


Universidade Federal
de Santa Catarina

EEL7020 – Sistemas Digitais
Aula 14:
Projeto de Sistemas Digitais no Nível RT
III

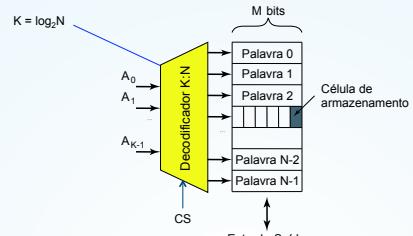
Prof. Djones Vinicius Lettnin
lettnin@eel.ufsc.br
<http://dlettnin.paginas.ufsc.br/>

Disclaimer: slides adapted for EEL7020 by D. Lettnin from the original slides made available by the author J. Guentzel.


Universidade Federal
de Santa Catarina

Memória

Organização de uma memória RAM



The diagram illustrates the organization of a RAM. An address bus of width $K = \log_2 N$ is decoded by a $K \times N$ decoder into N control lines (A_0, A_1, \dots, A_{K-1}). These lines select one of M bits in a row of memory cells. The row is identified by an address bus of width M bits. The output of the row is the **Entrada-Saída (M bits)**. A **Célula de armazenamento** is shown as a small grid of squares.

Fonte: RABAAY, Jan M. et al. Digital Integrated Circuits - A Design Perspective. 2nd edition (adaptado)
© J. Guentzel – Adapted by D. Lettnin


Universidade Federal
de Santa Catarina

Plano de Aula



- Estudo de caso : SAD (*Sum of Absolute Differences*)

3

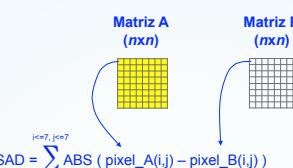

Universidade Federal
de Santa Catarina

SAD

Projetando um Sistema Digital

Exemplo 3: cálculo da SAD (*Sum of Absolute Differences*)

- A SAD é uma operação realizada sobre duas matrizes de pixels (A e B), gerando um valor único:



$$\text{SAD} = \sum_{i=0, j=0}^{i<7, j<7} \text{ABS} (\text{pixel_A}(i,j) - \text{pixel_B}(i,j))$$

© J. Guentzel – Adapted by D. Lettnin


Universidade Federal
de Santa Catarina

SAD

Projetando um Sistema Digital

Exemplo 3: cálculo da SAD (*Sum of Absolute Differences*)

- O valor calculado (SAD) é utilizado como uma medida do grau de semelhança entre as duas matrizes (e portanto, da semelhança entre as imagens por elas representadas): quanto menor for o valor “SAD” entre duas matrizes, mais semelhantes elas são.
- O objetivo deste exemplo é estudar sistemas digitais capazes de realizar o cálculo da SAD.
- A fim de contextualizar o exemplo, as próximas transparências abordam alguns dos princípios da compressão de vídeo digital

5

© J. Guentzel – Adapted by D. Lettnin


Universidade Federal
de Santa Catarina

SAD

Princípios de Vídeo Digital

- Os equipamentos (digitais ou analógicos) armazenam filmes como sequências de imagens estáticas (ou seja, sequências de fotos)
- Em um filme, cada imagem estática é chamada de “quadro” (frame, em inglês)
- Para que as transições entre as imagens estáticas não sejam percebidas pelo olho humano (de modo que enxerguemos um filme), é necessário que a taxa de exibição seja igual ou superior a 30 quadros por segundo (frames per second, ou simplesmente, fps)

6

© J. Guentzel – Adapted by D. Lettnin

SAD

Universidade Federal de Santa Catarina

► Princípios de Vídeo Digital

- A **qualidade da imagem** é diretamente proporcional ao **número de pixels de cada quadro**.
- Se não for usada alguma técnica de **compressão de vídeo**, **todos os pixels de todos os quadros precisarão ser armazenados**. Neste caso:
 - a **quantidade de memória** necessária para armazenar filmes (ou trechos de filmes) será enorme
 - a **taxa necessária para transmitir** um vídeo será enorme

© J. Guentzel – Adapted by D. Lettnin

7

SAD

Universidade Federal de Santa Catarina

► Princípios de Vídeo Digital

Seja um vídeo sem compressão, com 10 minutos de duração, **30 quadros por segundo**, sendo cada pixel armazenado com 24 bits:

Formato	Resolução	Memória para armazenar 10 minutos de vídeo
SDTV (e DVD)	720x480 pixels	19 GB
HDTV	1920x1080 pixels	112 GB

Formato	Resolução	Taxa requerida p/ transmitir 10 minutos de vídeo
SDTV (e DVD)	720x480 pixels	249 Mbps
HDTV	1920x1080 pixels	1,5 Gbps

Fonte: L. Agostini. Desenvolvimento de Arquiteturas de Alto Desempenho Dedicadas à Compressão de Vídeo Segundo o Padrão H.264/AVC. PPGC da UFRGS, 2007. Tese de doutorado.

© J. Guentzel – Adapted by D. Lettnin

8

SAD

Universidade Federal de Santa Catarina

► Princípio da Compressão de Vídeo

- Boa notícia!** Nos vídeos, normalmente quadros consecutivos são bastante semelhantes (Por que será?...)
- Logo, não é necessário armazenar todos os pixels de todos os quadros!**
- Basta que se busquem semelhanças
 - entre as regiões dentro de um quadro: esta é a chamada **previsão intraquadro (intraframe prediction)**
 - e/ou semelhanças entre quadros próximos: esta é a chamada **previsão interquadros (interframe prediction)**

Em vídeo, a previsão interquadros é responsável por altas taxas de compressão

© J. Guentzel – Adapted by D. Lettnin

9

SAD

Universidade Federal de Santa Catarina

► Previsão Interquadros

Princípio:

- De tantos em tantos quadros, somente **um quadro é armazenado completamente**. Ele recebe o nome de **quadro de referência (Qref)**.
- Os **demais quadros** que sucedem (e às vezes, alguns que antecedem também) são armazenados de maneira simplificada, por meio de “**vetores de movimento**”

© J. Guentzel – Adapted by D. Lettnin

10

SAD

Universidade Federal de Santa Catarina

► Previsão Interquadros

- Cada quadro é dividido em sub-matrizes de pixels, denominadas “**blocos**”. Exemplo hipotético:

Origem (0,0)

4 blocos

Cada bloco com 8x8 pixels

4 blocos

11

© J. Guentzel – Adapted by D. Lettnin

SAD

Universidade Federal de Santa Catarina

► Previsão Interquadros

- Um determinado quadro é escolhido para ser “referência”(Q_{ref}). Todos os pixels deste quadro serão armazenados.
- Para os quadros Q_i que sucedem Q_{ref} , serão calculados os “**vetores de movimento**” (pois somente estes serão armazenados).
- Assim, para cada bloco $b_j \in Q_i$ será encontrado um vetor de movimento da seguinte maneira:
 - É buscado o melhor “casamento” (*matching*) entre cada bloco $b_j \in Q_i$ e alguma porção (sub-matriz) de Q_{ref}
 - Encontrado o melhor casamento, o bloco de Q_i será representado como um par $\{x_i, y_i\}$ denominado de “**vetor de movimento**”, que aponta para a aresta superior esquerda da porção (sub-matriz) em Q_{ref} que contém os pixels que serão usados para representar o bloco $b_j (\in Q_i)$

© J. Guentzel – Adapted by D. Lettnin

12

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

13

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

14

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

15

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

16

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

17

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Comparando $b_j \in Q_i$ com bloco $\in Q_{ref}$ cuja origem é $x=0, y=0$

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

18

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Comparando $b_j \in Q_i$ com bloco $\in Q_{ref}$ cuja origem é $x=1, y=0$

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

19

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Comparando $b_j \in Q_i$ com bloco $\in Q_{ref}$ cuja origem é $x=2, y=0$

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

20

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Comparando $b_j \in Q_i$ com bloco $\in Q_{ref}$ cuja origem é $x=3, y=0$

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

21

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Comparando $b_j \in Q_i$ com bloco $\in Q_{ref}$ cuja origem é $x=4, y=0$

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

22

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

Melhor casamento! Logo, vetor de movimento: $x=5, y=0$

Q_{ref}

Q_i

© J. Guentzel – Adapted by D. Lettmann

23

SAD

Universidade Federal de Santa Catarina

► Etapa de “Estimação de Movimento”

