



Universidade Federal de Santa Catarina
Centro Tecnológico
Departamento de Informática e Estatística
Curso de Graduação em Ciências da Computação



Sistemas Digitais

INE 5406

Aula 7-T

4. Máquinas Seqüências Síncronas: Síntese de circuitos seqüenciais. Exemplos. Mapeamento e alternativas de implementação de máquinas de estado: "hardwired", PLA, ROM e PLD.

Prof. José Luís Güntzel
guntzel@inf.ufsc.br

www.inf.ufsc.br/~guntzel/ine5406/ine5406.html

4. Máquinas Seqüenciais Síncronas

▶ Alternativas de Implementação de FSMs

Registrador de Estados:

- **Tipos de registradores:**

- Feito com FFDs ou com FFJKs ou com FFTs
- Pode ser um registrador-deslocador
- Pode ser um registrador-contador

- **Quanto à forma de fabricação:**

- Registradores podem estar prontos, integrados em chips com 4 ou 8 bits, cascadeáveis (componentes MSI CMOS ou TTL)
- Registradores podem fazer uso de flip-flops pre-existentes dentro de um componente programável tipo SPLD, CPLD ou FPGA.
- Registradores podem ser especificados para serem fabricados do zero (opção de fabricação com máscaras ou *masked*)

4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs

Lógica de Próximo Estado e Lógica de Saída:

- Tipos de implementações:

hardwired

- Implementando as equações por meio de um circuito combinacional (“lógica aleatória”)
- Implementando as equações pela configuração de planos “E” e “OU” (PALs e PLAs)

Micropro-
gramação

- Gravando a tabela-verdade em bloco de memória (ROM, EPROM, EEPROM ou RAM)

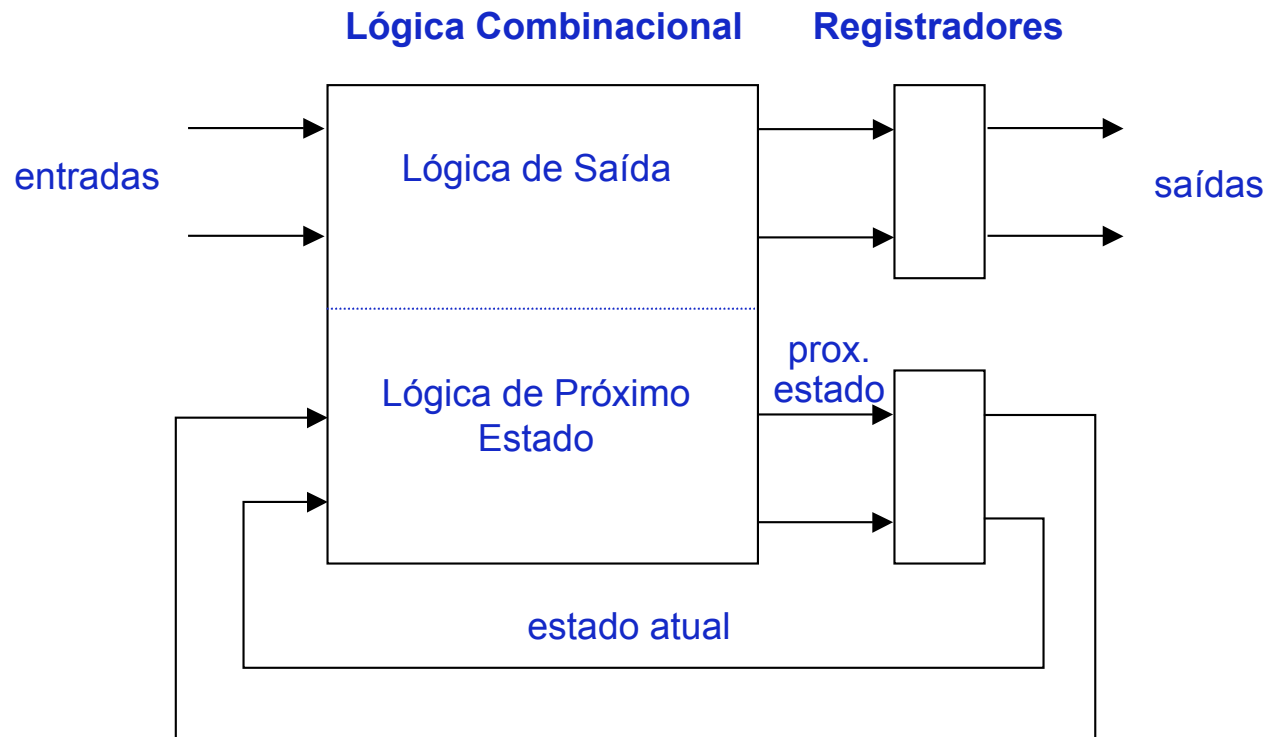
- Quanto à forma de fabricação:

- Usando chips de memória ROM, EPROM ou EEPROM
- Usando chips programáveis SPLDs: PLAs ou PALs
- Usando chips programáveis CPLDs ou FPGAs
- Mandando fabricar um chip do zero (*masked*)

4. Máquinas Seqüenciais Síncronas

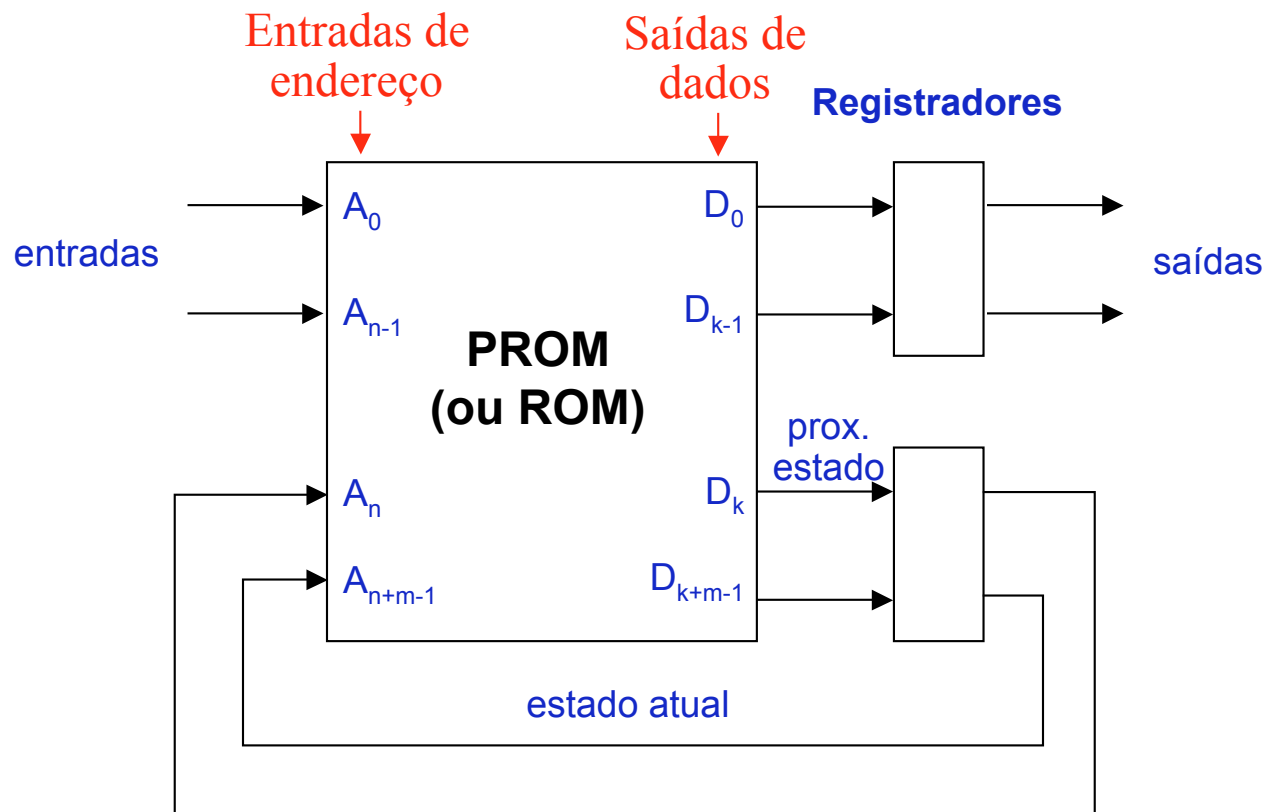
► Alternativas de Implementação de FSMs

Lógica de Próximo Estado e Lógica de Saída



4. Máquinas Seqüenciais Síncronas

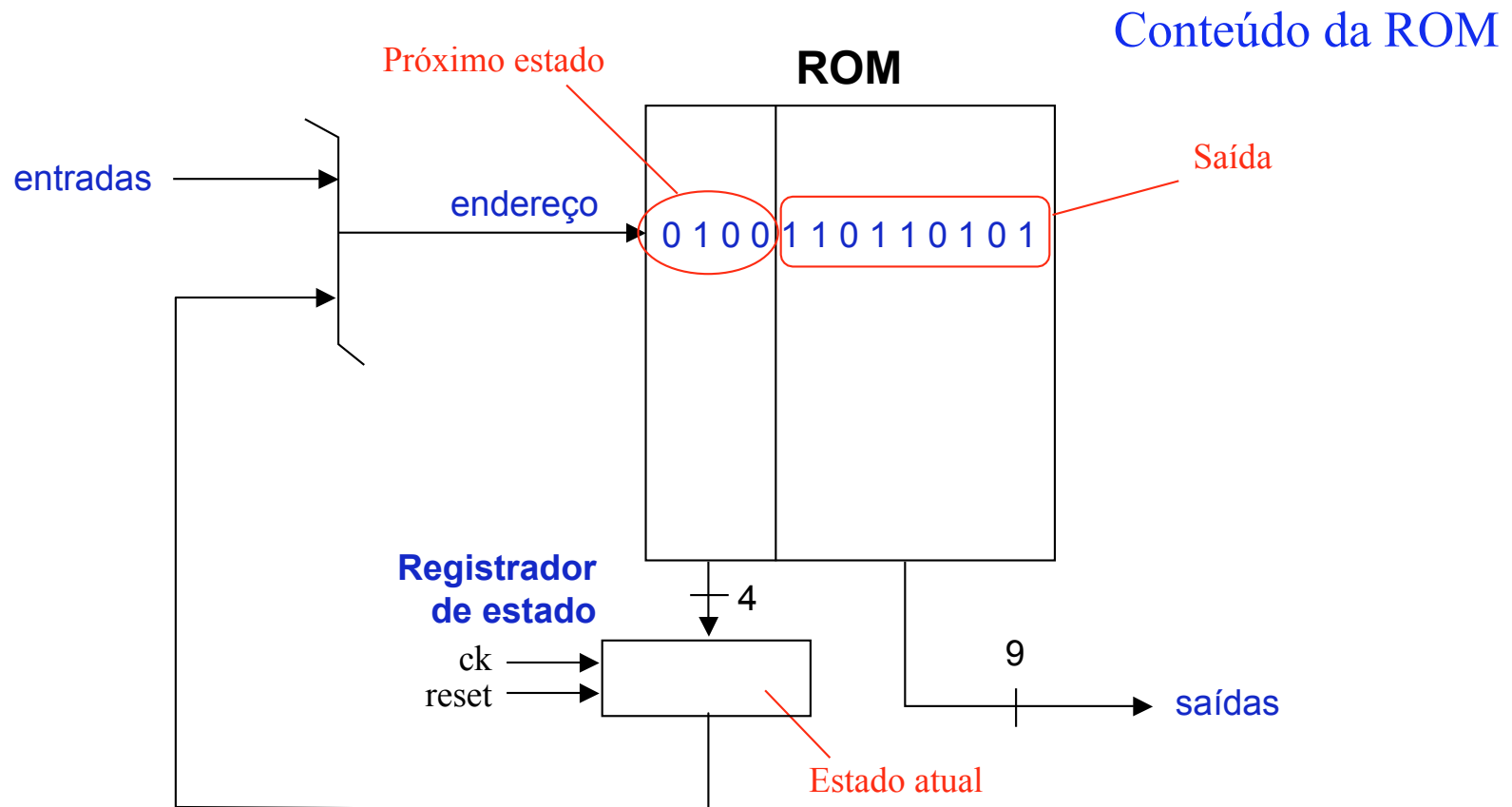
► Alternativas de Implementação de FSMs Lógica de Próximo Estado e Lógica de Saída como Bloco ROM



4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs

Lógica de Próximo Estado e Lógica de Saída como Bloco ROM

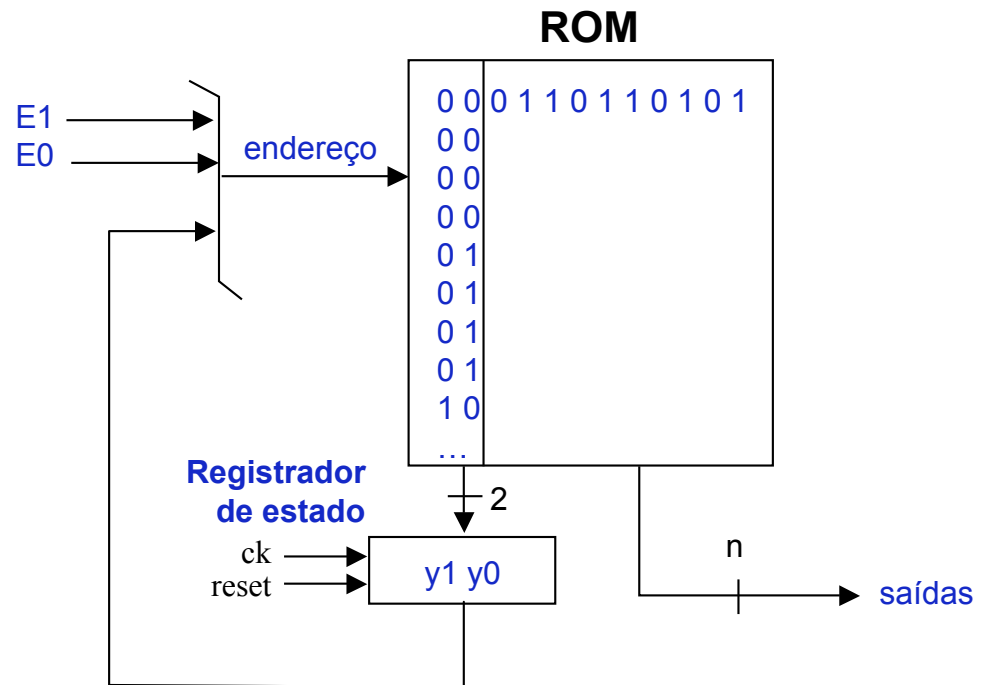


4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs

Para Cada Linha da Tabela de Estados uma Linha da ROM

| Estado atual y1y0 | E1 | E0 | Próximo estado Y1Y0 | saídas |
|----------------------|----|----|------------------------|--------|
| 00 | 0 | 0 | 00 | |
| 00 | 0 | 1 | 00 | |
| 00 | 1 | 0 | 01 | |
| 00 | 1 | 1 | 01 | |
| 01 | 0 | 0 | 01 | |
| 01 | 0 | 1 | 10 | |
| 01 | 1 | 0 | 01 | |
| 01 | 1 | 1 | 10 | |
| 10 | 0 | 0 | 11 | |
| 10 | 0 | 1 | 11 | |
| 10 | 1 | 0 | 11 | |
| 10 | 1 | 1 | 11 | |
| 11 | 0 | 0 | 00 | |
| 11 | 0 | 1 | 00 | |
| 11 | 1 | 0 | 00 | |
| 11 | 1 | 1 | 00 | |



