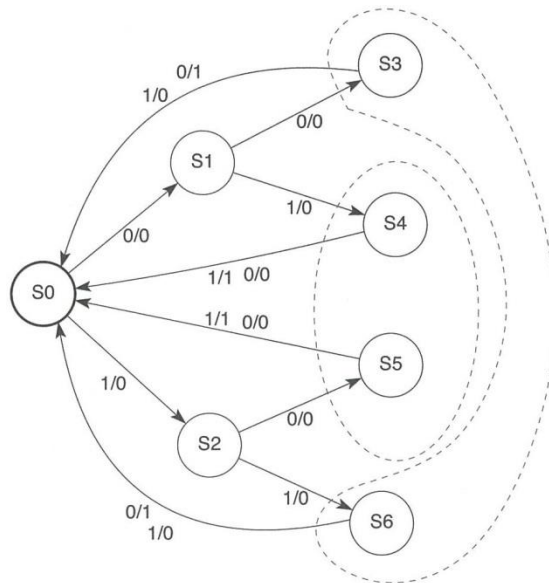


Sistemas Digitais (SD)

Síntese de Circuitos Sequenciais: Minimização do Número de Estados



S1	S1-S3 S2-S4					
S2	S1-S5 S2-S6	S3-S5 S4-S6				
S3	×	×	×			
S4	×	×	×	×		
S5	×	×	×	×	✓	
S6	×	×	×	✓	×	×
	S0	S1	S2	S3	S4	S5

■ Na aula anterior:

- ▶ Definição de circuito sequencial síncrono
- ▶ Máquinas de Mealy e de Moore
- ▶ Especificação de circuitos sequenciais síncronos:
 - Diagrama de estados
- ▶ Projecto de circuitos sequenciais síncronos:
 - Codificação dos estados
 - Tabela de transição de estados
 - Determinação das funções lógicas de saída e estado seguinte



SEMANA	TEÓRICA 1	TEÓRICA 2	PROBLEMAS/LABORATÓRIO
17/Fev a 21/Fev	Introdução	Sistemas de Numeração	
24/Fev a 28/Fev	CARNAVAL	Álgebra de Boole	P0
02/Mar a 06/Mar	Elementos de Tecnologia	Funções Lógicas	VHDL
9/Mar a 13/Mar	Minimização de Funções	Minimização de Funções	L0
16/Mar a 20/Mar	Def. Circuito Combinatório; Análise Temporal	Circuitos Combinatórios	P1
23/Mar a 27/Mar	Circuitos Combinatórios	Circuitos Combinatórios	L1
30/Mar a 03/Abr	Circuitos Sequenciais: Latches	Circuitos Sequenciais: Flip-Flops	P2
06/Abr a 10/Abr	FÉRIAS DA PÁSCOA	FÉRIAS DA PÁSCOA	FÉRIAS DA PÁSCOA
13/Abr a 17/Abr	Caracterização Temporal	Registos	L2
20/Abr a 24/Abr	Contadores	Circuitos Sequenciais Síncronos	P3
27/Abr a 01/Mai	Síntese de Circuitos Sequenciais Síncronos	Síntese de Circuitos Sequenciais Síncronos	L3
04/Mai a 08/Mai	Exercícios	Memórias	P4
11/Mai a 15/Mai	Máq. Estado Microprogramadas: Circuito de Dados e Circuito de Controlo	Máq. Estado Microprogramadas: Microprograma	L4
18/Mai a 22/Mai	Circuitos de Controlo, Transferência e Processamento de Dados de um Processador	Lógica Programável	P5
25/Mai a 29/Mai	P6	P6	L5

Teste 1

■ Tema da aula de hoje:

- ▶ Especificação e projecto de circuitos sequenciais síncronos:
 - Minimização do número de estados
- ▶ Exemplo (Mealy)

□ Bibliografia:

- M. Mano, C. Kime: Secções 5.4 a 5.7
- G. Arroz, J. Monteiro, A. Oliveira: Secção 7.1 a 7.4

■ Revisão: circuito combinatório vs circuito sequencial

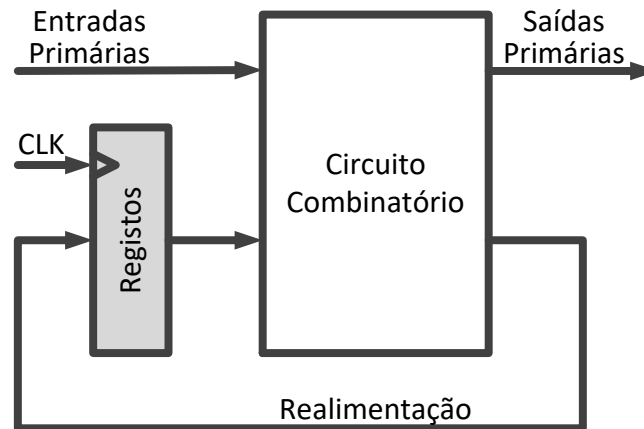
▶ Circuito Combinatório

- O valor da saída depende apenas do valor nas entradas nesse instante

▶ Circuito Sequencial

- O valor da saída depende do valor actual nas entradas, bem como da história anterior dos estados do circuito
 - Como? → através de elementos de memória (ex: latches e flip-flops)
- Podem ser divididos em:
 - **Síncronos**: o sinal de relógio sincroniza toda a actividade do circuito
 - **Assíncronos**: não usam sinal de relógio – as transições de estado ocorrem sempre que há uma alteração nas entradas do circuito

