



# ARQUITETURA E ORGANIZAÇÃO DE COMPUTADORES

ENGENHARIA DA COMPUTAÇÃO - UFC/SOBRAL

Prof. Wendley S. Silva

# O NÍVEL LÓGICO DIGITAL



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Introdução
- Portas e álgebra booleana
- Circuitos integrados
- Circuitos combinatórios
- Circuitos aritméticos
- Clocks
- Memórias de 1 bit
- Flip-flops
- Registradores



# INTRODUÇÃO

CIRCUITO DIGITAL – DOIS VALORES LÓGICOS - TRANSISTORES – COMUTADOR BINÁRIO

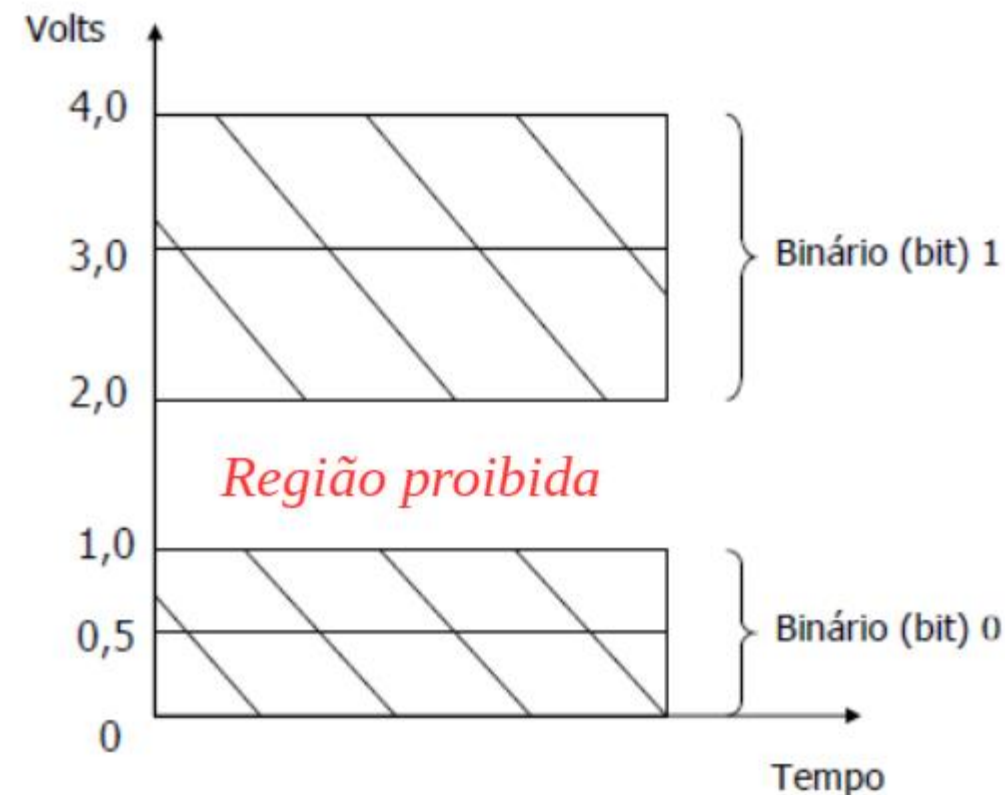


# CIRCUITO DIGITAL



UNIVERSIDADE  
FEDERAL DO CEARÁ

- A informação binária (0 ou 1) é representada em um sistema digital por sinais elétricos em dois níveis de intensidade, cada um correspondendo a um valor binário:
  - 0  $\rightarrow$  0,5 V variando de [0,0:1,0] V
  - 1  $\rightarrow$  3,0 V variando de [2,0:4,0] V

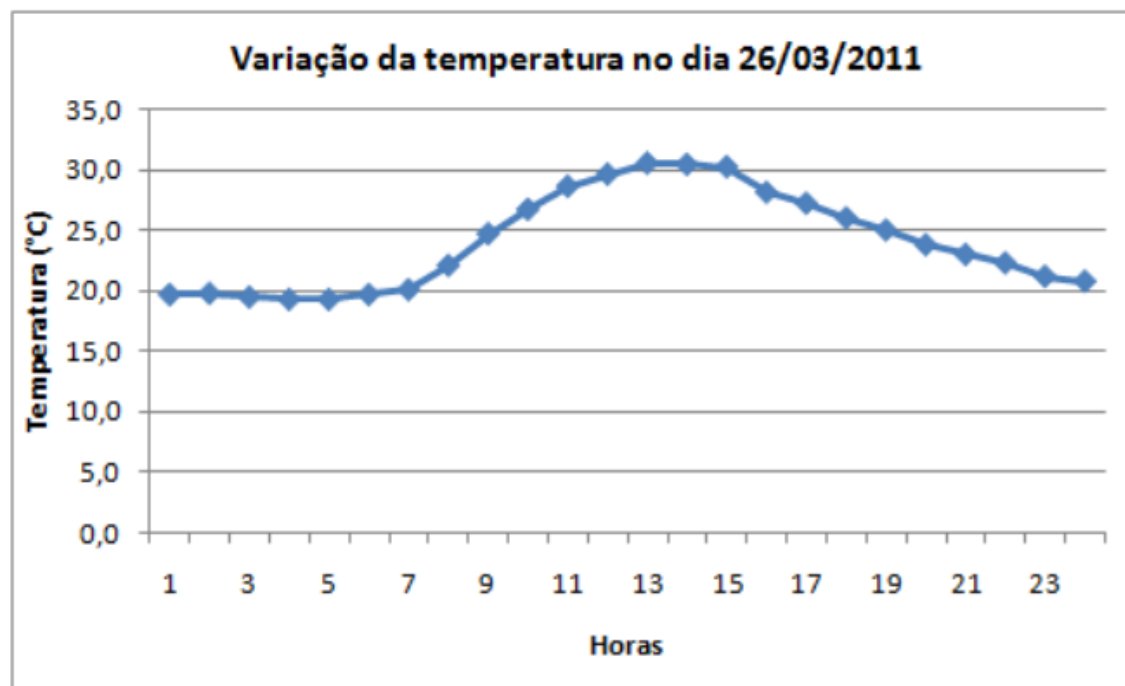


# GRANDEZAS ANALÓGICA E DIGITAL



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Uma grandeza **analógica** é aquela que apresenta valores contínuos
  - Ex.: a temperatura do ambiente varia numa faixa contínua de valores – durante um determinado dia, a temperatura não passa, digamos, de 24°C para 25° C instantaneamente; ela passa por uma infinidade de valores intermediários

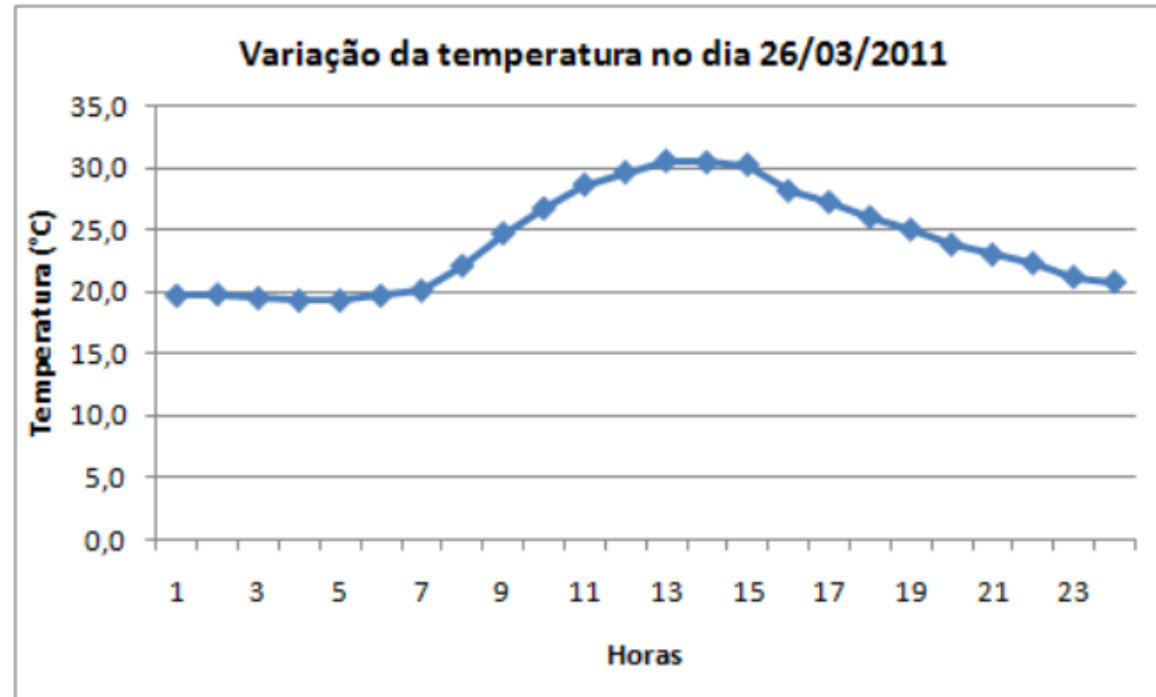


# GRANDEZAS ANALÓGICA E DIGITAL



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Uma grandeza **digital** é aquela que apresenta valores
  - Ex.: Se fizemos a leitura da temperatura apenas a cada hora, aí sim podemos ter variações instantâneas (no instante da leitura) de um valor inteiro para outro – de 7 para 9 horas, a temperatura muda de 20° C para 25° C



# GRANDEZAS ANALÓGICA E DIGITAL



UNIVERSIDADE  
FEDERAL DO CEARÁ

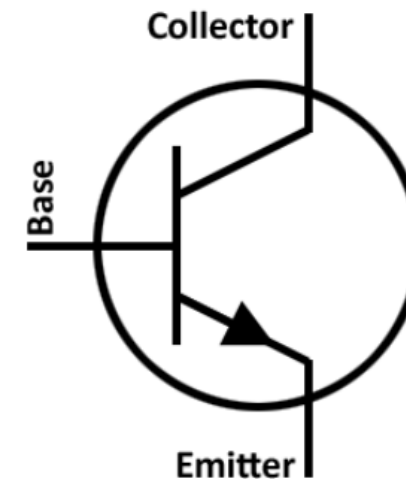
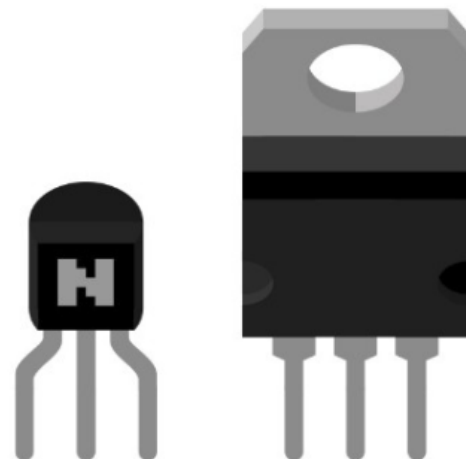
- O termo digital é derivado da forma com que os computadores realizam operações:  
**contando dígitos**

# TRANSISTOR



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Pode funcionar como um amplificador ou chave
  - No modo amplificador ele possui a capacidade de ampliar o nível de tensão
  - Como chave ele permite ligar cargas em sua saída
- Apenas tensão contínua







# PORTAS E ÁLGEBRA BOOLEANA

PORTAS (GATES) LÓGICAS E A ÁLGEBRA DE BOOLE

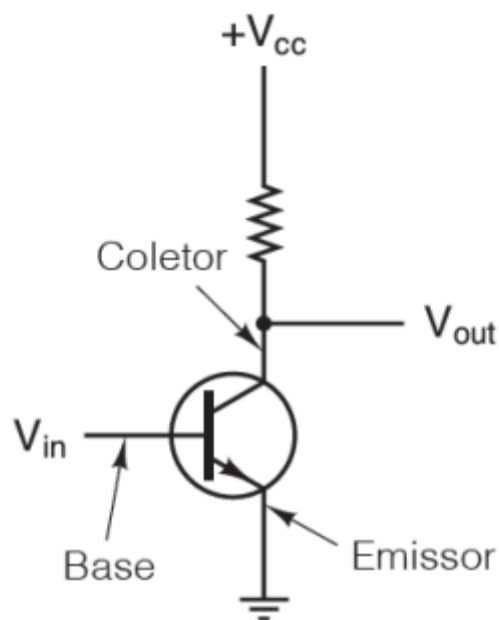


# PORTAS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Minúsculos dispositivos eletrônicos, denominados portas (gates), podem calcular várias funções dos sinais de dois valores.
  - Essas portas formam a base do hardware sobre a qual todos os computadores digitais são construídos



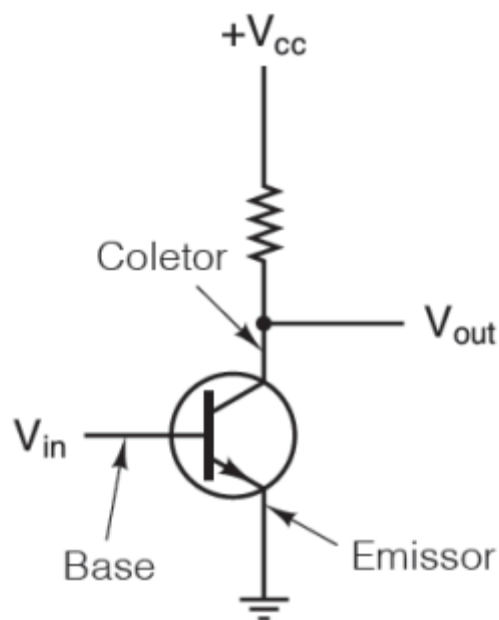
Quando a voltagem de entrada,  $V_{in}$ , está abaixo de certo valor crítico, o transistor desliga e age como uma resistência infinita. Isso faz com que a saída do circuito,  $V_{out}$ , assumam um valor próximo a  $V_{cc}$ , uma voltagem regulada externamente. Quando  $V_{in}$  excede o valor crítico, o transistor liga e age como um fio, fazendo  $V_{out}$  ficar conectado com a terra (por convenção, 0 V).

# PORTAS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Minúsculos dispositivos eletrônicos, denominados portas (gates), podem calcular várias funções dos sinais de dois valores.
  - Essas portas formam a base do hardware sobre a qual todos os computadores digitais são construídos



Quando  $V_{in}$  é baixa,  $V_{out}$  é alta, e vice-versa.

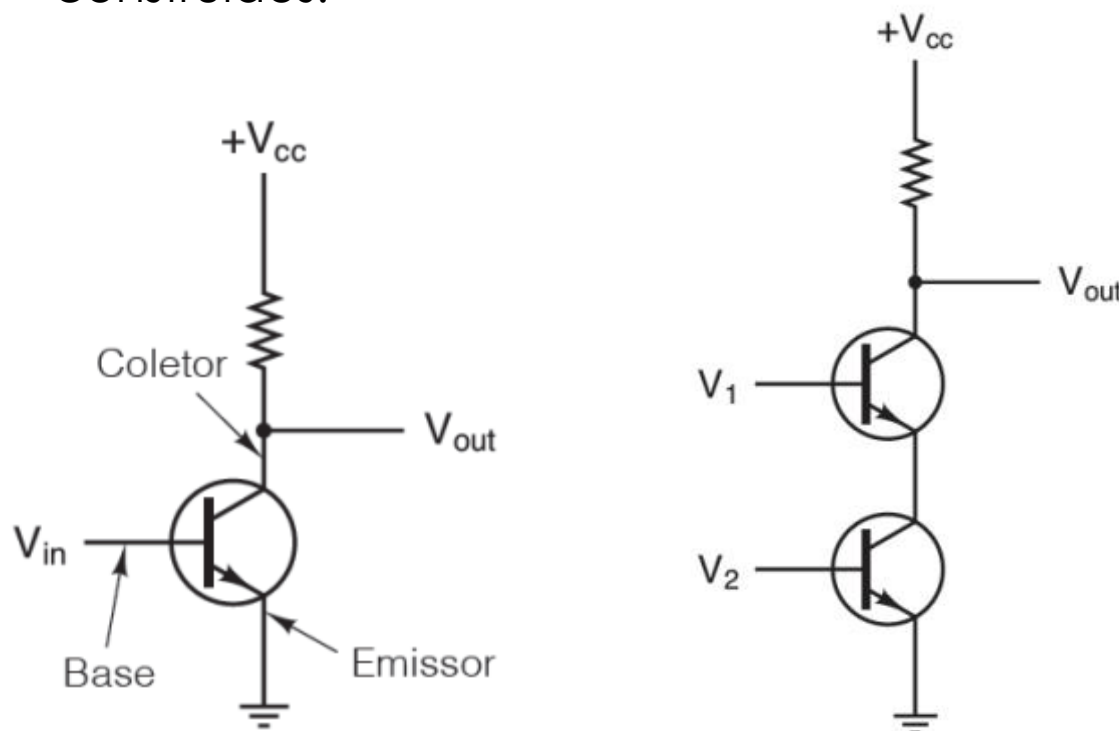
**Inversor de transistor**

# PORTAS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Minúsculos dispositivos eletrônicos, denominados portas (gates), podem calcular várias funções dos sinais de dois valores.
  - Essas portas formam a base do hardware sobre a qual todos os computadores digitais são construídos.



Se ambas,  $V_1$  e  $V_2$ , forem altas, ambos os transistores conduzirão e  $V_{out}$  cairá. Se qualquer das entradas for baixa, o transistor correspondente se desligará e a saída será alta.

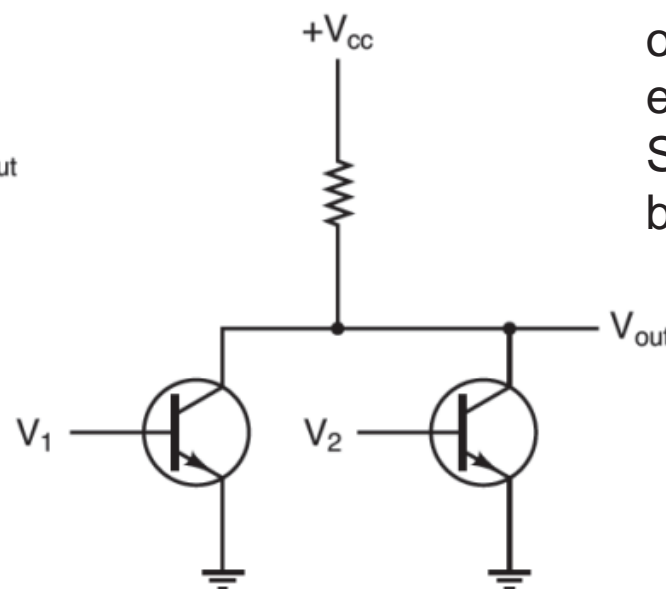
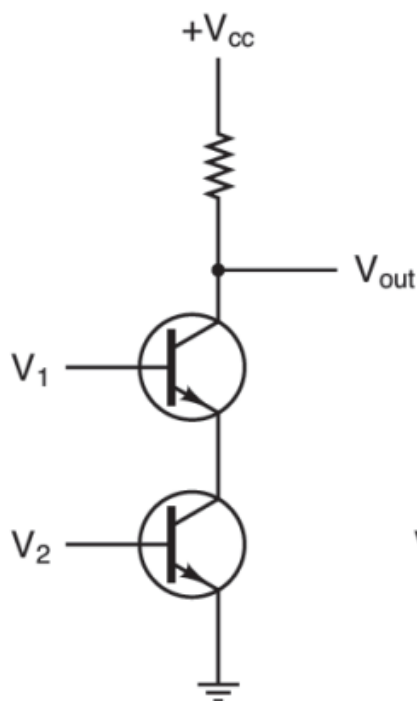
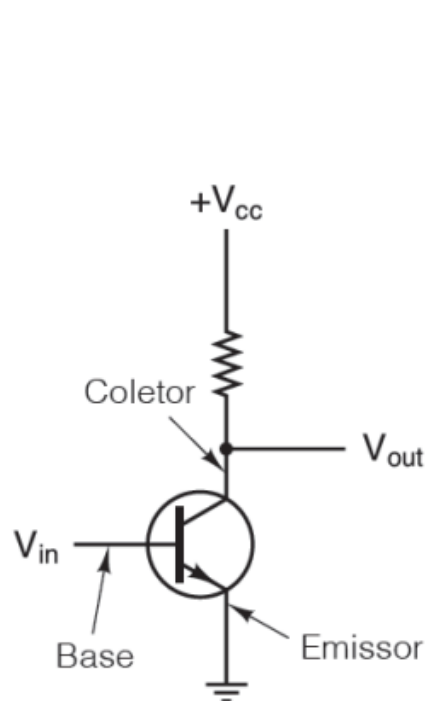
**Porta NAND**

# PORTAS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Minúsculos dispositivos eletrônicos, denominados portas (gates), podem calcular várias funções dos sinais de dois valores.
  - Essas portas formam a base do hardware sobre a qual todos os computadores digitais são construídos



Se qualquer das entradas for alta, o transistor correspondente ligará e conectará a saída com a terra. Se ambas as entradas forem baixas, a saída permanecerá alta.

**Porta NOR**

# PORTAS E ÁLGEBRA BOOLEANA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Variável: A, B, C, ... que podem assumir valores Verdadeiros (1) ou Falsos (0)
- A negação de uma variável é representada por uma barra superior horizontal ou um apóstrofo. É possível ainda utilizar o til ~

$$\overline{A} \quad B' \quad \sim C$$

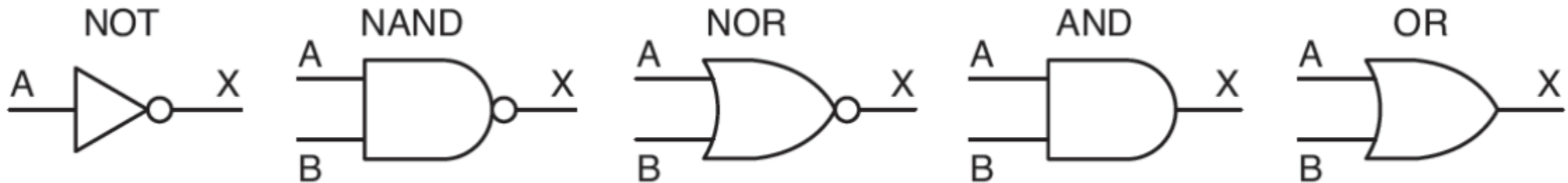
- As operações AND e OR são representadas pelo sinal da multiplicação e da soma, respectivamente
  - $AB$  ou  $A \cdot B \rightarrow A \text{ AND } B$
  - $C+D \rightarrow C \text{ OR } D$
- Ex.:
  - $F = A + B \cdot C$

# PORTAS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Símbolos e comportamento funcional das cinco portas básicas

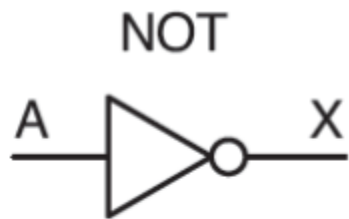


# TABELA VERDADE

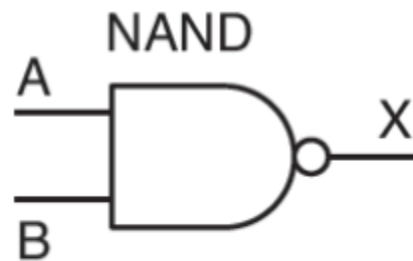


UNIVERSIDADE  
FEDERAL DO CEARÁ

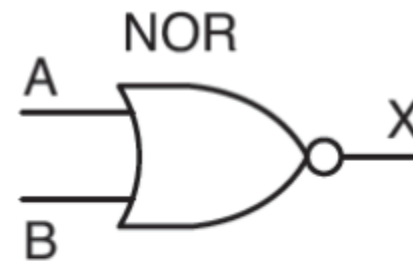
- Descreve as saídas para cada combinação em uma função booleana



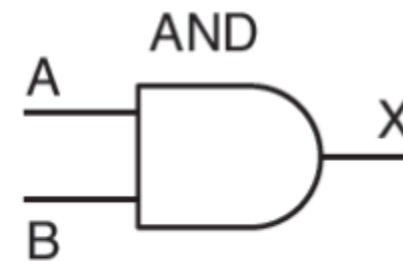
A	X
0	1
1	0



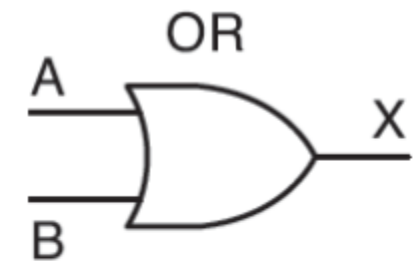
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0



A	B	X
0	0	1
0	1	0
1	0	0
1	1	0



A	B	X
0	0	0
0	1	0
1	0	0
1	1	1



A	B	X
0	0	0
0	1	1
1	0	1
1	1	1



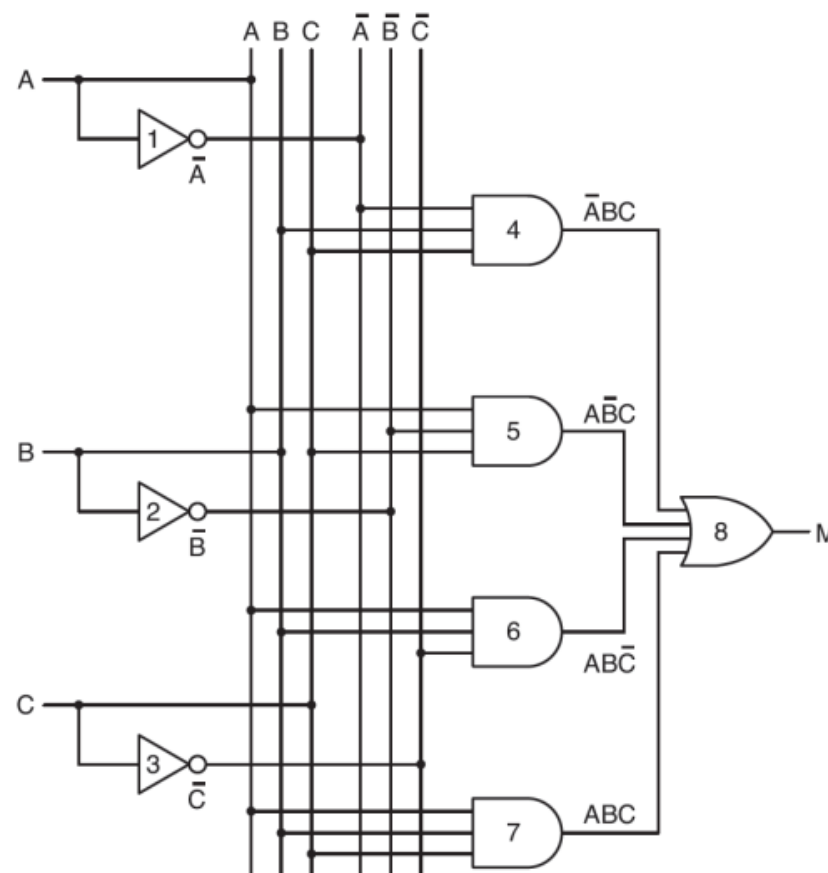
# PORTAS E ÁLGEBRA BOOLEANA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Tabela verdade para a função majoritária de três variáveis e respectivo circuito

