

# **ORGANIZAÇÃO E ARQUITETURA DE COMPUTADORES**

---

## **Circuitos Combinacionais**

### **Simplificação de Expressões Booleanas**

**Alexandre Amory  
Edson Moreno**

## Na Aula Anterior ...

---

- **Circuitos combinacionais**
- **Portas lógicas: NOT, AND, NAND, OR, NOR, XOR, MUX, HA, FA**
- **Tabela verdade e expressões booleanas**
- **ULA da Cleópatra**

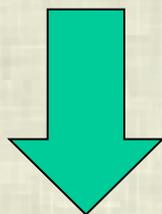
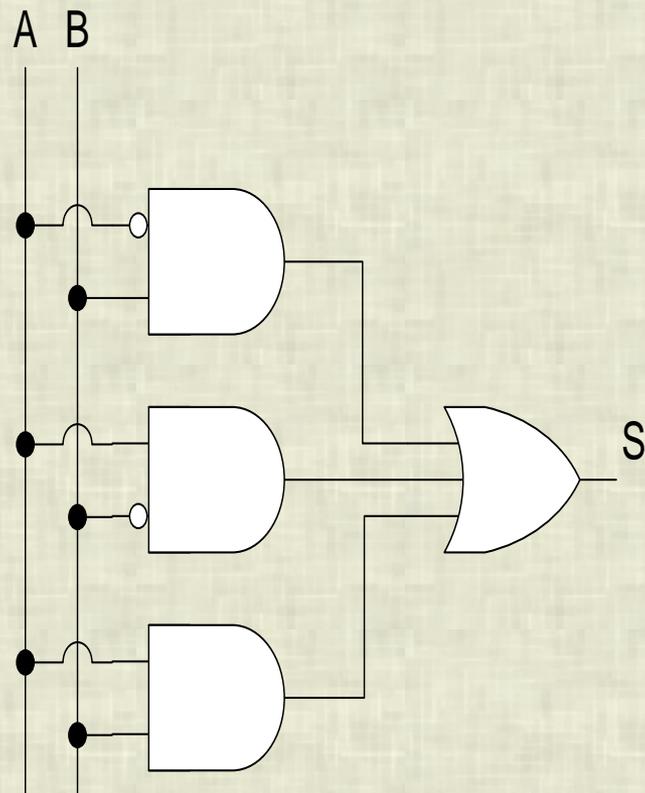
# Na Aula de Hoje ...

---

- **Equivalência lógica**
- **Minimização de lógica combinacional**
- **Mapas de Karnaugh**

# Da Tabela Verdade para o Circuito

Entradas		Saída	
A	B	Y	
0	0	0	
0	1	1	$\bar{A} \cdot B$
1	0	1	$A \cdot \bar{B}$
1	1	1	$A \cdot B$

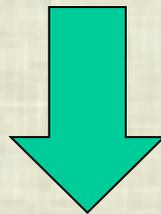
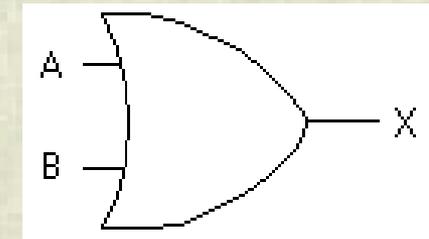


$$Y = \bar{A} \cdot B + A \cdot \bar{B} + A \cdot B$$

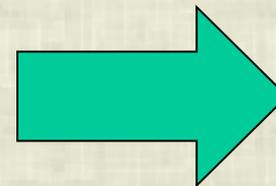


# Mas é Possível Fazer mais Otimizado

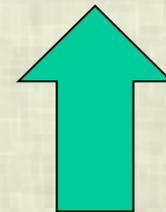
Entradas		Saída	
A	B	Y	
0	0	0	
0	1	1	$\bar{A} \cdot B$
1	0	1	$A \cdot \bar{B}$
1	1	1	$A \cdot B$