(x0,y0)	(x1=5, y1=0)	(x2,y2)	(x3,y3)
(x4,y4)	(x5,y5)	(x6,y6)	(x7,y7)
(x8,y8)	(x9,y9)	(x10,y10)	(x11,y11)
(x12,y12)	(x13,y13)	(x14,y14)	(x15,y15)

Q_{ref}

Q_i

Cada bloco de $b_j \in Q_i$ será representado por um vetor de movimento (ao invés 8x8 pixels)

© J. Guentzel – Adapted by D. Lettmann

24

The diagram shows two 4x4 pixel image blocks, Q_{ref} and Q_i , side-by-side. Each block contains a house, a sun, and a cloud. The pixels are color-coded: green for grass, blue for sky, yellow for the house, orange for the roof, and white for the sun and cloud. The blocks are overlaid with a grid. The top-left pixel of each block is highlighted in red, representing the reference pixel used for comparison. The text "SAD" is displayed at the top right, and "Etapa de ‘Estimação de Movimento’" is displayed below the first block.

SAD

Universidade Federal de Santa Catarina

Etapa de “Estimação de Movimento”

Cada vez que se compara um bloco $b_j \in Q_i$ com um bloco de Q_{ref} é preciso calcular a SAD entre duas matrizes de pixels

Q_{ref}

Q_i

SAD

Cálculo do SAD (*Sum of Absolute Differences*)

Cada vez que se compara um bloco $b_j \in Q_i$ com uma porção de Q_{ref} é preciso calcular a SAD entre duas matrizes

Matriz A:
Porção de 8x8 pixels
 $\in Q_{ref}$

Matriz B:
 $b_j \in Q_i$

$SAD = \sum_{i=0, j=0}^{i=7, j=7} ABS(pixel_A(i,j) - pixel_B(i,j))$

Calcular para cada par de pixels $(pixel_A(i,j), pixel_B(i,j))$ com $i \in [0,7]$ e $j \in [0,7]$

SAD

Universidade Federal de Santa Catarina

Projetando um Sistema Digital

Exemplo 3: cálculo da SAD (*Sum of Absolute Differences*)

Projetar um sistema digital capaz de realizar o cálculo da SAD entre um bloco $b_j \in Q_i$ e uma porção de pixels (de mesmo tamanho) $\in Q_{ref}$.

- Cada pixel é representado por 1 byte (8 bits)
- O bloco b_j e a porção de Q_i tem 8x8 pixels cada e (já) estão armazenados nas memórias B e A, respectivamente.

Diagrama de bloco do sistema SAD:

```

    graph LR
        subgraph SA [Sistema SAD]
            SA[Contém 8x8 pixels ∈ Qref] --> MA[Memória A]
            SA --> MB[Memória B]
            MA --> P_A["P_A"]
            MB --> P_B["P_B"]
            P_A --> C[Cálculo do SAD]
            P_B --> C
            C --> SAD[SAD]
            C --> PR[pronto]
        end
        CK[ck] --> C
        INICIO[inicio] --> C
        RESET[reset] --> C
        SAD --> PR
    
```

O bloco SAD contém 8x8 pixels de referência (Q_{ref}). Ele lê os dados de duas memórias: Memória A (que contém o bloco b_j) e Memória B (que contém a porção de Q_i). Os dados são lidos em sequência (read), com endereço (end) e controladores (P_A, P_B). O resultado é o valor da SAD, que é enviado para o bloco pronto.

© J. Guentzel – Adaptado by D. Letzkin

SAD

Universidade Federal de Santa Catarina

Projetando um Sistema Digital

Exemplo 3: cálculo da SAD

- O sinal $\text{inicio}=1$ indica que um cálculo de SAD deve iniciar.
- Funcionamento de cada memória: um pixel pode ser lido a cada ciclo de relógio; basta atualizar o “end” e manter “read=1”.
- O projeto deste sistema digital deve favorecer a otimização do custo.

