



## **Aula 01 – Introdução**



## Tópicos da aula

- **Objetivos da disciplina**
- **Metodologia de projeto – pequenos sistemas digitais**
- **Metodologia de projeto – grandes sistemas digitais**
- **Dispositivos lógicos programáveis**



## Introdução

- ❑ Neste curso será ensinada uma linguagem de síntese de hardware (VHDL)
- ❑ A cada semana serão apresentados novos conceitos da linguagem e, em seguida, serão realizados experimentos de laboratório, utilizando ferramenta de síntese
- ❑ A avaliação do desempenho na disciplinas será através de uma prova escrita (P1) e um trabalho final (T1)



Universidade Federal  
de Santa Catarina

## Metodologia de projeto Pequenos Sistemas Digitais

- ❑ Normalmente são implementados com componentes eletrônicos discretos – pequenas unidades com funções lógicas fixas
- ❑ Esta metodologia é útil para análise de partes de grandes sistemas digitais



Universidade Federal  
de Santa Catarina

## Exemplo de projeto de um pequeno sistema digital

Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1



## Exemplo de projeto de um pequeno sistema digital

Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

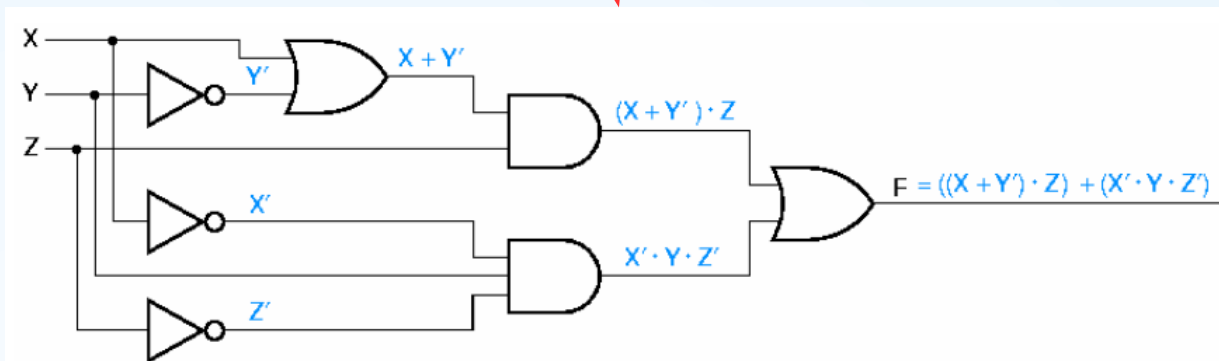

$$F = ((X + Y') \cdot Z) + (X' \cdot Y \cdot Z')$$



## Exemplo de projeto de um pequeno sistema digital

Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

$$F = ((X+Y') \cdot Z) + (X' \cdot Y \cdot Z')$$

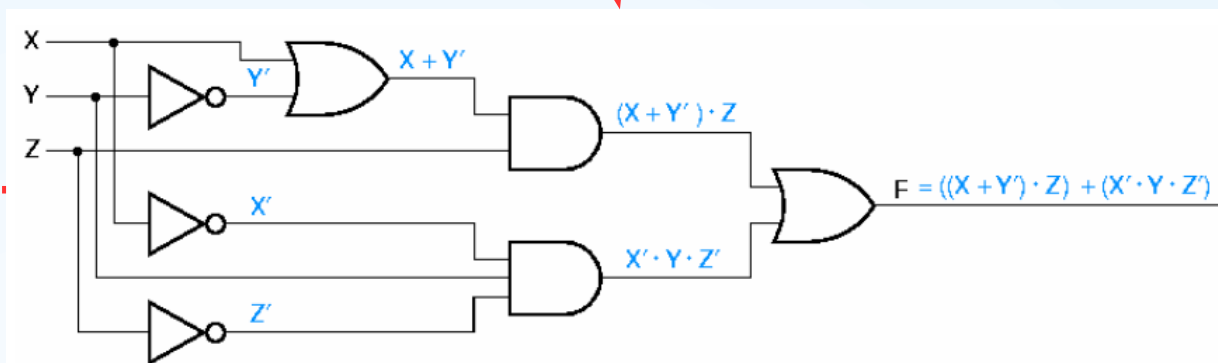




## Exemplo de projeto de um pequeno sistema digital

Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

$$F = ((X+Y') \cdot Z) + (X' \cdot Y \cdot Z')$$



XY		X			
		00	01	11	10
Z	0	0	2	6	4
	1	1	3	7	5

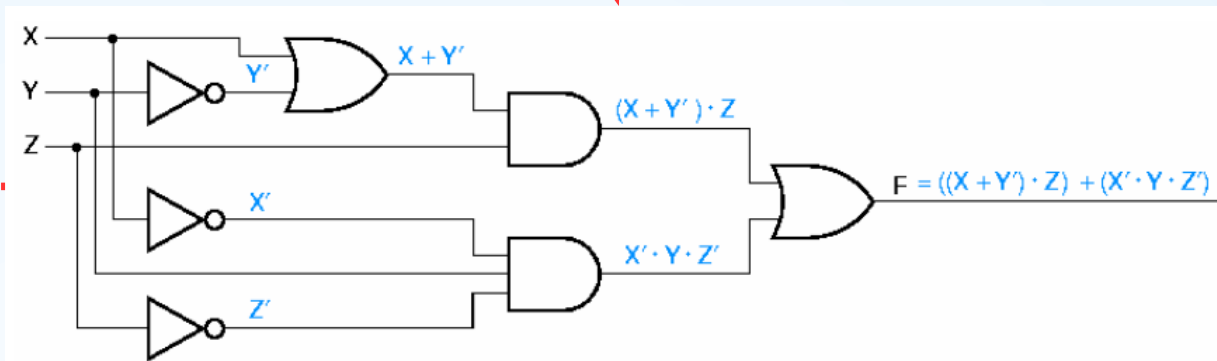




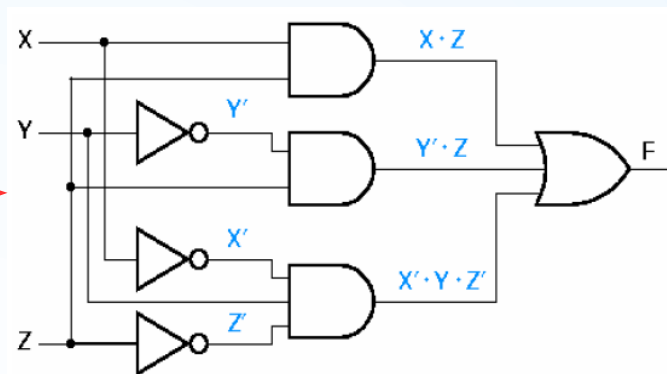
## Exemplo de projeto de um pequeno sistema digital

Row	X	Y	Z	F
0	0	0	0	0
1	0	0	1	1
2	0	1	0	1
3	0	1	1	0
4	1	0	0	0
5	1	0	1	1
6	1	1	0	0
7	1	1	1	1

$$F = ((X+Y') \cdot Z) + (X' \cdot Y \cdot Z')$$



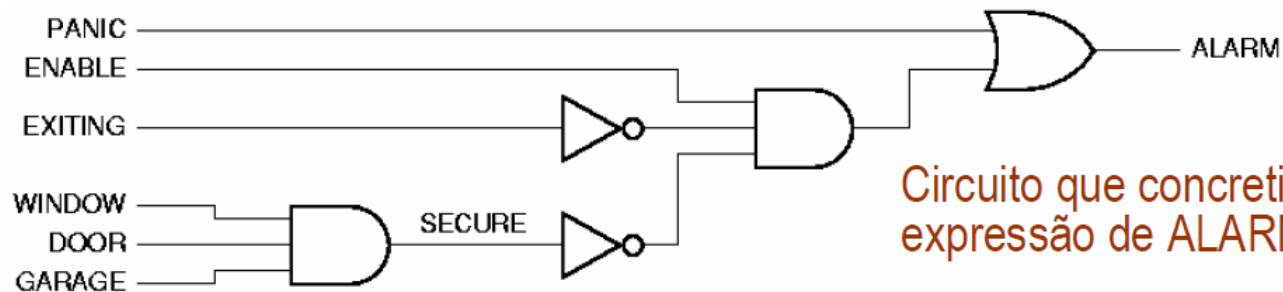
XY		X			
		00	01	11	10
Z	0	0	2	6	4
	1	1	3	7	5





## Exemplo: lógica de controle de um sistema de alarme

$ALARM = PANIC + ENABLE \cdot EXITING' \cdot SECURE'$   
 $SECURE = WINDOW \cdot DOOR \cdot GARAGE$