A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



# DESENHO DE CIRCUITO A PARTIR DE FUNÇÃO



UNIVERSIDADE  
FEDERAL DO CEARÁ

a)  $F = A$

b)  $F = A+B$

c)  $F = AB$

d)  $F = \overline{A}$

e)  $F = \overline{A}+B$

f)  $F = A+\overline{B}$

g)  $F = \overline{A}+\overline{B}$

h)  $F = \overline{A+B}$

i)  $F = \overline{AB}$

j)  $F = A + B \cdot C$

k)  $F = A + \overline{B} \cdot C$

l)  $F = \overline{AB + \overline{B}C}$

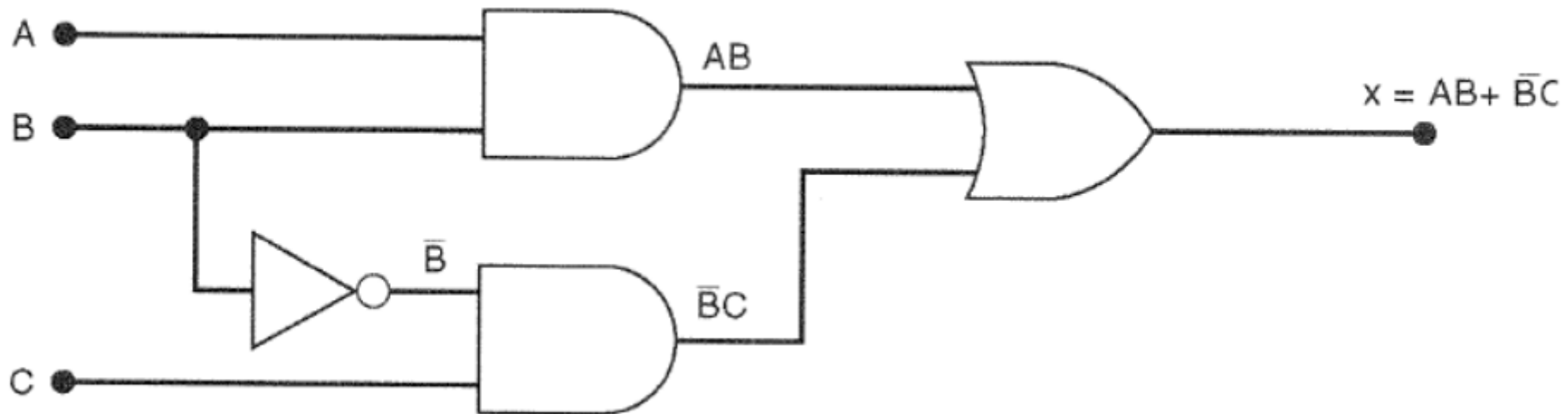
m)  $F = \overline{AB(C+D)}$

# DESENHO DE CIRCUITO A PARTIR DE FUNÇÃO



UNIVERSIDADE  
FEDERAL DO CEARÁ

■  $F = AB + \bar{B}C$

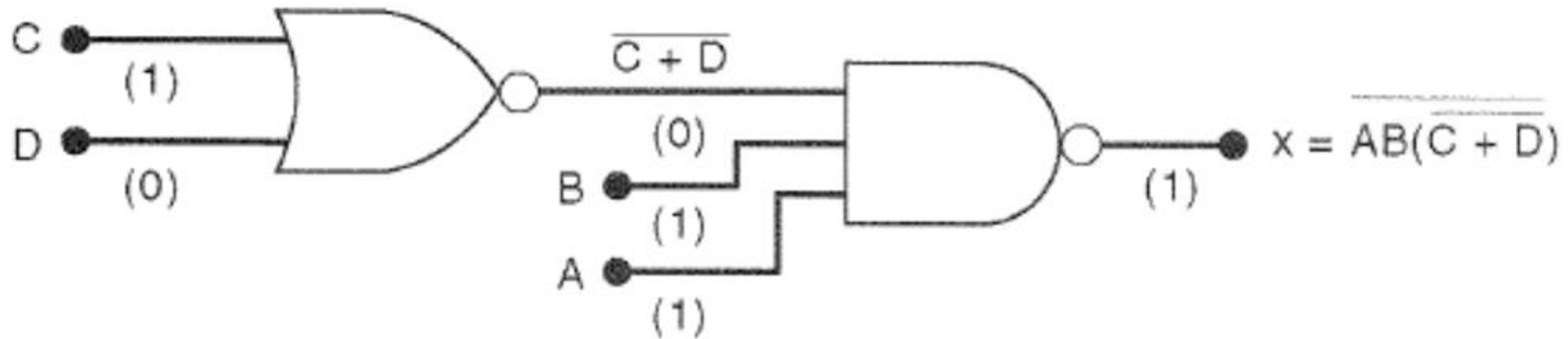


# DESENHO DE CIRCUITO A PARTIR DE FUNÇÃO



UNIVERSIDADE  
FEDERAL DO CEARÁ

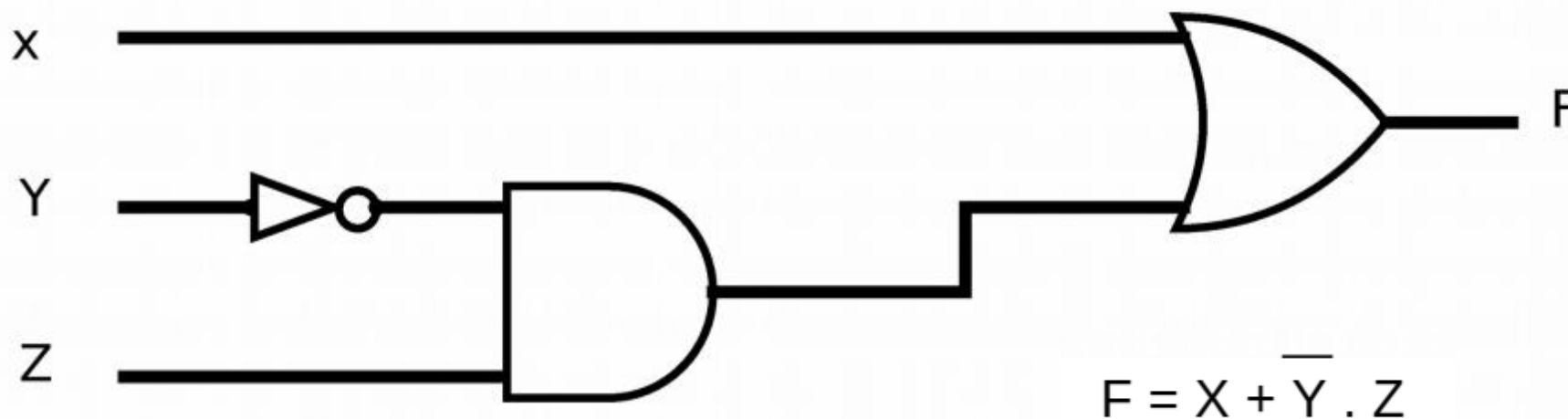
- $F = \overline{AB(C + D)}$



# DESENHO DE FUNÇÃO A PARTIR DE CIRCUITO



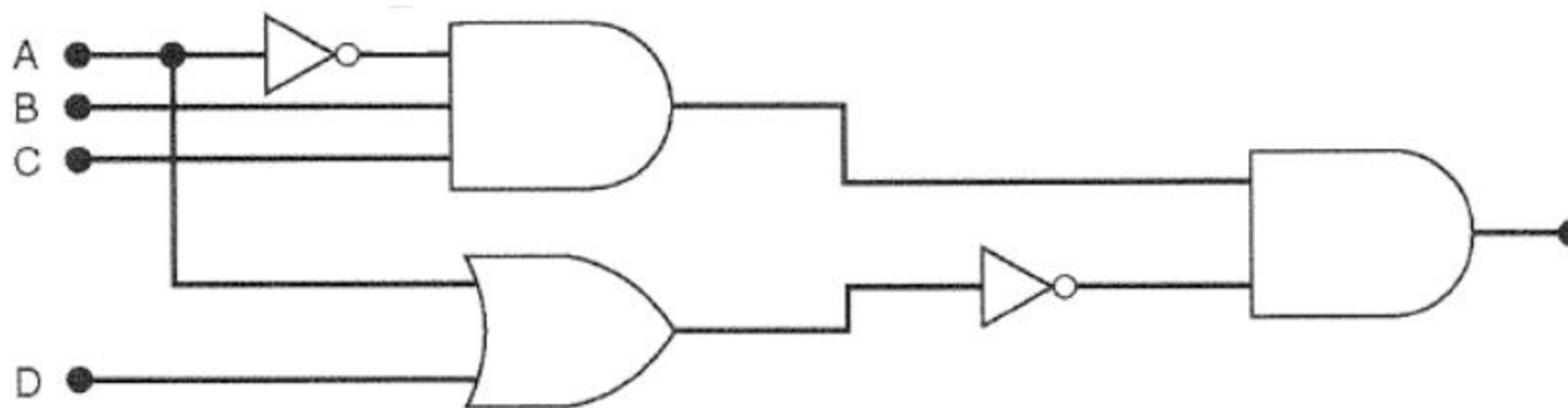
UNIVERSIDADE  
FEDERAL DO CEARÁ



# DESENHO DE FUNÇÃO A PARTIR DE CIRCUITO



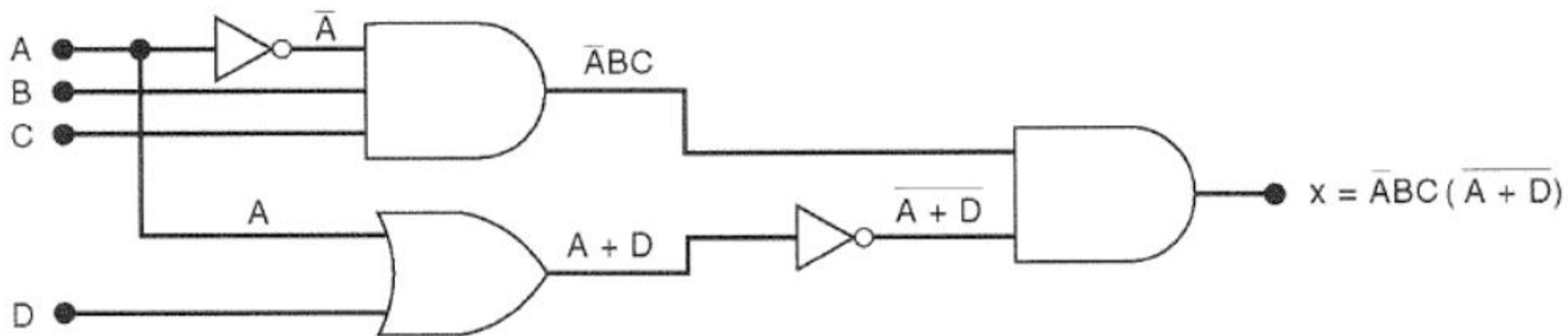
UNIVERSIDADE  
FEDERAL DO CEARÁ



# DESENHO DE FUNÇÃO A PARTIR DE CIRCUITO



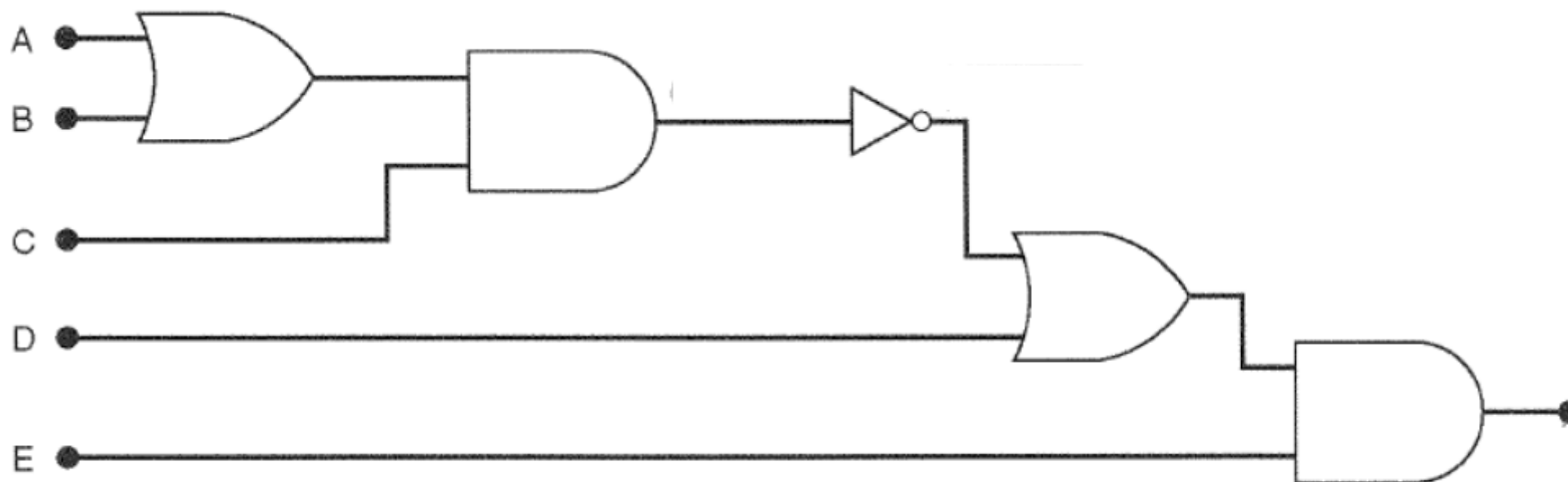
UNIVERSIDADE  
FEDERAL DO CEARÁ



# DESENHO DE FUNÇÃO A PARTIR DE CIRCUITO



UNIVERSIDADE  
FEDERAL DO CEARÁ

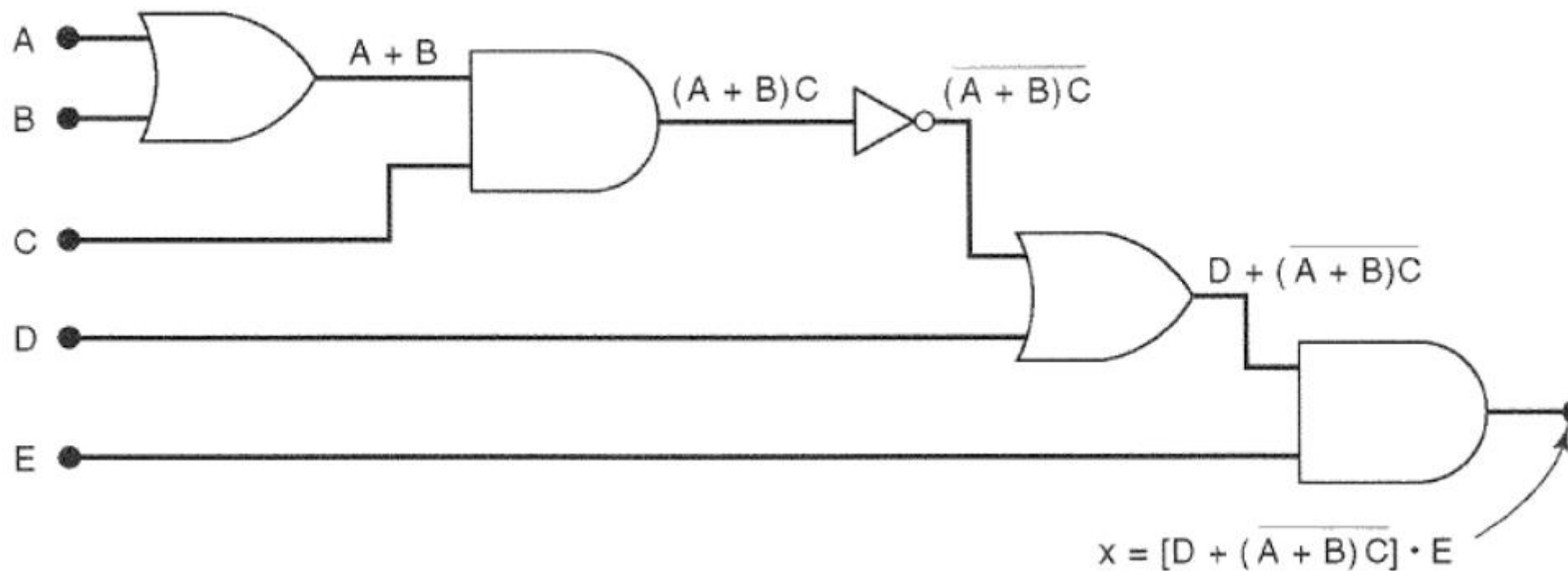




# DESENHO DE FUNÇÃO A PARTIR DE CIRCUITO



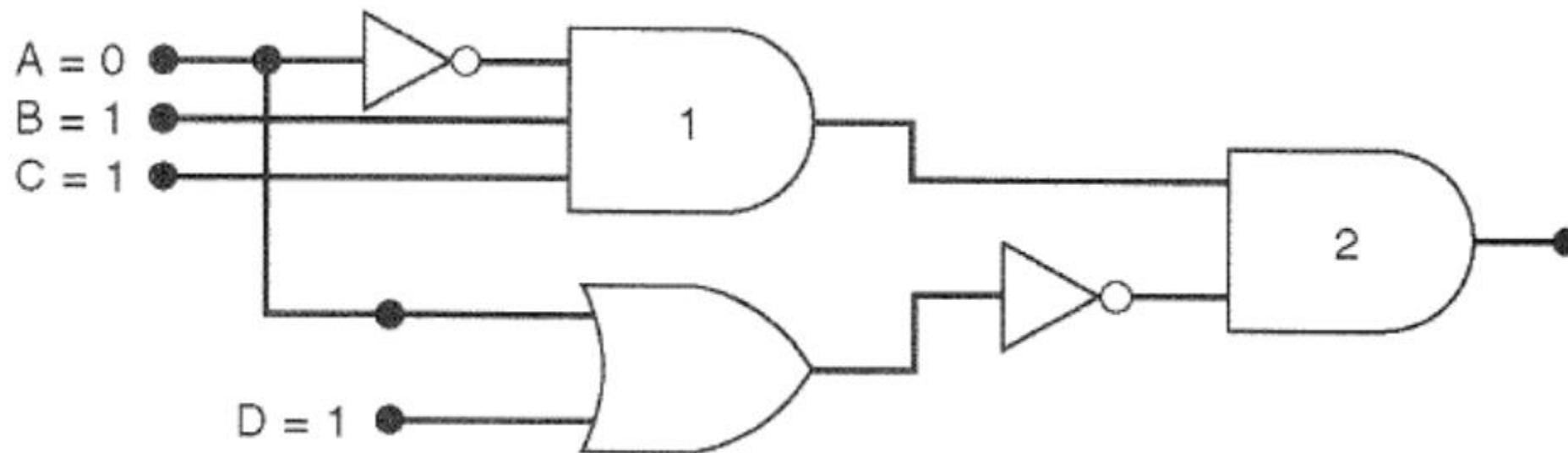
UNIVERSIDADE  
FEDERAL DO CEARÁ



# DETERMINAR O NÍVEL LÓGICO DE SAÍDA A PARTIR DO DIAGRAMA



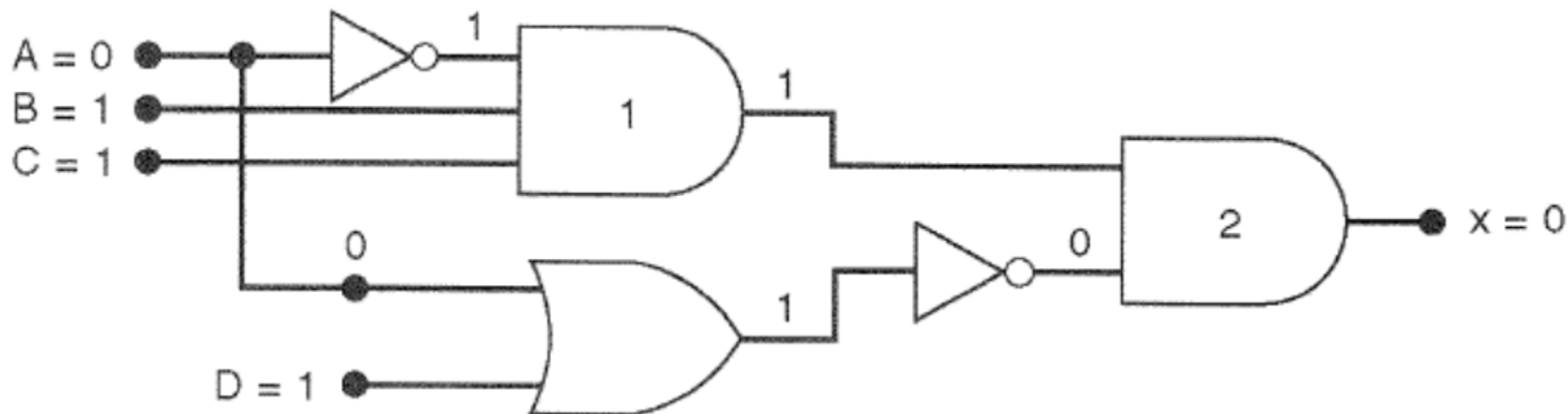
UNIVERSIDADE  
FEDERAL DO CEARÁ



# DETERMINAR O NÍVEL LÓGICO DE SAÍDA A PARTIR DO DIAGRAMA



UNIVERSIDADE  
FEDERAL DO CEARÁ



# EXPRESSÃO BOOLEANA A PARTIR DA TABELA VERDADE



UNIVERSIDADE  
FEDERAL DO CEARÁ

X	Y	Z	F
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

$$\Rightarrow \overline{X} . \overline{Y} . Z$$

$$\Rightarrow X . \overline{Y} . \overline{Z}$$

$$\Rightarrow X . \overline{Y} . Z$$

$$\Rightarrow X . Y . \overline{Z}$$

$$\Rightarrow X . Y . Z$$

+

$$F = (\overline{X} . \overline{Y} . Z) + (X . \overline{Y} . \overline{Z}) + \\ (X . \overline{Y} . Z) + (X . Y . \overline{Z}) + \\ (X . Y . Z)$$

# EXPRESSÃO BOOLEANA A PARTIR DA TABELA VERDADE



UNIVERSIDADE  
FEDERAL DO CEARÁ

A	B	X
0	0	1
0	1	0
1	0	1
1	1	0

# EXPRESSÃO BOOLEANA A PARTIR DA TABELA VERDADE



UNIVERSIDADE  
FEDERAL DO CEARÁ

A	B	C	D	x
0	0	0	0	0
0	0	0	1	0
0	0	1	0	0
0	0	1	1	1
0	1	0	0	1
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	0
1	1	0	1	0
1	1	1	0	0
1	1	1	1	1

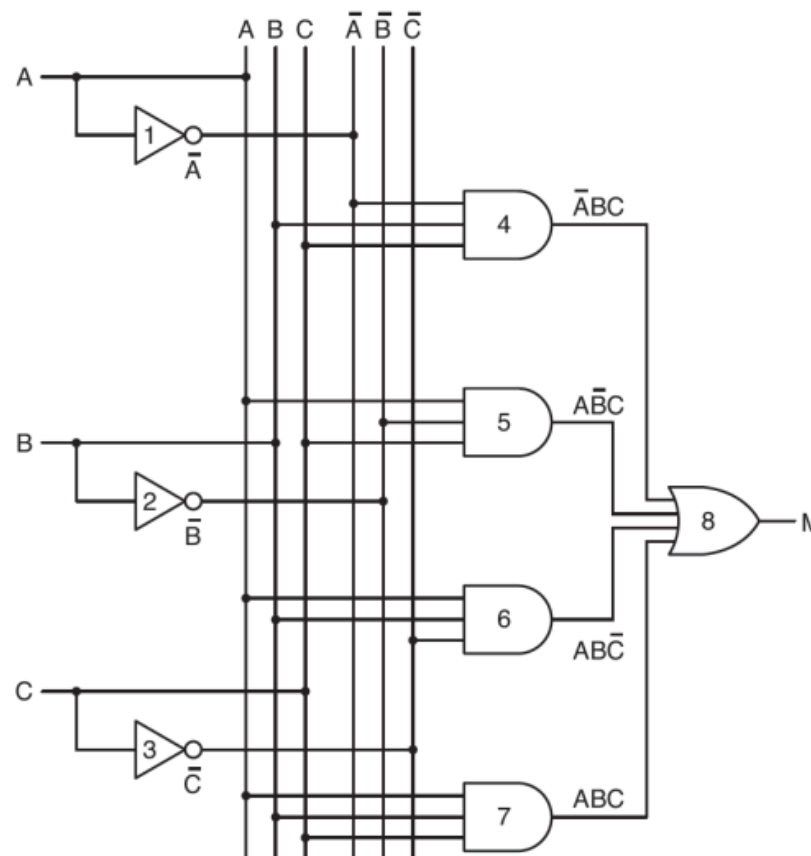
# PORTAS E ÁLGEBRA BOOLEANA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Tabela verdade para a função majoritária de três variáveis e respectivo circuito

A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



# PORTAS E ÁLGEBRA BOOLEANA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Para **reduzir a complexidade de um circuito**, o projetista tem de **encontrar outro circuito** que calcule a mesma função que o original, mas efetue essa operação com um número menor de portas
- A **álgebra booleana** pode ser uma ferramenta valiosa na busca de **circuitos equivalentes**
- Um projetista de circuitos começa com uma função booleana e depois aplica a ela as leis da álgebra booleana na tentativa de achar uma função mais simples
- **Um circuito pode ser construído com base na função final**



# PORTAS E ÁLGEBRA BOOLEANA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Leis Fundamentais e Propriedades da Álgebra Booleana
  - As leis da álgebra Booleana dizem respeito ao espaço Booleano (isto é, valores que uma variável pode assumir) e às operações elementares desse espaço
  - As propriedades podem ser deduzidas a partir das definições das operações
  - Sejam  $A$ ,  $B$  e  $C$  três variáveis Booleanas. Então, o espaço Booleano é definido:
    - se  $A \neq 0$ , então  $A = 1$
    - se  $A \neq 1$ , então  $A = 0$
  - O mesmo se aplica para  $B$ ,  $C$  ou outra variável do mesmo espaço

# PORTAS E ÁLGEBRA BOOLEANA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Operações básicas desse espaço:
  - AND (e lógico)
  - OR (ou lógico)
  - NOT (negação / inversão lógica)

# PORTAS E ÁLGEBRA BOOLEANA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Algumas identidades da álgebra booleana

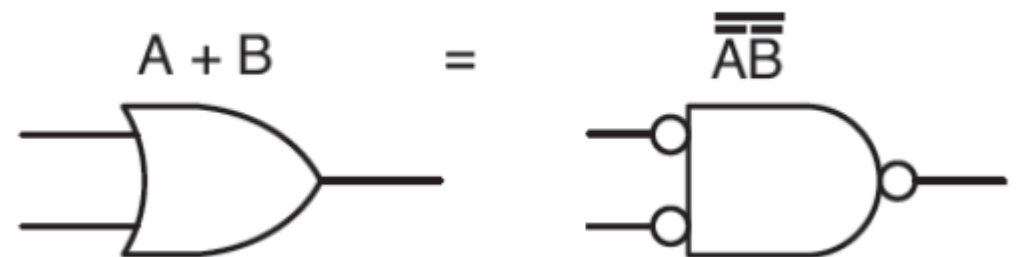
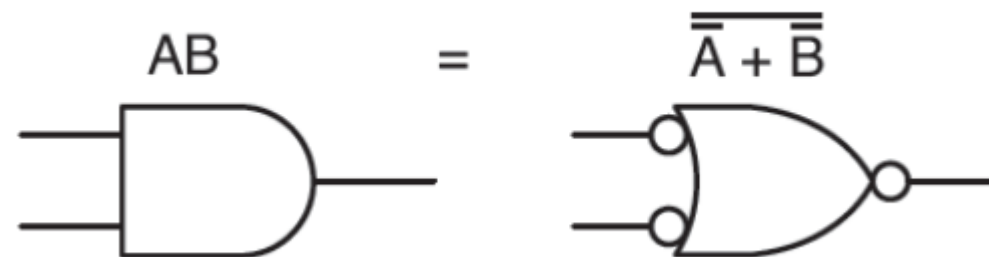
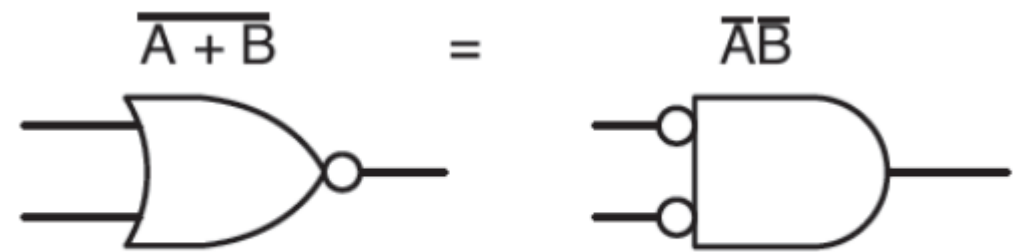
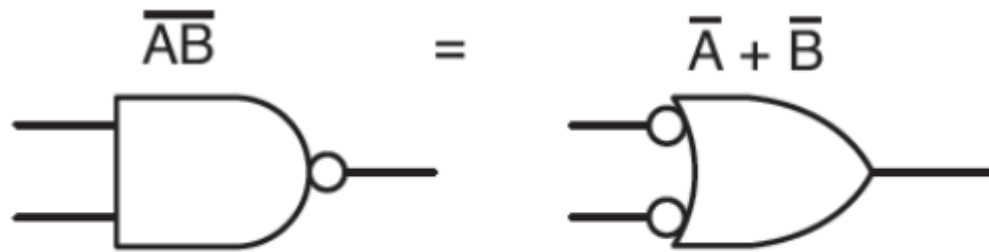
Nome	Forma AND	Forma OR
Lei da identidade	$1A = A$	$0 + A = A$
Lei do elemento nulo	$0A = 0$	$1 + A = 1$
Lei idempotente	$AA = A$	$A + A = A$
Lei do inverso	$\overline{\overline{A}} = A$	$A + \overline{A} = 1$
Lei comutativa	$AB = BA$	$A + B = B + A$
Lei associativa	$(AB)C = A(BC)$	$(A + B) + C = A + (B + C)$
Lei distributiva	$A + BC = (A + B)(A + C)$	$A(B + C) = AB + AC$
Lei da absorção	$A(A + B) = A$	$A + AB = A$
Lei de De Morgan	$\overline{AB} = \overline{A} + \overline{B}$	$\overline{A + B} = \overline{A}\overline{B}$
Identidade Auxiliar	$A + \overline{A}.B = A + B$	