■ Revisão: Circuito Sequencial Síncrono

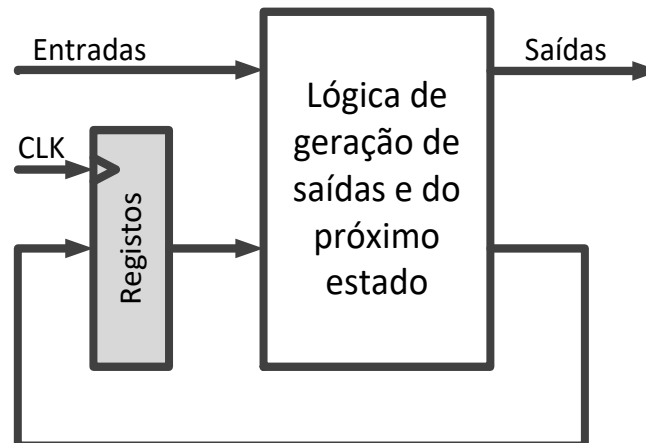


► Duas componentes:

- Bloco de lógica puramente combinatória
 - Implementa as funções necessárias para que o circuito tenha a transição entre estados pretendida
- Elementos de memória, controlados por um sinal de relógio
 - Mantém o estado do circuito ao longo do tempo

■ Revisão: Máquinas de Moore vs. Máquinas de Mealy

- ▶ As máquinas de estado síncronas podem ser divididas em:
 - **Máquinas de Moore:** a saída depende *apenas* das variáveis de estado actuais;
 - **Máquinas de Mealy:** a saída é função das variáveis de estado actuais e do valor das entradas presentes no circuito



■ Projecto de Circuitos Sequenciais Síncronos

► Procedimento:

■ Especificação formal:

- Diagrama de estados
- Fluxograma

■ Simplificação da especificação

■ Projecto:

1. Codificação dos estados
2. Tabelas de transição de estados
3. Determinação das funções lógicas de saída e estado seguinte

■ Projecto de Circuitos Sequenciais Síncronos

► Procedimento:

■ Especificação formal:

- Diagrama de estados
- Fluxograma

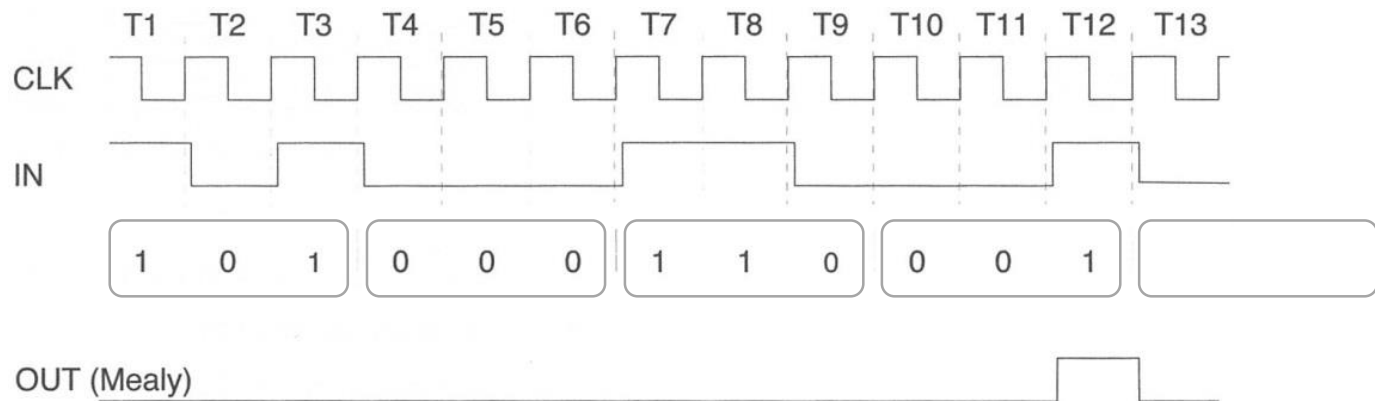
■ **Simplificação da especificação**

■ Projecto:

1. Codificação dos estados
2. Tabelas de transição de estados
3. Determinação das funções lógicas de saída e estado seguinte

■ Exemplo – Detector de Paridade

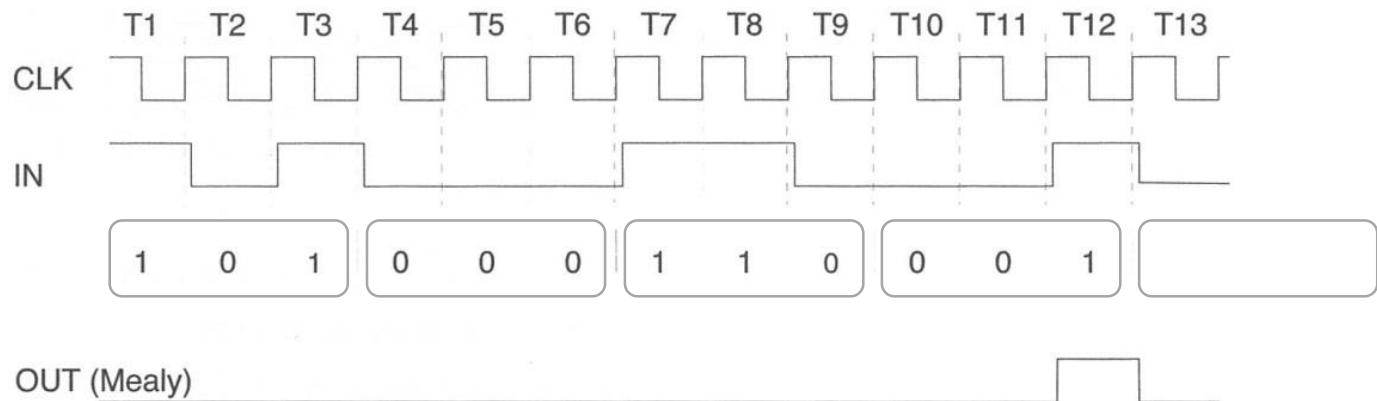
- ▶ Pretende-se enviar dados por uma linha, em grupos de 3 bits. A linha está sujeita a ruído, pelo que se implementou um protocolo de detecção de erros que garante que cada grupo de 3 bits tem um número par de bits a 1.
- ▶ O circuito sequencial pretendido deverá assinalar na sua saída sempre que ocorrer um erro de transmissão, identificado por um número ímpar de bits a 1 num grupo de 3 bits