Circuito que concretiza a expressão de ALARM

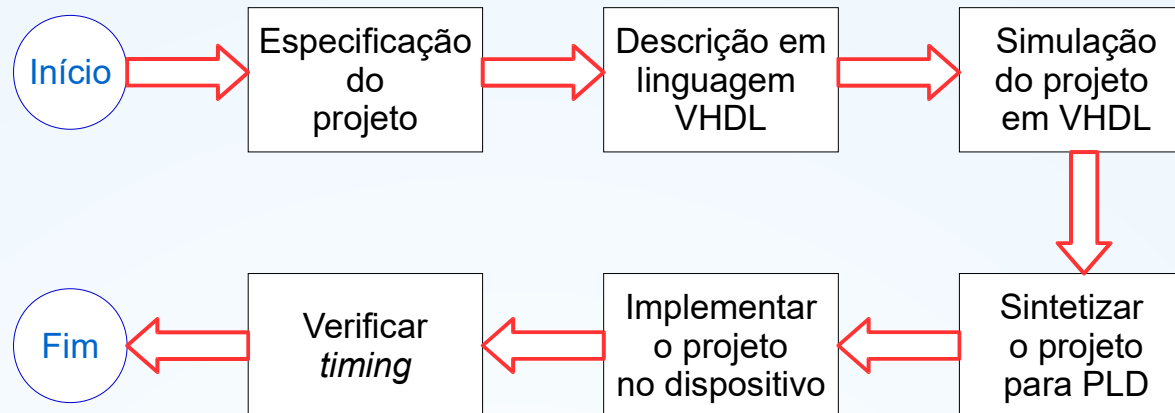


## Metodologia de projeto Grandes Sistemas Digitais

- ❏ Pequenos sistemas digitais normalmente são implementados em lógica discreta
- ❏ Sistemas digitais de maior porte são fisicamente inviáveis de serem feitos somente com lógica discreta
- ❏ Para grandes sistemas digitais o projeto é implementado em dispositivos lógicos programáveis
  - ❏ Menor área com alta densidade de blocos lógicos
  - ❏ Evita problemas com aquecimento
  - ❏ Evita problema com estabilidade de sinais



## Metodologia de projeto Grandes Sistemas Digitais

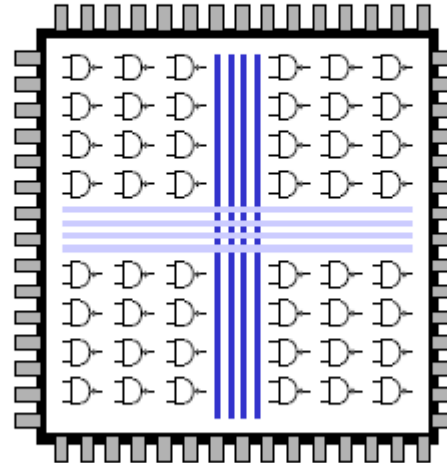




## O que é um PLD?

### ☐ Um PLD é um Dispositivo Lógico Programável

- ☐ Um arranjo configurável de portas lógicas que pode ser “programado” para implementar uma função lógica qualquer

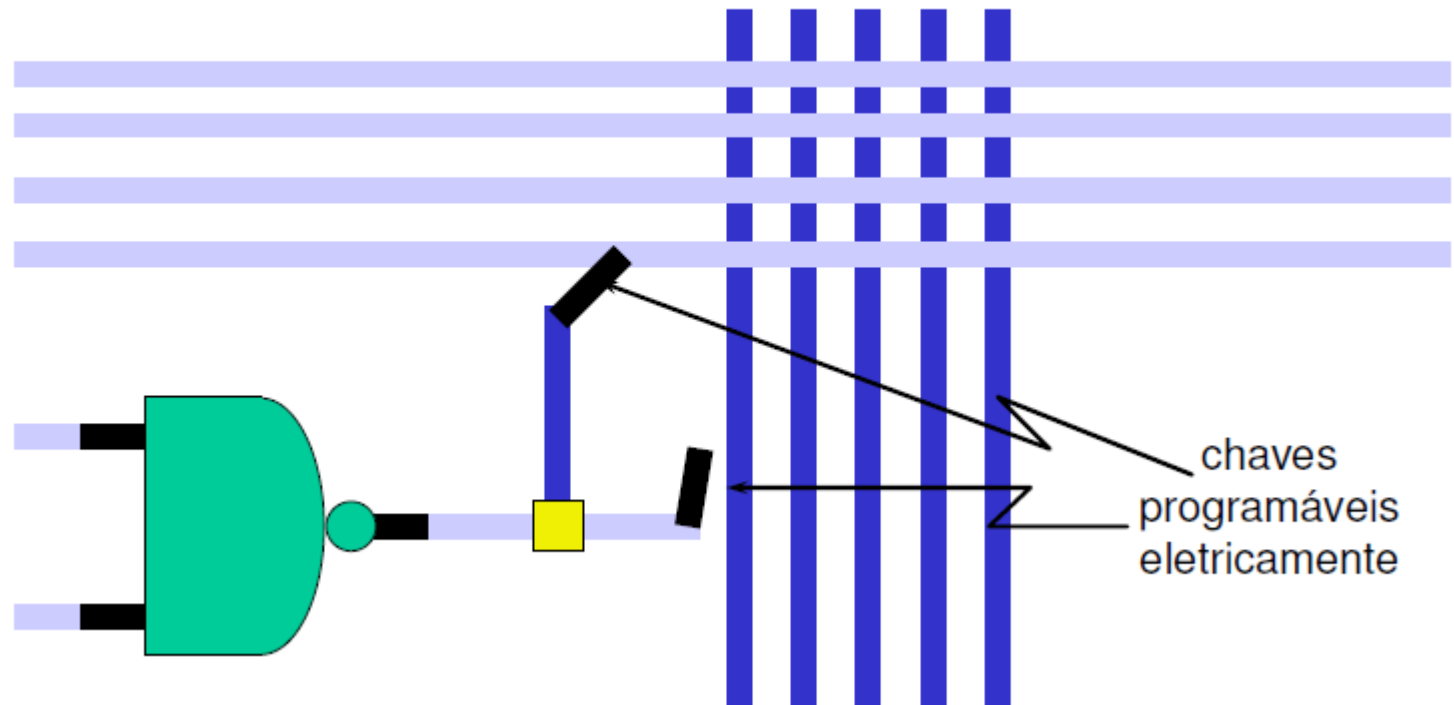


Arranjo de portas lógicas com interconexão  
programável eletricamente



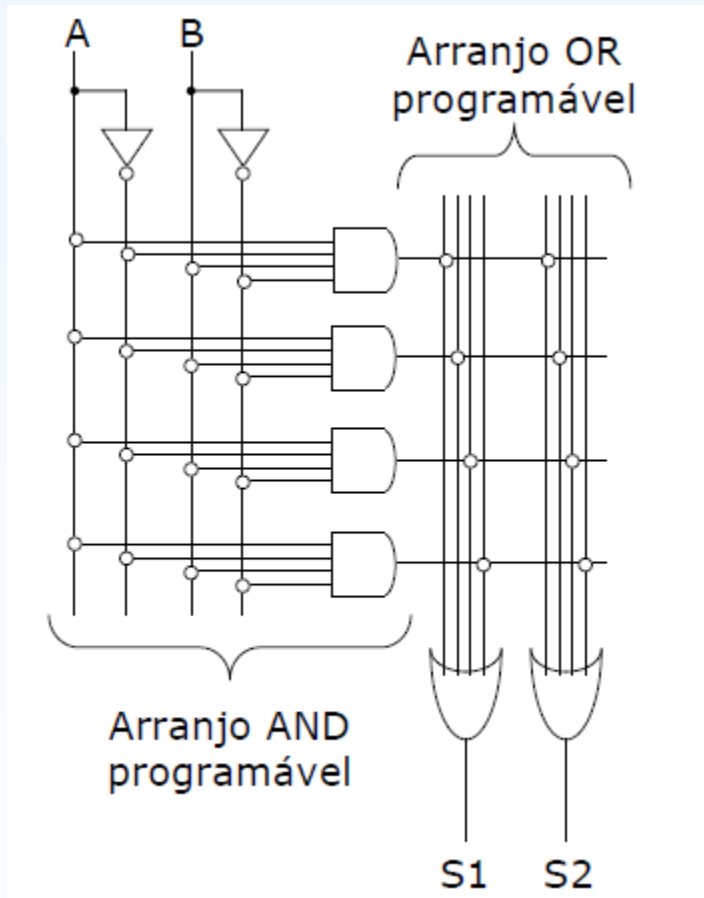
## O que é um PLD?

### ☐ Interconexão programada através de chaves



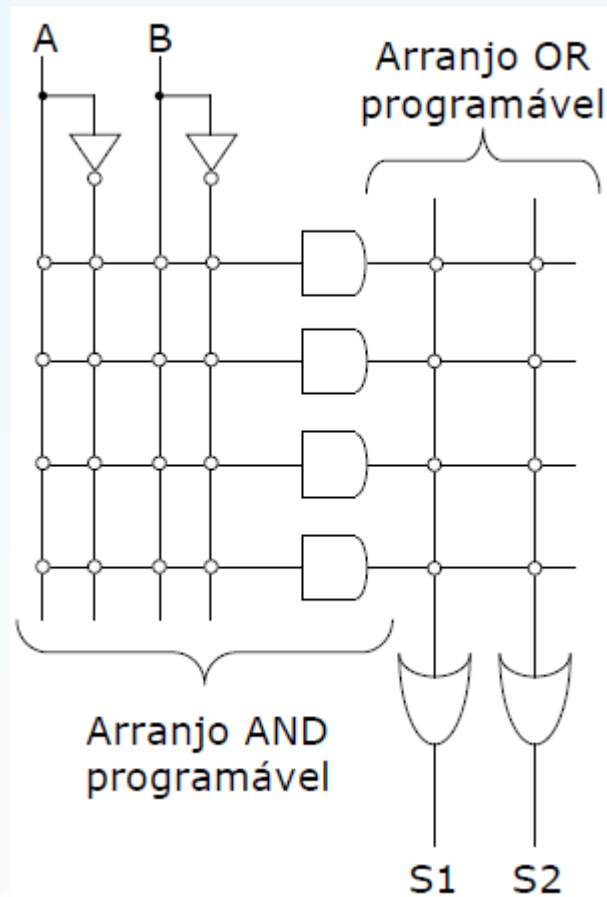


## Um PLD simples





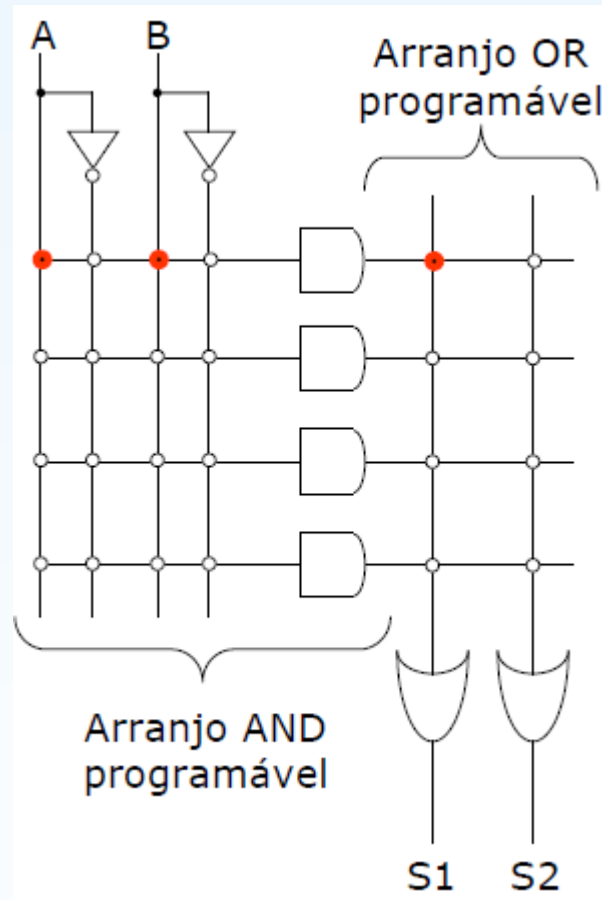
## Um PLD simples





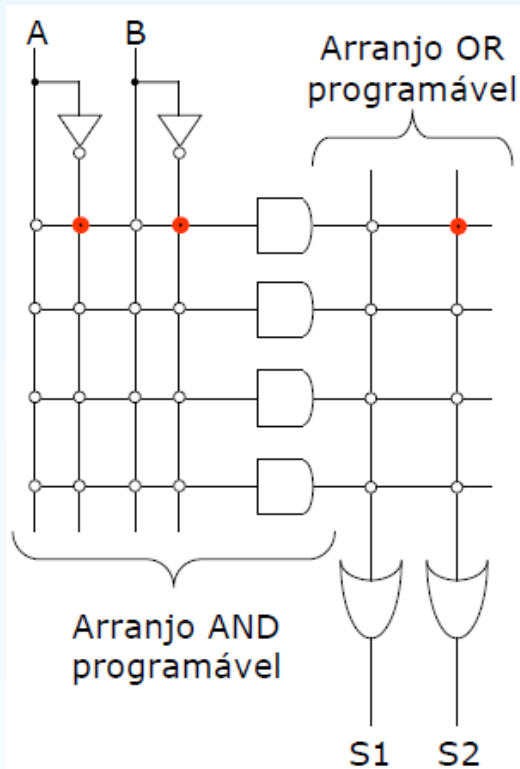


$$S1 = A.B$$



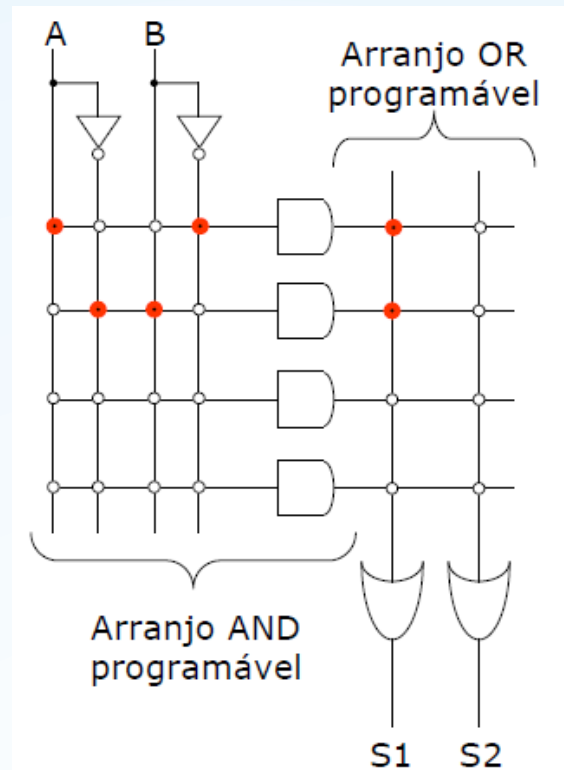


$$S1 = A'.B'$$





$$S1 = AB' + A'B$$



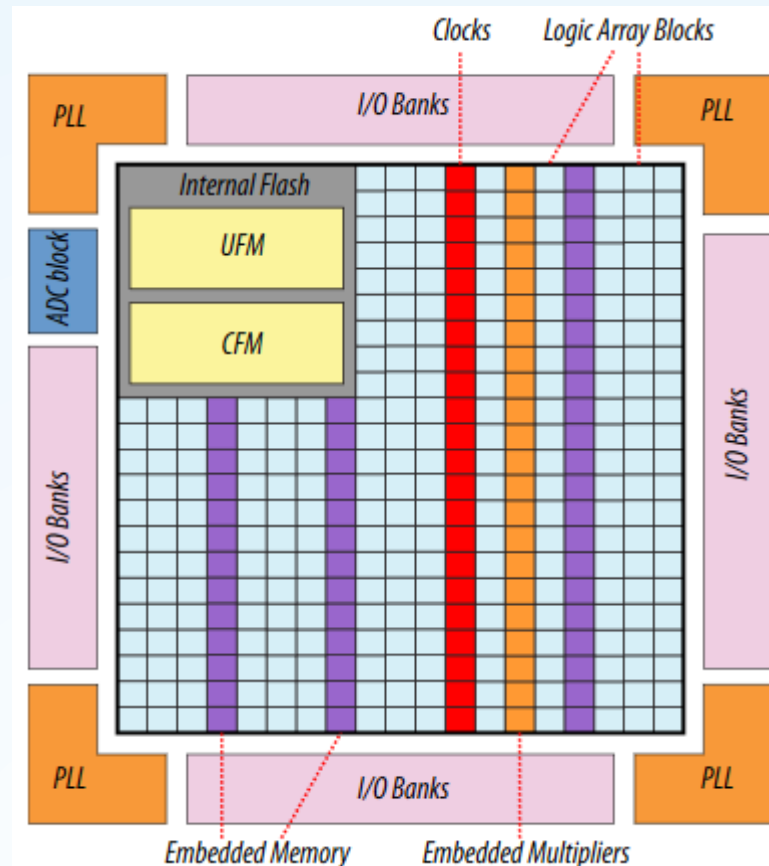


## CPLD – Complex PLD

- ☐ São PLDs de alta densidade lógica
- ☐ Um arranjo de múltiplos PLDs agrupados em blocos
- ☐ Outros nomes
  - ☐ EPLD - Enhanced PLD
  - ☐ superPAL
  - ☐ megaPAL

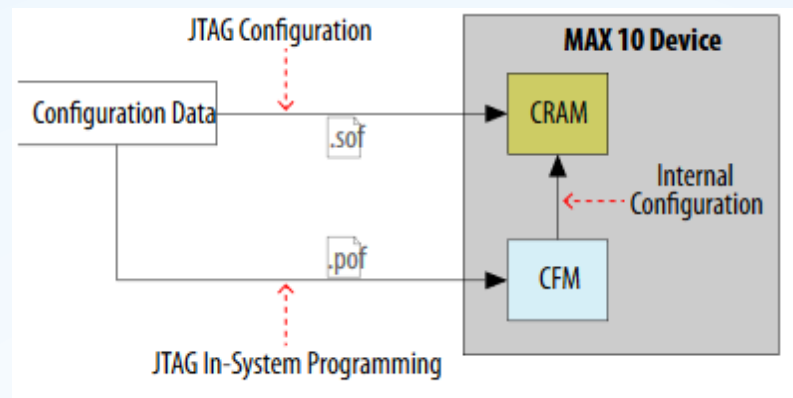


