



FUNDAÇÃO EDSON QUEIROZ
UNIVERSIDADE DE FORTALEZA
ENSINANDO E APRENDENDO

T566 –SISTEMAS DIGITAIS AVANÇADOS

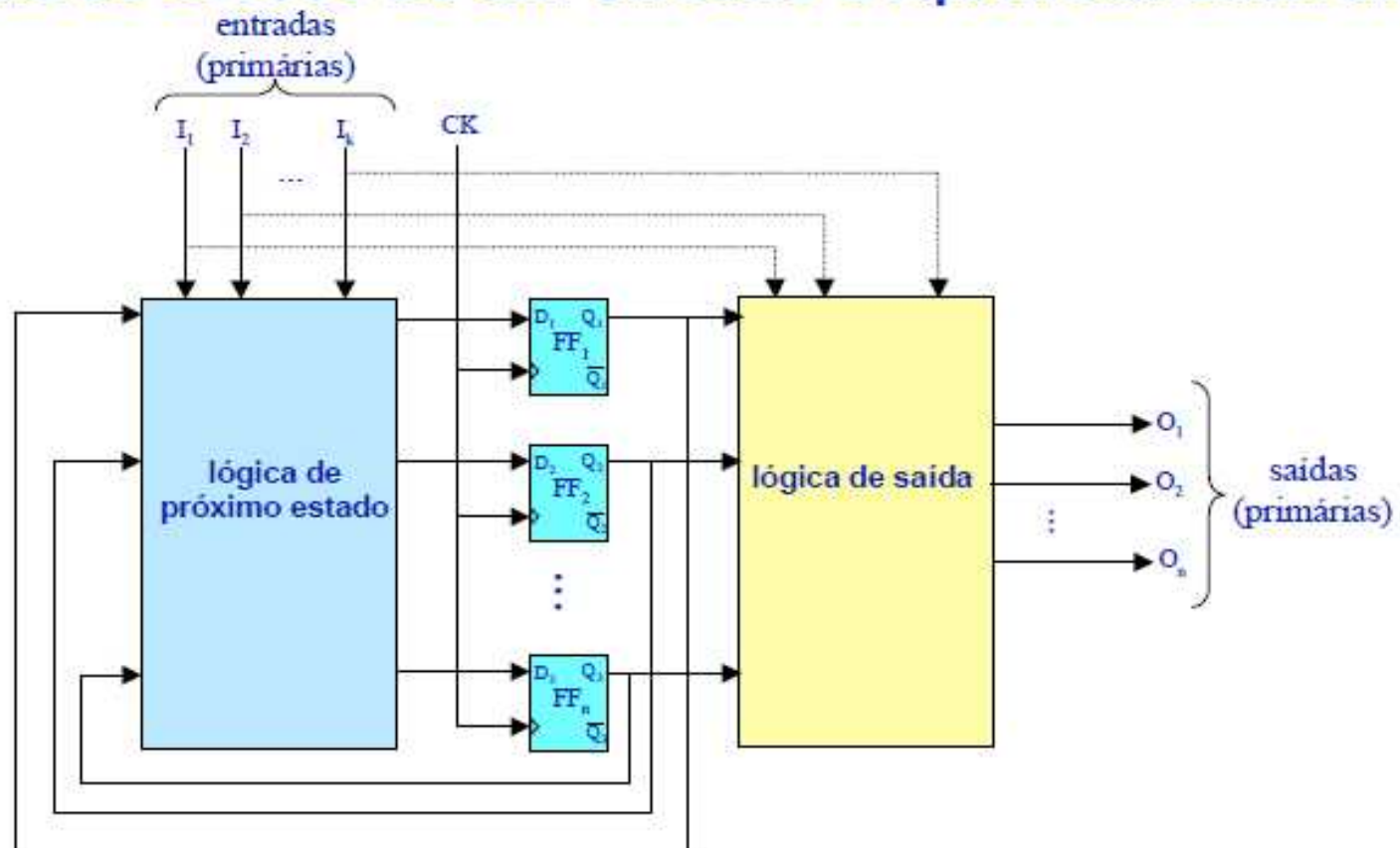
Aula 17- Máquinas de Estados Finitas (FSM)

Prof. Danilo Reis



Máquinas Seqüenciais Síncronas

▶ Diagrama de Blocos de um Circuito Seqüencial Síncrono





Máquinas Seqüenciais Síncronas

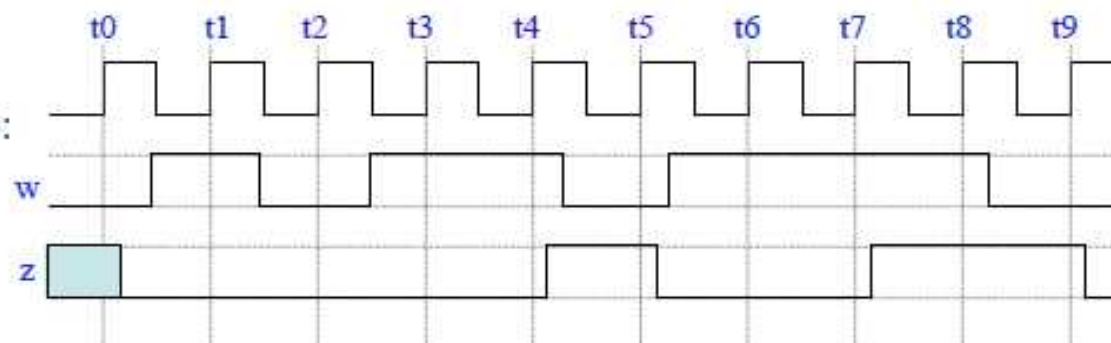
► Síntese de Circuitos Seqüenciais

Exemplo 2:

Projete um circuito que satisfaça às seguintes especificações:

1. O circuito possui uma entrada, **w**, e uma saída, **z**.
2. Todas as mudanças de valores no circuito ocorrem na borda de subida do sinal de relógio.
3. Quando o circuito detectar que a entrada **w** vale “0”, a saída **z** deve valer “0” no ciclo de relógio seguinte. Porém, quando o circuito detectar que a entrada **w** vale “1” durante duas bordas de relógio consecutivas, a saída **z** deve passar a valer “1” no ciclo de relógio seguinte à segunda ocorrência do valor “1”. As mudanças de **z** estão sincronizadas com a borda de relógio ativa.

Exemplo de funcionamento:

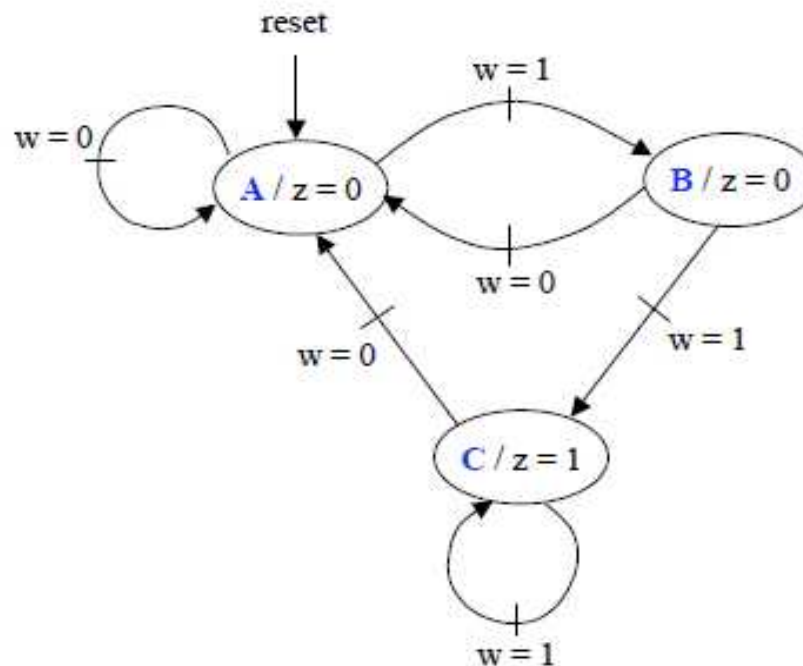




Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2: Diagrama de Estados





Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2: Diagrama de Blocos

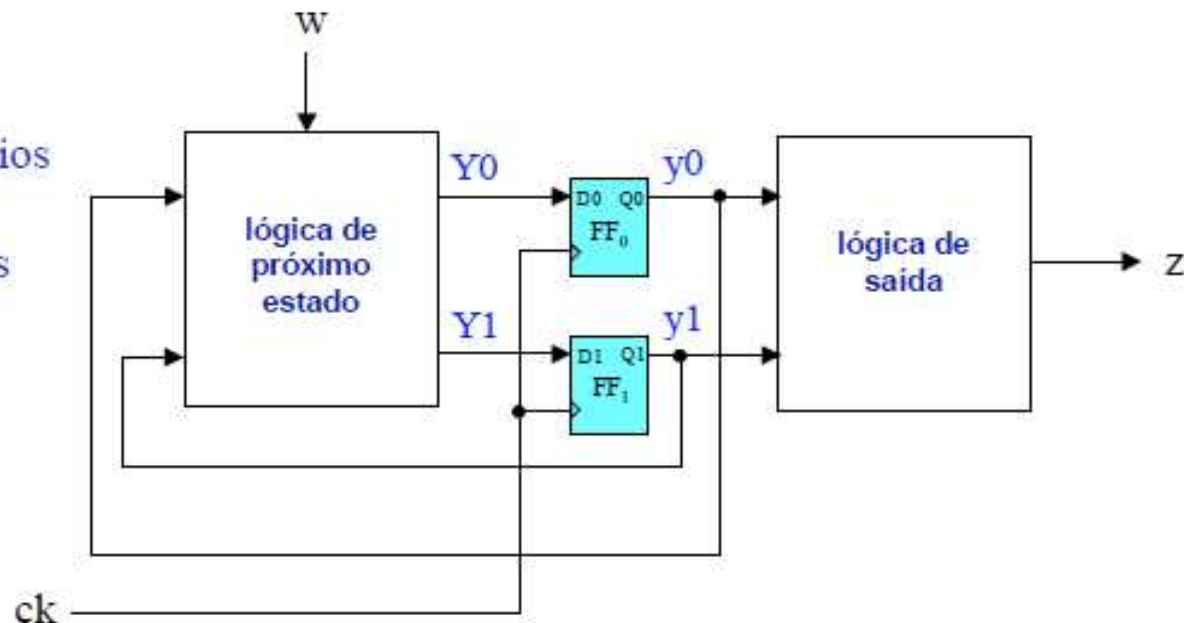
Sinais de entrada: w

Sinais de saída: z

Número de estados: 3

⇒ Logo, são necessários

2 flip-flops para
armazenar as variáveis
de estado





Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2: Assinalamento de Estados

Supondo o seguinte assinalamento: A=00, B=01, C=10

Estado atual	w	Próximo estado
A	0	A
A	1	B
B	0	A
B	1	C
C	0	A
C	1	C



	Estado atual y1y0	w	Próximo estado Y1Y0	
A	00	0	00	A
A	00	1	01	B
B	01	0	00	A
B	01	1	10	C
C	10	0	00	A
C	10	1	10	C
-	11	0	XX	-
-	11	1	XX	-



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2: Projetando a lógica de próximo estado

y ₁ y ₀	w	Y ₁ Y ₀
00	0	00
00	1	01
01	0	00
01	1	10
10	0	00
10	1	10
11	0	XX
11	1	XX

Y ₁	$\overline{y_1}\overline{y_0}$	$\overline{y_1}y_0$	$y_1\overline{y_0}$	y_1y_0
\overline{w}	0	0	X	0
w	0	1	X	1

$w \cdot y_0$ (points to the '1' in row w, column $\overline{y_1}y_0$)

$w \cdot y_1$ (points to the '1' in row w, column $y_1\overline{y_0}$)

$$Y_1 = w \cdot y_1 + w \cdot y_0 \\ = w (y_1 + y_0)$$

Y ₀	$\overline{y_1}\overline{y_0}$	$\overline{y_1}y_0$	$y_1\overline{y_0}$	y_1y_0
\overline{w}	0	0	X	0
w	1	0	X	0

$w \cdot \overline{y_1} \cdot \overline{y_0}$ (points to the '1' in row w, column $\overline{y_1}\overline{y_0}$)

$$Y_0 = w \cdot \overline{y_1} \cdot \overline{y_0}$$



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2: Projetando a lógica de saída

	Estado y1y0	z
A	00	0
B	01	0
C	10	1
	11	X

z	$\overline{y0}$	y0
$\overline{y1}$	0	0
y1	1	X

A red box highlights the cell containing '1' at the intersection of y1=1 and $\overline{y0}$. A red line points from this cell to the label 'y1' below the table.