# PORTAS E ÁLGEBRA BOOLEANA



UNIVERSIDADE  
FEDERAL DO CEARÁ

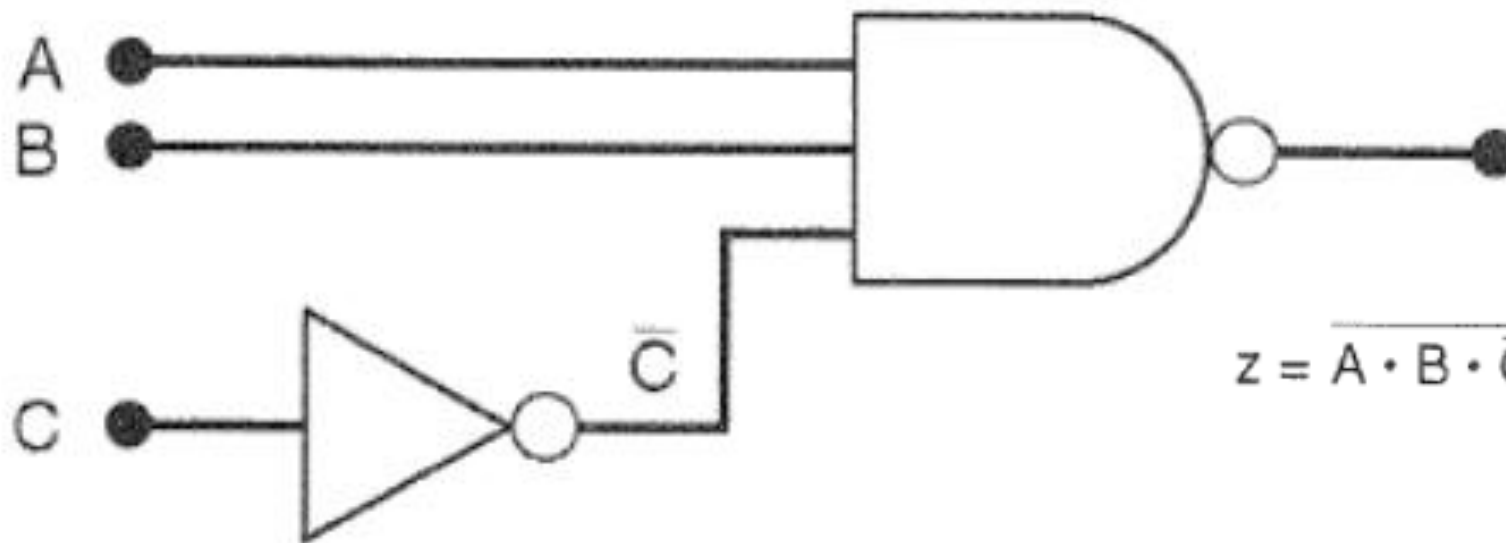
- Símbolos alternativos para algumas portas: NAND. NOR. AND. OR



# SIMPLIFICAÇÃO ALGÉBRICA



UNIVERSIDADE  
FEDERAL DO CEARÁ

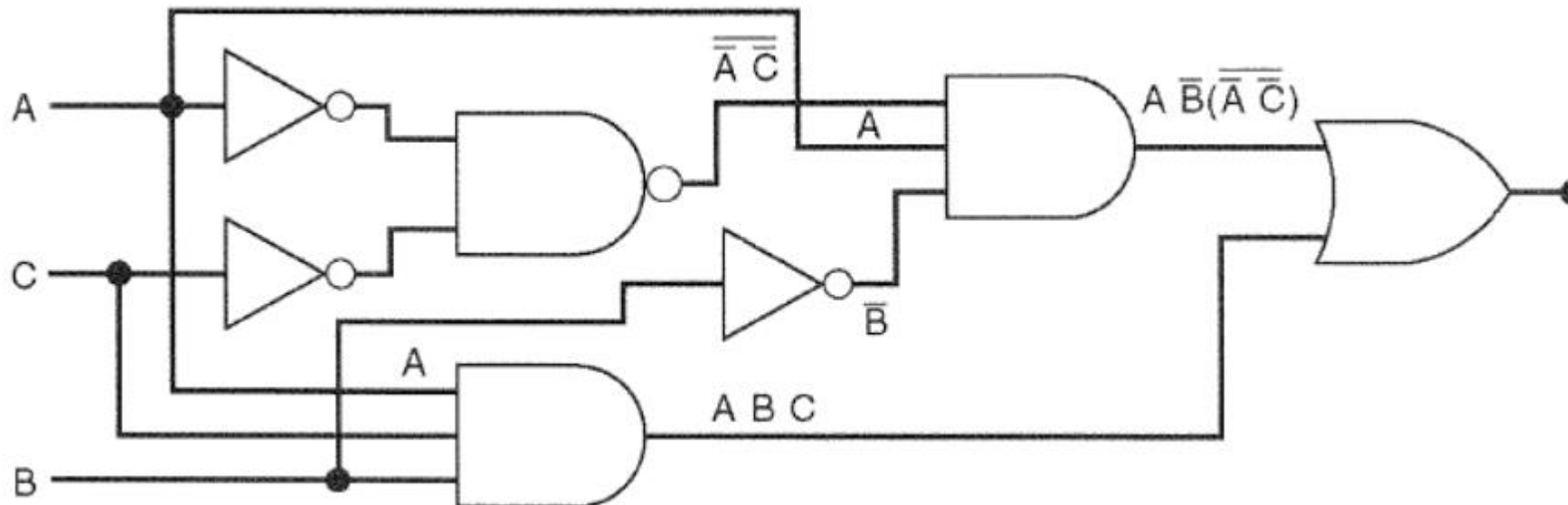


$$z = \overline{A \cdot B \cdot \bar{C}} = \bar{A} + \bar{B} + \bar{\bar{C}} = \bar{A} + \bar{B} + C$$

# SIMPLIFICAÇÃO ALGÉBRICA



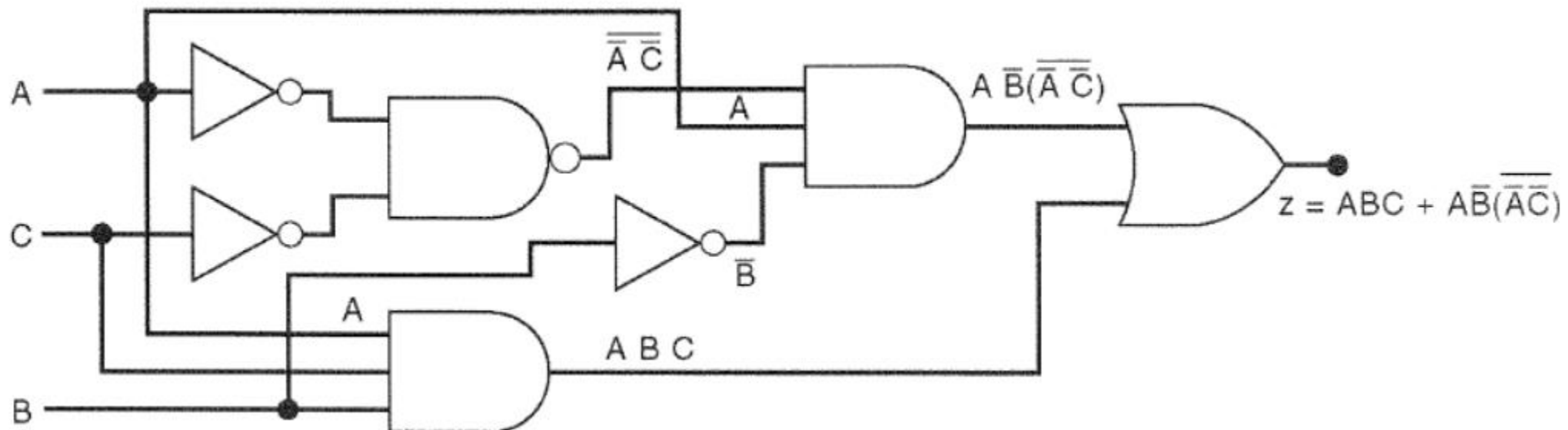
UNIVERSIDADE  
FEDERAL DO CEARÁ



# SIMPLIFICAÇÃO ALGÉBRICA



UNIVERSIDADE  
FEDERAL DO CEARÁ



# SIMPLIFICAÇÃO ALGÉBRICA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Simplifique:

1.  $S = ABC + AC' + AB'$

2.  $S = (ABC').(A' + B' + C')$

3.  $S = ((AC)' + B + D)' + C.(ACD)'$



# SIMPLIFICAÇÃO ALGÉBRICA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- $(A + \bar{B} + A\bar{B})(AB + \bar{A}C + BC)$

$$CB'A' + BA$$

- $(A + B + C)(\bar{A} + \bar{B} + C)$

$$B'A + BA' + C$$

- $\overline{AB} + \overline{A + C}$

$$B' + A'$$



# CIRCUITOS LÓGICOS

CIRCUITOS INTEGRADOS

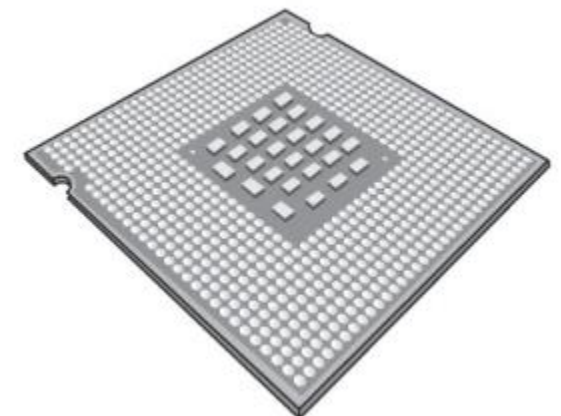
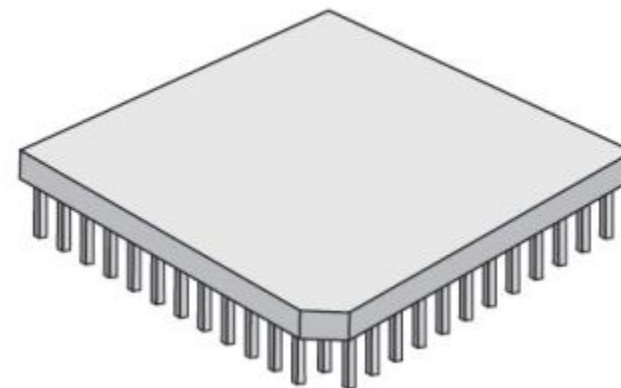
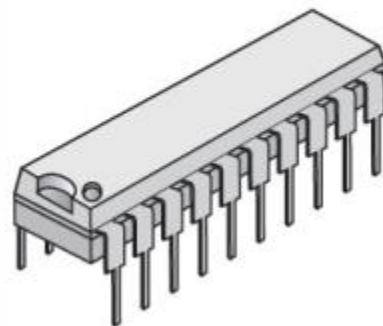


# CIRCUITOS INTEGRADOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Portas não são fabricadas nem vendidas individualmente, mas em unidades denominadas circuitos integrados, muitas vezes denominados ICs, CIs ou chips.
- Um CI é um pedaço quadrado de silício de tamanho variado, dependendo de quantas portas são necessárias para executar os componentes do chip.
  - Substratos de 2x2 mm a 18x18 mm



# CIRCUITOS INTEGRADOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Portas não são fabricadas nem vendidas individualmente, mas em unidades denominadas circuitos integrados, muitas vezes denominados ICs, CIs ou chips.
- Um CI é um pedaço quadrado de silício de tamanho variado, dependendo de quantas portas são necessárias para executar os componentes do chip.
  - Substratos de 2x2 mm a 18x18 mm
- ICs costumam ser montados em pacotes retangulares (ou quadrados) de plástico ou cerâmica.



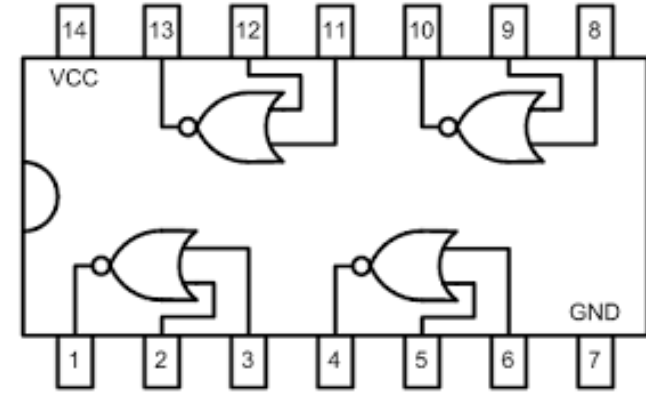
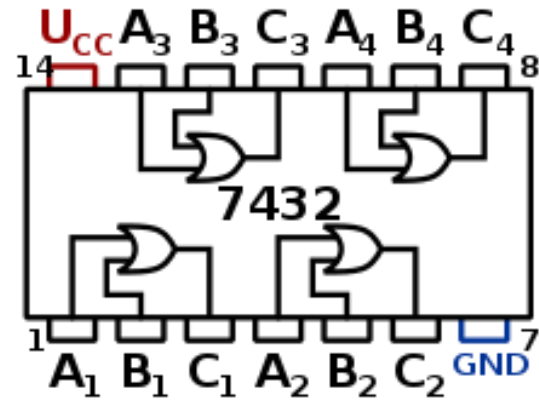
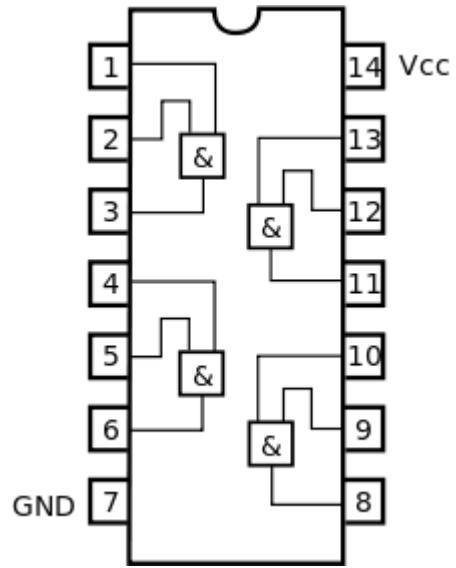
# CIRCUITOS INTEGRADOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Níveis de integração
  - SSI (Integração em pequena escala):
    - Constituída de 1 a 10 portas;
  - MSI (Integração em média escala):
    - Constituída de até 100 portas;
  - LSI (Integração em larga escala):
    - Constituída de até 100.000 portas;
  - VLSI (Integração em larguíssima escala):
    - Chips com mais de 100.000 portas.
- A tecnologia moderna permite colocar mais de 1 bilhão de transistores em um único chip

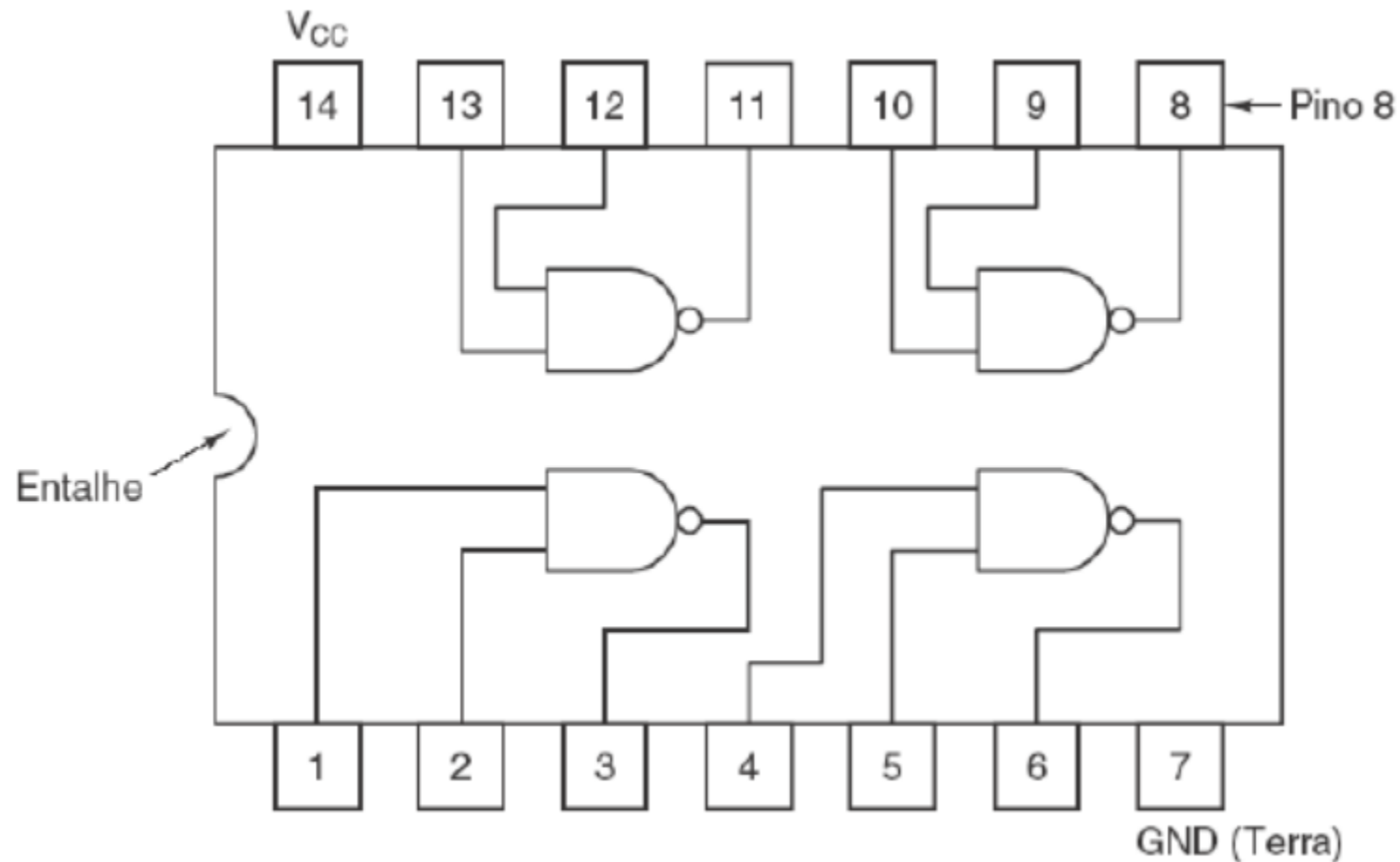
# CI 7408 7432 7402



# CIRCUITOS INTEGRADOS



UNIVERSIDADE  
FEDERAL DO CEARÁ



- Operação NAND
- Operação AND
- Operação NOT
- Operação OR
- Operação NOR

# CIRCUITOS INTEGRADOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Podem ser basicamente de dois tipos
- Circuitos combinacionais (combinatórios)
  - São circuitos em que a saída está diretamente relacionada aos valores de entrada, não possuindo elementos de memória em seu interior
- Circuitos sequenciais (de memória)
  - São aqueles em que o valor de saída não depende somente dos valores de entrada, mas também do “estado interno” do circuito – dependem também da memória do circuito





# CIRCUITOS LÓGICOS

CIRCUITOS COMBINACIONAIS



# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Muitas aplicações de lógica digital requerem um circuito com **múltiplas entradas e múltiplas saídas**, no qual as saídas são determinadas exclusivamente pelas entradas em questão – Esses circuitos são denominados **circuitos combinacionais (ou circuitos combinatórios)**
- Exemplos de circuitos combinatórios
  - Somador parcial
  - Somador completo
  - Decodificador
  - Multiplexador
  - Comparador

# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- No nível lógico, um **multiplexador** é um circuito com  **$2^n$  entradas de dados, uma saída de dados e  $n$  entradas de controle** que selecionam uma das entradas de dados.
- A entrada selecionada é dirigida para a saída
- Conversor de dados de paralelo para serial
- Exemplo: teclado – o toque em uma tecla define implicitamente um número (geralmente) de 7 bits que deve ser enviado serialmente para a CPU

# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Configurações de multiplexadores

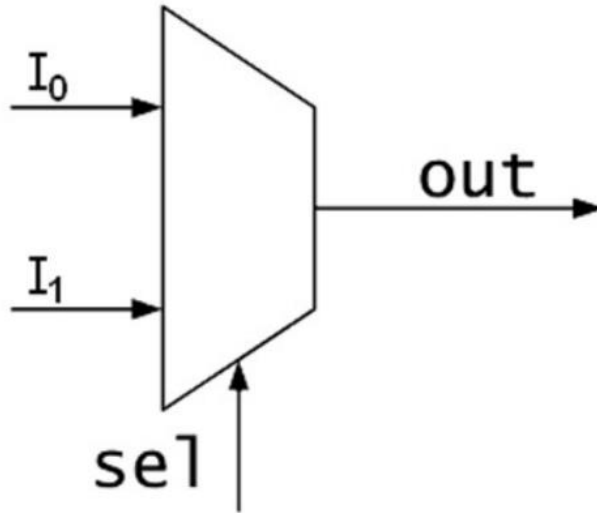
Multiplexador	Número de entradas	Número de linhas de seleção
2-para-1	2	1
4-para-1	4	2
8-para-1	8	3
16-para-1	16	4

# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Multiplexador de duas entradas



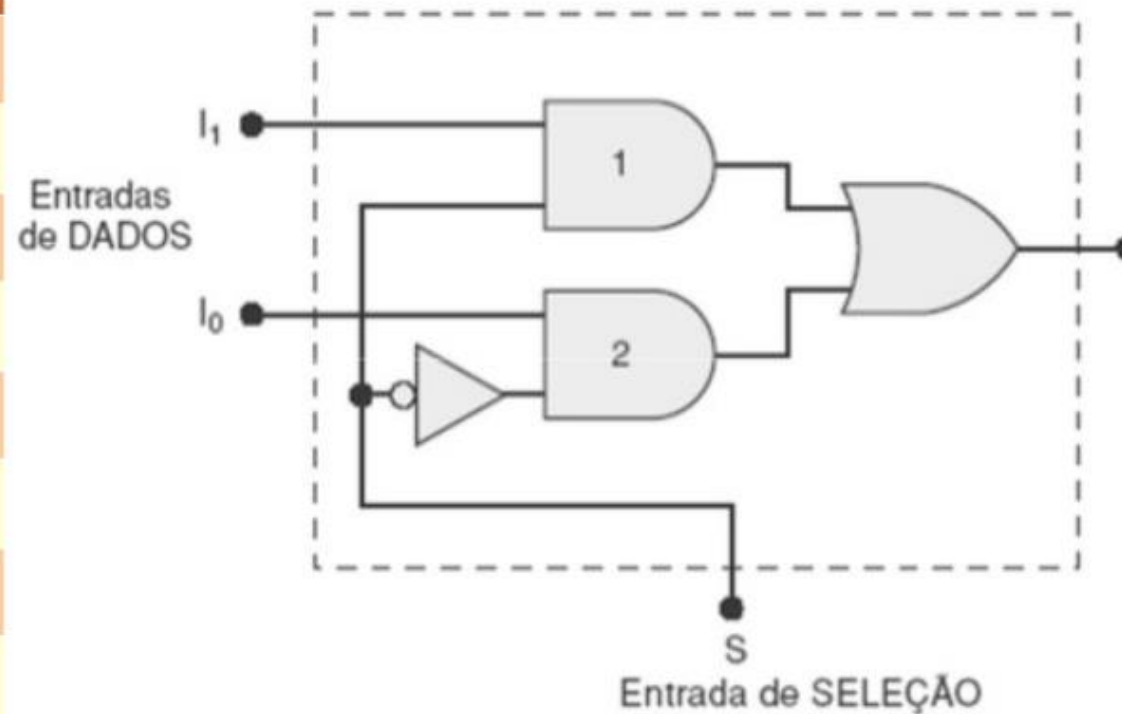
# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Multiplexador de duas entradas

$I_0$	$I_1$	$S$	$Z$
0	0		
0	0		
0	1		
0	1		
1	0		
1	0		
1	1		
1	1		



$$Z = I_0 \cdot \bar{S} + I_1 \cdot S$$

$S$	Saída
0	$Z = I_0$
1	$Z = I_1$

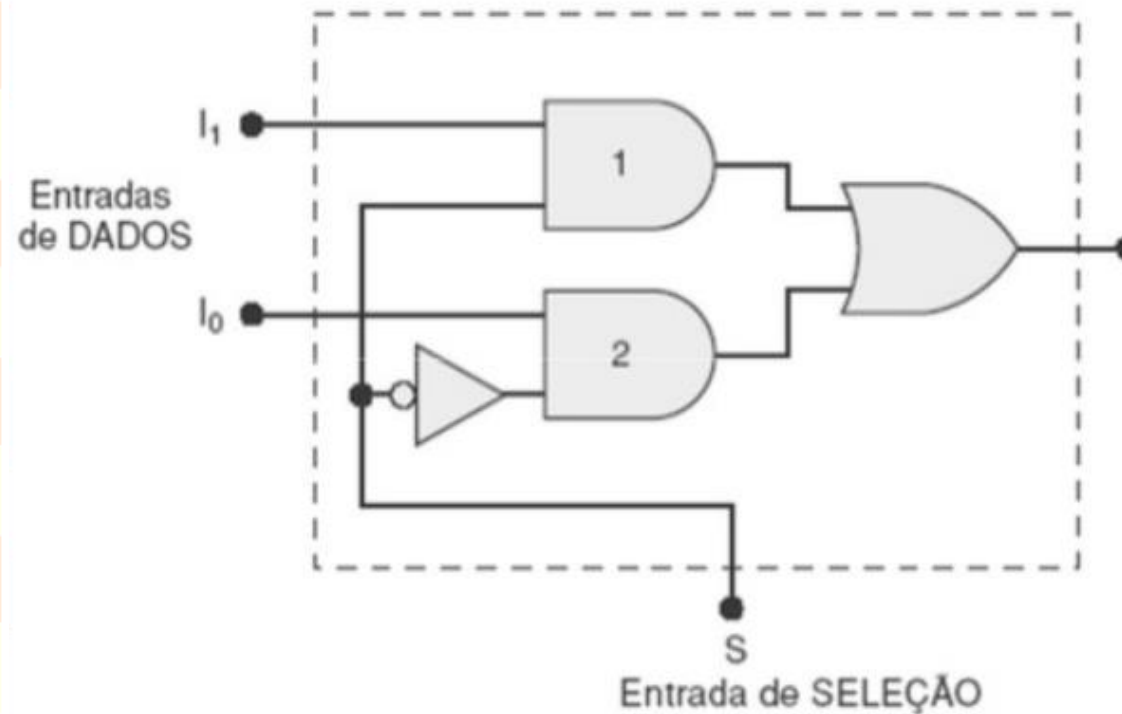
# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

## ■ Multiplexador de duas entradas

$I_0$	$I_1$	$S$	$Z$
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	



$$Z = I_0 \cdot \bar{S} + I_1 \cdot S$$

$S$	Saída
0	$Z = I_0$
1	$Z = I_1$

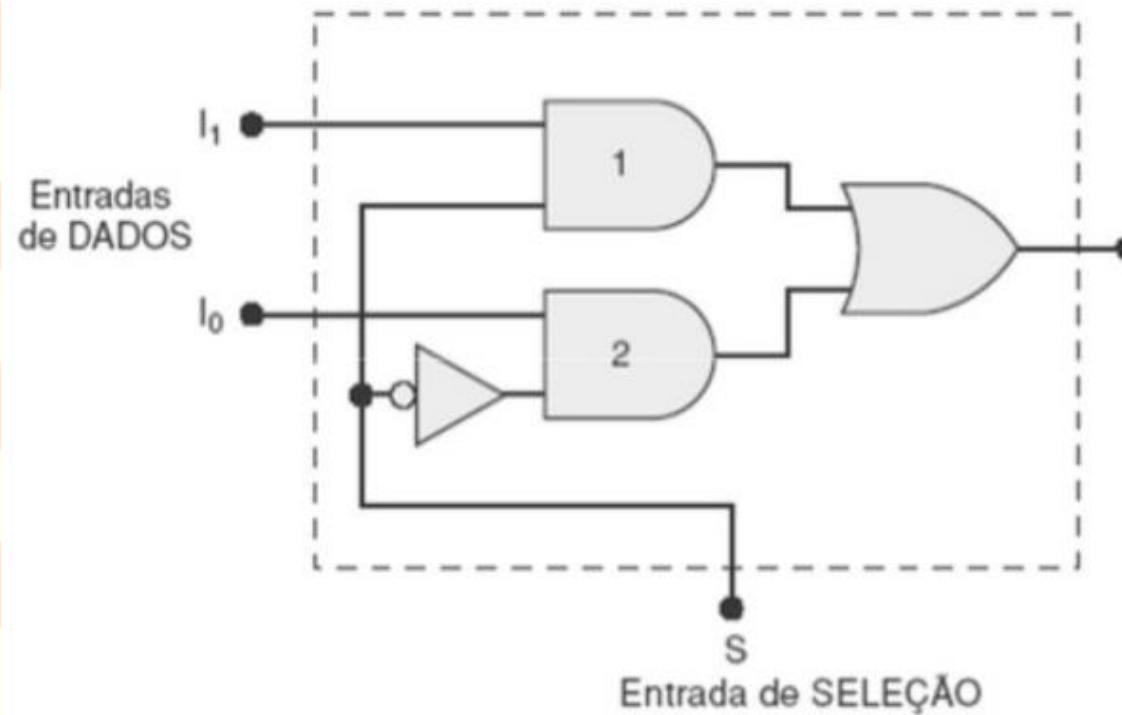
# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