## CPLD MAX<sup>®</sup> 10 (Altera)





## CPLD MAX<sup>®</sup> 10 Modo de configuração





## FPGA – Field Programmable Gate Array

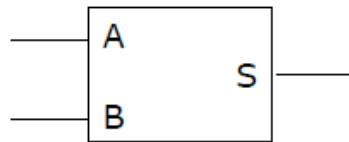
- ❑ **Densidade lógica maior que dos CPLDs**
- ❑ **Arquitetura baseada em LUTs – Look Up Tables**
  - ❑ Não em arranjos de portas
- ❑ **Integram muito mais flip-flops por porta lógica do que os CPLDs**



## LUT – Look Up Table

- ❑ Uma LUT de N entradas implementa qualquer função lógica com N variáveis

- ❑ Exemplo



Exemplos de Funções implementadas:

$S = \text{not } A$

$S = \text{not } B$

$S = A \text{ and } B$

$S = A \text{ or } B$

$S = A \text{ nand } B$

$S = A \text{ nor } B$

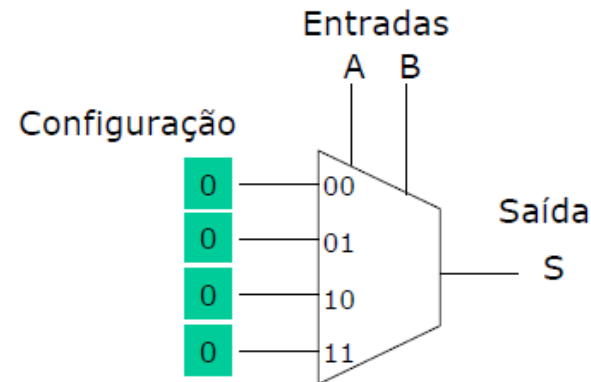
...