**Aplica-se técnica  
de minimização  
(e.g. Karnaugh /  
propriedades)**



$$X = A + B$$



# Fluxo Com ou Sem Minimização Lógica

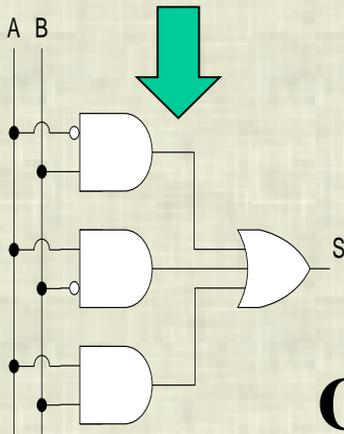
Entradas		Saída	
A	B	Y	
0	0	0	
0	1	1	$\bar{A} \cdot B$
1	0	1	$A \cdot \bar{B}$
1	1	1	$A \cdot B$

Fluxo com minimização

Aplica-se técnica de minimização (e.g. Karnaugh / propriedades)

Fluxo sem minimização

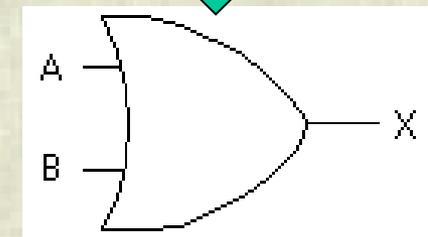
$$Y = \bar{A} \cdot B + A \cdot \bar{B} + A \cdot B$$



equivalente

equivalente

$$X = A + B$$



O que se ganha com a minimização lógica ?

# Propriedades da Álgebra Booleana

- **Postulados**

$A \cdot 0 = 0$	$A + 0 = A$	$A + 1 = 1$	$A \cdot 1 = A$
$A + \bar{A} = 1$	$A \cdot \bar{A} = 0$	$A + A = A$	$A \cdot A = A$

- **Propriedade Comutativa**

$A + B = B + A$	$A \cdot B = B \cdot A$
-----------------	-------------------------

- **Propriedade Associativa**

$(A + B) + C = A + (B + C)$	$(A \cdot B) \cdot C = (B \cdot C) \cdot A$
-----------------------------	---

- **Propriedade Distributiva**

$A \cdot (B + C) = A \cdot B + A \cdot C$
---

- **Teorema de De Morgan**

$\overline{A \cdot B \cdot C \dots} = \overline{A + B + C + \dots}$	$\overline{A + B + C + \dots} = \overline{A \cdot B \cdot C \dots}$
---	---

# Minimização Aplicando as Leis da Álgebra Booleana

---

- **Aplicando os postulados e leis da álgebra Booleana as funções Booleanas podem ser minimizadas**
  - O circuito equivalente pode ser menor
  - Variáveis de entrada podem ser eliminadas da função equivalente
- **Exemplos:**

$$\text{a) } S1 = X \cdot Y + X \cdot \bar{Y} \rightarrow X$$

$$\text{b) } S2 = X + X \cdot \bar{Y} \rightarrow X$$

$$\text{c) } S3 = \overline{(X + Y + \bar{Z}) \cdot (X + \bar{Y} + Z + \bar{W})} \cdot 0 \rightarrow 1$$

$$\text{d) } S4 = 1 + \overline{X \cdot \bar{Y} \cdot Z + W \cdot Z + Z \cdot Y} \rightarrow 1$$

$$\text{e) } S5 = X \cdot Y + \overline{(X + \bar{Y}) \cdot (\bar{X} + Y)} \rightarrow X + Y$$

$$\text{f) } S6 = \overline{(\bar{X} + \bar{Y} + \bar{Z}) \cdot (\bar{X} + \bar{Y} + Z) \cdot (\bar{X} + Y + \bar{Z}) \cdot (\bar{X} + Y + Z)} \rightarrow X$$

# Mapa de karnaugh

---

- **Método de auxílio à minimização de funções booleanas**
- **Mapeamento feito a partir da tabela verdade de um circuito**
- **Explora o agrupamento de termos equivalentes (e.g. '1')**
- **Permite alcançar visualmente as reduções do circuito base**

