

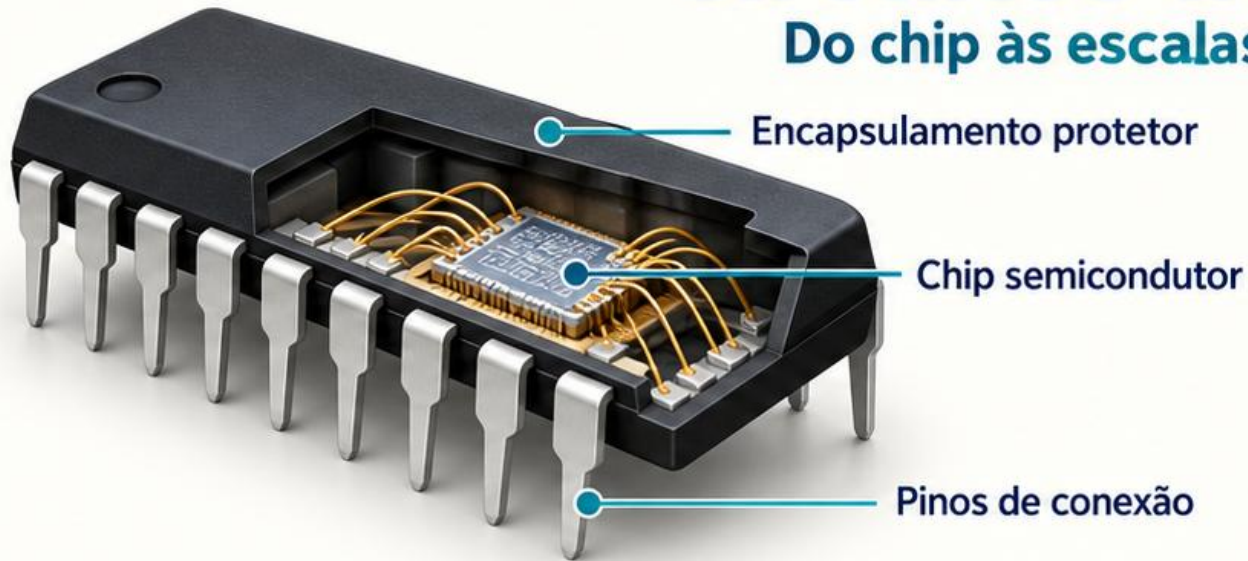
Circuitos combinacionais MSI

Sistemas Digitais

Prof MS Clayton J A Silva

Circuitos integrados

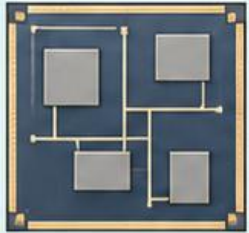
Do chip às escalas de integração



Um CI reúne elementos eletrônicos em um substrato semiconductor, protegido por um encapsulamento.

Os pinos conectam o circuito ao exterior.

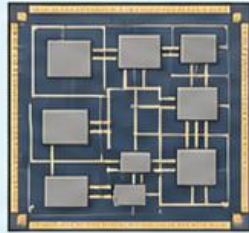
Grau de integração = número de portas lógicas no substrato



SSI

Pequena escala

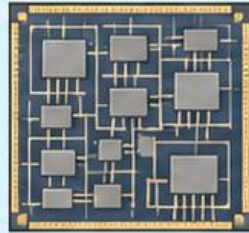
< 12 portas



MSI

Média escala

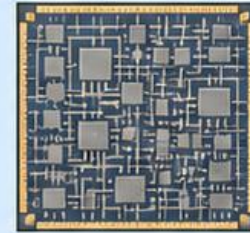
12–99 portas



LSI

Larga escala

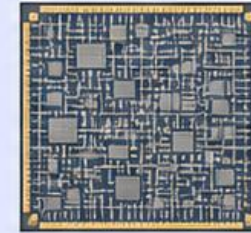
100–9.999 portas



VLSI

Muito larga escala

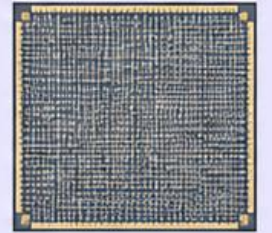
10.000–99.999
portas



ULSI

Ultralarga escala

100.000–999.999
portas



GSI

Gigaescala

> 1.000.000
portas

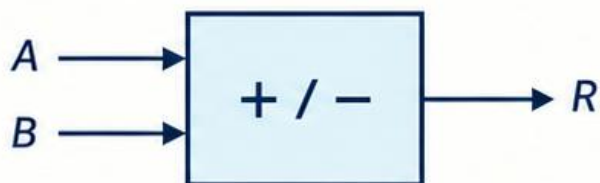
Mais portas integradas

Circuitos combinacionais

As saídas dependem apenas das entradas atuais, após o tempo de propagação.

Não armazenam estado.

Aritméticos



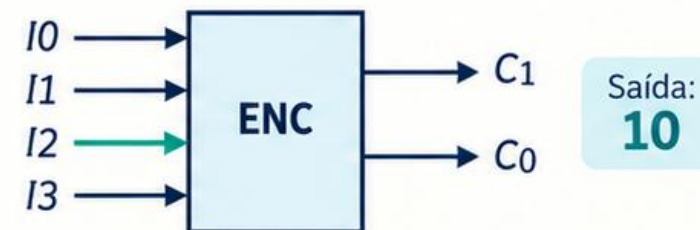
Realizam operações sobre números binários, como adição e subtração.

Comparadores



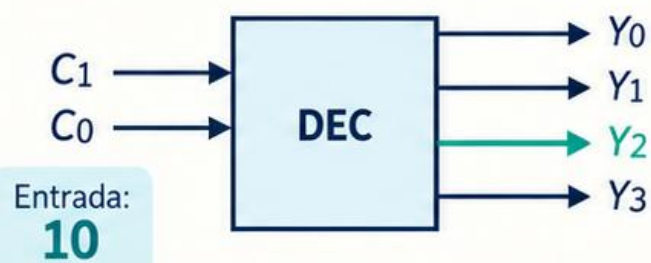
Comparam dois números binários e indicam sua relação de magnitude.

Codificadores (encoder)



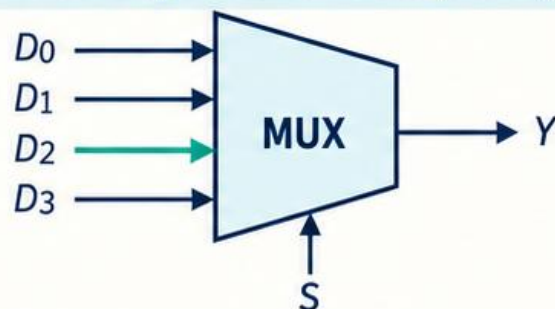
Geram um código binário que identifica a entrada ativa. Com prioridade: prevalece a entrada de maior prioridade.

Decodificadores (decoder)



Convertem um código de entrada na saída ou no padrão de saídas correspondente.