- O número de linhas da ROM é limitado
- Pode haver diversas linhas da tabela de transição que resultem nos mesmo valores para as saídas

4. Máquinas Seqüenciais Síncronas

- ▶ **Alternativas de Implementação de FSMs**
A Solução é Agrupar Casos Equivalentes na Tabela de Transição

| Estado atual y1y0 | E1 | E0 | Próximo estado Y1Y0 |
|----------------------|----|----|------------------------|
| 00 | 0 | 0 | 00 |
| 00 | 0 | 1 | 00 |
| 00 | 1 | 0 | 01 |
| 00 | 1 | 1 | 01 |
| 01 | 0 | 0 | 01 |
| 01 | 0 | 1 | 10 |
| 01 | 1 | 0 | 01 |
| 01 | 1 | 1 | 10 |
| 10 | X | X | 11 |
| 11 | X | X | 00 |

Necessita de **10** linhas da ROM



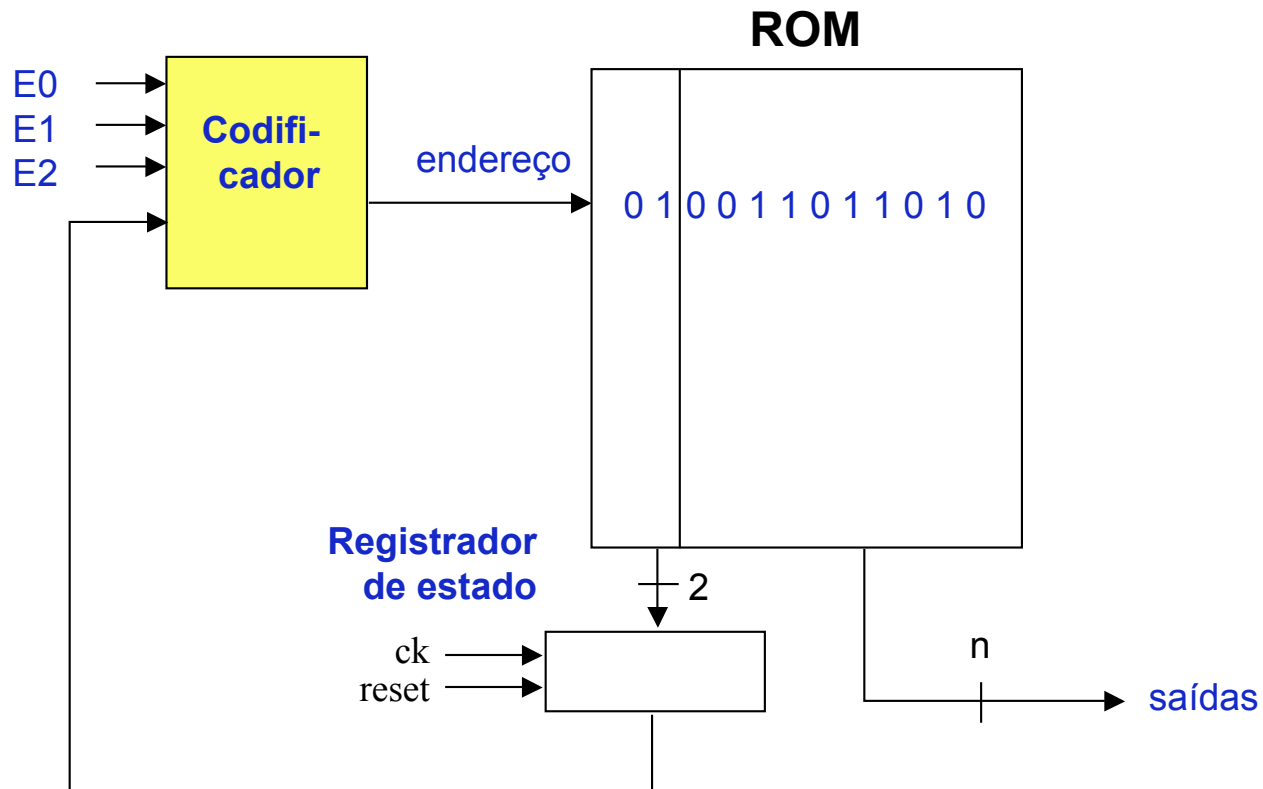
Agrupando-se
os *don't cares*

| Estado atual y1y0 | E1 | E0 | Próximo estado Y1Y0 |
|----------------------|----|----|------------------------|
| 00 | 0 | X | 00 |
| 00 | 1 | X | 01 |
| 01 | X | 0 | 01 |
| 01 | X | 1 | 10 |
| 10 | X | X | 11 |
| 11 | X | X | 00 |

Necessita de **6** linhas da ROM

4. Máquinas Seqüenciais Síncronas

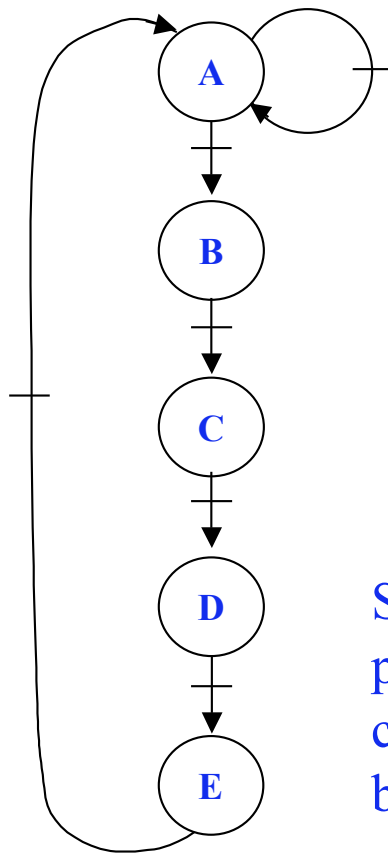
- ▶ **Alternativas de Implementação de FSMs**
O Agrupamento Pode Ser Implementado Por um Codificador



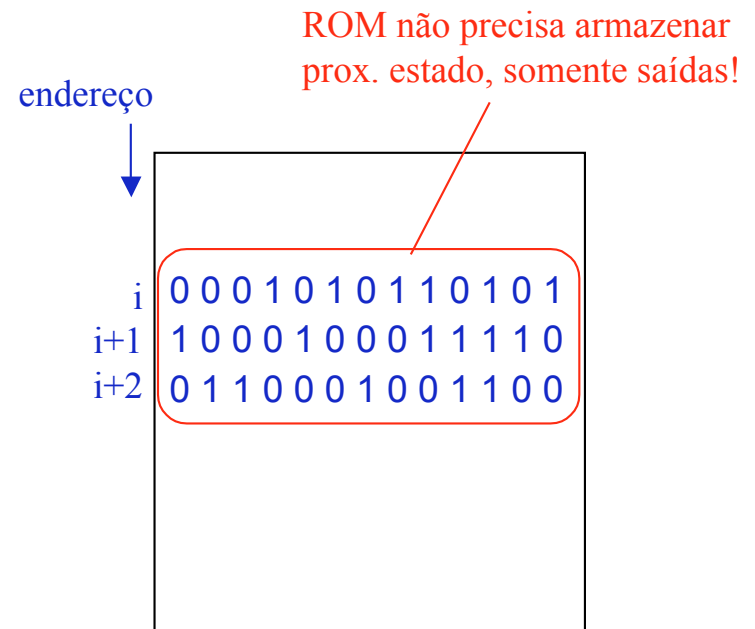
4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs

Explorando Características da FSM

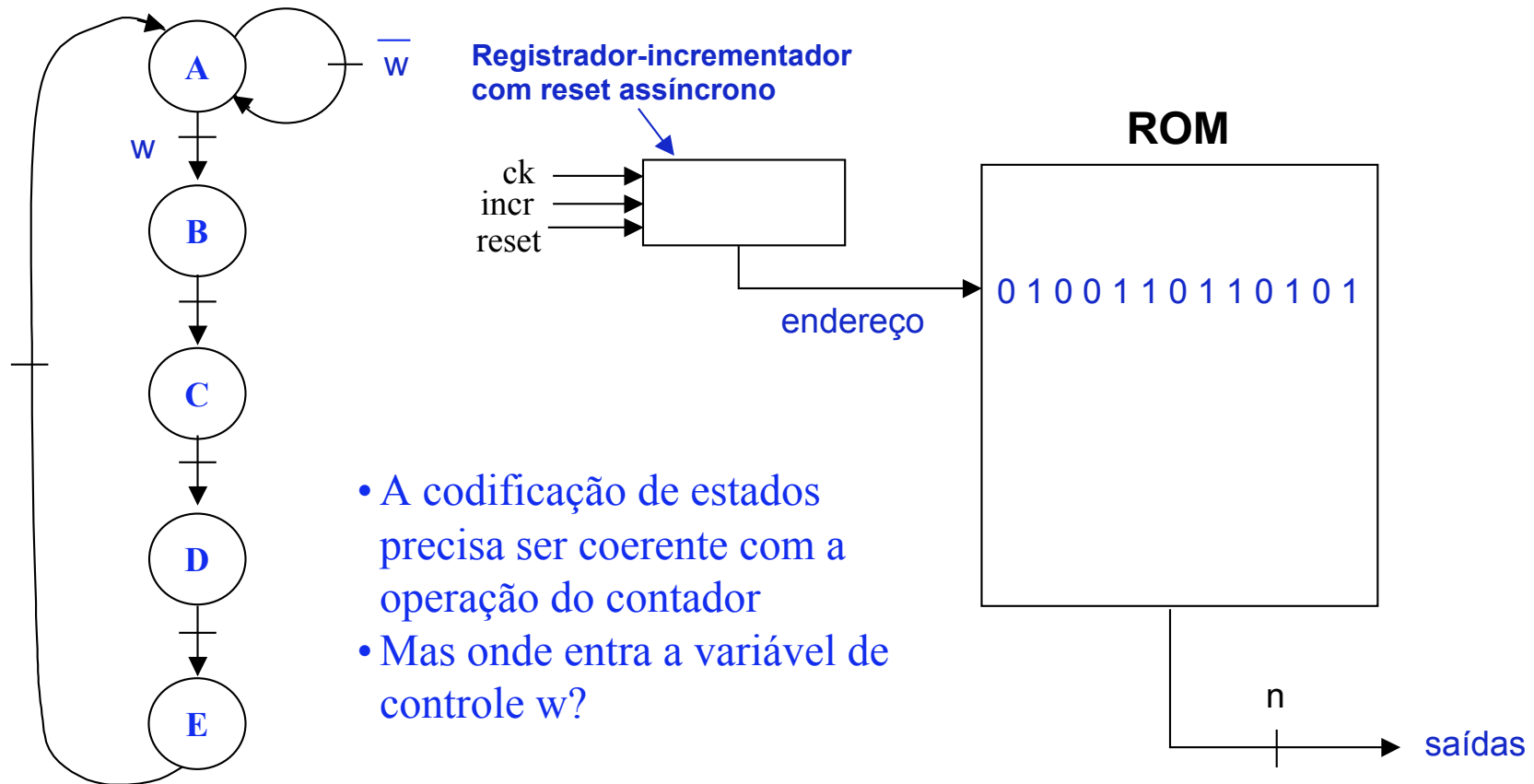


Seqüências de estados podem ser implementados como contagem em binário...



4. Máquinas Seqüenciais Síncronas

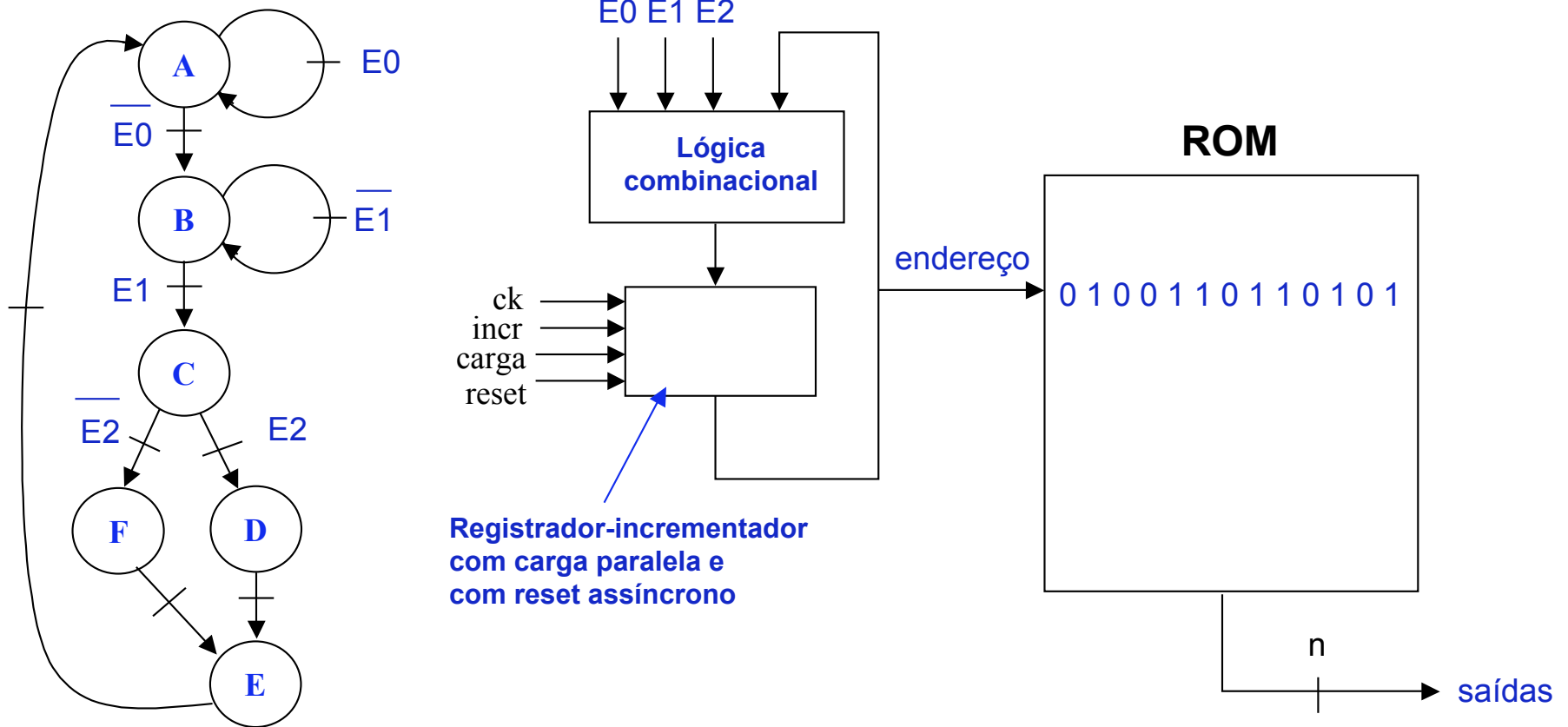
► Alternativas de Implementação de FSMs Bloco ROM + Registrador Contador (Incrementador)