SAD

Universidade Federal de Santa Catarina

Projetando um Sistema Digital

Exemplo 3: cálculo da SAD

Possível Organização da “Memória A” e da “Memória B”

Matriz A:
Porção de 8x8 pixels
 $\in Q_{\text{ref}}$

Matriz B:
 $b_j \in Q_i$

$\sum_{i=0}^{i=7} \sum_{j=0}^{j=7} \text{ABS}(\text{pixel}_A(i,j) - \text{pixel}_B(i,j))$

Calcular para cada par de pixels
 $\{ \text{pixel}_A(i,j), \text{pixel}_B(i,j) \}$ com
 $i \in [0,7]$ e $j \in [0,7]$

SAD

Universidade Federal de Santa Catarina

Projetando um Sistema Digital

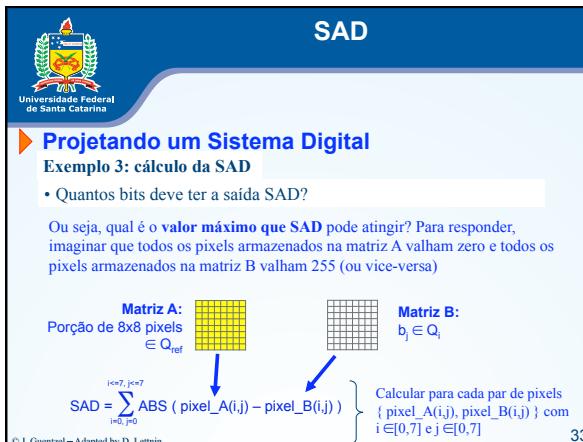
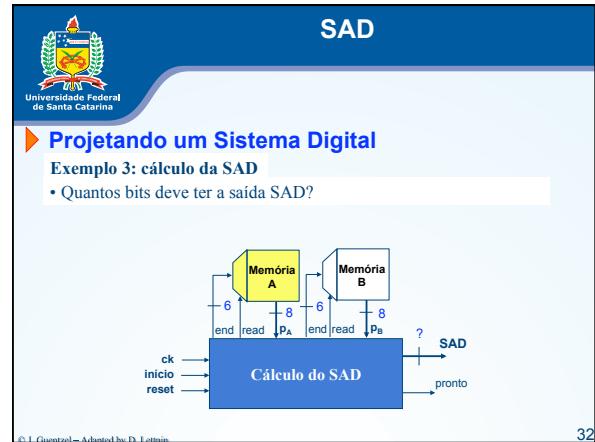
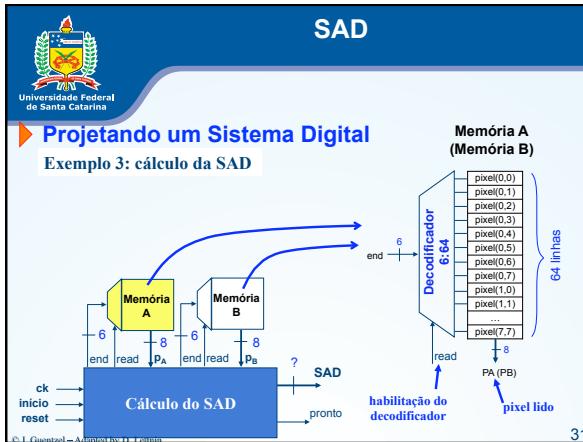
Exemplo 3: cálculo da SAD

Possível Organização da “Memória A” e da “Memória B”

Memória A	Memória B
pixel(0,0)	pixel(0,0)
pixel(0,1)	pixel(0,1)
pixel(0,2)	pixel(0,2)
pixel(0,3)	pixel(0,3)
pixel(0,4)	pixel(0,4)
pixel(0,5)	pixel(0,5)
pixel(0,6)	pixel(0,6)
pixel(0,7)	pixel(0,7)
pixel(1,0)	pixel(1,0)
pixel(1,1)	pixel(1,1)
...	...
pixel(7,7)	pixel(7,7)

Matriz A:
Porção de 8x8 pixels $\in Q_{ref}$

Matriz B: $b_j \in Q_i$



SAD

Projetando um Sistema Digital

Exemplo 3: Passo 1 (captura do comportamento com FSMD)

```

    graph TD
        subgraph "Cálculo do SAD"
            direction TB
            CA[Memória A] --> CS[Cálculo do SAD]
            CB[Memória B] --> CS
            CS -- pronto --> S0
            CS -- SAD --> S4
        end
        S0((S0)) -- inicio --> S1((S1))
        S0 -- read --> S1
        S1 -- "i >= 64" --> S2((S2))
        S2 -- "i <- i + 1" --> S3((S3))
        S3 -- "soma <- soma + ABS(pA - pB)" --> S4((S4))
        S4 -- "SAD reg <- soma" --> S4
        S4 -- "read <- 0" --> S4
    
```