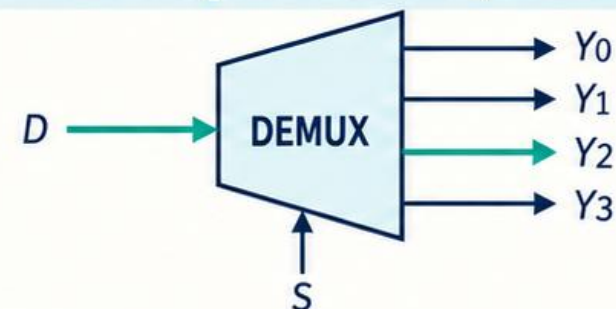
Multiplexadores (MUX)



Selecionam uma entre várias entradas de dados e a encaminham à saída.

MUX e DEMUX: o caminho é definido pelos sinais de seleção.

Demultiplexadores (DEMUX)



Encaminham uma entrada de dados para uma entre várias saídas.

MUX e DEMUX: o caminho é definido pelos sinais de seleção.

Circuitos aritméticos

Soma binária: bit a bit

Do bit menos significativo ao carry final

	Final	bit 4	bit 3	bit 2	bit 1	bit 0
Carry recebido		1	1	1	1	—
A		1	0	1	1	1
+ B		0	1	1	0	1
Resultado	1	0	0	1	0	0

↑
Carry final $C_5 = 1$

← Sentido do cálculo: bit 0 → bit 4

Soma de dois bits

Sem carry de entrada

A	B	Soma S	Carry C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

$$1 + 1 = 10_2 \rightarrow S = 0 \text{ e } C = 1$$

Bit 0 • meio-somador

Soma A_0 e B_0 . Não recebe carry.

Demais bits • somador completo

Somam os dois bits da posição e o carry recebido.

Conferência: $23 + 13 = 36 \rightarrow 100100_2$

Meio-somador

Soma dois bits de entrada, sem carry de entrada.

Portas AND, OR e NOT

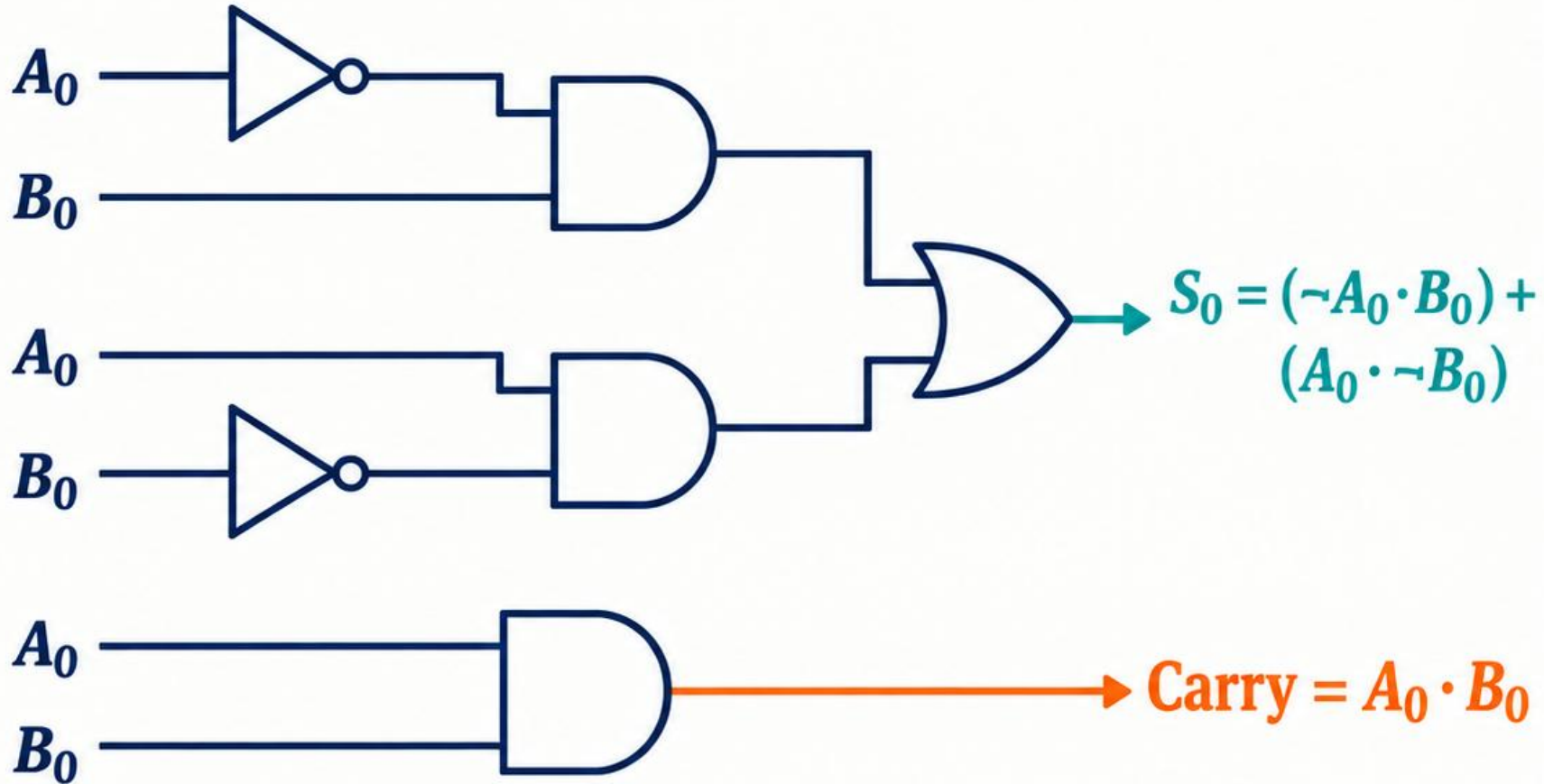


Tabela-verdade

A_0	B_0	S_0	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Somador completo

Soma dois bits e o carry recebido da posição anterior.