■ Exemplo – Detector de Paridade

► Problema:

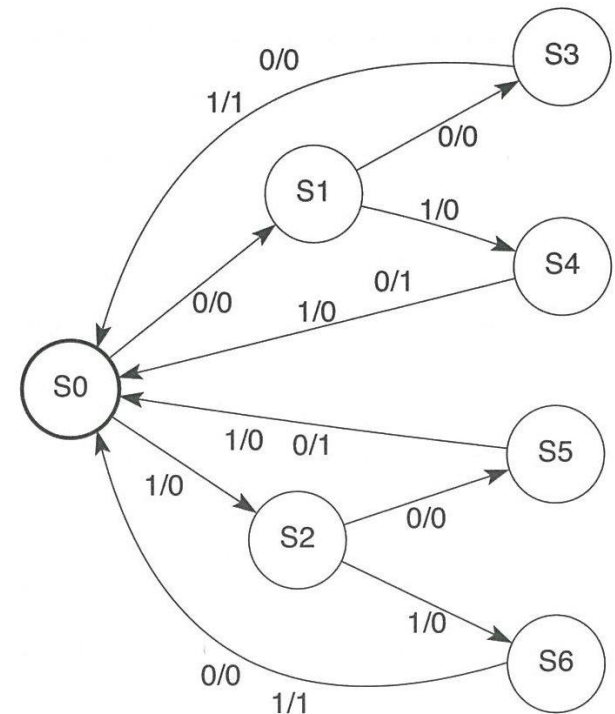
- Como construir o diagrama de estados?
- A solução é única?
- É possível otimizar o número de estados?



■ Construção do Diagrama de Estados

- O diagrama de estados pode ser construído directamente a partir da definição do problema:

- Enumerar todas as possíveis combinações de estados que podem ocorrer a partir do estado de Reset (S0), e gerar o valor 1 na saída quando o número de 1's for ímpar, retornando ao estado S0 para processar a próxima sequência de 3 bits.

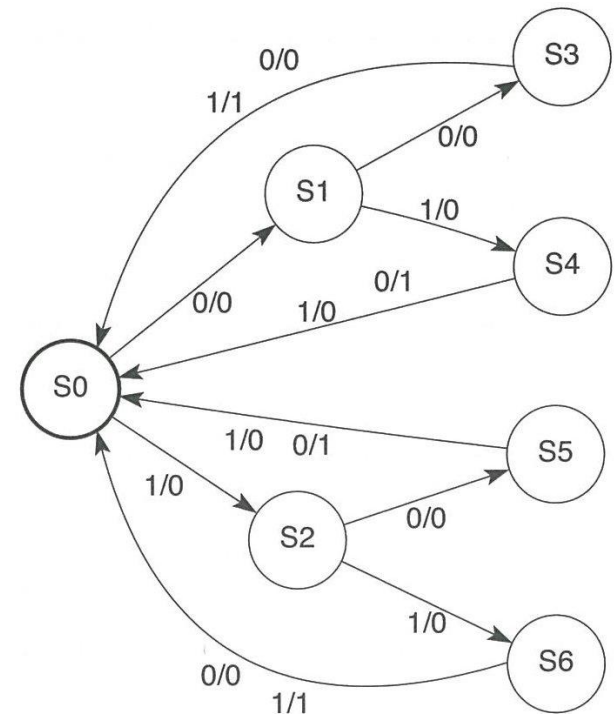


■ Construção do Diagrama de Estados

- ▶ Este diagrama de estados pode ser simplificado?
- ▶ Existem estados equivalentes que podem ser fundidos?

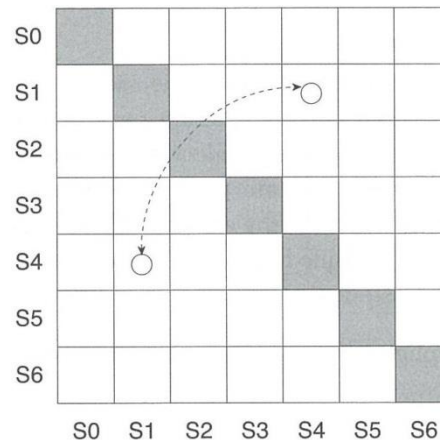
Definição:

Dois estados dizem-se **estados equivalentes** se, e só se, para cada combinação possível nas entradas, eles geram a mesma saída e transitam para o mesmo estado ou para estados que também sejam equivalentes.



■ Tabela de Implicações

- ▶ Uma linha e uma coluna por estado;
- ▶ Indica quais os pares de estados que são equivalentes;
- ▶ Uma vez que esta matriz é simétrica, apenas se considera a componente inferior à diagonal principal:
 - Exemplo: $S1 \leftrightarrow S4 \Leftrightarrow S4 \leftrightarrow S1$

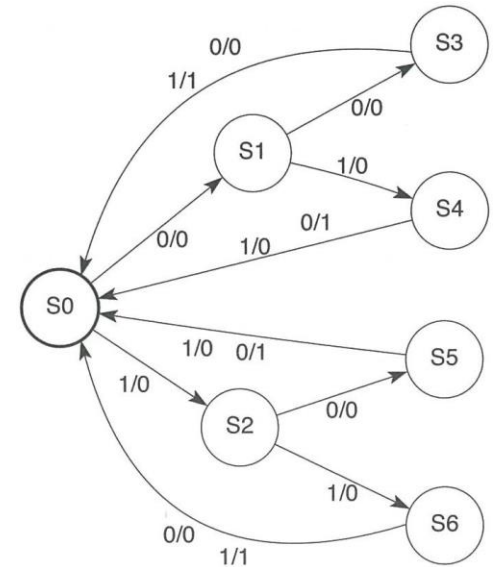


■ Tabela de Implicações

- **1º Passo:** identificar os pares de estados que não podem ser equivalentes, porque geram saídas diferentes para a mesma entrada.

Exemplos:

- Os estados S0 e S6 são necessariamente não equivalentes, porque geram saídas diferentes para a entrada 1
- O estado S0, cuja saída é sempre 0, não pode ser equivalente ao estado S3, pois este gera 1 na sua saída quando a entrada é 1;
Idem para os estados S4, S5 ou S6.



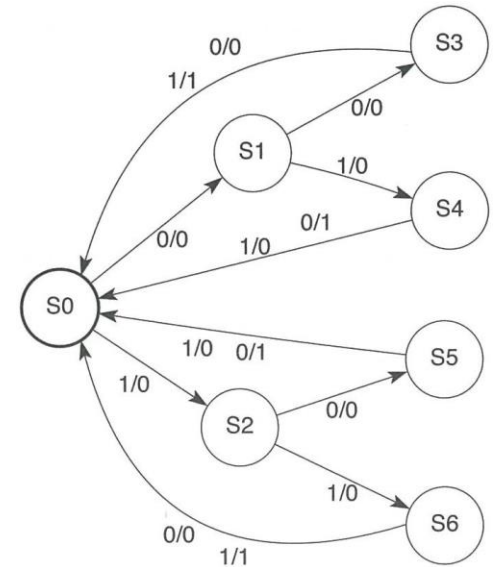
S1						
S2						
S3	×	×	×			
S4	×	×	×	×		
S5	×	×	×	×		
S6	×	×	×		×	×
	S0	S1	S2	S3	S4	S5

■ Tabela de Implicações

- **2º Passo:** identificar os pares de estados que são equivalentes, pois não só geram as mesmas saídas, como transitam para os mesmos estados (ou equivalentes).

Exemplos:

- Os estados S4 e S5 são equivalentes, porque têm as mesmas saídas, para ambos os valores de entrada, e transitam para o estado S0, para ambos os valores de entrada;
Idem para os estados S3 e S6.



S1						
S2						
S3	×	×	×			
S4	×	×	×	×		
S5	×	×	×	×	✓	
S6	×	×	×	✓	×	×
	S0	S1	S2	S3	S4	S5

■ Tabela de Implicações

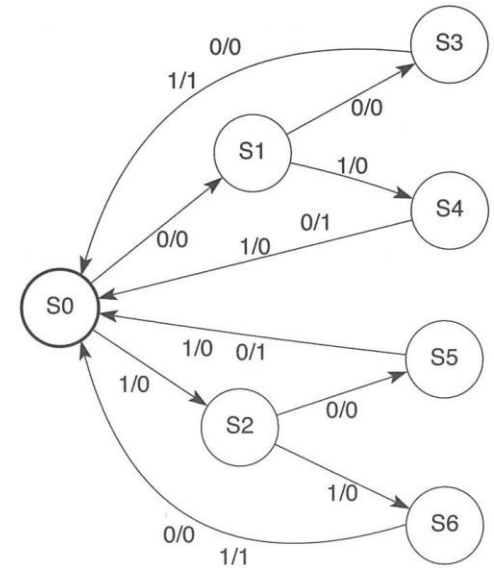
- **3º Passo:** identificar os pares de estados que poderão ser equivalentes caso outros pares também o sejam.

Exemplos:

- Os estados S1 e S2 apenas poderão ser equivalentes se os estados S3 e S5 forem equivalentes e se os estados S4 e S6 forem equivalentes



a entrada correspondente ao par (S1,S2) deve ser preenchida com os pares (S3,S5) e (S4,S6)



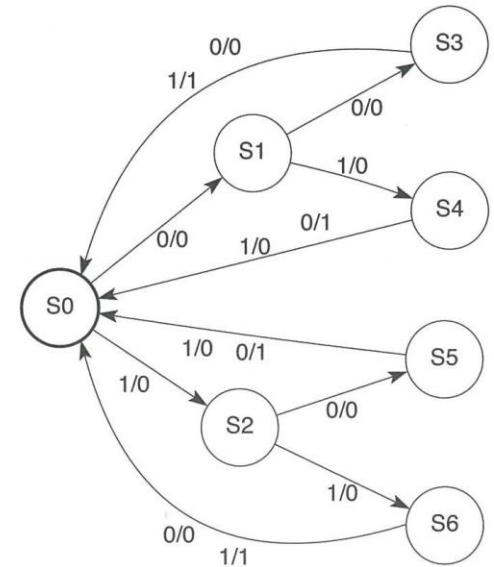
S1	S1-S3 S2-S4					
S2	S1-S5 S2-S6	S3-S5 S4-S6				
S3	×	×	×			
S4	×	×	×	×		
S5	×	×	×	×	✓	
S6	×	×	×	✓	×	×
	S0	S1	S2	S3	S4	S5

■ Tabela de Implicações

- **4º Passo:** eliminar, através de passagens sucessivas da tabela, os elementos que não podem ser equivalentes, dado que a sua equivalência depende da equivalência de outros estados que a tabela indica como não sendo equivalentes.

