

 Universidade Federal de Santa Catarina

EEL7020 – Sistemas Digitais

Aula 11: Circuitos sequenciais – Vending Machine e datapath

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 authors J. Guentzel, E. Batista e Vahid.

 Universidade Federal de Santa Catarina

Codificação de Estados

Codificação de Estados

Assinalamento A=00, B=01, C=10

$$\begin{aligned} Y1 &= w \cdot (y_1 + y_0) \\ Y0 &= w \cdot y_1 \cdot y_0 \\ z &= y_1 \end{aligned}$$

Assinalamento A=00, B=01, C=11 (Código Gray)

$$\begin{aligned} Y1 &= \bar{w} \cdot w \\ Y0 &= w \\ z &= y_1 \end{aligned}$$

Codificação “One Hot”

$$\begin{aligned} Y2 &= \bar{y}_0 \cdot w \\ Y1 &= \bar{y}_0 \cdot w \\ Y0 &= w \\ z &= y_2 \end{aligned}$$

© J. Guentzel – Adapted by D. Lettnin

2

 Universidade Federal de Santa Catarina

Plano de Aula

- Estudo de caso
 - *Vending machine*
- Datapath*



3

 Universidade Federal de Santa Catarina

Estudo de Caso 3

Exemplo 6: a máquina de vendas (*vending machine*)

Projetar o bloco de controle (FSM) de uma máquina automática de vendas. As principais características da máquina são:

- Vende somente um tipo de produto, doravante referenciado por “item”, cujo preço é R\$ 1,50.
- Aceita somente moedas de R\$ 0,50 e de R\$ 1,00.
- Não fornece troco.
- Assim que o montante inserido na máquina atingir (ou ultrapassar) R\$ 1,50, a máquina libera um item.

© J. Guentzel – Adapted by D. Lettnin

4

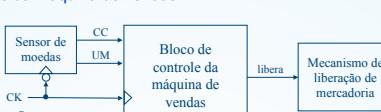
 Universidade Federal de Santa Catarina

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: a máquina de vendas (*vending machine*)

Interfaces da máquina de vendas



CC = moeda de R\$ 0,50 detectada
UM = moeda de R\$ 1,00 detectada

© J. Guentzel – Adapted by D. Lettnin

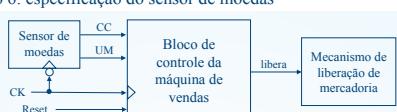
5

 Universidade Federal de Santa Catarina

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: especificação do sensor de moedas



- Sensor de moedas é sincronizado pela borda de descida de CK.
- Quando uma moeda de R\$ 0,50 é inserida, CC=1 durante um ciclo de relógio.
- Quando uma moeda de R\$ 1,00 é inserida, UM=1 durante um ciclo de relógio.
- Quando uma moeda diferente é inserida ela cai de volta (e CC=0 & UM=0).

© J. Guentzel – Adapted by D. Lettnin

6

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: especificação do mecanismo de liberação de mercadoria

- Para liberar um item, basta que libera=1 durante um ciclo de relógio (CK).
- O item liberado cai em uma cesta.

© J. Guentzel – Adapted by D. Letmin

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: mais uma assertiva

- No início de uma operação de venda a máquina se encontra em um estado inicial (S_0), que corresponde ao seu estado de reset.

© J. Guentzel – Adapted by D. Letmin

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: representando o comportamento

- Uma vez compreendido o comportamento, deve-se mapeá-lo para alguma forma de representação mais adequada à síntese da FSM.
- Porém, às vezes pode ser mais seguro iniciar **listando-se todas as sequências de entradas ou de "configurações"** que o sistema pode assumir. No caso em questão, a listagem de todas as sequências de moedas possíveis é perfeitamente factível:

Sequência de moedas	Sequência de sinais
0,50; 0,50; 0,50	CC, CC, CC
0,50; 0,50; 1,00	CC, CC, UM
0,50; 1,00	CC, UM
1,00; 0,50	UM, CC
1,00; 1,00	UM, UM

© J. Guentzel – Adapted by D. Letmin

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: representando o comportamento (diagrama de estados)

- Representando graficamente todas as sequências possíveis
- Neste diagrama de estados não há arestas com destino igual a origem!
- Ele pode ser simplificado, pois há excesso de estados...

© J. Guentzel – Adapted by D. Letmin

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: representando o comportamento (diagrama de estados)

- Os estados S_4 , S_5 , S_6 , S_7 e S_8 têm a mesma função (abrir o dispensador) e portanto, podem ser combinados em um único estado.
- Para reduzir ainda mais o número de estados, podemos imaginar que cada estado represente o **total de dinheiro recebido** pela máquina até um dado momento. (E neste caso, não importa se o total de R\$ 1,00 foi atingido pela inserção de uma moeda de R\$ 1,00 ou por duas moedas de R\$ 0,50...)

© J. Guentzel – Adapted by D. Letmin

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: minimização de estados

Sequências possíveis:

- CC, CC, CC
- CC, CC, UM
- CC, UM
- UM, CC
- UM, UM

- E se o comprador demorar para inserir a segunda moeda?
- O que ocorre após o item ser liberado?

© J. Guentzel – Adapted by D. Letmin

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: uma representação mais completa

- E se o comprador demorar para inserir a segunda moeda?
- Resp.: a máquina deve ficar parada no estado R\$0,50 ou R\$1,00.
- O que ocorre após o item ser liberado?
- Resp.: a máquina deve ir para um estado no qual esteja disponível para nova venda.

© J. Guentzel – Adapted by D. Lettin
13

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: uma representação mais completa

- E se uma terceira moeda for inserida quando a máquina estiver no estado R \$1,50?

© J. Guentzel – Adapted by D. Lettin
14

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: uma representação mais completa

- E se uma terceira moeda for inserida quando a máquina estiver no estado R\$1,50?
- Resp.: considerar o retorno para o respectivo estado (R\$0,50 ou R \$1,00). Mas será que tal problema existe na prática?

© J. Guentzel – Adapted by D. Lettin
15

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: uma representação mais completa

- O comprador sabe quando inseriu moedas suficientes (p. ex., o sinal liberar pode fazer soar um aviso e/ou acender uma luz).
- O tempo entre a inserção de duas moedas consecutivas é muito maior do que o período do sinal de relógio CK. (Um sinal de relógio geralmente tem período da ordem de kHz ou MHz, o ser humano trabalha na frequência de Hz = 1/1s).
- Logo, a solução ao lado é realista!

© J. Guentzel – Adapted by D. Lettin
16

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: projeto usando Modelo de Moore

© J. Guentzel – Adapted by D. Lettin
17

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: tabela de transição de estados e tabela de saída

Estado atual	Entradas	Próximo estado	Saída (Moore)
	UM CC		libera
R\$ 0	0 0	R\$ 0	0
	0 1	R\$ 0,50	0
	1 0	R\$ 1,00	0
	1 1	-	-
R\$ 0,50	0 0	R\$ 0,50	0
	0 1	R\$ 1,00	0
	1 0	R\$ 1,50	0
	1 1	-	-
R\$ 1,00	0 0	R\$ 1,00	0
	0 1	R\$ 1,50	0
	1 0	R\$ 1,50	0
	1 1	-	-
R\$ 1,50	0 0	R\$ 0	1
	0 1	R\$ 0	1
	1 0	R\$ 0	1
	1 1	-	-

© J. Guentzel – Adapted by D. Lettin
18

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: tabela de transição de estados e tabela de saída

Supondo a seguinte codificação de estados:

- (A) RS 0 → 00
- (B) RS 0,50 → 01
- (C) RS 1,00 → 10
- (D) RS 1,50 → 11

	Estado atual	Entradas	Próximo estado	Saída (Moore)
A	y1y0	UM CC	Y1Y0	libera
	0 0	0 0	0 0	0
	0 1	0 1	0 1	0
	1 0	0 1	1 0	0
	1 1	-	-	-
B	0 1	0 0	0 1	0
	0 1	0 1	1 0	0
	1 0	0 1	0 1	0
	1 1	-	-	-
C	1 0	0 0	1 0	0
	0 1	0 1	1 1	0
	1 0	0 1	0 1	0
	1 1	-	-	-
D	1 1	0 0	0 0	1
	0 1	0 0	0 0	1
	1 0	0 0	0 0	1
	1 1	-	-	-

© J. Guentzel – Adapted by D. Lettmn

19

Estudo de Caso 3

Mapa de Karnaugh

Estado atual	Entradas	Próximo estado	Saída
A	y1y0	UM CC	Y1Y0
	0 0	0 0	0 0
	0 1	0 1	0 1
	1 0	1 0	1 0
	1 1	-	-
B	0 1	0 0	0 1
	0 1	0 1	1 0
	1 0	0 1	0 1
	1 1	-	-
C	1 0	0 0	1 0
	0 1	0 1	1 1
	1 0	0 1	0 1
	1 1	-	-
D	1 1	0 0	0 0
	0 1	0 0	0 0
	1 0	0 0	0 0
	1 1	-	-

$Y1 = y1 \cdot y0' + y1' \cdot UM + y1' \cdot y0 \cdot CC$

Custo = ?

© J. Guentzel – Adapted by D. Lettmn

20

Estudo de Caso 3

Mapa de Karnaugh

Estado atual	Entradas	Próximo estado	Saída (Moore)
A	y1y0	UM CC	Y1Y0
	0 0	0 0	0 0
	0 1	0 1	0 1
	1 0	1 0	0 0
	1 1	-	-
B	0 1	0 0	0 1
	0 1	0 1	1 0
	1 0	0 1	0 1
	1 1	-	-
C	1 0	0 0	1 0
	0 1	0 1	1 1
	1 0	0 1	0 1
	1 1	-	-
D	1 1	0 0	0 0
	0 1	0 0	0 0
	1 0	0 0	0 0
	1 1	-	-

$Y0 = y1' \cdot y0 \cdot CC + y0' \cdot CC + y1 \cdot y0 \cdot UM$

Custo = ?

© J. Guentzel – Adapted by D. Lettmn

21

Estudo de Caso 3

Síntese de Circuitos Sequenciais

Exemplo 6: tabela de transição de estados e tabela de saída

Codificação de estados usando código Gray:

- (A) RS 0 → 00
- (B) RS 0,50 → 01
- (D) RS 1,00 → 10
- (C) RS 1,50 → 11

Estado atual	UM	CC	Próx. estado	(saída) Lib
A	0 0	0 0	0 0	0 0
	0 1	0 1	0 1	0 1
	1 0	1 1	1 1	1 0
	1 1	-	-	-
B	0 1	0 0	0 1	0
	0 1	0 1	1 0	0
	1 0	1 1	0 1	0
	1 1	-	-	-
C	1 1	0 0	0 1	1 1
	0 1	0 1	1 0	0
	1 0	0 1	0 1	1
	1 1	-	-	-
D	1 0	0 0	0 0	0 0
	0 1	0 0	0 0	0 0
	1 0	0 0	0 0	0 0
	1 1	-	-	-

$Y1 = y1 \cdot y0 + !y1 \cdot UM + y0 \cdot CC$

© J. Guentzel – Adapted by D. Lettmn

22

Estudo de Caso 3

Mapa de Karnaugh

Estado atual	UM	CC	Próx. estado	(saída) Lib
A	y1y0	UM CC	Y1 Y0	
	0 0	0 0	0 0	0
	0 1	0 1	0 1	0
	1 0	1 1	1 0	0
	1 1	-	-	-
B	0 1	0 0	0 1	0
	0 1	0 1	1 1	0
	1 0	1 1	0 1	0
	1 1	-	-	-
C	1 1	0 0	0 1	1 1
	0 1	0 1	1 0	0
	1 0	0 1	0 1	1
	1 1	-	-	-
D	1 0	0 0	0 0	0
	0 1	0 0	0 0	0
	1 0	0 0	0 0	0
	1 1	-	-	-

$Y1 = y1 \cdot y0 + !y1 \cdot UM + y0 \cdot CC$

Custo = ?

© J. Guentzel – Adapted by D. Lettmn

23

Estudo de Caso 3

Mapa de Karnaugh

Estado atual	UM	CC	Próx. estado	(saída) Lib
A	y1y0	UM CC	Y1 Y0	
	0 0	0 0	0 0	0
	0 1	0 1	0 1	0
	1 0	1 1	1 0	0
	1 1	-	-	-
B	0 1	0 0	0 0	0
	0 1	0 1	1 1	0
	1 0	1 1	0 1	0
	1 1	-	-	-
C	1 1	0 0	0 1	1 1
	0 1	0 1	1 0	0
	1 0	0 1	0 1	1
	1 1	-	-	-
D	1 0	0 0	0 0	0
	0 1	0 0	0 0	0
	1 0	0 0	0 0	0
	1 1	-	-	-

$Y0 = !y1 \cdot CC + !y1 \cdot y0 \cdot UM + y0 \cdot !UM \cdot !CC$

Custo = ?

© J. Guentzel – Adapted by D. Lettmn

24

Estudo de Caso 3

Universidade Federal de Santa Catarina

- Codificação 1: Y0 tem custo = 10, Y1 tem custo = 11 => custo total = 21
- Codificação 2: Y0 tem custo = 9, Y1 tem custo = 11 => custo total = 20
- Custo do registrador de estados é o mesmo, pois em ambos os casos são duas variáveis de estados
- Custo da saída Lib é igual para ambos os casos
- Logo, a codificação 2 (código de Gray) resulta em uma FSM com menor custo.

© J. Guentzel – Adapted by D. Lettin

25

Plano de Aula

Universidade Federal de Santa Catarina

- Estudo de caso
 - Vending machine
- Datapath

26

O Modelo Bloco Operativo / Bloco de Controle

Universidade Federal de Santa Catarina

Entradas (de dados) → Bloco Operativo (datapath) → Saídas (de dados)

Entradas (de controle) → Bloco de Controle (control) → Saídas (de controle)

comandos → Bloco Operativo (datapath)

status → Bloco de Controle (control)

© J. Guentzel – Adapted by D. Lettin

27

O Modelo Bloco Operativo / Bloco de Controle

Universidade Federal de Santa Catarina

Entradas (de dados) → Bloco Operativo (datapath) → Saídas (de dados)

Entradas (de controle) → Bloco de Controle (control) → Saídas (de controle)

comandos → Bloco Operativo (datapath)

status → Bloco de Controle (control)

Bloco Operativo:

- Realiza transformações sobre dados, geralmente provenientes do ambiente externo
- As transformações são realizadas em um ou mais passos, cada passo demorando um ciclo de relógio
- Gera sinais de "status" que são usados pelo Bloco de Controle para definir a sequência de operações a serem realizadas (às vezes são chamados de "flags")

© J. Guentzel – Adapted by D. Lettin

28

Registradores

Universidade Federal de Santa Catarina

Registradores

Registrador com carga paralela (versão 1)

entradas individuais

sinal de carga (relógio)

saídas individuais

© J. Guentzel – Adapted by D. Lettin

29

Registradores

Universidade Federal de Santa Catarina

Registrador com carga paralela (versão 2)

carga CK

sinal de carga separado do relógio

símbolo

carga

Reg

© J. Guentzel – Adapted by D. Lettin

30

Registradores

Registrador com carga paralela (versão 2) Exemplo 5.1

© J. Guentzel – Adapted by D. Lettmann

31

Registradores de Deslocamento

© E. Batista – Adapted by D. Lettmann

32

Registradores de Deslocamento

- Deslocamento Serial:

© E. Batista – Adapted by D. Lettmann

33

Registradores de Deslocamento

- Deslocamento Serial:

© E. Batista – Adapted by D. Lettmann

34

Registradores de Deslocamento

- Deslocamento Serial:

© E. Batista – Adapted by D. Lettmann

35

Registradores de Deslocamento

- Deslocamento Serial:

© E. Batista – Adapted by D. Lettmann

36

Registradores de Deslocamento

Universidade Federal de Santa Catarina

- Deslocamento Serial:

Entrada serial → 1 → 0 → 0 → 1 → 0 → 0 → 0 → 0 → Saída serial

© E. Batista – Adapted by D. Lettmann 37

Registradores de Deslocamento

Universidade Federal de Santa Catarina

- Deslocamento Serial:

Entrada serial → 0 → 1 → 1 → 0 → 0 → 1 → 0 → 0 → Saída serial

© E. Batista – Adapted by D. Lettmann 38

Registradores de Deslocamento

Universidade Federal de Santa Catarina

- Deslocamento Serial:

Entrada serial → 0 → 0 → 1 → 1 → 0 → 0 → 1 → 0 → 0 → Saída serial

© E. Batista – Adapted by D. Lettmann 39

Registradores de Deslocamento

Universidade Federal de Santa Catarina

- Deslocamento Serial:

Entrada serial → 0 → 0 → 0 → 1 → 1 → 0 → 0 → 1 → 0 → Saída serial

© E. Batista – Adapted by D. Lettmann 40

Registradores de Deslocamento

Universidade Federal de Santa Catarina

- Deslocamento Paralelo:

Entrada serial → 0 → 1 → 0 → 1 → 0 → 0 → 0 → 1 → Saída serial

© E. Batista – Adapted by D. Lettmann 41

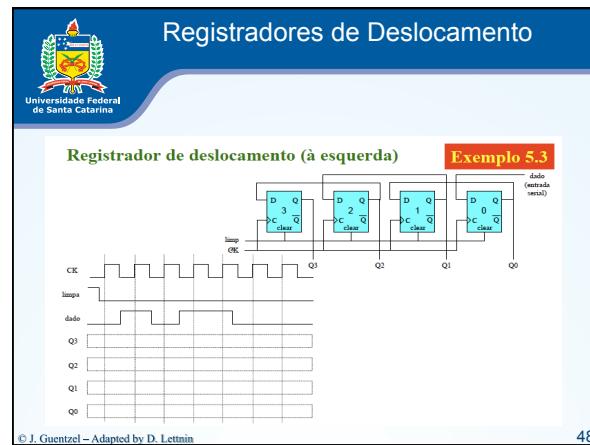
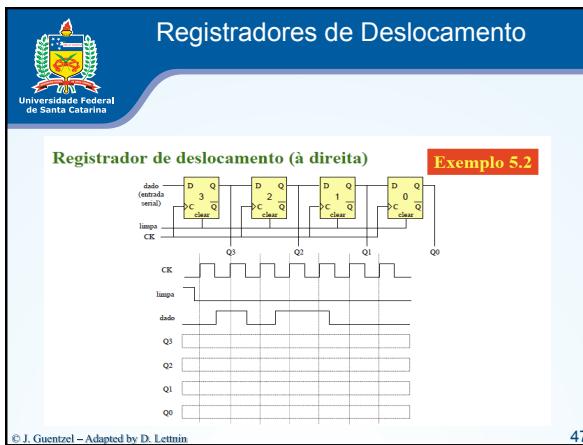
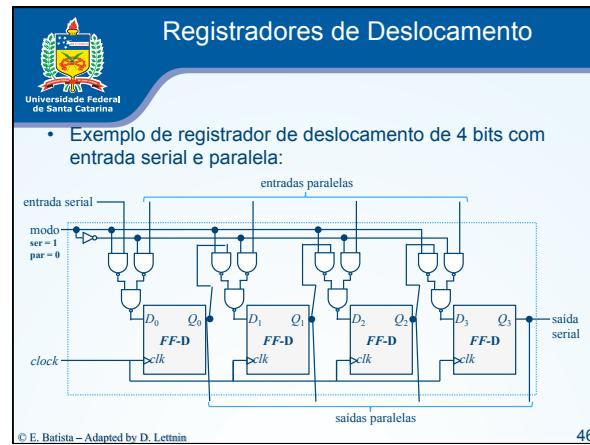
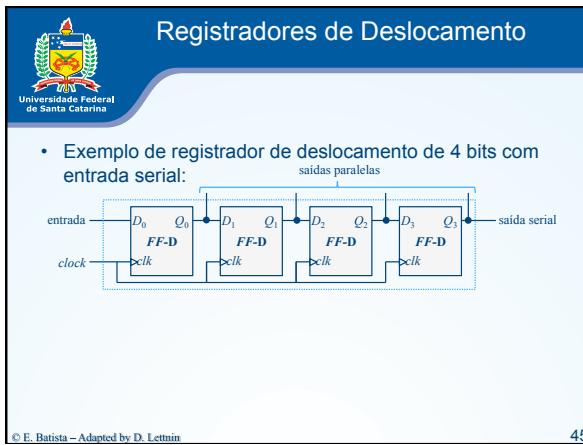
Registradores de Deslocamento

Universidade Federal de Santa Catarina

- Deslocamento Paralelo:

Entrada serial → 0 → 1 → 0 → 1 → 1 → 0 → 0 → 1 → Saída serial

© E. Batista – Adapted by D. Lettmann 42



Registradores de Deslocamento

Registrador de deslocamento com carga paralela

clear	CK	C1	C0	operação
0	$\Rightarrow \uparrow$	X	X	mantém conteúdo
0	\uparrow	0	0	mantém conteúdo
0	\uparrow	0	1	desloca à esquerda
0	\uparrow	1	0	desloca à direita
0	\uparrow	1	1	carga paralela
1	X	X	X	zera conteúdo

© J. Guentzel – Adapted by D. Lettmann

49

Counters

4.6

- N-bit up-counter:** N-bit register that can increment (add 1) to its own value on each clock cycle
 - 0000, 0001, 0010, 0011, ..., 1110, 1111, 0000
 - Note how count “rolls over” from 1111 to 0000
 - Terminal (last) count, tc, equals 1 during value just before rollover
- Internal design**
 - Register, incrementer, and N-input AND gate to detect terminal count

© Vahid – Adapted by D. Lettmann

50

Down-Counter

Universidade Federal de Santa Catarina

- 4-bit down-counter**
 - 1111, 1110, 1101, 1100, ..., 0011, 0010, 0001, 0000, 1111, ...
 - Terminal count is 0000
 - Use NOR gate to detect
 - Need decrementer (-1) – design like designed incrementer

© Vahid – Adapted by D. Lettmann

51

Up/Down-Counter

Universidade Federal de Santa Catarina

- Can count either up or down**
 - Includes both incrementer and decrementer
 - Use dir input to select, using 2x1: dir=0 means up
 - Likewise, dir selects appropriate terminal count value

© Vahid – Adapted by D. Lettmann

52

Comparators

4.5

N-bit equality comparator: Outputs 1 if two N-bit numbers are equal

- 4-bit equality comparator with inputs A and B
 - a3 must equal b3, a2 = b2, a1 = b1, a0 = b0
 - Two bits are equal if both 1, or both 0
 - $eq = (a3b3 + a3'b3') * (a2b2 + a2'b2') * (a1b1 + a1'b1') * (a0b0 + a0'b0')$
 - Recall that XNOR outputs 1 if its two input bits are the same
 - $eq = (a3 xnor b3) * (a2 xnor b2) * (a1 xnor b1) * (a0 xnor b0)$

0110 = 0111 ?

© Vahid – Adapted by D. Lettmann

53

Magnitude Comparator

Universidade Federal de Santa Catarina

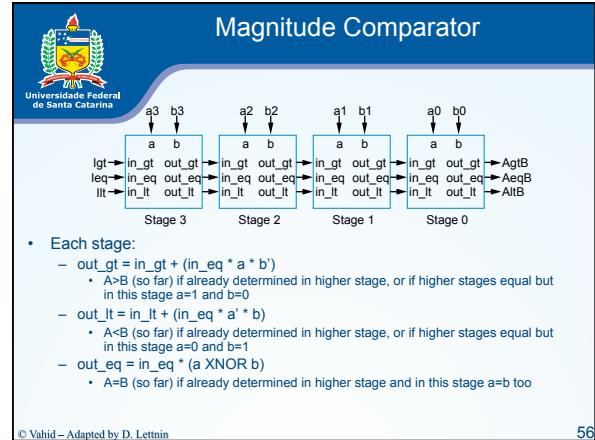
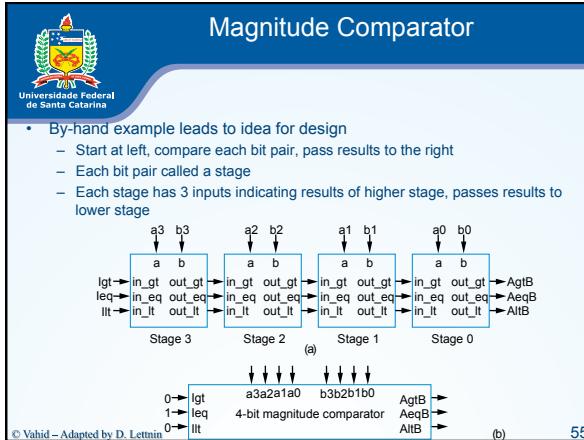
N-bit magnitude comparator: Indicates whether A>B, A=B, or A<B, for its two N-bit inputs A and B

- How design? Consider how compare by hand. First compare a3 and b3. If equal, compare a2 and b2. And so on. Stop if comparison not equal -- whichever's bit is 1 is greater. If never see unequal bit pair, A=B.

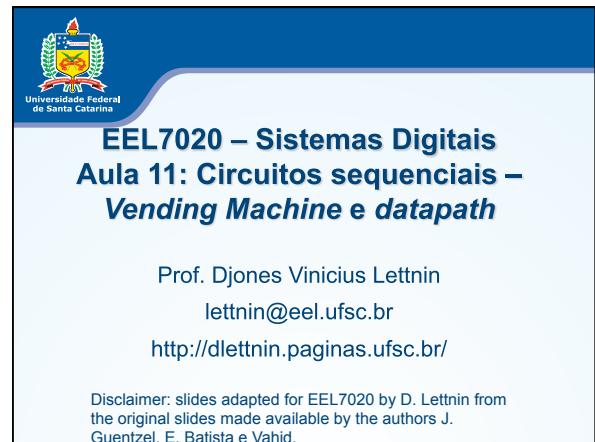
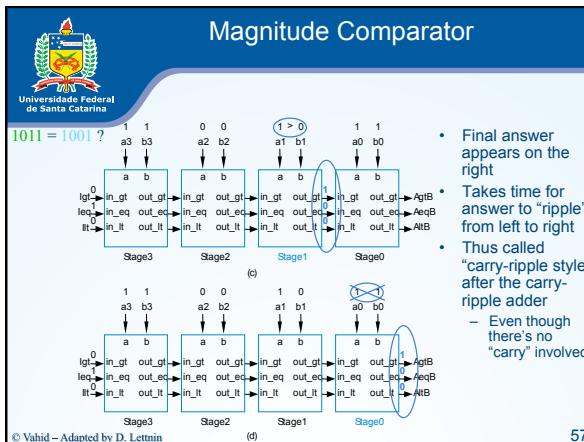
A=1011	B=1001
1011	1001 Equal
1011	1001 Equal
1011	1001 Unequal
So A > B	

© Vahid – Adapted by D. Lettmann

54



55



57