Portas AND, OR e NOT

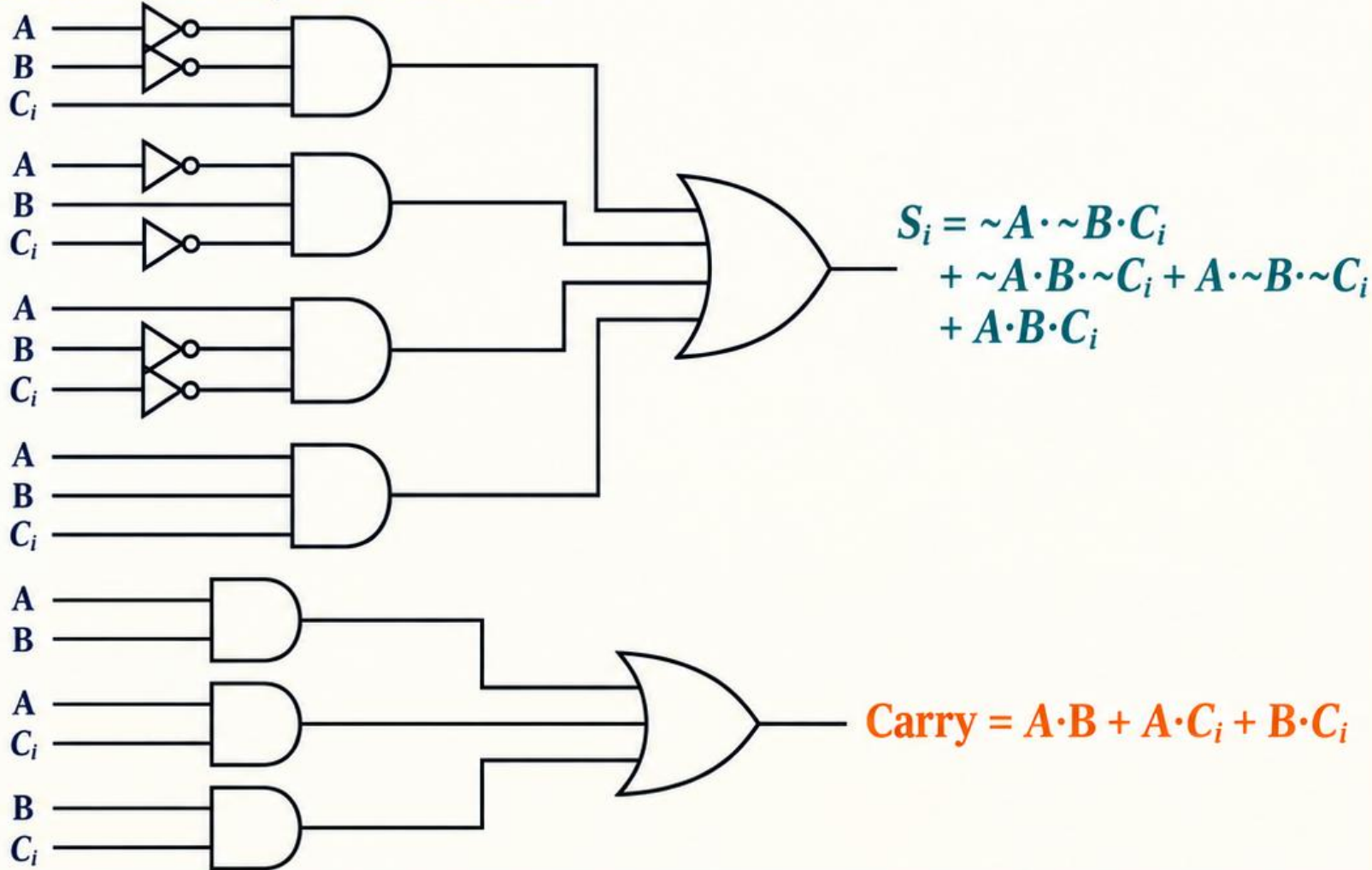
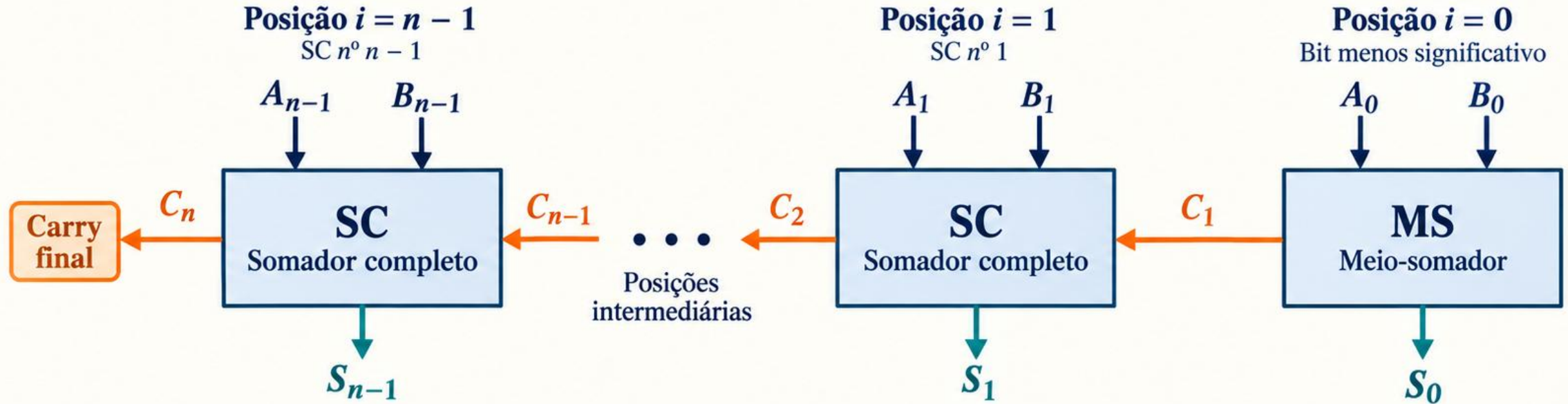


Tabela-verdade

A	B	C _i	S _i	Carry
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

Somador binário de n bits

1 meio-somador + $(n - 1)$ somadores completos



Resultado completo: $C_n S_{n-1} \cdots S_1 S_0$

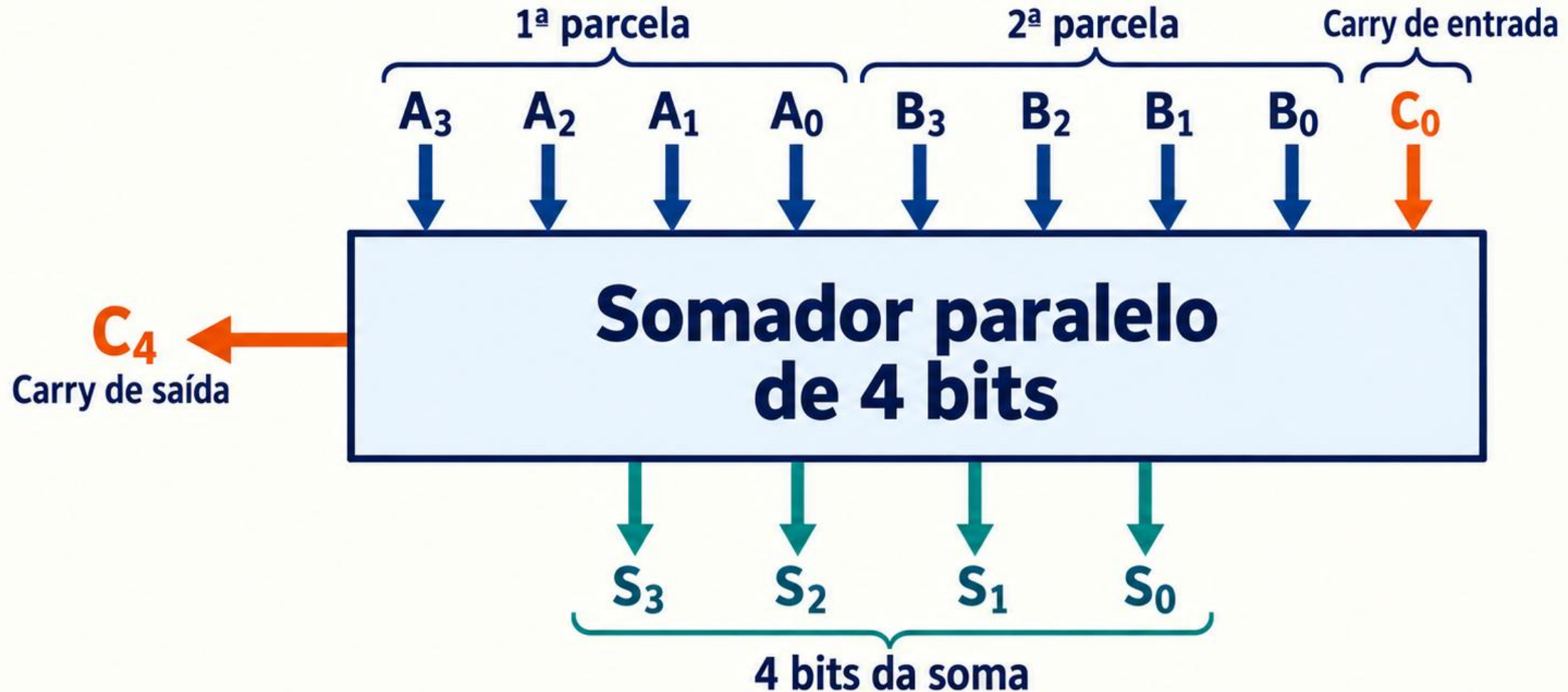
Até $n + 1$ bits

Overflow na soma sem sinal

Se o resultado for limitado a n bits, $C_n = 1$ indica overflow.
Mantendo C_n , o resultado completo é representado em $n + 1$ bits.

