

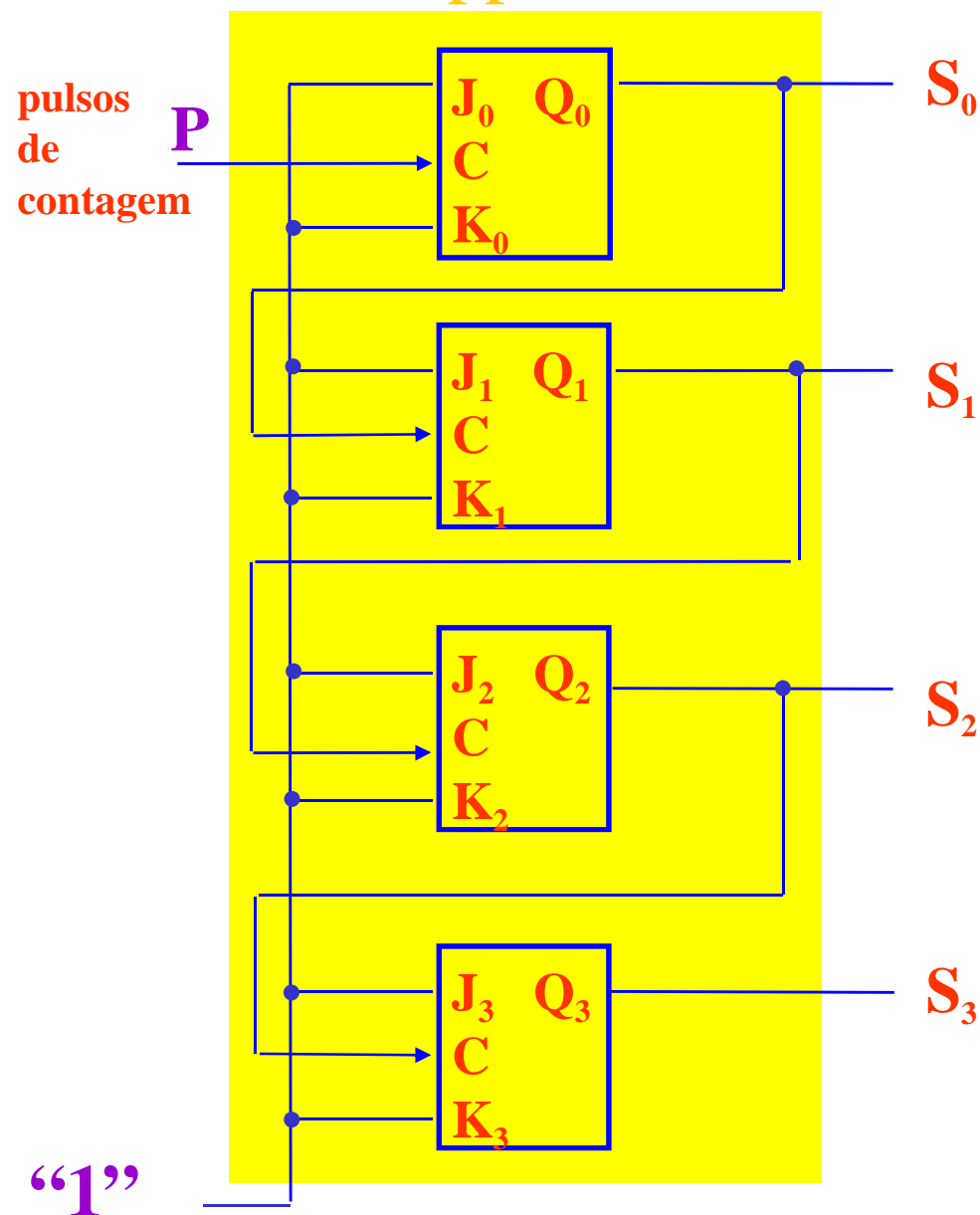
INF01 118

# Técnicas Digitais para Computação

## Contadores



## 2. Contador ripple binário



Supondo FF's sensíveis à transição negativa do sinal do controle

Também possível com FF's tipo T

JK Mestre-escravo ou sensível à borda do relógio

- como  $J = K = 1$  em todos os FF's

- cada transição negativa da entrada C causa complemento do FF

- portanto

cada transição negativa de P  $\Rightarrow$  S0 complementado

“ “ “ “ S0  $\Rightarrow$  S1 “