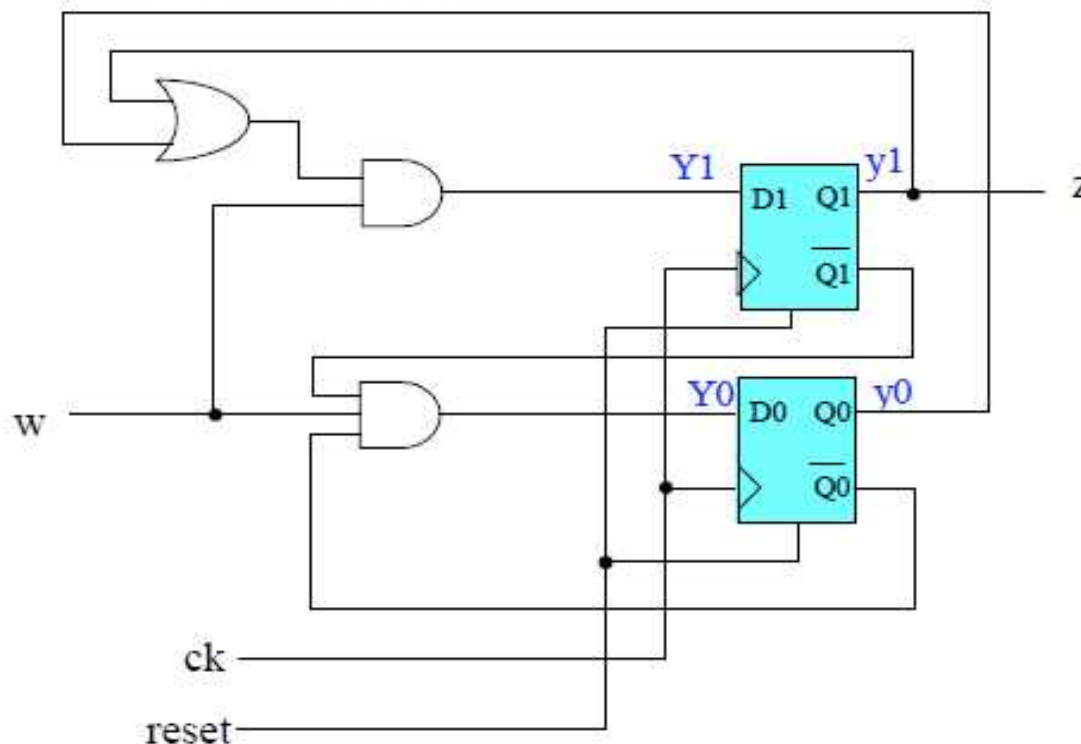
$$z = y1$$



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2: circuito final



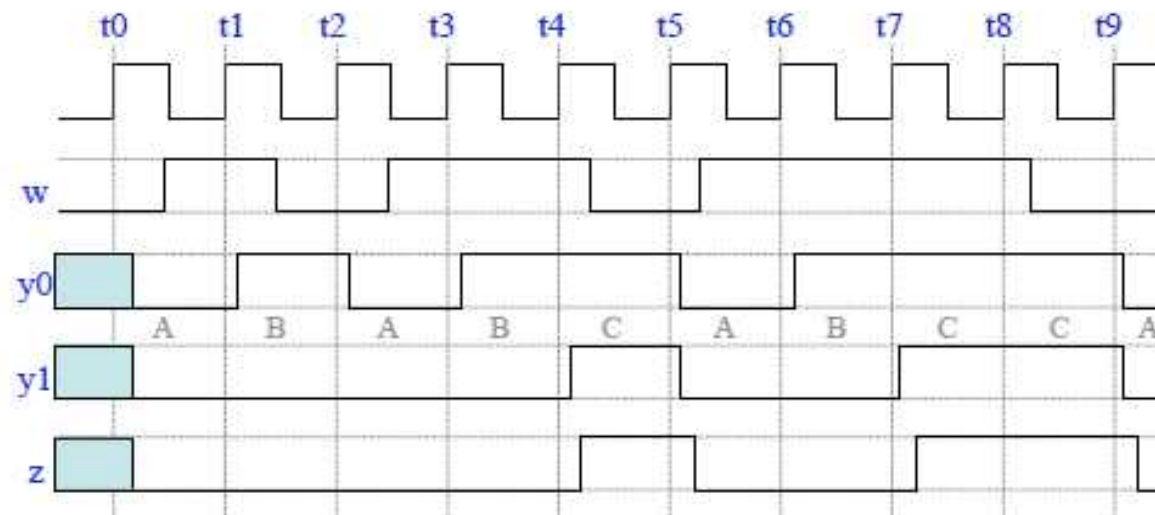
$$\begin{aligned} Y1 &= w \cdot y1 + w \cdot y0 \\ &= w (y1 + y0) \\ Y0 &= w \cdot \overline{y1} \cdot \overline{y0} \\ z &= y1 \end{aligned}$$



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2: Funcionamento dinâmico

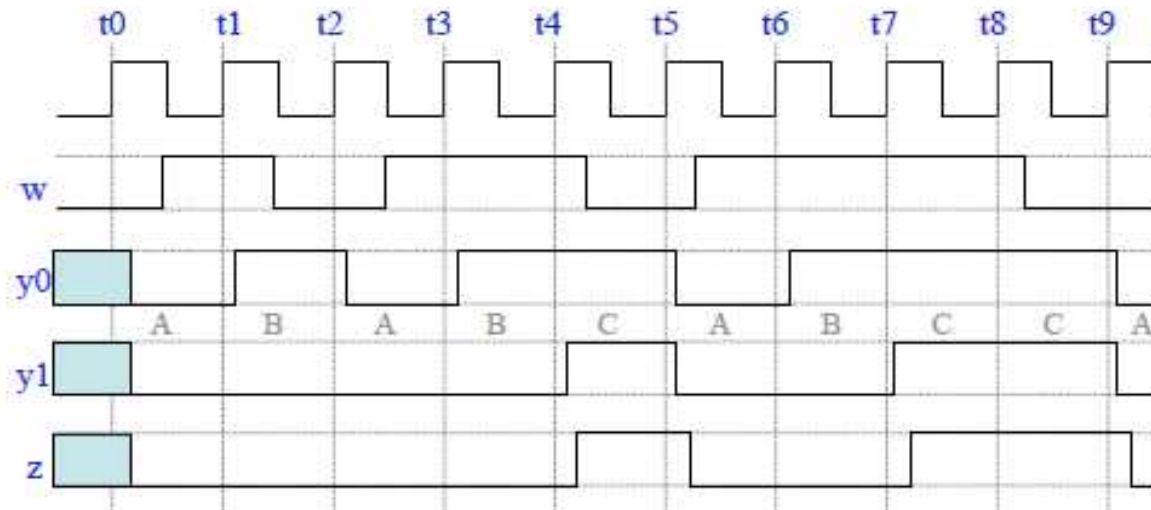




Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2: Funcionamento dinâmico



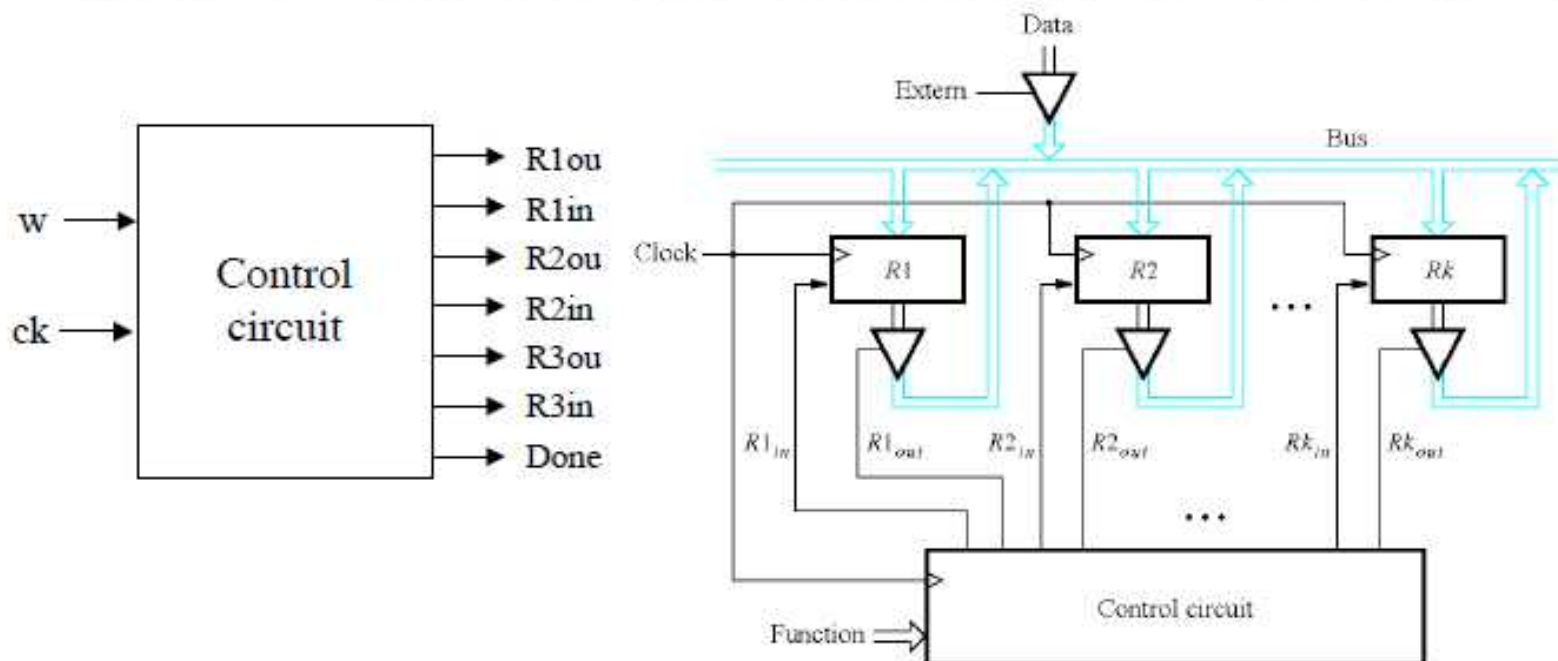


Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3:

Projetar o circuito de controle que permita realizar um *swap* entre R1 e R2, utilizando R3 como temporário. (Desconsiderar outras possíveis operações.)





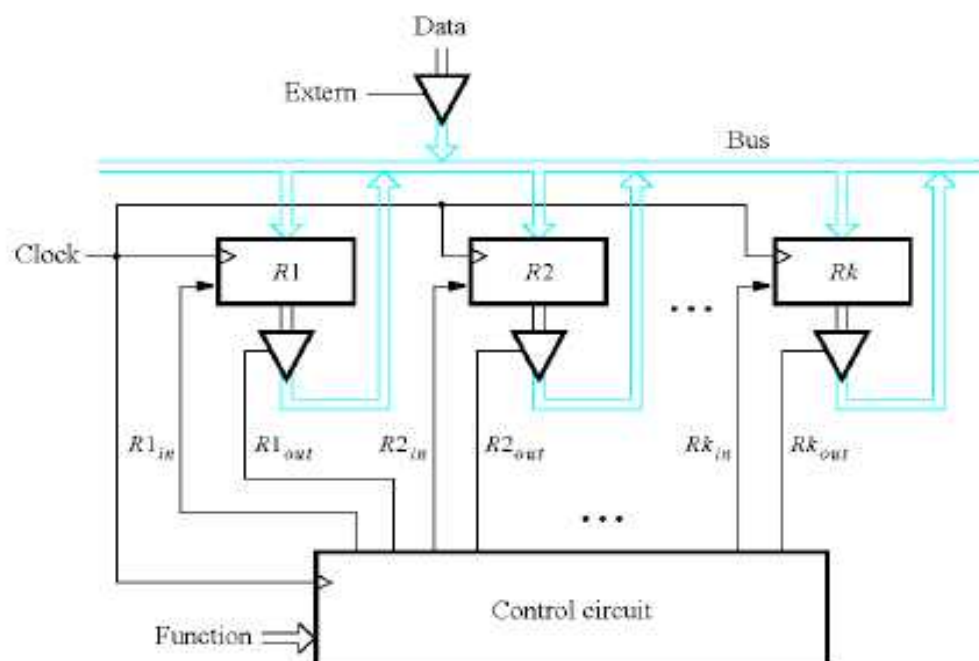
Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3:

Passos:	Sinais de controle
$R3 \leftarrow R2;$	$R2_{out} = 1; R3_{in} = 1;$
$R2 \leftarrow R1;$	$R1_{out} = 1; R2_{in} = 1;$
$R1 \leftarrow R3;$	$R3_{out} = 1; R1_{in} = 1;$
	Done = 1;

Obs: os sinais não citados na tabela devem valer "0"



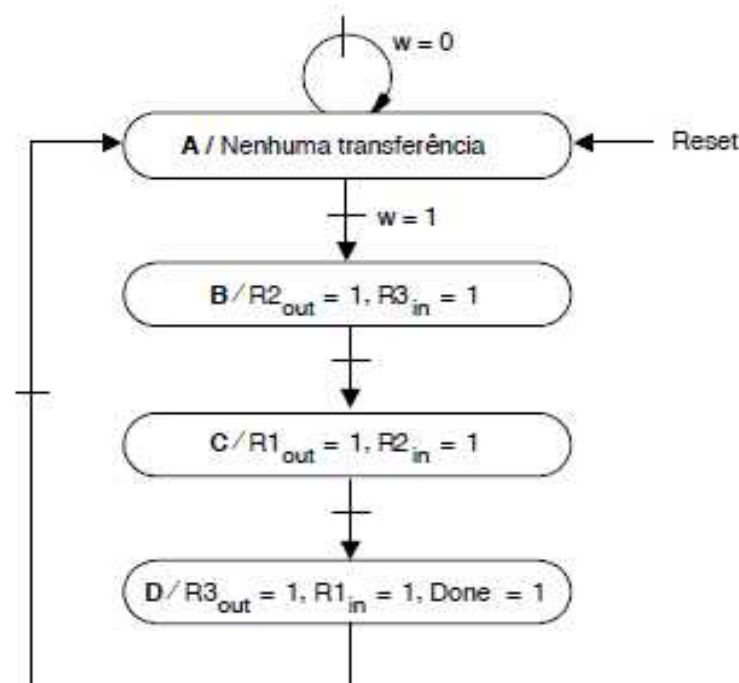


Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3: Diagrama de estados

Passos:	Sinais de controle
$R3 \leftarrow R2;$	$R2_{out} = 1; R3_{in} = 1;$
$R2 \leftarrow R1;$	$R1_{out} = 1; R2_{in} = 1;$
$R1 \leftarrow R3;$	$R3_{out} = 1; R1_{in} = 1;$ $Done = 1;$





Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3: Tabelas de (Transição de) Estados e de Saída

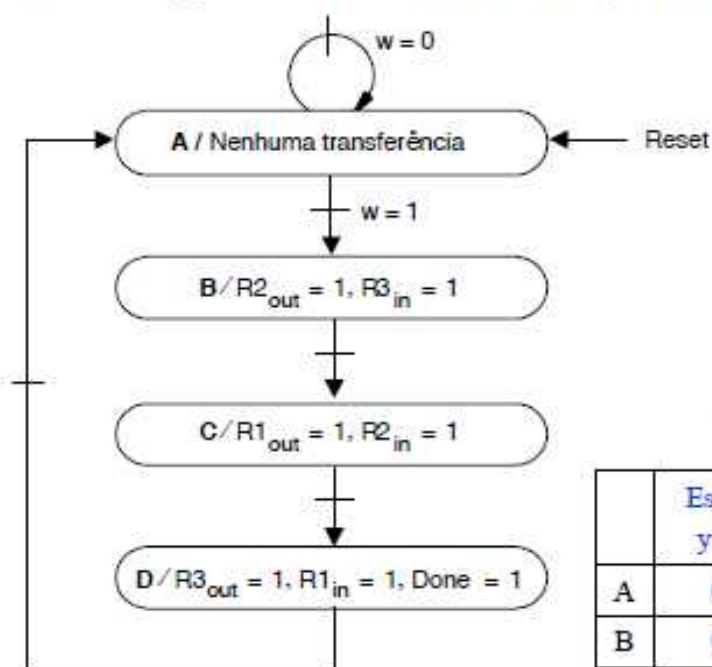


Tabela de
(transição de)
estados

Estado atual	w	Próximo estado
A	0	A
A	1	B
B	X	C
C	X	D
D	X	A

Tabela de saída

	Estado yly0	R1out	R1in	R2out	R2in	R3out	R3in	Done
A	00	0	0	0	0	0	0	0
B	01	0	0	1	0	0	1	0
C	10	1	0	0	1	0	0	0
D	11	0	1	0	0	1	0	1



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3: Diagrama de Blocos

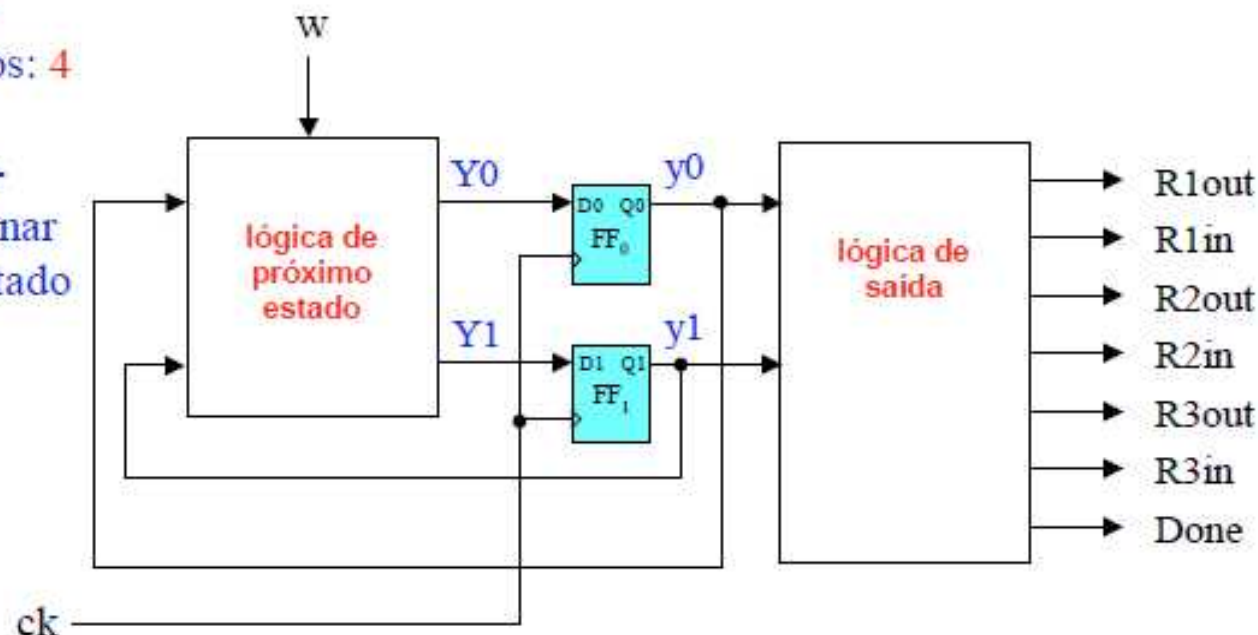
Sinais de entrada: w

Sinais de saída: z

Número de estados: 4

⇒ Logo, são

necessários 2 flip-flops para armazenar as variáveis de estado





Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3: Assinalamento de Estados

Supondo o seguinte assinalamento: A=00, B=01, C=10, D=11

Estado atual	w	Próximo estado
A	0	A
A	1	B
B	X	C
C	X	D
D	X	A



	Estado atual y1y0	w	Próximo estado Y1Y0	
A	00	0	00	A
A	00	1	01	B
B	01	X	10	C
C	10	X	11	D
D	11	X	00	A



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3: Projetando a lógica de próximo estado

	Estado atual y_1y_0	w	Próximo estado Y_1Y_0	
A	00	0	00	A
A	00	1	01	B
B	01	X	10	C
C	10	X	11	D
D	11	X	00	A

Y1	$\bar{y}_1\bar{y}_0$	\bar{y}_1y_0	y_1y_0	$y_1\bar{y}_0$
\bar{w}	0	1	0	1
w	0	1	0	1

$\bar{y}_1 \cdot y_0$

$y_1 \cdot \bar{y}_0$

$$Y_1 = \bar{y}_1 \cdot y_0 + y_1 \cdot \bar{y}_0 = y_1 \oplus y_0$$

Y0	$\bar{y}_1\bar{y}_0$	\bar{y}_1y_0	y_1y_0	$y_1\bar{y}_0$
\bar{w}	0	0	0	1
w	1	0	0	1

$y_1 \cdot \bar{y}_0$

$w \cdot \bar{y}_0$

$$Y_0 = w \cdot \bar{y}_0 + y_1 \cdot \bar{y}_0$$



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3: Projetando a lógica de saída

	Estado y1y0	R1out	R1in	R2out	R2in	R3out	R3in	Done
A	00	0	0	0	0	0	0	0
B	01	0	0	1	0	0	1	0
C	10	1	0	0	1	0	0	0
D	11	0	1	0	0	1	0	1

$$R1out = R2in = y1 \cdot \overline{y0}$$

$$R1in = R3out = Done = y1 \cdot y0$$

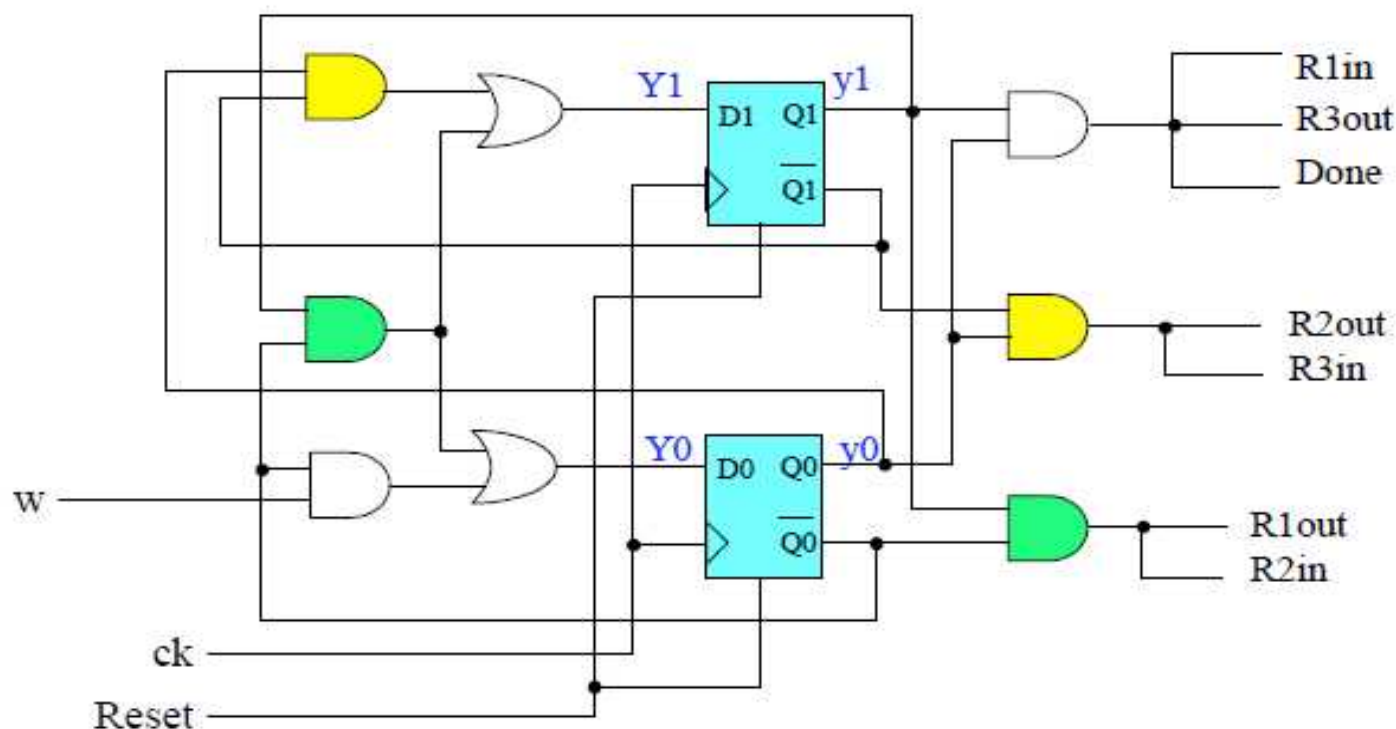
$$R2out = R3in = \overline{y1} \cdot y0$$



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3: Circuito de controle, versão 1

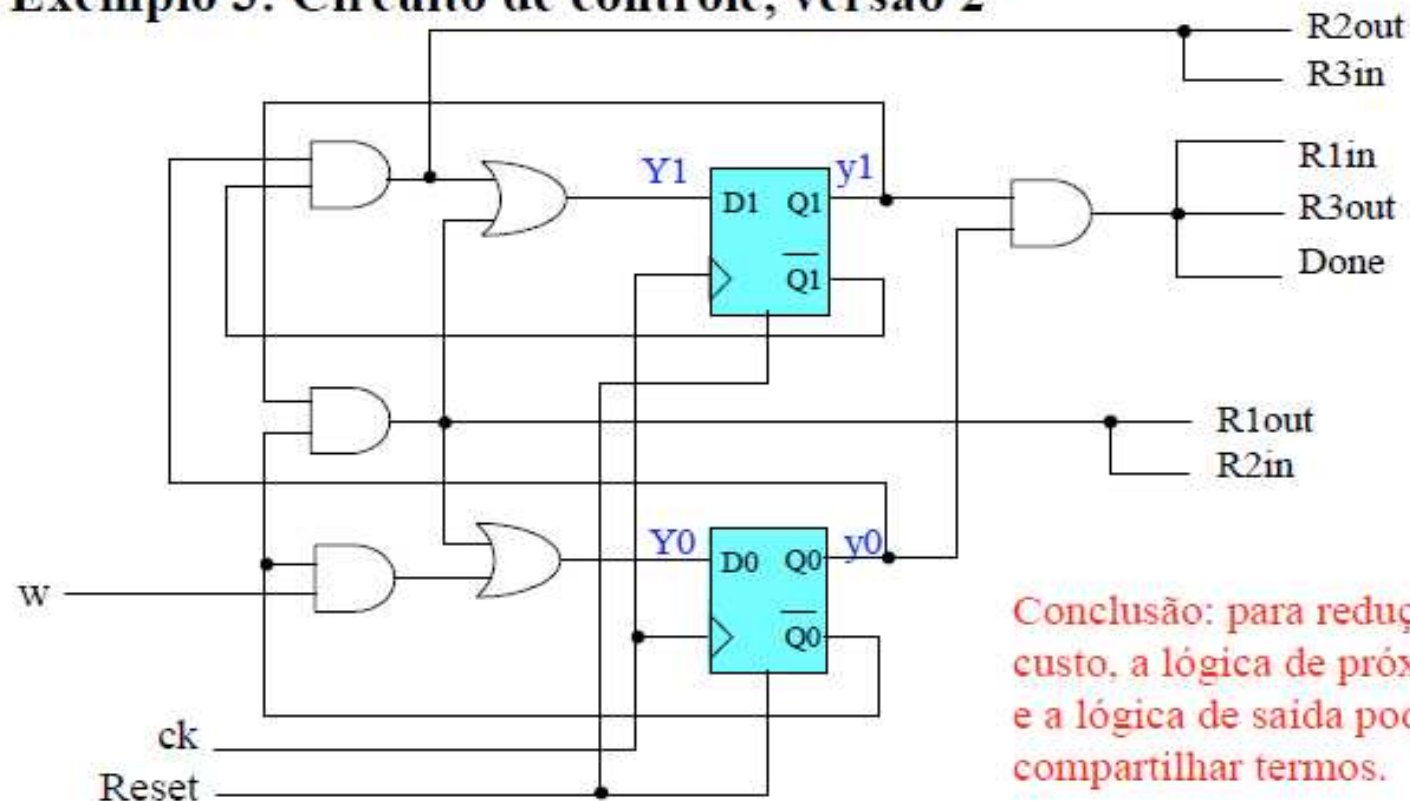




Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 3: Circuito de controle, versão 2



Conclusão: para redução do custo, a lógica de próximo estado e a lógica de saída podem compartilhar termos.



Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

Considerando o exemplo 2, porém com o assinalamento de estados
A=00, B=01, C=11

	Estado atual y1y0	w	Próximo estado Y1Y0	
A	00	0	00	A
A	00	1	01	B
B	01	0	00	A
B	01	1	11	C
-	10	0	XX	-
-	10	1	XX	-
C	11	0	00	A
C	11	1	11	C

	Estado y1y0	z
A	00	0
B	01	0
-	10	X
C	11	1



Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

Considerando o exemplo 2, porém com o assinalamento de estados
 $A=00$, $B=01$, $C=11$

y_1y_0	z
00	0
01	0
10	X
11	1

z	$\overline{y_0}$	y_0
$\overline{y_1}$	0	0
y_1	X	1

y_1

$$z = y_1$$

Coincidentemente, a equação de saída não mudou.



Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

Considerando o exemplo 2, porém com o assinalamento de estados
 $A=00$, $B=01$, $C=11$

y_1y_0	z
00	0
01	0
10	X
11	1

z	$\overline{y_0}$	y_0
$\overline{y_1}$	0	0
y_1	X	1

y_1

$$z = y_1$$

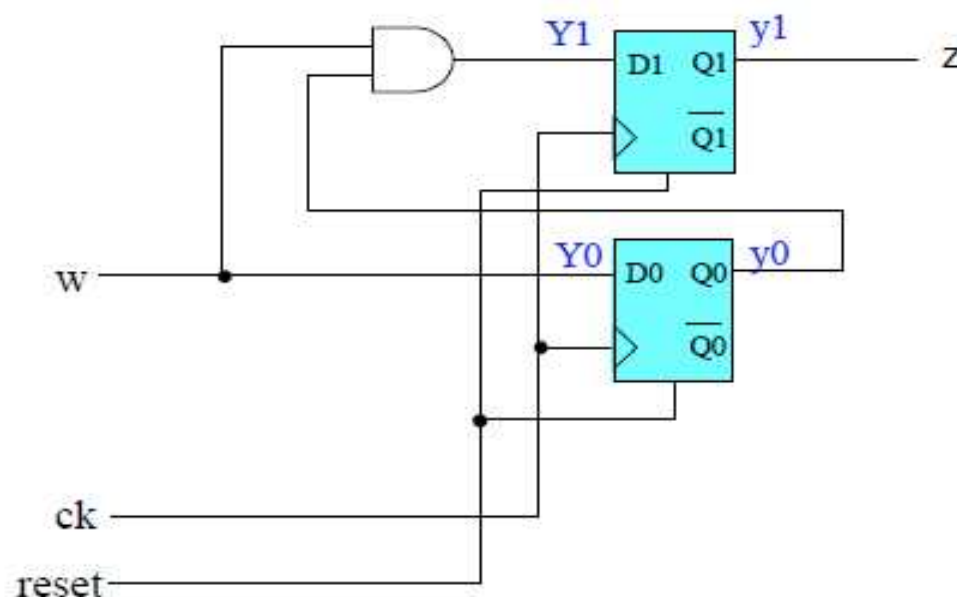
Coincidentemente, a equação de saída não mudou.



Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

Considerando o exemplo 2, porém com o assinalamento de estados
 $A=00$, $B=01$, $C=11$



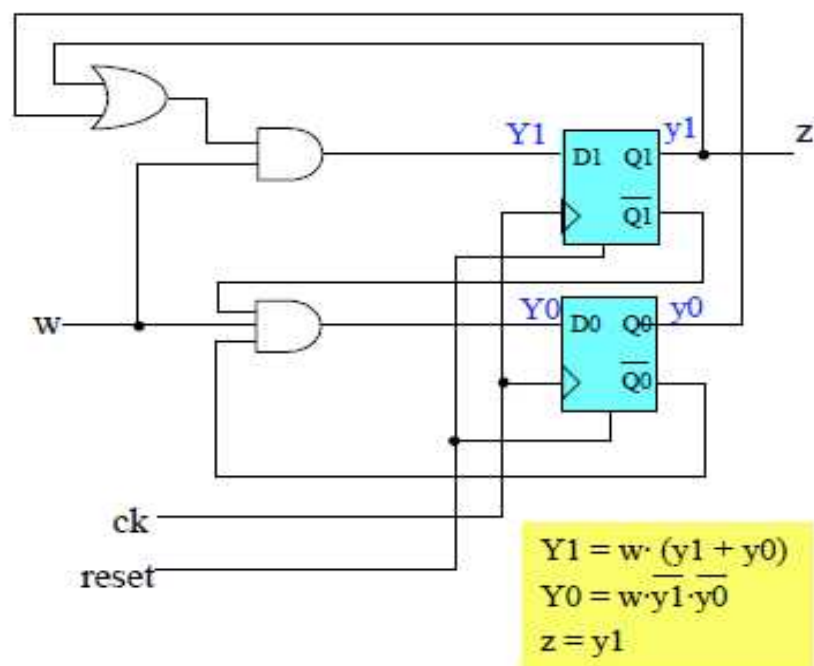
$$\begin{aligned} Y1 &= w \cdot y0 \\ Y0 &= w \\ z &= y1 \end{aligned}$$



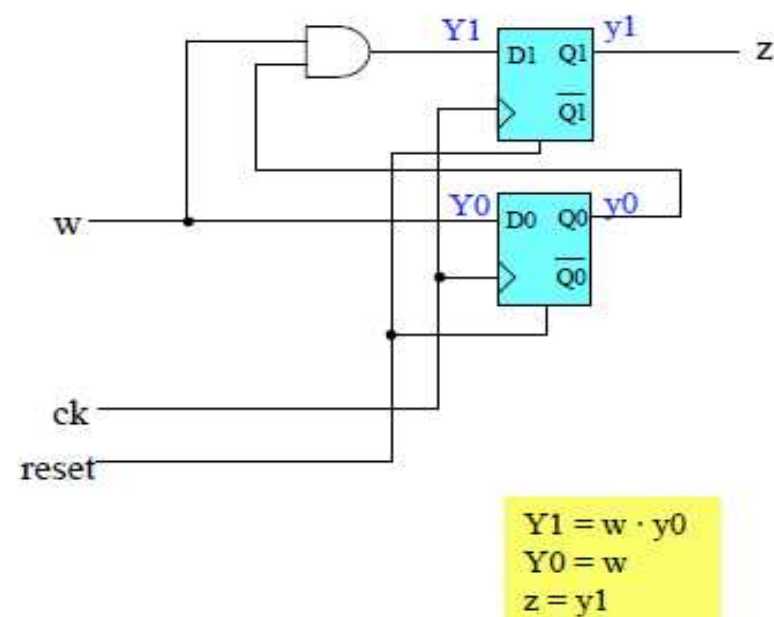
Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

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



Assinalamento A=00, B=01, C=11





Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

Considerando o exemplo 3, porém com o assinalamento de estados
 $A=00$, $B=01$, $C=11$, $D=10$

Estado atual	w	Próximo estado
A	0	A
A	1	B
B	X	C
C	X	D
D	X	A



	Estado atual y_1y_0	w	Próximo estado Y_1Y_0	
A	00	0	00	A
A	00	1	01	B
B	01	X	11	C
D	10	X	00	A
C	11	X	10	D



Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

Considerando o exemplo 3, porém com o assinalamento de estados
 $A=00$, $B=01$, $C=11$, $D=10$

	Estado atual y_1y_0	w	Próximo estado Y_1Y_0	
A	00	0	00	A
A	00	1	01	B
B	01	X	11	C
D	10	X	00	A
C	11	X	10	D

Y_1	$\bar{y}_1\bar{y}_0$	\bar{y}_1y_0	$y_1\bar{y}_0$	y_1y_0
\bar{w}	0	1	1	0
w	0	1	1	0

$$Y_1 = y_0$$

Y_0	$\bar{y}_1\bar{y}_0$	\bar{y}_1y_0	$y_1\bar{y}_0$	y_1y_0
\bar{w}	0	1	0	0
w	1	1	0	0

$$w \cdot \bar{y}_1$$

$$\bar{y}_1 \cdot y_0$$

$$Y_0 = w \cdot \bar{y}_1 + \bar{y}_1 \cdot y_0$$



Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

Considerando o exemplo 3, porém com o assinalamento de estados
 $A=00$, $B=01$, $C=11$, $D=10$

	Estado y_1y_0	R1out	R1in	R2out	R2in	R3out	R3in	Done
A	00	0	0	0	0	0	0	0
B	01	0	0	1	0	0	1	0
D	10	0	1	0	0	1	0	1
C	11	1	0	0	1	0	0	0

$$R1out = R2in = y_1 \cdot y_0$$

$$R1in = R3out = Done = y_1 \cdot \overline{y_0}$$

$$R2out = R3in = \overline{y_1} \cdot y_0$$



Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

Assinalamento A=00, B=01, C=11, D=10

$$Y1 = \overline{y1} \cdot y0 + y1 \cdot \overline{y0}$$

$$Y0 = w \cdot \overline{y0} + y1 \cdot \overline{y0}$$

$$R1_{out} = R2_{in} = y1 \cdot y0$$

$$R1_{in} = R3_{out} = \text{Done} = y1 \cdot y0$$

$$R2_{out} = R3_{in} = y1 \cdot y0$$

Assinalamento A=00, B=01, C=10, D=11

$$Y1 = y0$$

$$Y0 = w \cdot \overline{y1} + \overline{y1} \cdot y0$$

$$R1_{out} = R2_{in} = y1 \cdot y0$$

$$R1_{in} = R3_{out} = \text{Done} = y1 \cdot \overline{y0}$$

$$R2_{out} = R3_{in} = \overline{y1} \cdot y0$$



Máquinas Seqüenciais Síncronas

► Codificação de Estados

Codificação “One-Hot”

- Consiste em usar tantas variáveis de estado quantos forem os estados
- Cada estado é codificado de modo que somente uma das variáveis de estado vale “1” e todas as demais valem “0”



Máquinas Seqüenciais Síncronas

► Codificação de Estados

Aplicando a Codificação “One-Hot” ao Exemplo 2:

A=001, B=010, C=100

Estado atual	w	Próximo estado
A	0	A
A	1	B
B	0	A
B	1	C
C	0	A
C	1	C



	Estado atual y2 y1 y0	w	Próximo estado Y2 Y1 Y0	
A	001	0	001	A
A	001	1	010	B
B	010	0	001	A
B	010	1	100	C
C	100	0	001	A
C	100	1	100	C



As combinações de entrada não citadas
têm como próximo estado XXX



Máquinas Seqüenciais Síncronas

► Codificação de Estados

Aplicando a Codificação “One-Hot” ao Exemplo 2:

A=001, B=010, C=100

	Estado atual y2 y1 y0	w	Próximo estado Y2 Y1 Y0	
A	001	0	001	A
A	001	1	010	B
B	010	0	001	A
B	010	1	100	C
C	100	0	001	A
C	100	1	100	C

y2	$\overline{y_2}y_1$	$\overline{y_2}\overline{y_1}$	y_2y_1	$y_2\overline{y_1}$
$\overline{y_0}w$	X	0	X	0
$\overline{y_0}\overline{w}$	X	1	X	1
y_0w	0	X	X	X
$y_0\overline{w}$	0	X	X	X

$\overline{y_0} \cdot w$



Máquinas Seqüenciais Síncronas

► Codificação de Estados

Aplicando a Codificação “One-Hot” ao Exemplo 2:
A=001, B=010, C=100

	Estado atual $y_2 y_1 y_0$	w	Próximo estado $Y_2 Y_1 Y_0$	
A	001	0	001	A
A	001	1	010	B
B	010	0	001	A
B	010	1	100	C
C	100	0	001	A
C	100	1	100	C

y_1	$\overline{y_2} \overline{y_1}$	$\overline{y_2} y_1$	$y_2 \overline{y_1}$	$y_2 y_1$
$\overline{y_0} \overline{w}$	X	0	X	0
$\overline{y_0} w$	X	0	X	0
$y_0 \overline{w}$	1	X	X	X
$y_0 w$	0	X	X	X

$y_0 \cdot w$



Máquinas Seqüenciais Síncronas

► Codificação de Estados

Aplicando a Codificação “One-Hot” ao Exemplo 2:
A=001, B=010, C=100

	Estado atual y ₂ y ₁ y ₀	w	Próximo estado Y ₂ Y ₁ Y ₀	
A	001	0	001	A
A	001	1	010	B
B	010	0	001	A
B	010	1	100	C
C	100	0	001	A
C	100	1	100	C

Y ₀	$\overline{y_2}y_1$	$\overline{y_2}y_1$	$y_2\overline{y_1}$	$y_2\overline{y_1}$
$\overline{y_0}w$	X	1	X	1
$\overline{y_0}w$	X	0	X	0
y_0w	0	X	X	X
$y_0\overline{w}$	1	X	X	X



Máquinas Seqüenciais Síncronas

► Codificação de Estados

Aplicando a Codificação “One-Hot” ao Exemplo 2:

A=001, B=010, C=100

	Estado $y_2 y_1 y_0$	z
A	001	0
B	010	0
C	100	1



As combinações de entrada
não citadas têm como saída X

z	$\overline{y_2}\overline{y_1}$	$\overline{y_2}y_1$	$y_2\overline{y_1}$	y_2y_1
$\overline{y_0}$	X	0	X	1
y_0	0	X	X	X

y_2

$$z = y_2$$



Máquinas Seqüenciais Síncronas

► Minimização e Codificação de Estados

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

$$\begin{aligned}Y1 &= w \cdot (y1 + y0) \\ Y0 &= w \cdot y1 \cdot y0 \\ z &= y1\end{aligned}$$

Assinalamento A=00, B=01, C=11

$$\begin{aligned}Y1 &= w \cdot y0 \\ Y0 &= w \\ z &= y1\end{aligned}$$

Codificação “One Hot”

$$\begin{aligned}Y2 &= \overline{y0} \cdot w \\ Y1 &= y0 \cdot w \\ Y0 &= w \\ z &= y2\end{aligned}$$



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 2:

Projete um circuito que satisfaça às seguintes especificações:

1. O circuito possui uma entrada, **w**, e uma saída, **z**.
2. Todas as mudanças de valores no circuito ocorrem na borda de subida do sinal de relógio.
3. Quando o circuito detectar que a entrada **w** vale “0”, a saída **z** deve valer “0” no ciclo de relógio seguinte. Porém, quando o circuito detectar que a entrada **w** vale “1” durante duas bordas de relógio consecutivas, a saída **z** deve passar a valer “1” no ciclo de relógio seguinte à segunda ocorrência do valor “1”. As mudanças de **z** estão sincronizadas com a borda de relógio ativa.