Em complemento de dois, o carry final isolado não caracteriza overflow.

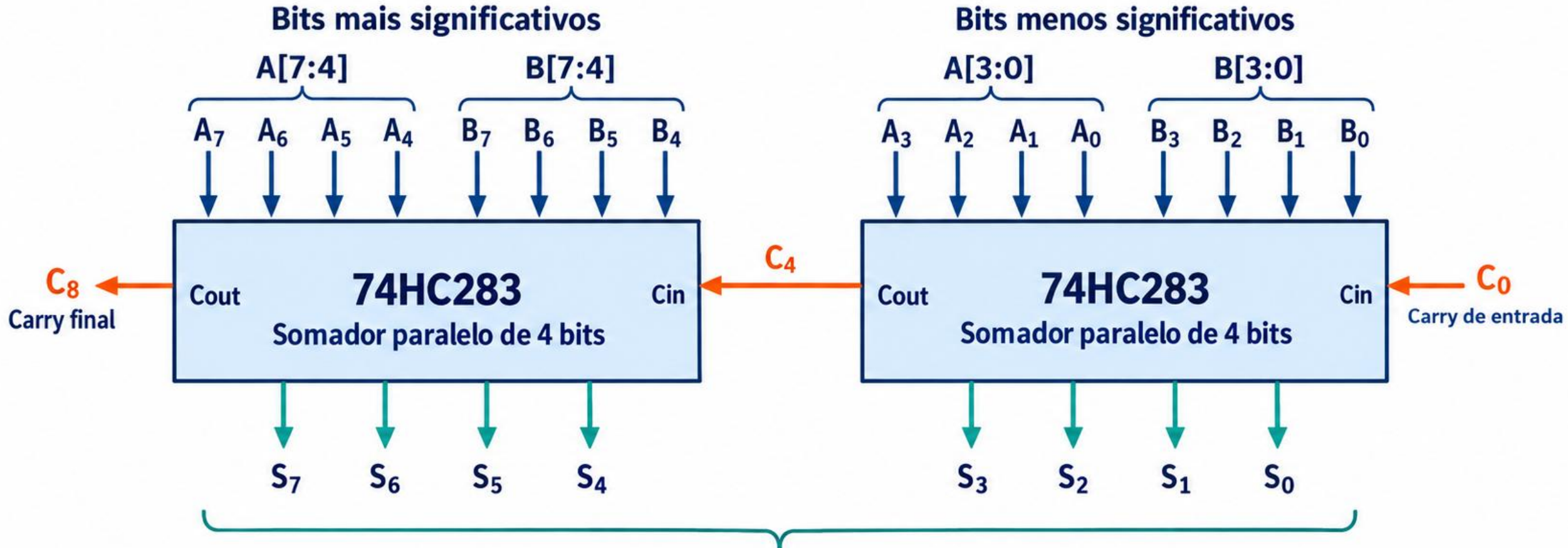
Somador binário de 4 bits



$$A + B + C_0 = (C_4 S_3 S_2 S_1 S_0)_2$$

Somador de 8 bits

Cascadeamento de dois somadores de 4 bits • 74HC283



Resultado de 8 bits: $S_7 S_6 S_5 S_4 S_3 S_2 S_1 S_0$

C_4 conecta o carry de saída do bloco menos significativo ao carry de entrada do próximo bloco.

$$A + B + C_0 = 256 \cdot C_8 + S$$

Somador BCD

Motorola • MC14560B • NBCD Adder

A ↓ / B →	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001
0000	0 0000 / 0 0001	0 0001 / 0 0010	0 0010 / 0 0011	0 0011 / 0 0100	0 0100 / 0 0101	0 0101 / 0 0110	0 0110 / 0 0111	0 0111 / 0 1000	0 1000 / 0 1001	0 1001 / 1 0000
0001	0 0001 / 0 0010	0 0010 / 0 0011	0 0011 / 0 0100	0 0100 / 0 0101	0 0101 / 0 0110	0 0110 / 0 0111	0 0111 / 0 1000	0 1000 / 0 1001	0 1001 / 1 0000	1 0000 / 1 0001
0010	0 0010 / 0 0011	0 0011 / 0 0100	0 0100 / 0 0101	0 0101 / 0 0110	0 0110 / 0 0111	0 0111 / 0 1000	0 1000 / 0 1001	0 1001 / 1 0000	1 0000 / 1 0001	1 0001 / 1 0010
0011	0 0011 / 0 0100	0 0100 / 0 0101	0 0101 / 0 0110	0 0110 / 0 0111	0 0111 / 0 1000	0 1000 / 0 1001	0 1001 / 1 0000	1 0000 / 1 0001	1 0001 / 1 0010	1 0010 / 1 0011
0100	0 0100 / 0 0101	0 0101 / 0 0110	0 0110 / 0 0111	0 0111 / 0 1000	0 1000 / 0 1001	0 1001 / 1 0000	1 0000 / 1 0001	1 0001 / 1 0010	1 0010 / 1 0011	1 0011 / 1 0100
0101	0 0101 / 0 0110	0 0110 / 0 0111	0 0111 / 0 1000	0 1000 / 0 1001	0 1001 / 1 0000	1 0000 / 1 0001	1 0001 / 1 0010	1 0010 / 1 0011	1 0011 / 1 0100	1 0100 / 1 0101
0110	0 0110 / 0 0111	0 0111 / 0 1000	0 1000 / 0 1001	0 1001 / 1 0000	1 0000 / 1 0001	1 0001 / 1 0010	1 0010 / 1 0011	1 0011 / 1 0100	1 0100 / 1 0101	1 0101 / 1 0110
0111	0 0111 / 0 1000	0 1000 / 0 1001	0 1001 / 1 0000	1 0000 / 1 0001	1 0001 / 1 0010	1 0010 / 1 0011	1 0011 / 1 0100	1 0100 / 1 0101	1 0101 / 1 0110	1 0110 / 1 0111
1000	0 1000 / 0 1001	0 1001 / 1 0000	1 0000 / 1 0001	1 0001 / 1 0010	1 0010 / 1 0011	1 0011 / 1 0100	1 0100 / 1 0101	1 0101 / 1 0110	1 0110 / 1 0111	1 0111 / 1 1000
1001	0 1001 / 1 0000	1 0000 / 1 0001	1 0001 / 1 0010	1 0010 / 1 0011	1 0011 / 1 0100	1 0100 / 1 0101	1 0101 / 1 0110	1 0110 / 1 0111	1 0111 / 1 1000	1 1000 / 1 1001