## LUT – Look Up Table

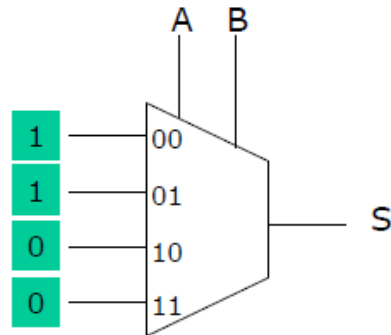
- ❑ É implementada por um multiplexador com  $2^N$  entradas



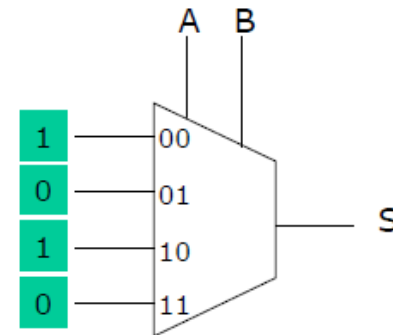


## LUT – Look Up Table

### Exemplos



**S = not A**

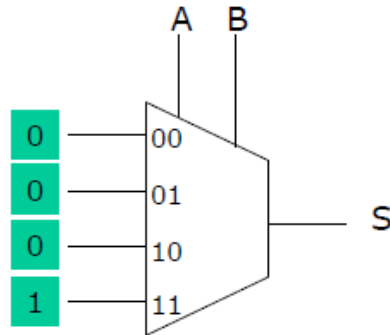


**S = not B**

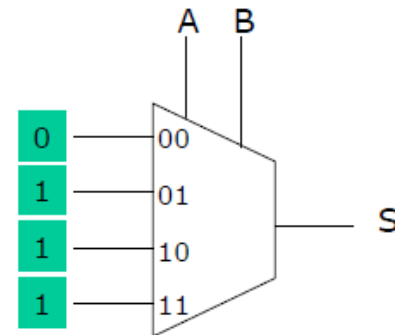


## LUT – Look Up Table

### Exemplos



$$S = A \text{ and } B$$

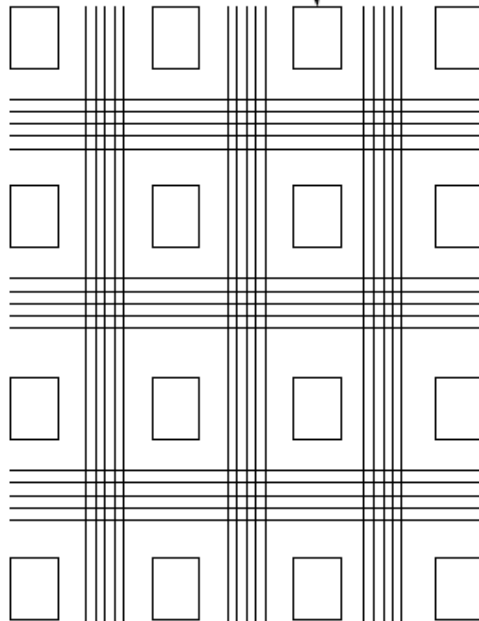


$$S = A \text{ or } B$$



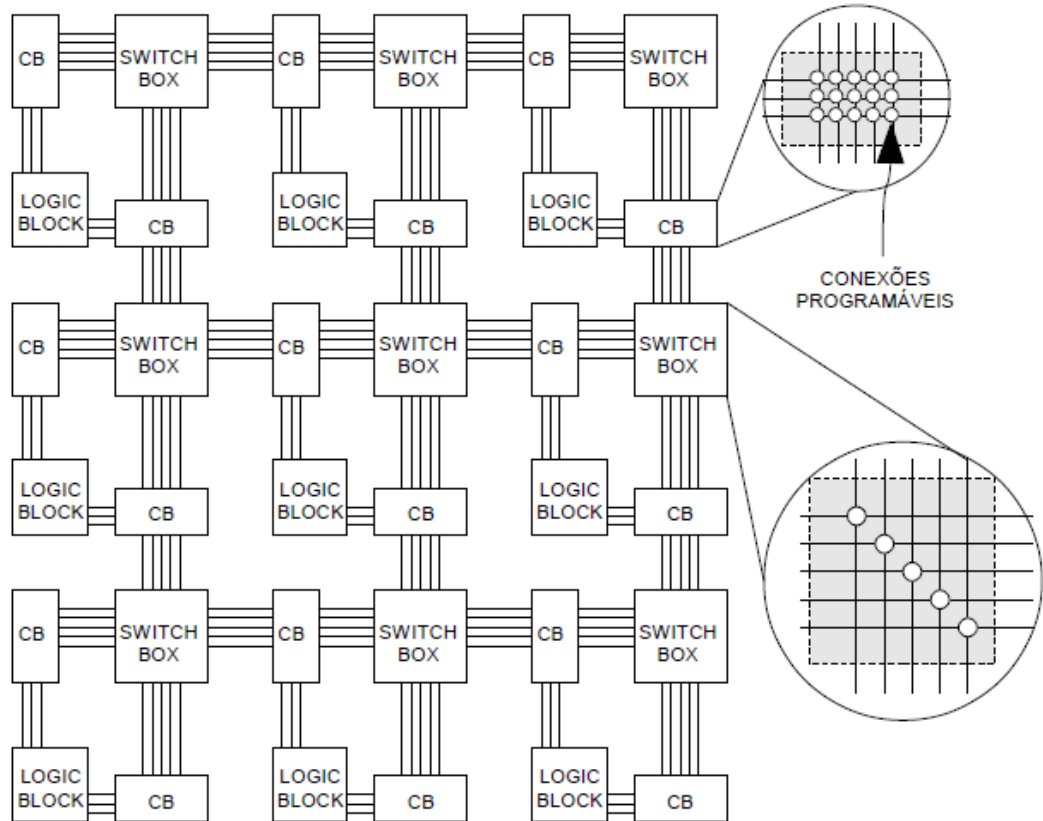
## Estrutura interna de um FPGA

BLOCOS LÓGICOS



INTERCONEXÕES

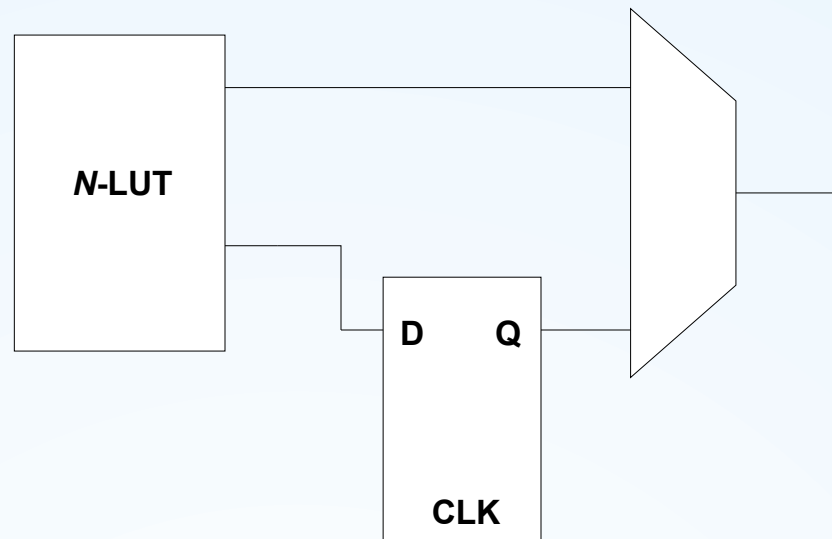
(a)



(b)



## Estrutura genérica de um Bloco Lógico

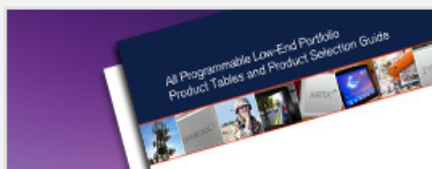






## Famílias de FPGA - Xilinx

### Product Tables and Product Selection Guides



#### Low-End Portfolio

Spartan-6	Artix-7	Zynq
-----------	---------	------



#### 7 Series

Artix-7	Kintex-7	Virtex-7
---------	----------	----------



#### UltraScale

Kintex UltraScale	Virtex UltraScale
-------------------	-------------------



