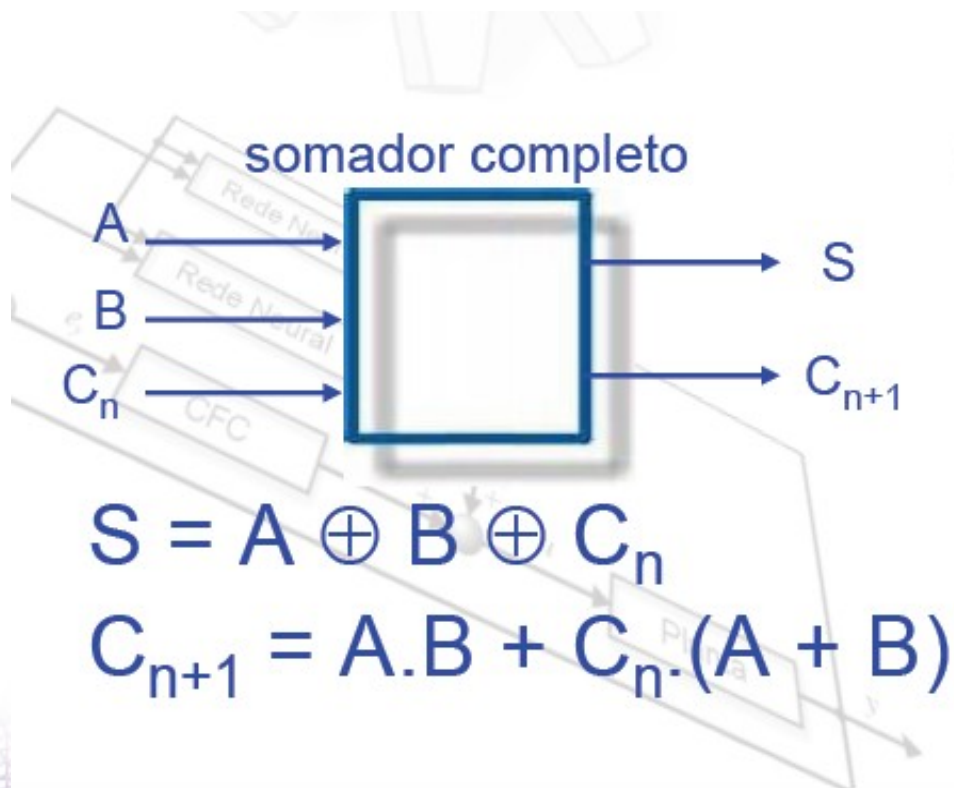
- Quando se trata de somar palavras de 2 bits, ou mais, o circuito vira um “Somador Completo”
- Exemplos de operação:

		
10	1011	101101
+ 01	+ 101	+ 110110
<hr/>	<hr/>	<hr/>
11	10000	1100011

Carry da casa binária n , entra na soma da casa binária $(n+1)$

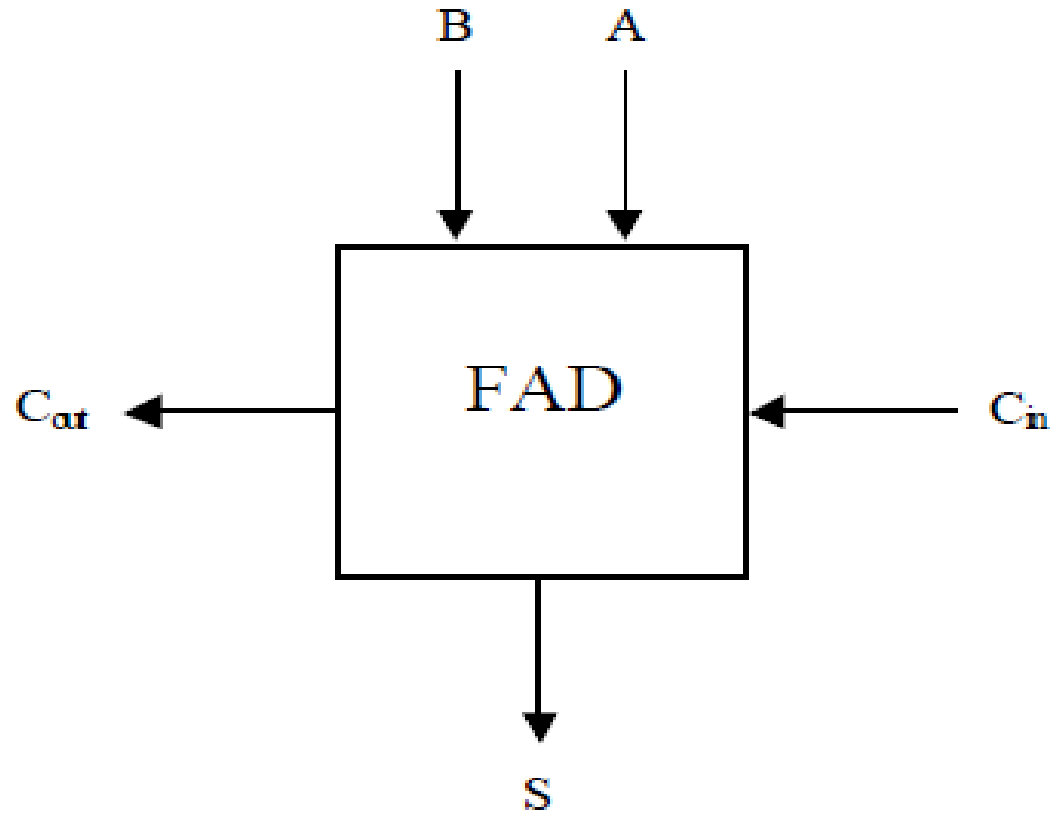
CIRCUITO SOMADOR COMPLETO

- Este leva em conta o carry da casa binária anterior, na soma dos bits atuais:



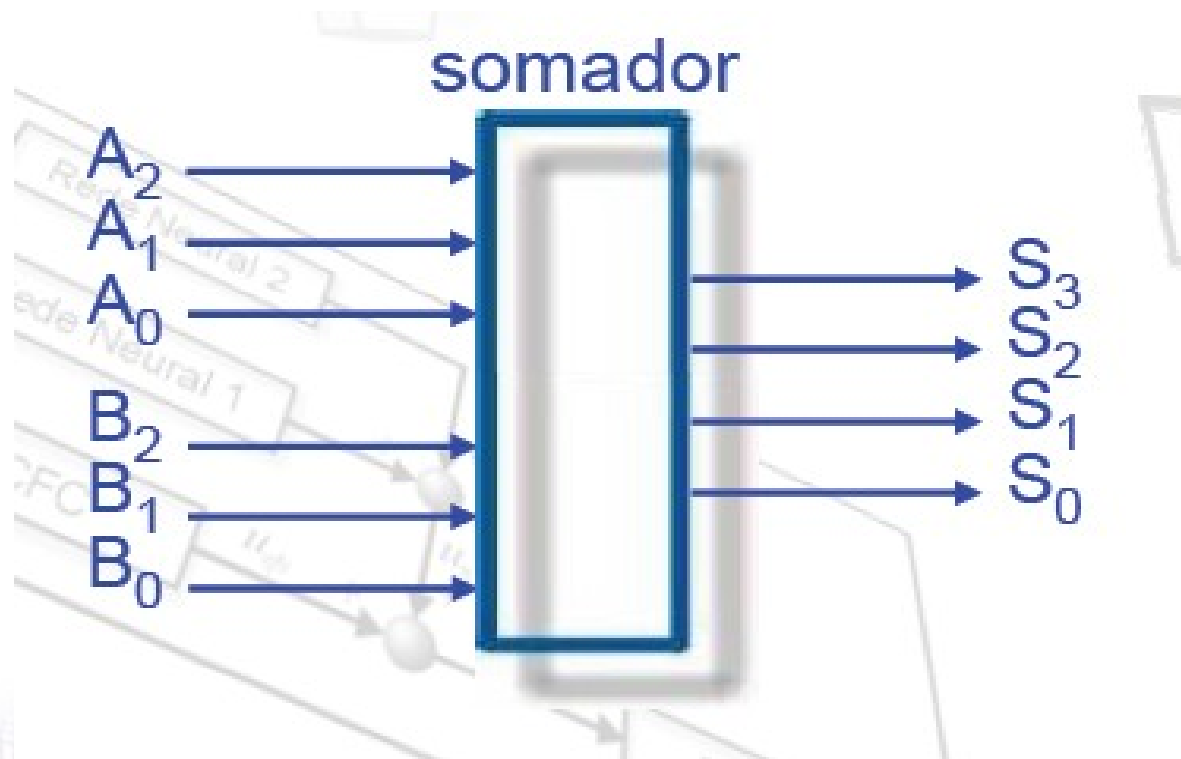
A	B	C_n	S	C_{n+1}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