## ■ Multiplexador de duas entradas

$I_0$	$I_1$	$S$	$Z$
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1



$$Z = I_0 \cdot \bar{S} + I_1 \cdot S$$

$S$	Saída
0	$Z = I_0$
1	$Z = I_1$

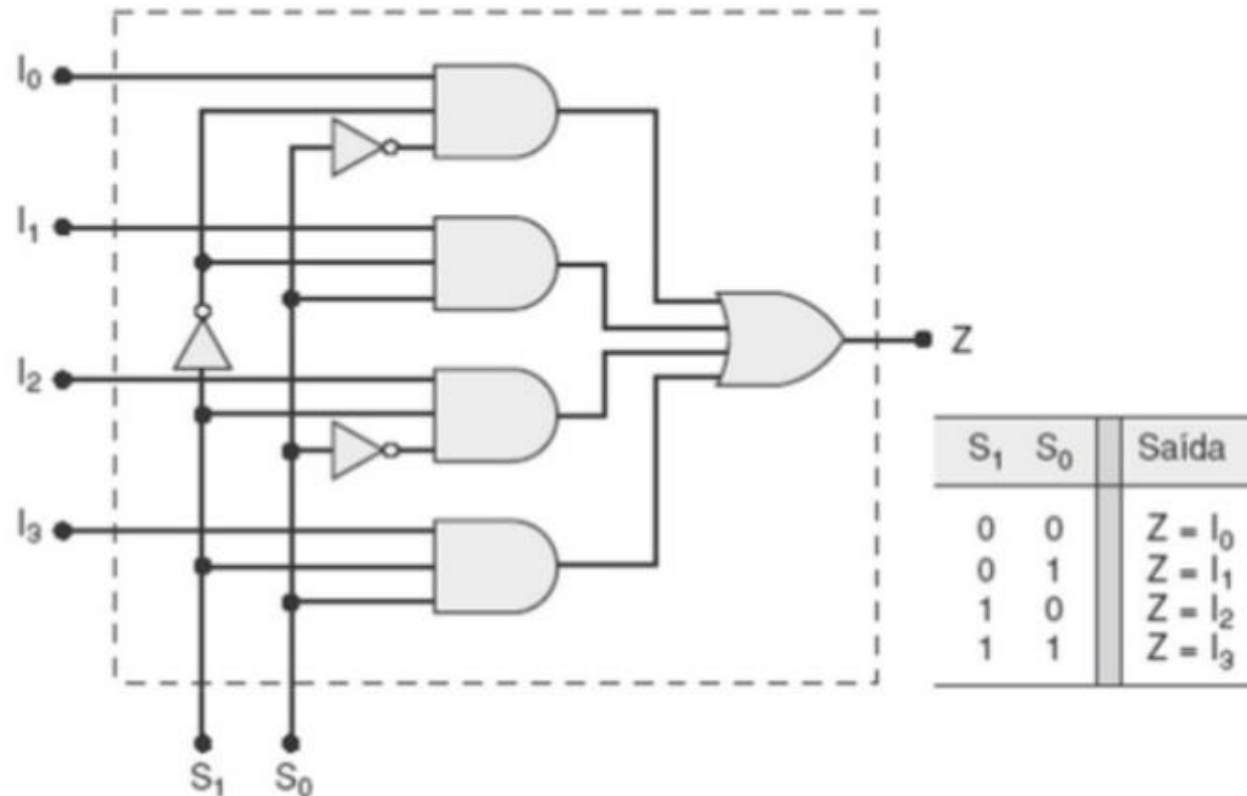


# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Multiplexador de quatro entradas

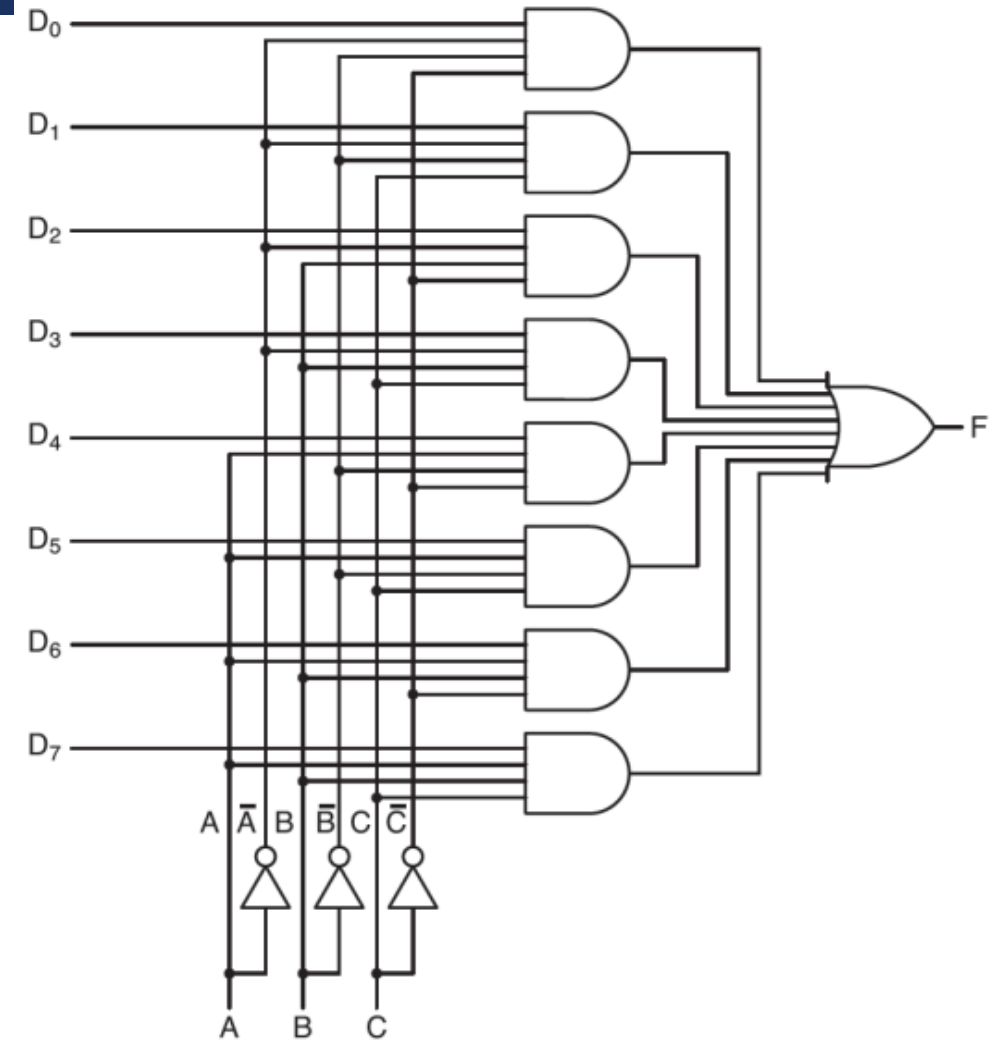


# CIRCUITOS COMBINACIONAIS

- Multiplexador de oito entradas



UNIVERSIDADE  
FEDERAL DO CEARÁ



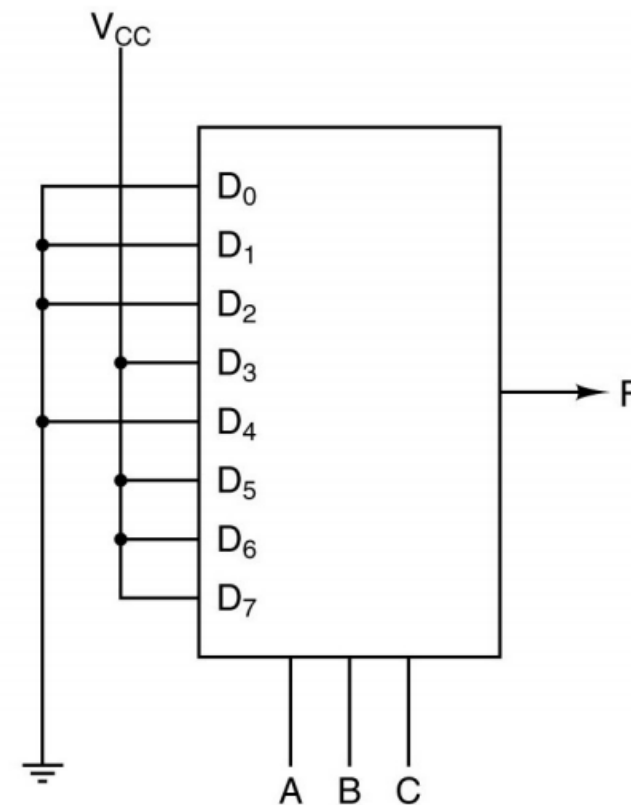
# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Multiplexador para a função majoritária de três variáveis

A	B	C	M
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

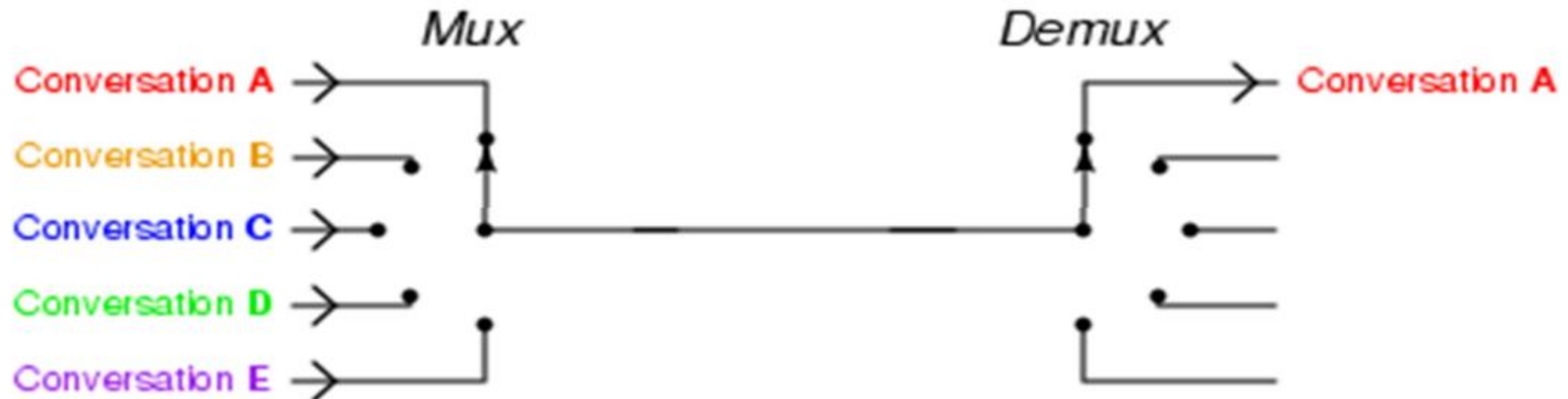


# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- O Demultiplexador realiza a operação inversa – dirige uma entrada para uma das  $2^n$  saídas

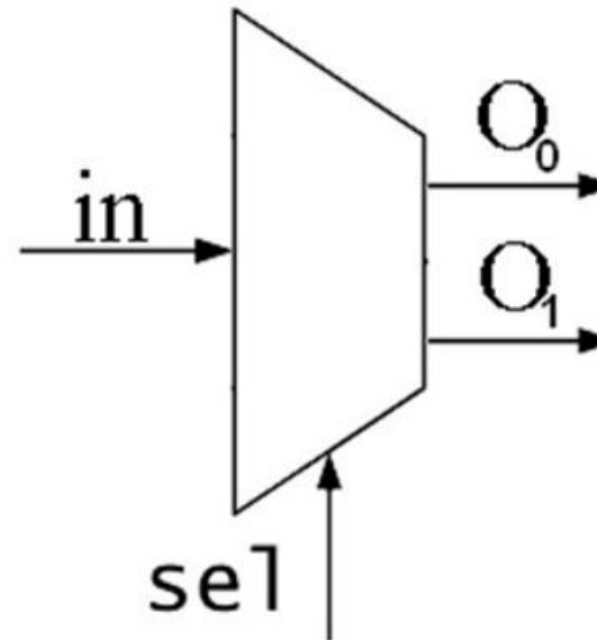
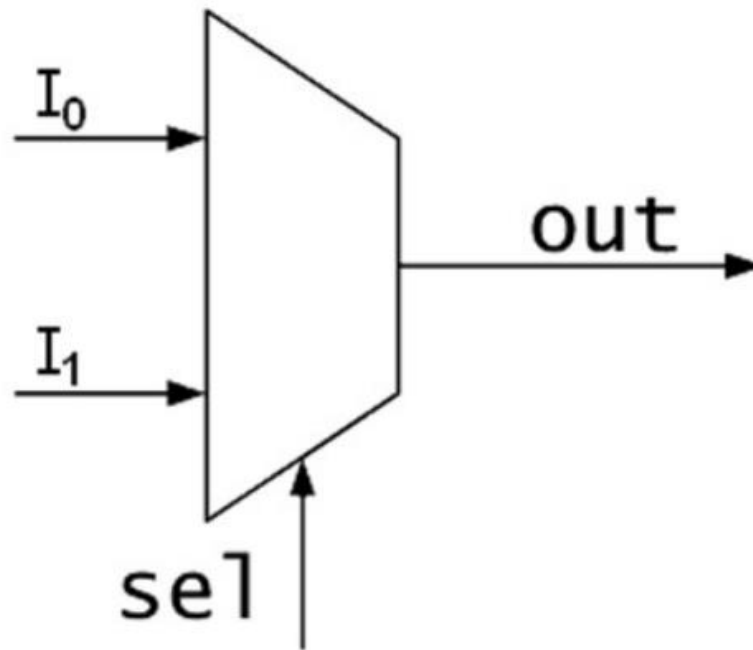


# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- O Demultiplexador realiza a operação inversa – dirige uma entrada para uma das  $2^n$  saídas



# CIRCUITOS COMBINACIONAIS

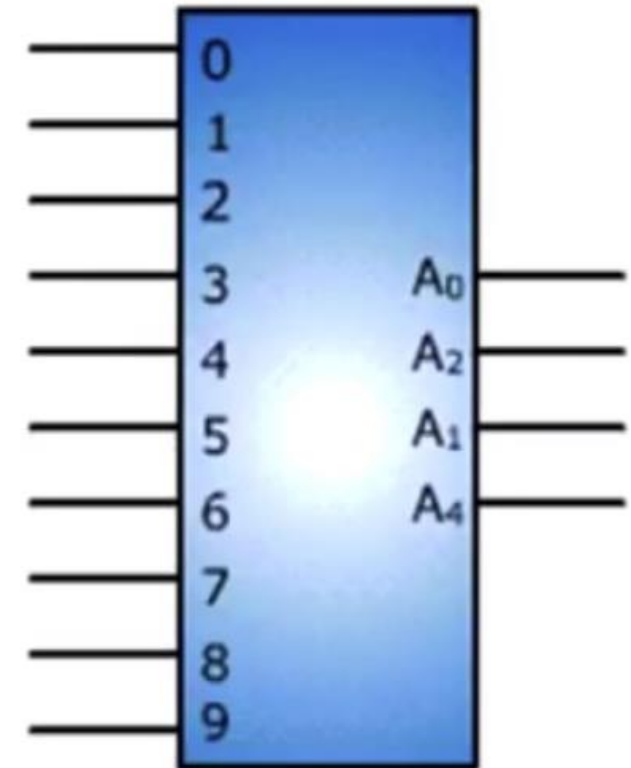


UNIVERSIDADE  
FEDERAL DO CEARÁ

## ■ Codificador

- Possuem várias linhas de entrada
- Saída: tantas linhas quantas forem necessárias

## ■ Ex.: Codificar decimal em binário



# CIRCUITOS COMBINACIONAIS

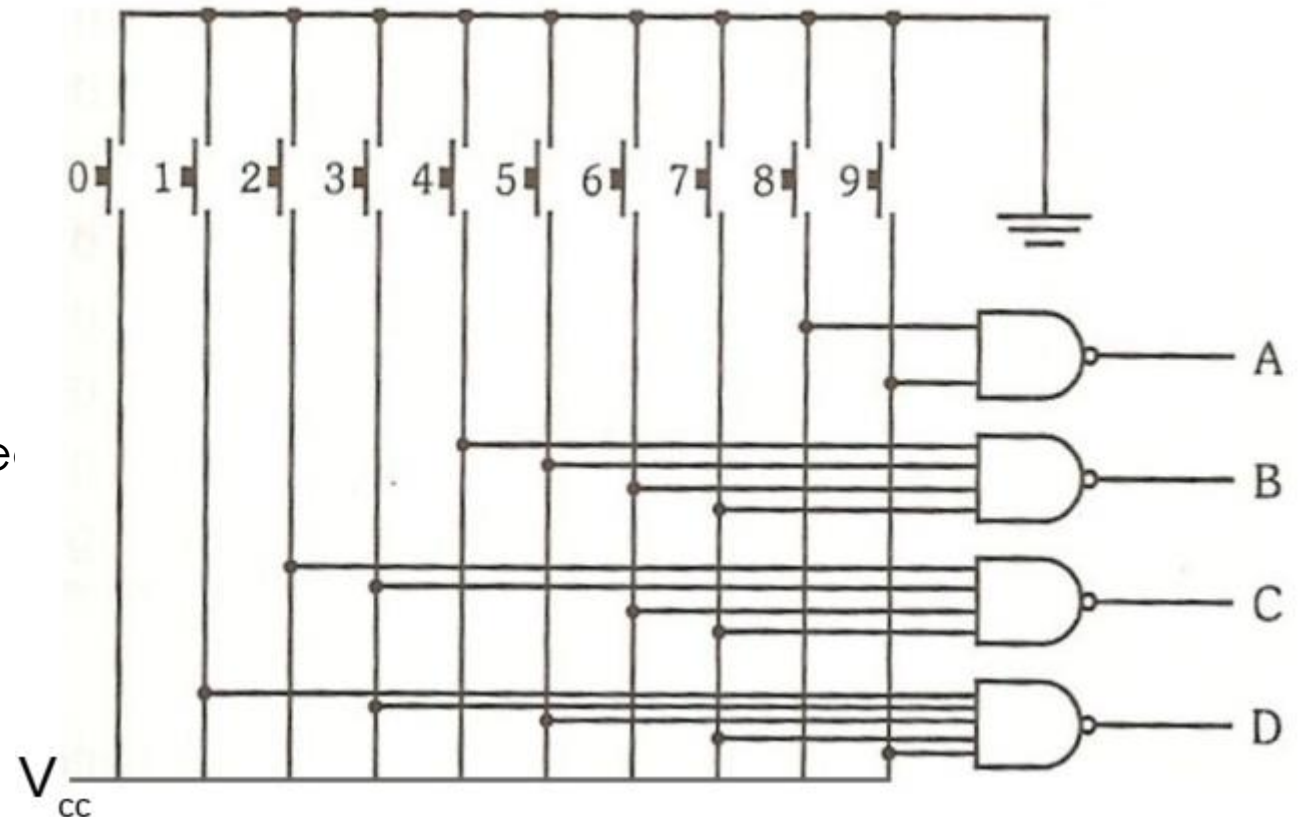


UNIVERSIDADE  
FEDERAL DO CEARÁ

## ■ Codificador

- Possuem várias linhas de entrada
- Saída: tantas linhas quantas forem ne

■ Ex.: Codificar decimal em binário

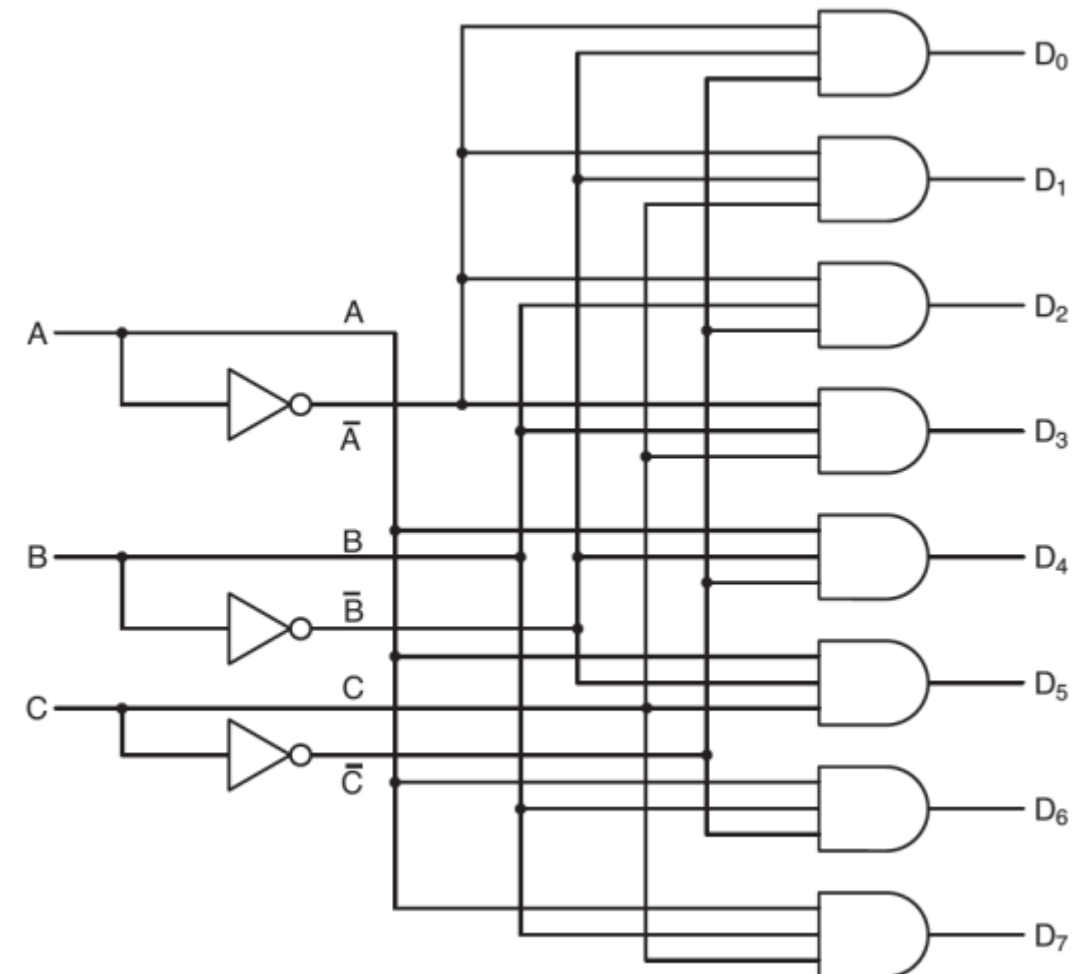


# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- **Decodificador** – circuito que toma um número de  $n$  bits como entrada e usa para selecionar (isto é, definir um 1) exatamente uma das  $2^n$  linhas de saída
- Cada porta AND tem três entradas, das quais a primeira é  $A$  ou  $A'$ , a segunda é  $B$  ou  $B'$  e a terceira é  $C$  ou  $C'$
- Cada porta é habilitada por uma combinação (única) diferente de entradas
- $D_0$ :  $A' B' C'$
- $D_1$ :  $A' B' C$
- etc

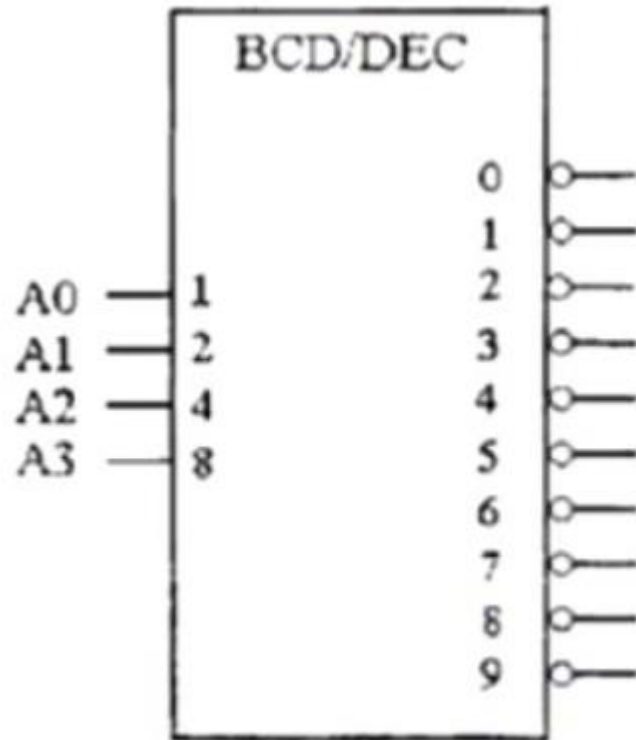




# CIRCUITOS COMBINACIONAIS



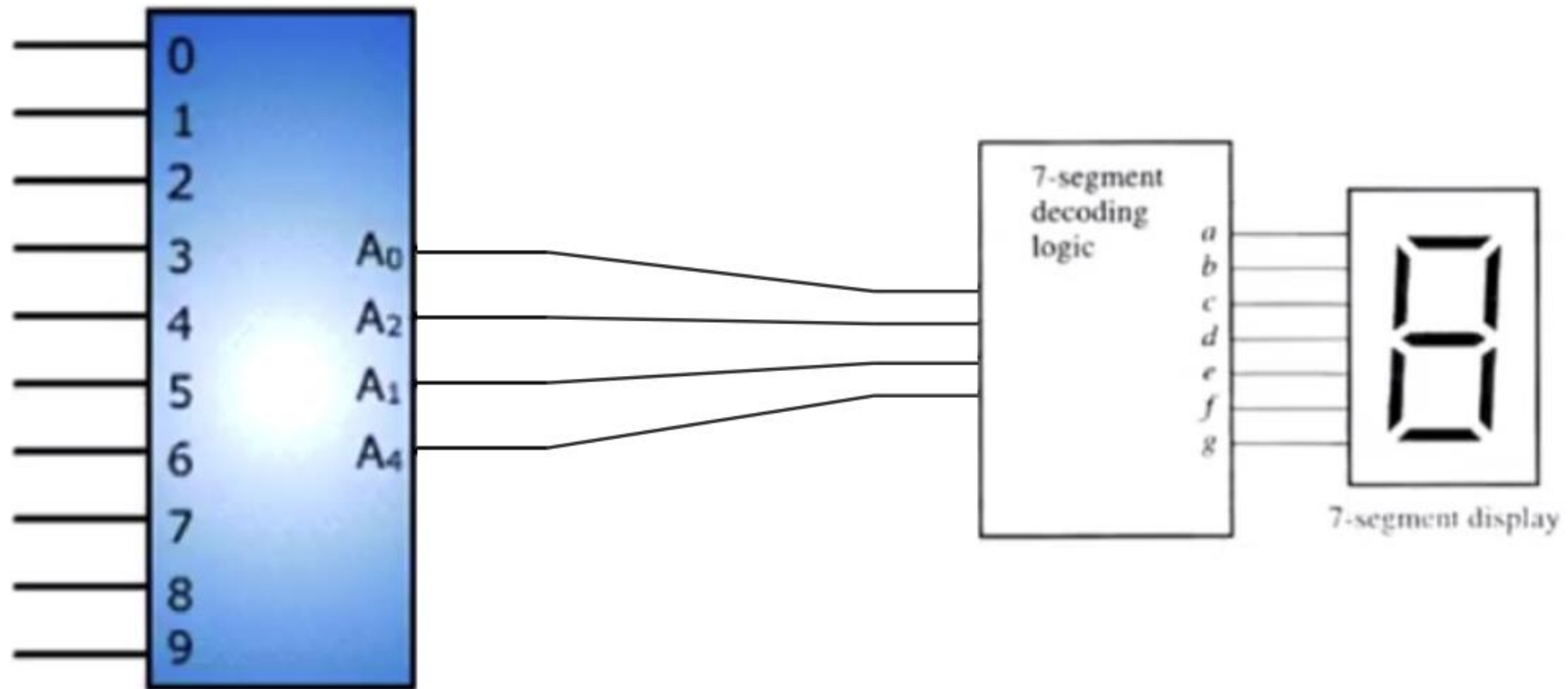
UNIVERSIDADE  
FEDERAL DO CEARÁ



# CIRCUITOS COMBINACIONAIS



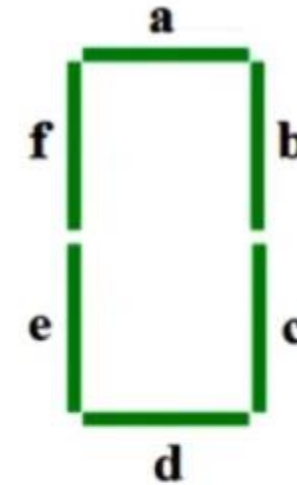
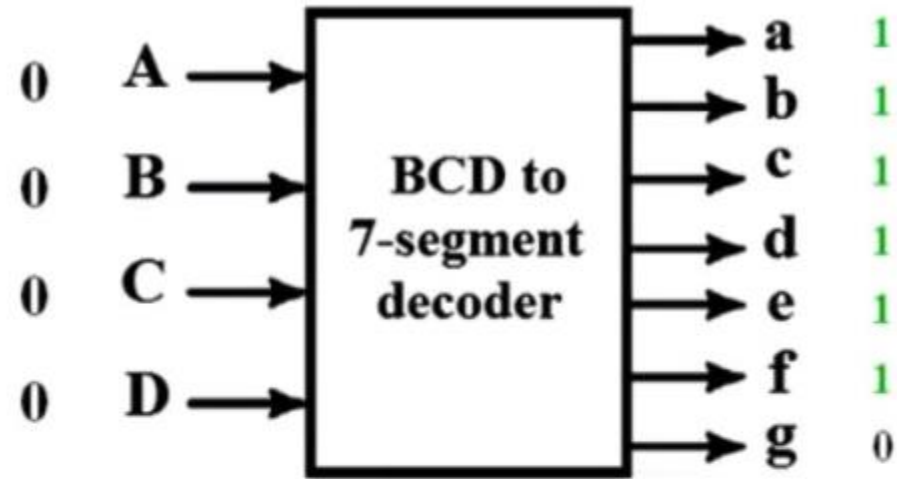
UNIVERSIDADE  
FEDERAL DO CEARÁ



