

Sistemas Digitais Contadores

João Paulo Carvalho

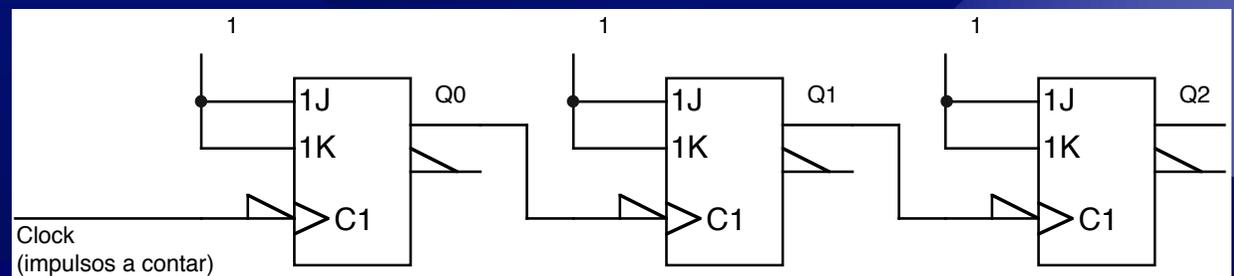
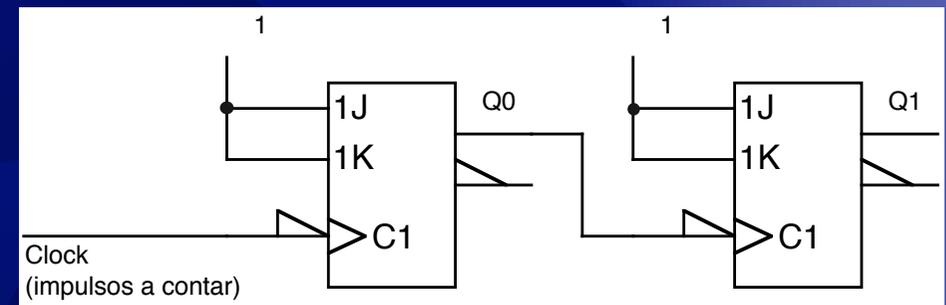
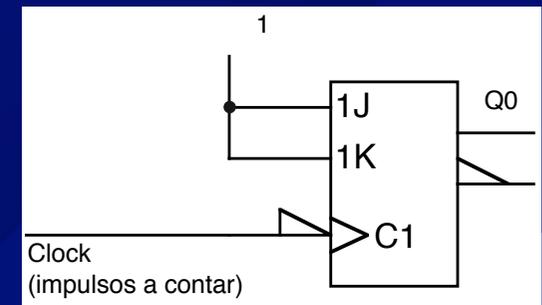


Contadores Assíncronos

- Um contador binário de 3 bits é um circuito que evolui controladamente ao longo da seguinte sequência (de contagem):

Sequência de contagem
Q2 Q1 Q0
0 0 0
0 0 1
0 1 0
0 1 1
1 0 0
1 0 1
1 1 0
1 1 1
0 0 0
...

- Um contador binário assíncrono pode ser implementado por FF's JK aproveitando a propriedade de Toggle quando *J* e *K* estão a '1':



Contadores Assíncronos (II)

Sequência de contagem

Q2 Q1 Q0

0 0 0

0 0 1

0 1 0

0 1 1

1 0 0

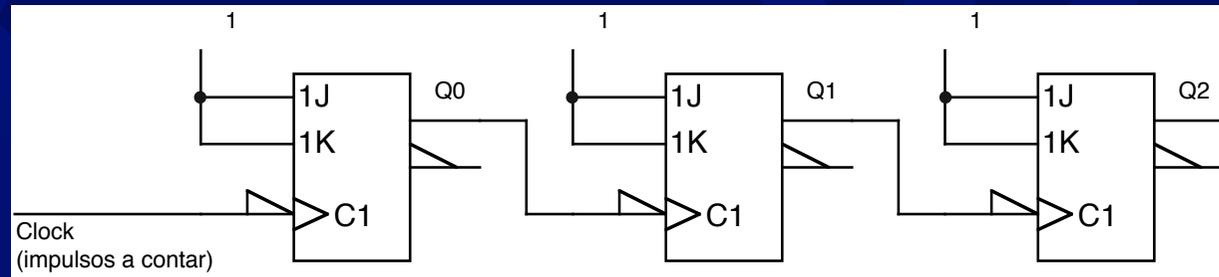
1 0 1

1 1 0

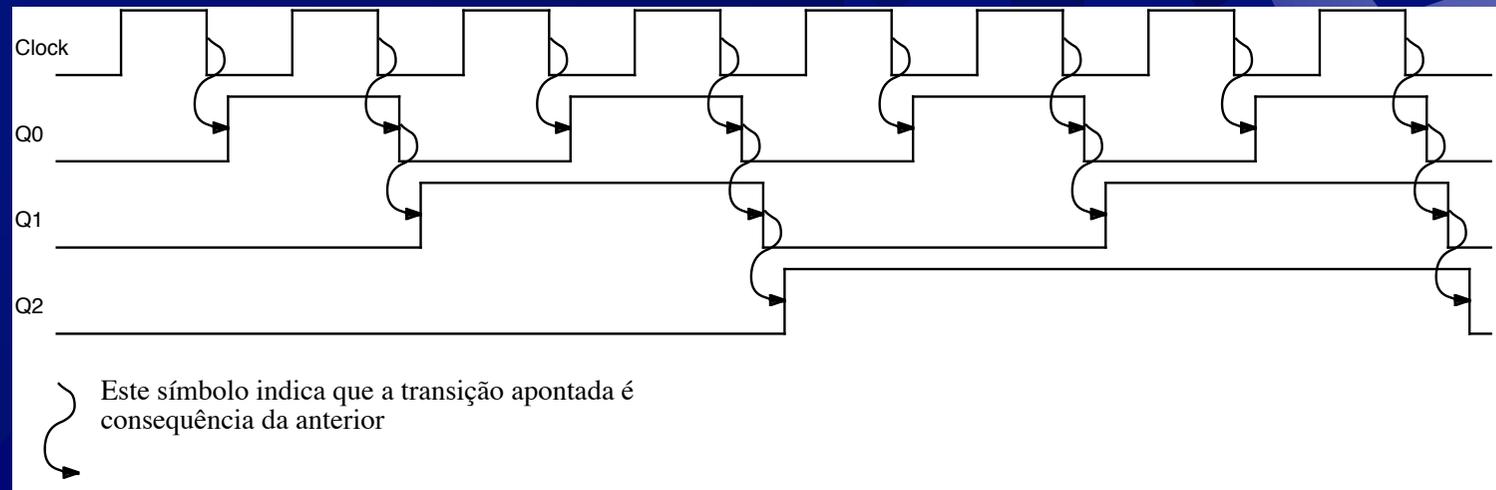
1 1 1

0 0 0

...