Em cada célula: saída para Cin = 0 / saída para Cin = 1

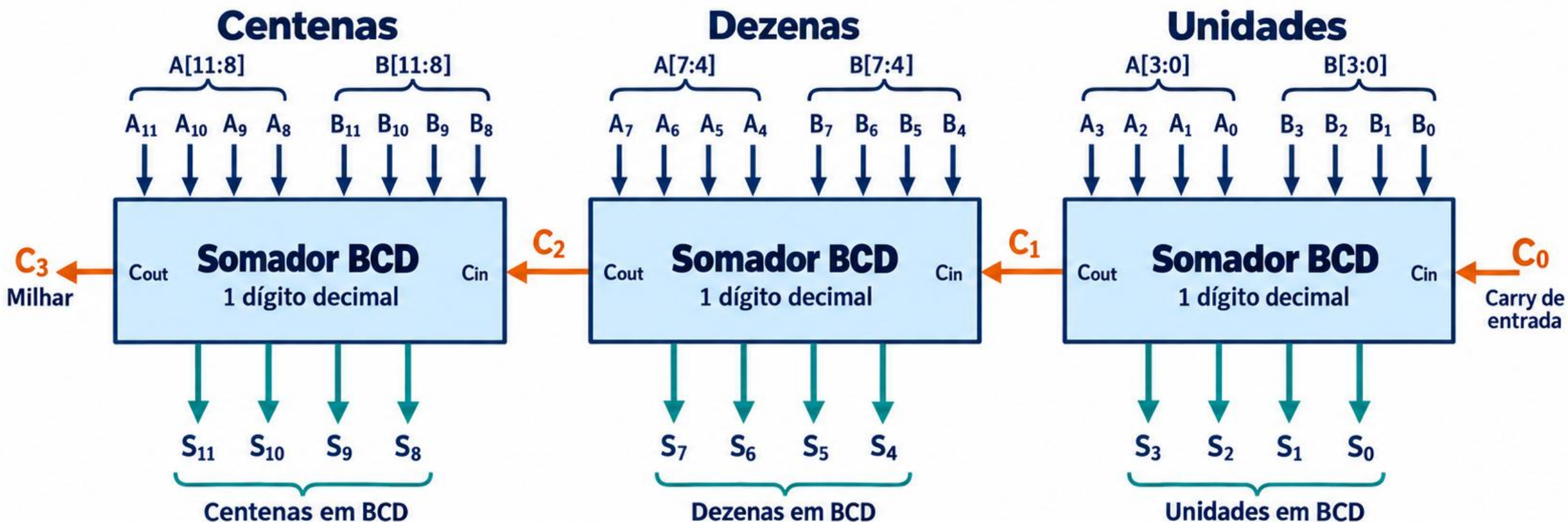
Saída: carry decimal (laranja) + 4 bits da soma BCD

O somador BCD pressupõe entradas entre 0000 e 1001.

Fora dessa faixa, não há garantia de soma correta; a saída não deve ser interpretada como uma soma binária pura.

Somador BCD de 3 dígitos

Cascadeamento de somadores BCD • unidades, dezenas e centenas



Cada grupo de 4 bits representa um dígito decimal (0000 a 1001).

O carry é decimal: unidades → dezenas → centenas → milhar.

Resultado: C_3 | centenas | dezenas | unidades

Circuitos comparadores

Comparador de dois dígitos binários

Entradas A e B: 1 bit cada

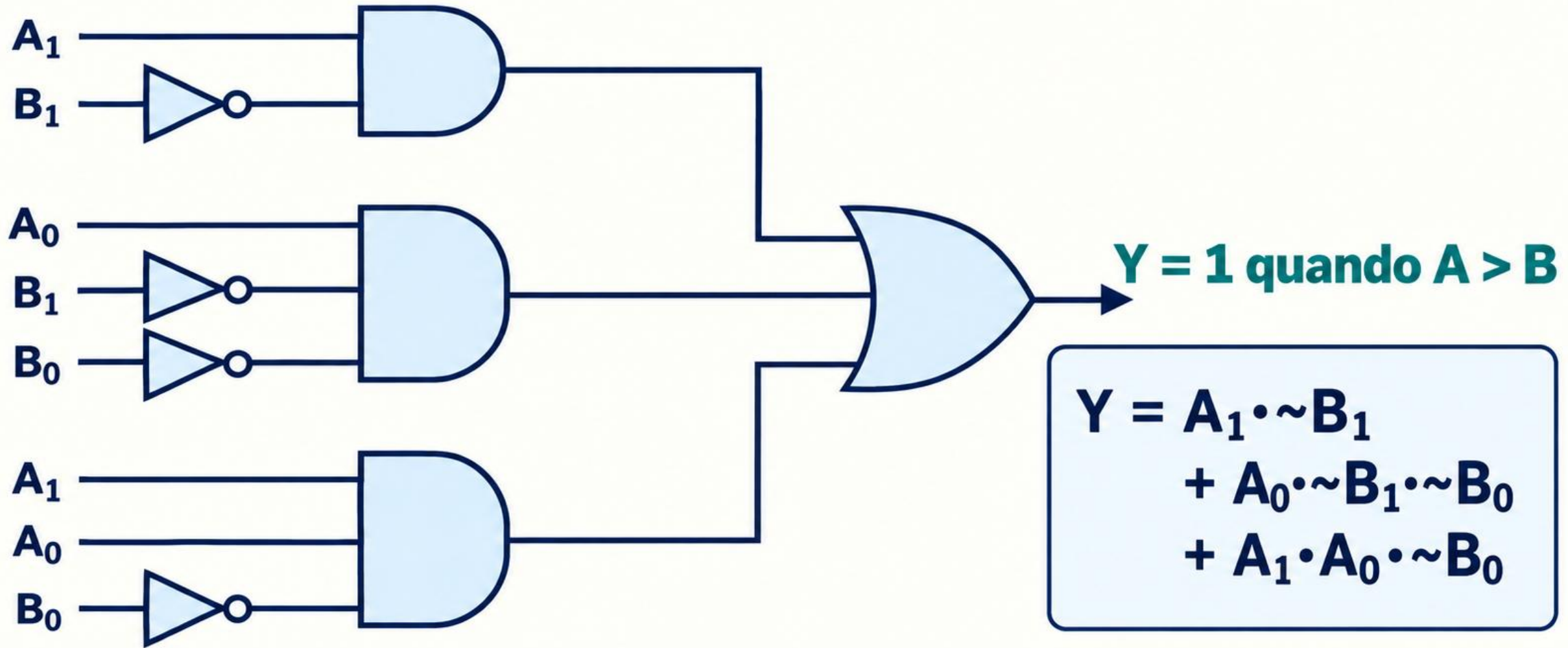


Saída 1: condição verdadeira
Saída 0: condição falsa

Desafio: projete o circuito que produz essas três saídas.

Comparador de 2 bits: $A > B$

$$A = A_1A_0 \quad \bullet \quad B = B_1B_0$$



Compare primeiro o bit mais significativo; se forem iguais, compare o bit menos significativo.