Início
pronto <- 0; soma <- 0; i <- 0;
Enquanto $i \geq 64$ faça
{
 soma <- soma + ABS(pA - pB);
 i <- i + 1;
}
SAD reg <- soma; pronto <- 1;
Fim

© I. Guantzel - Adaptado by D. Lettmin

37

SAD

Projetando um Sistema Digital

Exemplo 3: Passo 2 (projeto do BO)

```

    graph TD
        subgraph "Cálculo do SAD"
            direction TB
            CA[Memória A] --> CS[Cálculo do SAD]
            CB[Memória B] --> CS
            CS -- pronto --> S0
            CS -- SAD --> S4
        end
        S0((S0)) -- inicio --> S1((S1))
        S0 -- read --> S1
        S1 -- "i >= 64" --> S2((S2))
        S2 -- "i <- i + 1" --> S3((S3))
        S3 -- "soma <- soma + ABS(pA - pB)" --> S4((S4))
        S4 -- "SAD reg <- soma" --> S4
        S4 -- "read <- 0" --> S4
    
```

Quais variáveis são usadas para armazenar dados?
• Duas: "soma" e "SAD_reg" ("pA" e "pB" são entradas que ficam estáveis, já que provem das memória A e B)
• Logo, teremos dois registradores: "soma" e "SAD_reg"

© I. Guantzel - Adaptado by D. Lettmin

38

SAD

Projetando um Sistema Digital

Exemplo 3: Passo 2 (projeto do BO)

```

    graph TD
        subgraph "Cálculo do SAD"
            direction TB
            CA[Memória A] --> CS[Cálculo do SAD]
            CB[Memória B] --> CS
            CS -- pronto --> S0
            CS -- SAD --> S4
        end
        S0((S0)) -- inicio --> S1((S1))
        S0 -- read --> S1
        S1 -- "i >= 64" --> S2((S2))
        S2 -- "i <- i + 1" --> S3((S3))
        S3 -- "soma <- soma + ABS(pA - pB)" --> S4((S4))
        S4 -- "SAD reg <- soma" --> S4
        S4 -- "read <- 0" --> S4
    
```

Note: que há uma variável ("i") que armazena o endereço a ser acessado nas memórias e também serve para controlar o laço. Logo, será preciso um registrador (decrementador) para esta variável. Chamemo-lo de "i".

© I. Guantzel - Adaptado by D. Lettmin

39

SAD

Projetando um Sistema Digital

Exemplo 3: Passo 2 (projeto do BO)

```

    graph TD
        subgraph "Cálculo do SAD"
            direction TB
            CA[Memória A] --> CS[Cálculo do SAD]
            CB[Memória B] --> CS
            CS -- pronto --> S0
            CS -- SAD --> S4
        end
        S0((S0)) -- inicio --> S1((S1))
        S0 -- read --> S1
        S1 -- "i >= 64" --> S2((S2))
        S2 -- "i <- i + 1" --> S3((S3))
        S3 -- "soma <- soma + ABS(pA - pB)" --> S4((S4))
        S4 -- "SAD reg <- soma" --> S4
        S4 -- "read <- 0" --> S4
    
```

Quais operações são realizadas sobre dados?
• Dentro do laço há: uma subtração, uma extração de módulo e uma adição.

© I. Guantzel - Adaptado by D. Lettmin

40

SAD

Projetando um Sistema Digital

Exemplo 3: Passo 2 (projeto do BO)

```

    graph TD
        subgraph "Cálculo do SAD"
            direction TB
            CA[Memória A] --> CS[Cálculo do SAD]
            CB[Memória B] --> CS
            CS -- pronto --> S0
            CS -- SAD --> S4
        end
        S0((S0)) -- inicio --> S1((S1))
        S0 -- read --> S1
        S1 -- "i >= 64" --> S2((S2))
        S2 -- "i <- i + 1" --> S3((S3))
        S3 -- "soma <- soma + ABS(pA - pB)" --> S4((S4))
        S4 -- "SAD reg <- soma" --> S4
        S4 -- "read <- 0" --> S4
    
```

Quais operações são realizadas sobre variáveis de controle?
• Dentro do laço há: um incremento sobre "i".
• Também há comparação com 64