4. Máquinas Seqüenciais Síncronas

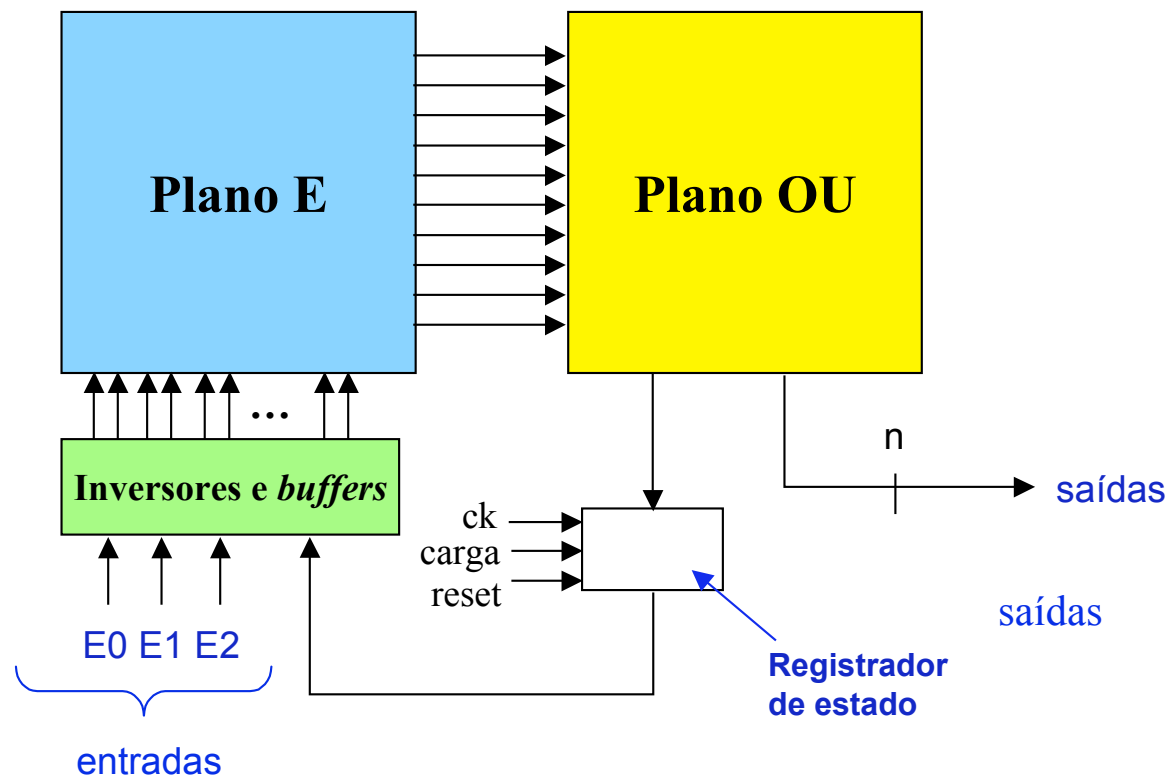
► Alternativas de Implementação de FSMs

Bloco ROM + Registrador Contador (Incrementador)



4. Máquinas Seqüenciais Síncronas

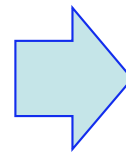
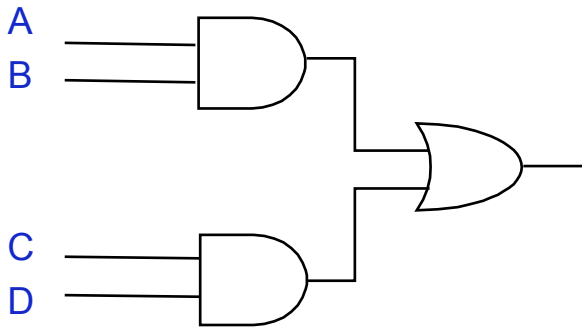
► Alternativas de Implementação de FSMs Lógica de Próximo Estado e Lógica de Saída como PLA



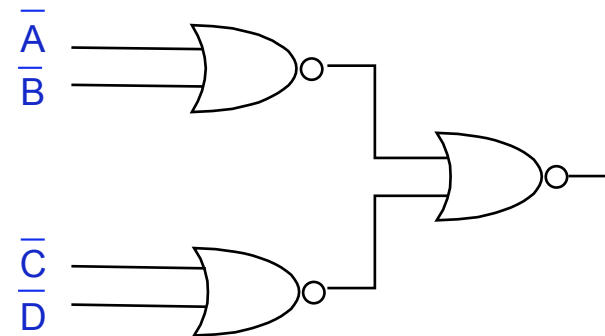
4. Máquinas Seqüenciais Síncronas

▶ Alternativas de Implementação de FSMs PLA em Tecnologia CMOS

Soma de Produtos (SdP)



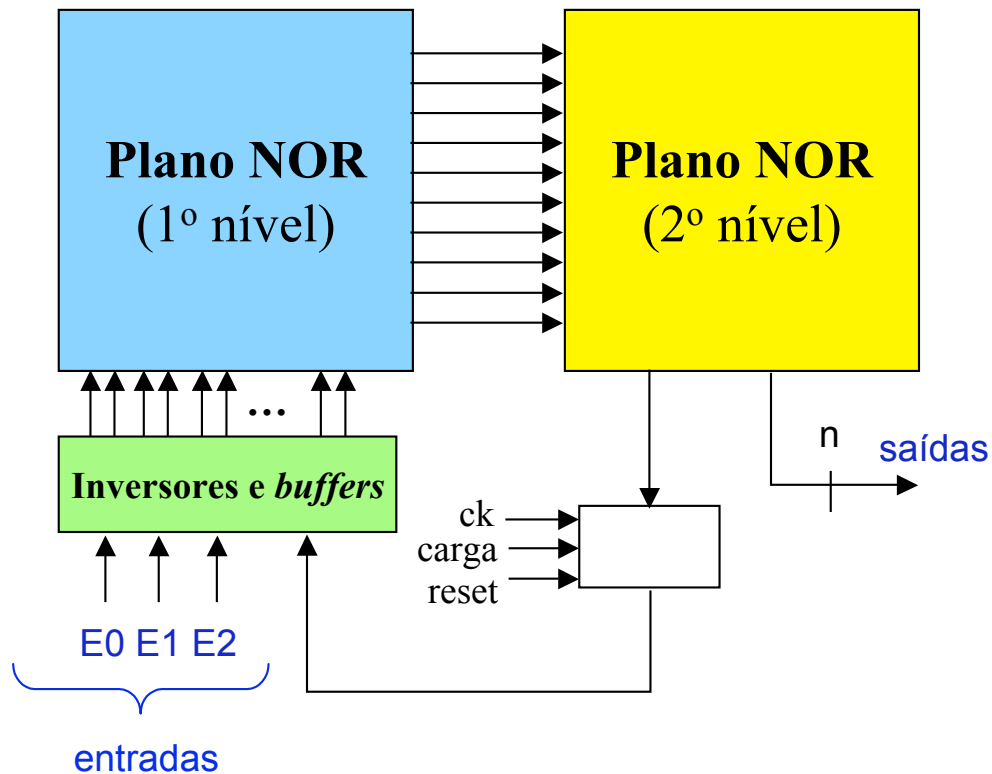
“NOR de NORs”



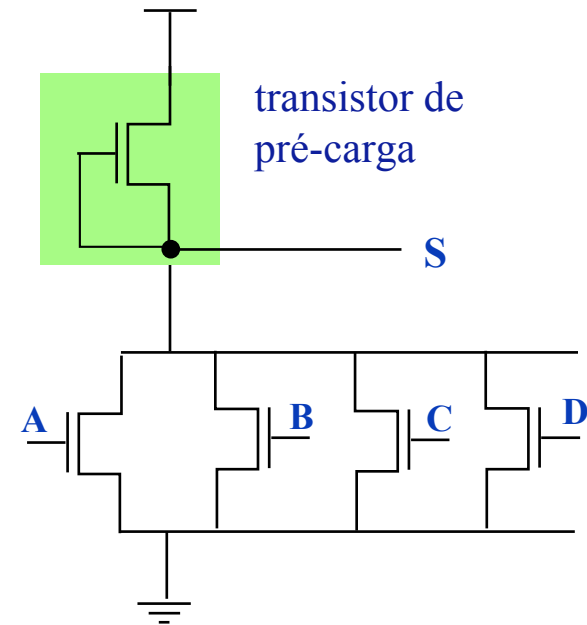
Negando as entradas e aplicando De Morgan...

4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs Lógica de Próximo Estado e Lógica de Saída como PLA

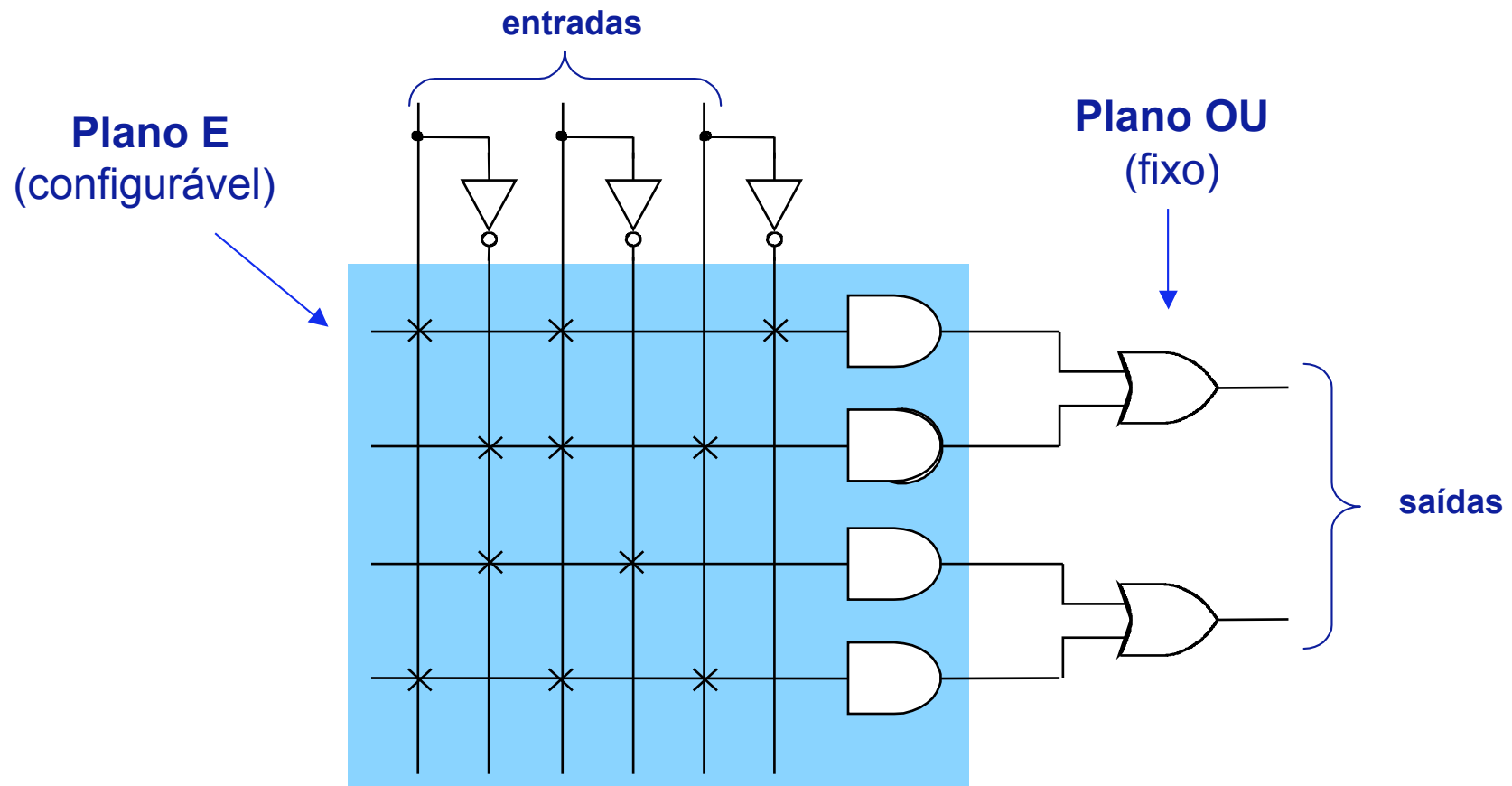


NOR CMOS com muitas entradas



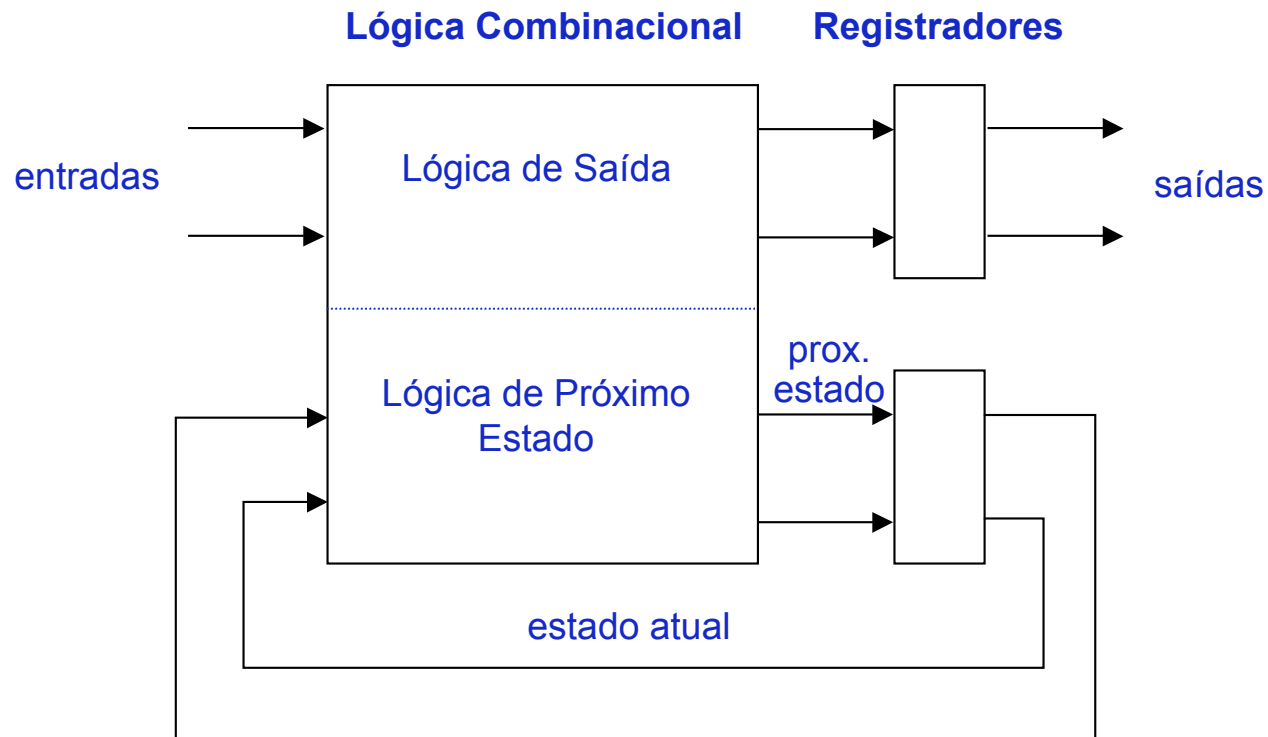
4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs Lógica de Próximo Estado e Lógica de Saída como PAL