Codificar decimal em binário

Decodificador 7-segmentos

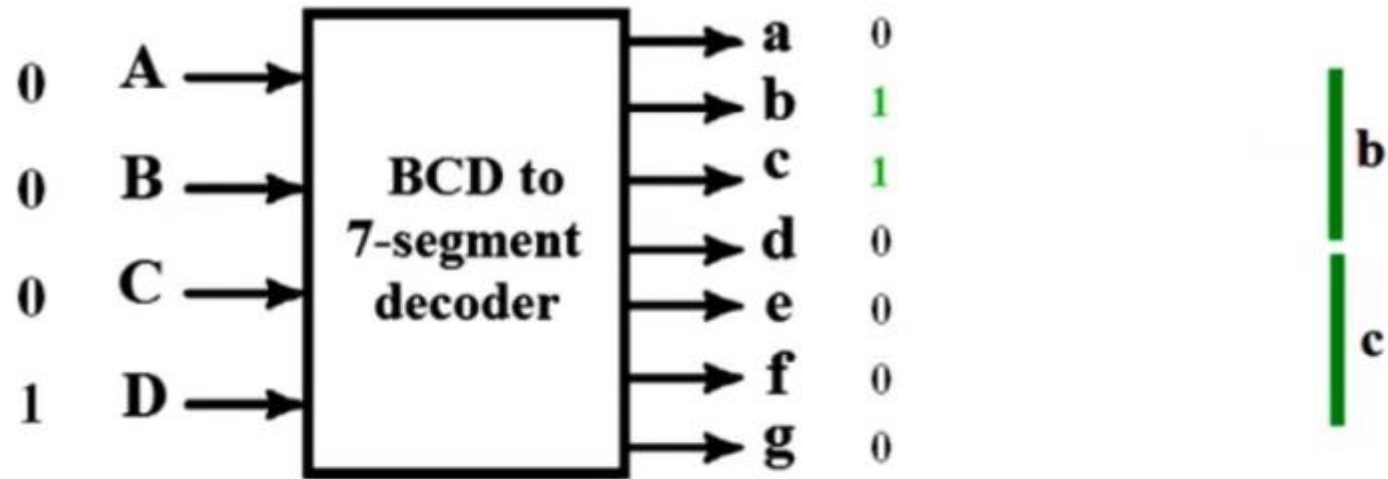
# BCD to 7-segment decoder



A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

# BCD to 7-segment decoder

CIRCUITOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

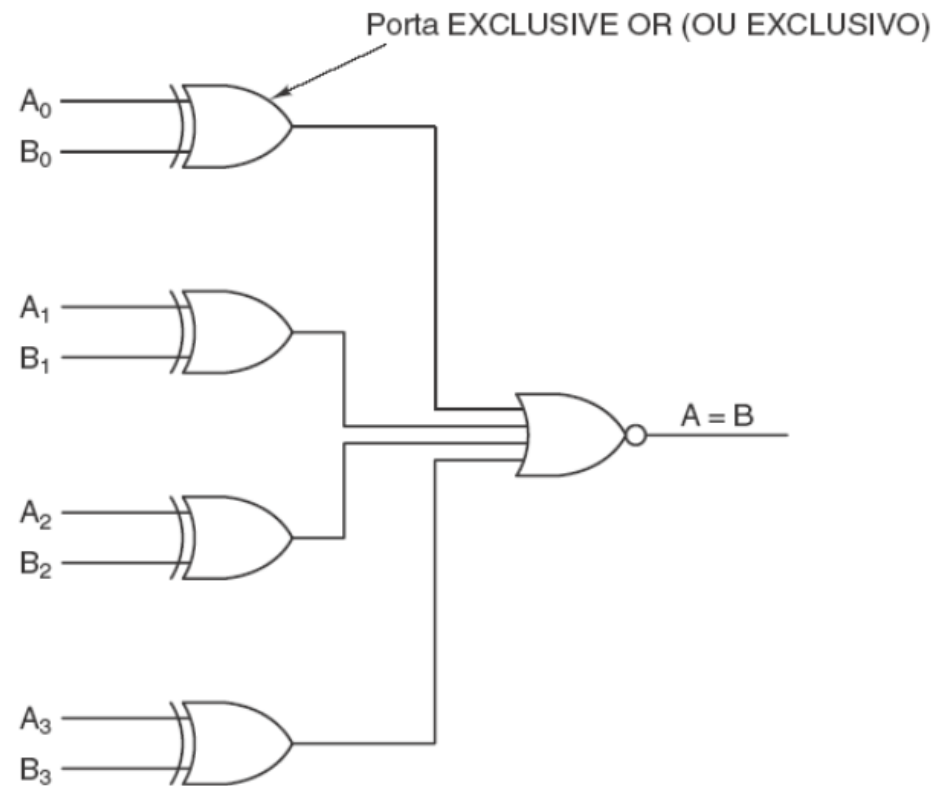
A	B	C	D	a	b	c	d	e	f	g
0	0	0	0	1	1	1	1	1	1	0
0	0	0	1	0	1	1	0	0	0	0
0	0	1	0	1	1	0	1	1	0	1
0	0	1	1	1	1	1	1	0	0	1
0	1	0	0	0	1	1	0	0	1	1
0	1	0	1	1	0	1	1	0	1	1
0	1	1	0	1	0	1	1	1	1	1
0	1	1	1	1	1	1	0	0	0	0
1	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	1	1	1	0	1	1

# CIRCUITOS COMBINACIONAIS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Outro circuito útil é o **comparador**, que compara duas palavras de entrada





# CIRCUITOS LÓGICOS

CIRCUITOS ARITMÉTICOS



# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

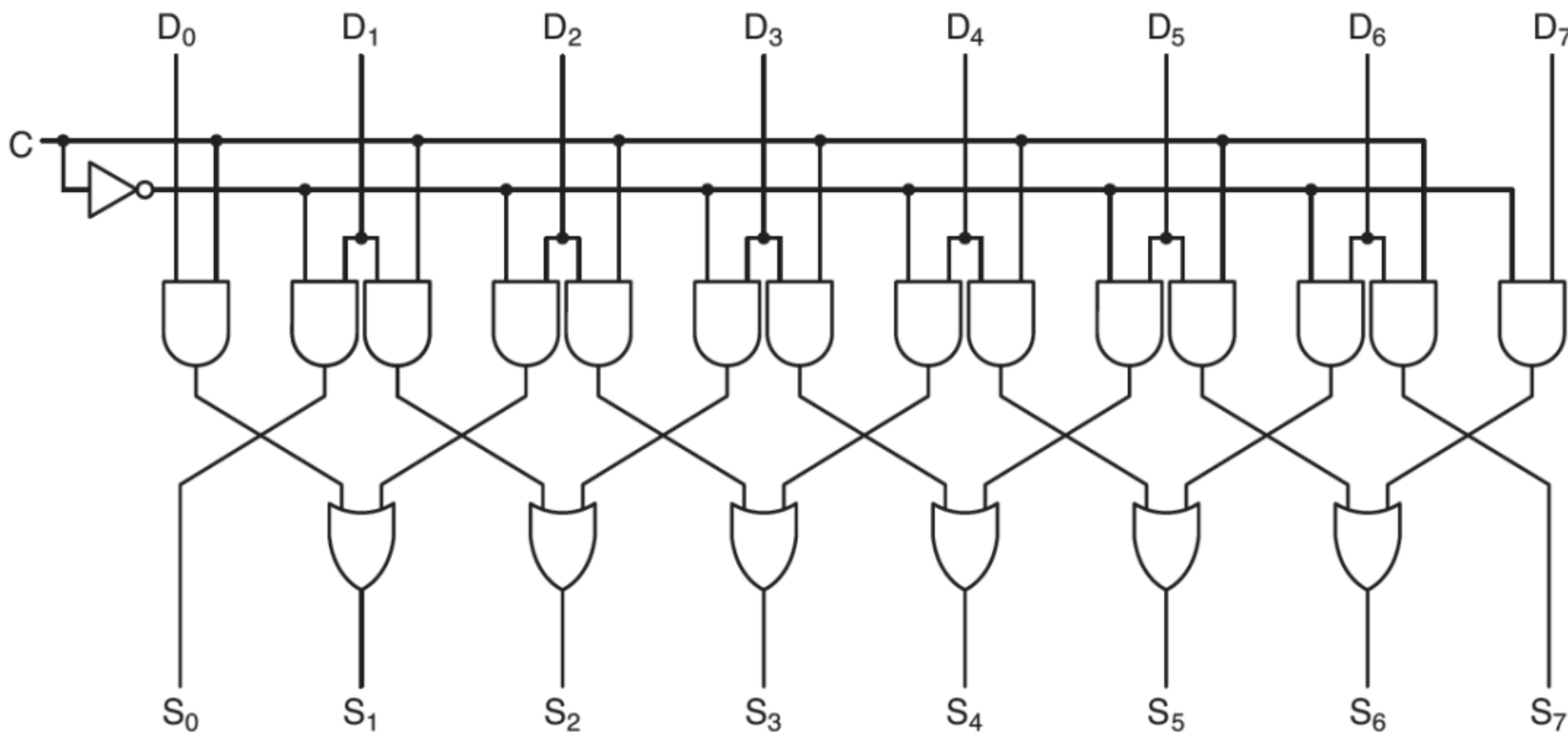
- Deslocadores
- Somadores
  - Parcial
  - Completo

# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Nosso primeiro circuito aritmético é um **deslocador** de oito entradas e oito saídas.



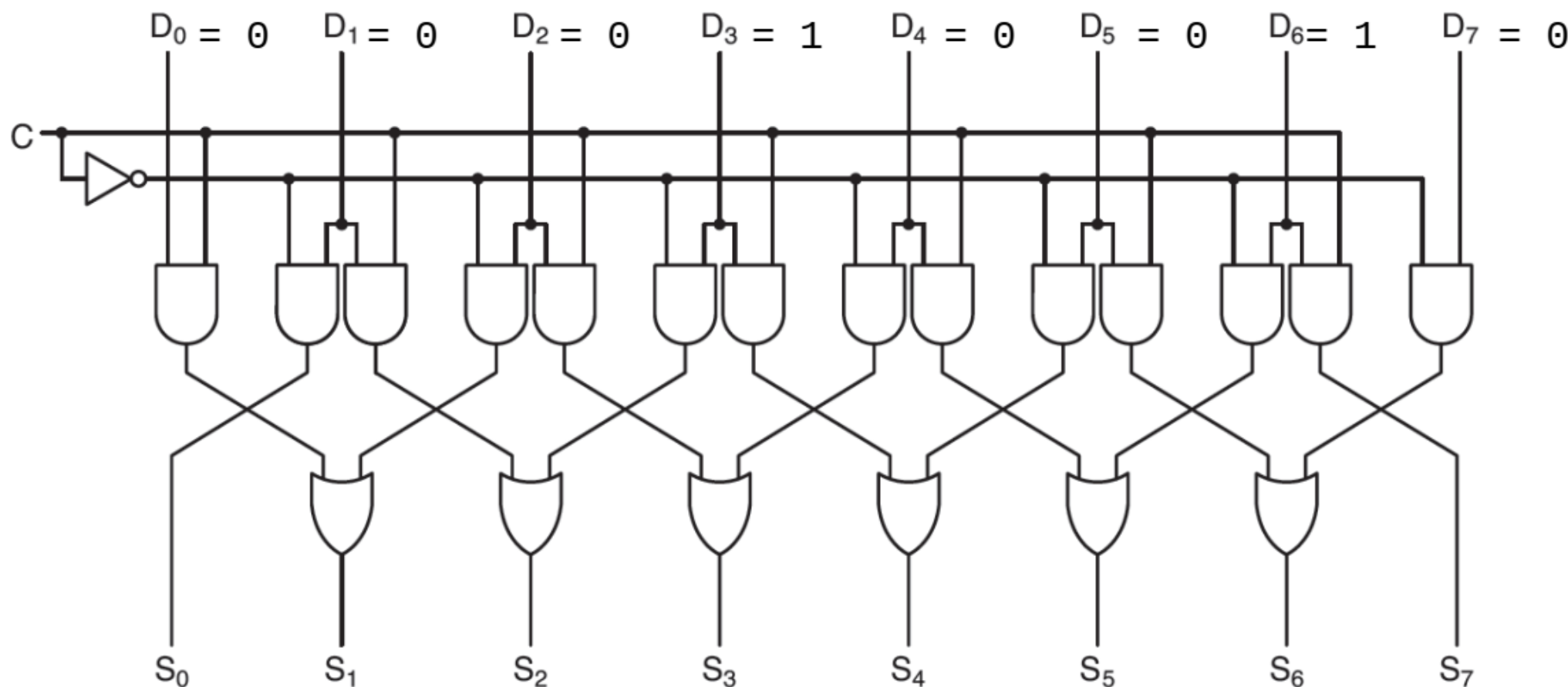


# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Nosso primeiro circuito aritmético é um **deslocador** de oito entradas e oito saídas.

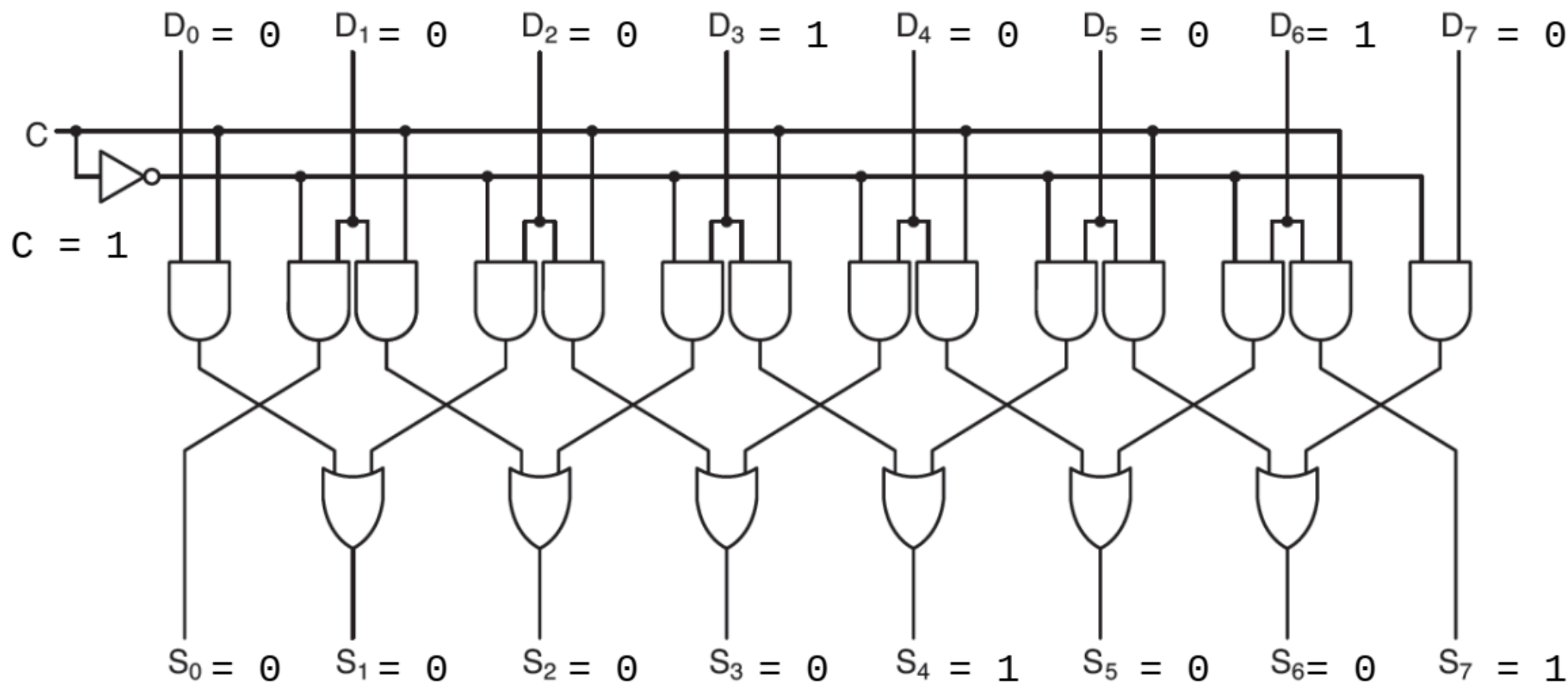


# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

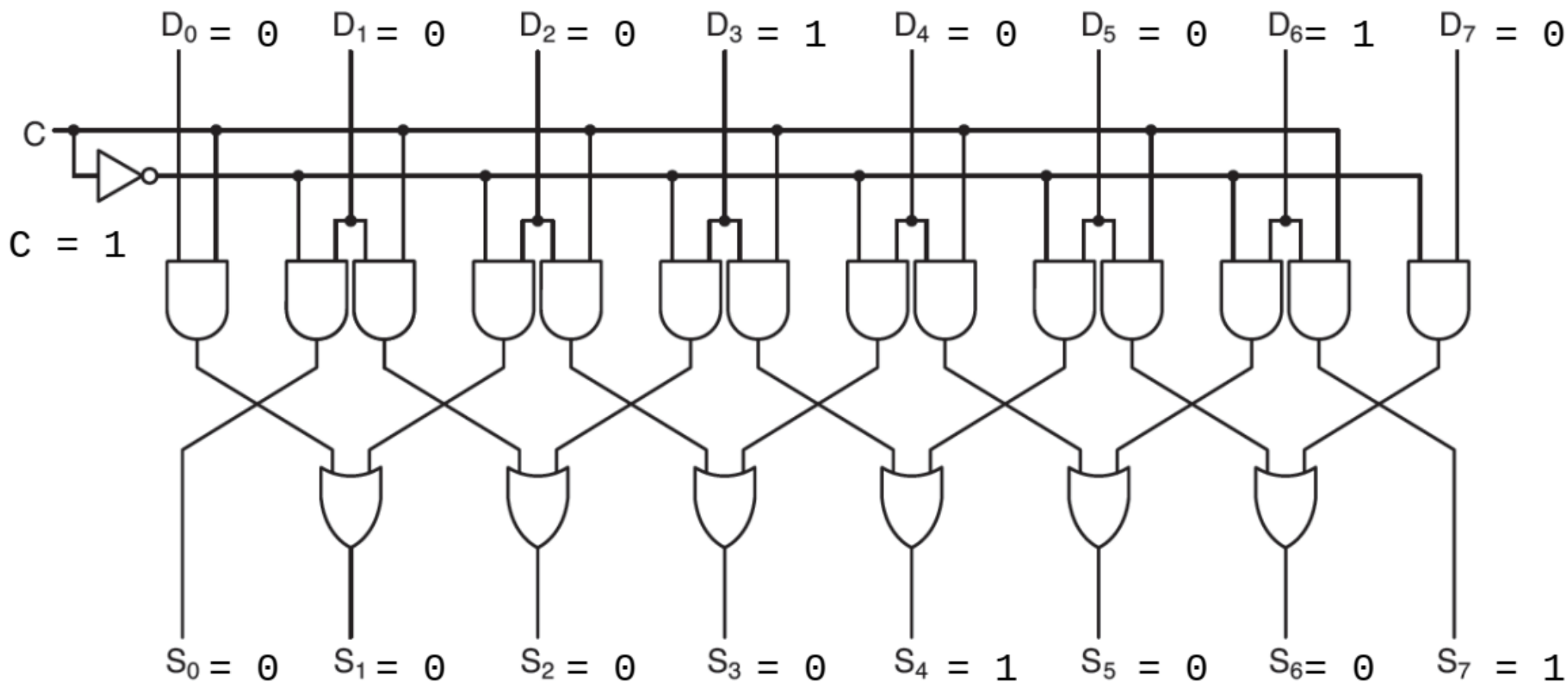
- Nosso primeiro circuito aritmético é um **deslocador** de oito entradas e oito saídas.



# CIRCUITOS ARITMÉTICOS



- Nosso primeiro circuito aritmético é um **deslocador** de oito entradas e oito saídas.

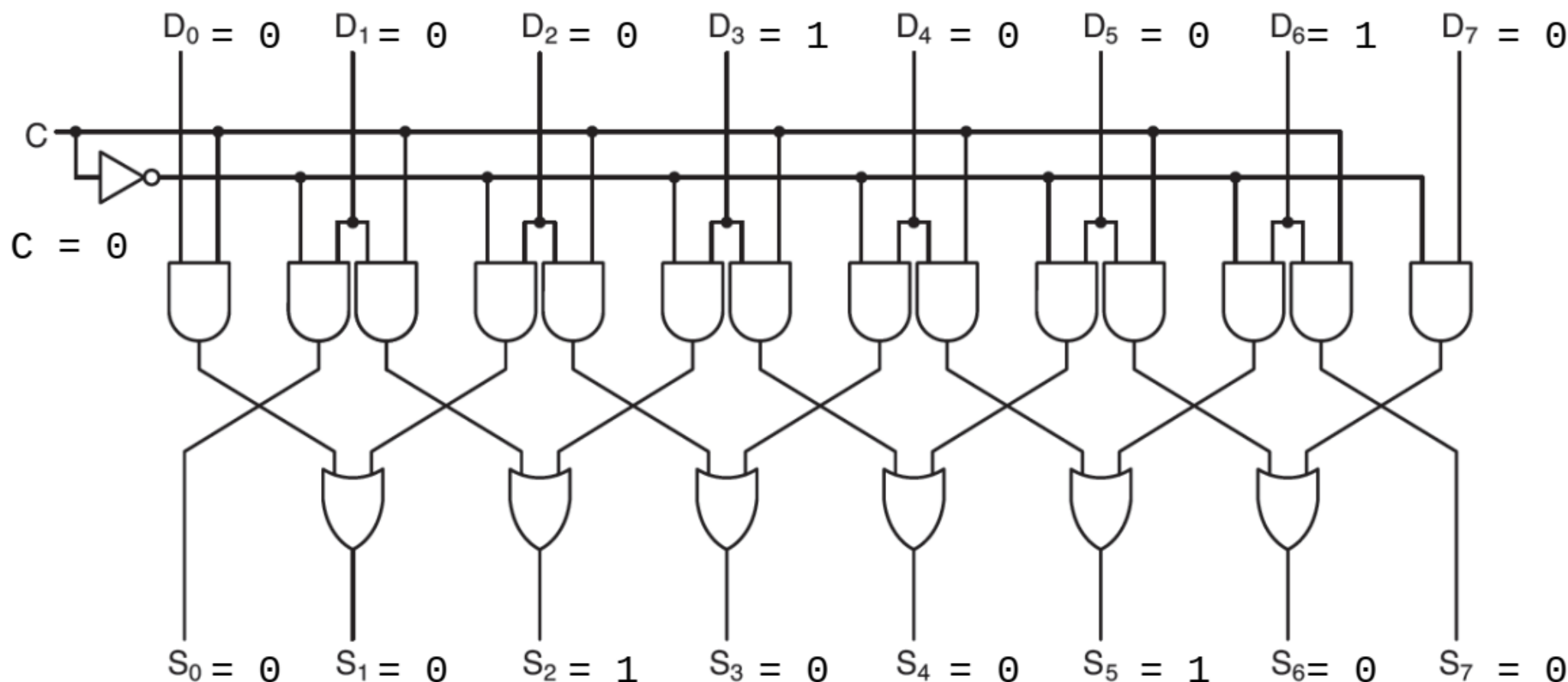


# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Nosso primeiro circuito aritmético é um **deslocador** de oito entradas e oito saídas.



# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Um computador que não possa somar números inteiros é quase inimaginável
- Um hardware para efetuar adição é parte essencial em toda CPU
- Somador Parcial
- Somador Completo

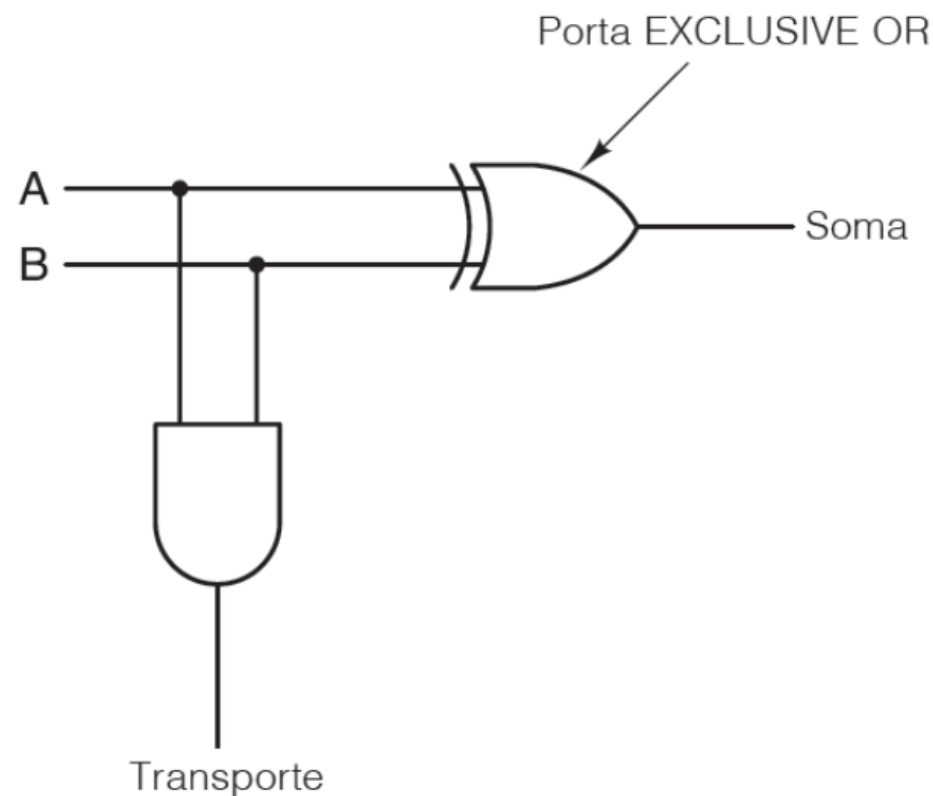
# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Um circuito para calcular o bit de soma e o de transporte é conhecido como um meio-somador (somador parcial)

A	B	Soma	Transporte
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1



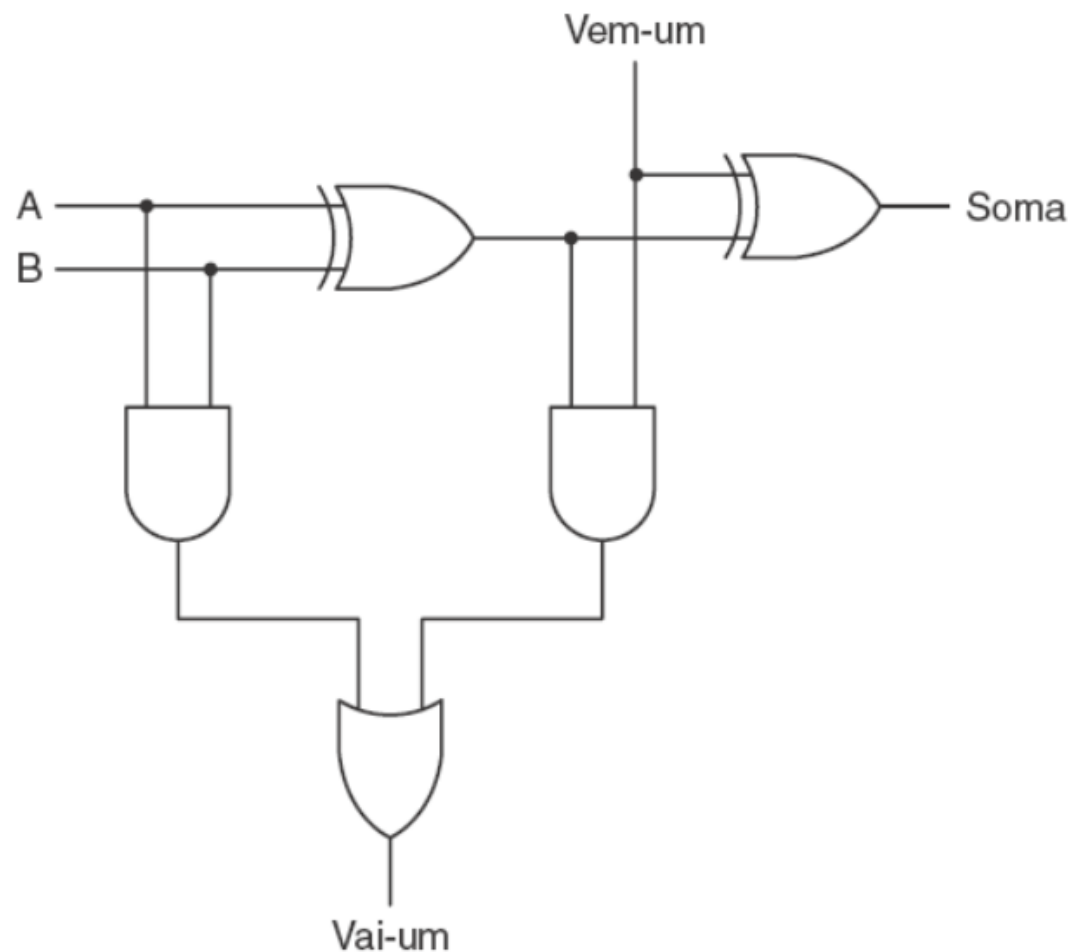
# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