© I. Guantzel - Adaptado by D. Lettmin

41

SAD

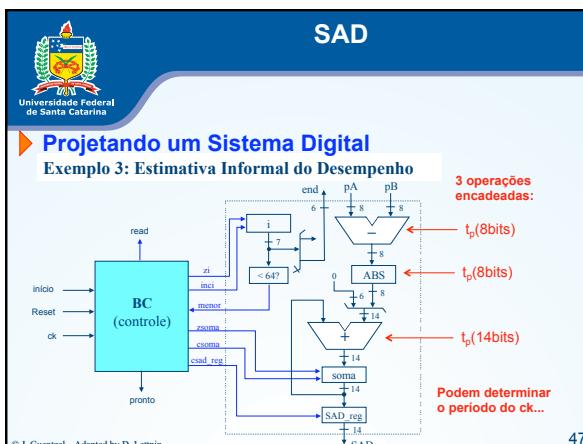
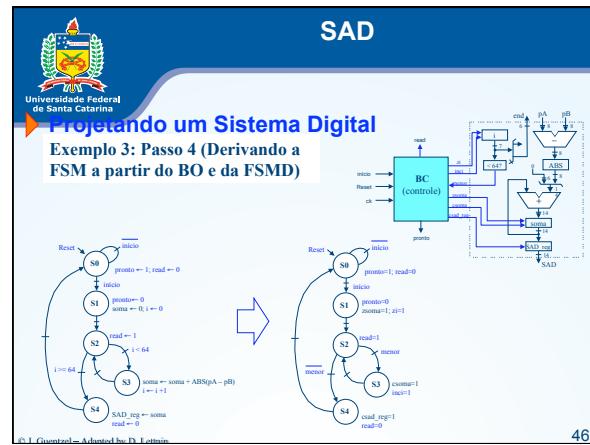
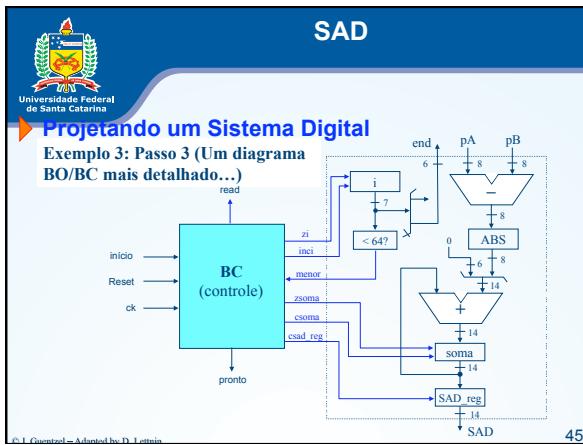
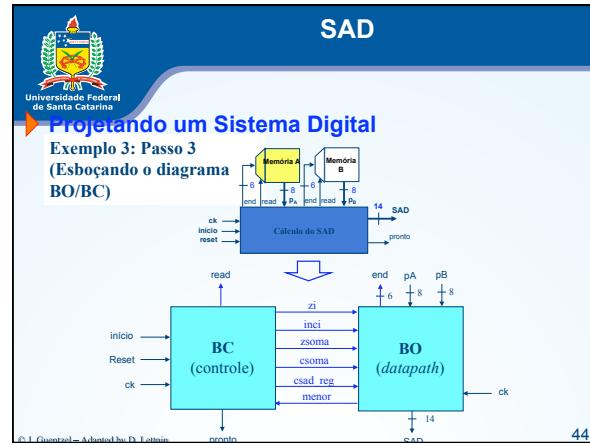
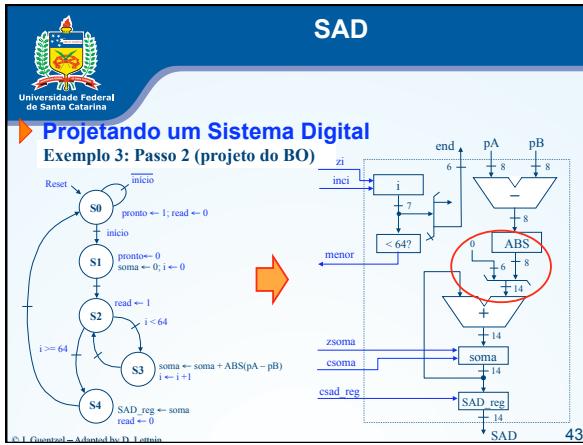
Projetando um Sistema Digital

