INF01058

Circuitos Digitais

Contadores

Aula 19c

1. Introdução

- **contador** = registrador que passa por seqüência de estados quando são aplicados pulsos de entrada

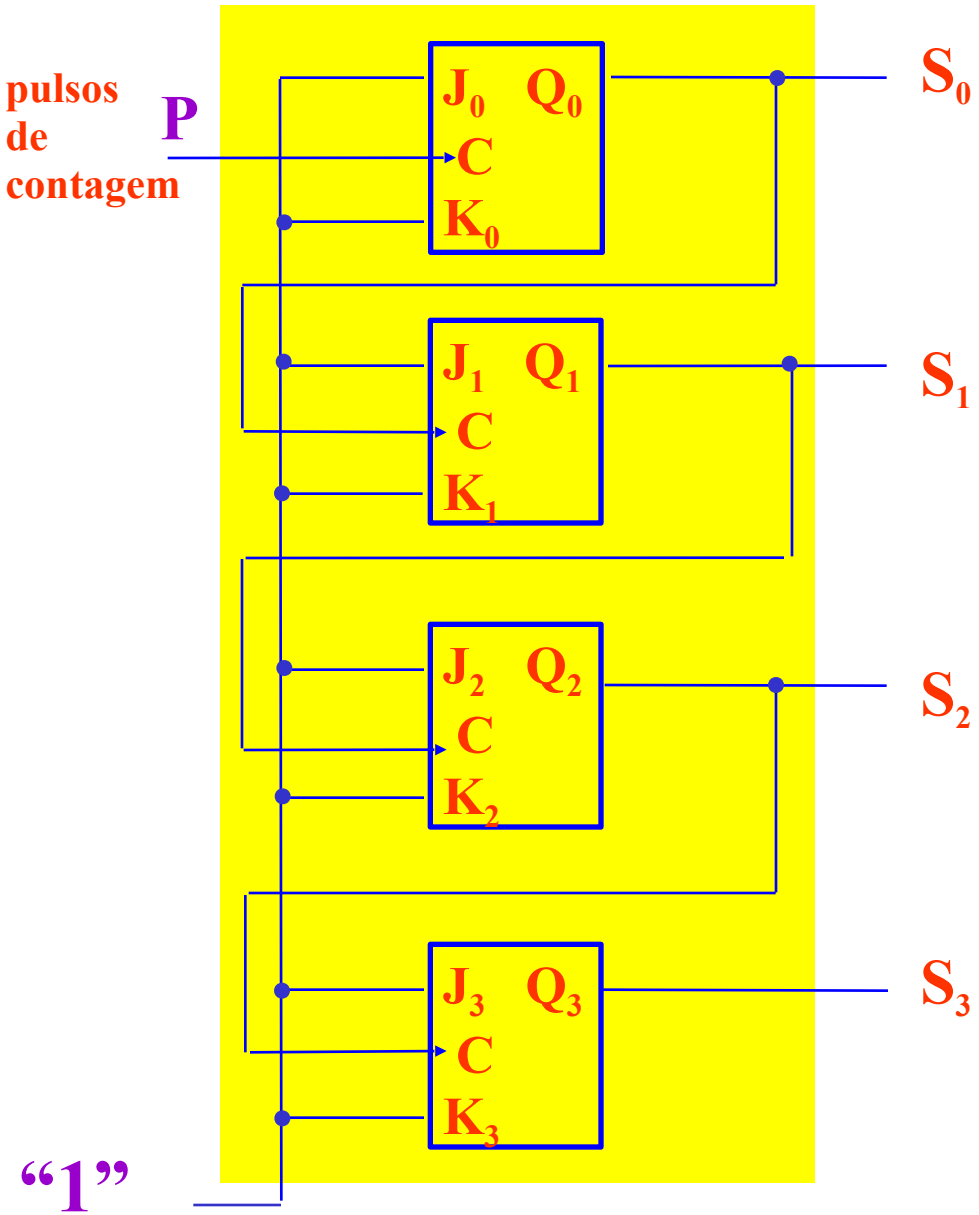
  É uma FSM em que [**as únicas saídas são os estados**
  a única entrada é o pulso de contagem

- **seqüência de estados**
  - Seqüência de números binários - contador binário
  - Qualquer outra seqüência - contador BCD
    outros códigos
    seqüências arbitrárias (1,2,3,5,7...)