#### UltraScale+

Kintex UltraScale+	Virtex UltraScale+
--------------------	--------------------



Universidade Federal  
de Santa Catarina

## Famílias de FPGA - Altera

### High-End FPGAs



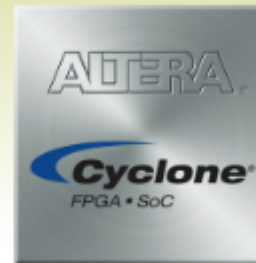
- Highest bandwidth, highest density
- Integrated transceiver variants
- Design entire systems on a chip

### Midrange FPGAs



- Balanced cost, power, and performance
- Integrated transceiver and processor variants
- Comprehensive design protection

### Lowest Cost and Power FPGAs



- Lowest system cost and power
- Integrated transceiver and processor variants
- Fastest time to market

### Non-Volatile FPGAs and Low-Cost CPLDs



- Instant-on, non-volatile solution
- Single-chip, dual-configuration non-volatile FPGA
- Low-cost, low-power CPLDs





## Ferramentas de desenvolvimento

### ☐ Quartus II

### ☐ Características básicas

- ☐ Entrada: esquemático ou HDL
- ☐ Síntese
- ☐ Simulação
- ☐ Análise de timing
- ☐ Floorplaning
- ☐ Configuração/Programação de PLD



## **FIM AULA I**