Exemplos:

- O par (S0,S1) não pode ser equivalente, porque depende do par (S1,S3) que a tabela mostra como sendo não equivalente
- O par (S0,S2) não pode ser equivalente, porque depende do par (S1,S5) que a tabela mostra como sendo não equivalente
- O par (S1,S2) não pode ser equivalente, porque depende do par (S3,S5) que a tabela mostra como sendo não equivalente

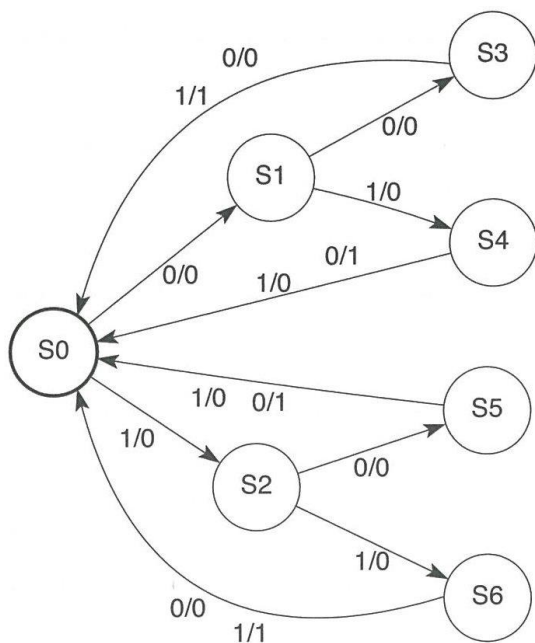


S1	S0-S1	S0-S2	S0-S3			
S2	S1-S2	S1-S3	S1-S4	S1-S5		
S3	X	X	X			
S4	X	X	X	X		
S5	X	X	X	X	✓	
S6	X	X	X	✓	X	X
	S0	S1	S2	S3	S4	S5

■ Simplificação do Diagrama de Estados

► De acordo com o processo de simplificação realizado, conclui-se que:

- O estado S3 é equivalente ao estado S6
- O estado S4 é equivalente ao estado S5

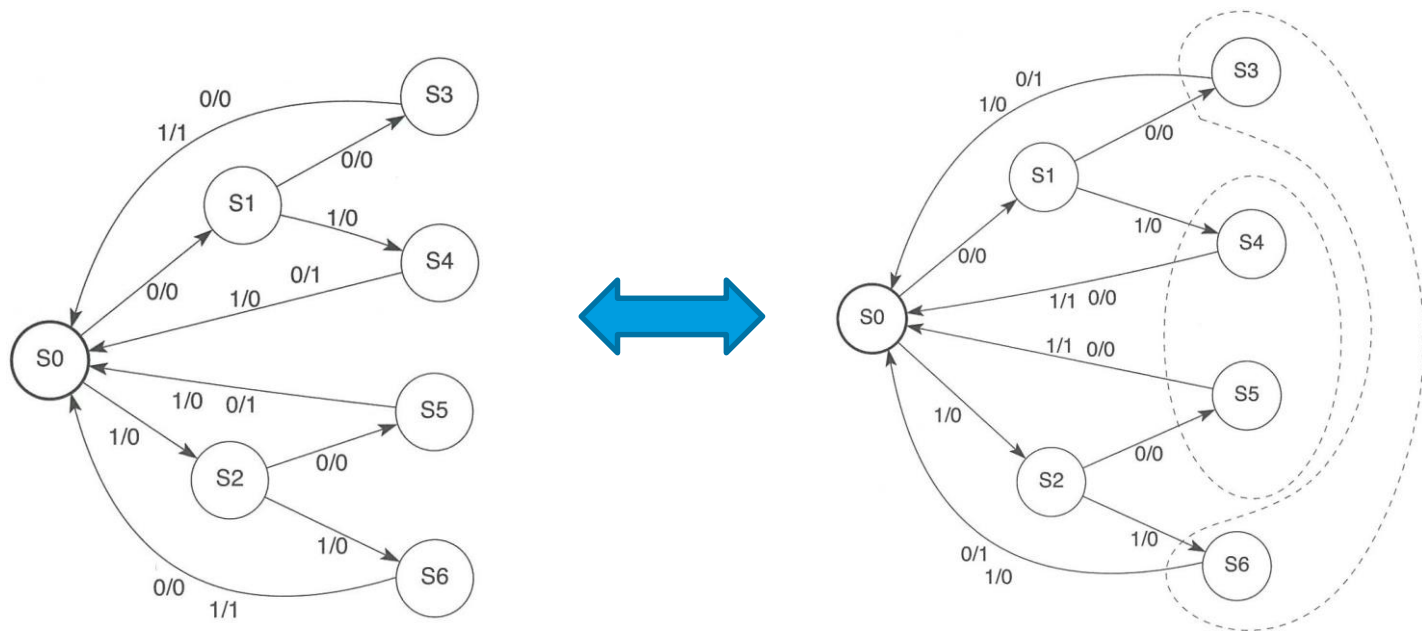


S1	×					
S2	×	×				
S3	×	×	×			
S4	×	×	×	×		
S5	×	×	×	×	✓	
S6	×	×	×	✓	×	×
	S0	S1	S2	S3	S4	S5

■ Simplificação do Diagrama de Estados

► De acordo com o processo de simplificação realizado, conclui-se que:

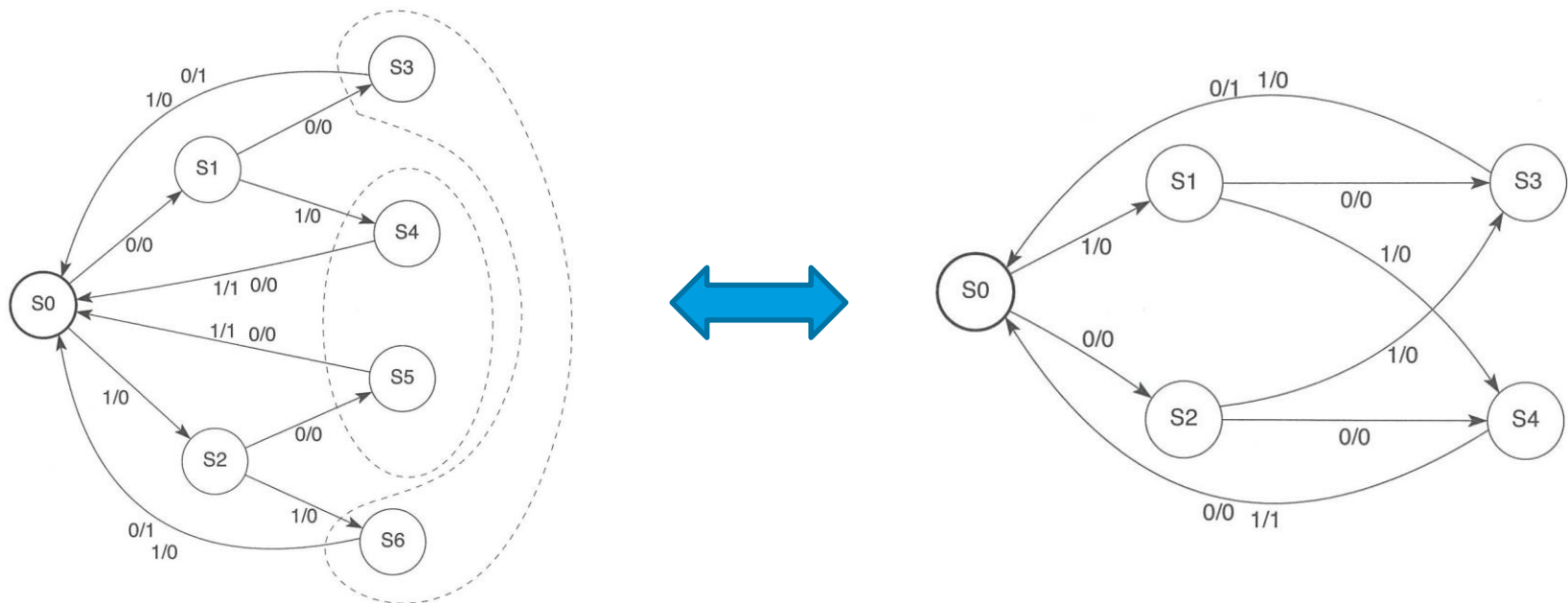
- O estado S3 é equivalente ao estado S6
- O estado S4 é equivalente ao estado S5



■ Simplificação do Diagrama de Estados

► De acordo com o processo de simplificação realizado, conclui-se que:

- O estado S3 é equivalente ao estado S6
- O estado S4 é equivalente ao estado S5



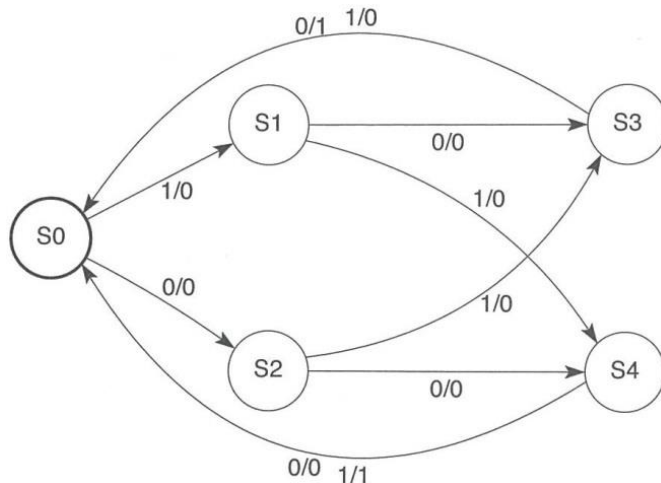
■ Projecto de Circuitos Sequenciais Síncronos

► Procedimento:

- Especificação formal:
 - Diagrama de estados
 - Fluxograma
- Simplificação da especificação
- Projecto:
 - 1. Codificação dos estados**
 2. Tabelas de transição de estados
 3. Determinação das funções lógicas de saída e estado seguinte

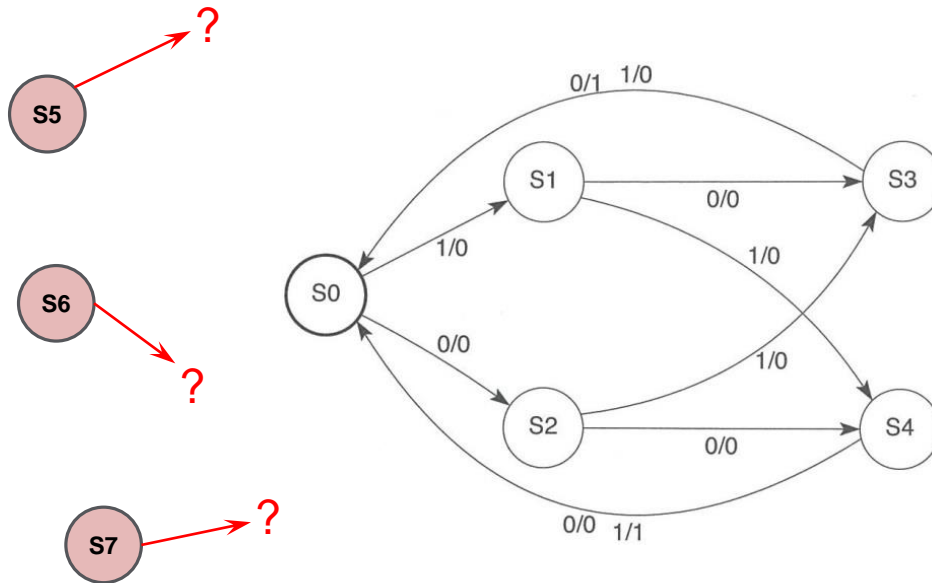
■ Codificação dos Estados

- Considerando a existência de 5 estados (S_0, S_1, S_2, S_3, S_4), a codificação usando código binário natural irá usar k flip-flops, em que $k = \lceil \log_2(5) \rceil = \lceil 2.321 \rceil = 3$