- necessidade de lógica combinacional para controlar seqüência de contagem

- **tipos de contadores**
  - "Ripple Counters " - transição de um FF serve para disparar transição do próximo (assíncronos)
  - "Contadores Síncronos" - todos os FF's são carregados simultaneamente pelo clock

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 → S0 complementado

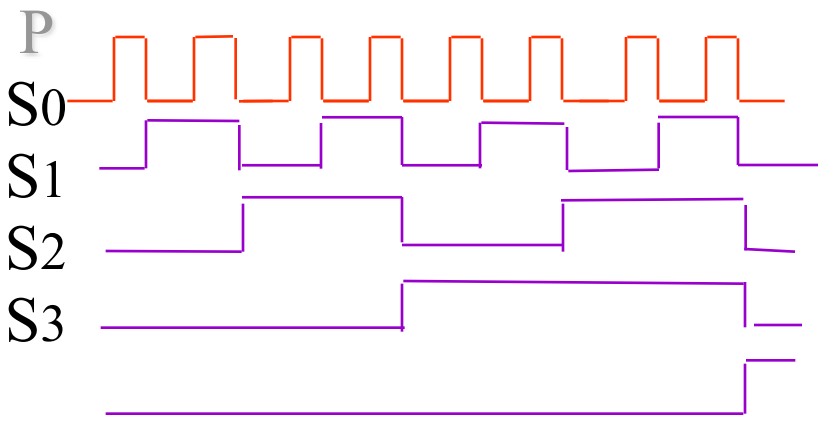
    " " " " S0 → S1 "

    " " " " S1 → S2 "

    " " " " S2 → S3 "

- **sequência de estados**

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

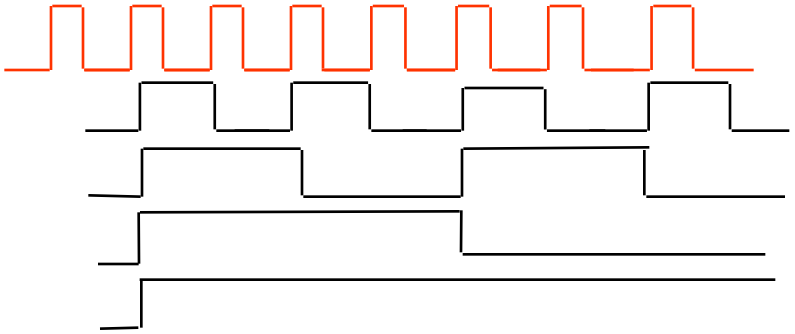


⋮

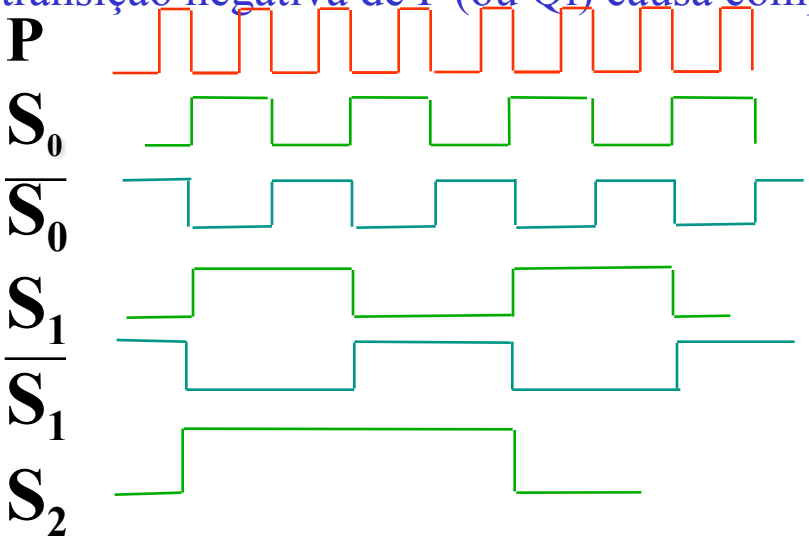
- **para contar para baixo**

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

S_3	S_2	S_1	S_0	P
0	0	0	0	S_0
0	0	0	1	S_1
0	0	1	0	S_2
0	0	1	1	S_3
⋮				

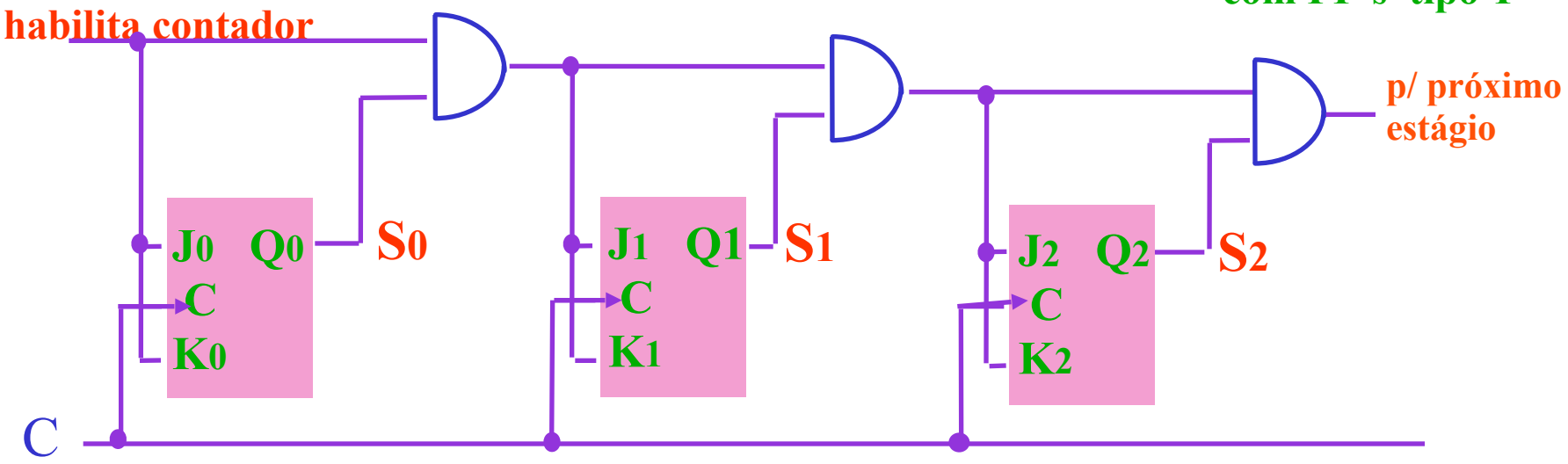


- alternativa 3 - ligar saída Q de cada FF à entrada C do FF seguinte
transição negativa de P (ou Qi) causa complemento de Qi + 1



3. Contador síncrono binário

também possível com FF's tipo T



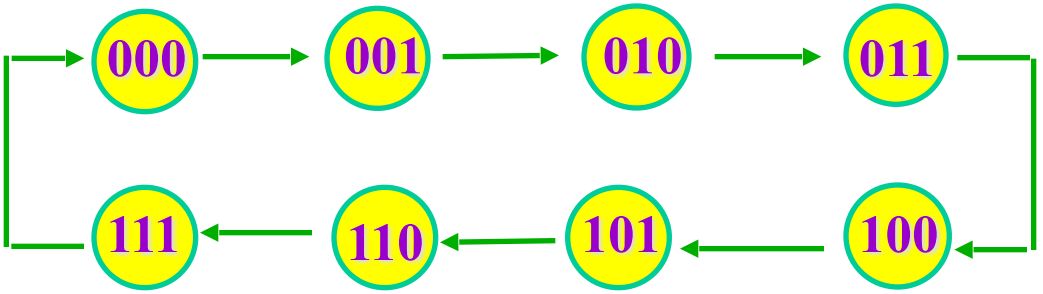
- Como J0 = K0 = 1 no primeiro FF, cada transição do clock (positiva ou negativa, à escolha) causa complemento de S0
- Quando S0 = 1 ⇒ J1 = K1 = 1, próxima transição do clock causa complemento de S1
- Quando S0 = 1 e S1 = 1 ⇒ J2 = K2 = 2 , próxima transição do clock complementa S2

seqüência de valores

S2	S1	S0
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

T2

S_2 \ $S_1 S_0$	00	01	11	10
0			1	
1			1	

$T2 = S1.S0$

T1

S_2 \ $S_1 S_0$	00	01	11	10
0		1	1	
1		1	1	

$T1 = S0$

T0

S_2 \ $S_1 S_0$	00	01	11	10
0	1	1	1	1
1	1	1	1	1

$T0 = 1$

O resultado é o circuito já mostrado

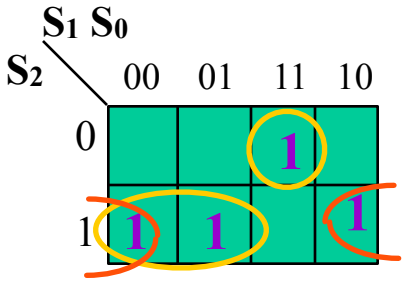
D) Projeto com FF's tipo D

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

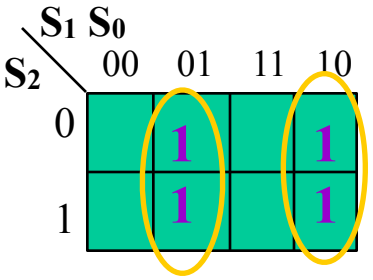


$D2 = S_2\overline{S_1} + S_2\overline{S_0} + \overline{S_2}S_1S_0$

ou seja, D2 =1 quando

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

D1

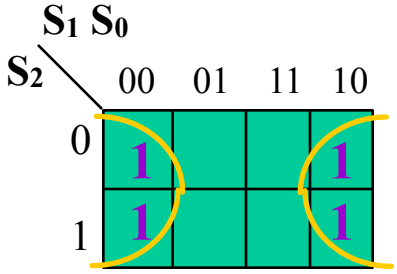


$D1 = \overline{S_1}S_0 + S_1\overline{S_0}$

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

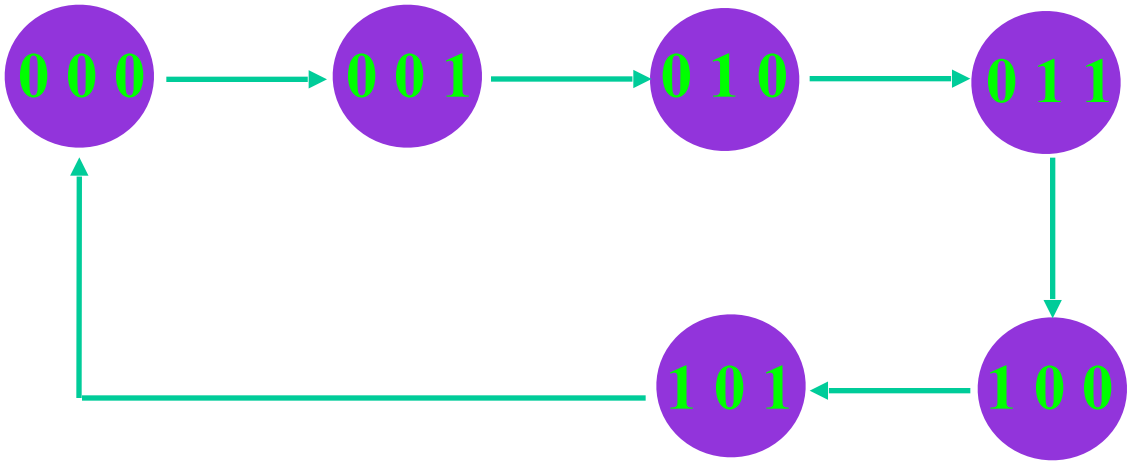
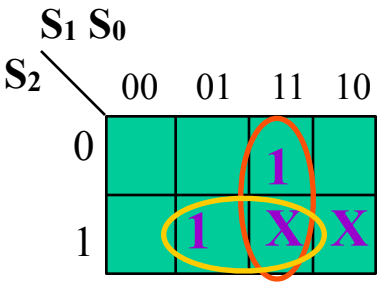


Tabela de Estados e Equações de entrada para FF's tipo T

Estado Atual			Próximo Estado			Equação Entrada T		
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	0	0	0	1	0	1
1	1	0	X	X	X	X	X	X
1	1	1	X	X	X	X	X	X

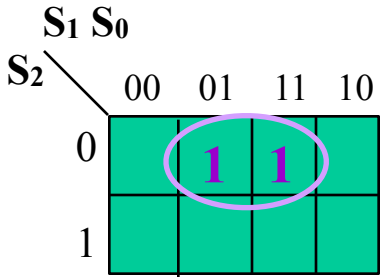
Mapa de Karnaugh

T2



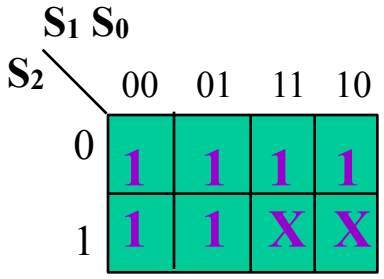
$$T2 = S_1S_0 + S_2S_0 = S_0(S_1 + S_2)$$

T1



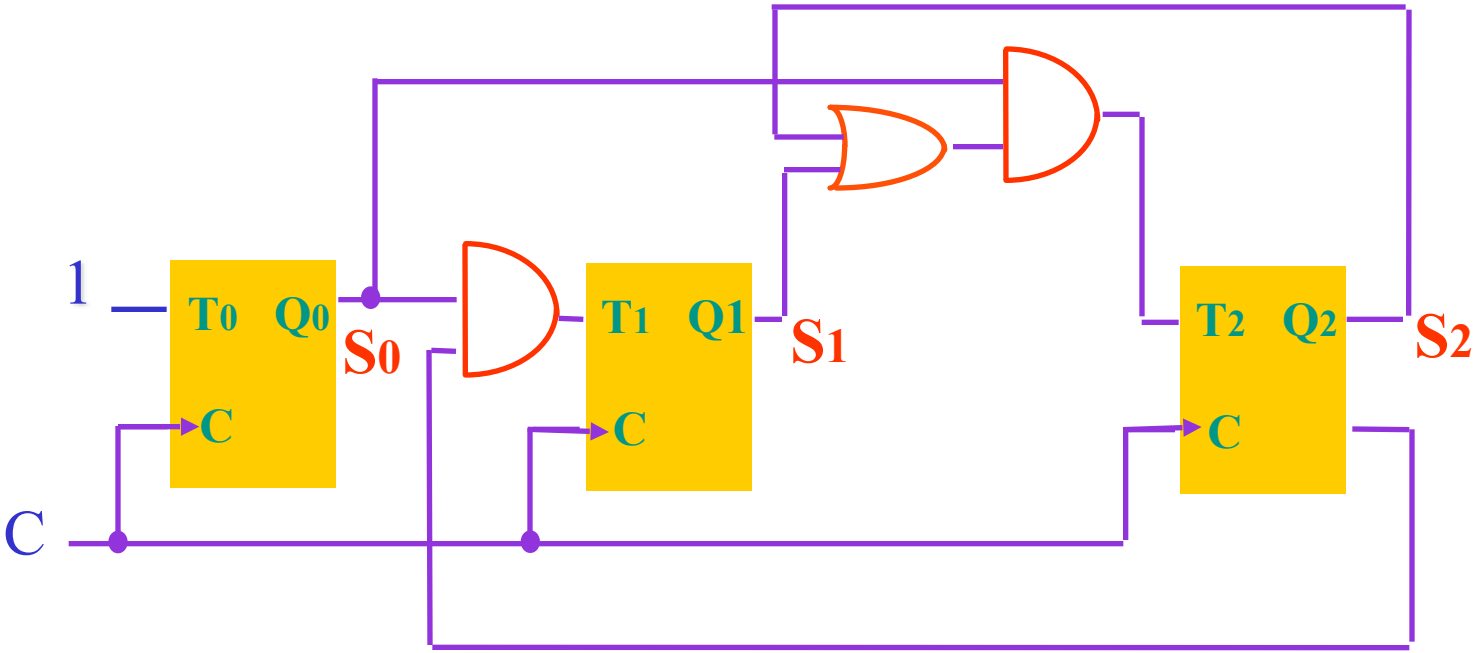
$$T1 = S_0\overline{S}_2$$

T0

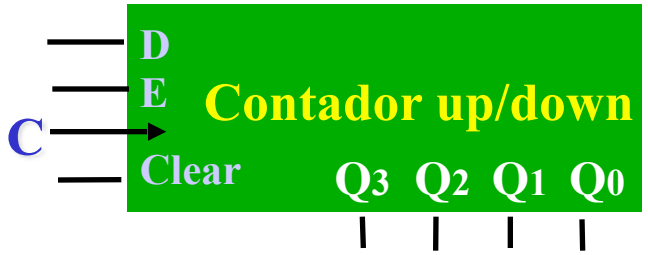


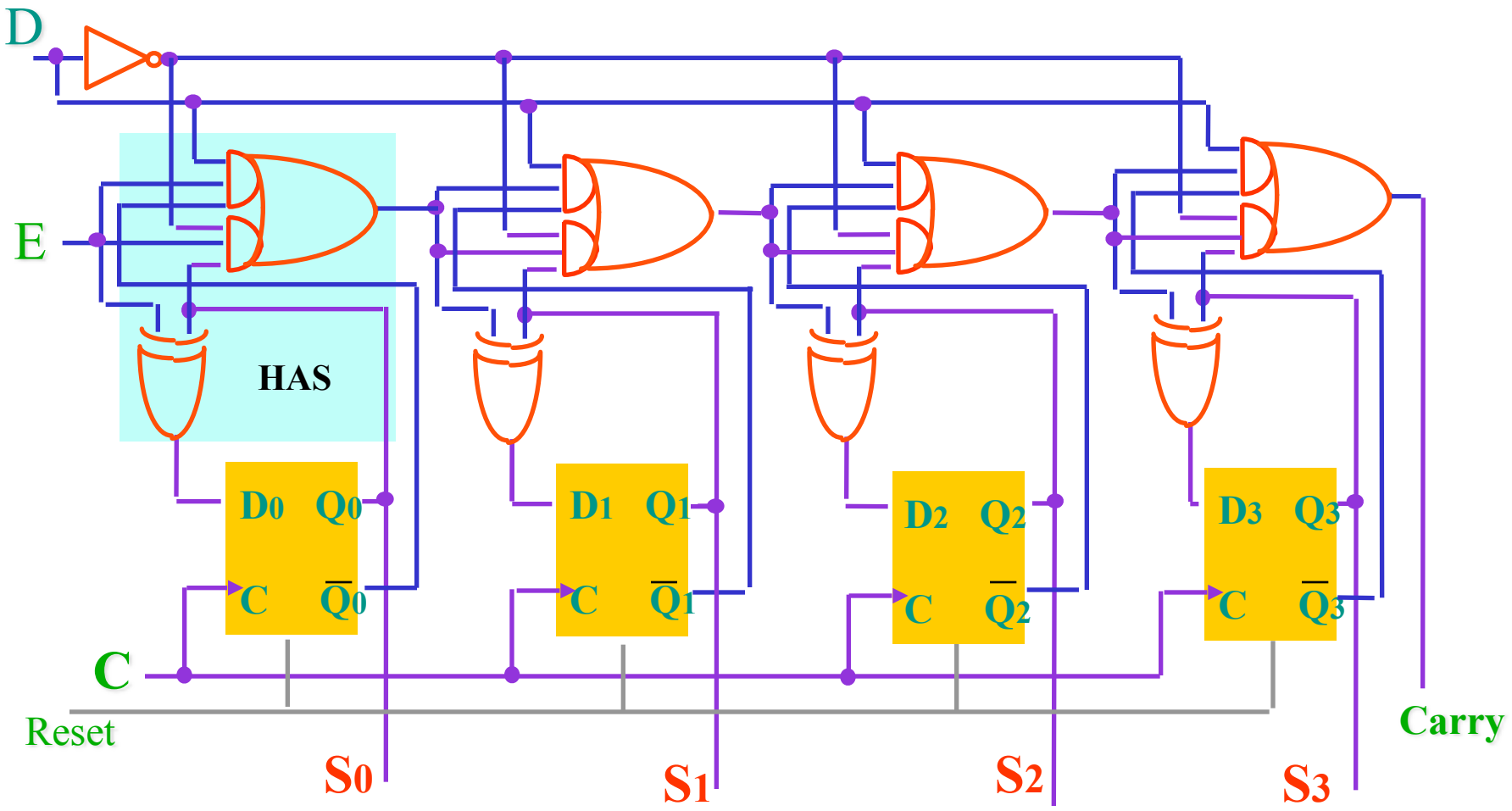
$$T0 = 1$$

Implementação do contador de módulo 5
(contador até 5)

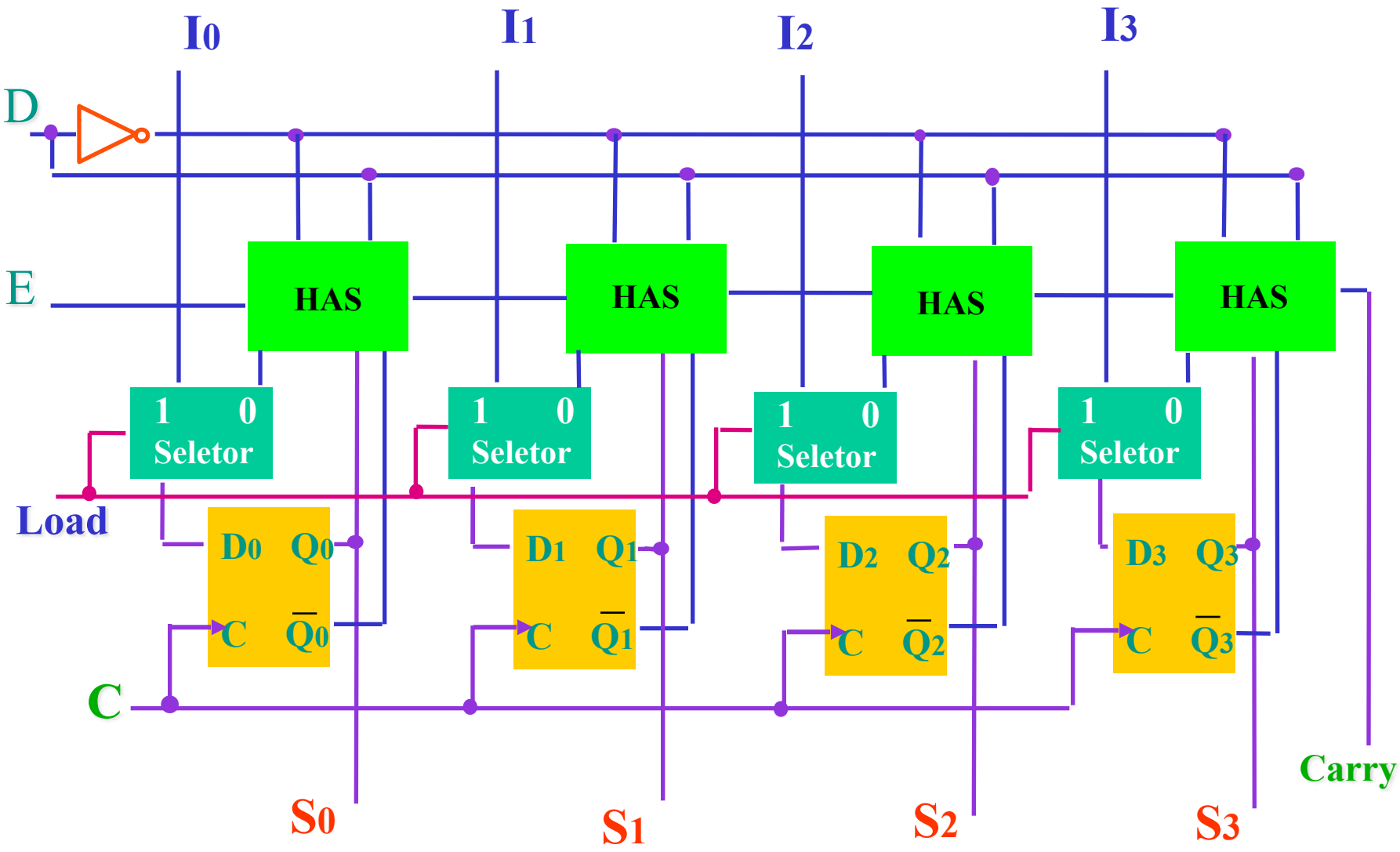


Contador binário up/down





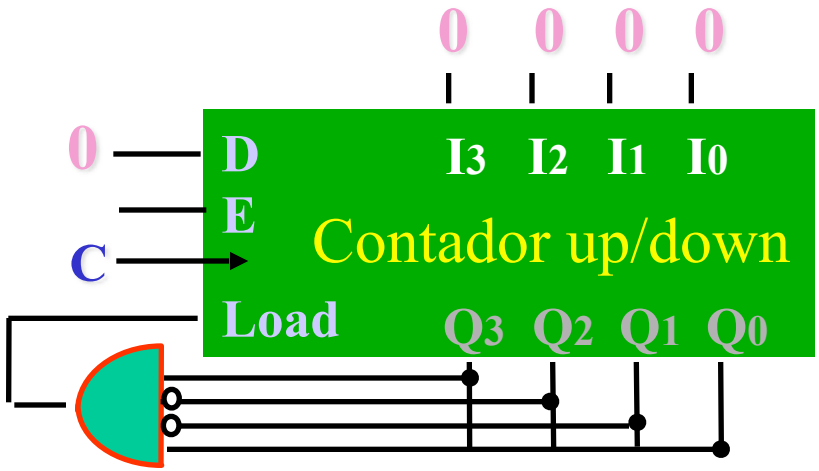
Contador up/down com carga paralela