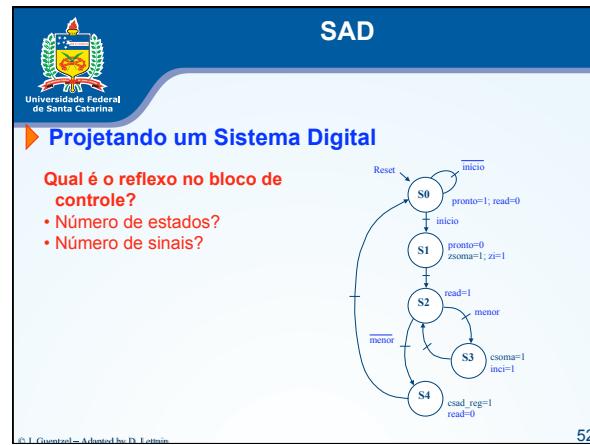
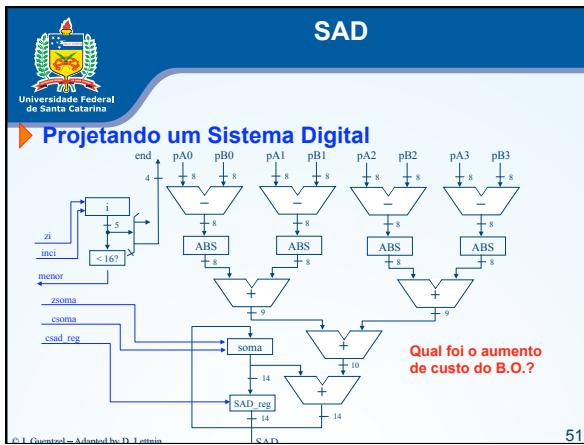
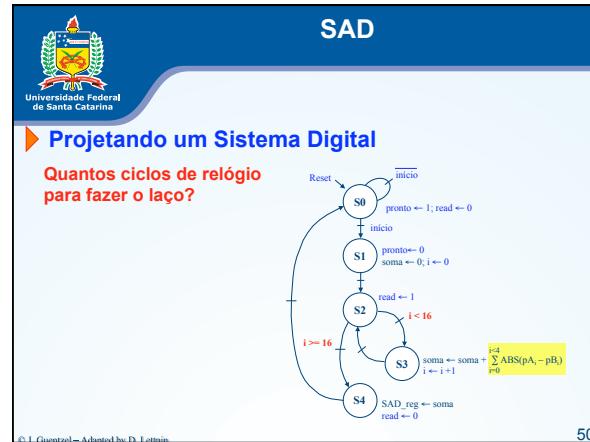
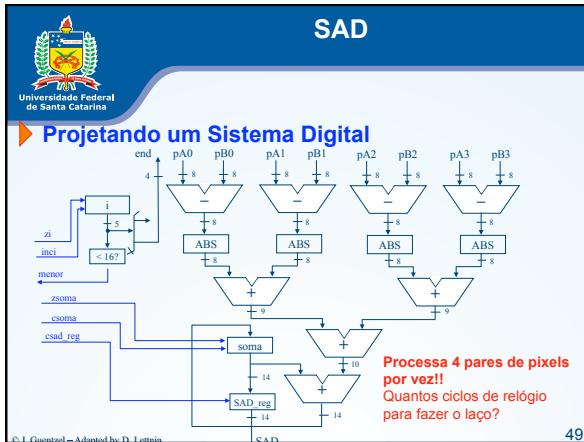
Exemplo 3: Passo 2 (projeto do BO)

Detailed block diagram showing the internal logic of the 'Cálculo do SAD' block. It includes a subtractor (pA - pB), an absolute value block (ABS), a multiplier (8), and adder blocks for 'soma' and 'csoma'. Control signals include 'end', 'read', 'pA', 'pB', 'menor', 'zSoma', 'csoma', and 'csad reg'.

© I. Guantzel - Adaptado by D. Lettmin

42





EEL7020 – Sistemas Digitais
Aula 14:
Projeto de Sistemas Digitais no Nível RT
III

Prof. Djones Vinicius Lettnin
lettnin@eel.ufsc.br
<http://dlettnin.paginas.ufsc.br/>

Disclaimer: slides adapted for EEL7020 by D. Lettnin from the original slides made available by the author J. Guentzel.