Estado	Codificação		
	Q_2	Q_1	Q_0
S_0	0	0	0
S_1	0	0	1
S_2	0	1	0
S_3	0	1	1
S_4	1	0	0

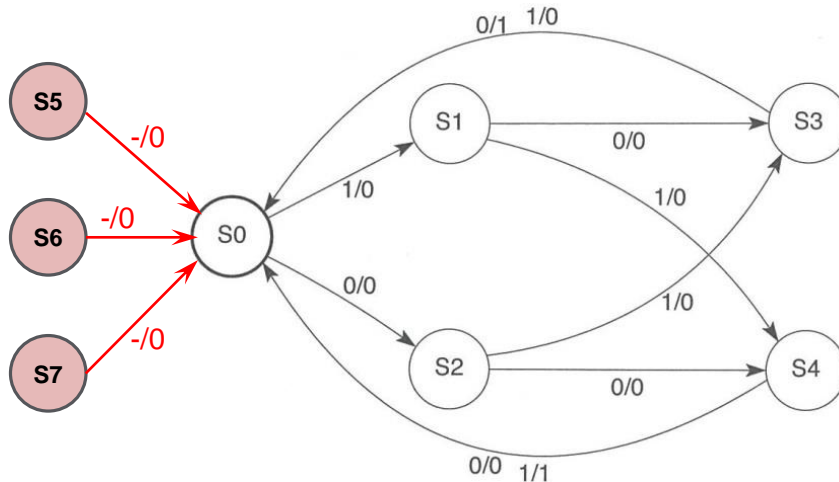
■ Tabela de Transição de Estados



Entradas da Tabela		Saídas da Tabela	
Entrada	Estado Presente	Estado Seguinte	Saída
0	S ₀	S ₂	0
1	S ₀	S ₁	0
0	S ₁	S ₃	0
1	S ₁	S ₄	0
0	S ₂	S ₄	0
1	S ₂	S ₃	0
0	S ₃	S ₀	1
1	S ₃	S ₀	0
0	S ₄	S ₀	0
1	S ₄	S ₀	1

- O que acontece se a máquina transitar para um estado inválido (S₅, S₆, S₇)? → **Lock-out!!!**
- **Solução:** obrigar a máquina a transitar para um estado válido (ex: S₀)

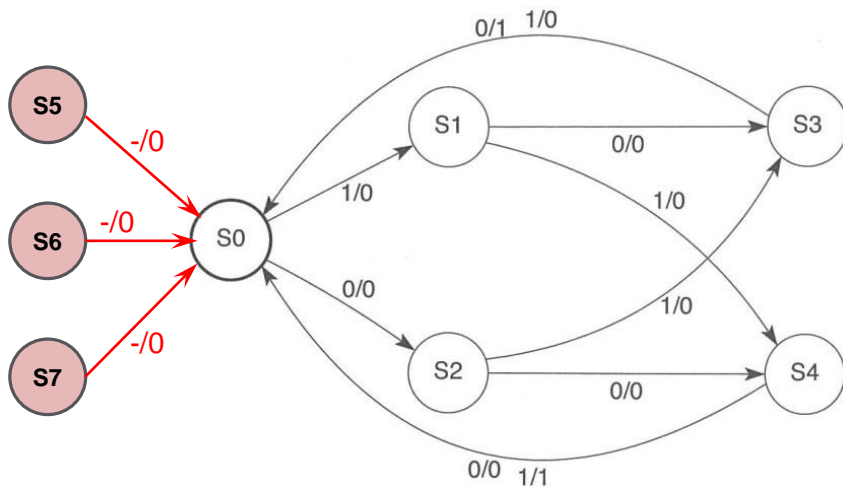
■ Tabela de Transição de Estados



Entradas da Tabela		Saídas da Tabela	
Entrada	Estado Presente	Estado Seguinte	Saída
0	S ₀	S ₂	0
1	S ₀	S ₁	0
0	S ₁	S ₃	0
1	S ₁	S ₄	0
0	S ₂	S ₄	0
1	S ₂	S ₃	0
0	S ₃	S ₀	1
1	S ₃	S ₀	0
0	S ₄	S ₀	0
1	S ₄	S ₀	1
X	S ₅	S ₀	0
X	S ₆	S ₀	0
X	S ₇	S ₀	0

Preenchimento com os estados adicionais, para evitar situações de Lock-out

■ Tabela de Transição de Estados



Entradas da Tabela				Saídas da Tabela			
Entrada	Estado Presente			Estado Seguinte			Saída
	$Q_2(n-1)$	$Q_1(n-1)$	$Q_0(n-1)$	$Q_2(n)$	$Q_1(n)$	$Q_0(n)$	
0	0	0	0	0	1	0	0
1	0	0	0	0	0	1	0
0	0	0	1	0	1	1	0
1	0	0	1	1	0	0	0
0	0	1	0	1	0	0	0
1	0	1	0	0	1	1	0
0	0	1	1	0	0	0	1
1	0	1	1	0	0	0	0
0	1	0	0	0	0	0	0
1	1	0	0	0	0	0	1
X	1	0	1	0	0	0	0
X	1	1	0	0	0	0	0
X	1	1	1	0	0	0	0

■ Determinação das Funções Lógicas

Entradas da Tabela				Saídas da Tabela			
Entrada	Estado Presente			Estado Seguinte			Saída
	$Q_2(n-1) Q_1(n-1) Q_0(n-1)$			$Q_2(n)$	$Q_1(n)$	$Q_0(n)$	
				\downarrow $D_2(n)$	\downarrow $D_1(n)$	\downarrow $D_0(n)$	
0	0	0	0	0	1	0	0
1	0	0	0	0	0	1	0
0	0	0	1	0	1	1	0
1	0	0	1	1	0	0	0
0	0	1	0	1	0	0	0
1	0	1	0	0	1	1	0
0	0	1	1	0	0	0	1
1	0	1	1	0	0	0	0
0	1	0	0	0	0	0	0
1	1	0	0	0	0	0	1
X	1	0	1	0	0	0	0
X	1	1	0	0	0	0	0
X	1	1	1	0	0	0	0