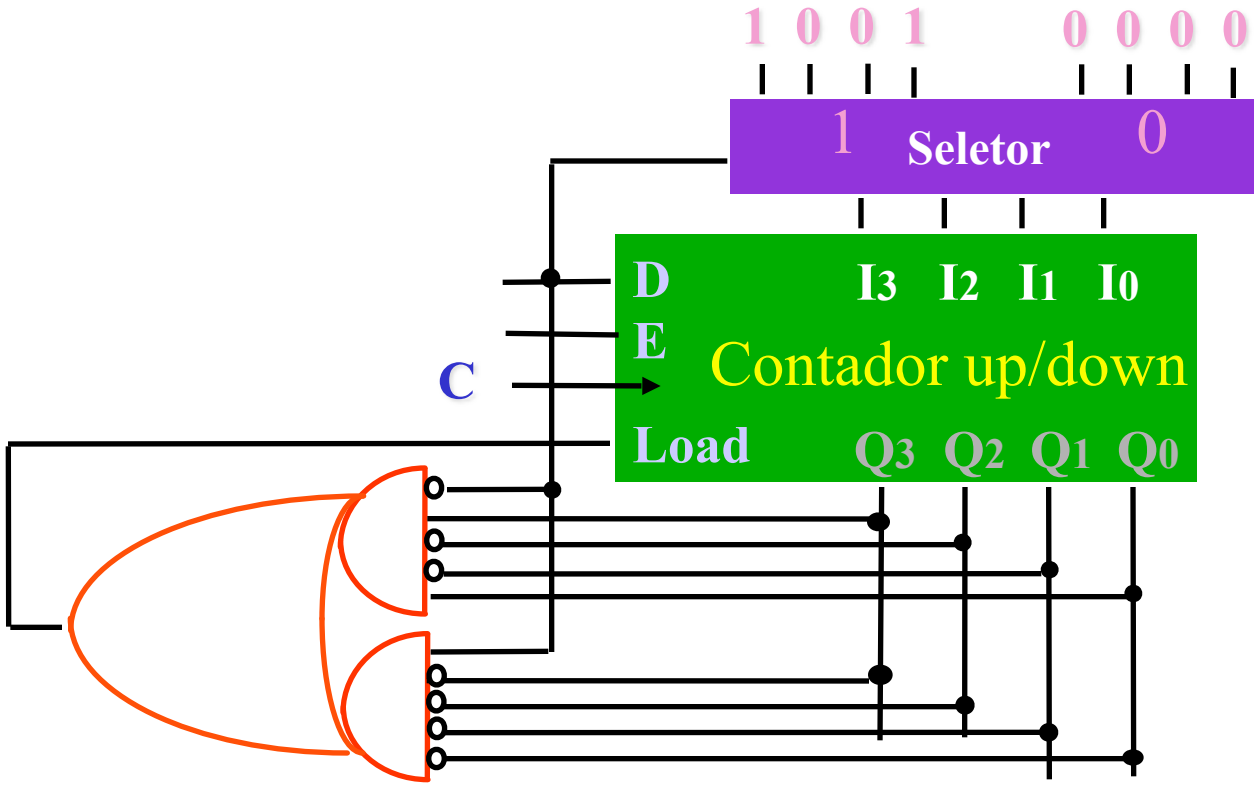
5. Contador BCD

Conta de 0 a 9 e então reinicia contagem.
Lógica combinacional detecta quando a contagem chega a 9.