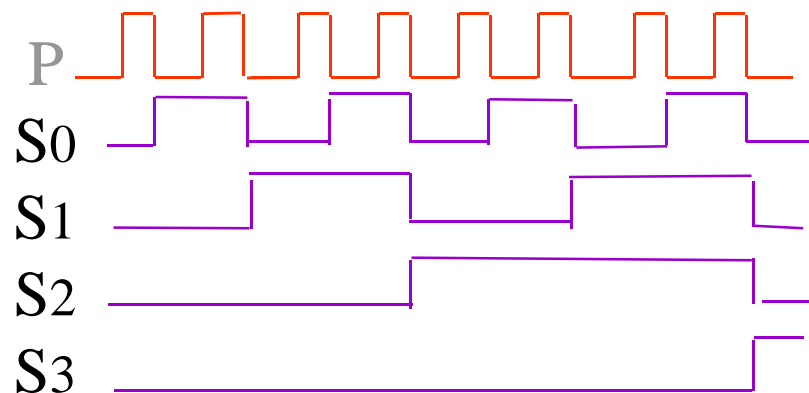
“ “ “ “ S1  $\Rightarrow$  S2 “

“ “ “ “ S2  $\Rightarrow$  S3 “

- sequência de estados

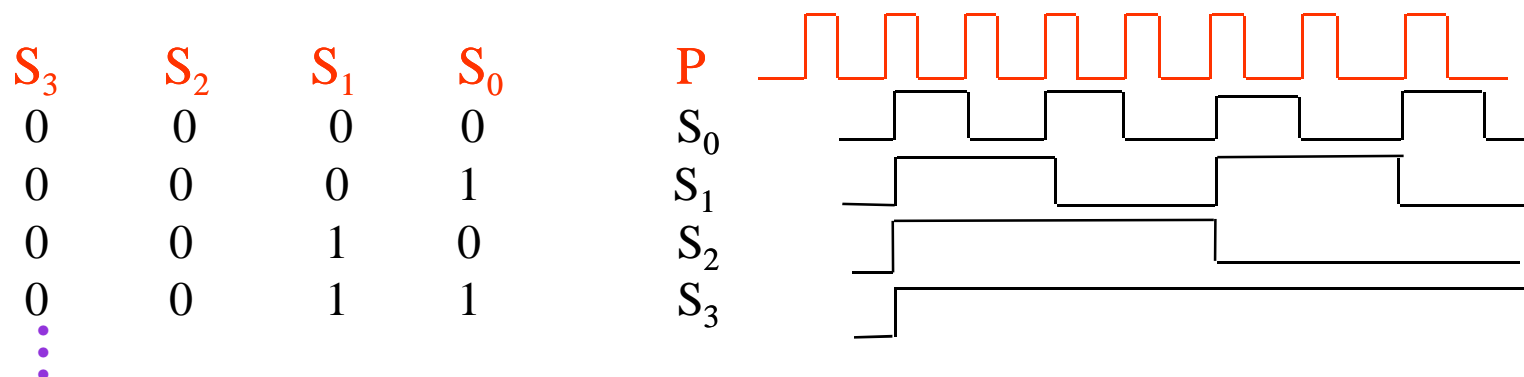
S3	S2	S1	S0
0	0	0	0
0	0	0	1
0	0	1	0
0	0	1	1

⋮

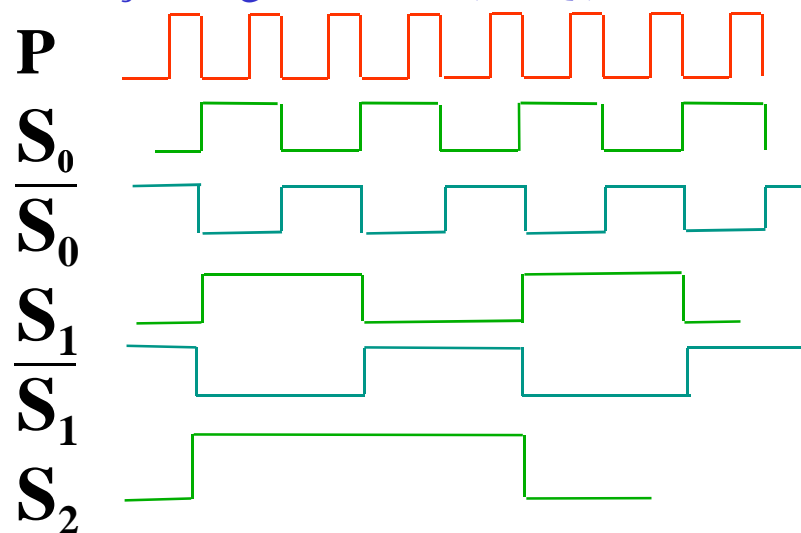


## • para contar para baixo

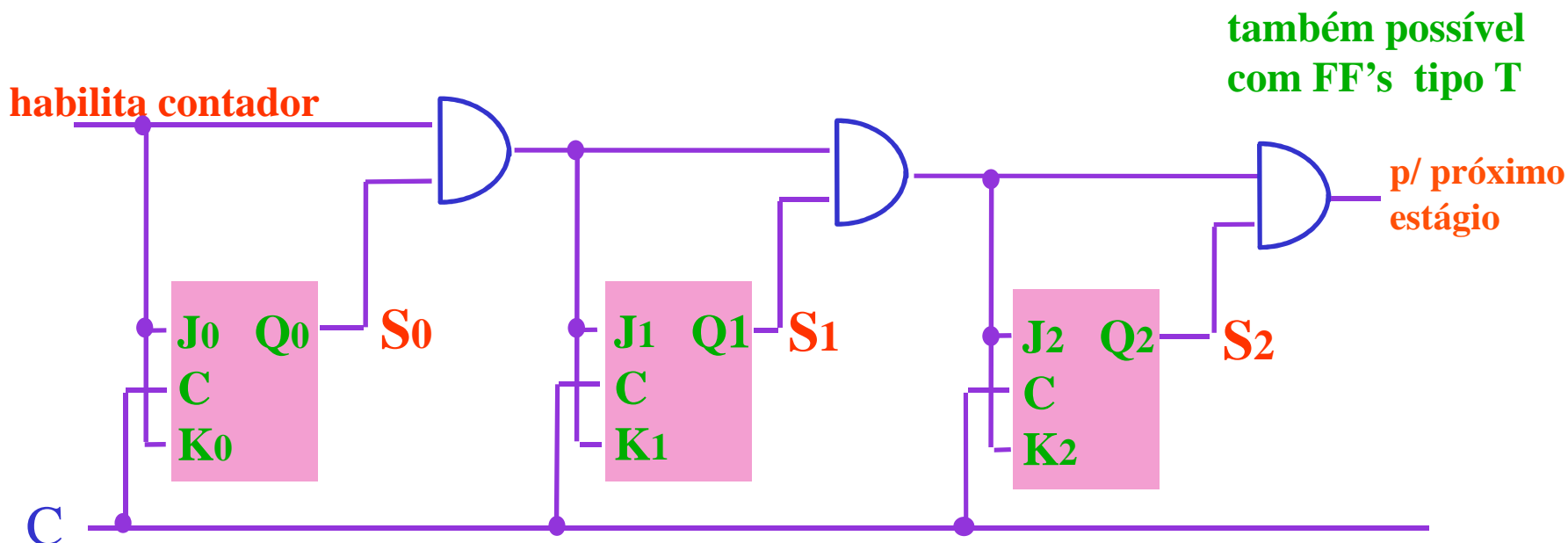
- **alternativa 1** - pegar saídas complementadas dos FF's como saídas do contador
- **alternativa 2** - usar FF's sensíveis à transição de descida do sinal de controle(CLK)



- **alternativa 3** - ligar saída  $\overline{Q}$  de cada FF à entrada C do FF seguinte  
transição negativa de P (ou  $\overline{Q_i}$ ) causa complemento de  $Q_{i+1}$



### 3. Contador síncrono binário



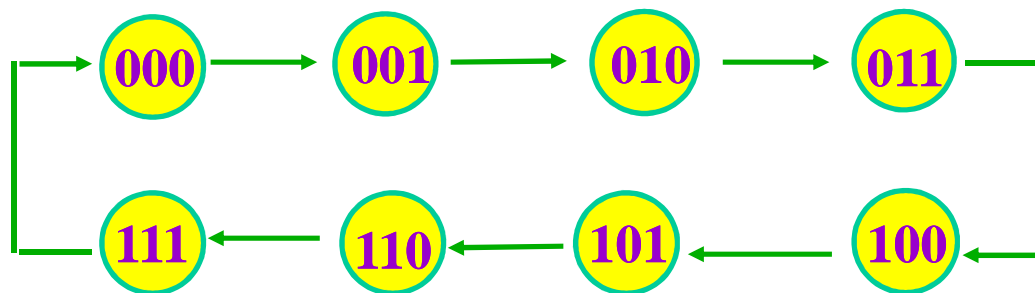
- Como  $J_0 = K_0 = 1$  no primeiro FF, cada transição do clock (positiva ou negativa, à escolha) causa complemento de  $S_0$
- Quando  $S_0 = 1 \Rightarrow J_1 = K_1 = 1$ , próxima transição do clock causa complemento de  $S_1$
- Quando  $S_0 = 1$  e  $S_1 = 1 \Rightarrow J_2 = K_2 = 1$ , próxima transição do clock complementa  $S_2$

seqüência  
de valores

$S_2$	$S_1$	$S_0$
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
⋮		

## Projeto de um contador síncrono binário

### A) FSM



### B) Tabela de Estados

Est. Atual			Próx. Est.			Eq. Entrada		
S2	S1	S0	S2	S1	S0	T2	T1	T0
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	1	1	0	0	1	1
1	1	0	1	1	1	0	0	1
1	1	1	0	0	0	1	1	1

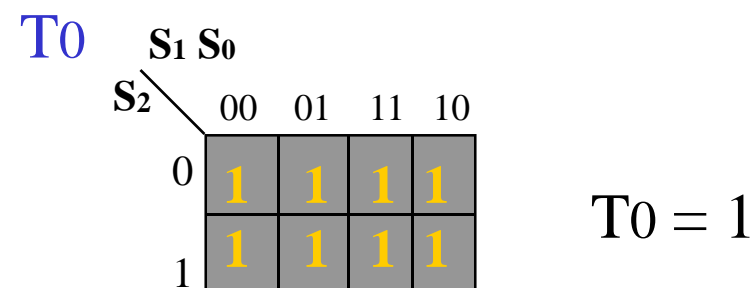
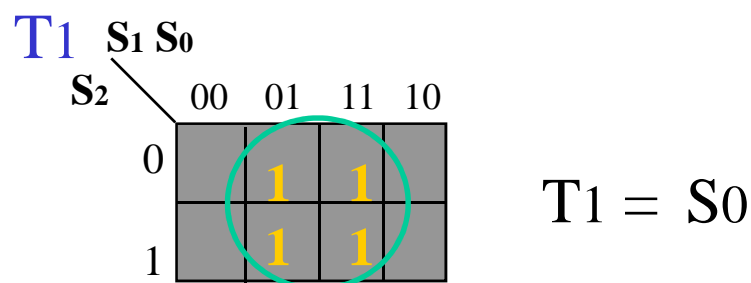
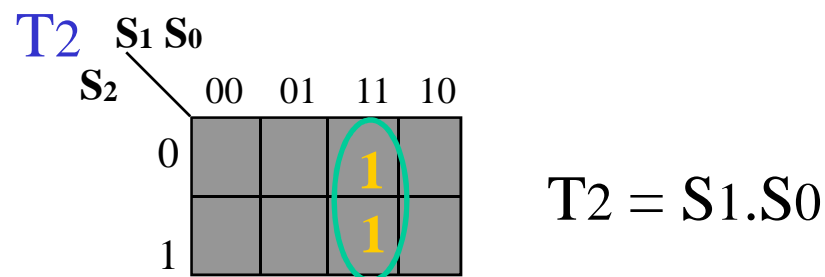
## C) Projeto com FF's tipo T