DM74LS85 • Comparador de 4 bits

National Semiconductor • 54LS85 / DM54LS85 / DM74LS85

Descrição

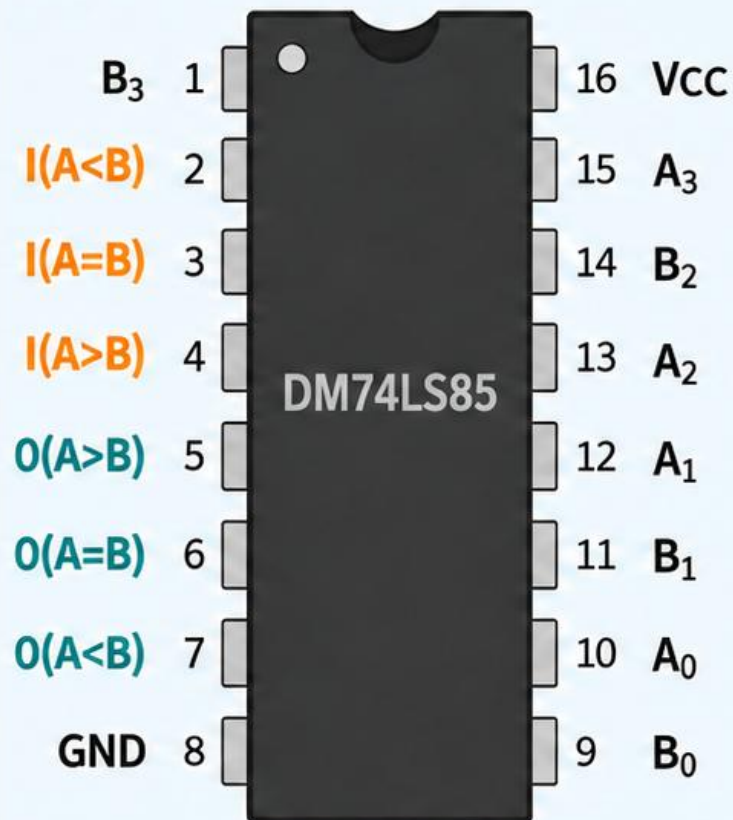
Compara duas palavras de 4 bits, em binário sem sinal ou BCD.

Três saídas ativas em nível alto: $A > B$, $A = B$ e $A < B$.

A_3 e B_3 são os bits mais significativos.

Permite cascata para comparar palavras maiores.

Pinagem • DIP-16



Vista superior • I = entrada; O = saída

Tabela funcional resumida

Comparação A:B	I(>,<=,<)	O(>)	O(=)	O(<)
$A > B$	X X X	1	0	0
$A < B$	X X X	0	0	1
$A = B$	1 0 0	1	0	0
$A = B$	X 1 X	0	1	0
$A = B$	0 0 1	0	0	1

X = indiferente

Se $A = B$ e $I(A = B) = 1$, a saída de igualdade é ativada; as outras entradas são indiferentes.

Cascata: saídas do bloco menos significativo → entradas do bloco mais significativo.

Uso isolado

$I(A > B) = 0$

$I(A = B) = 1$

$I(A < B) = 0$

Entradas de cascata
(pinos 4, 3 e 2).

Circuitos codificadores/decodificadores

Código binário: representando elementos discretos

Palavras de bits associadas a elementos de um conjunto

1 • Representação

Um código binário associa uma palavra de bits a cada elemento.

	Círculo	→	00
	Quadrado	→	01
	Triângulo	→	10
	Estrela	→	11

Os bits podem representar símbolos, estados ou categorias.

2 • Sem ambiguidade

Elementos diferentes devem ter códigos diferentes.

	Círculo	→	00		Colisão: 00 não identifica um único elemento.
	Quadrado	→	00		

Em palavras de tamanho fixo, a leitura deve respeitar os limites entre as palavras.

Em tamanho variável, a sequência também precisa ser decodificável de modo único.

3 • Número mínimo de bits

M elementos distintos • palavras de tamanho fixo

$$2^n \geq M \quad n_{\min} = \lceil \log_2 M \rceil \quad (M \geq 2)$$

4 elementos	→	2 bits
-------------	---	--------

5 elementos	→	3 bits
-------------	---	--------

Com 3 bits: 8 palavras possíveis; para 5 elementos, sobram 3.

4 • Distância de Hamming

Número de posições em que duas palavras diferem.

1	2	3	4	5	
1	0	1	1	0	(10110)
1	0	0	0	1	(10001)

$$d_H(10110, 10001) = 3$$

$d_{\min}$: menor distância entre palavras válidas

Detecta até $d_{\min} - 1$ erros de bit.

Corrige até $\lfloor (d_{\min} - 1)/2 \rfloor$ erros de bit.

Maior proteção exige redundância.

Código BCD 8421

Cada dígito decimal é representado separadamente por 4 bits.

Descrição geral

BCD significa **B**inary-**C**oded **D**ecimal: decimal codificado em binário. No código 8421, os bits têm pesos 8, 4, 2 e 1.



$$\text{Dígito} = 8b_3 + 4b_2 + 2b_1 + b_0$$

Exemplo: 59 em BCD



$59_{10} \rightarrow 0101\ 1001$ (BCD)

Em binário puro: $59_{10} = 111011_2$

Tabela de codificação

Decimal	Bits BCD			
	8	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Combinações não utilizadas

1010, 1011, 1100, 1101, 1110 e 1111
Não representam dígitos decimais no BCD 8421.

Código Gray • Distância unitária

UCD — Unit Distance Code

Descrição geral

No código Gray, palavras consecutivas diferem em **apenas um bit**. É um código **não ponderado**: os bits não têm pesos fixos como 8, 4, 2 e 1.

A tabela mostra o código Gray binário refletido de 4 bits.

Uma mudança por vez

Transição do decimal 7 para 8:

Binário:

0111 → 1000
4 bits mudam

Gray:

0100 → 1100
1 bit muda • distância de Hamming = 1

Aplicação típica



Em encoders de posição, reduz ambiguidades de leitura durante a transição entre posições adjacentes.

Distância unitária vale entre posições consecutivas; não entre quaisquer duas palavras.

Tabela de codificação • 4 bits

Decimal	Binário	Gray
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0111
6	0110	0101
7	0111	0100
8	1000	1100
9	1001	1101
10	1010	1111
11	1011	1110
12	1100	1010
13	1101	1011
14	1110	1001
15	1111	1000

Gray cíclico: Também no fechamento do ciclo: 1000 → 0000 (15 → 0).

Código ASCII

American Standard Code for Information Interchange

Código alfanumérico

Representa letras, algarismos e símbolos especiais.
O ASCII associa códigos a caracteres de texto e de controle.

7 bits → 128 códigos (0 a 127)

95 imprimíveis + 33 controles

95 caracteres imprimíveis

- códigos 32 – 126

Letras, algarismos, pontuação e símbolos; inclui o espaço.

33 caracteres de controle

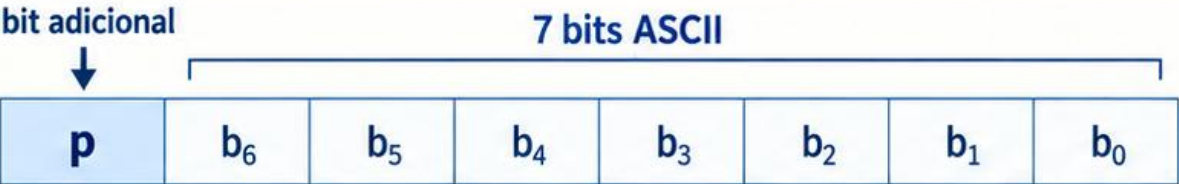
- códigos 0 – 31 e 127

Exemplos: NUL, LF e DEL.