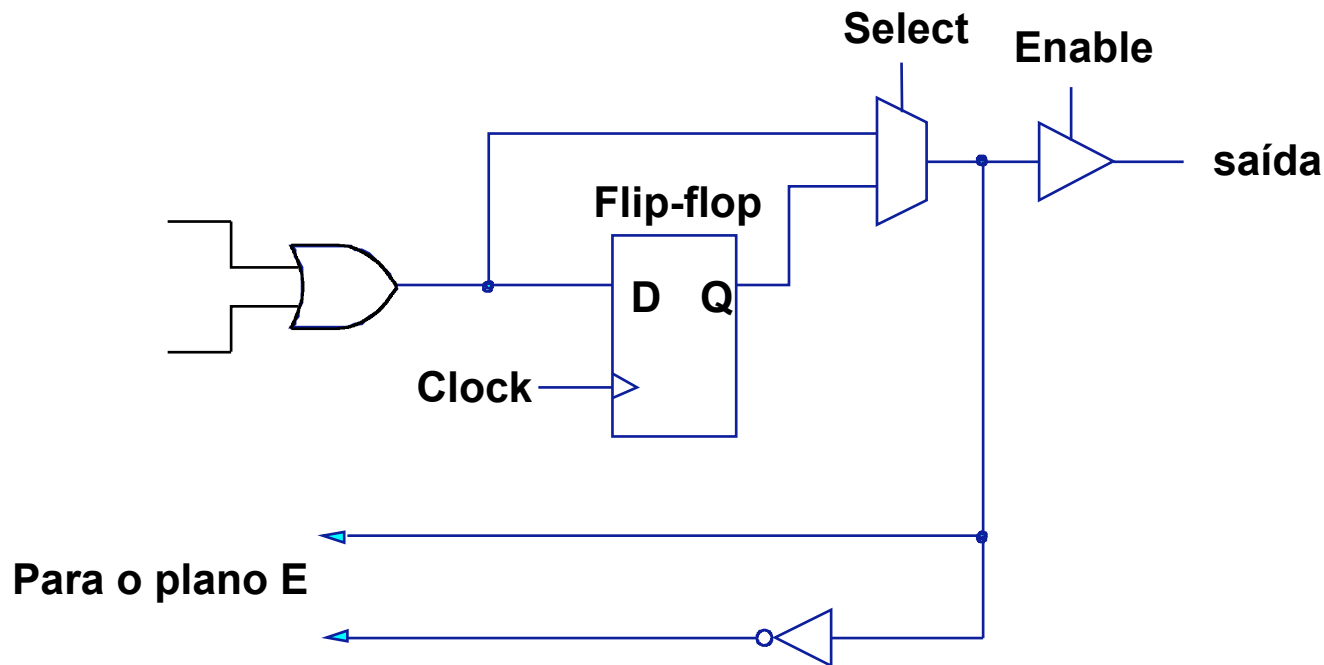
4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs Implementando uma FSM Completa com PAL



4. Máquinas Seqüenciais Síncronas

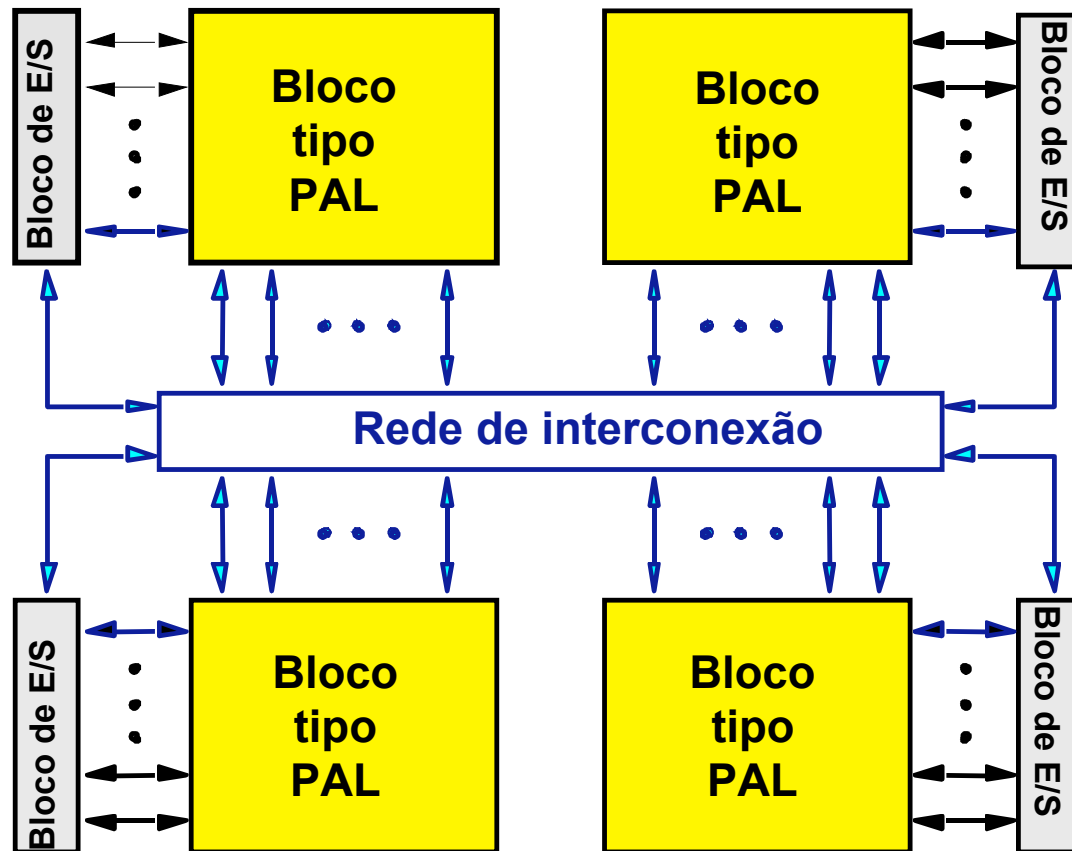
► Alternativas de Implementação de FSMs Implementando uma FSM Completa com PAL



4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs

Implementando uma FSM Completa em um CPLD



Estrutura de um CPLD

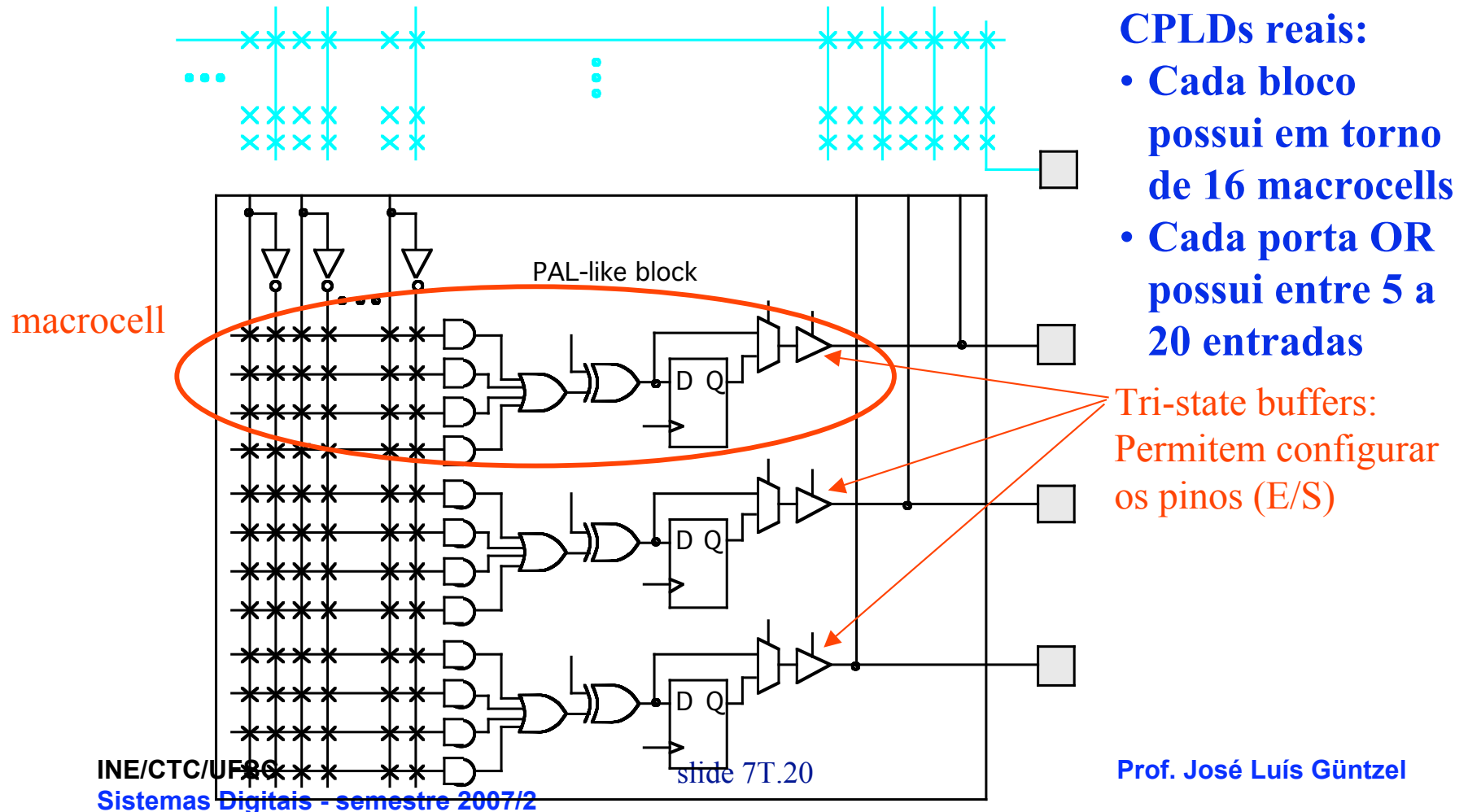
CPLDs existentes no mercado:

- Possuem entre 2 a 100 blocos tipo PAL
- Variedade de encapsulamentos, com até 200 pinos (QFP)

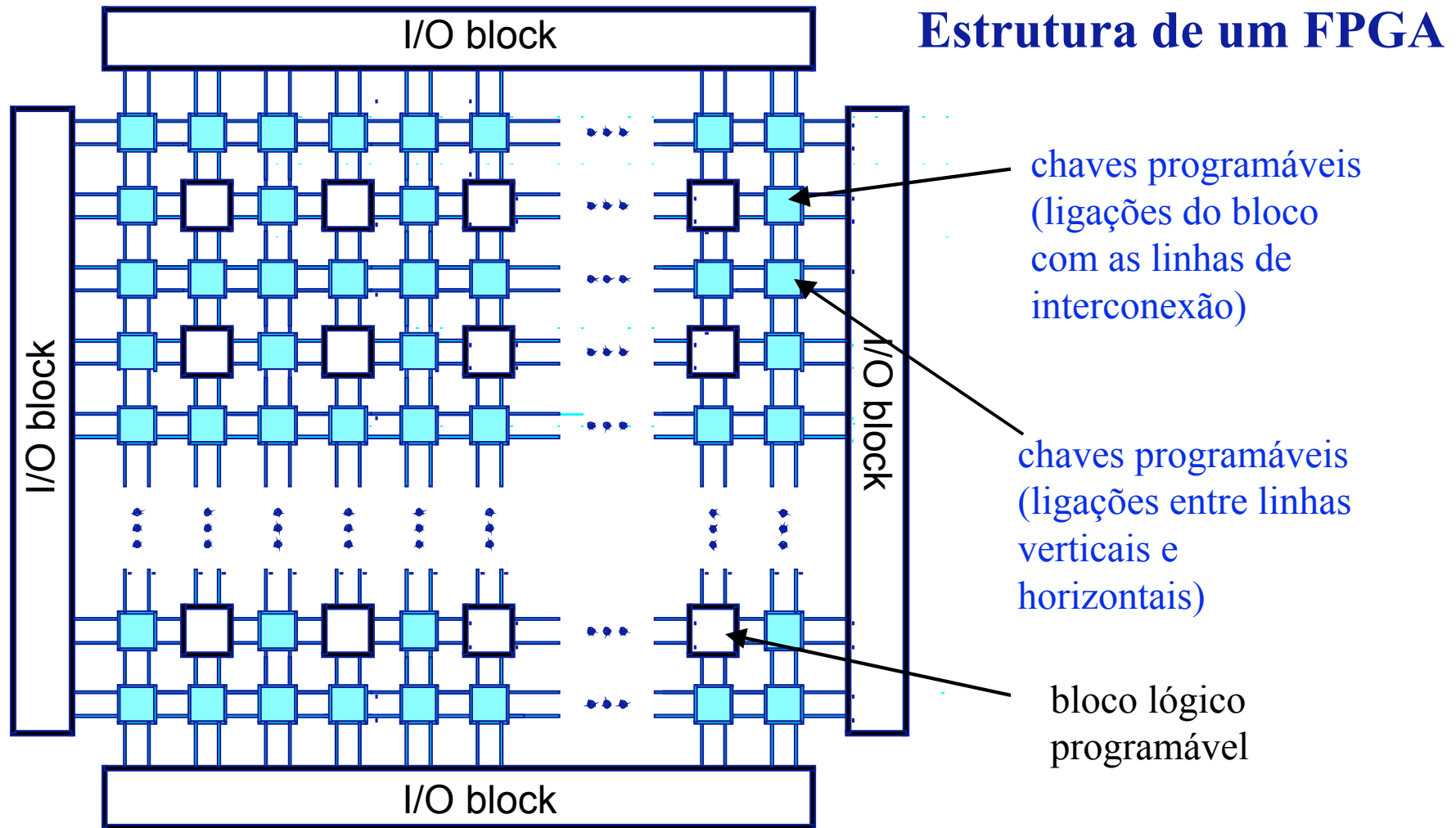
4. Máquinas Seqüenciais Síncronas

▶ Alternativas de Implementação de FSMs

Implementando uma FSM Completa em um CPLD

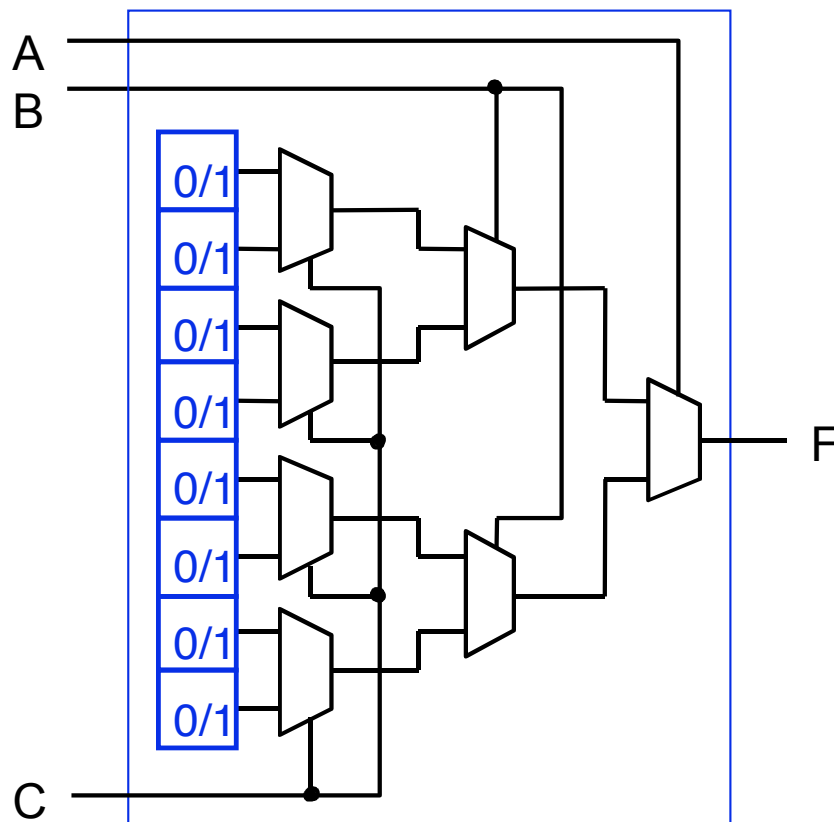


4. Máquinas Seqüenciais Síncronas



4. Máquinas Seqüenciais Síncronas

► FPGAs: LUTs (*Lookup Tables*)