Contador BCD para cima



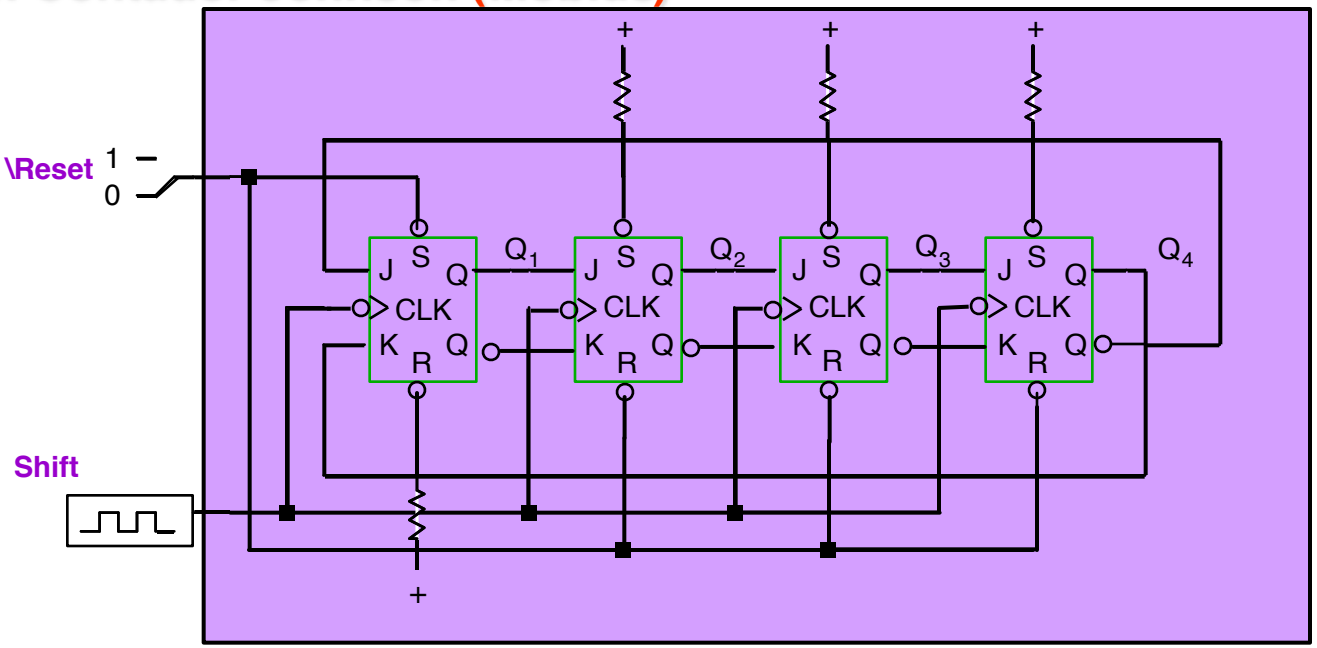
Contador BCD up/down

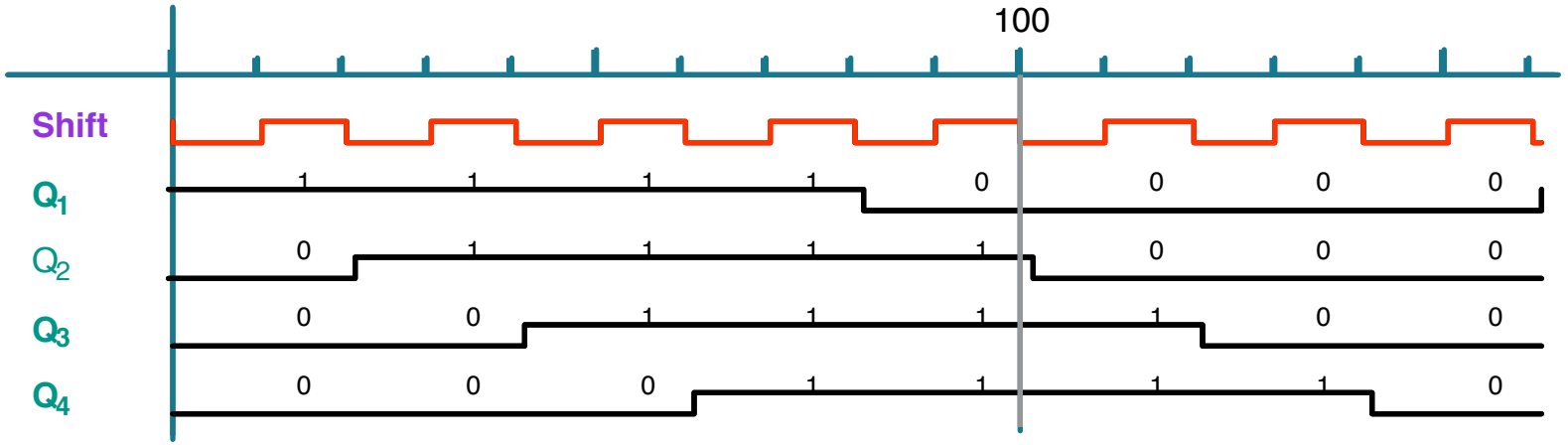


D = 0 ➡ contagem para cima (UP)

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