## ■ Somador completo

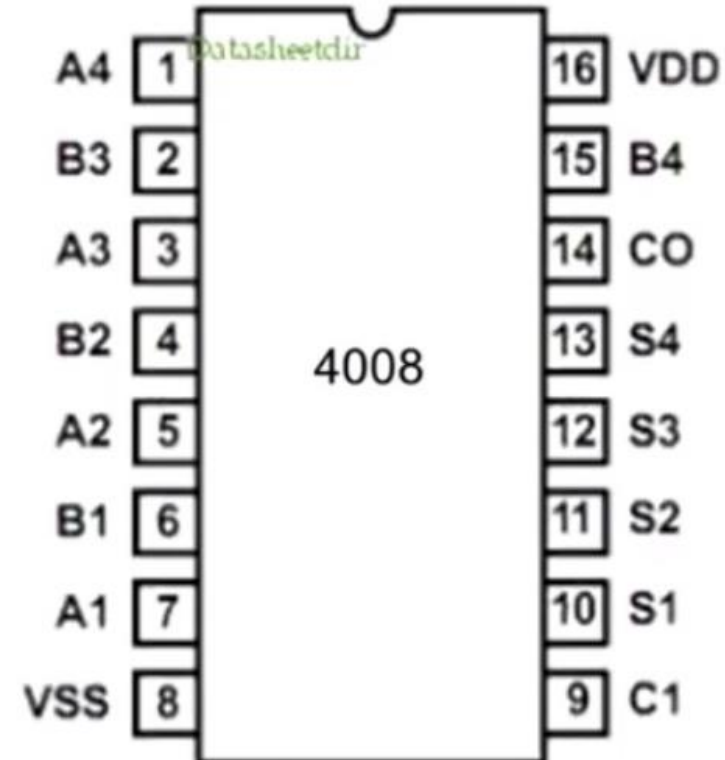
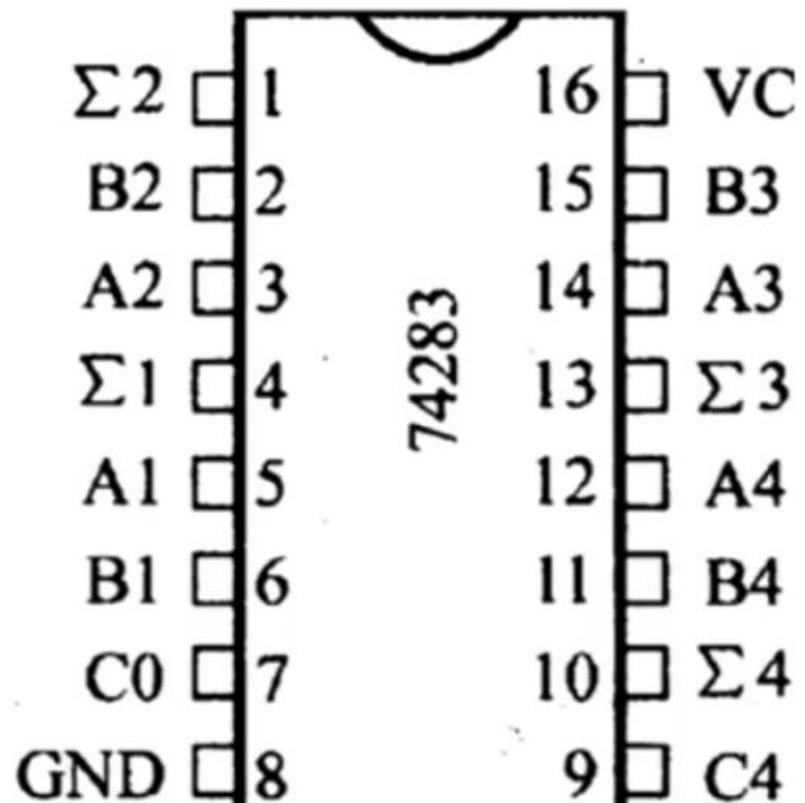
A	B	Vem-um	Soma	Vai-um
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ

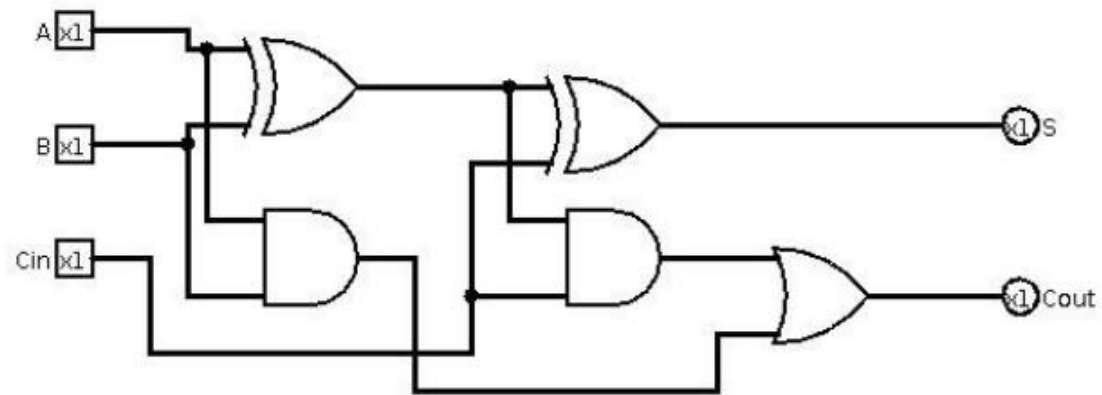




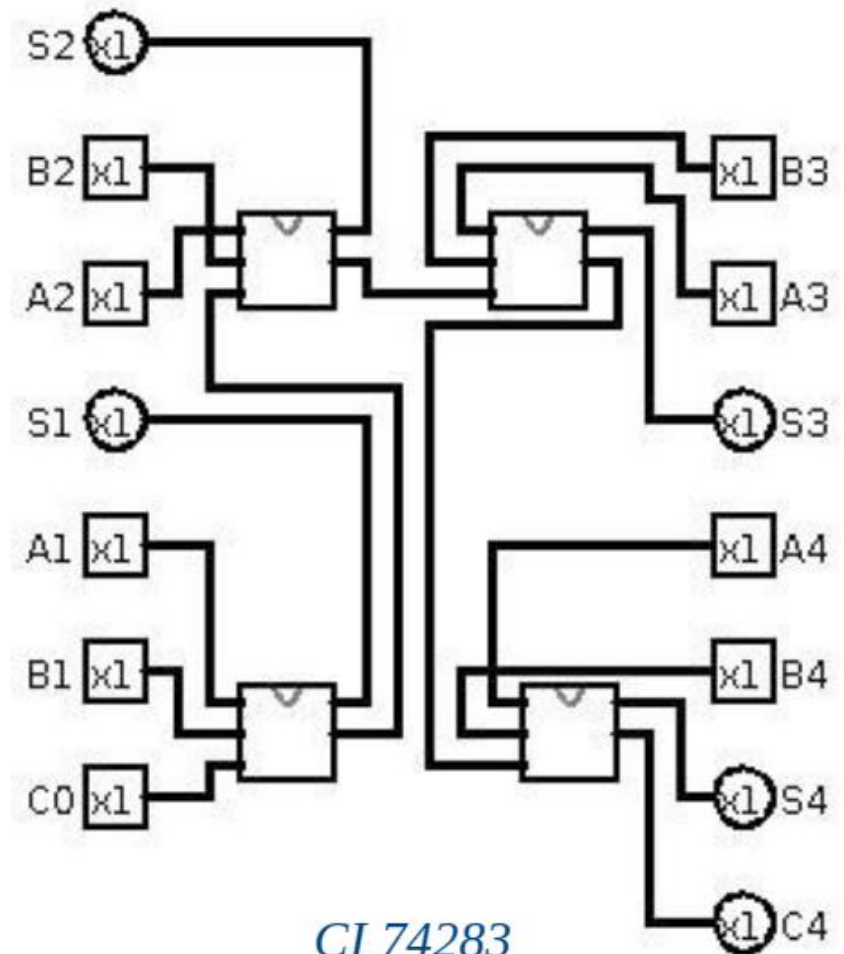
# CIRCUITOS ARITMÉTICOS



UNIVERSIDADE  
FEDERAL DO CEARÁ



*Somador completo*



*CI 74283*

# UNIDADES LÓGICA E ARITMÉTICA



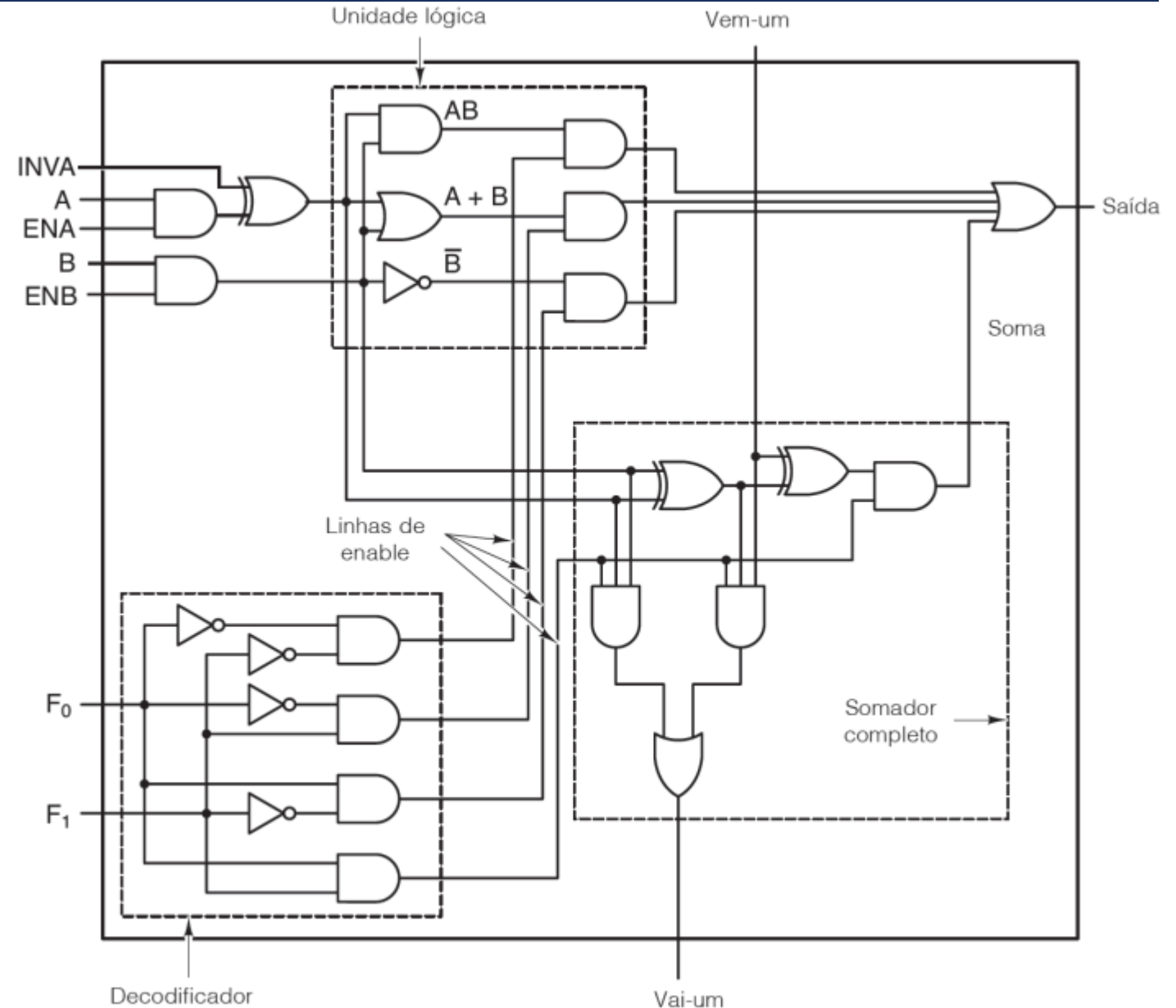
UNIVERSIDADE  
FEDERAL DO CEARÁ

- Grande parte dos computadores contém um único circuito para efetuar AND, OR e soma de duas palavras de máquina.
- No caso típico, tal circuito para palavras de  $n$  bits é composto de  $n$  circuitos idênticos para as posições individuais de bits.

# UNIDADES LÓGICA E ARITMÉTICA

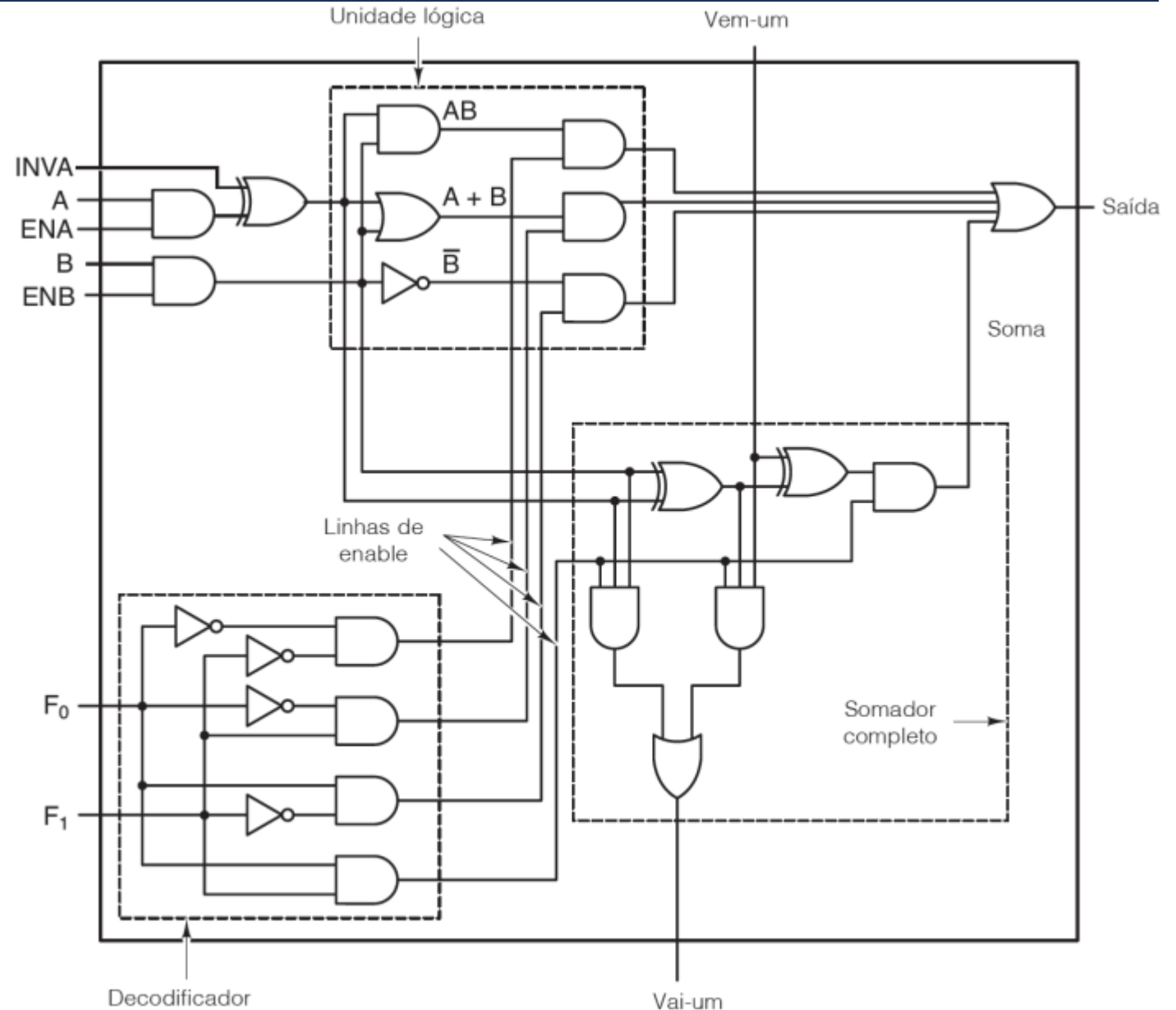
## ULA de 1 bit

- Entrada: A e B
- É possível forçar A ou B para negando ENA e ENB, respectivamente.
- É possível obter  $\bar{A}$  ativando INVA.
- Decodificador para as quatro operações com base nos sinais de controle F0 e F1.
- Somador completo



# UNIDADES LÓGICA E ARITMÉTICA

- ULA de 1 bit
- Pode calcular qualquer uma das quatro funções – A **and** B, A **or** B,  $\bar{B}$  ou  $A + B$ , dependendo de as linhas de entrada de seleção de função  $F_0$  e  $F_1$  conterem 00, 01, 10 ou 11 (binário).

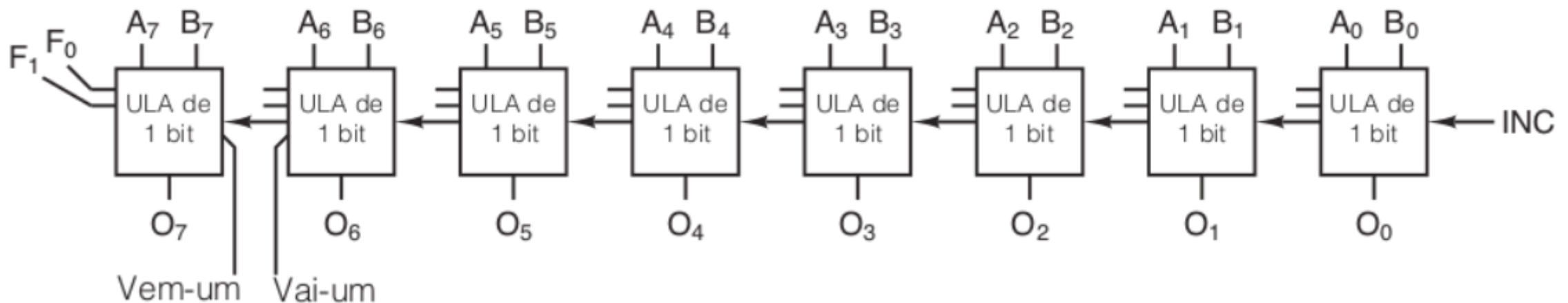


# UNIDADES LÓGICA E ARITMÉTICA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- 8 ULAs de 1 bit conectadas para formar uma ULA de 8 bits.
  - Por simplicidade, os sinais de habilitação e de inversão são omitidos
  - **INC** só é necessário para operação de soma. Para quaisquer outras operações, a variação de **INC** não altera o resultado (cálculos de soma como  $A+1$  e  $A+B+1$ ).

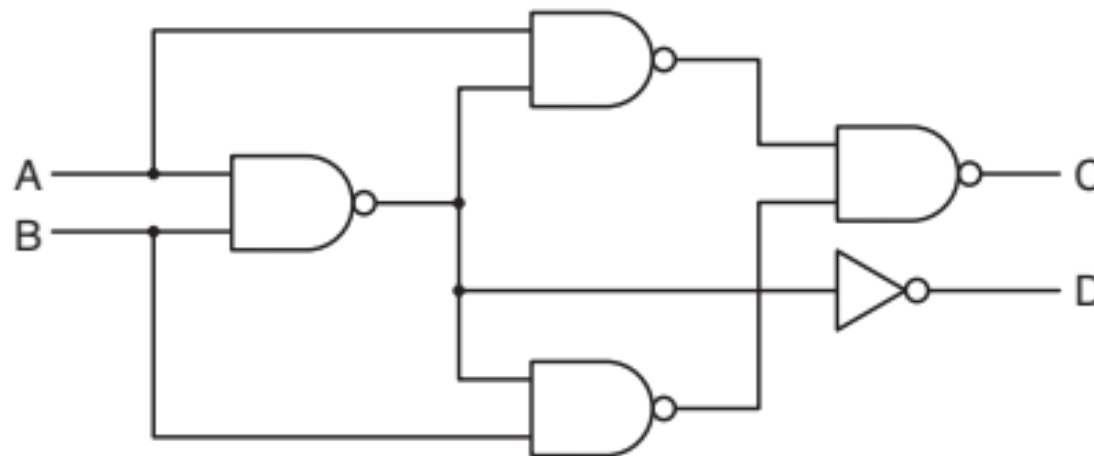


# ATIVIDADES



UNIVERSIDADE  
FEDERAL DO CEARÁ

1. O que esse circuito faz?



2. Desenhe o diagrama lógico para um demultiplexador de 2 bits, um circuito cuja única linha de entrada é direcionada para uma das quatro linhas de saída dependendo do estado das duas linhas de controle.



# CIRCUITOS LÓGICOS

CLOCKS



# CLOCKS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Em inglês, significa **relógio**.
- Para que serve o relógio? Para que haja sincronia entre várias pessoas.
  - Ex.: No caso em que uma pessoa que marca uma consulta médica às 9h, esse horário é um acordo em comum entre o paciente e o médico – quando o relógio do paciente e o do médico marcarem 9h, a pessoa estará no consultório e o médico terá de lhe atender. Às 9h, ambos deverão estar sincronizados – supõe-se que pouco antes, ambos se preparem para às 9h estarem disponíveis um para o outro.
- No computador, o clock é uma espécie de **relógio que mantém os circuitos sincronizados**



# CLOCKS



UNIVERSIDADE  
FEDERAL DO CEARÁ

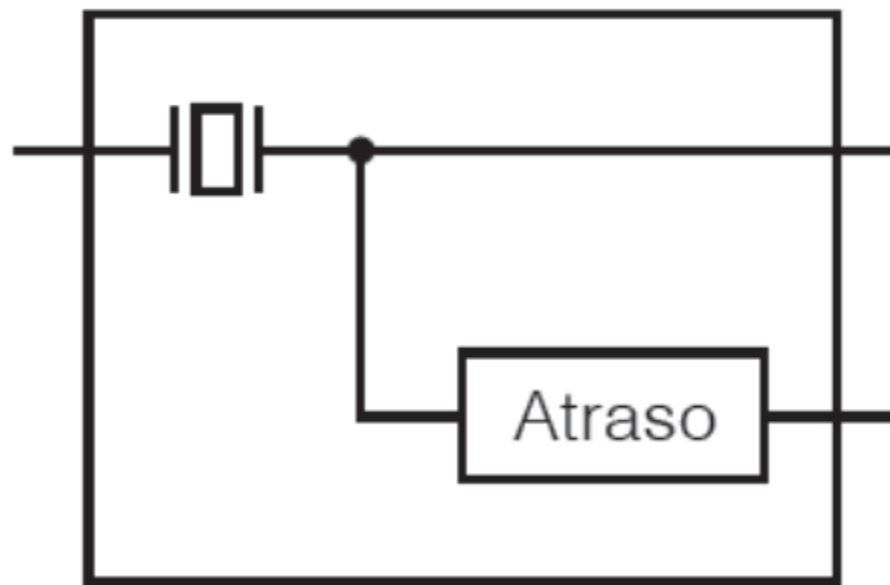
- Em muitos circuitos digitais, a ordem em que os eventos ocorrem é crítica. Às vezes um evento deve preceder outro, às vezes dois eventos devem ocorrer simultaneamente. Para permitir que os projetistas consigam as relações de temporização requeridas, muitos circuitos digitais usam clocks para prover sincronização.
- Nesse contexto, um clock é um circuito que emite uma série de pulsos com uma largura de pulso precisa e intervalos precisos entre pulsos consecutivos.
- O intervalo de tempo entre as arestas correspondentes de dois pulsos consecutivos é denominado tempo de ciclo de clock.
  - Em geral, as frequências de pulso estão entre 100 MHz e 4 GHz, correspondendo a ciclos de clock de 10 nanossegundos a 250 picossegundos.

# CLOCKS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Um **clock** é um circuito que emite uma série de pulsos com uma largura de pulso precisa e intervalos precisos entre pulsos consecutivos.
- O intervalo de tempo entre as arestas correspondentes de dois pulsos consecutivos é denominado tempo de *ciclo de clock*.

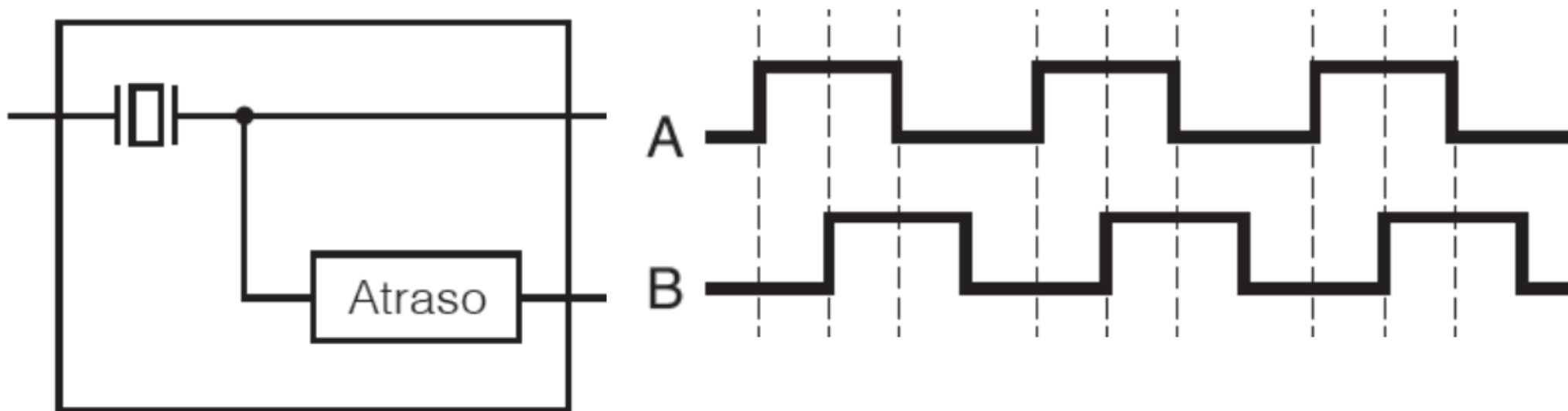


# CLOCKS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Um **clock** é um circuito que emite uma série de pulsos com uma largura de pulso precisa e intervalos precisos entre pulsos consecutivos.
- Durante um *ciclo de clock* diversos eventos podem ocorrer. Se esses eventos devem ocorrer em ordem específica, o ciclo deve ser dividido em subciclos.

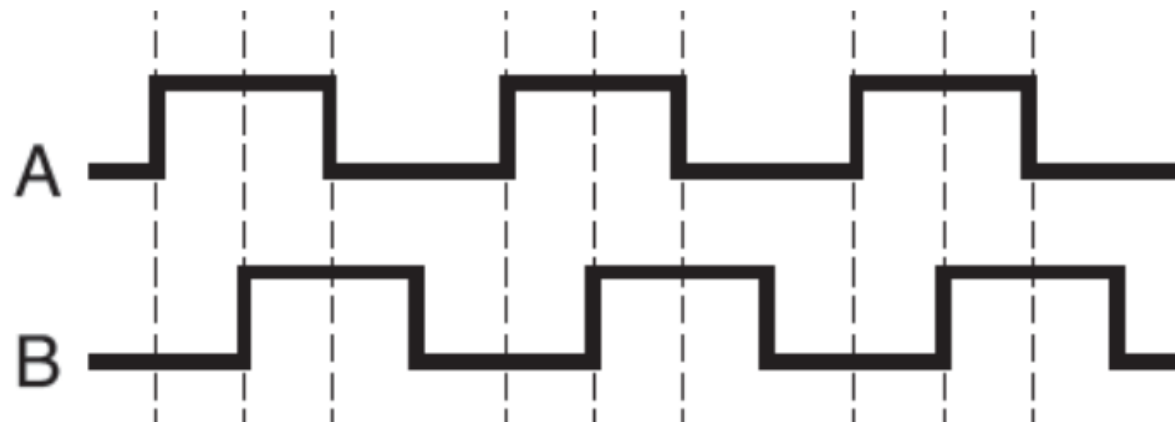
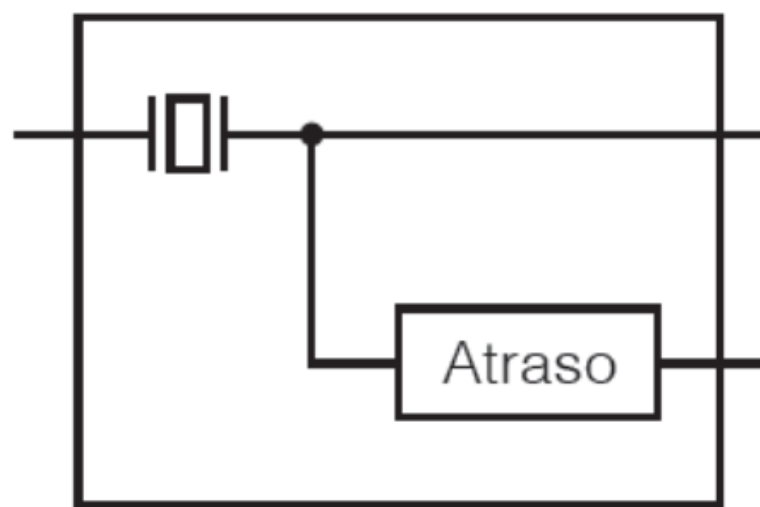


# CLOCKS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- O diagrama de temporização dá quatro referências de tempo para eventos discretos:
  1. Fase ascendente de A.
  2. Fase descendente de A.
  3. Fase ascendente de B.
  4. Fase descendente de B.