Considere a seguinte modificação da especificação acima:

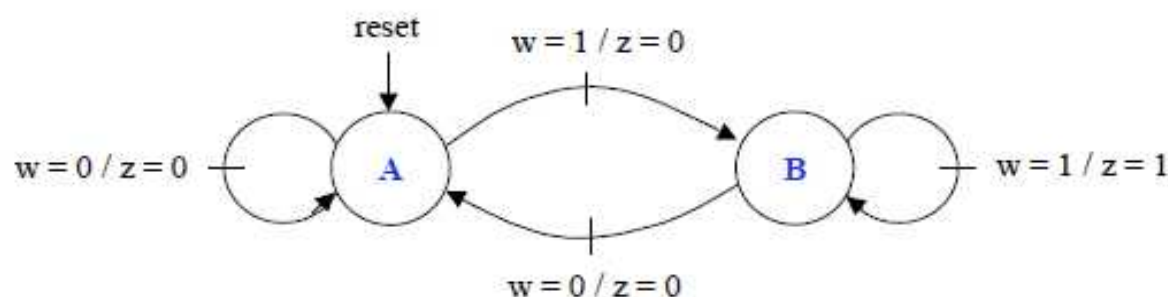
- O sinal de saída **z** não precisa esperar que um segundo valor igual a “1” seja amostrado da entrada **w**.
- Porém, se $z = 1$ e **w** muda de “1” para “0”, **z** deve também mudar para “0”, independentemente da borda ativa do relógio



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 2: Diagrama de Estados



Interpretação do Diagrama de Estados:

- Durante o ciclo de relógio atual, o valor da saída z corresponde ao rótulo assinalado em alguma das arestas que partem do estado atual.
- No caso do estado B, por exemplo, z pode valer “0” ou valer “1”, conforme for o valor de w . Isto implica que z pode mudar de valor antes que a máquina de estados mude de estado.



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 2: Tabelas de (Transição de) Estados e de Saída

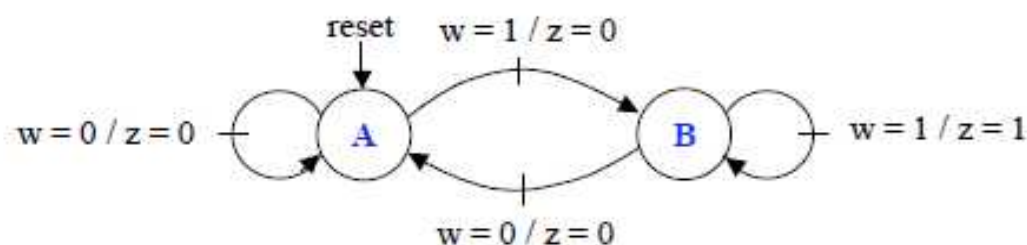


Tabela de (transição de) estados

Estado atual	w	Próximo estado
A	0	A
A	1	B
B	0	A
B	1	B

Tabela de saída

Estado	w	z
A	0	0
A	1	0
B	0	0
B	1	1



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 2: Projetando a lógica de próximo estado e a lógica de saída

- Como há somente dois estados, iremos utilizar somente uma variável de estado.
- Assumindo o seguinte assinalamento de estados: A=0, B=1

Lógica de Próximo Estado

	y	w	Y	
A	0	0	0	A
A	0	1	1	B
B	1	0	0	A
B	1	1	1	B

$$Y = \overline{y} \cdot w + y \cdot w = w$$

Lógica de Saída

y	w	z
0	0	0
0	1	0
1	0	0
1	1	1



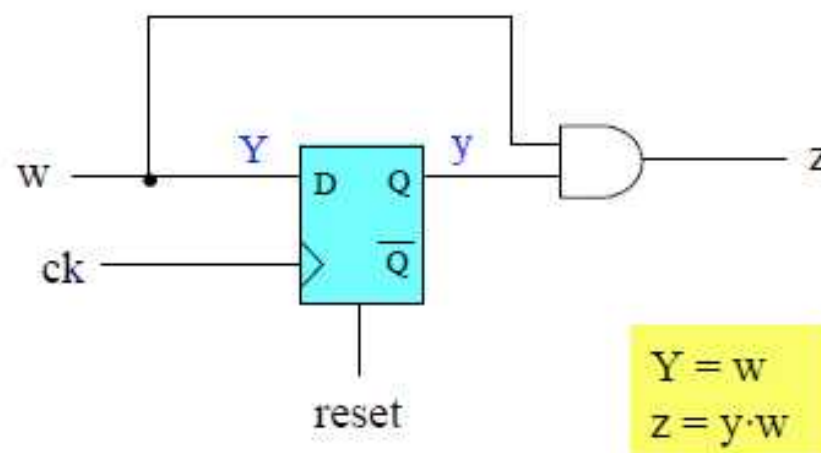
$$z = y \cdot w$$



Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 2: circuito final

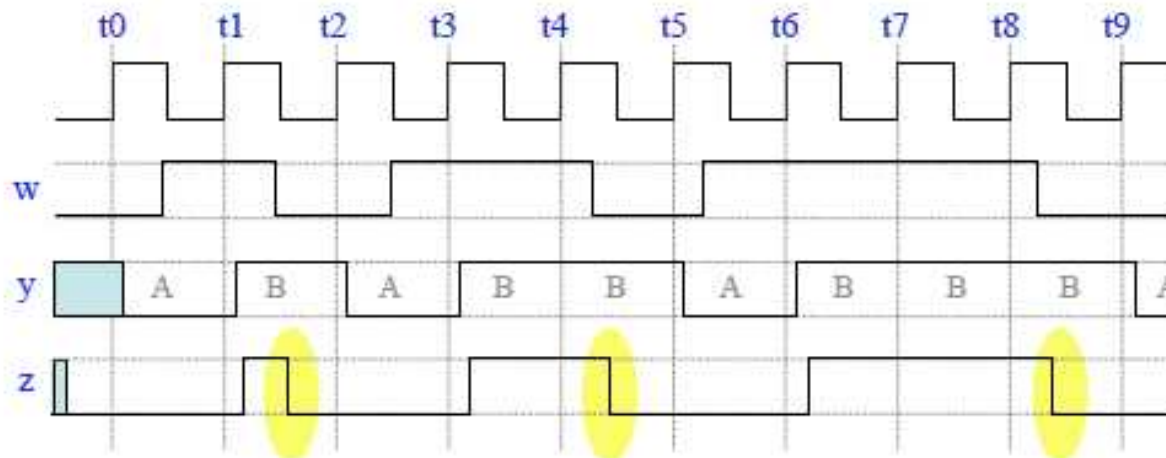




Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 2: exemplo de funcionamento dinâmico



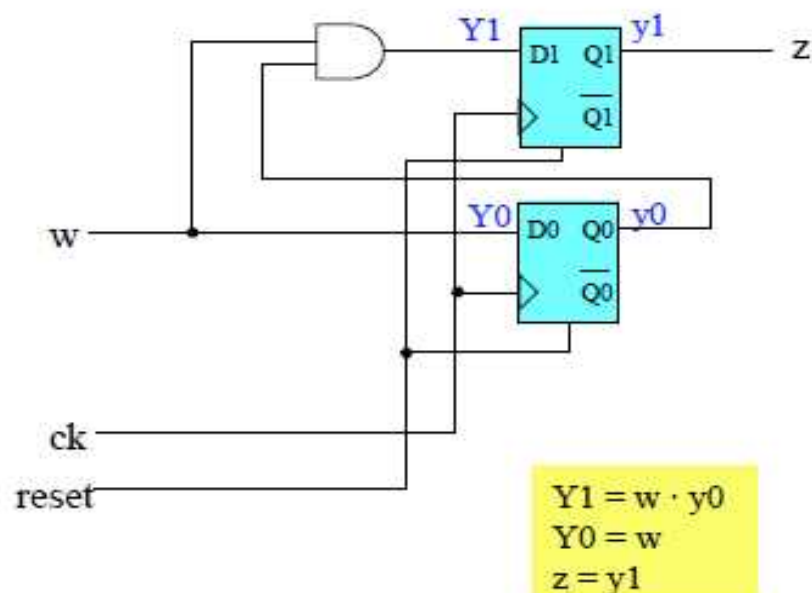


Máquinas Seqüenciais Síncronas

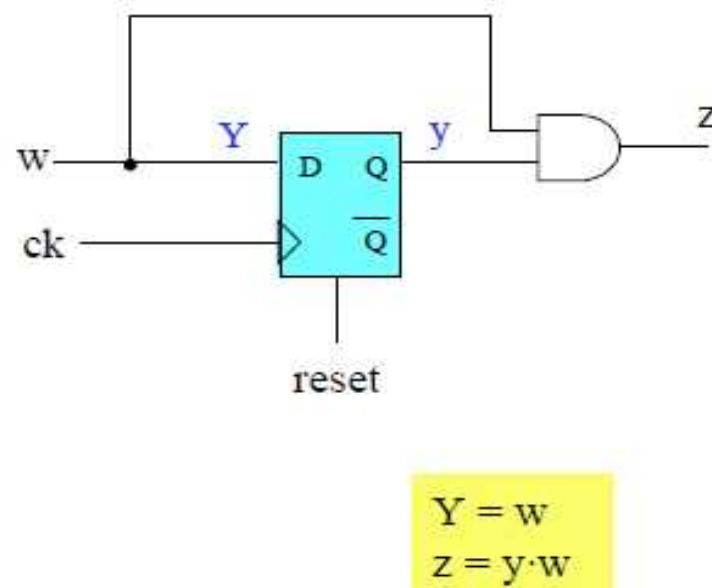
► Síntese de Circuitos Seqüenciais

Exemplo 2: Comparando Moore e Mealy

Moore



Mealy



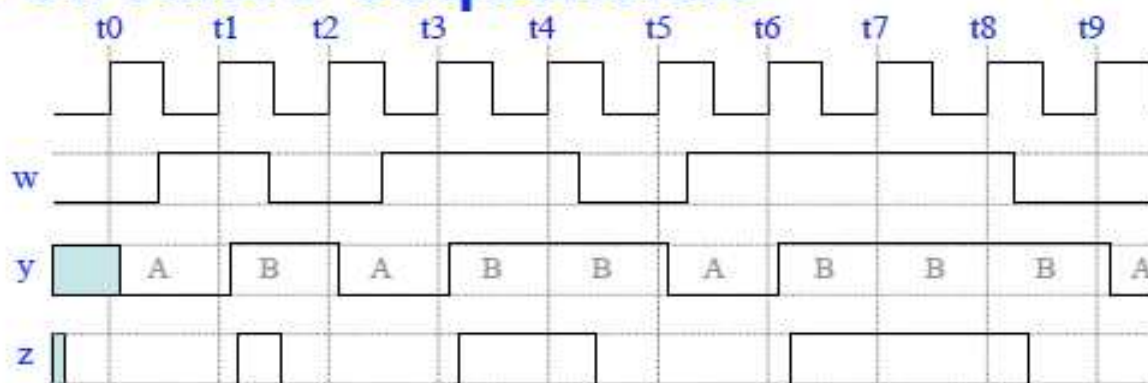


Máquinas Seqüenciais Síncronas

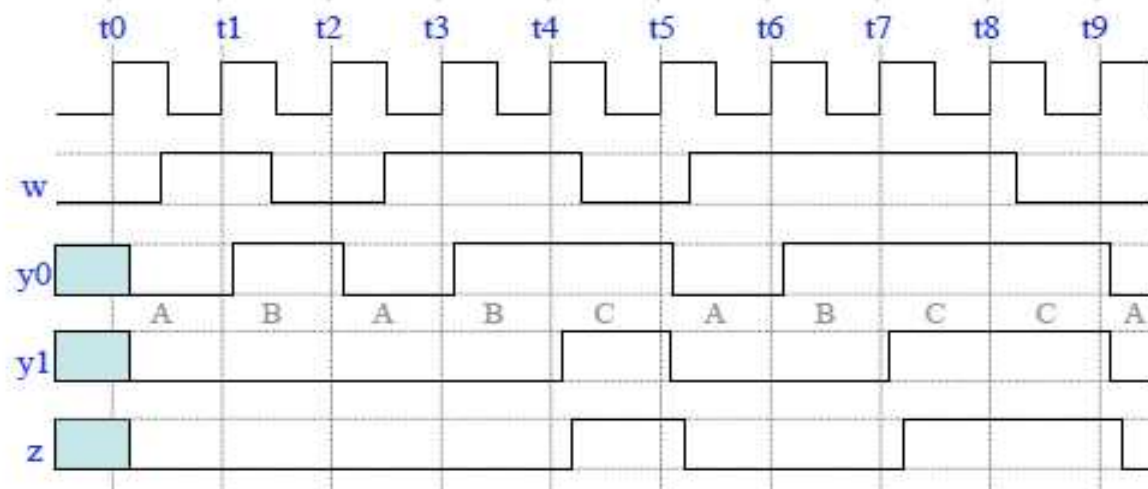
► Síntese de Circuitos Seqüenciais

Exemplo 2:

Mealy



Moore



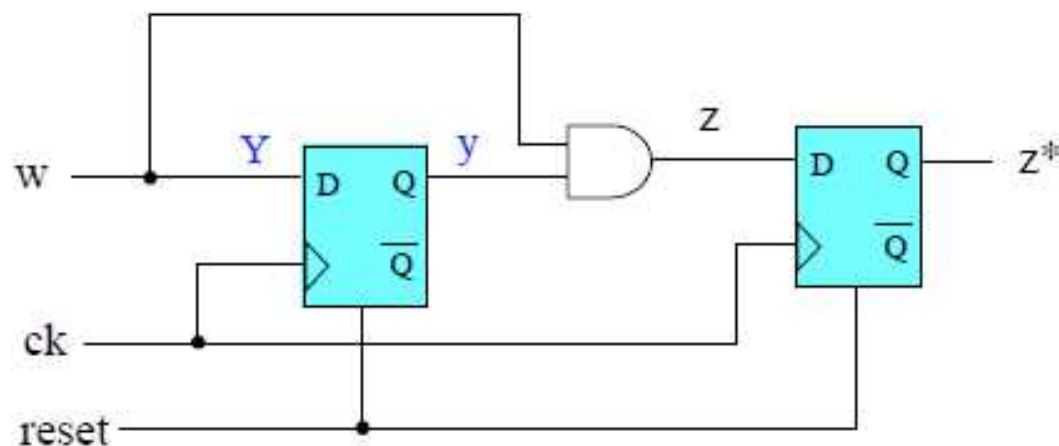


Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais

Exemplo 2:

Porém, se passarmos a saída z por um segundo flip-flop, filtraremos o comportamento assíncrono. De fato, estaremos transformando o circuito para o Modelo de Moore...



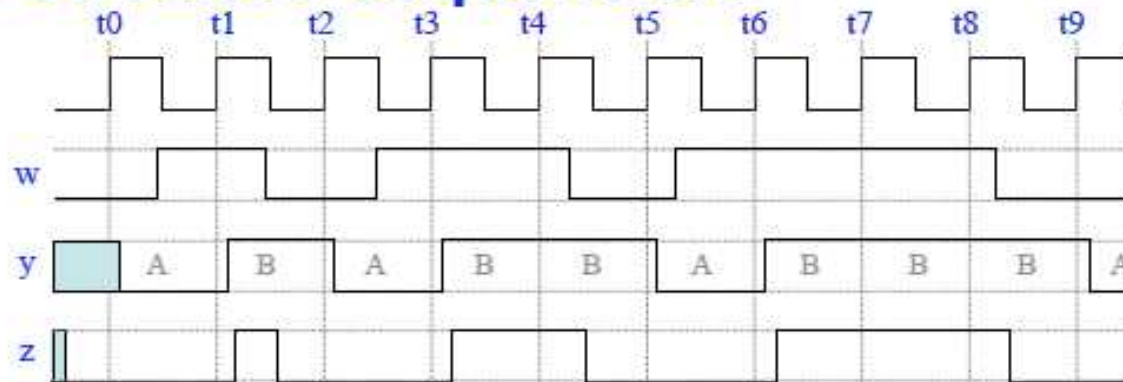


Máquinas Seqüenciais Síncronas

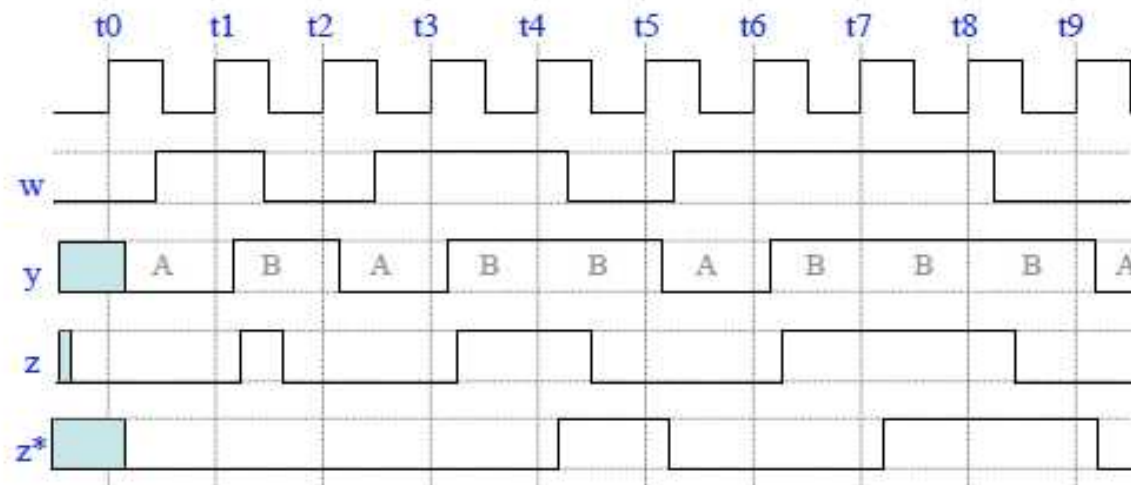
► Síntese de Circuitos Seqüenciais

Exemplo 2:

Mealy



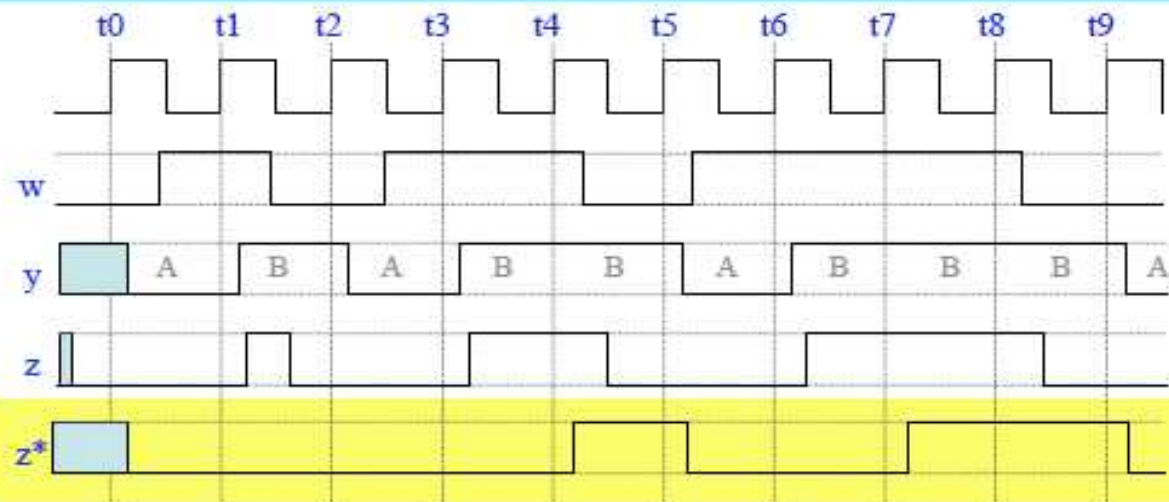
**Mealy
transform.
em Moore**



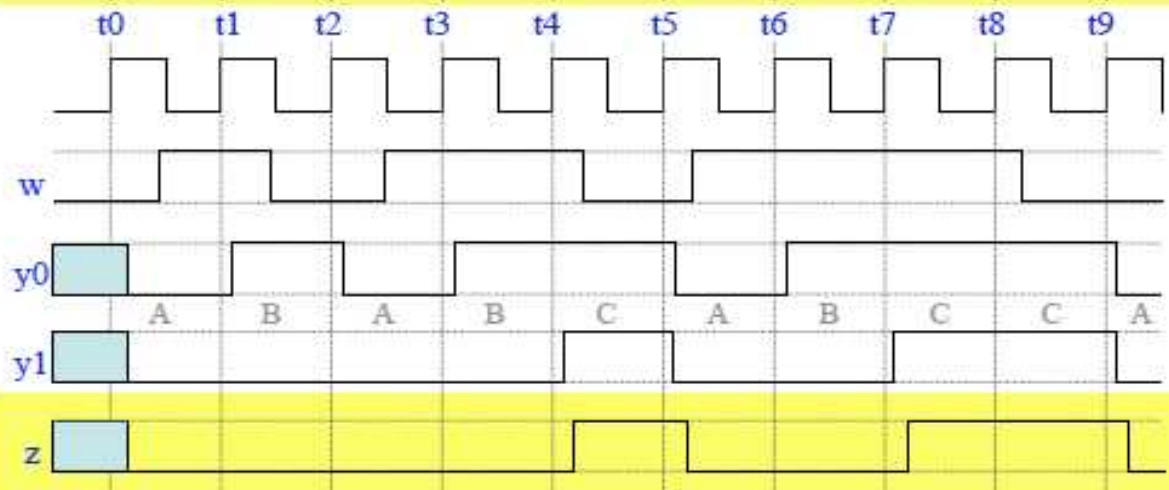


Máquinas Seqüenciais Síncronas

Mealy
transform.
em Moore



Moore





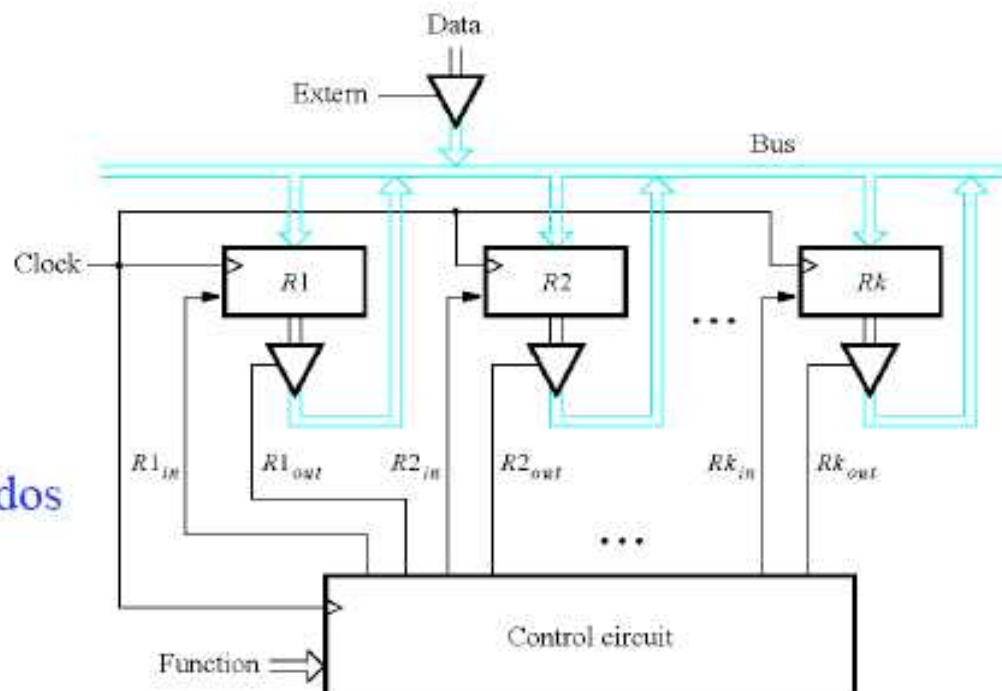
Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 3:

Passos:	Sinais de controle
$R3 \leftarrow R2;$	$R2_{out} = 1; R3_{in} = 1;$
$R2 \leftarrow R1;$	$R1_{out} = 1; R2_{in} = 1;$
$R1 \leftarrow R3;$	$R3_{out} = 1; R1_{in} = 1;$ $Done = 1;$

Obs: os sinais de saída não citados na tabela devem valer "0"