### C.1 Equações de entrada dos FF's

**1** indica “tem que complementar”

**0** indica “não precisa complementar”

### C.2 Mapas de Karnaugh



O resultado é o circuito já mostrado



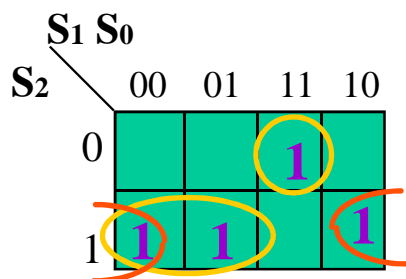
## D) Projeto com FF's tipo D

### D.1 Equação de entrada dos FF's

Iguais aos valores de próximo estado (  $Q = D$  no FF tipo D)

### D.2 Mapas de Karnaugh

**D2**

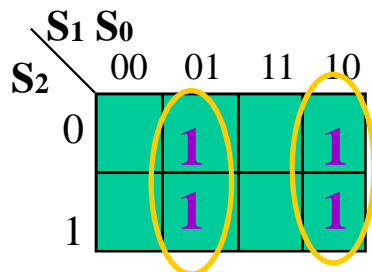


$$D2 = S_2 \overline{S_1} + S_2 \overline{S_0} + \overline{S_2} S_1 S_0$$

ou seja, D2 = 1 quando

S2	S1	S0
1	0	X
1	X	0
0	1	1

**D1**

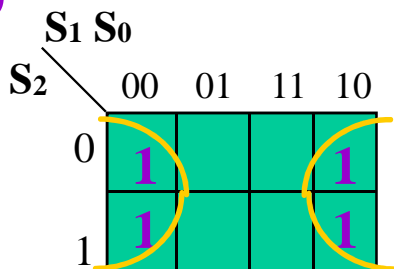


$$D1 = \overline{S_1} S_0 + S_1 \overline{S_0}$$

ou seja, D1 = 1 quando

S2	S1	S0
X	0	1
X	1	0

**D0**



$$D0 = \overline{S_0}$$

ou seja, D0 = 1 quando

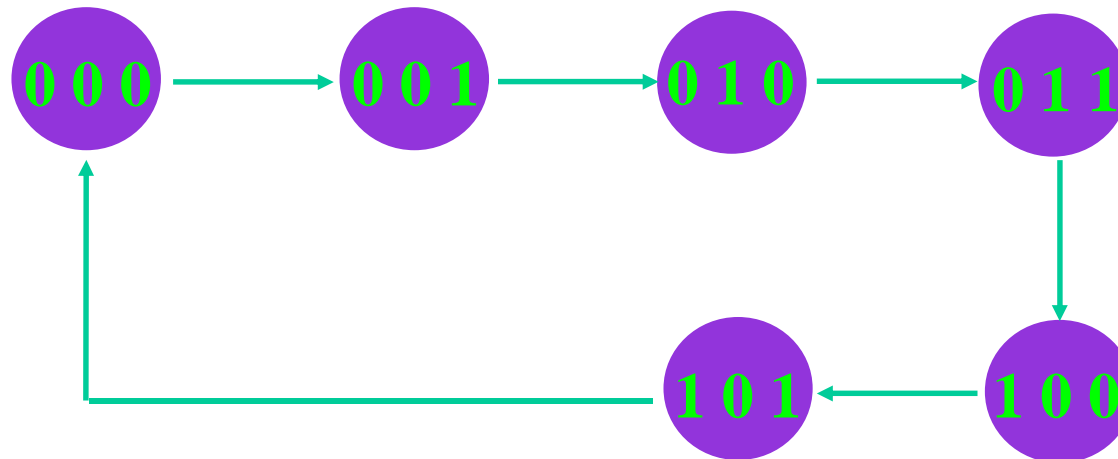
S2	S1	S0
X	X	0

equações mais complexas com o uso de FF's tipo D

## 4. Contadores módulo N

Para  $m$  flip-flops, supor um circuito que conte até  $N < 2^m - 1$ , ou seja, que não use todos os  $2^m$  estados possíveis

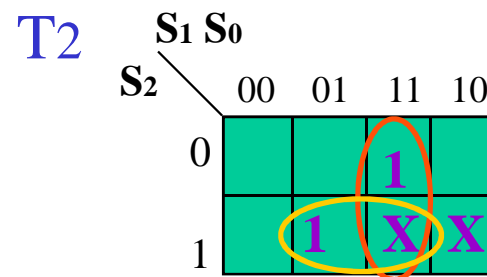
*exemplo:* contador até 5



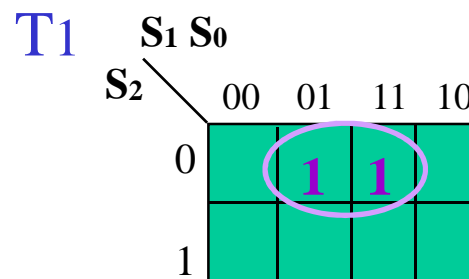
## Tabela de Estados e Equações de entrada para FF's tipo T

Estado Atual			Próximo Estado			Equação Entrada T		
S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>	S <sub>2</sub>	S <sub>1</sub>	S <sub>0</sub>	T <sub>2</sub>	T <sub>1</sub>	T <sub>0</sub>
0	0	0	0	0	1	0	0	1
0	0	1	0	1	0	0	1	1
0	1	0	0	1	1	0	0	1
0	1	1	1	0	0	1	1	1
1	0	0	1	0	1	0	0	1
1	0	1	0	0	0	1	0	1
1	1	0	X	X	X	X	X	X
1	1	1	X	X	X	X	X	X