# CLOCKS



UNIVERSIDADE  
FEDERAL DO CEARÁ

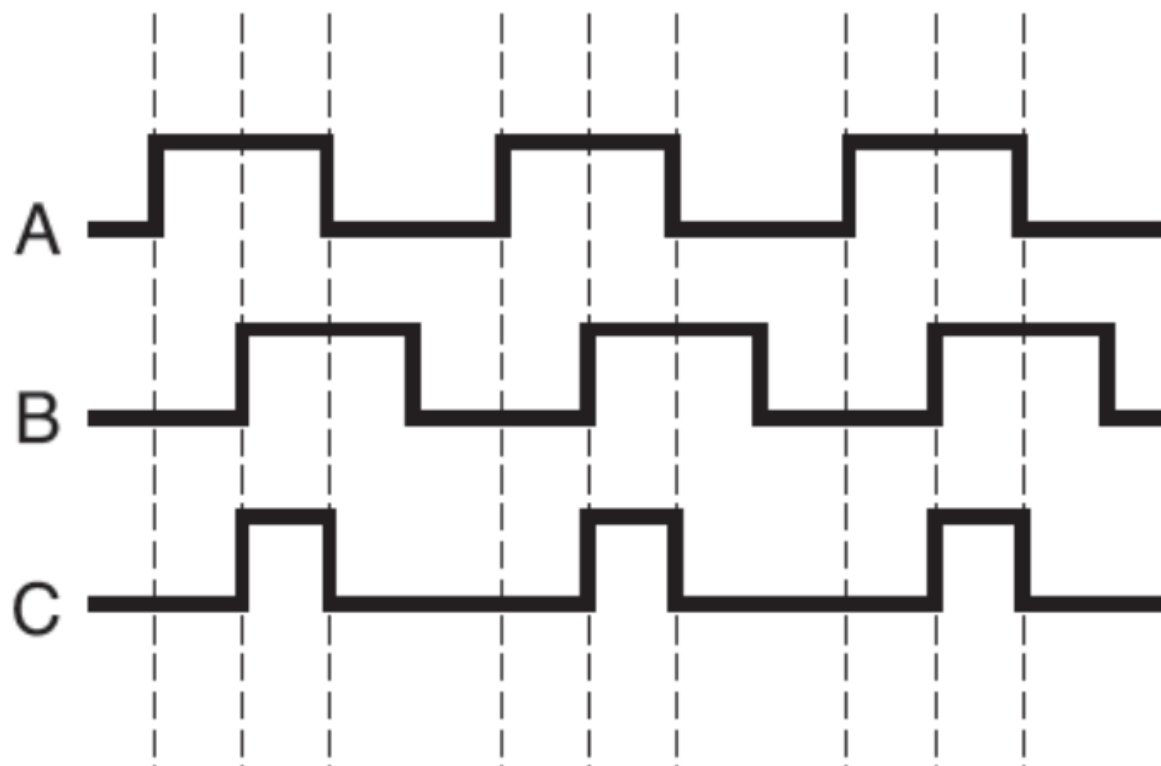
- Vinculando diferentes eventos às várias fases, pode-se conseguir a sequência requerida. Se forem necessárias mais do que quatro referências de tempo dentro de um ciclo de clock, podem-se puxar mais linhas da linha primária, com diferentes atrasos, se for preciso.
- No exemplo anterior podem-se distinguir quatro intervalos distintos:  $\bar{A}$  and  $\bar{B}$ ,  $\bar{A}$  and  $B$ ,  $A$  and  $\bar{B}$  e  $A$  and  $B$ .

# CLOCKS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Geração de um clock assimétrico.
- Para gerar um trem de pulsos assimétrico, o clock básico é deslocado usando um circuito de atraso e efetuando uma operação AND com o sinal original.





# CIRCUITOS LÓGICOS

CIRCUITOS SEQUENCIAIS



# MEMÓRIA DE 1 BIT



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Para criar uma memória de 1 bit ("latch"), precisamos de um circuito que "se lembre", de algum modo, de valores de entrada anteriores.
- Tal circuito pode ser construído com base em duas portas **NOR**:
  - Circuitos análogos podem ser construídos com portas NAND, porém, não vamos mais mencioná-los porque são conceitualmente idênticos às versões nor.

A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

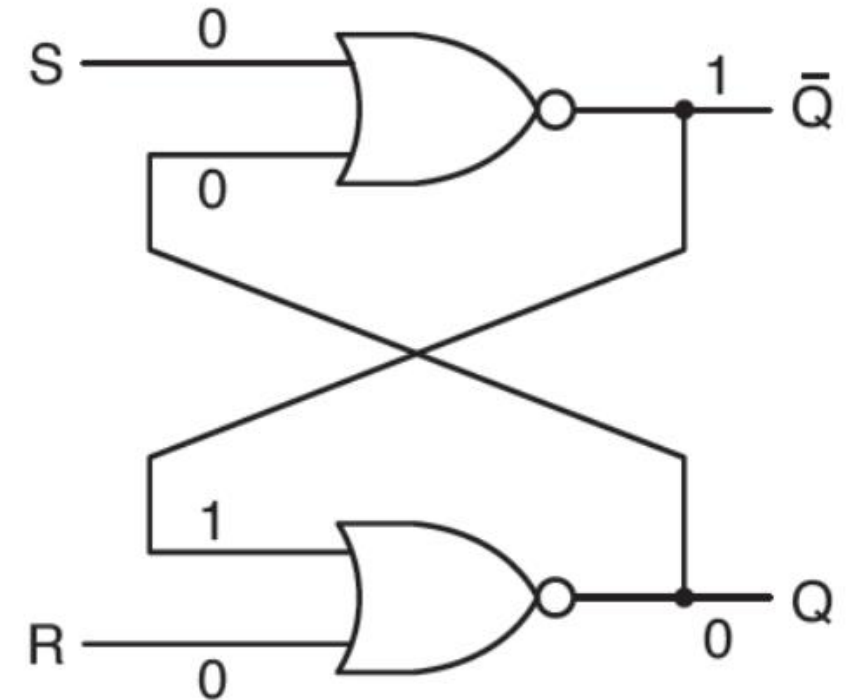


# MEMÓRIA DE 1 BIT



UNIVERSIDADE  
FEDERAL DO CEARÁ

- O circuito a seguir é denominado Latch SR.
- Possui duas entradas  $S$ , para ativar (*setting*) o latch e  $R$ , para restaurá-lo (*resetting*).
- O circuito possui duas saídas,  $Q$  e  $\bar{Q}$

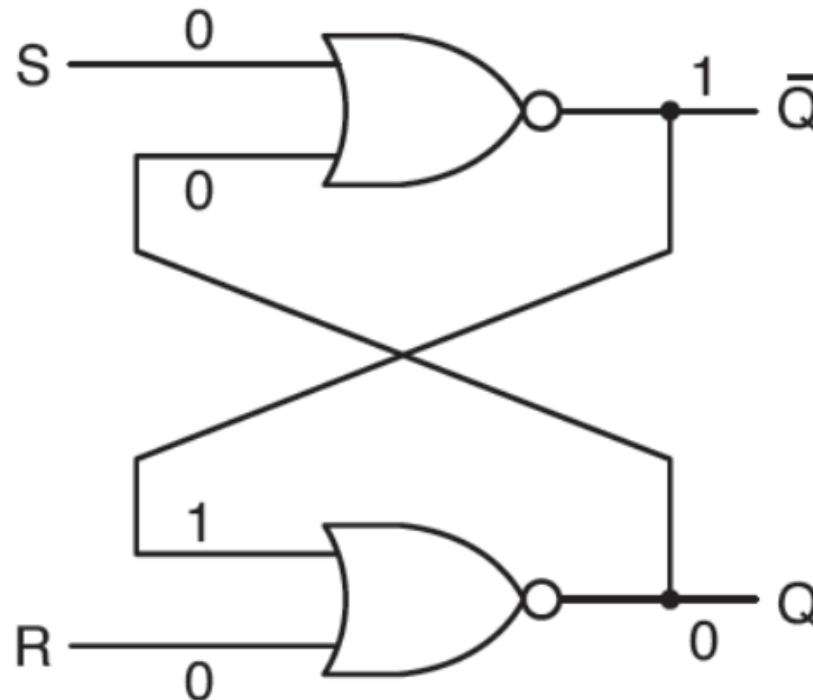


# MEMÓRIA DE 1 BIT



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Latch NOR no estado 0. Tabela verdade para NOR



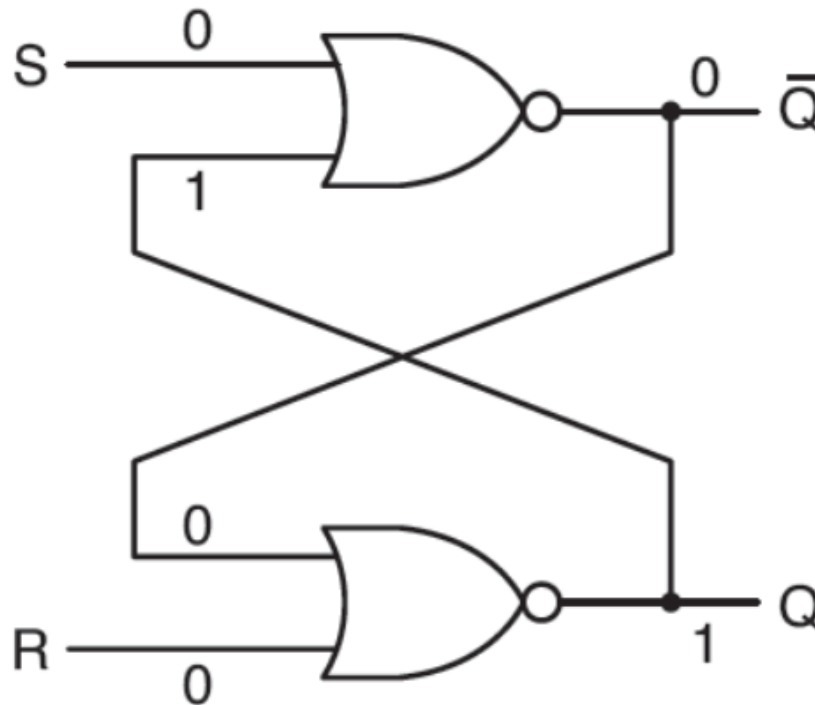
A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

# MEMÓRIA DE 1 BIT



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Latch NOR no estado 1. Tabela verdade para NOR



A	B	NOR
0	0	1
0	1	0
1	0	0
1	1	0

# MEMÓRIA DE 1 BIT



UNIVERSIDADE  
FEDERAL DO CEARÁ

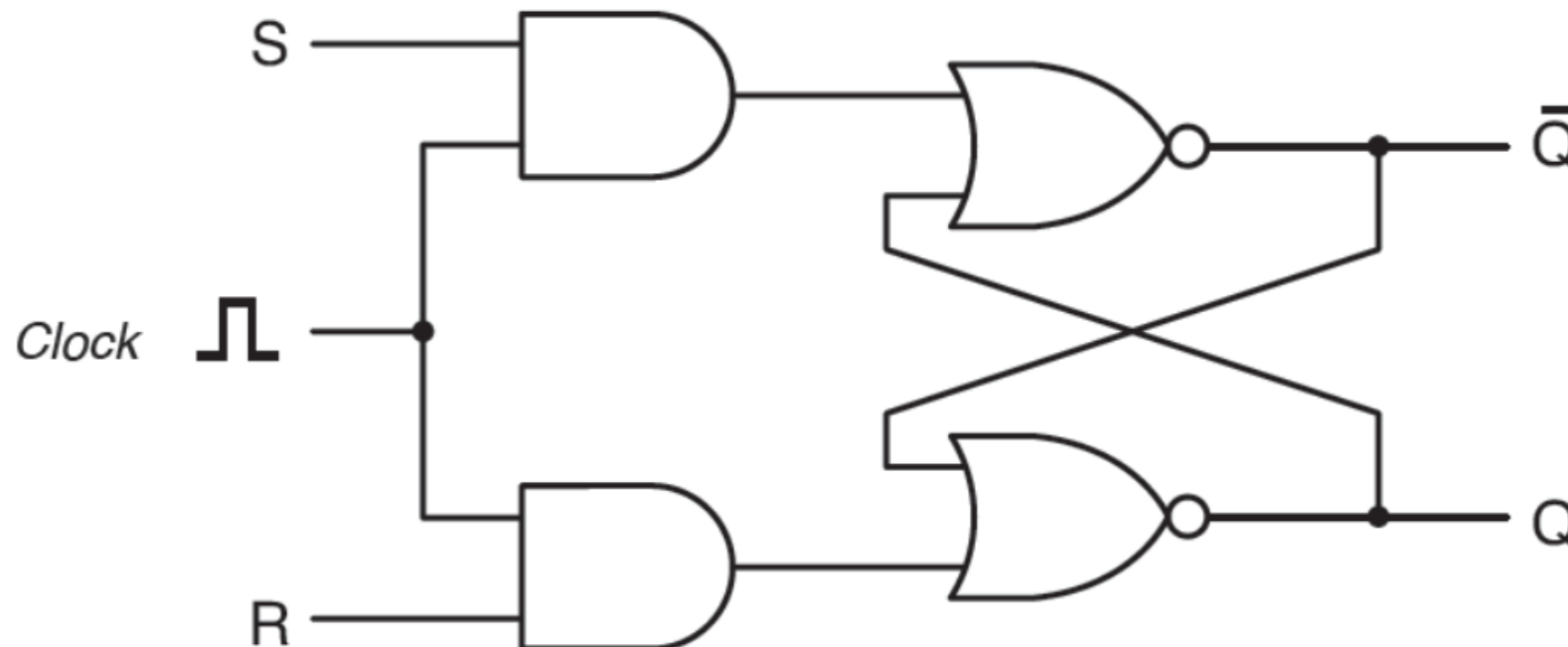
- Nossa conclusão é simples: para  $R = S = 0$ , o latch tem dois estados estáveis, que denominaremos 0 e 1, dependendo de  $Q$ .
- Suponha que se  $S$  se torne igual a 1 enquanto  $Q = 0$ .
  - Ativar o estado  $S$  muda o estado de 0 para 1.
- Definir  $R$  em 1 quando o latch está em estado 0 não tem efeito algum.
- Definir  $S$  em 1 quando o latch está em estado 1 não tem efeito algum.
- Definir  $R$  em 1 leva o latch ao estado  $Q = 0$ .
- O circuito “se lembra” se foi  $S$  ou  $R$  definido por último. Usando essa propriedade podemos construir memórias de computadores.

# MEMÓRIA DE 1 BIT



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Muitas vezes é conveniente impedir que o *latch* mude de estado. Para atingir esse objetivo, fazemos uma ligeira modificação no circuito básico para obter um *latch* SR com clock.



# MEMÓRIA DE 1 BIT



UNIVERSIDADE  
FEDERAL DO CEARÁ

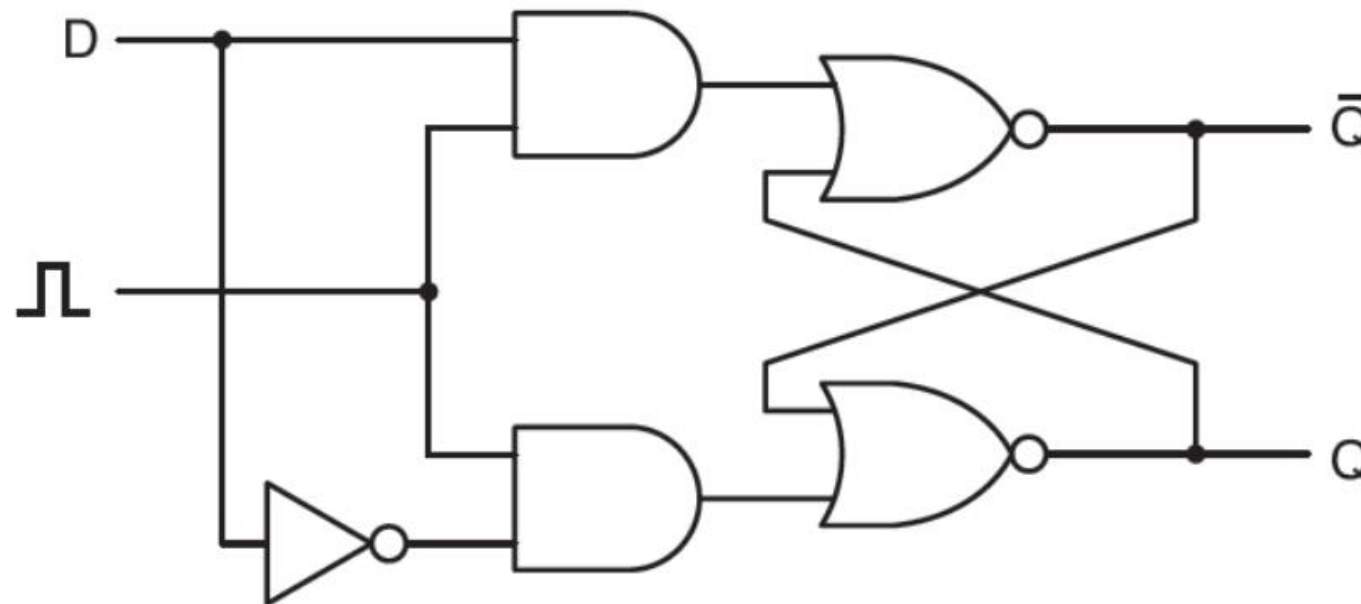
- Até aqui evitamos falar no que acontece quando ambos, S e R, são 1, por uma boa razão: o circuito se torna não determinístico quando ambos, R e S, finalmente retornam a 0. O único estado coerente para  $S = R = 1$  é  $Q = \bar{Q} = 0$ ; porém, assim que ambas as entradas voltam para 0, o latch deve saltar para um de seus dois estados estáveis.
- Se quaisquer das entradas cair para 0 antes da outra, a que permanecer em 1 por mais tempo vence, porque, quando apenas uma entrada for 1, ela força o estado. Se ambas as entradas voltarem a 0 ao mesmo tempo (o que é muito improvável), o latch salta aleatoriamente para um de seus estados estáveis.

# MEMÓRIA DE 1 BIT



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Uma boa maneira de resolver a instabilidade do latch SR (causada quando  $S = R = 1$ ) é evitar que ela ocorra.
- Quando o clock for 1, o valor corrente de D é lido e armazenado no latch. Esse circuito, denominado **latch D com clock**.





# CIRCUITOS LÓGICOS

FLIP-FLOPS





# FLIP-FLOPS



UNIVERSIDADE  
FEDERAL DO CEARÁ

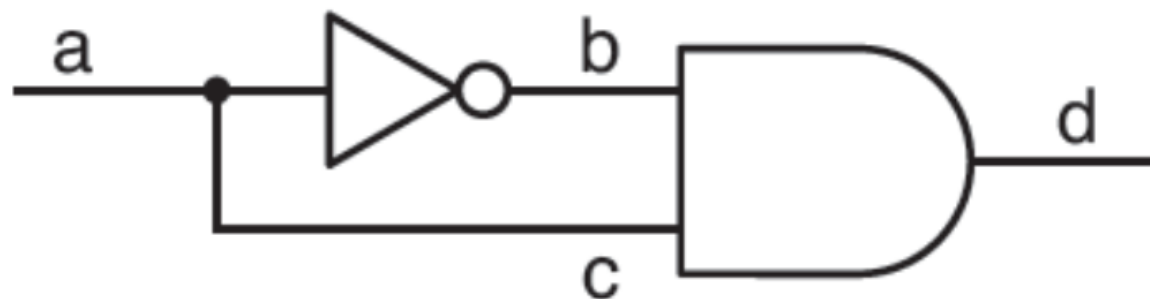
- Na variante, denominada flip-flop, a transição de estado não ocorre quando o clock é 1, mas durante a transição de 0 para 1 (borda ascendente), ou de 1 para 0 (borda descendente).
- Gerador de pulso

# FLIP-FLOPS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Em muitos circuitos é necessário ler o valor em determinada linha em dado instante, e armazená-lo.
- Na variante, denominada flip-flop, a transição de estado não ocorre quando o clock é 1, mas durante a transição de 0 para 1 (borda ascendente), ou de 1 para 0 (borda descendente).
- Gerador de pulso





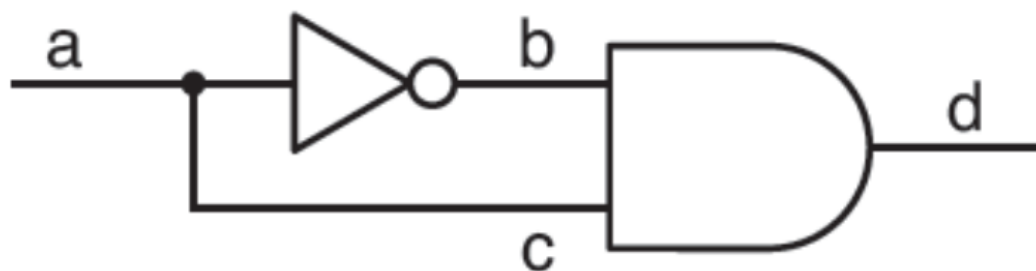
- Para dar ênfase, vamos repetir qual é a diferença entre um flip-flop e um latch. Um flip-flop é disparado pela borda, enquanto um latch é disparado pelo nível.
- A primeira vista, poderia parecer que a saída da porta and seria sempre zero, uma vez que a operação AND de qualquer sinal com seu inverso é zero, mas a situação é um pouco diferente disso.
- O inversor tem um atraso de propagação pequeno, mas não zero, e é esse atraso que faz o circuito funcionar.

# FLIP-FLOPS

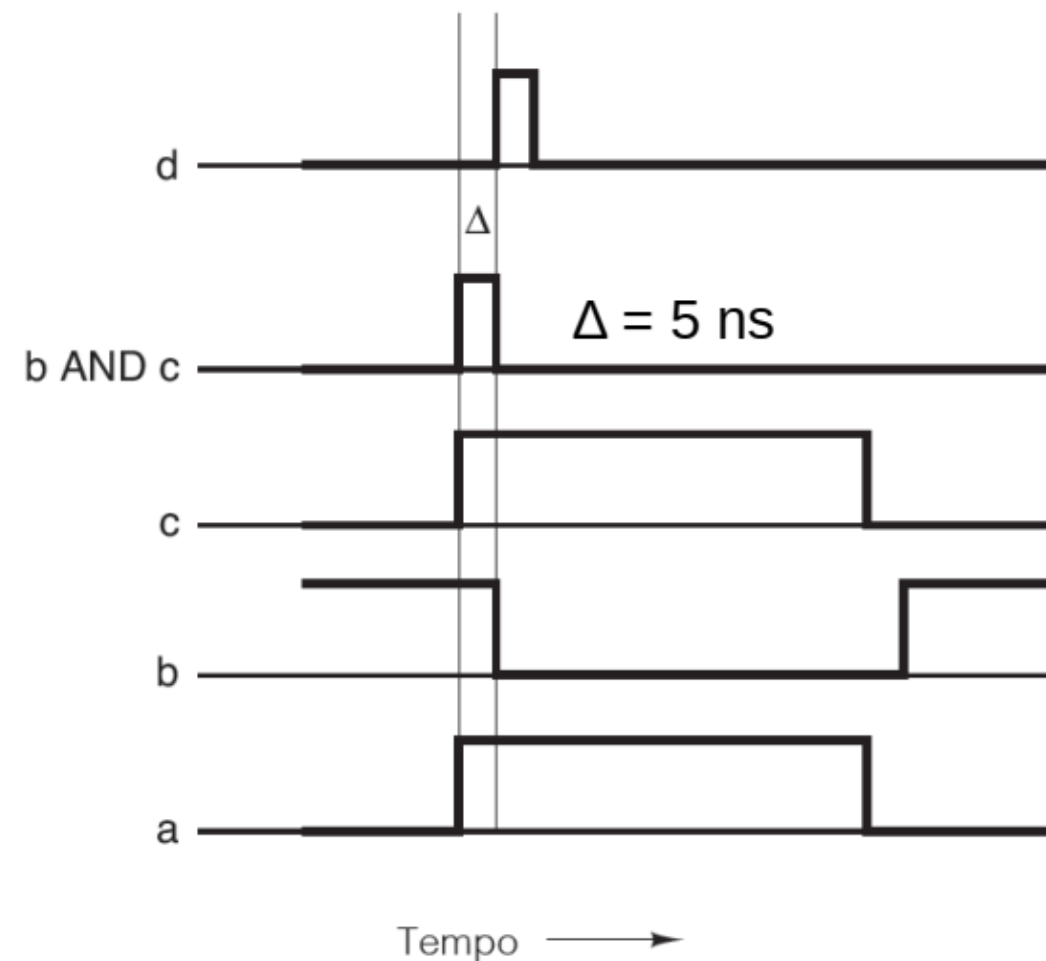


UNIVERSIDADE  
FEDERAL DO CEARÁ

- Temporização em quatro pontos do circuito.



- Na prática, o próprio sinal c possui um atraso em relação ao sinal a, porém é bem menor do que o atraso na inversora



# FLIP-FLOPS



UNIVERSIDADE  
FEDERAL DO CEARÁ

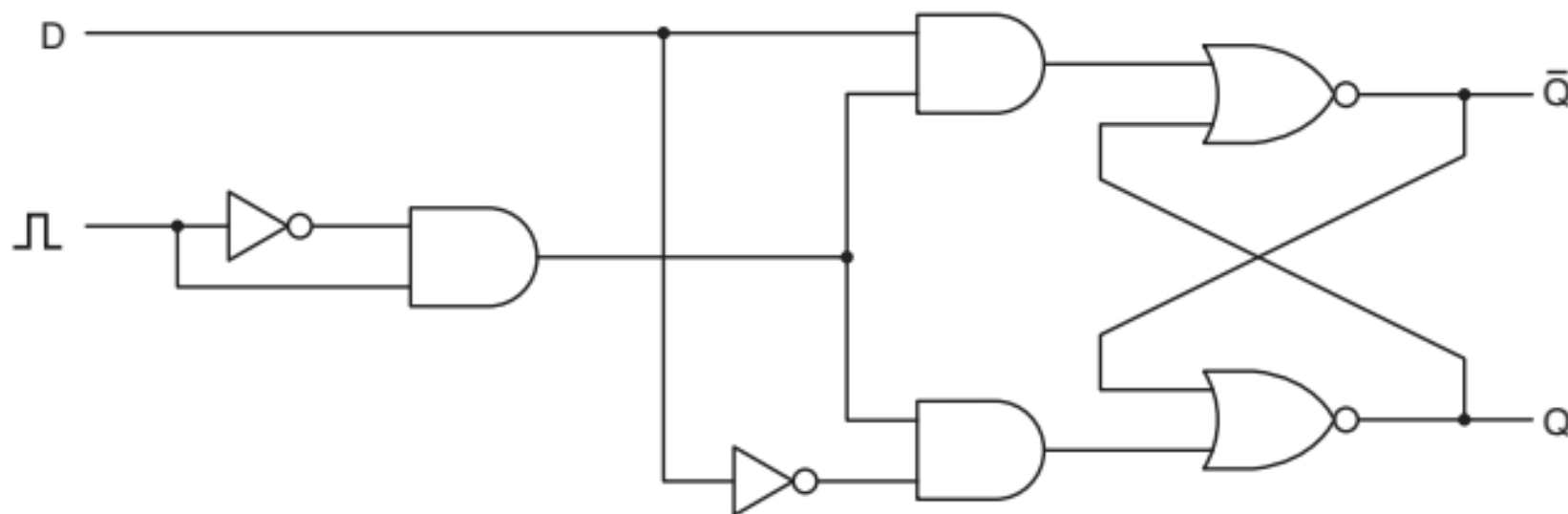
- Quando se efetua uma operação and com as entradas para a porta and, b e c, o resultado é um pulso curto, onde a largura do pulso,  $\Delta$ , é igual ao atraso da porta do inversor, em geral 5 ns ou menos.
- A saída da porta AND é exatamente esse pulso deslocado pelo atraso da porta AND.
- Esse deslocamento de tempo significa apenas que o latch D será ativado com um atraso fixo após a fase ascendente do clock, mas não tem efeito sobre a largura do pulso.
- Em uma memória com tempo de ciclo de 10 ns, um pulso de 1 ns para informar quando ler a linha D pode ser curto o bastante.

# FLIP-FLOPS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Vale a pena observar que esse projeto de flip-flop é atraente porque é fácil de entender, embora, na prática, sejam usados flip-flops mais sofisticados.