- Implementadas com muxes 2:1 e bits de memória, SRAM (reprogramabilidade...)
- Normalmente, possuem 4 ou 5 entradas
- Implementam qualquer função lógica. Para 4 entradas, existem

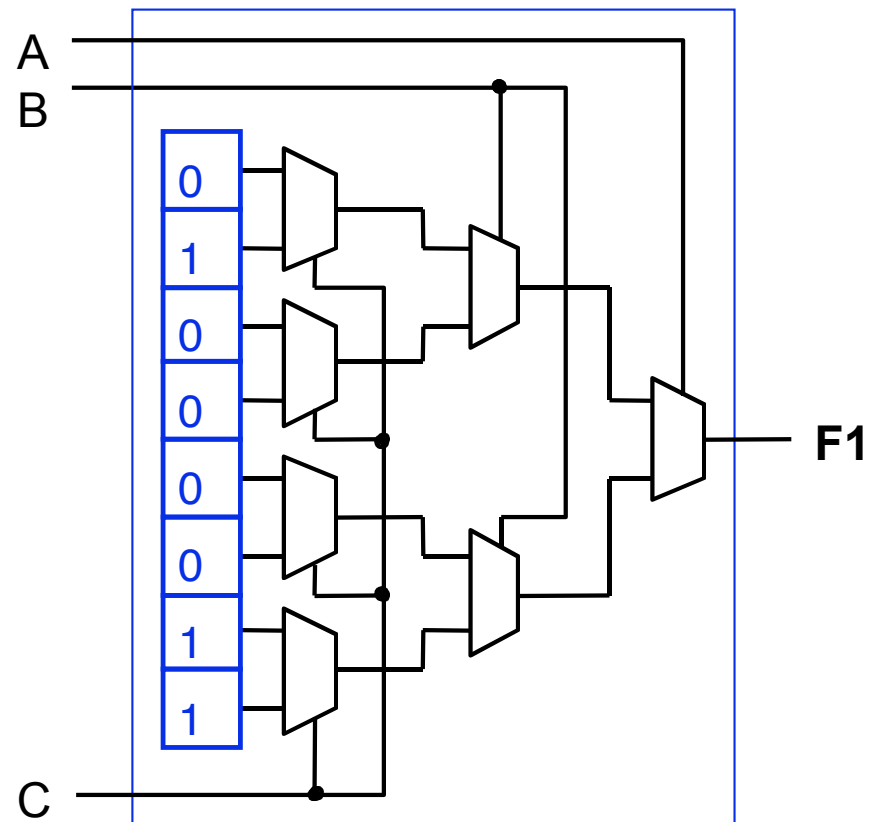
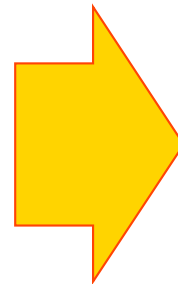
$$2^{2^4} = 65.536 \text{ diferentes funções!!!}$$

4. Máquinas Seqüenciais Síncronas

► FPGAs: LUTs (*Lookup Tables*)

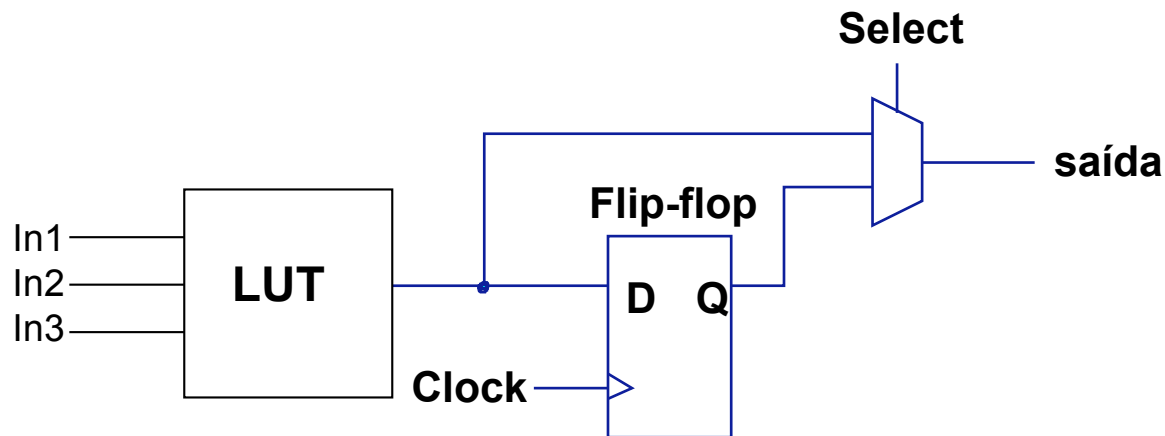
Programando LUTs

| A | B | C | F1 | F2 |
|---|---|---|----|----|
| 0 | 0 | 0 | 0 | 0 |
| 0 | 0 | 1 | 1 | 1 |
| 0 | 1 | 0 | 0 | 0 |
| 0 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | 1 |



4. Máquinas Seqüenciais Síncronas

► Alternativas de Implementação de FSMs Implementando uma FSM Completa em um FPGA



4. Máquinas Seqüenciais Síncronas

Arquitetura do CLB do dispositivo VIRTEX-II

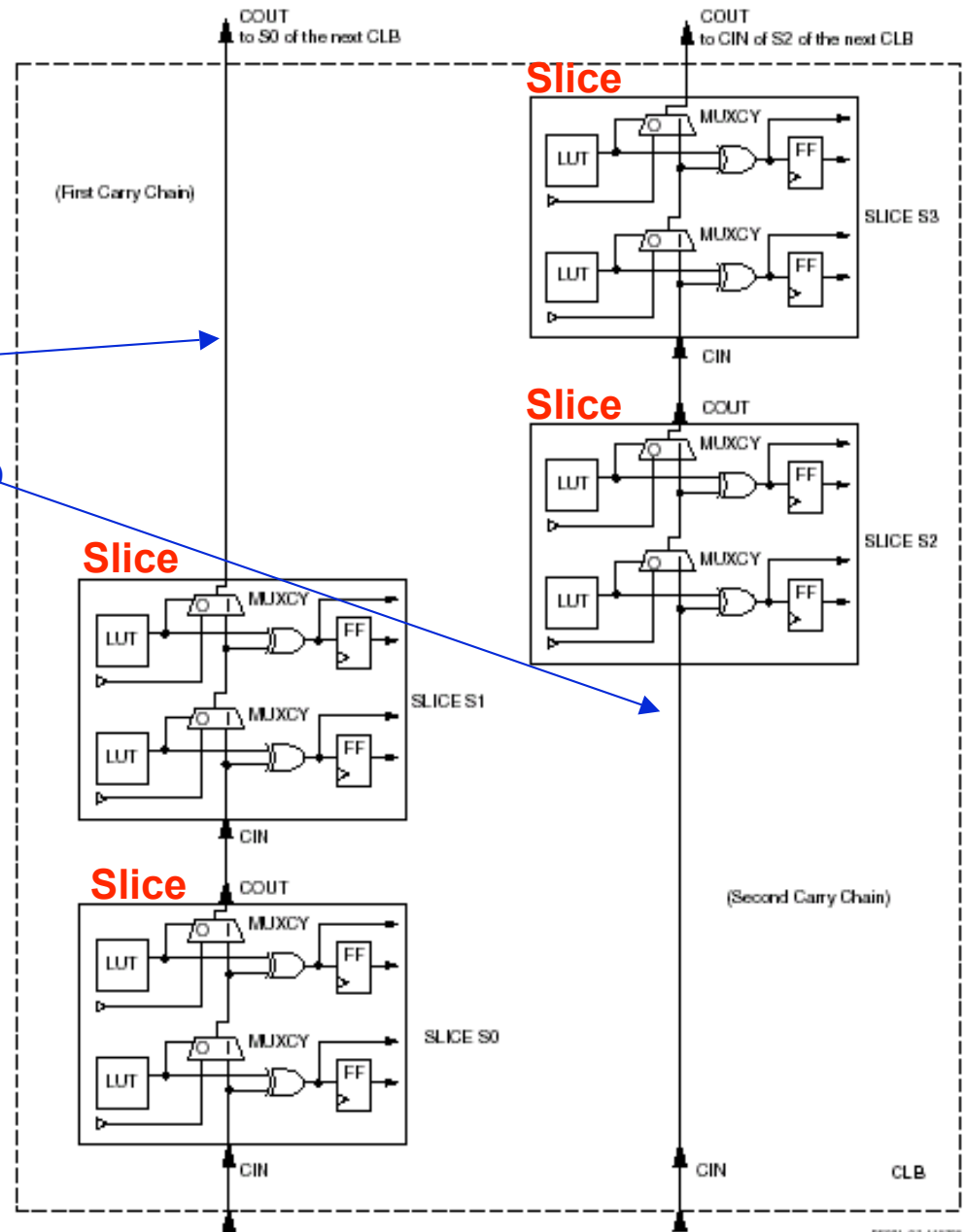
- Fast Carry Logic Path
- Provides fast arithmetic add and sub

RESUMINDO O CLB

- 4 Slices
- 8 LUTS / 8 Flip-Flops
- 2 cadeias de vai-um
- 64 bits para memória
- 64 bits para shift-register

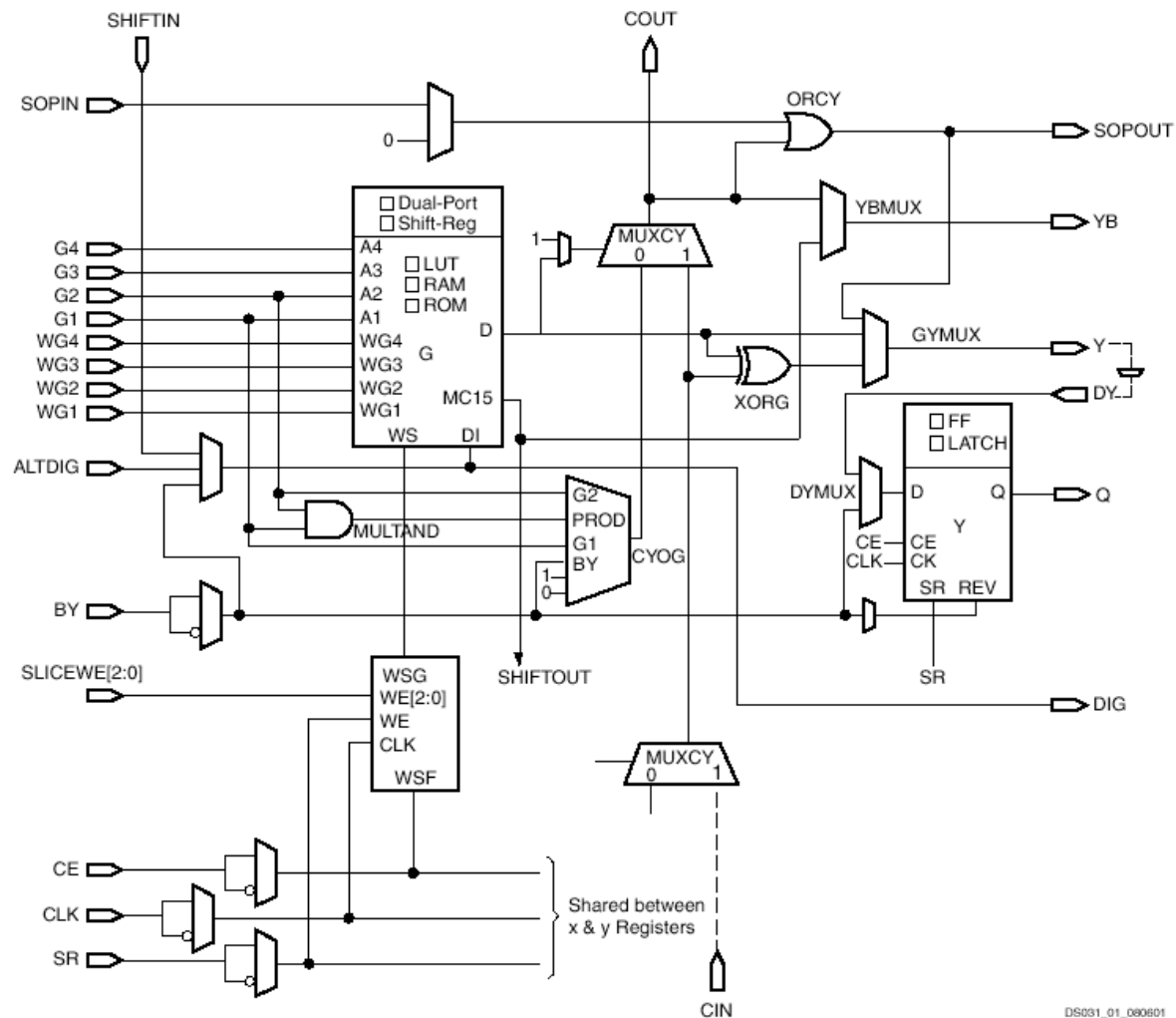
Transparência de F. Moraes (PUCRS)