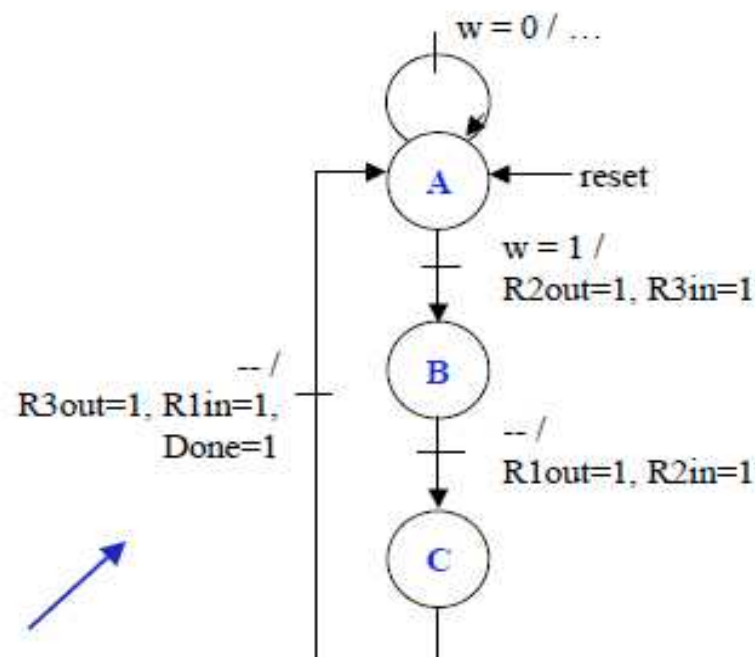


Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 3:

Passos:	Sinais de controle
$R3 \leftarrow R2;$	$R2_{out} = 1; R3_{in} = 1;$
$R2 \leftarrow R1;$	$R1_{out} = 1; R2_{in} = 1;$
$R1 \leftarrow R3;$	$R3_{out} = 1; R1_{in} = 1;$ $Done = 1;$



Obs: os sinais de saída não citados na tabela devem valer "0"

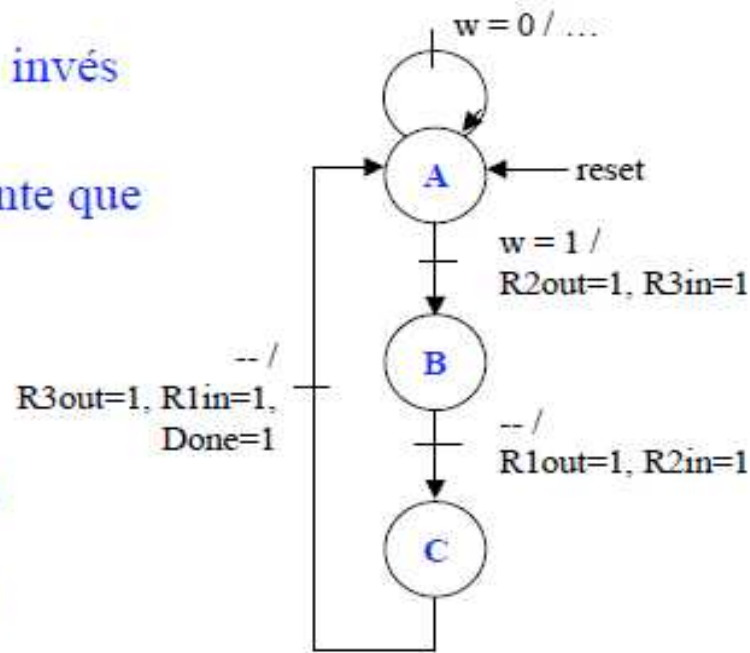


Máquinas Seqüenciais Síncronas

► Síntese de Circuitos Seqüenciais: Modelo de Mealy

Exemplo 3:

- A versão Mealy requer três estados (ao invés de quatro da versão Moore)
- Porém, isto não significa necessariamente que o circuito será menor, pois ainda são necessários dois flip-flops...
- A versão Mealy para o exemplo 3 gera os sinais de controle um ciclo de relógio mais cedo que a versão Moore
- Logo, para realizar o swap entre dois registradores a versão Mealy necessita de 3 ciclos de relógio, enquanto a versão Moore necessita de 4 ciclos.





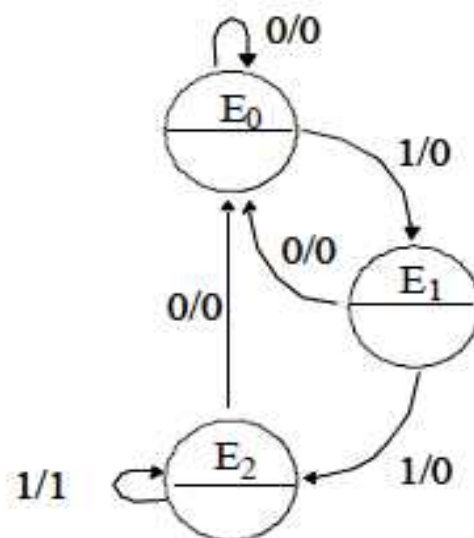
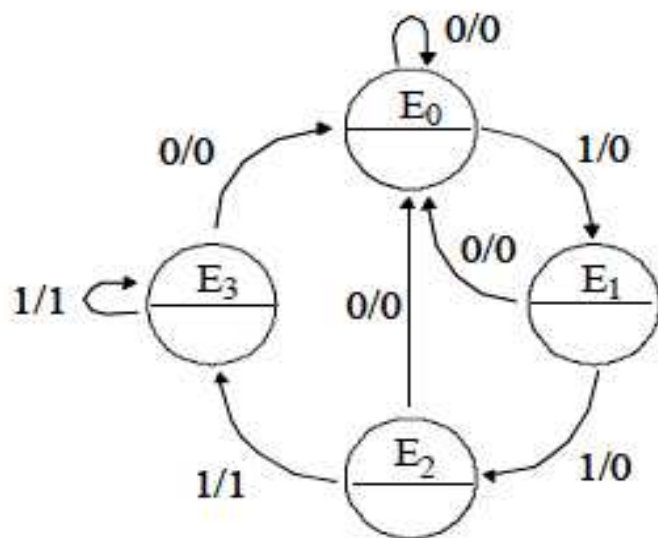
Otimização de Máquinas de Estado

- minimização de estados:
 - objetivo: reduzir o número de estados agrupando estados equivalentes
 - pode reduzir o tamanho do registrador de estados
 - pode reduzir a complexidade da lógica de interconexões
- atribuição de estados
 - objetivo: atribuir códigos aos estados de forma a simplificar a lógica de transição de estados e a lógica de geração de sinais de saída
 - evitar a geração de sinais espúrios e entrada em estados inválidos



Redução de Estados

- as duas máquinas abaixo tem o mesmo comportamento para qualquer sequência de entradas \Rightarrow máquinas equivalentes
- estados E_2 e E_3 produzem as mesmas saídas e próximos estados para todas as combinações de entrada \Rightarrow estados equivalentes, podem ser agrupados
- implementações diferentes !





Minimização de Estados

- procedimento: desenhar uma tabela de transição de estados
- identificar estados com o mesmo comportamento
- agrupar estados equivalentes criando novos estados para representá-los
- repetir o processo até não existir mais estados equivalentes
- métodos a serem analisados:
 - row matching (equivalência de linhas)
 - implication chart (diagramas de implicação)•



Equivalência de Linhas

- exemplo: considere uma máquina que deve reconhecer dois padrões de 4 bits *



- $Z(t) = 1$ se $X(t) = 0, X(t-1) = 1, X(t-2) = 1$ e $X(t-3) = 0$ ou
- $Z(t) = 1$ se $X(t) = 1, X(t-1) = 0, X(t-2) = 1$ e $X(t-3) = 0$
- máquina retorna ao estado inicial depois de qualquer sequência de 4 bits
- exemplo de comportamento:
 - $X = 0010\ 0110\ 1100\ 1010\ 0011\ \dots$
 - $Z = 0000\ 0001\ 0000\ 0001\ 0000\ \bullet$



Equivalência de Linhas

- Tabela Inicial

Input Sequence	Present State	Next State		Output	
		X=0	X=1	X=0	X=1
Reset	S ₀	S ₁	S ₂	0	0
0	S ₁	S ₃	S ₄	0	0
1	S ₂	S ₅	S ₆	0	0
00	S ₃	S ₇	S ₈	0	0
01	S ₄	S ₉	S ₁₀	0	0
10	S ₅	S ₁₁	S ₁₂	0	0
11	S ₆	S ₁₃	S ₁₄	0	0
000	S ₇	S ₀	S ₀	0	0
001	S ₈	S ₀	S ₀	0	0
010	S ₉	S ₀	S ₀	0	0
011	S ₁₀	S ₀	S ₀	1	0
100	S ₁₁	S ₀	S ₀	0	0
101	S ₁₂	S ₀	S ₀	1	0
110	S ₁₃	S ₀	S ₀	0	0
111	S ₁₄	S ₀	S ₀	0	0



Equivalência de Linhas ...

Input Sequence	Present State	Next State		Output	
		X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	S_3	S_4	0	0
1	S_2	S_5	S_6	0	0
00	S_3	S_7	S_8	0	0
01	S_4	S_9	S_{10}	0	0
10	S_5	S_{11}	S_{12}	0	0
11	S_6	S_{13}	S_{14}	0	0
000	S_7	S_0	S_0	0	0
001	S_8	S_0	S_0	0	0
010	S_9	S_0	S_0	0	0
011	S_{10}	S_0	S_0	1	0
100	S_{11}	S_0	S_0	0	0
101	S_{12}	S_0	S_0	1	0
110	S_{13}	S_0	S_0	0	0
111	S_{14}	S_0	S_0	0	0

- estados equivalentes:
substituir por S_{10}



Equivalência de Linhas ...

Input Sequence	Present State	Next State		Output	
		X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	S_3	S_4	0	0
1	S_2	S_5	S_6	0	0
00	S_3	S_7	S_8	0	0
01	S_4	S_9	S'_{10}	0	0
10	S_5	S_{11}	S'_{10}	0	0
11	S_6	S_{13}	S_{14}	0	0
000	S_7	S_0	S_0	0	0
001	S_8	S_0	S_0	0	0
010	S_9	S_0	S_0	0	0
011 or 101	S'_{10}	S_0	S_0	1	0
100	S_{11}	S_0	S_0	0	0
110	S_{13}	S_0	S_0	0	0
111	S_{14}	S_0	S_0	0	0

• estados equivalentes:
substituir por S_7



Equivalência de Linhas ...

Input Sequence	Present State	Next State		Output	
		X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	S_3	S_4	0	0
1	S_2	S_5	S_6	0	0
00	S_3	S'_7	S'_7	0	0
01	S_4	S'_7	S'_{10}	0	0
10	S_5	S'_7	S'_{10}	0	0
11	S_6	S'_7	S'_7	0	0
not(01 1 or 101)	S'_7	S_0	S_0	0	0
01 1 or 101	S'_{10}	S_0	S_0	1	0

- atualização dos próximos estados:
gera novas equivalências

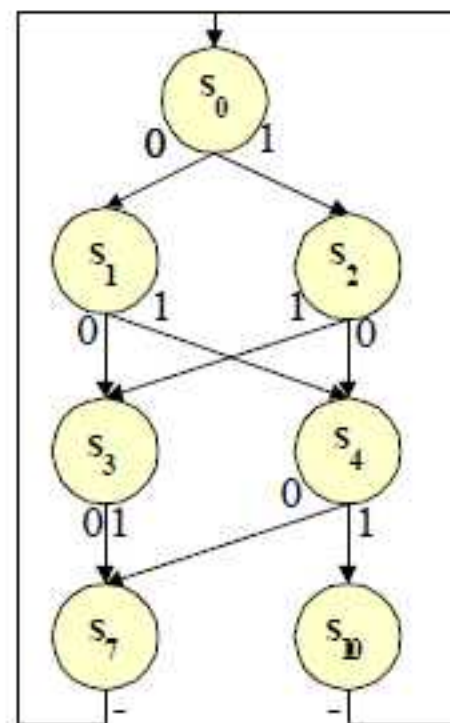




Equivalência de Linhas ...

Tabela Final:

Input Sequence	Present State	Next State		Output	
		X=0	X=1	X=0	X=1
Reset	S0	S1	S2	0	0
0	S1	S3'	S4'	0	0
1	S2	S4'	S3'	0	0
00 or 11	S3'	S7'	S7'	0	0
01 or 10	S4'	S7'	S10'	0	0
not(011 or 101)	S7'	S0	S0	0	0
011 or 101	S10'	S0	S0	1	0





Equivalência de Linhas ...

- vantagem:
 - fácil de entender e realizar
- desvantagem:
 - nem sempre produz o melhor resultado
 - exemplo de desvantagem: verificador de paridade

Present State	Next State		Output
	X=0	X=1	
S_0	S_0	S_1	0
S_1	S_1	S_2	1
S_2	S_2	S_1	0

não detecta equivalência entre S_0 e S_1

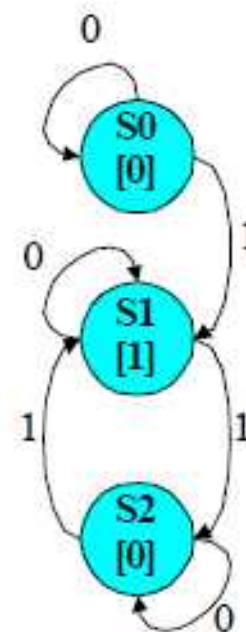




Diagrama de Implicações

- Outro Exemplo: circuito que reconhece sequências 010 e 110 *

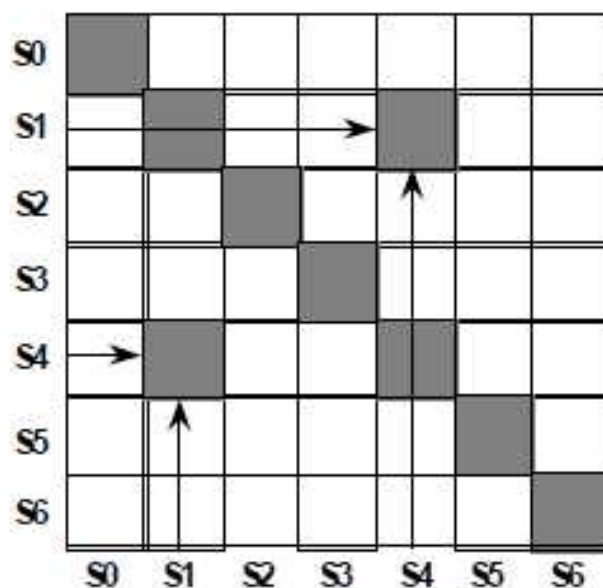
Input Sequence	Present State	Next State		Output	
		X=0	X=1	X=0	X=1
Reset	S_0	S_1	S_2	0	0
0	S_1	S_3	S_4	0	0
1	S_2	S_5	S_6	0	0
00	S_3	S_0	S_0	0	0
01	S_4	S_0	S_0	1	0
10	S_5	S_0	S_0	0	0
11	S_6	S_0	S_0	1	0



Diagrama de Implicações ...

- Enumerar todas as combinações de estados 2 a 2

próximos estados
para todas combi-
nações de entrada



simplificação: simétrico com
relação a diagonal

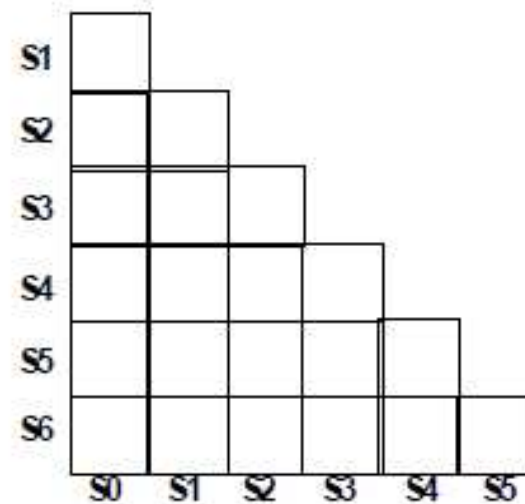


diagrama de implicações



Diagrama de Implicações ...

- preenchimento:
 - entrada X_{ij} : linha é S_i , coluna é S_j
 - S_i é equivalente a S_j se ambos tem as mesmas saídas e próximos estados equivalentes
 - os próximos estados de S_i e S_j são colocados dentro de X_{ij}
 - se S_i e S_j tem saídas diferentes, a entrada X_{ij} é marcado com um "x"
- exemplo:
 - S_0 transiciona para S_1 em 0, S_2 em 1;
 - S_1 transiciona para S_3 em 0, S_4 em 1;
 - a entrada X_{01} contém: S_1 - S_3 , S_2 - S_4 , que são os pares de estados que devem ser equivalentes para que S_0 e S_1 também o sejam

S_0

S_1 - S_3 S_2 - S_4

S_1



Diagrama de Implicações ...

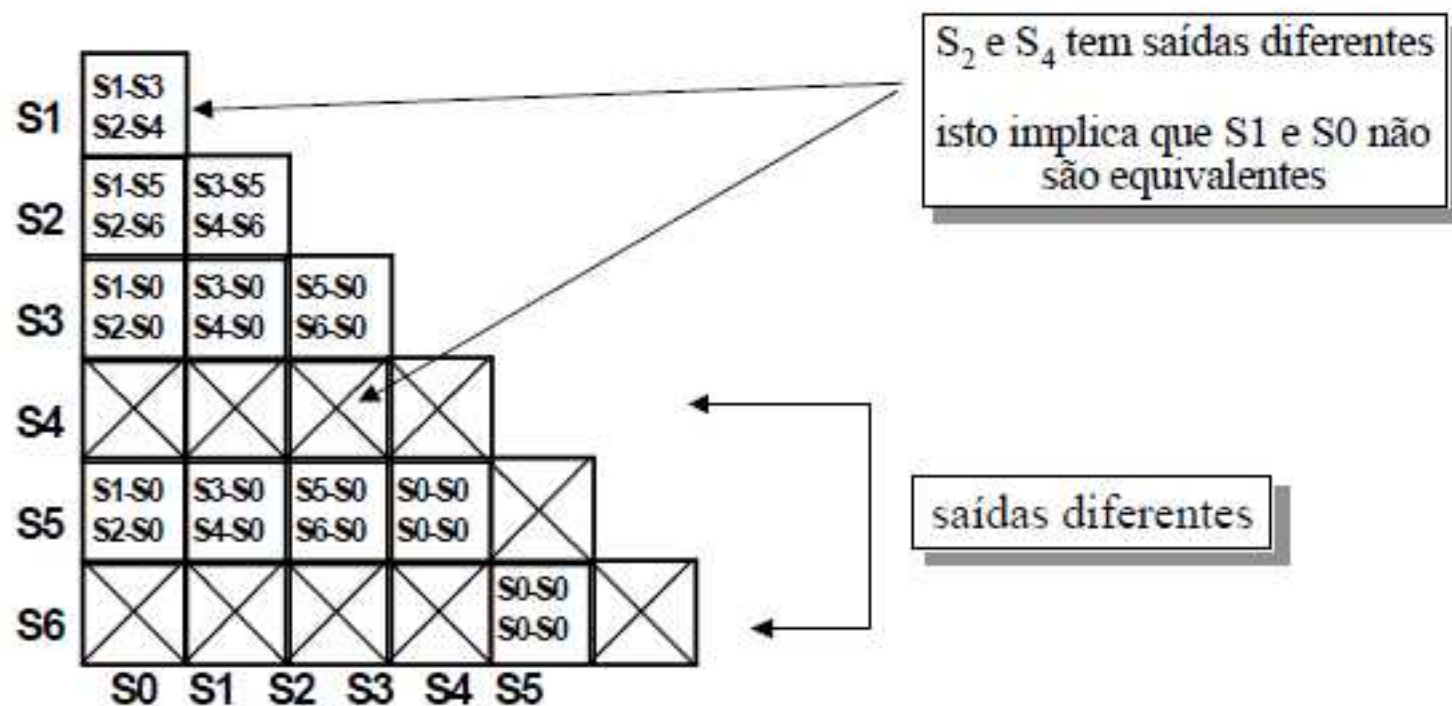
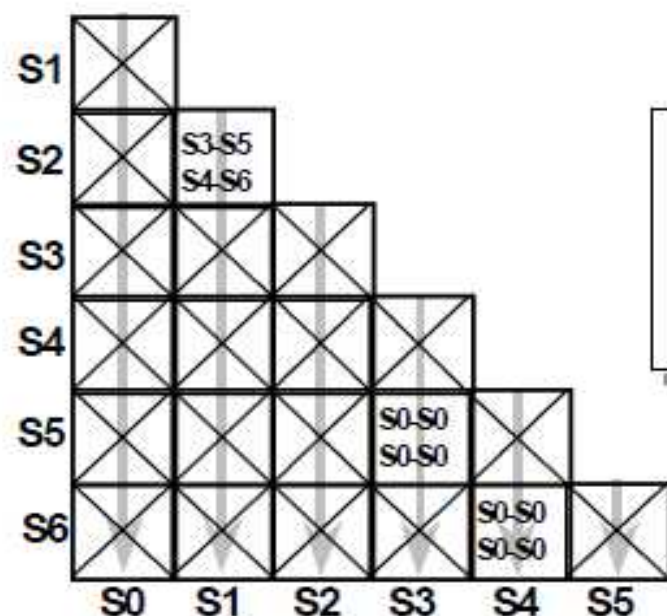


Diagrama Inicial



Diagrama de Implicações ...



- resultado do primeiro passo:
S₃ e S₅ são equivalentes
S₄ e S₆ são equivalentes
=> S₁ e S₂ são equivalentes
(implicação)

tabela reduzida

Input Sequence	Present State	Next State		Output	
		X=0	X=1	X=0	X=1
Reset	S ₀	S ₁ '	S ₁ '	0	0
0 or 1	S ₁ '	S ₃ '	S ₄ '	0	0
00 or 10	S ₃ '	S ₀	S ₀	0	0
01 or 1 1	S ₄ '	S ₀	S ₀	1	0



Máquinas com Múltiplas Entradas

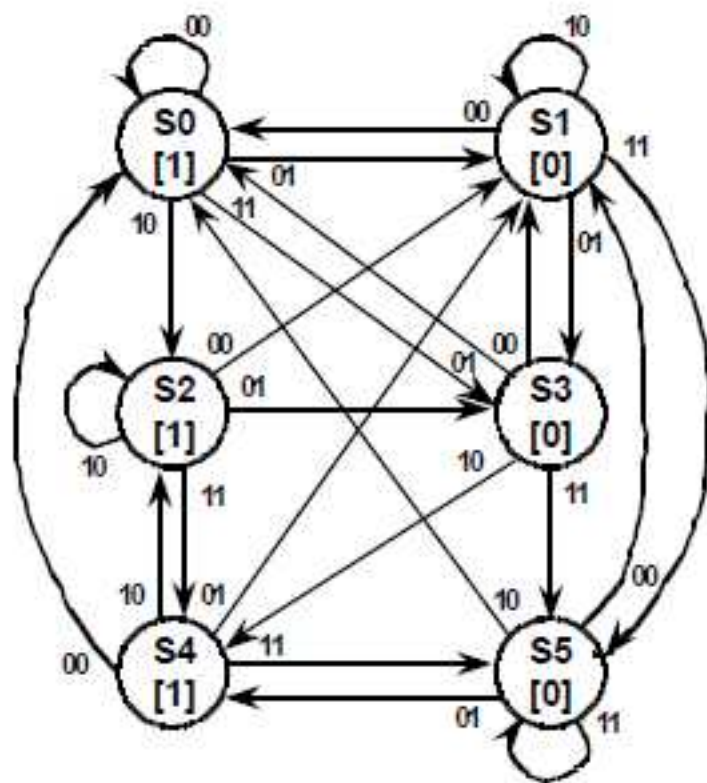


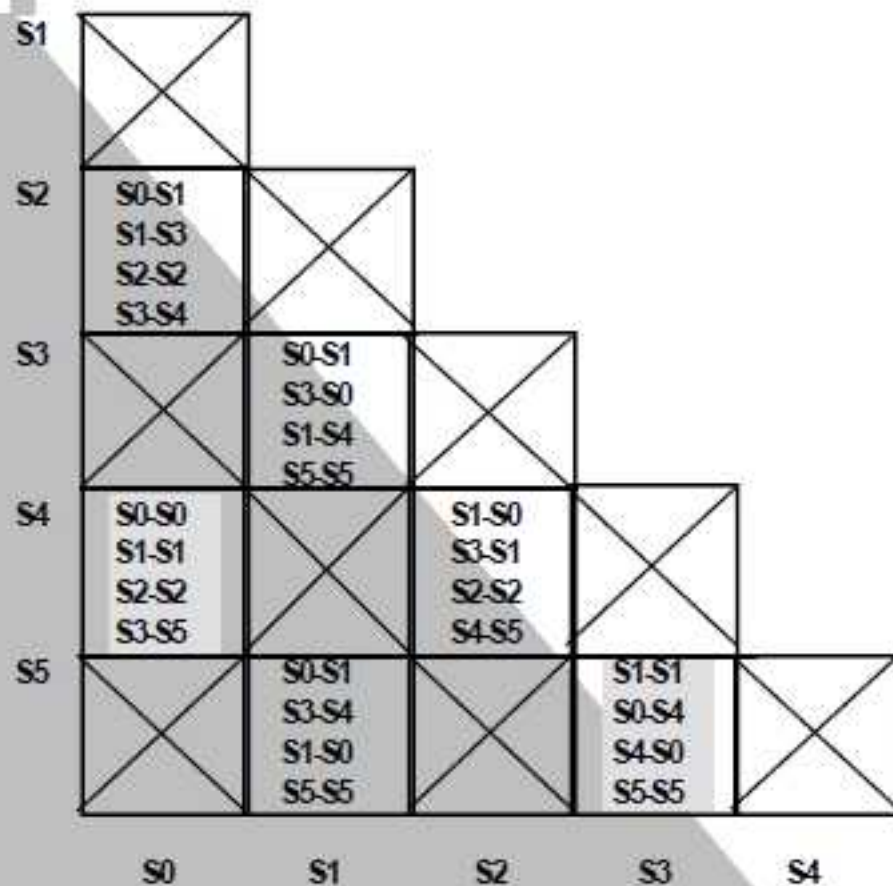
Diagrama de estados

Present State	Next State				Output
	00	01	10	11	
S ₀	S ₀	S ₁	S ₂	S ₃	1
S ₁	S ₀	S ₃	S ₁	S ₅	0
S ₂	S ₁	S ₃	S ₂	S ₄	1
S ₃	S ₁	S ₀	S ₄	S ₅	0
S ₄	S ₀	S ₁	S ₂	S ₅	1
S ₅	S ₁	S ₄	S ₀	S ₅	0

Tabela de Estados



Máquinas com Múltiplas Entradas ...



Present State	Next State				Output
	00	01	10	11	
S_0'	S_0'	S_1	S_2	S_3'	1
S_1	S_0'	S_3'	S_1	S_3'	0
S_2	S_1	S_3'	S_2	S_0'	1
S_3'	S_1	S_0'	S_0'	S_3'	0

Tabela Máquina Minimizada

Diagrama de Implicação



Referências

- Curso oficial da Altera "Advanced Verilog Design Techniques"