Exemplo rápido

Caractere “5”: ASCII 0110101 (53) | Número 5: binário 101

ASCII e o oitavo bit



Exemplos em ASCII

Caractere	Decimal	Binário (7 bits)
LF (nova linha)	10	0001010
Espaço	32	0100000
5	53	0110101
A	65	1000001
a	97	1100001
DEL	127	1111111

Maiúsculas e minúsculas têm códigos diferentes.

O ASCII original usa **7 bits**. Em um formato de 8 bits, o bit adicional pode ser 0, servir à paridade ou integrar uma extensão, conforme a convenção.
O oitavo bit não faz parte do ASCII original; o bit b₆ faz parte do código.

Base: material da disciplina • seção 6.1.3 — Códigos alfanuméricos

Codificador com prioridade e indicação de validade

8 entradas • 9 estados codificados • 4 bits de saída

Diagrama em blocos



Prioridade das entradas:

$A_7 > A_6 > A_5 > A_4 > A_3 > A_2 > A_1 > A_0$

$$\lceil \log_2 9 \rceil = 4 \text{ bits}$$

8 entradas selecionáveis + nenhuma entrada ativa = 9 estados.

Tabela compacta • entradas ativas em 1

A_7	A_6	A_5	A_4	A_3	A_2	A_1	A_0	V	O_2	O_1	O_0
1	X	X	X	X	X	X	X	1	1	1	1
0	1	X	X	X	X	X	X	1	1	1	0
0	0	1	X	X	X	X	X	1	1	0	1
0	0	0	1	X	X	X	X	1	1	0	0
0	0	0	0	1	X	X	X	1	0	1	1
0	0	0	0	0	1	X	X	1	0	1	0
0	0	0	0	0	0	1	X	1	0	0	1
0	0	0	0	0	0	0	1	1	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0

X = 0 ou 1: não altera a seleção de maior prioridade.

0000 → nenhuma entrada ativa

1000 → A_0 selecionada

V = 1: existe entrada ativa • V = 0: nenhuma entrada ativa

As 256 combinações de entrada são agrupadas em 9 estados pela regra de prioridade.

Codificador com prioridade • 3 teclas

3 teclas + nenhuma pressionada = 4 estados → 2 bits

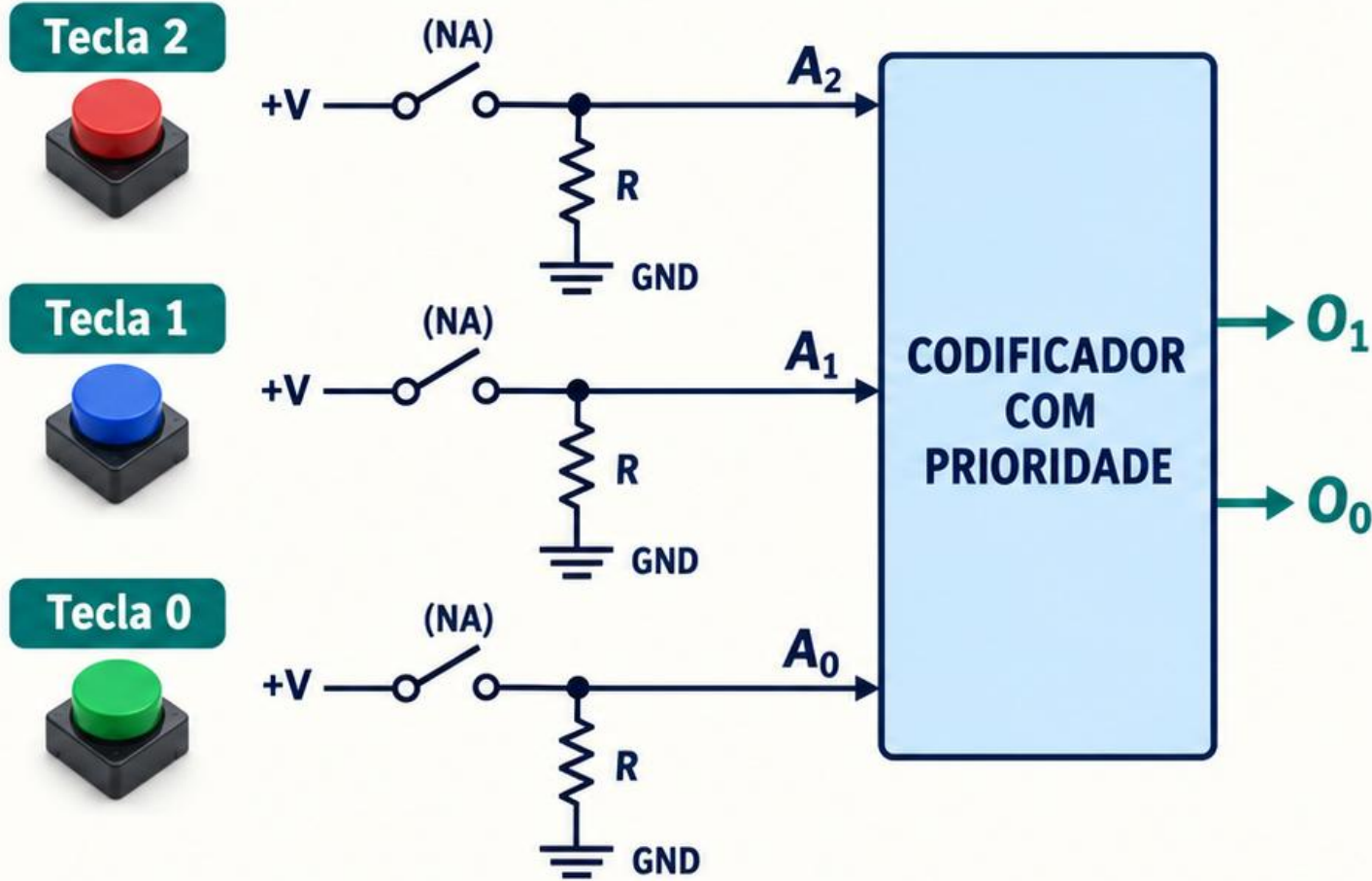


Tabela-verdade compacta

A ₂	A ₁	A ₀	O ₁	O ₀	Seleção
1	X	X	1	1	Tecla 2
0	1	X	1	0	Tecla 1
0	0	1	0	1	Tecla 0
0	0	0	0	0	Nenhuma

X = 0 ou 1: entrada de menor prioridade indiferente.
As 4 linhas resumem as 8 combinações de entrada.

Solta = 0 • Pressionada = 1
Prioridade: Tecla 2 > Tecla 1 > Tecla 0

00: nenhuma • 01: Tecla 0 • 10: Tecla 1 • 11: Tecla 2

Decodificador binário (decoder)

Um código de entrada $\rightarrow$ uma única saída ativa

Circuito combinacional que interpreta uma palavra de m bits e ativa a saída correspondente.