CIRCUITO SOMADOR COMPLETO



CIRCUITOS SOMADOR COMPLETO

- Um somador de 3 bits é representado por:



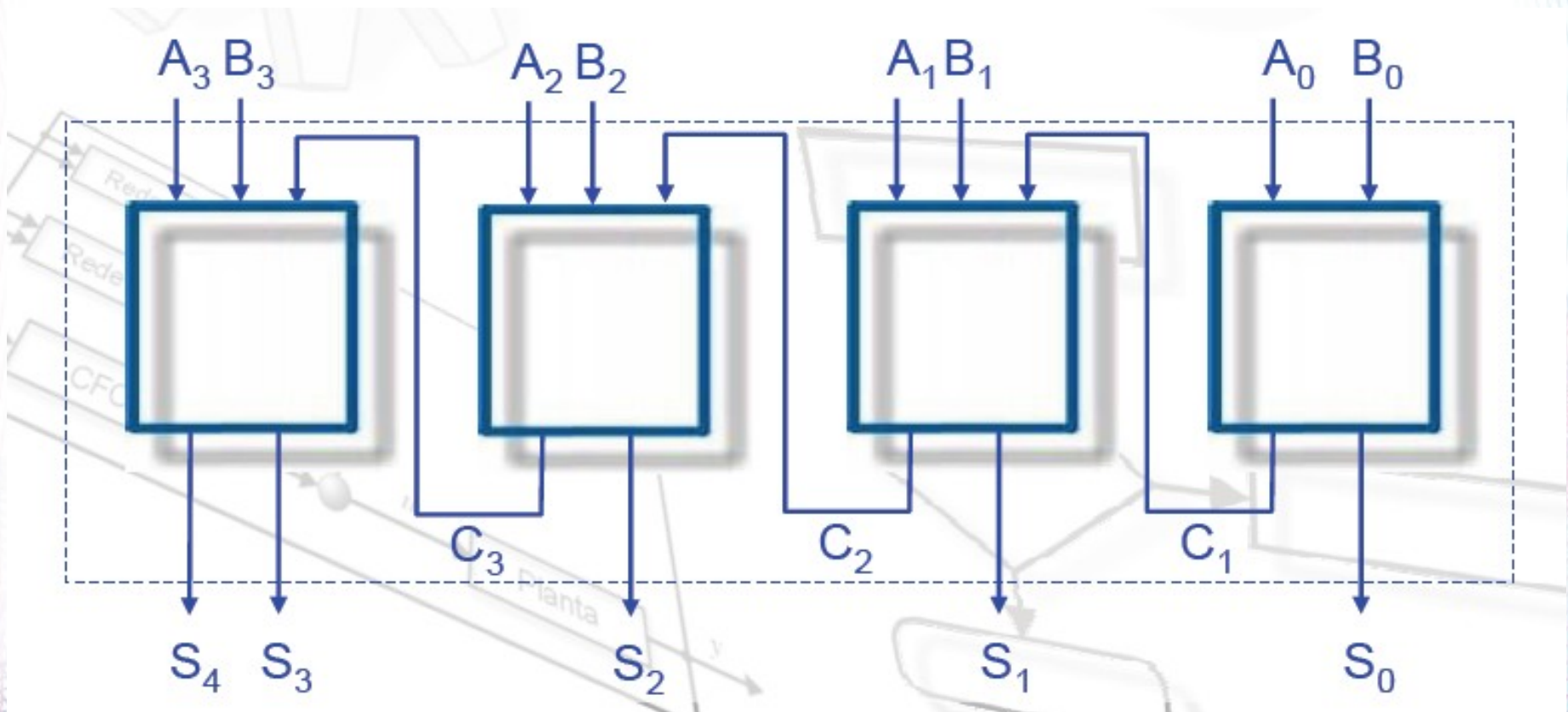
CIRCUITO SOMADOR COMPLETO

- Com a seguinte TV:

[illegible]

CIRCUITO SOMADOR COMPLETO

- E o somador de m bits genérico será do tipo:



CIRCUITO SOMADOR COMPLETO

- ou:

