



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



Sistemas Digitais

INE 5406

Aula 6-T

4. Máquinas Seqüências Síncronas: Síntese de circuitos seqüenciais. Exemplos. Minimização de Estados.

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

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

4. Máquinas Seqüenciais Síncronas

▶ Minimização de Estados

- Para circuitos seqüenciais mais complexos do que os vistos até aqui, pode ocorrer que uma primeira versão projetada possua mais estados do que o necessário
- O número de estados determina o número de flip-flops necessários para o projeto do circuito seqüencial
- Uma redução no número de estados pode levar a uma redução da complexidade da lógica de estados (não necessariamente...)
- Logo, procedimentos para minimização de estados devem ser aplicados

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

- Se o número de estados de uma máquina de estados finitos (FSM) pode ser reduzido, então alguns dos estados da versão inicial devem ser equivalentes a outros estados, no que se refere a sua contribuição ao funcionamento do circuito como um todo.

Definição 1

Dois estados S_i e S_j são ditos equivalentes se e somente se, para toda seqüência possível de entradas, a mesma seqüência de saídas for produzida, independentemente se o estado inicial for S_i ou S_j .

4. Máquinas Seqüenciais Síncronas

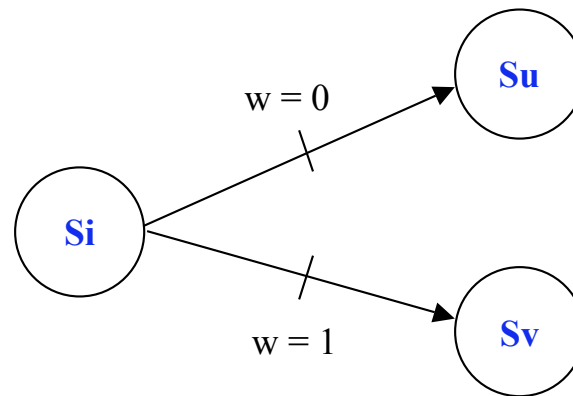
▶ Minimização de Estados

- **É possível desenvolver um procedimento que busca por estados equivalentes. Porém, tal procedimento é extremamente entediante para ser feito manualmente**
- **As ferramentas de síntese de circuitos normalmente possuem alguma implementação algorítmica de tal procedimento**
- **Para efeito de compreensão do processo de minimização, pode-se utilizar um procedimento que procura por estados que **não** são equivalentes entre si**

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Supondo que uma FSM possui uma única entrada, w



- Su é um sucessor-0 de Si
- Sv é um sucessor-1 de Si
- Su e Sv pertencem ao conjunto dos sucessores-k de Si (onde k corresponde a todas as combinações possíveis de vales de variáveis de entrada)

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

- Se os estados S_i e S_j são equivalentes, então seus sucessores- k correspondentes são também equivalentes
- Podemos formular um procedimento de minimização de estados o qual considera os estados como um conjunto, o qual vai sendo dividido em partições, de modo que cada partição contém estados equivalentes entre si

Definição 2

Uma partição consiste de um ou mais blocos, tal que cada bloco corresponde a um subconjunto de estados que possivelmente são equivalentes. Porém, os estados que estão em uma dada partição (definitivamente) não são equivalentes aos estados que estão em outra partição qualquer.

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 4

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		saída
	w=0	w=1	z
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

$$P1 = (A, B, C, D, E, F, G)$$

Criando o particionando P2:

- Agrupam-se todos os estados em que $z=0$
- Agrupam-se todos os estados em que $z=1$

$$P2 = (A, B, D) (C, E, F, G)$$

Pergunta: esta FSM corresponde a Mealy ou Moore?

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 4

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		saída
	w=0	w=1	z
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

$$P2 = (A, B, D) (C, E, F, G)$$

Criando o particionamento P3:

Sucessores-0 de (A, B, D) \rightarrow (B, D, B)

1. Todos os estados de (B, D, B) pertencem a um mesmo bloco em P2.

Sucessores-1 de (A, B, D) \rightarrow (C, F, G)

2. Todos os estados de (C, F, G) pertencem a um mesmo bloco de P2.

Por 1 e 2 acima, conclui-se que em P3 (A, B, D) continuará sendo um único grupo.

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 4

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		saída
	w=0	w=1	z
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

$$P2 = (A, B, D) (C, E, F, G)$$

Criando o particionamento P3:

Sucessores-0 de (C, E, F, G) \rightarrow (F, F, E, F)

1. Todos os estados de (F, F, E, F) pertencem a um mesmo bloco em P2.

Sucessores-1 de (C, E, F, G) \rightarrow (E, C, D, G)

2. (E, C, D, G): E, C, G pertencem a um mesmo bloco de P2, mas D pertence a outro bloco.

Logo, (C, E, F, G) dará origem a dois blocos em P3: (C, E, G) e (F)

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 4

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		saída
	w=0	w=1	z
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

$$P3 = (A, B, D) (C, E, G,) (F)$$

Criando o particionamento P4:

Sucessores-0 de (A, B, D) \rightarrow (B, D, B)

1. Todos os estados de (B, D, B) pertencem a um mesmo bloco em P3.

Sucessores-1 de (A, B, D) \rightarrow (C, F, G)

2. (C, F, G): C e G pertencem a um mesmo bloco de P3, porém F pertence a outro bloco. Logo, em P4 (A, B, D) ficará particionado em (A, D) (B).

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 4

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		saída
	w=0	w=1	z
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

$$P3 = (A, B, D) (C, E, G,) (F)$$

Criando o particionamento P4:

Sucessores-0 de (C, E, G) \rightarrow (F, F, F)

1. Todos os estados de (F, F, F) pertencem a um mesmo bloco em P3.

Sucessores-1 de (C, E, G) \rightarrow (E, C, G)

2. Todos os estados de (E, C, G) pertencem a um mesmo bloco de P3.

Por 1 e 2 acima, conclui-se que (C, E, G) continuará sendo um único grupo em P4.

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 4

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		saída
	w=0	w=1	z
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

$$P4 = (A, D) (B) (C, E, G,) (F)$$

Criando o particionamento P5:

Sucessores-0 de (A, D) \rightarrow (B, B)

1. Todos os estados de (B, B) pertencem a um mesmo bloco em P4.

Sucessores-1 de (A, D) \rightarrow (C, G)

2. Todos os estados de (C, G) pertencem a um mesmo bloco de P4.

Por 1 e 2 acima, conclui-se que (A, D) continuará sendo um único grupo em P5.

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 4

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		saída
	w=0	w=1	z
A	B	C	1
B	D	F	1
C	F	E	0
D	B	G	1
E	F	C	0
F	E	D	0
G	F	G	0

$$P4 = (A, D) (B) (C, E, G,) (F)$$

Criando o particionamento P5:

Sucessores-0 de (C, E, G) \rightarrow (F, F, F)

1. Todos os estados de (F, F, F) pertencem a um mesmo bloco em P4.

Sucessores-1 de (C, E, G) \rightarrow (E, C, G)

2. Todos os estados de (E, C, G) pertencem a um mesmo bloco de P4.

Por 1 e 2 acima, conclui-se que (C, E, G) continuará sendo um único grupo em P5.

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 4

Aplicando o procedimento de particionamento:

$$P5 = (A, D) (B) (C, E, G,) (F)$$

- Como $P5 = P4$, o procedimento pára.
- O resultado é que teremos apenas 4 estados, ao invés de 7.
- Chamando (A, D) de A e (C, E, G) de C, a nova tabela de transição/saída ficará:

Estado atual	Próximo estado		saída
	w=0	w=1	z
A	B	C	1
B	A	F	1
C	F	C	0
F	C	A	0

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

Inicialmente, vamos assumir que ambos DCs da saída z valem “0”. Então, teremos:

$$P1 = (A, B, C, D, E, F, G)$$

$$P2 = (A, B, D, G) (C, E, F)$$

Pergunta: esta FSM corresponde a Mealy ou Moore?

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P2 = (A, B, D, G) (C, E, F)$$

Criando o particionamento P3:

Sucessores-0 de (A, B, D, G) \rightarrow (B, D, B, F)

1. (B, D, B, F): B, D pertencem a um mesmo bloco em P2. Porém, F pertence a outro bloco. Logo, (A, B, D, G) ficará particionado em (A, B, D) e (G).

Sucessores-1 de (A, B, D) \rightarrow (C, -, G)

2. (C, -, G): C pertence a um mesmo bloco de P2, mas G pertence a um outro bloco. Logo, (A, B, D) poderá ser particionado em (A, B) e (D).

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P2 = (A, B, D, G) (C, E, F)$$

Criando o particionamento P3:

Sucessores-0 de (C, E, F) \rightarrow (F, F, E)

1. Todos os estados de (F, F, E) pertencem a um mesmo bloco em P2.

Sucessores-1 de (C, E, F) \rightarrow (E, C, D)

2. (E, C, D): E e C pertencem a um mesmo bloco de P2, mas D pertence a um outro bloco. Logo, (C, E, F) será ser particionado em (C, E) e (F).

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P3 = (A, B) (D) (G) (C, E) (F)$$

Criando o particionamento P4:

Sucessores-0 de (A, B) \rightarrow (B, D)

1. (B, D): B e D pertencem a blocos diferentes de P2. Logo, (A, B) ficará particionado em (A) e (B).

Não há necessidade de se analisar os sucessores-1 de (A, B), pois a análise dos sucessores-0 já particionou este bloco em dois...

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P3 = (A, B) (D) (G) (C, E) (F)$$

Criando o particionamento P4:

Sucessores-0 de (C, E) \rightarrow (F, F)

1. Todos os estados de (F, F) pertencem a um mesmo bloco em P2.

Sucessores-1 de (C, E) \rightarrow (E, C)

2. Todos os estados de (E, C) pertencem a um mesmo bloco de P2.

Pelos resultados 1 e 2, (C, E) não será mais particionado...

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P4 = (A) (B) (D) (G) (C, E) (F)$$

Criando o particionamento P5:

Sucessores-0 de (C, E) \rightarrow (F, F)

1. Todos os estados de (F, F) pertencem a um mesmo bloco em P2.

Sucessores-1 de (C, E) \rightarrow (E, C)

2. Todos os estados de (E, C) pertencem a um mesmo bloco de P2.

Pelos resultados 1 e 2, (C, E) não será mais particionado...

Logo, P5 = P4

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

Agora vamos assumir que ambos DCs da saída z valem “1”. Então, teremos:

$$P1 = (A, B, C, D, E, F, G)$$

$$P2 = (A, D) (B, C, E, F, G)$$

Pergunta: esta FSM corresponde a Mealy ou Moore?

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P2 = (A, D) (B, C, E, F, G)$$

Criando o particionamento P3:

Sucessores-0 de (A, D) \rightarrow (B, B)

1. Obviamente, todos os estados de (B, B) pertencem a um mesmo bloco em P2.

Sucessores-1 de (A, D) \rightarrow (C, G)

2. Todos os estados de (C, G) pertence a um mesmo bloco de P2.

Logo, (A, D) não será particionado em P3.

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P2 = (A, D) (B, C, E, F, G)$$

Criando o particionamento P3:

Sucessores-0 de (B, C, E, F, G) \rightarrow (D, F, F, E, F)

1. (D, F, F, E, F): D pertence a um bloco de P2 e os demais estados pertencem a outro bloco de P2.

Logo, (B, C, E, F, G) será particionado em (B) e (C, E, F, G).

Sucessores-1 de (C, E, F, G) \rightarrow (E, C, D, -)

2. (E, C, D, -): E e C pertencem a um mesmo bloco de P2, mas D pertence a um outro bloco. Logo, (C, E, F, G) será particionado em (C, E, G) e (F).

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P3 = (A, D) (B) (C, E, G) (F)$$

Criando o particionamento P4:

Sucessores-0 de (A, D) \rightarrow (B, B)

1. Obviamente, todos os estados de (B, B) pertencem a um mesmo bloco de P2.

Sucessores-1 de (A, D) \rightarrow (C, G)

2. (C, G): C e G estão no mesmo bloco em P3.

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)

Aplicando o procedimento de particionamento:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
D	B	G	0	0
E	F	C	0	1
F	E	D	0	1
G	F	-	0	-

$$P3 = (A, D) (B) (C, E, G) (F)$$

Criando o particionamento P4:

Sucessores-0 de (C, E, G) \rightarrow (F, F, F)

1. Obviamente, todos os estados de (F, F, F) pertencem a um mesmo bloco em P2.

Sucessores-1 de (C, E, G) \rightarrow (E, C, -)

2. Todos os estados de (E, C, -) pertencem a um mesmo bloco de P2.

Pelos resultados 1 e 2, (C, E, G) não será mais particionado...

Logo, P4 = P3

4. Máquinas Seqüenciais Síncronas

► Minimização de Estados

Exemplo 5 (Função Incompletamente Especificada)
Comparando-se os dois resultados obtidos:

Assumindo-se “0” para os DCs da saída z:

$P5 = (A)(B)(D)(G)(C, E)(F) \rightarrow 6$ estados

Assumindo-se “1” para os DCs da saída z:

$P4 = (A, D)(B)(C, E, G)(F) \rightarrow 4$ estados

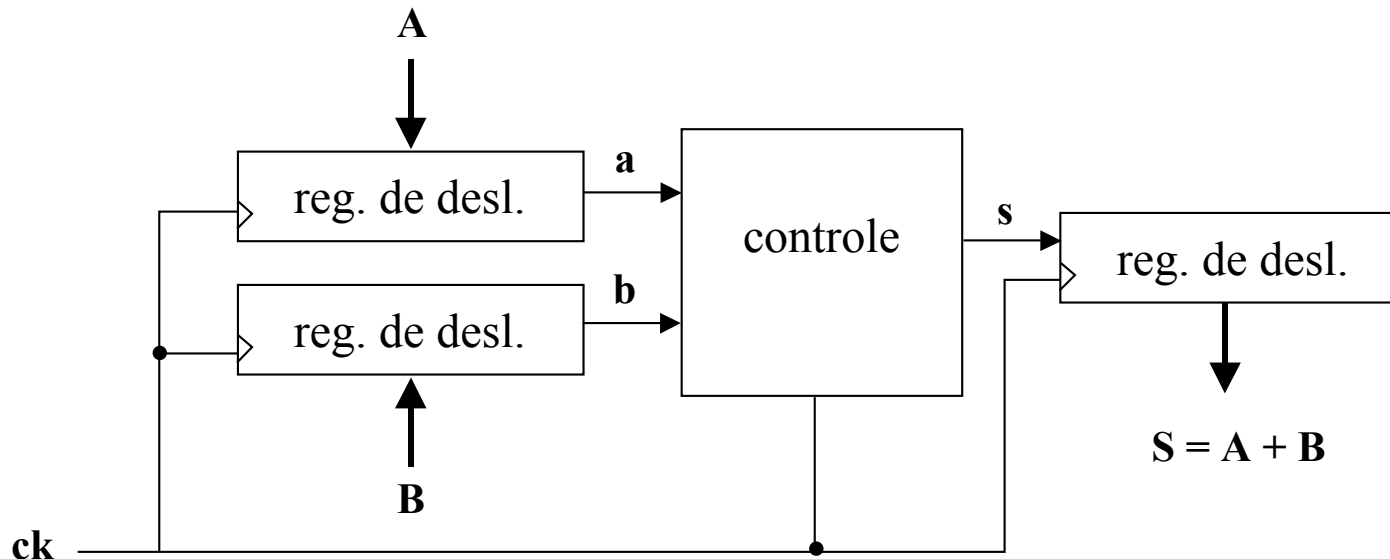
A nova tabela de transição/saída será:

Estado atual	Próximo estado		Saída z	
	w=0	w=1	w=0	w=1
A	B	C	0	0
B	D	-	0	-
C	F	E	0	1
F	E	D	0	1

4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

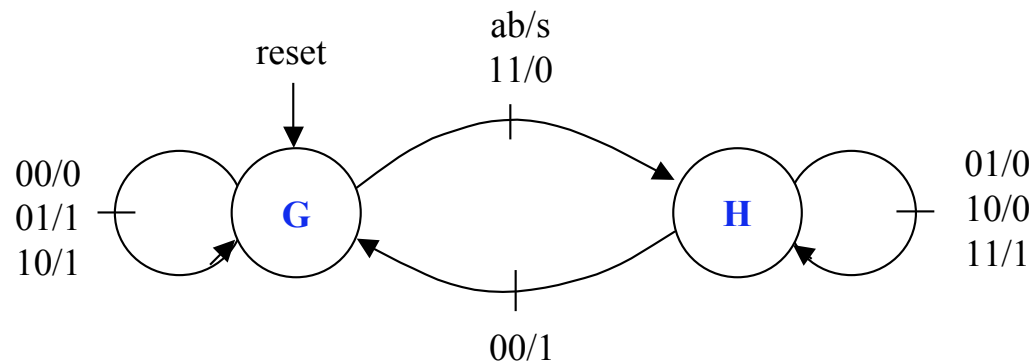
Exemplo 7: Somador (Bit) Serial



4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Mealy



Estado G: carry-in = 0

Estado H: carry-in = 1

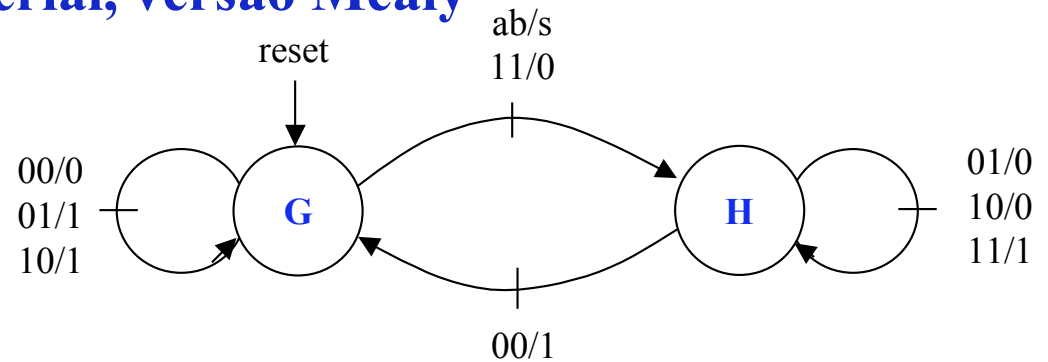
4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Mealy

Tabela de (transição de) estados

Estado atual	a	b	Próximo estado
G	0	0	G
G	0	1	G
G	1	0	G
G	1	1	H
H	0	0	G
H	0	1	H
H	1	0	H
H	1	1	H



Estado G: carry-in = 0

Estado H: carry-in = 1

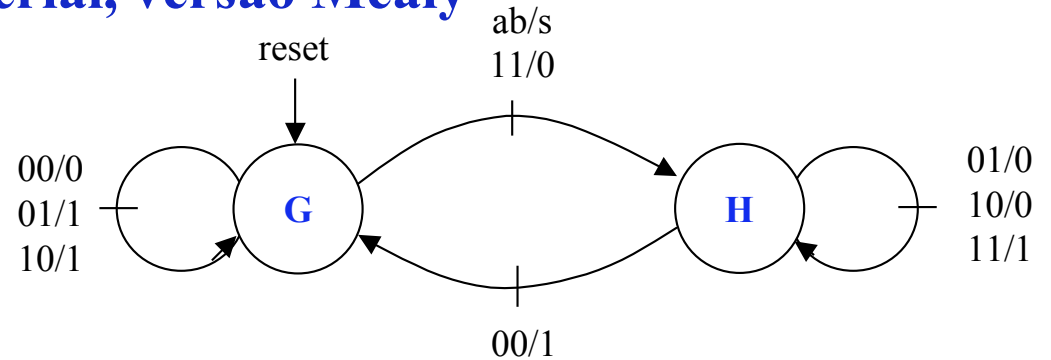
4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Mealy

Tabela de (transição de) estados

y	a	b	Y
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Y	$\bar{a}\bar{b}$	$\bar{a}b$	ab	$a\bar{b}$	$a \cdot b$
\bar{y}	0	0	1	0	
y	0	1	1	1	$a \cdot y$

$b \cdot y$ (points to the 1 in row y, column $\bar{a}b$)

$$Y = a \cdot b + a \cdot y + b \cdot y$$

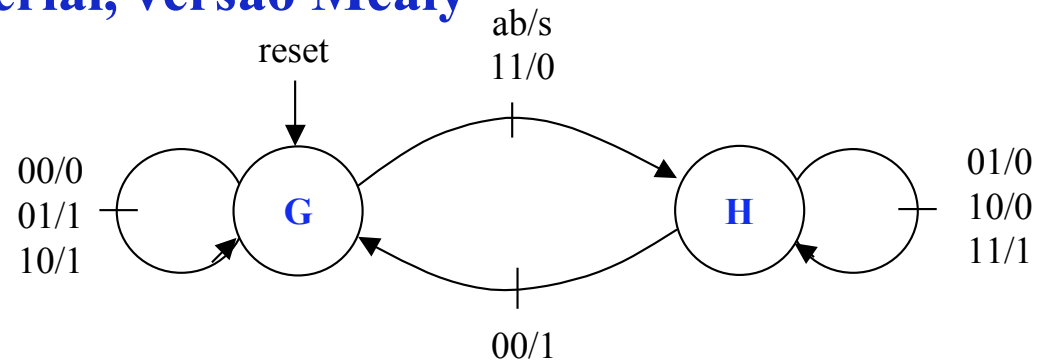
4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Mealy

Tabela de saída

y	a	b	s
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	0
1	1	1	1



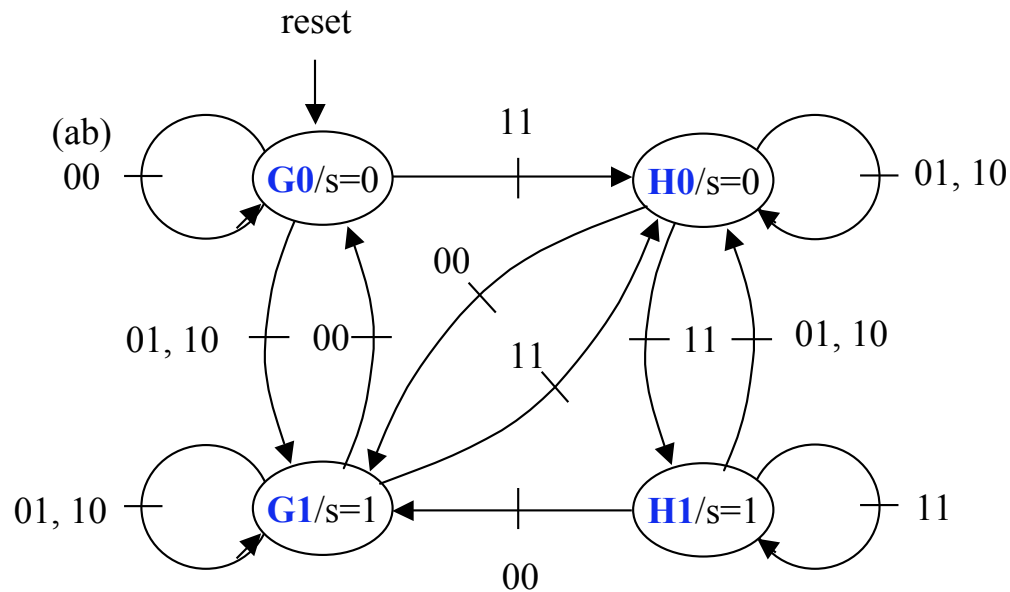
Y	$\bar{a}\bar{b}$	$\bar{a}b$	ab	$a\bar{b}$
\bar{y}	0	1	0	1
y	1	0	1	0

$$s = a \oplus b \oplus y$$

4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Moore



Estados G0 e G1: carry-out = 0

Estados H0 e H1: carry-out = 1

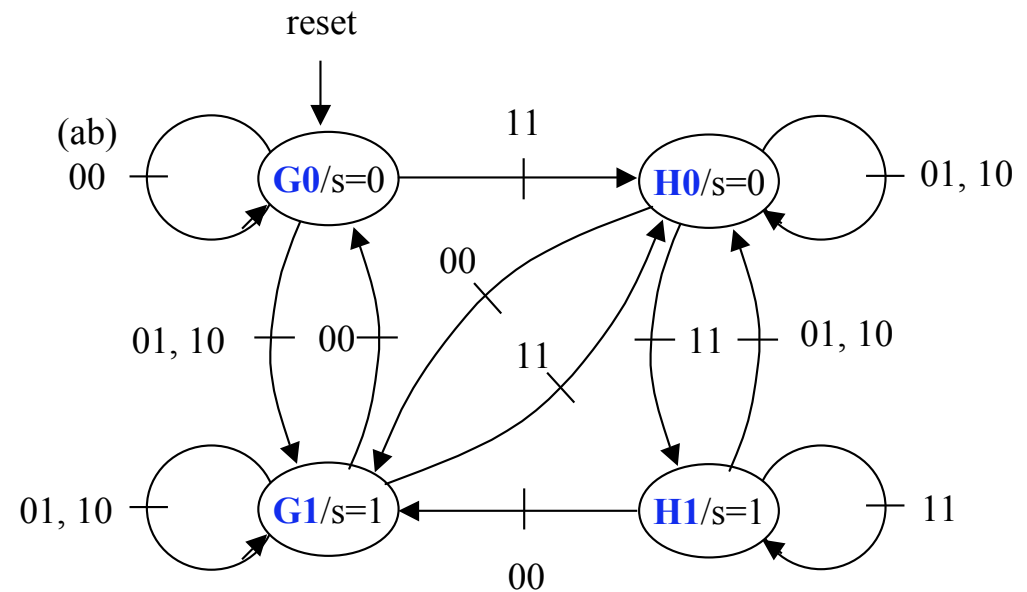
4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Moore

Tabela de estados e de saída

Estado atual	Próximo estado				saída
	00	01	10	01	
G0	G0	G1	G1	H0	0
G1	G0	G1	G1	H0	1
H0	G1	H0	H0	H1	0
H1	G1	H0	H0	H1	1



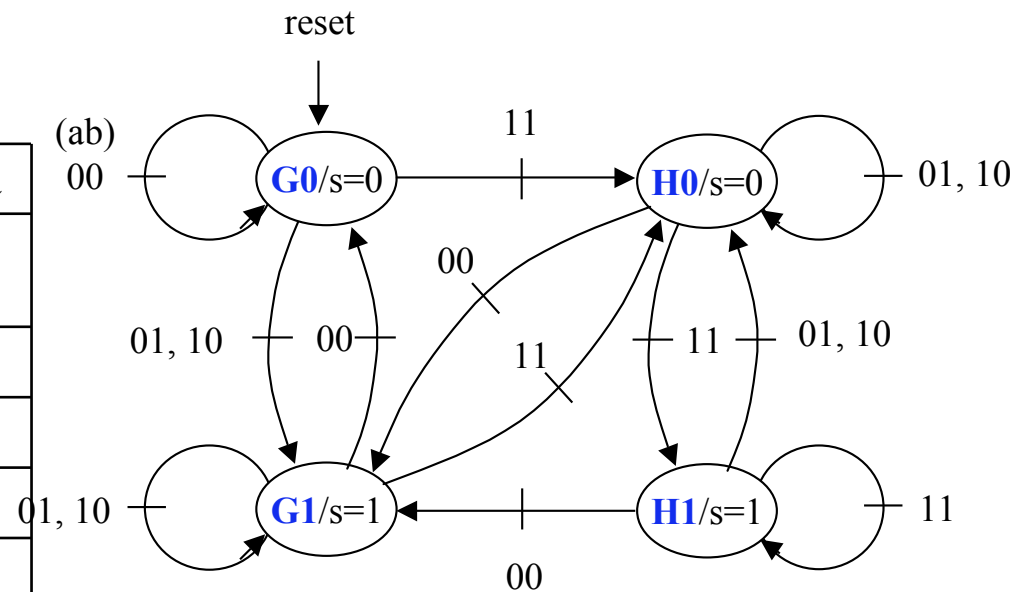
4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Moore

Tabela de estados e de saída
(Após a codificação dos estados)

Estado atual y_1y_0	Próximo estado (Y_1Y_0)				saída s
	00	01	10	01	
00	00	01	01	10	0
01	00	01	01	10	1
10	01	10	10	11	0
11	01	10	10	11	1



4. Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Moore

Estado atual y_1y_0	Próximo estado (Y_1Y_0)				saída s
	00	01	10	01	
00	00	01	01	10	0
01	00	01	01	10	1
10	01	10	10	11	0
11	01	10	10	11	1

$$Y_0 = a \oplus b \oplus y_1$$

$$Y_1 = a \cdot b + a \cdot y_1 + b \cdot y_1$$

$$s = y_0$$

4. Máquinas Seqüenciais Síncronas

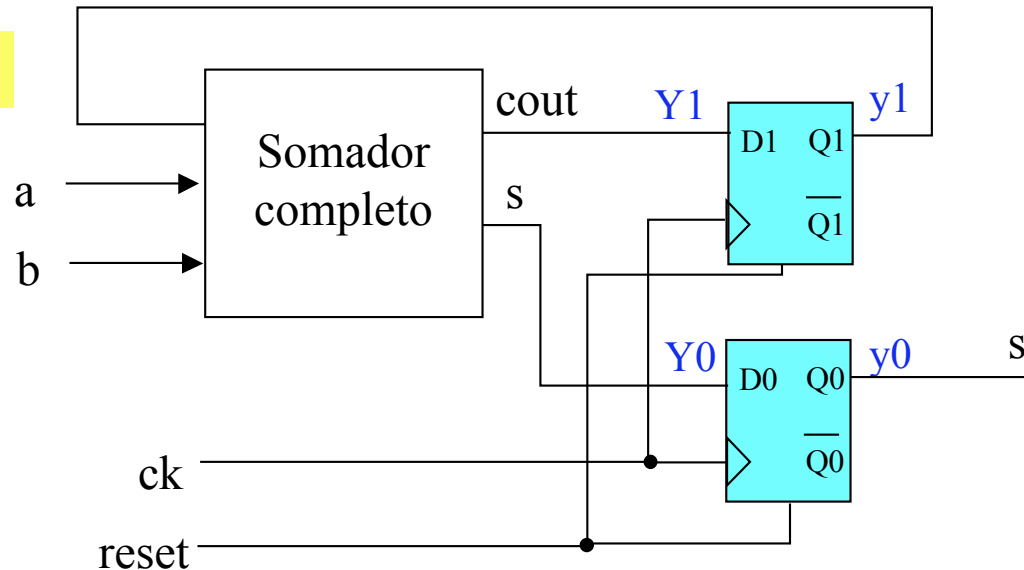
► Síntese de Circuitos Seqüenciais

Exemplo 7: Somador (Bit) Serial, versão Moore

$$Y0 = a \oplus b \oplus y1$$

$$Y1 = a \cdot b + a \cdot y1 + b \cdot y1$$

$$s = y0$$



4. Máquinas Seqüenciais Síncronas

VHDL para o Exemplo 7

Deslocador

```
LIBRARY ieee ;
USE ieee.std_logic_1164.all ;

-- registrador-deslocador para a direita, com carga paralela e sinal de enable
ENTITY shiftrne IS
    GENERIC ( N : INTEGER := 4 ) ;
    PORT ( R      : IN          STD_LOGIC_VECTOR(N-1 DOWNT0 0) ;
          L, E, w : IN          STD_LOGIC ;
          Clock   : IN          STD_LOGIC ;
          Q       : BUFFER      STD_LOGIC_VECTOR(N-1 DOWNT0 0) ) ;
END shiftrne ;

ARCHITECTURE Behavior OF shiftrne IS
BEGIN
    PROCESS
    BEGIN
        WAIT UNTIL Clock'EVENT AND Clock = '1' ;
        IF E = '1' THEN
            IF L = '1' THEN
                Q <= R ;
            ELSE
                Genbits: FOR i IN 0 TO N-2 LOOP
                    Q(i) <= Q(i+1) ;
                END LOOP ;
                Q(N-1) <= w ;
            END IF ;
        END IF ;
    END PROCESS ;
END Behavior ;
```

Retirado de BROWN, S.; VRANESIC, Z. Fundamentals of Digital Logic with VHDL Design. McGraw-Hill, 2000.

4. Máquinas Seqüenciais Síncronas

VHDL para o Exemplo 7

Deslocador

```
1 LIBRARY ieee ;
2 USE ieee.std_logic_1164.all ;

3 ENTITY serial IS
4     GENERIC ( length : INTEGER := 8 ) ;
5     PORT ( Clock : IN          STD_LOGIC ;
6           Reset  : IN          STD_LOGIC ;
7           A, B   : IN          STD_LOGIC_VECTOR(length-1 DOWNTO 0) ;
8           Sum   : BUFFER      STD_LOGIC_VECTOR(length-1 DOWNTO 0) );
9 END serial ;

10 ARCHITECTURE Behavior OF serial IS
11     COMPONENT shiftreg
12         GENERIC ( N : INTEGER := 4 ) ;
13         PORT ( R      : IN          STD_LOGIC_VECTOR(N-1 DOWNTO 0) ;
14               L, E, w : IN          STD_LOGIC ;
15               Clock  : IN          STD_LOGIC ;
16               Q      : BUFFER      STD_LOGIC_VECTOR(N-1 DOWNTO 0) ) ;
17     END COMPONENT ;

18 SIGNAL QA, QB, Null_in : STD_LOGIC_VECTOR(length-1 DOWNTO 0) ;
19 SIGNAL s, Low, High, Run : STD_LOGIC ;
20 SIGNAL Count : INTEGER RANGE 0 TO length ;
21 TYPE State_type IS (G, H) ;
22 SIGNAL y : State_type ;
    continua no próximo slide
```

Retirado de BROWN, S.; VRANESIC, Z. Fundamentals of Digital Logic with VHDL Design. McGraw-Hill, 2000.

4. Máquinas Seqüenciais Síncronas

VHDL para o Exemplo 7

Deslocador

```
23 BEGIN
24     Low <= '0' ; High <= '1' ;
25     ShiftA: shiftrne GENERIC MAP (N => length)
26         PORT MAP ( A, Reset, High, Low, Clock, QA ) ;
27     ShiftB: shiftrne GENERIC MAP (N => length)
28         PORT MAP ( B, Reset, High, Low, Clock, QB ) ;
29     AdderFSM: PROCESS ( Reset, Clock )
30     BEGIN
31         IF Reset = '1' THEN
32             y <= G ;
33         ELSIF Clock'EVENT AND Clock = '1' THEN
34             CASE y IS
35                 WHEN G =>
36                     IF QA(0) = '1' AND QB(0) = '1' THEN y <= H ;
37                     ELSE y <= G ;
38                     END IF ;
39                 WHEN H =>
40                     IF QA(0) = '0' AND QB(0) = '0' THEN y <= G ;
41                     ELSE y <= H ;
42                     END IF ;
43             END CASE ;
44         END IF ;
45     END PROCESS AdderFSM ;
```

Continua no próximo slide

Retirado de BROWN, S.; VRANESIC, Z. Fundamentals of Digital Logic with VHDL Design. McGraw-Hill, 2000.

4. Máquinas Seqüenciais Síncronas

```
46 WITH y SELECT
47     s <= QA(0) XOR QB(0) WHEN G,
48         NOT ( QA(0) XOR QB(0) ) WHEN H ;
49 Null_in <= (OTHERS => '0') ;
50 ShiftSum: shiftrne GENERIC MAP ( N => length )
51         PORT MAP ( Null_in, Reset, Run, s, Clock, Sum ) ;
52 Stop: PROCESS
53 BEGIN
54     WAIT UNTIL (Clock'EVENT AND Clock = '1') ;
55     IF Reset = '1' THEN
56         Count <= length ;
57     ELSIF Run = '1' THEN
58         Count <= Count -1 ;
59     END IF ;
60 END PROCESS ;
61 Run <= '0' WHEN Count = 0 ELSE '1' ; -- stops counter and ShiftSum
62 END Behavior ;
```

Retirado de BROWN, S.; VRANESIC, Z. Fundamentals of Digital Logic with VHDL Design. McGraw-Hill, 2000.