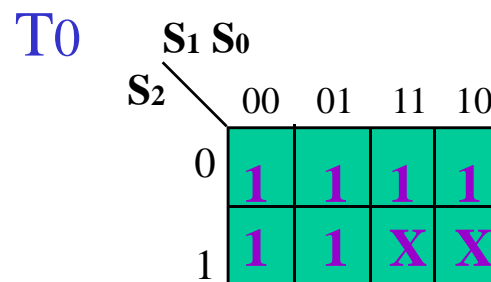
## Mapa de Karnaugh



$$T_2 = S_1 S_0 + S_2 S_0 = S_0 (S_1 + S_2)$$

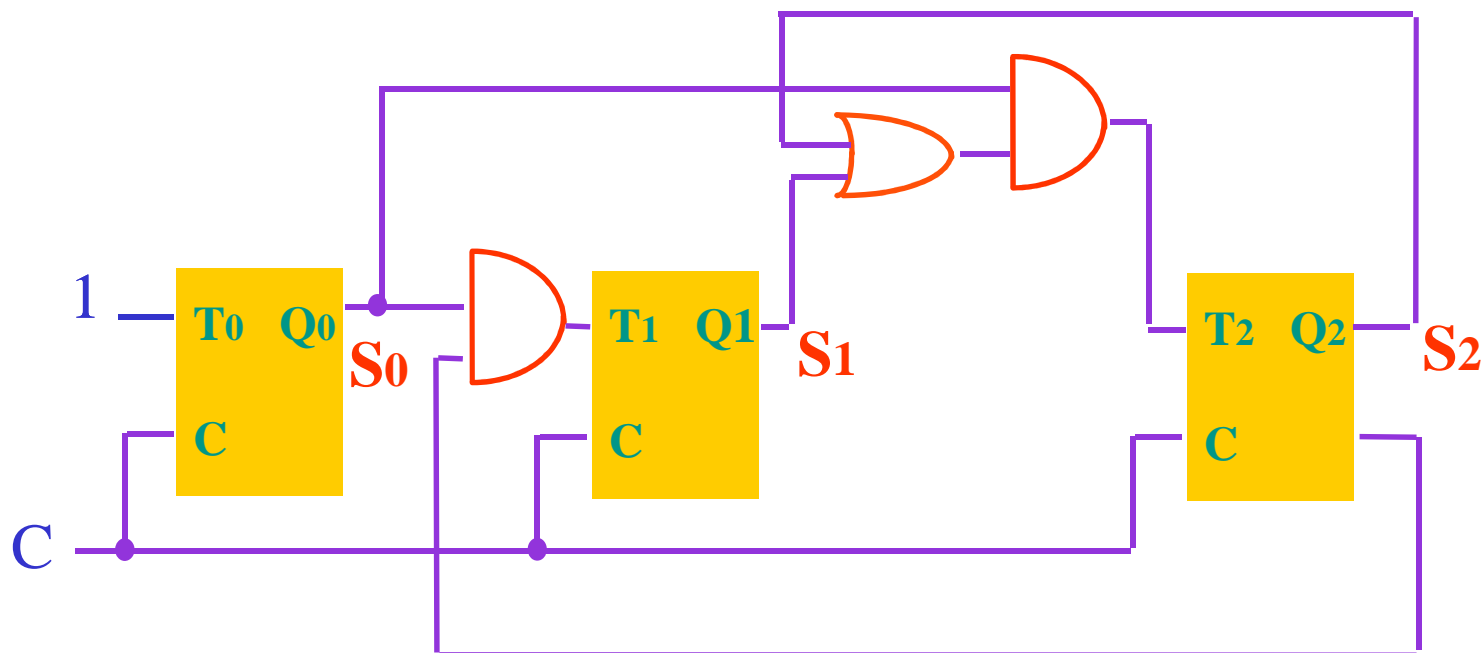


$$T_1 = S_0 \bar{S}_2$$