INE/CTC/UFSC
Sistemas Digitais - semestre 2007/2



4. Máquinas Seqüenciais Síncronas

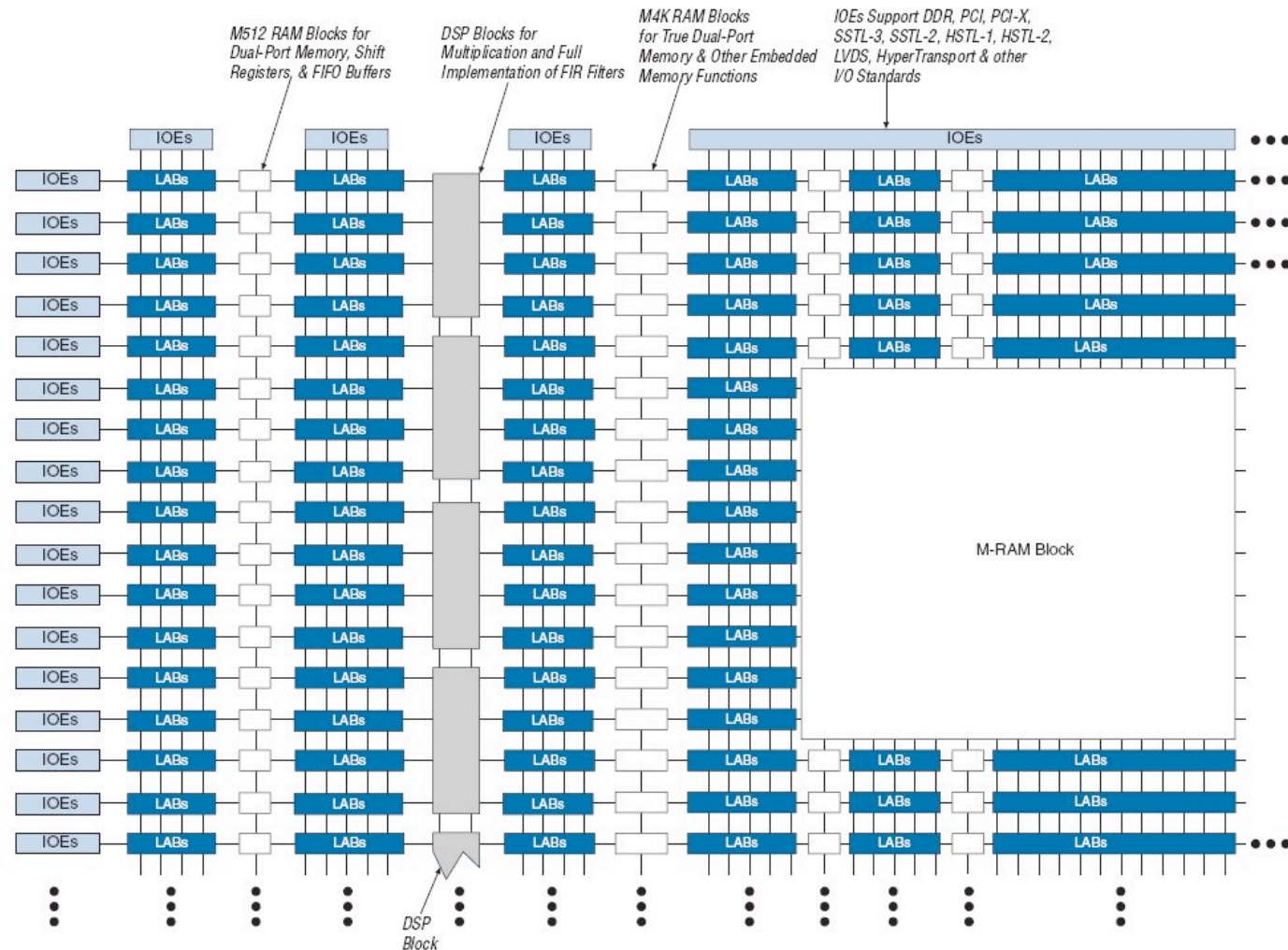
Arquitetura (metade) do Slice



4. Máquinas Seqüenciais Síncronas

► FPGAs Altera: Stratix II

Estrutura Básica da Matriz do Stratix II



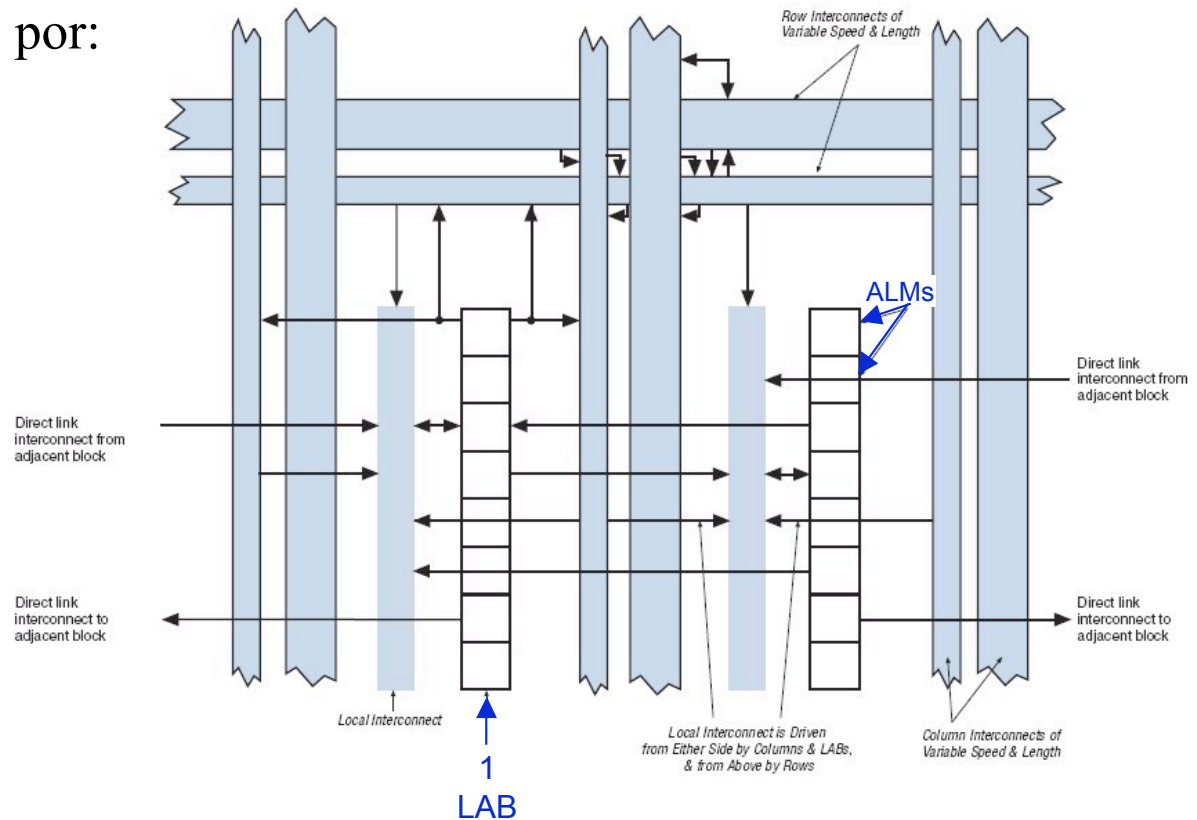
4. Máquinas Seqüenciais Síncronas

► FPGAs Altera: Stratix II

Estrutura de um LAB (*Logic Array Block*)

Cada LAB é constituído por:

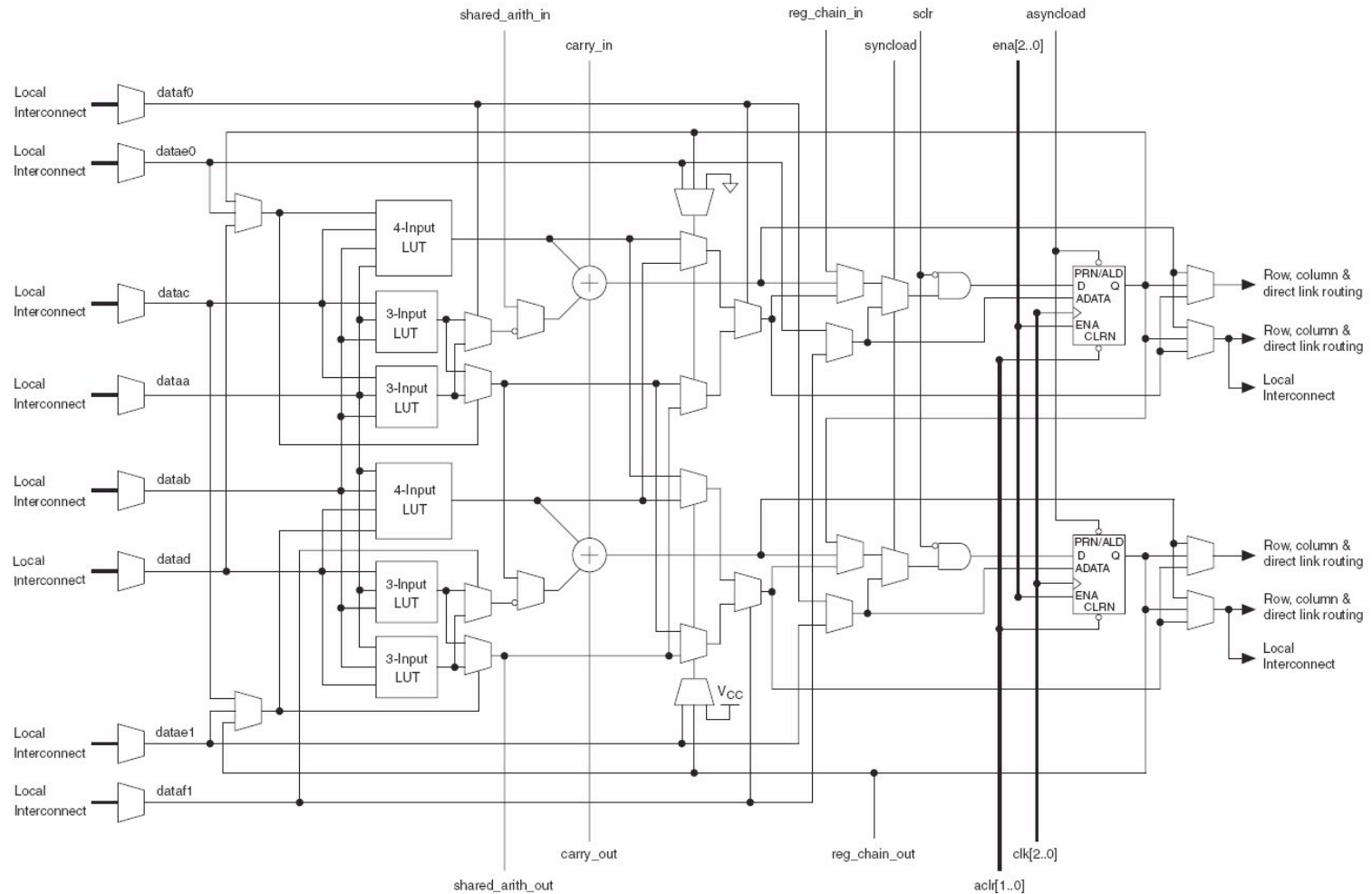
- 8 ALMs (*Adaptative Logic Modules*)
- Cadeia de *carry*
- Cadeia aritmética compartilhada
- Sinais de controle do LAB
- Conexões locais
- Cadeia de registradores



4. Máquinas Seqüenciais Síncronas

► FPGAs Altera: Stratix II

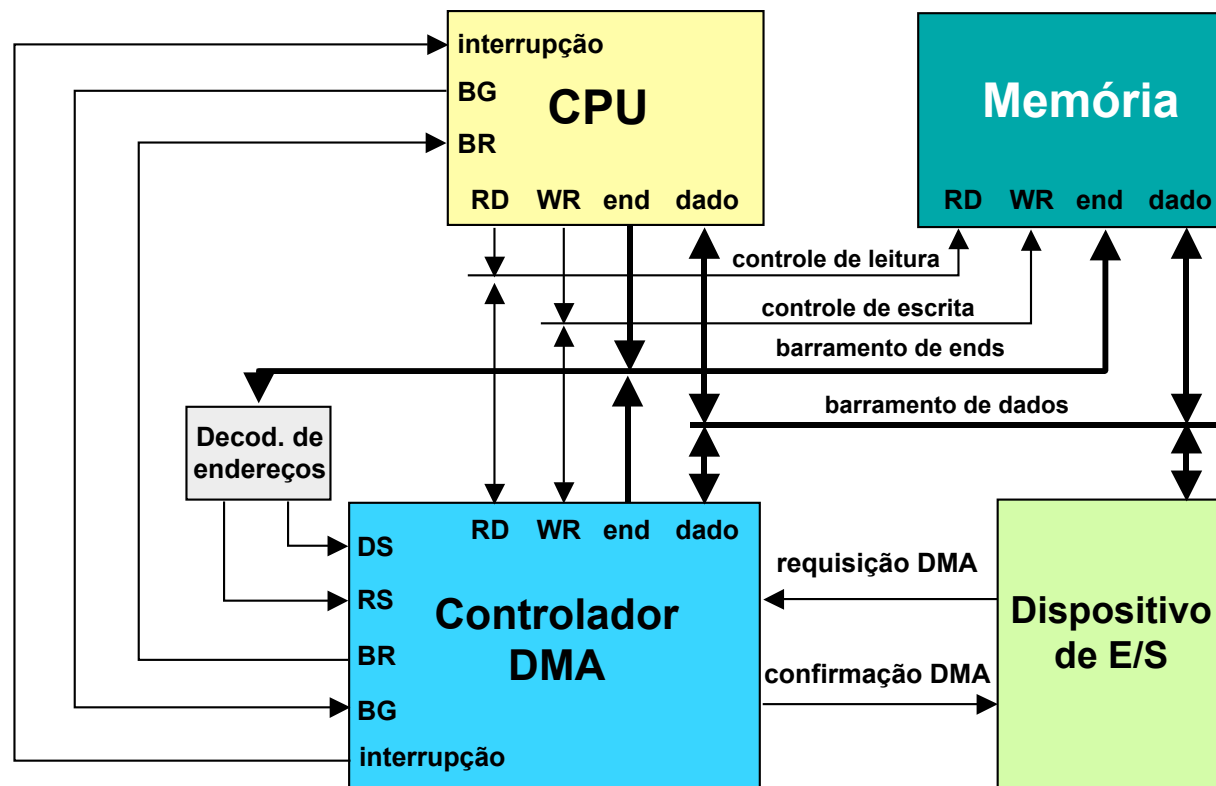
Detalhes de um ALM



4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 8: Controlador de DMA



Sinais:

BR= Bus Request

BG= Bus Grant

RD= Read

WR= Write

DS= DMA select

RS= register select

4. Máquinas Seqüenciais Síncronas

▶ Síntese de Circuitos Seqüenciais

Exemplo 8: Controlador de DMA

Passo 1: processador ativa o DMAC (controlador de DMA)

- Informando a identidade do dispositivo
- A operação a ser realizada pelo dispositivo
- O endereço de memória que é a fonte ou o destino do dado e
- O número de bytes a serem transferidos

4. Máquinas Seqüenciais Síncronas

▶ Síntese de Circuitos Seqüenciais

Exemplo 8: Controlador de DMA

Passo 2: O DMAC inicia a operação sobre o dispositivo:

- **Arbitrando o barramento**
- **Quando os dados estiverem disponíveis ele os transfere**
- **Fornece os endereços de memória a serem lidos ou escritos**
- **Gera o próximo endereço de memória e inicia a nova transferência...**