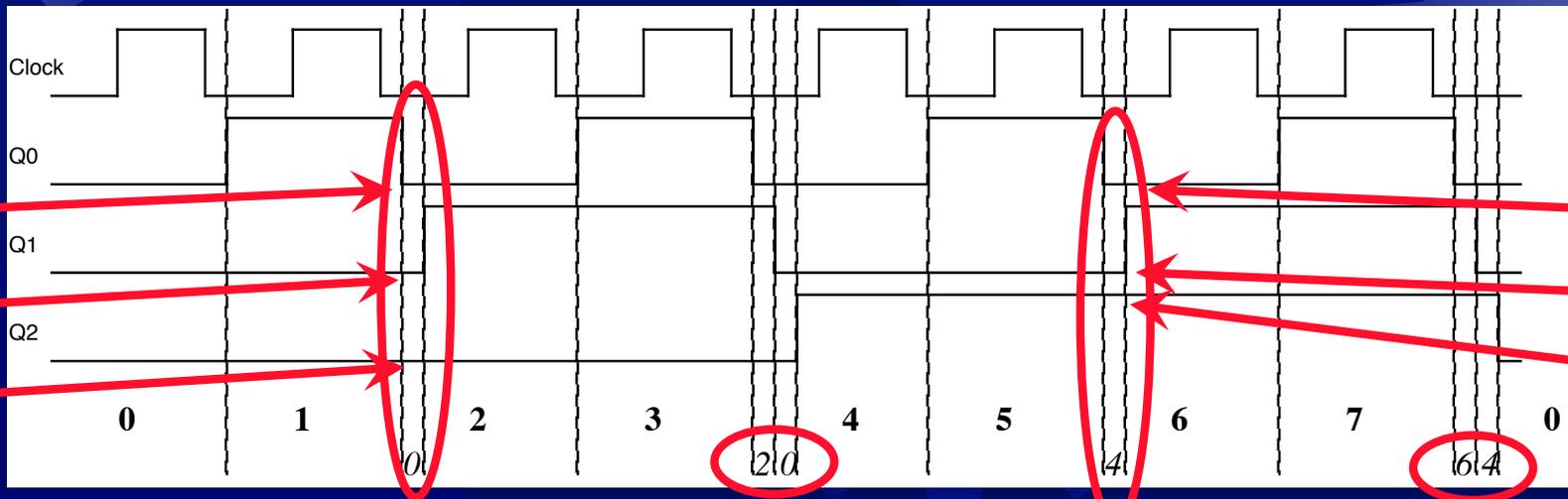
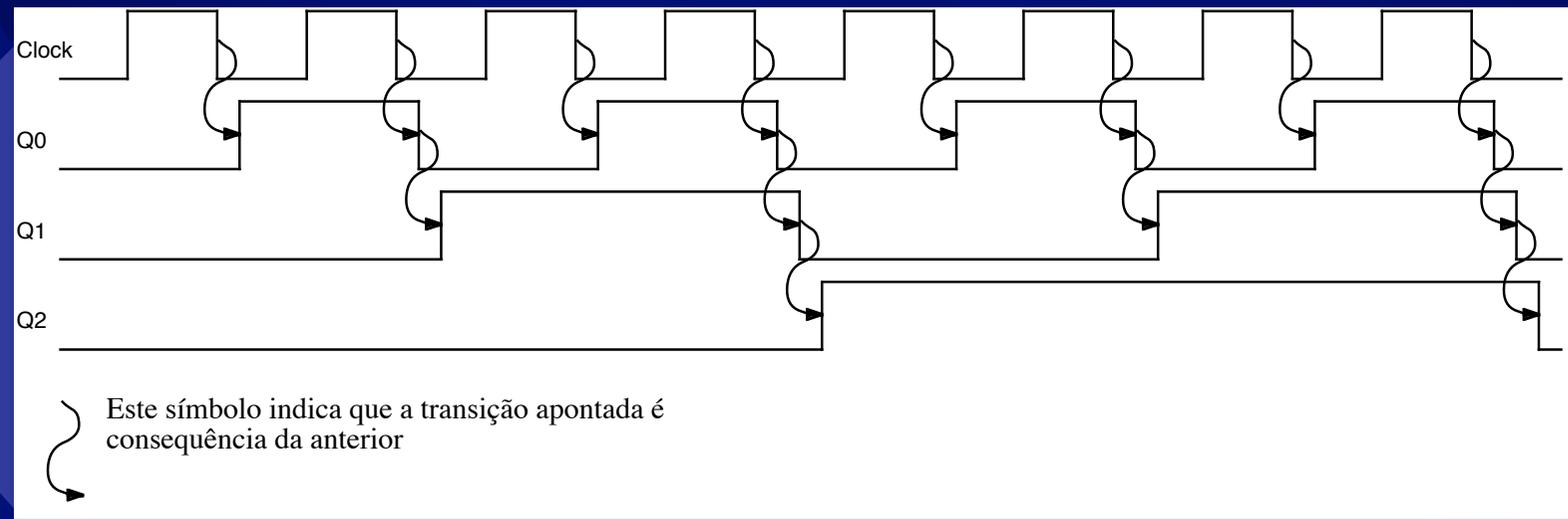
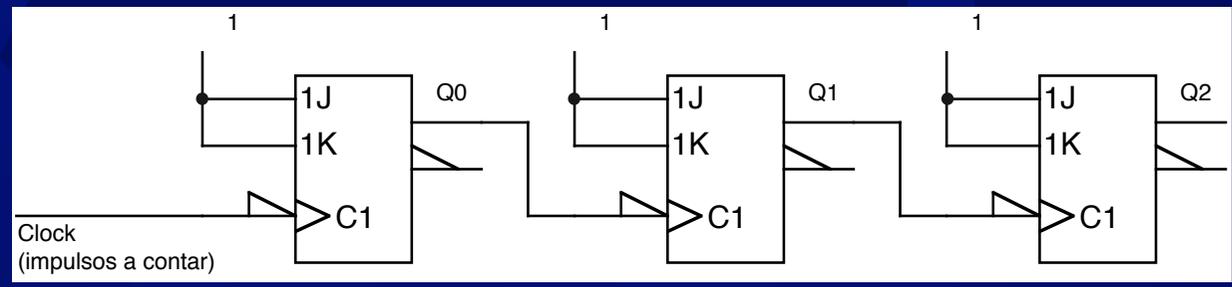
- ★ O diagrama temporal deste contador é o seguinte (de notar que se mostram os tempos de atraso dos FFs):



- ★ Os contadores assíncronos têm a desvantagem de possuírem estados transitórios que ocorrem sempre que um dos FF muda o estado de H para L