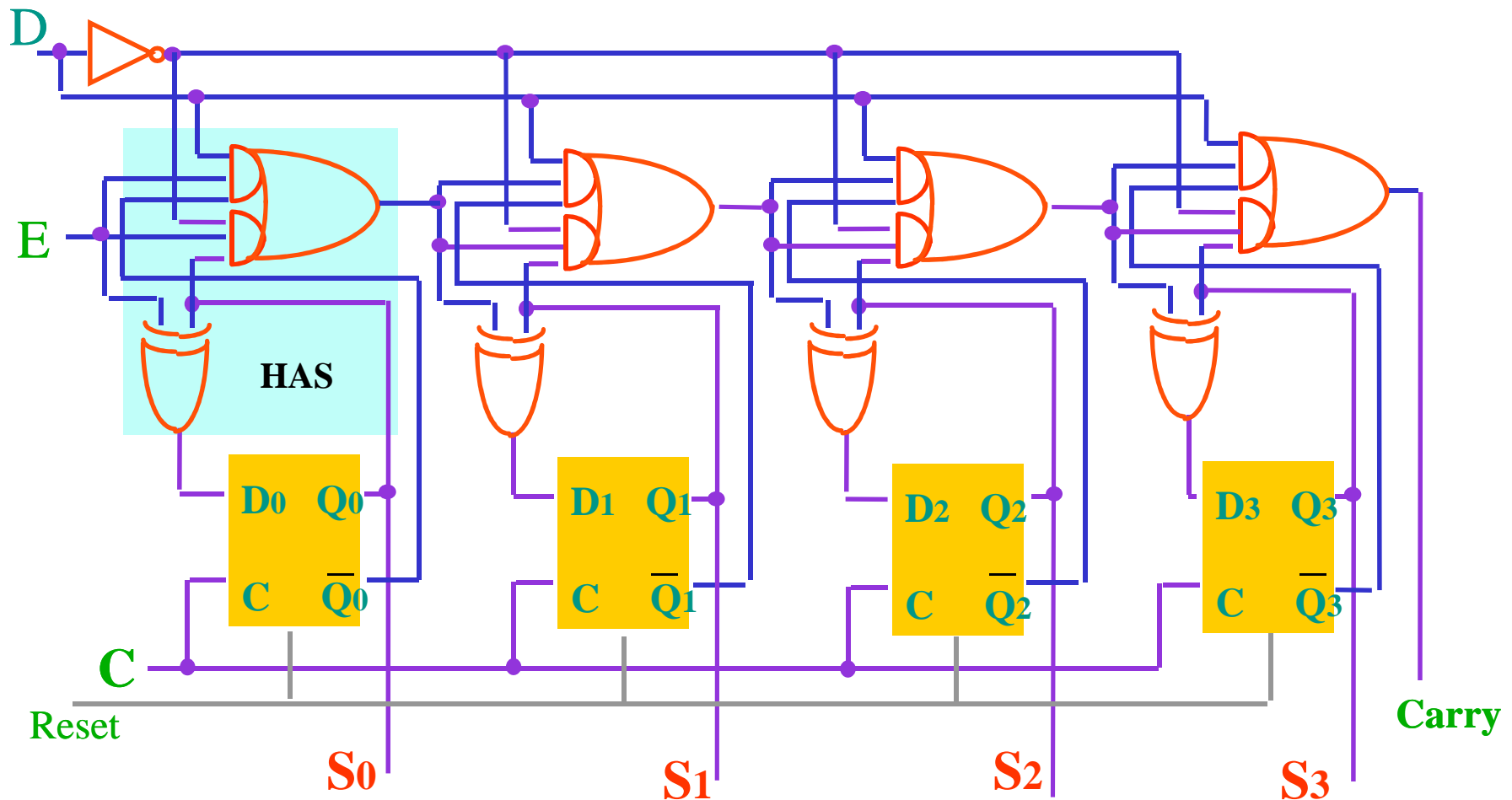
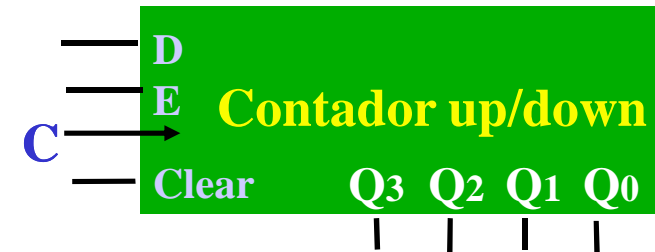


$$T_0 = 1$$

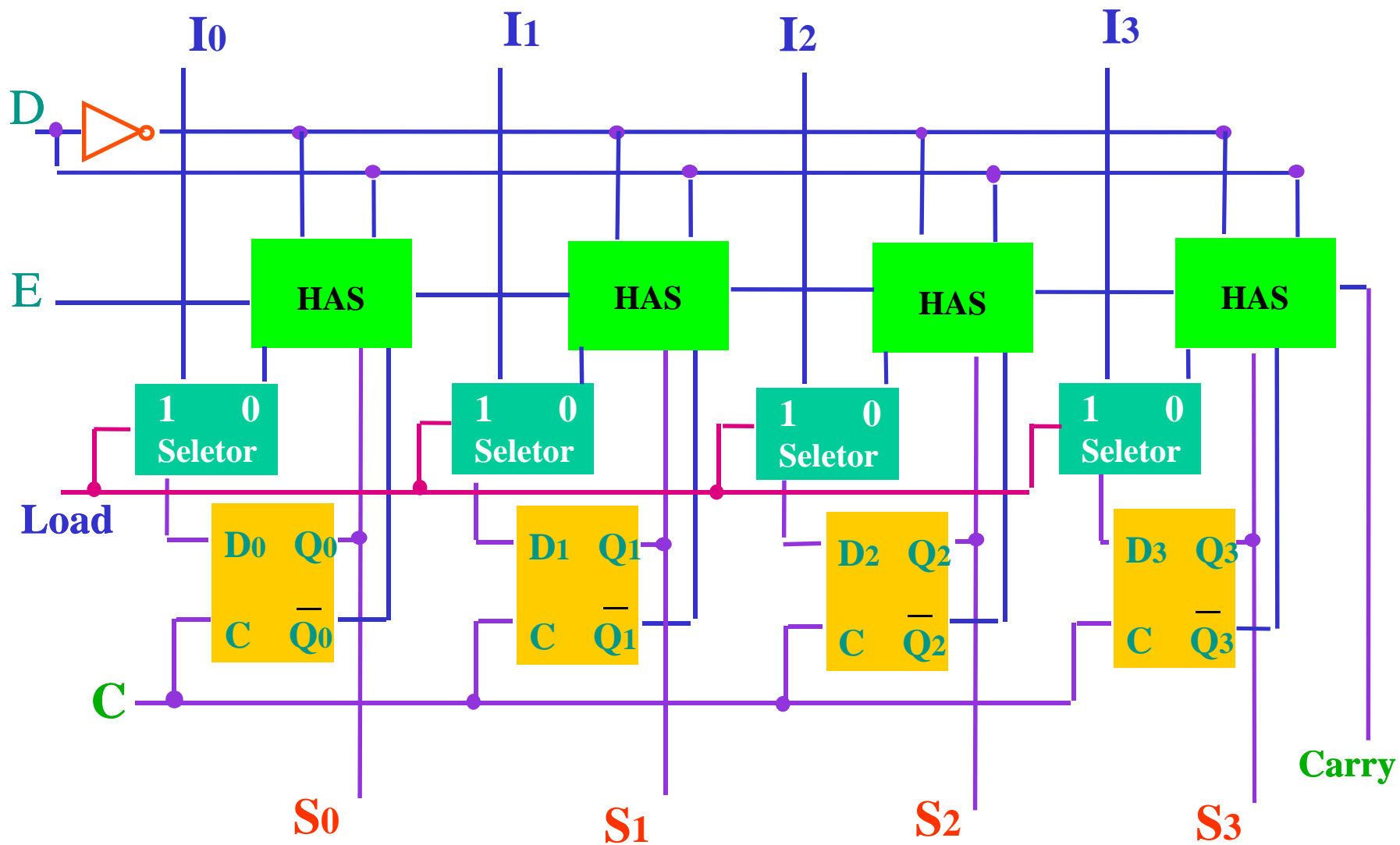
## Implementação do contador de módulo 5 (contador até 5)