# Passos para a Minimização com mapa de karnaugh

---

1. Criar a tabela verdade da lógica
2. Escrever a expressão lógica não minimizada (opcional)
3. Desenhar o seu circuito equivalente (opcional)
4. Desenha o Mapa de Karnaugh
5. Preencher os '1's no Mapa
6. Identificar os '1's adjacentes
7. Escrever a expressão lógica minimizada
8. Desenhar o seu circuito equivalente (opcional)

# Mais um Exemplo: com 3 variáveis

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

- **Passo 1 (tab. verdade)**

A	B	C	Y	
0	0	0	0	
0	0	1	1	$\bar{A} \cdot \bar{B} \cdot C$
0	1	0	1	$\bar{A} \cdot B \cdot \bar{C}$
0	1	1	1	$\bar{A} \cdot B \cdot C$
1	0	0	0	
1	0	1	1	$A \cdot \bar{B} \cdot C$
1	1	0	0	
1	1	1	1	$A \cdot B \cdot C$

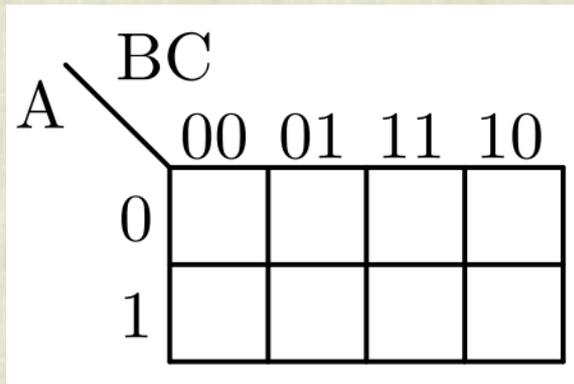
- **Passo 2 (exp. booleana não min.)**

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

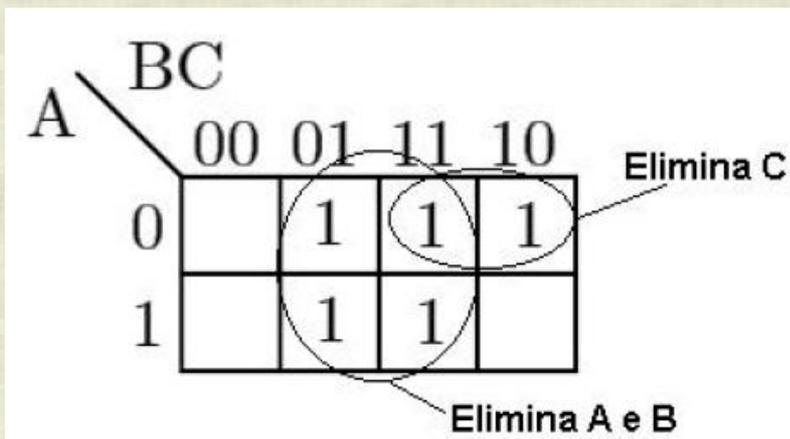
- **Passo 3 (desenhar circuito)**

## Mais um Exemplo (cont)

- Passo 4 (desenhar mapa)



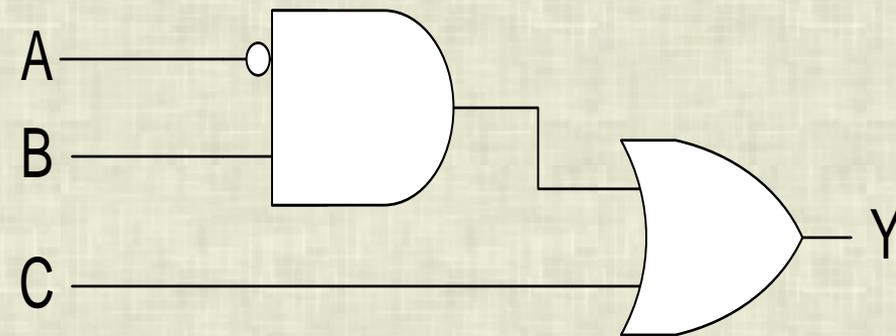
- Passos 5 e 6 (colocar '1's)



- Passo 7 (exp. booleana min)

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

- Passo 8 (desenhar circuito)