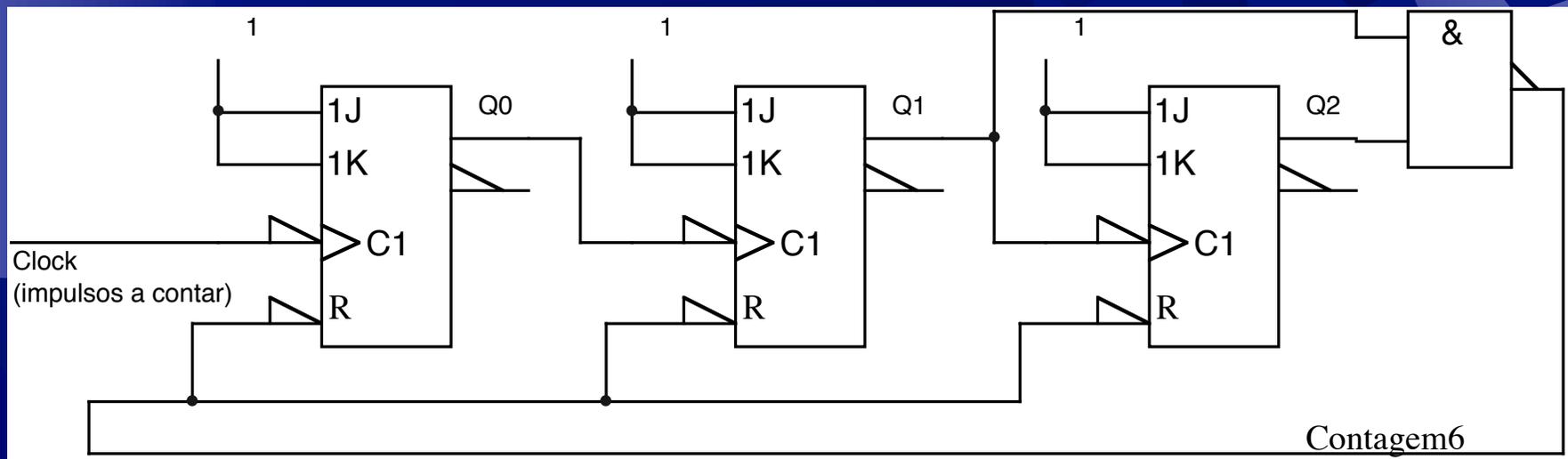


Contadores Assíncronos (III)



Contadores de Módulo não potência de 2

- Para fazer um contador assíncrono cujo módulo seja a potência n de 2, basta ligar n Flip-Flops JK em modo Toggle
- Para fazer um contador cujo módulo m seja diferente de uma potência de 2, utilizam-se n FFs (garantindo que $m < 2^n$), e força-se o contador a zero quando a contagem for m
- Exemplo: para um contador de módulo 6 (conta de 0 até 5), constrói-se um contador de módulo $2^3=8$, e faz-se *reset* ao detectar o 6 ($Q_2Q_1Q_0=110$):



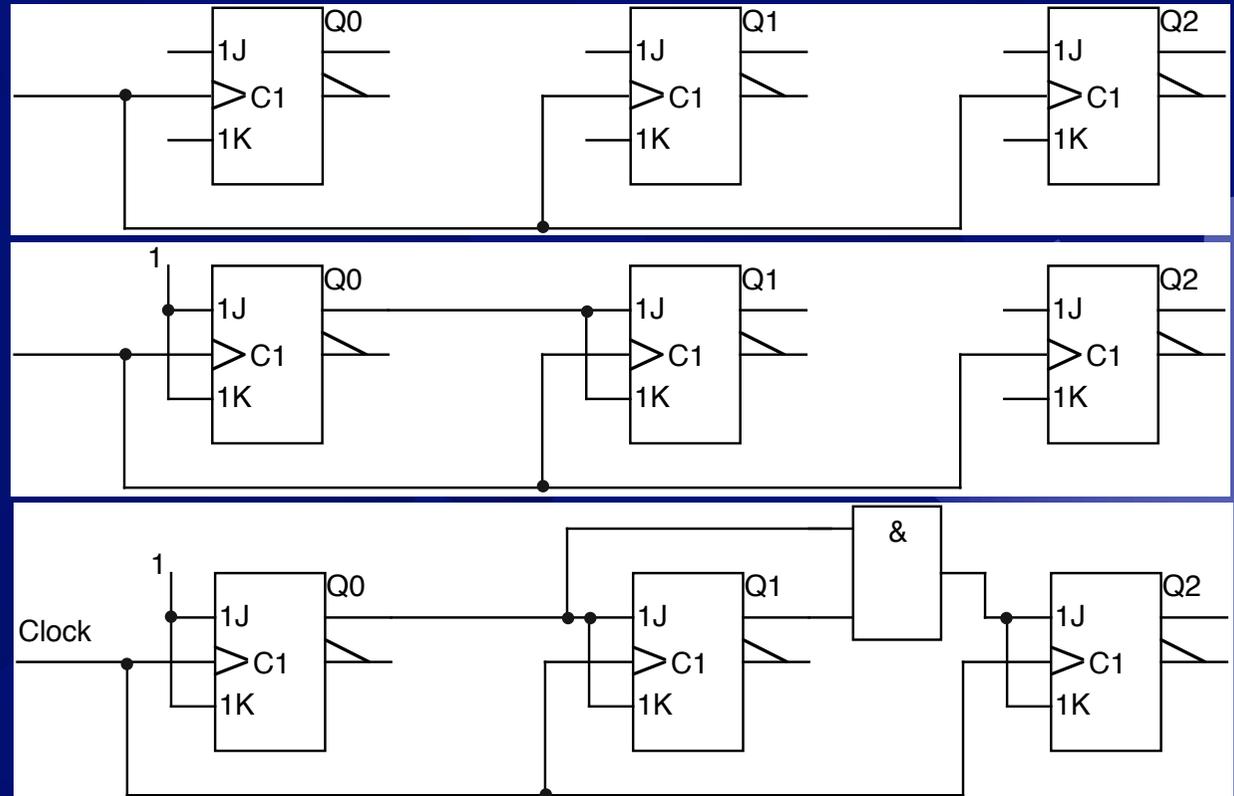
- Repare-se que para detectar o 110 basta “apanhar” os bits Q_2Q_1 a 1. Q_0 não é necessário porque 110 é a primeira situação a aparecer