## Contador binário up/down



## Contador up/down com carga paralela

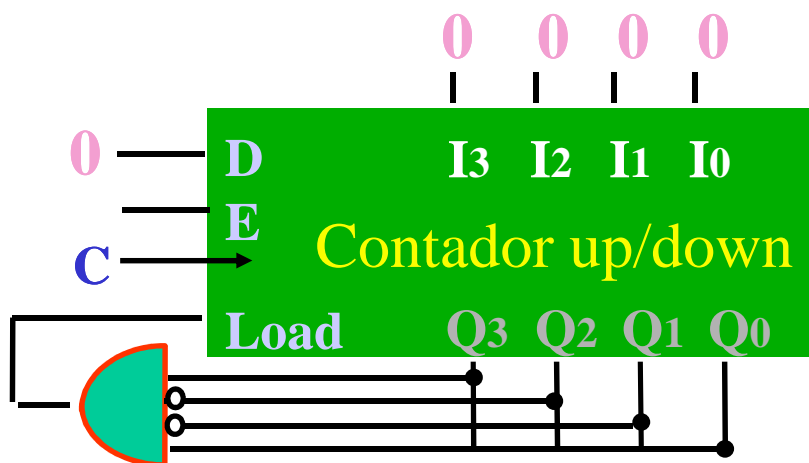


## 5. Contador BCD

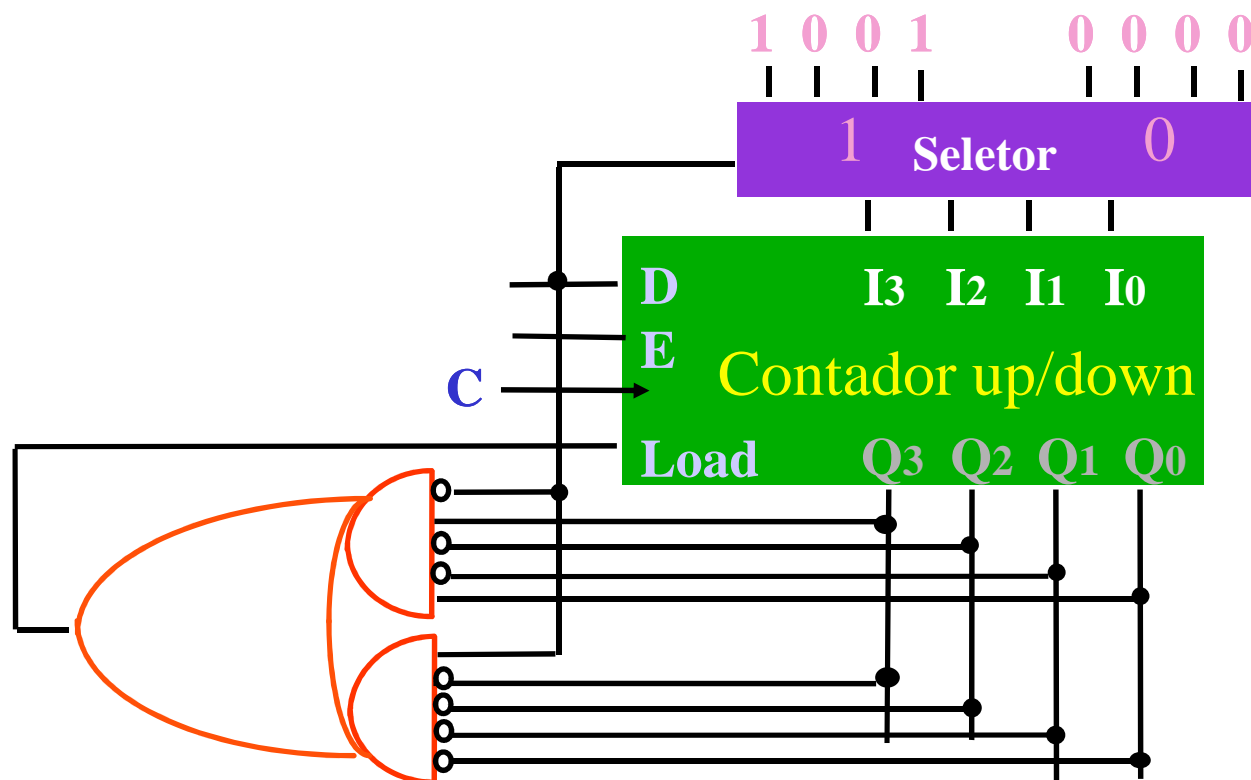
Conta de 0 a 9 e então reinicia contagem.