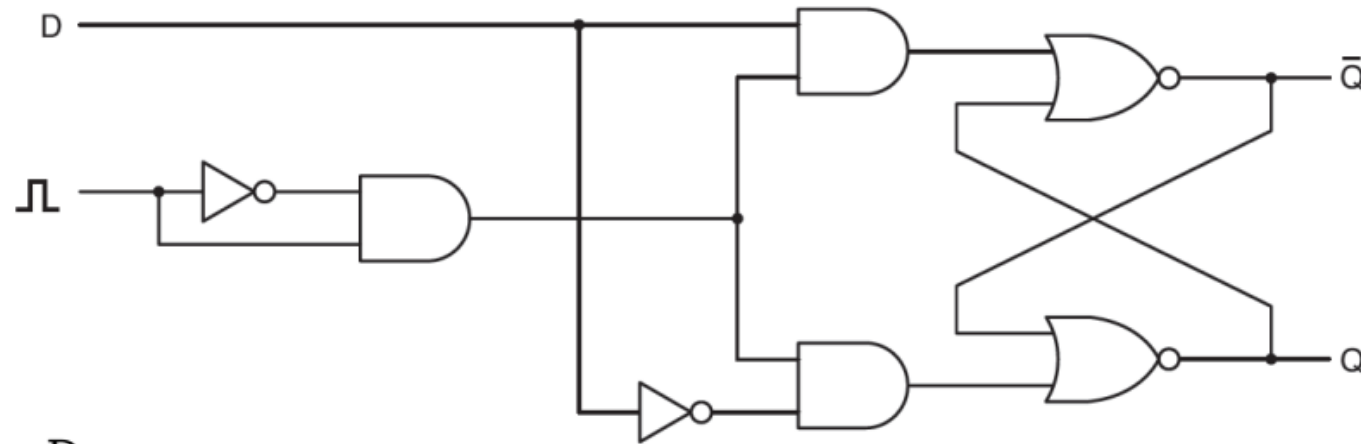


# FLIP-FLOPS

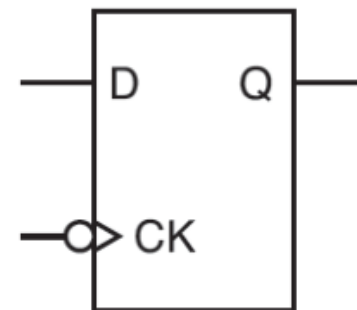
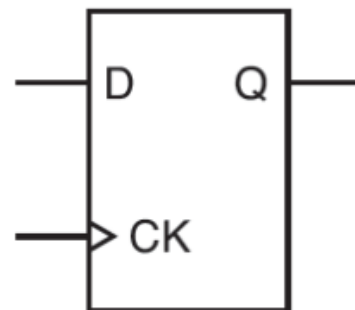
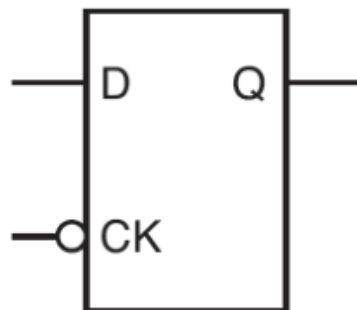
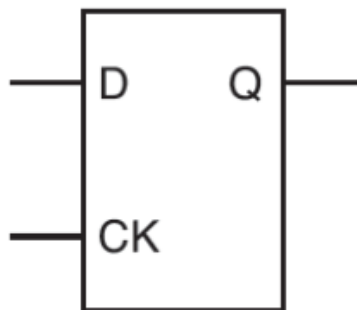


UNIVERSIDADE  
FEDERAL DO CEARÁ

- *Flip-flop D*



- *Latches e flip-flops D*





# CIRCUITOS LÓGICOS

REGISTRADORES



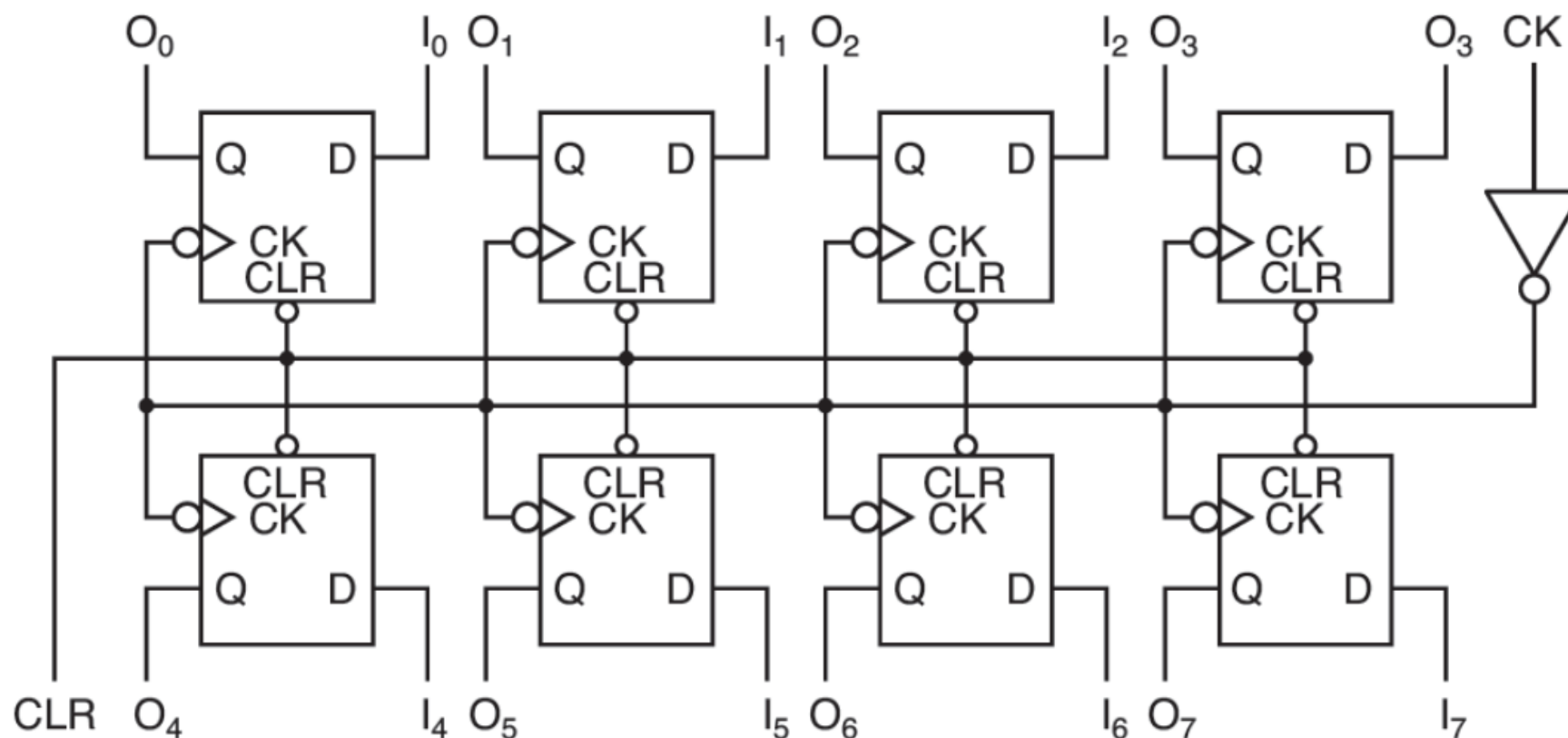


# REGISTRADORES



UNIVERSIDADE  
FEDERAL DO CEARÁ

- O registrador abaixo mostra como oito flip-flops podem ser ligados para formar um registrador armazenador de 8 bits

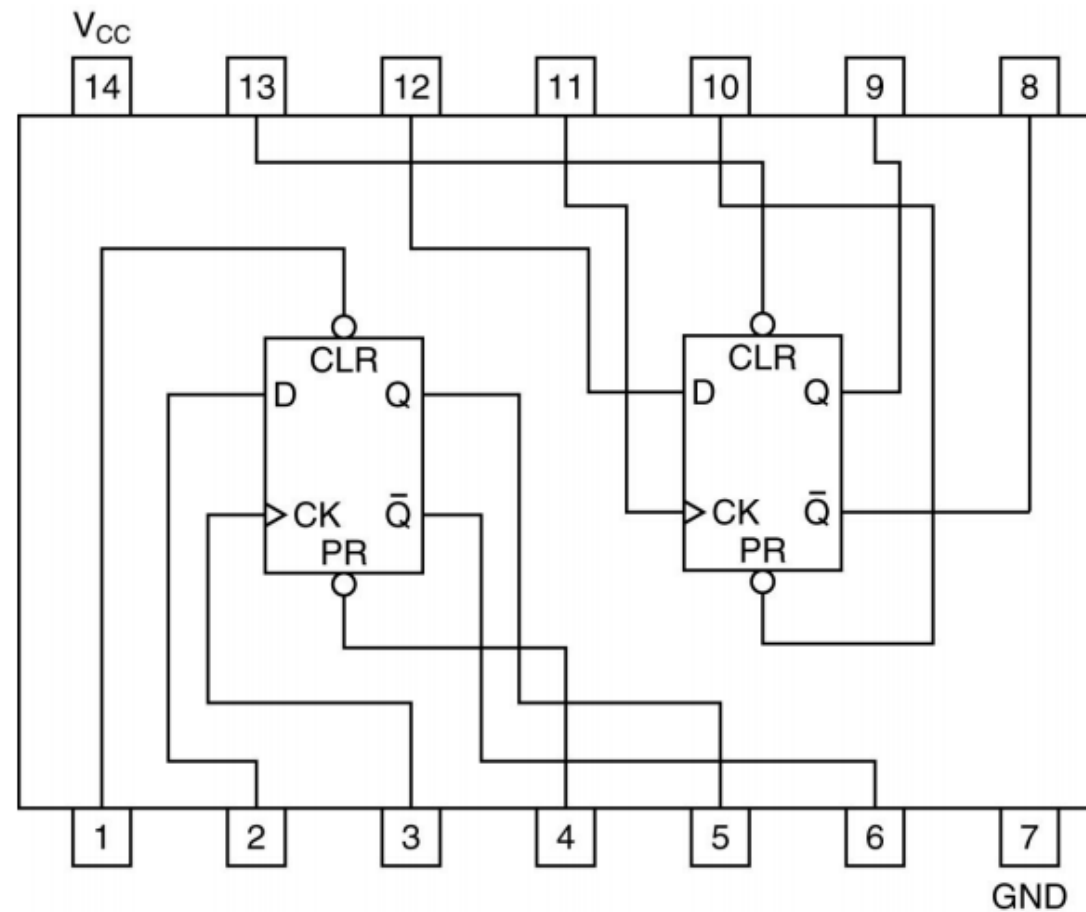


# REGISTRADORES



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Flip-flop do tipo D dual

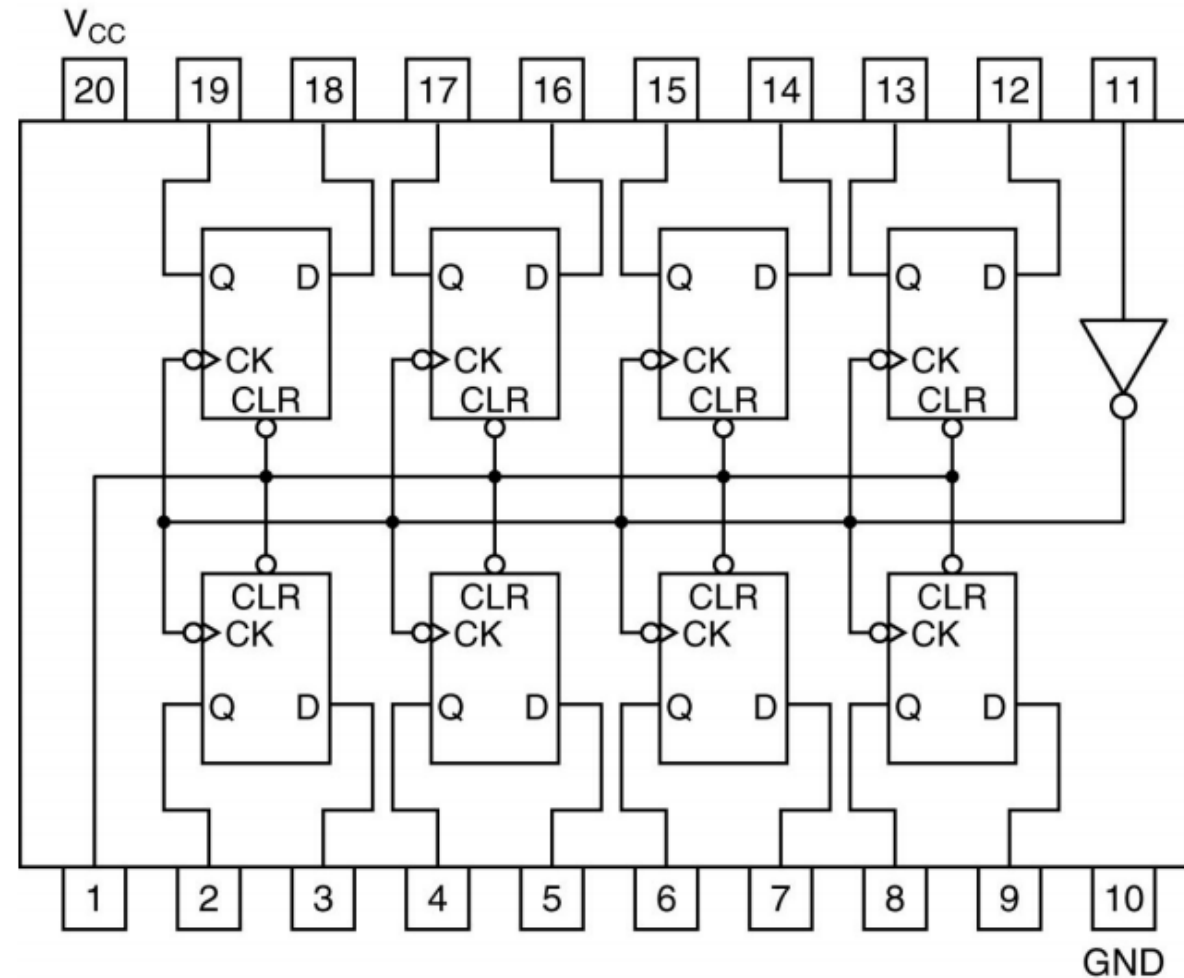


# REGISTRADORES



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Flip-flop do tipo D octal





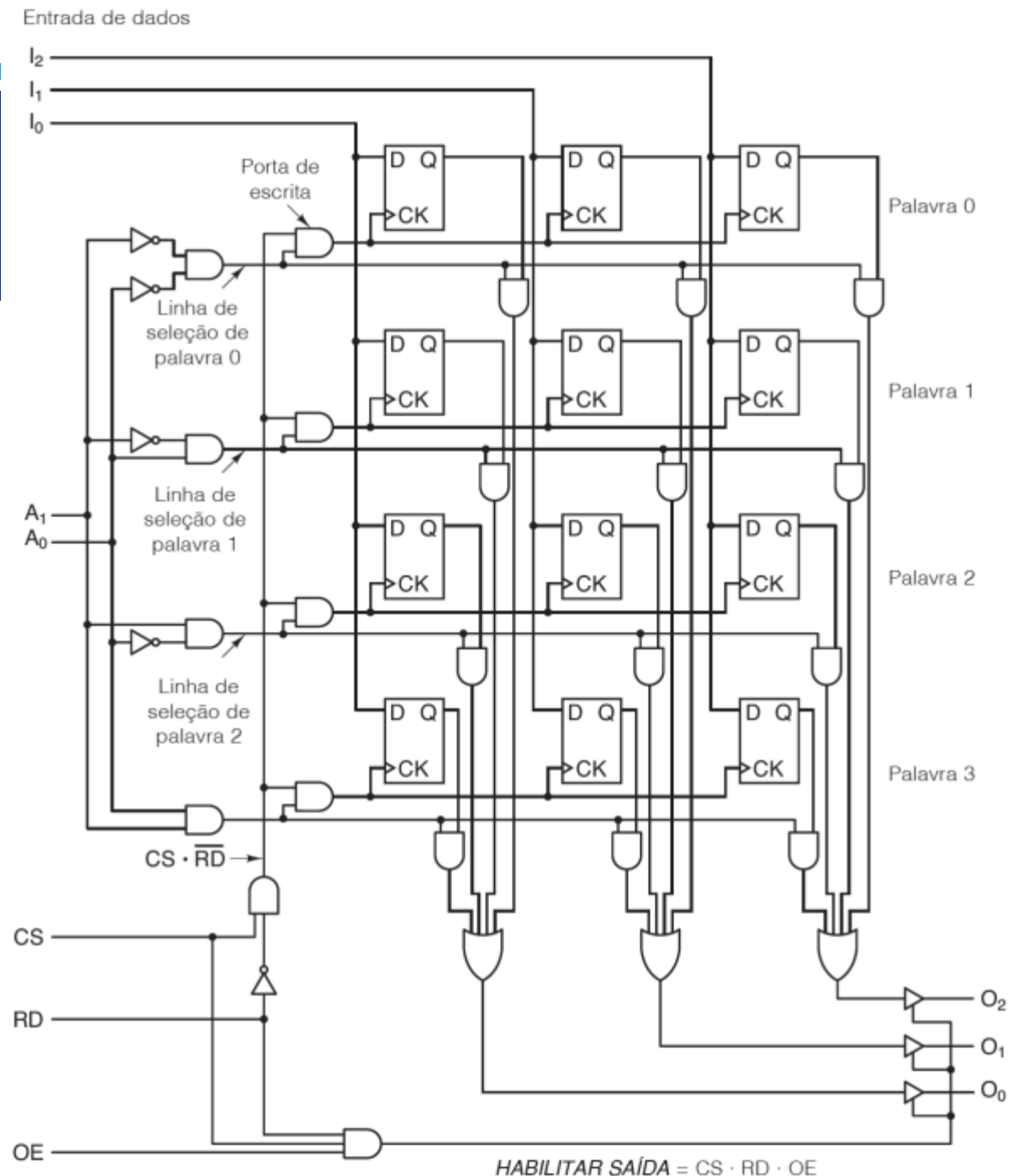
# CIRCUITOS LÓGICOS

ORGANIZAÇÃO DA MEMÓRIA



# ORGANIZAÇÃO DA MEMÓRIA

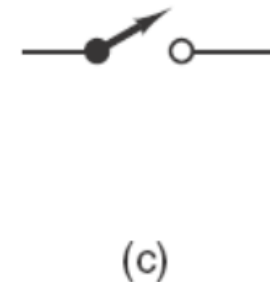
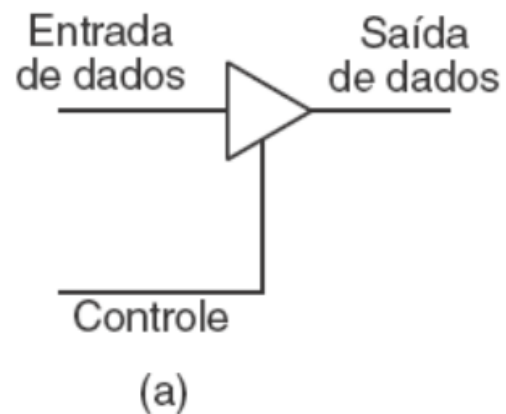
- A figura a seguir mostra um diagrama lógico para uma memória 4 x 3
  - Cada linha é uma das quatro palavras de 3 bits
  - Uma operação de leitura ou escrita sempre lê ou escreve uma palavra completa
  - Observe que o número de palavras é sempre uma potência de 2



# ORGANIZAÇÃO DA MEMÓRIA



UNIVERSIDADE  
FEDERAL DO CEARÁ



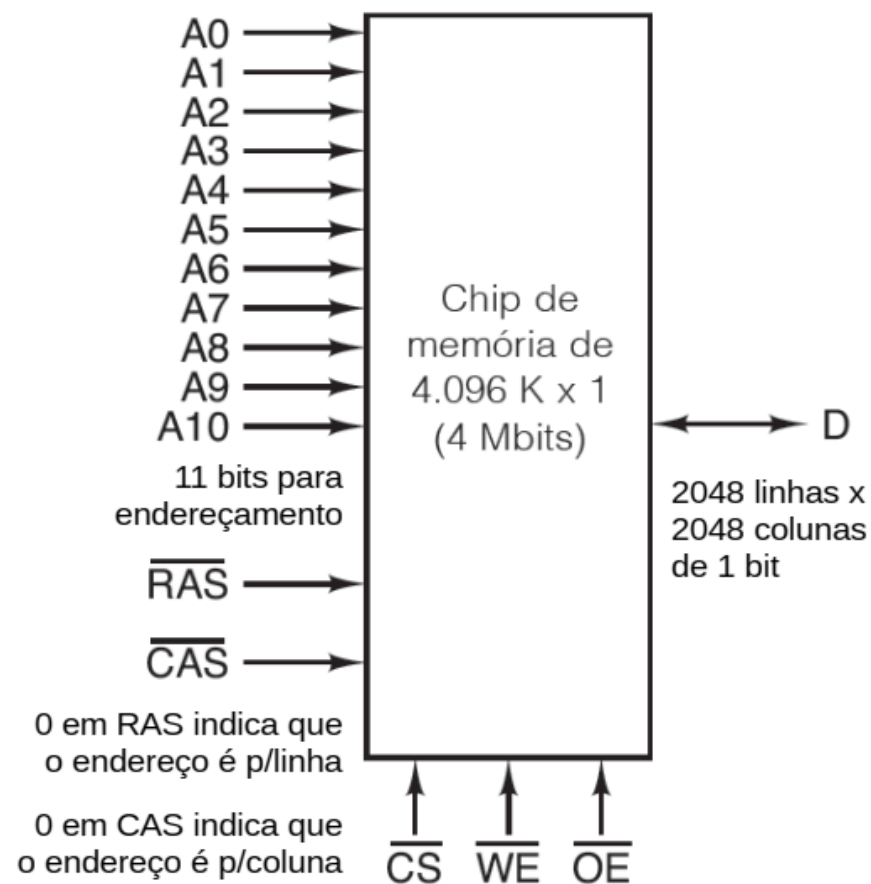
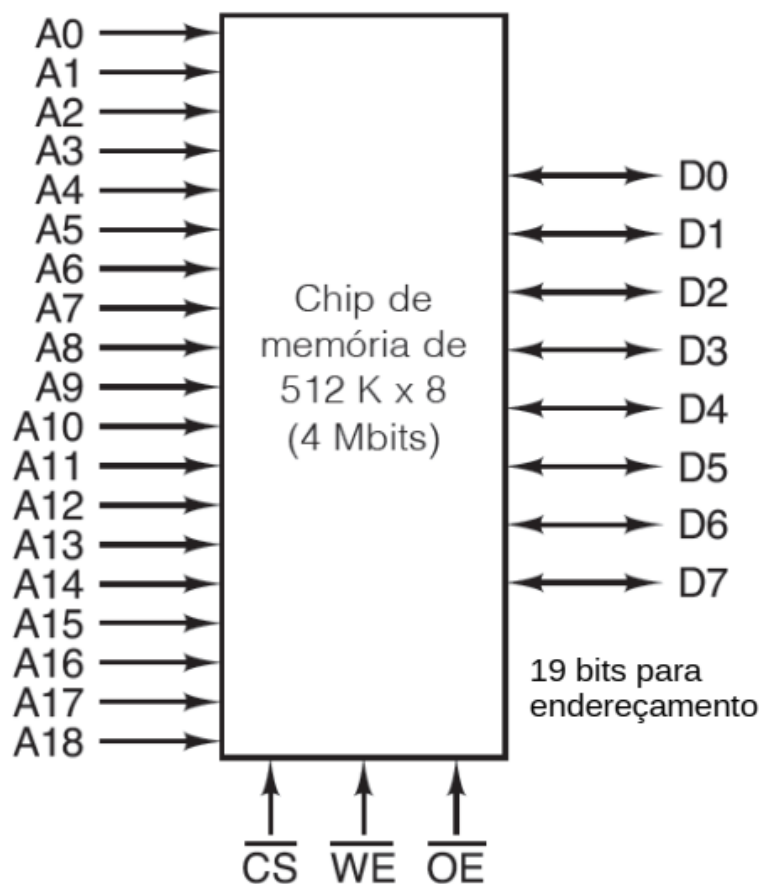
- a) **Buffer**, não inversor
- b) Efeito de (a) quando o controle está alto
- c) Efeito de (a) quando o controle está baixo

# CHIPS DE MEMÓRIA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Dois modos de organizar um chip de memória de 4 Mbits.

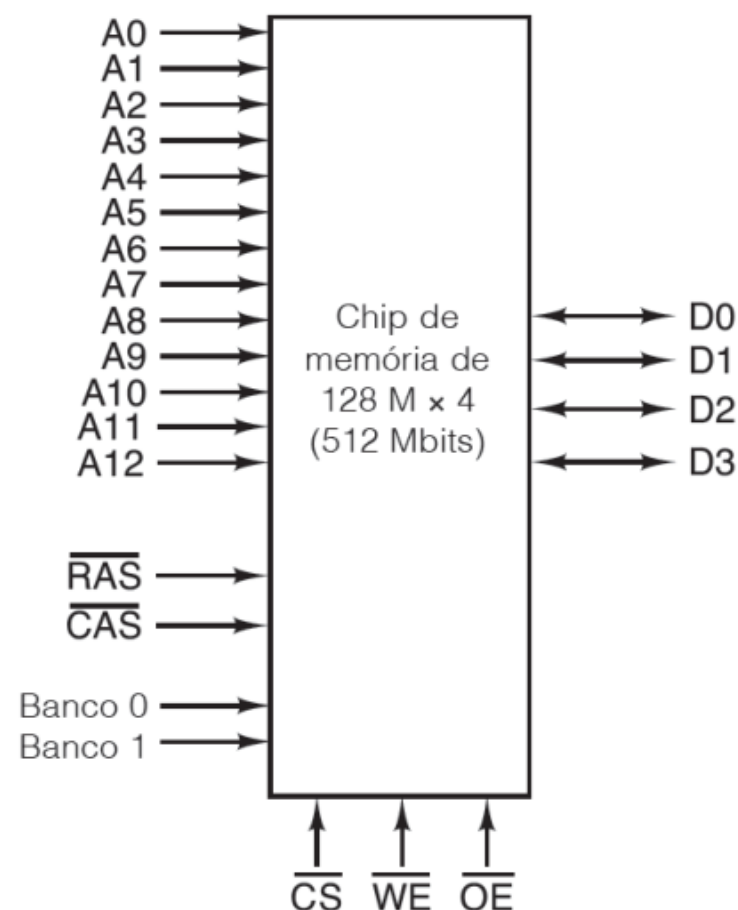
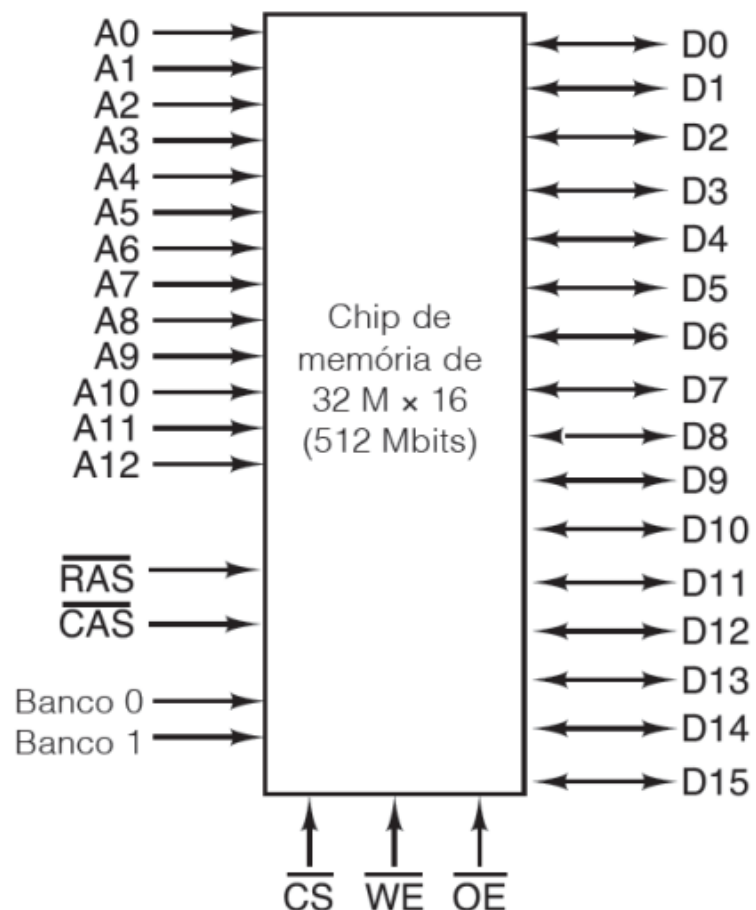


# CHIPS DE MEMÓRIA



UNIVERSIDADE  
FEDERAL DO CEARÁ

- Dois modos de organizar um chip de memória de 512 Mbits.





# REFERÊNCIAS



UNIVERSIDADE  
FEDERAL DO CEARÁ

- TANENBAUM, A. S. Organização Estruturada de Computadores. Editora LTC, 5 ed, Rio de Janeiro, 2007.
- STALLINGS, W. Arquitetura e Organização de Computadores. Editora Prentice Hall, 5 edição, 2002.