4. Máquinas Seqüenciais Síncronas

▶ Síntese de Circuitos Seqüenciais

Exemplo 8: Controlador de DMA

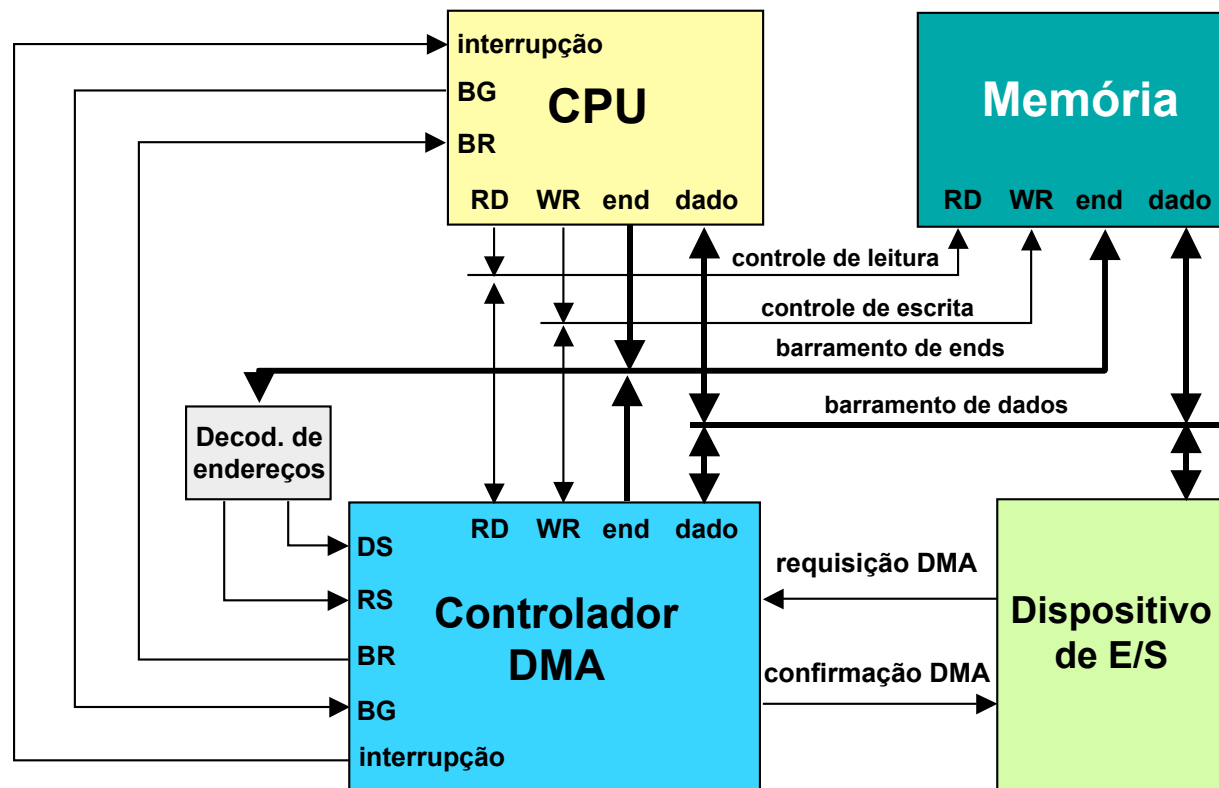
Passo 3: Uma vez completada a transferência sob o controle do DMAC:

- O DMAC interrompe o processador,
- O processador deve verificar se toda a operação de transferência foi realizada com sucesso.
- O processador retoma o controle do barramento

4. Máquinas Seqüenciais Síncronas

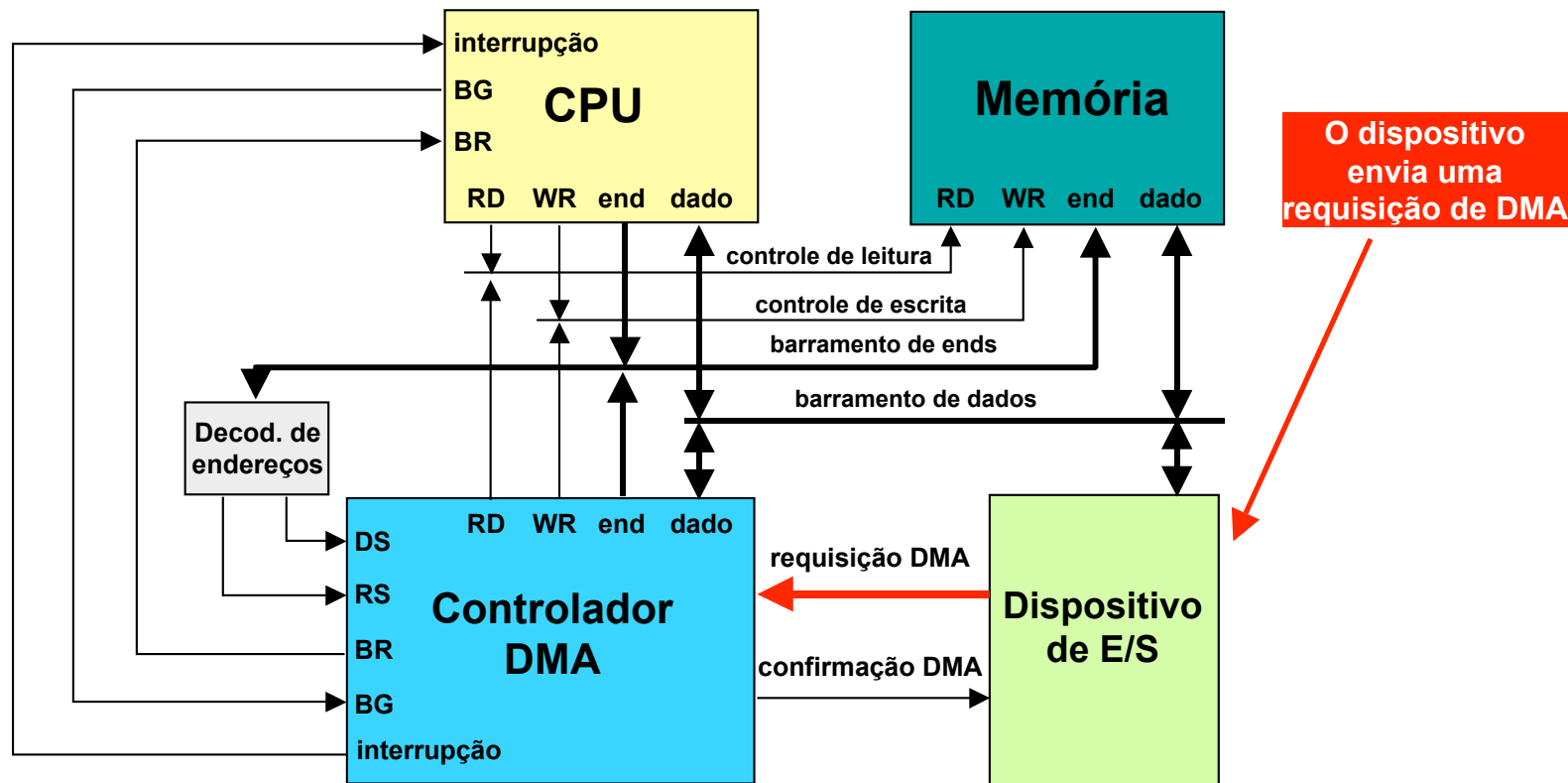
▶ Acesso Direto à Memória (DMA)