Lógica combinacional detecta quando a contagem chega a 9.

Contador BCD para cima



## Contador BCD up/down

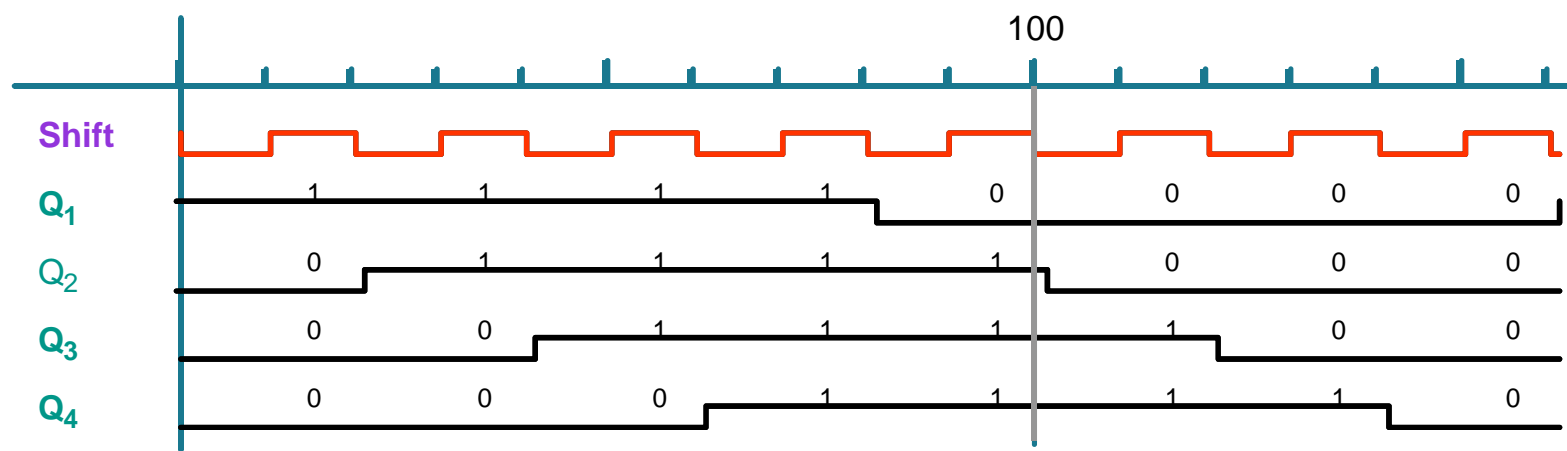
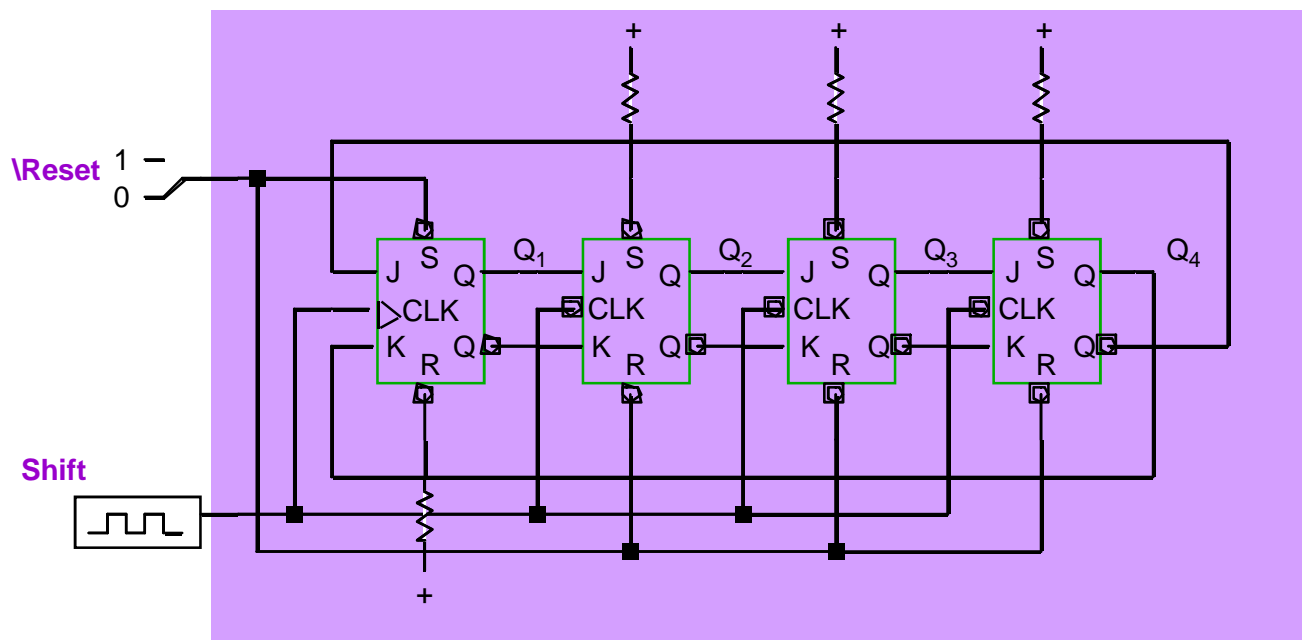


$D = 0$  ➡ contagem para cima (UP)

$D = 1$  ➡ contagem para baixo (DOWN)



## 6. Contador Johnson (Mobius)



8 estados possíveis, muda só um bit em cada transição de estado, útil para evitar hazards