Q_1Q_0	00	01	11	10
$E Q_2$				
00	0	0	0	1
01	0	0	0	0
11	0	0	0	0
10	0	1	0	0

$$D_2 = E \overline{Q_2} \overline{Q_1} \overline{Q_0} + E \overline{Q_2} Q_1 \overline{Q_0}$$

Q_1Q_0	00	01	11	10
$E Q_2$				
00	1	1	0	0
01	0	0	0	0
11	0	0	0	0
10	0	0	0	1

$$D_1 = E \overline{Q_2} \overline{Q_1} + E \overline{Q_2} Q_1 \overline{Q_0}$$

Q_1Q_0	00	01	11	10
$E Q_2$				
00	0	1	0	0
01	0	0	0	0
11	0	0	0	0
10	1	0	0	1

$$D_0 = E \overline{Q_2} \overline{Q_0} + E \overline{Q_2} Q_1 \overline{Q_0}$$

Q_1Q_0	00	01	11	10
$E Q_2$				
00	0	0	1	0
01	0	0	0	0
11	1	0	0	0
10	0	0	0	0

$$S = E \overline{Q_2} \overline{Q_1} \overline{Q_0} + E \overline{Q_2} Q_1 \overline{Q_0}$$

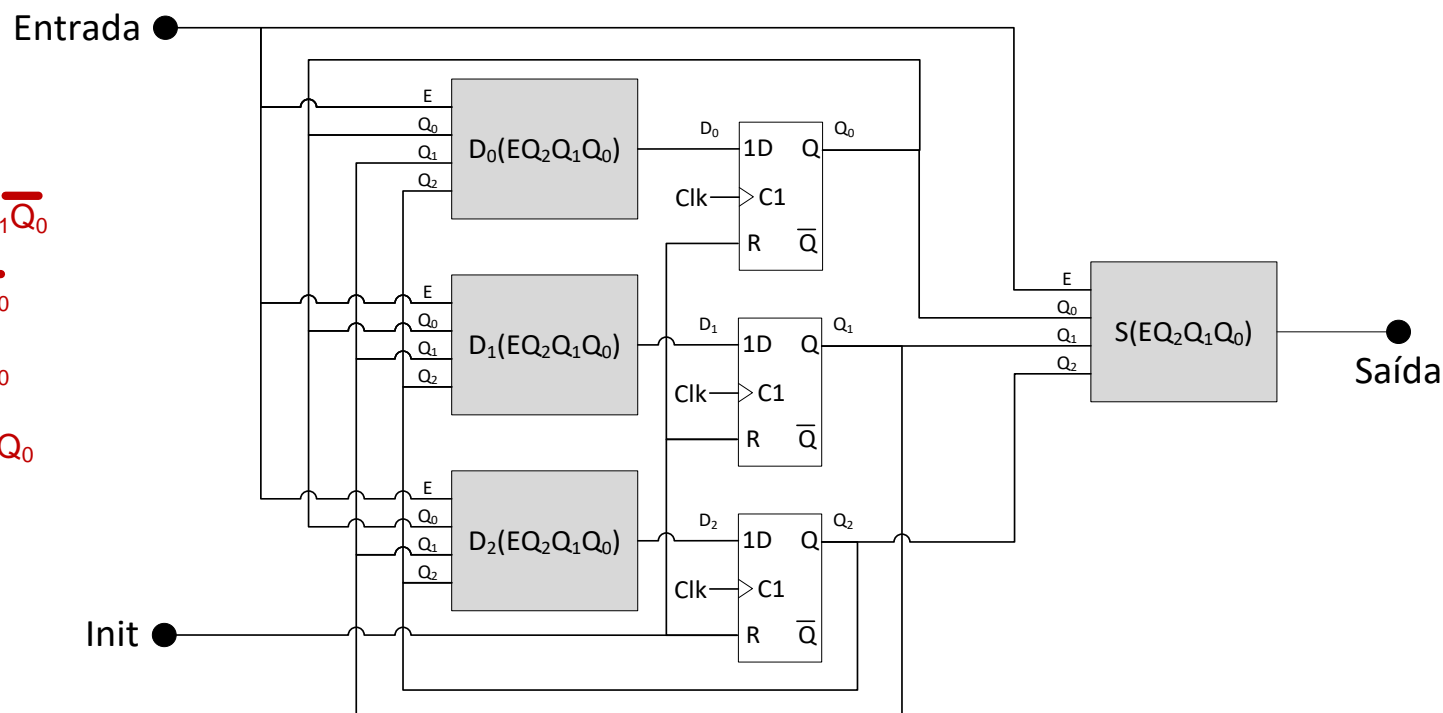
■ Circuito Lógico

$$D_2 = E \overline{Q_2} \overline{Q_1} \overline{Q_0} + E \overline{Q_2} \overline{Q_1} Q_0$$

$$D_1 = E \overline{Q_2} \overline{Q_1} Q_0 + E \overline{Q_2} Q_1 \overline{Q_0}$$

$$D_0 = E \overline{Q_2} \overline{Q_1} Q_0 + E \overline{Q_2} Q_1 Q_0$$

$$S = E \overline{Q_2} \overline{Q_1} \overline{Q_0} + E \overline{Q_2} \overline{Q_1} Q_0$$





■ Tema da Próxima Aula:

- ▶ Exemplo (Moore)
- ▶ Projecto de circuitos sequenciais baseados em contadores

Agradecimentos

Algumas páginas desta apresentação resultam da compilação de várias contribuições produzidas por:

- Nuno Roma
- Guilherme Arroz
- Horácio Neto
- Nuno Horta
- Pedro Tomás