# Exemplos de Agrupamento de Quadros: 3 Variáveis

	$\bar{C}$	$C$
$\bar{A}\bar{B}$	0	0
$\bar{A}B$	1	0
$AB$	1	0
$A\bar{B}$	0	0

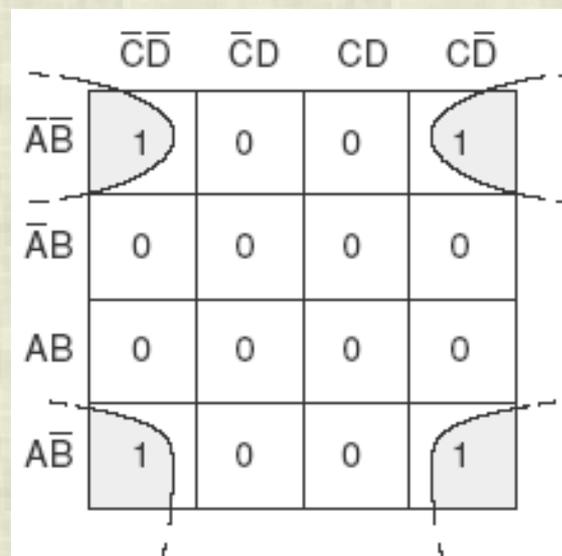
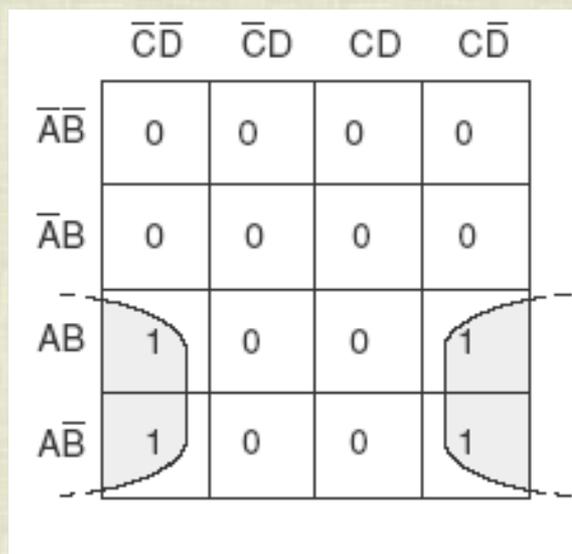
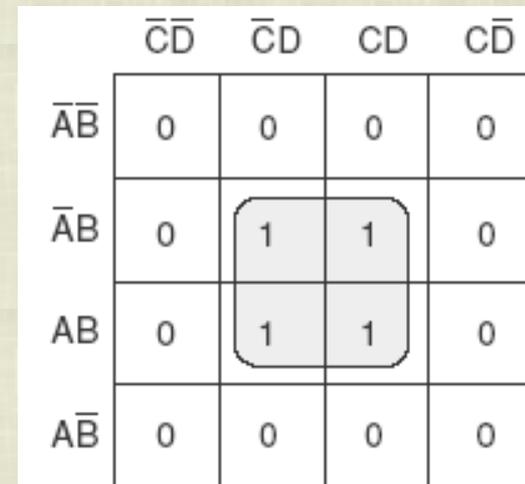
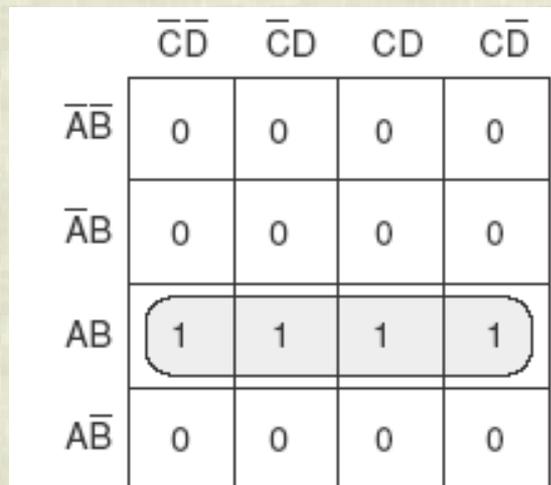
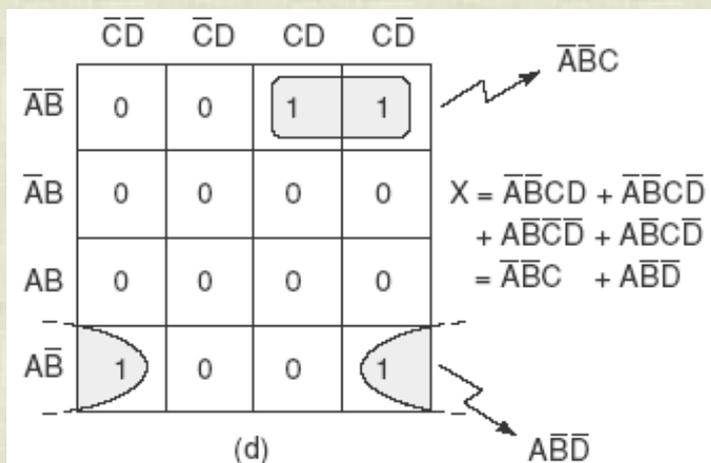
	$\bar{C}$	$C$
$\bar{A}\bar{B}$	0	0
$\bar{A}B$	1	1
$AB$	0	0
$A\bar{B}$	0	0