O controlador de DMA no sistema



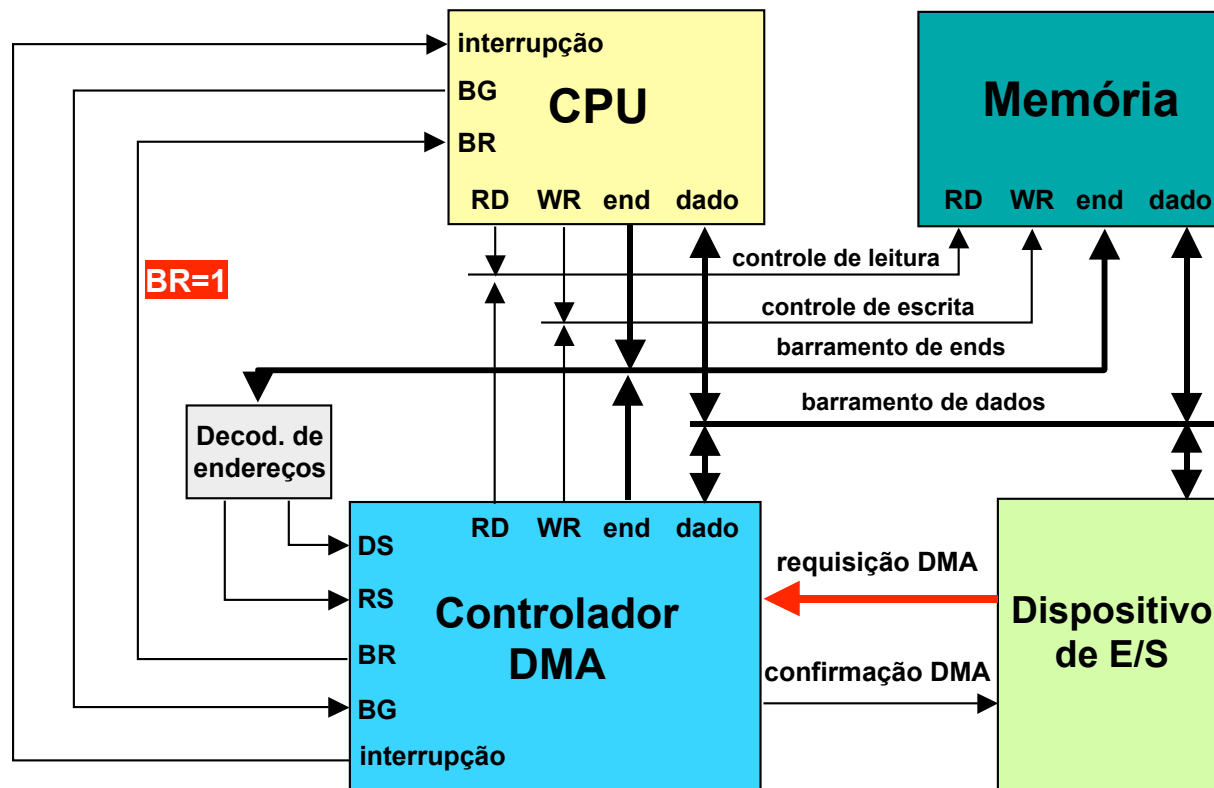
4. Máquinas Seqüenciais Síncronas

▶ Transferência DMA



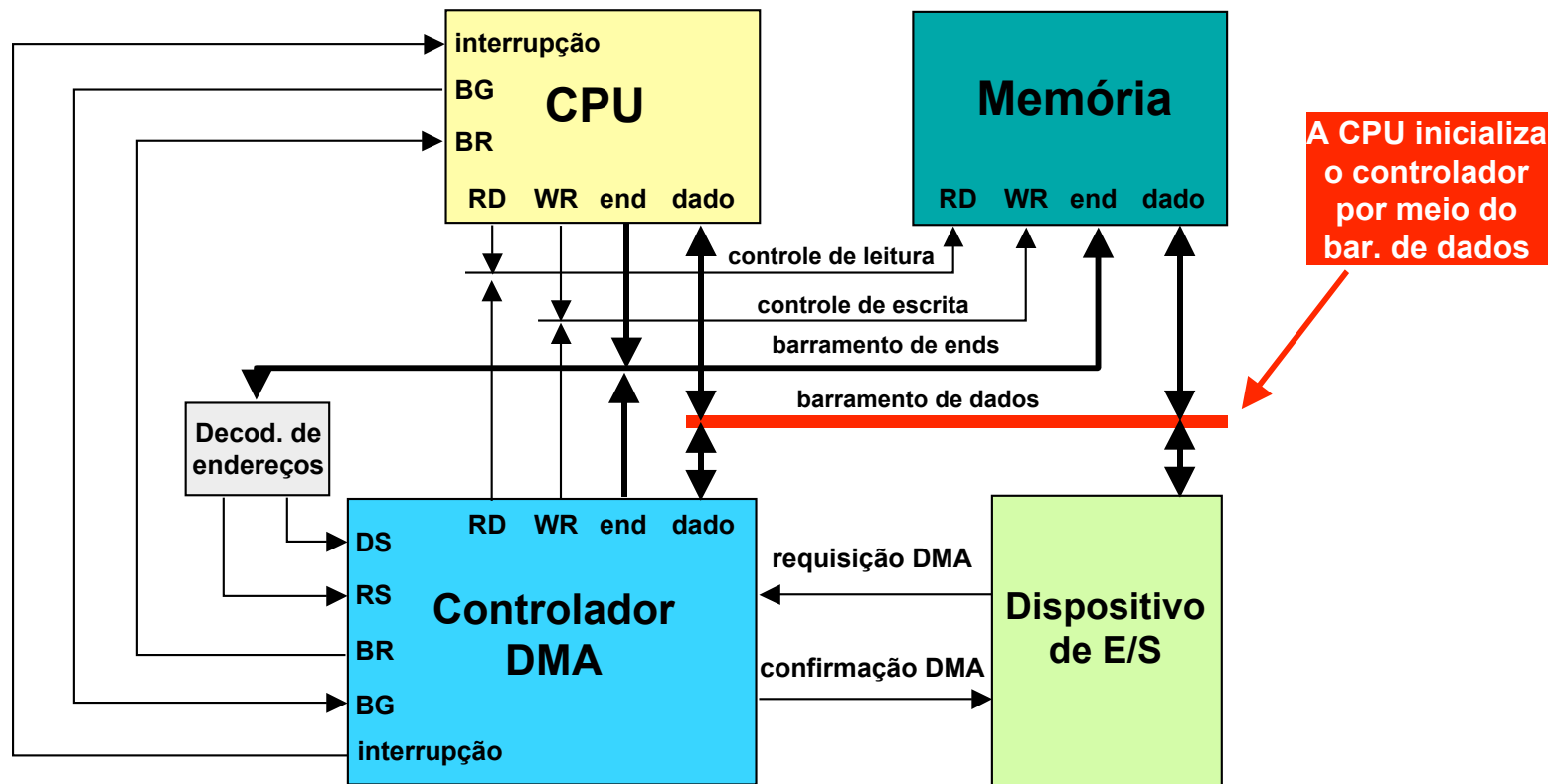
4. Máquinas Seqüenciais Síncronas

▶ Transferência DMA



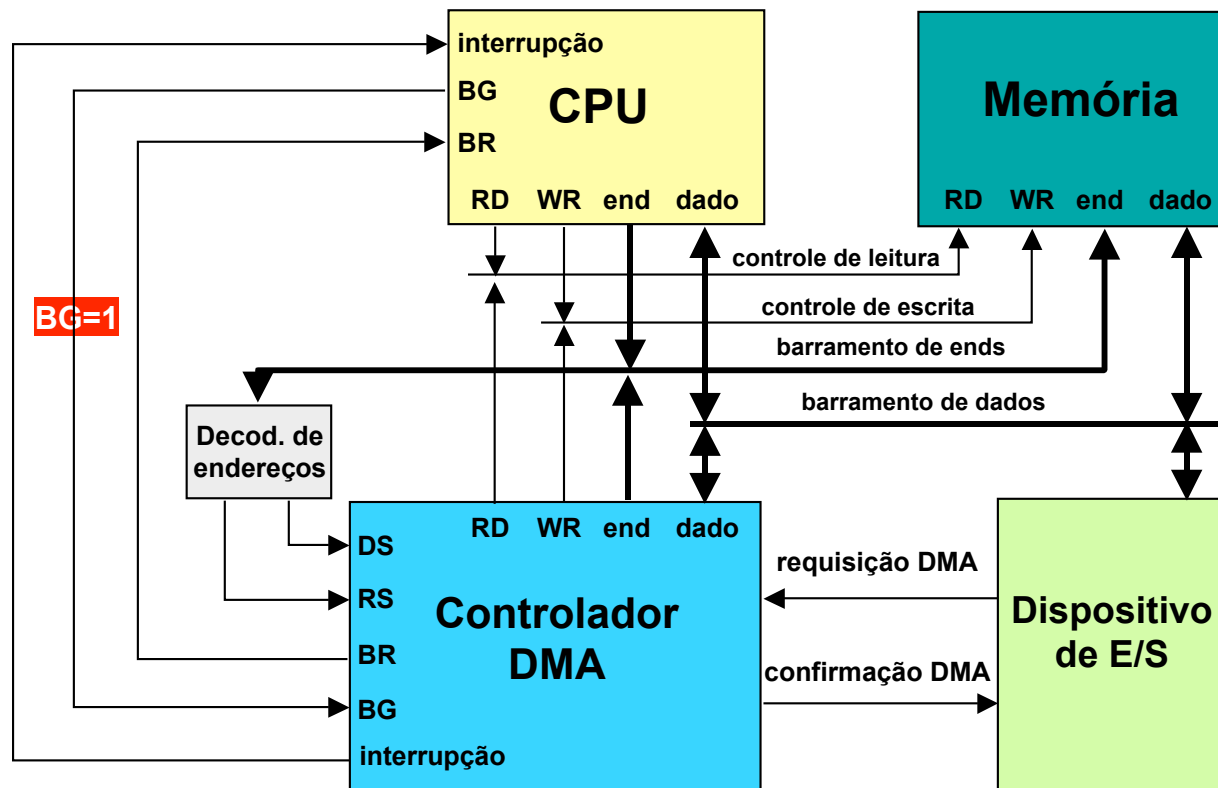
4. Máquinas Seqüenciais Síncronas

▶ Transferência DMA



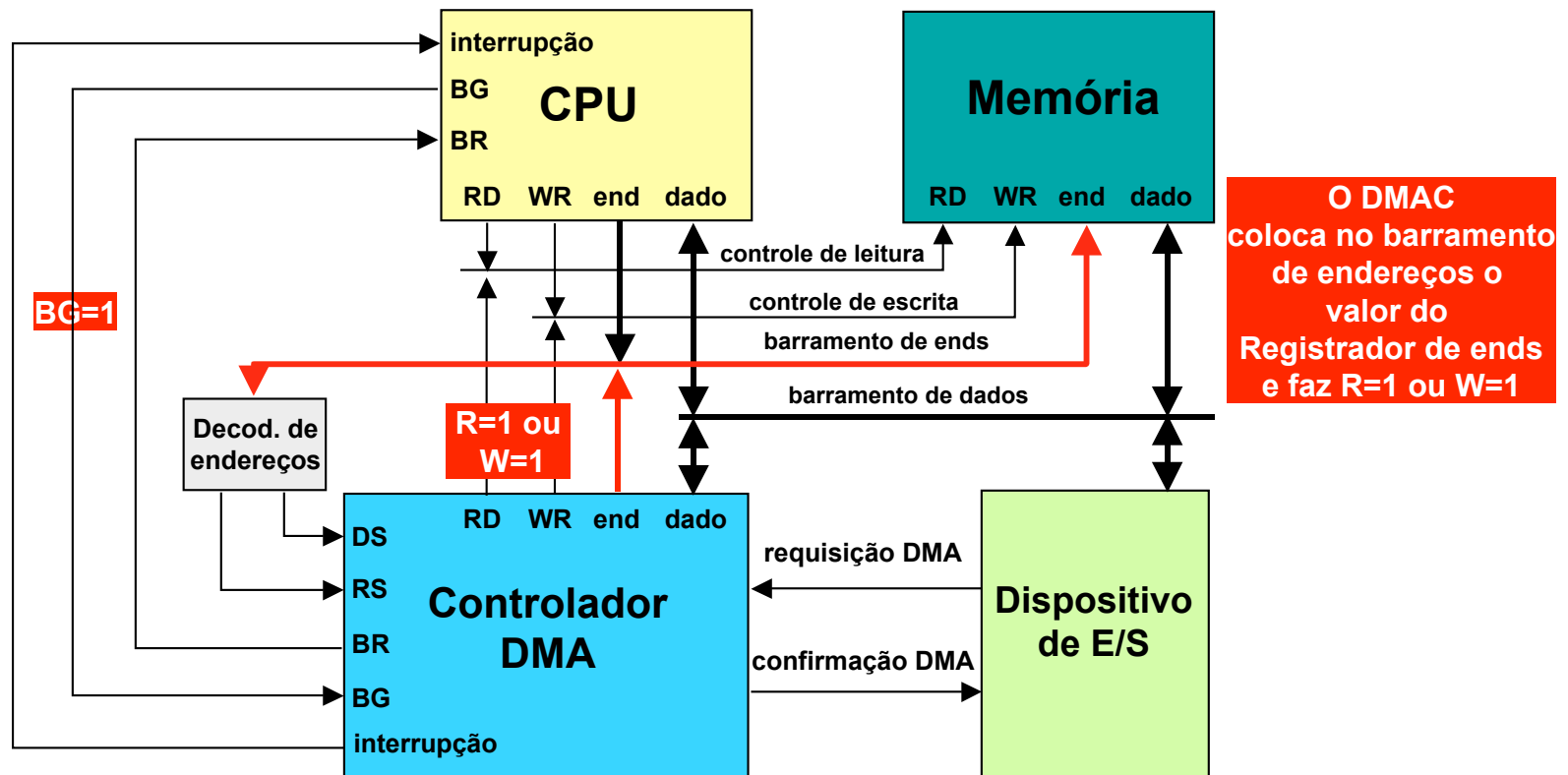
4. Máquinas Seqüenciais Síncronas

▶ Transferência DMA



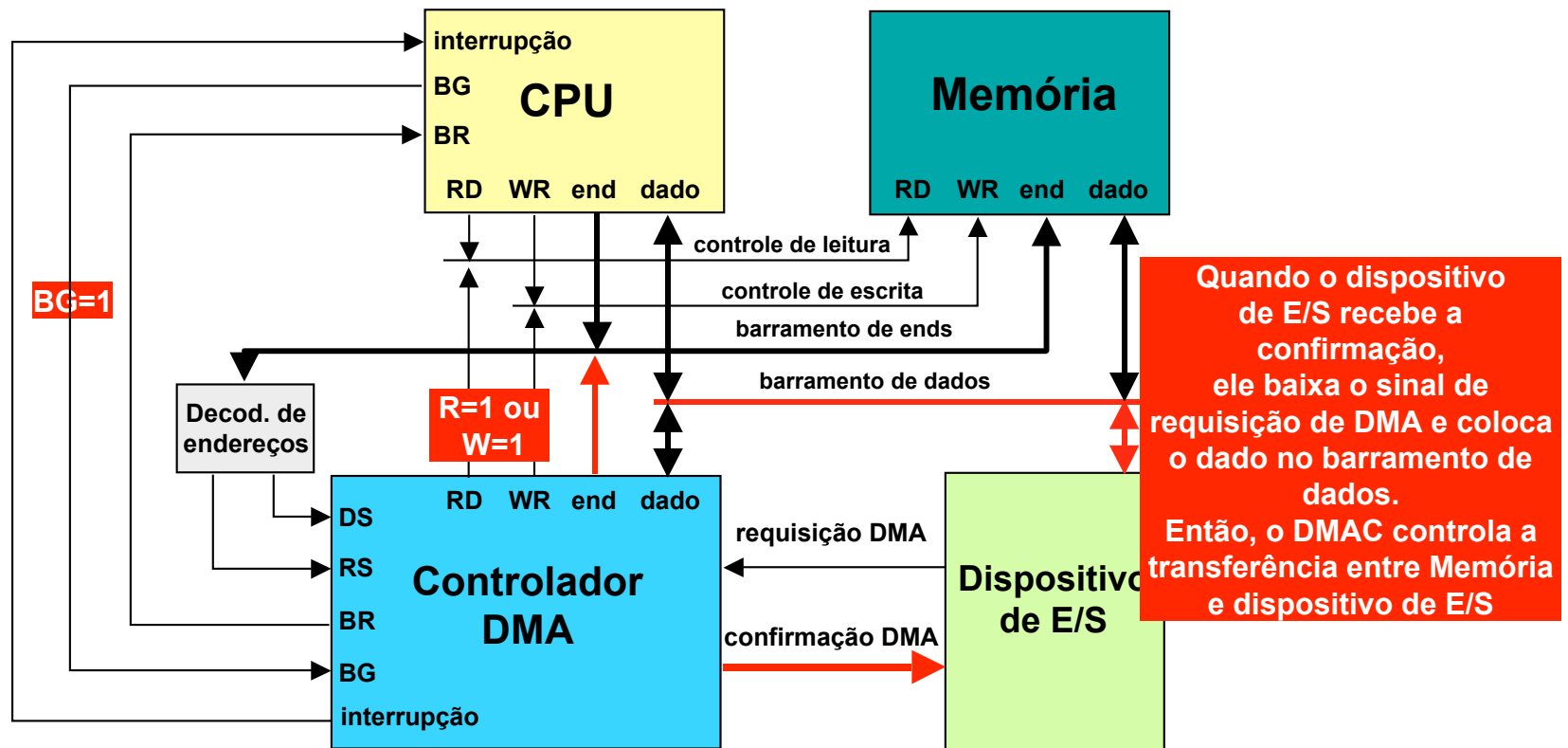
4. Máquinas Seqüenciais Síncronas

▶ Transferência DMA



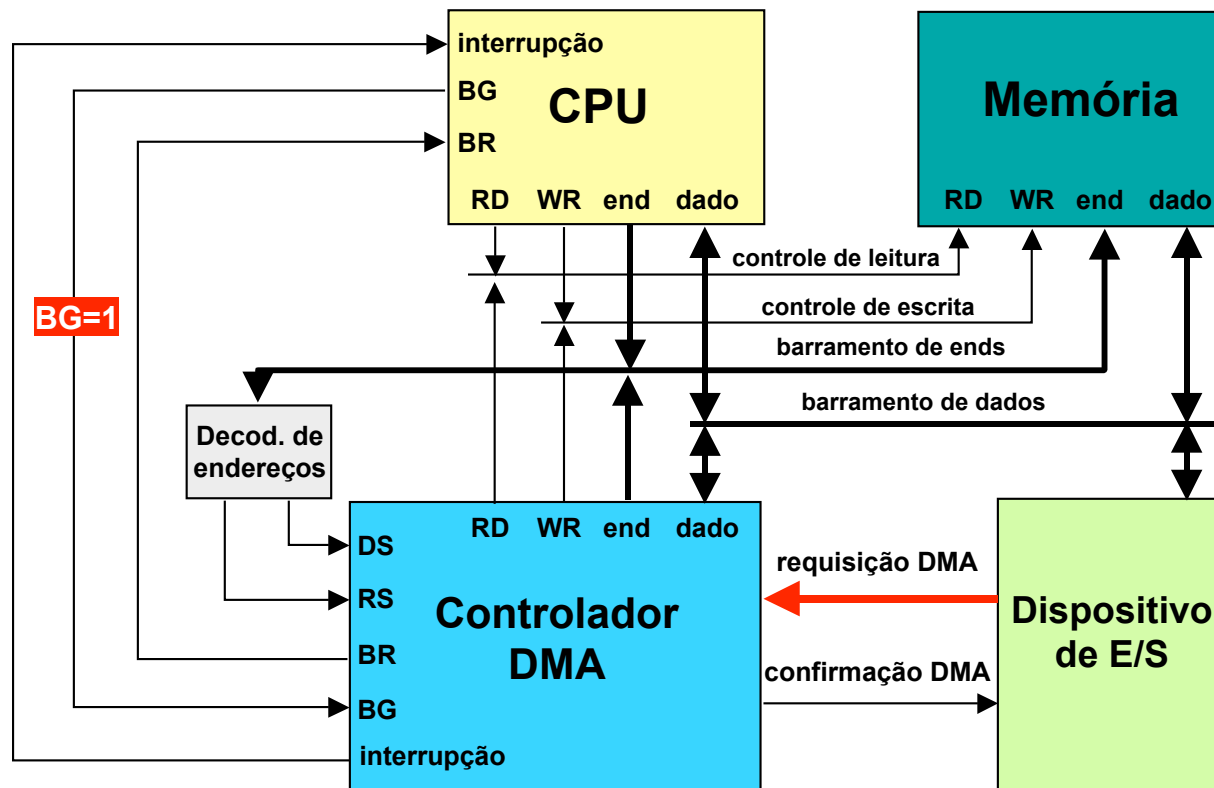
4. Máquinas Seqüenciais Síncronas

▶ Transferência DMA



4. Máquinas Seqüenciais Síncronas

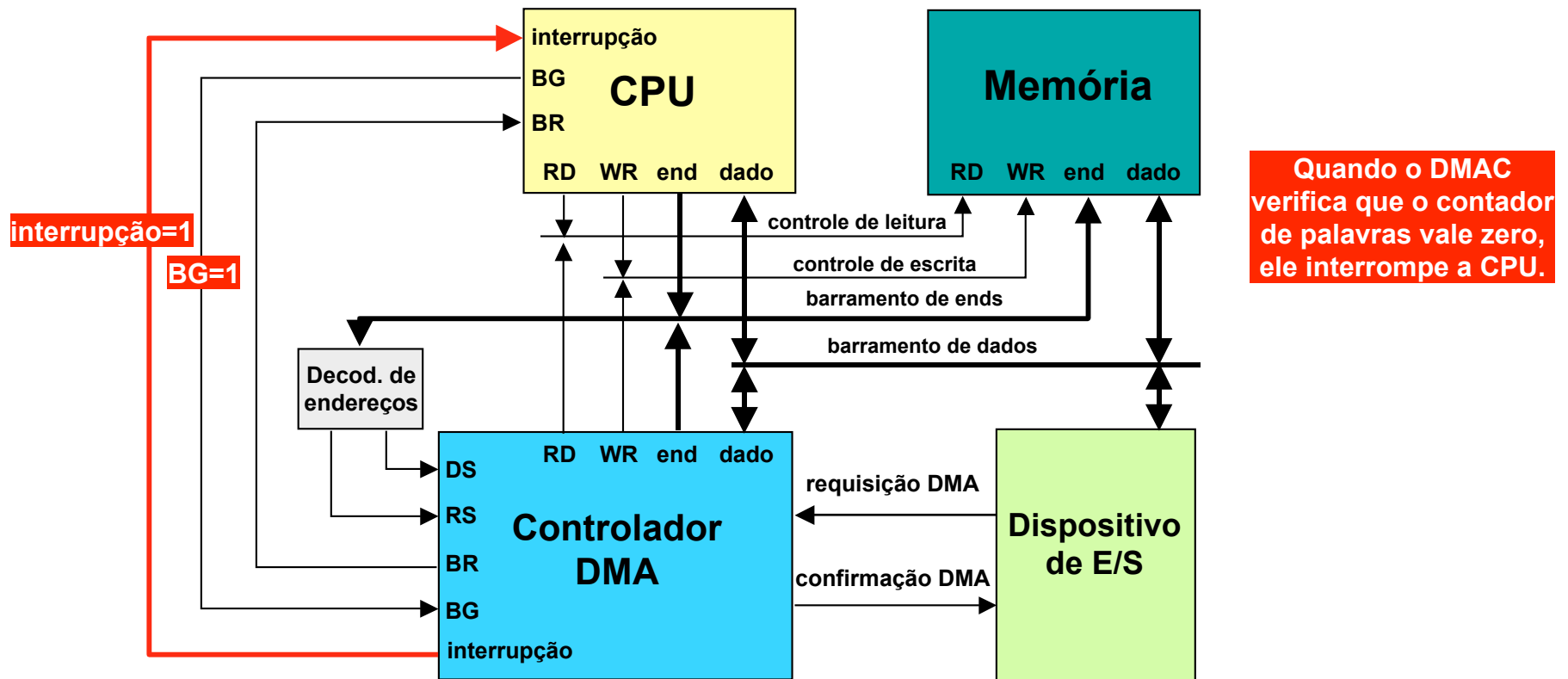
▶ Transferência DMA



O DMAC testa se o contador de palavras vale zero. Caso negativo, ele fica a espera do dispositivo de E/S sinalizar que está pronto para nova transferência.

4. Máquinas Seqüenciais Síncronas

▶ Transferência DMA



4. Máquinas Seqüenciais Síncronas

▶ Transferência DMA