Decodificação completa: m entradas $\rightarrow 2^m$ saídas

Exemplo: decodificador $2 \rightarrow 4$

A_1	A_0	O_3	O_2	O_1	O_0
0	0	0	0	0	1
0	1	0	0	1	0
1	0	0	1	0	0
1	1	1	0	0	0

$$A_1 A_0 = 10 \rightarrow O_2 = 1$$

As demais saídas permanecem em 0.

Saídas ativas em nível alto • decodificador habilitado

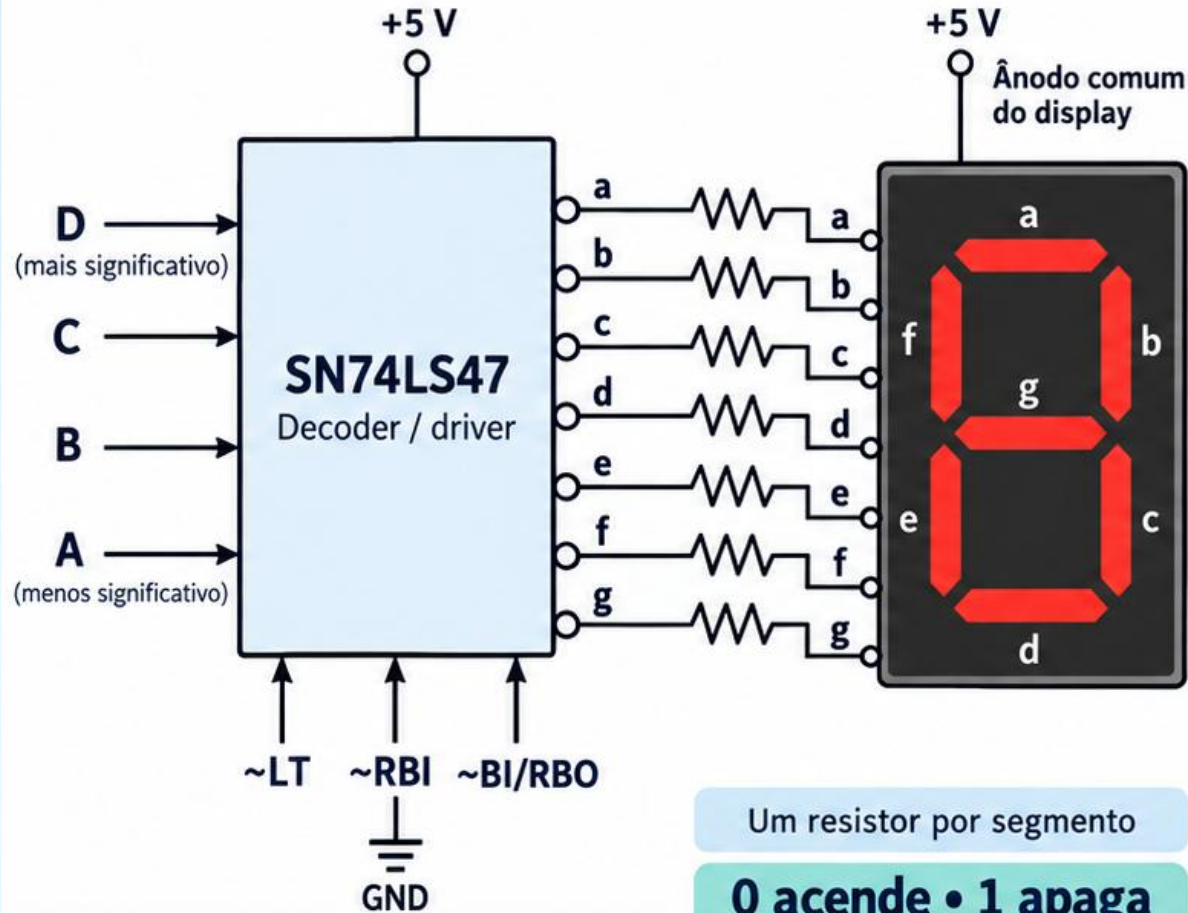
Para cada combinação de entrada, exatamente uma saída fica em 1.

A palavra 00 também seleciona uma saída: O_0 .

Decodificador BCD → 7 segmentos

SN74LS47 • Display de ânodo comum

Diagrama conceitual



Um resistor por segmento

0 acende • 1 apaga

(níveis lógicos medidos com o display conectado)

~ indica controle ativo em 0.

Tabela verdade resumida

Função	~LT	~RBI	~BI ¹	DCBA	a	b	c	d	e	f	g
0	1	1	1	0000	0	0	0	0	0	0	1
1	1	X	1	0001	1	0	0	1	1	1	1
2	1	X	1	0010	0	0	1	0	0	1	0
3	1	X	1	0011	0	0	0	0	1	1	0
4	1	X	1	0100	1	0	0	1	1	0	0
5	1	X	1	0101	0	1	0	0	1	0	0
6	1	X	1	0110	1	1	0	0	0	0	0
7	1	X	1	0111	0	0	0	1	1	1	1
8	1	X	1	1000	0	0	0	0	0	0	0
9	1	X	1	1001	0	0	0	1	1	0	0
Apagar	X	X	0	XXXX	1	1	1	1	1	1	1
Teste	0	X	1	XXXX	0	0	0	0	0	0	0
Suprimir 0	1	0	1	0000	1	1	1	1	1	1	1

¹ ~BI: comando externo; 1 = sem forçar apagamento. Na supressão do zero, ~RBO fica em 0.

X = 0 ou 1: não altera o resultado.

1010 a 1111: fora do BCD; padrões específicos do CI, omitidos neste resumo.

7442 • Decodificador BCD para decimal

4 entradas • 10 saídas ativas em 0

Diagrama em blocos



Entrada válida: uma saída em 0; as demais em 1.

D: bit mais significativo • A: bit menos significativo

Tabela verdade

D	C	B	A	Saída ativa (em 0)
0	0	0	0	$\sim O_0$
0	0	0	1	$\sim O_1$
0	0	1	0	$\sim O_2$
0	0	1	1	$\sim O_3$
0	1	0	0	$\sim O_4$
0	1	0	1	$\sim O_5$
0	1	1	0	$\sim O_6$
0	1	1	1	$\sim O_7$
1	0	0	0	$\sim O_8$
1	0	0	1	$\sim O_9$
1	0	1	0	Nenhuma
1	0	1	1	Nenhuma
1	1	0	0	Nenhuma
1	1	0	1	Nenhuma
1	1	1	0	Nenhuma
1	1	1	1	Nenhuma

0 = nível BAIXO • 1 = nível ALTO

1010 a 1111: entradas inválidas em BCD → todas as saídas permanecem em 1.

~ indica saída ativa em nível baixo.

Conversor de códigos • um único projeto

Dois sistemas representam os mesmos **N** elementos com códigos diferentes



Exemplo: os mesmos 4 elementos

Elemento	Entrada A	Saída B
● Círculo	00	10
■ Quadrado	01	11
▲ Triângulo	10	00
★ Estrela	11	01

Exemplo: 01 corresponde ao quadrado



O projeto recebe uma palavra do sistema1 e gera a palavra correspondente para o sistema2.

$$2^m \geq N \text{ e } 2^p \geq N$$

Os códigos A e B podem ter comprimentos diferentes.

Cada palavra de saída é um conjunto de **p bits**.