	$\bar{C}$	$C$
$\bar{A}\bar{B}$	1	0
$\bar{A}B$	0	0
$AB$	0	0
$A\bar{B}$	1	0

	$\bar{C}$	$C$
$\bar{A}\bar{B}$	0	1
$\bar{A}B$	0	1
$AB$	0	1
$A\bar{B}$	0	1

$$X = C$$

# Exemplos de Agrupamento de Quadros: 4 Variáveis



# Exemplos de Agrupamento de Quadros: 4 Variáveis

	$\bar{C}\bar{D}$	$\bar{C}D$	$CD$	$C\bar{D}$
$\bar{A}\bar{B}$	0	0	0	0
$\bar{A}B$	1	1	1	1
$AB$	1	1	1	1
$A\bar{B}$	0	0	0	0

	$\bar{C}\bar{D}$	$\bar{C}D$	$CD$	$C\bar{D}$
$\bar{A}\bar{B}$	1	1	0	0
$\bar{A}B$	1	1	0	0
$AB$	1	1	0	0
$A\bar{B}$	1	1	0	0

	$\bar{C}\bar{D}$	$\bar{C}D$	$CD$	$C\bar{D}$
$\bar{A}\bar{B}$	1	1	1	1
$\bar{A}B$	0	0	0	0
$AB$	0	0	0	0
$A\bar{B}$	1	1	1	1

	$\bar{C}\bar{D}$	$\bar{C}D$	$CD$	$C\bar{D}$
$\bar{A}\bar{B}$	1	0	0	1
$\bar{A}B$	1	0	0	1
$AB$	1	0	0	1
$A\bar{B}$	1	0	0	1

# Relembrar é viver

- **Funcionamento do Multiplexador**

- Seleciona a porta enviada para a saída

- Se sel = '0', então S=B
- Se sel = '1', então S=A

- **Tabela Verdade**

A	B	sel	S
0	0	0	0
0	0	1	0
0	1	0	1 $\rightarrow (\bar{A}.B.\bar{Sel}) +$
0	1	1	0
1	0	0	0
1	0	1	1 $\rightarrow (A.\bar{B}.Sel) +$
1	1	0	1 $\rightarrow (A.B.\bar{Sel}) +$
1	1	1	1 $\rightarrow (A.B.Sel)$

- **Resultado a ser alcançado:**

- $(A.Sel) + (B.\bar{Sel})$

- **Método a ser aplicado:**

- Mapa de karnaugh

# Resumo

---

- **Vimos que é possível descrever uma mesma lógica (equivalente) de várias formas**
  - Algumas serão não mínimas
- **Vimos como utilizar Mapas de Karnaugh para efetuar a minimização lógica**
- **O impacto da minimização lógica é a redução de área de hardware e redução do atraso do circuito**