Contadores Síncronos

- Para um contador ser síncrono, todos os FF devem ser actuados ao mesmo tempo. Logo, a linha de relógio deve ser comum
- Concepção heurística (i.e., “a olho”):

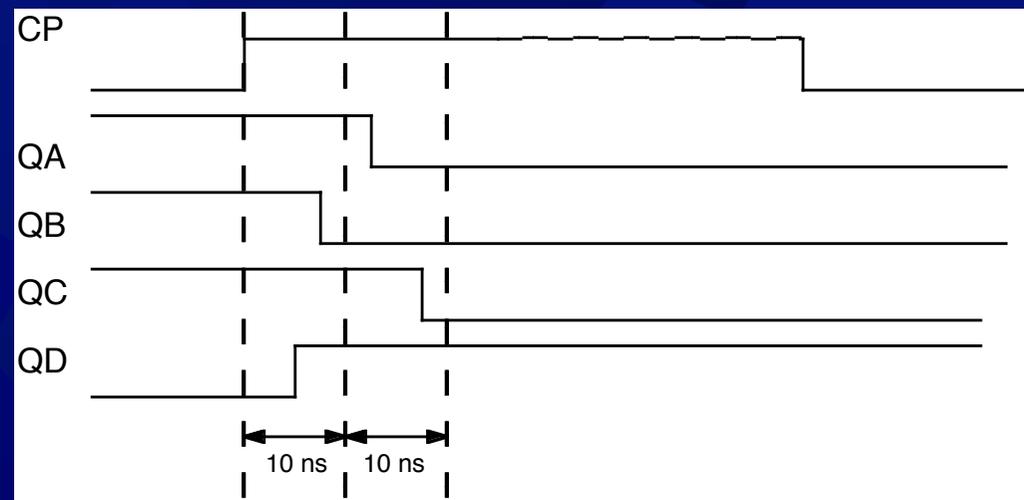
- Começa-se por 3 FF
- As entradas do FF₀ ligam-se a 1, pois deverá fazer Toggle a cada impulso de relógio
- Liga-se a saída do FF₀ à entrada do FF₁, pois pretende-se que o FF₁ faça Toggle quando o FF₀ está a 1
- Já a entrada do FF₂ não pode ser uma ligação directa de Q₁: só se quer que o FF₂ faça Toggle quando Q₀ e Q₁ estiverem simultaneamente a 1



- Generalizando para n FFs: a entrada de um FF deve ser activada pelo produto lógico de todas as saídas de peso inferior

Contadores Síncronos (Estados Instáveis)

- ✦ Apesar de todos os FF de um contador síncrono reagirem ao mesmo sinal de relógio, verifica-se que cada FF pode ser mais ou menos lento a reagir (devido a vários aspectos envolvidos no seu fabrico)
- ✦ Logo, num contador síncrono surgem estados instáveis sempre que numa mudança de estado está envolvido mais do que um FF
- ✦ Exemplo: Mudança de 7 (0111) para 8 (1000)



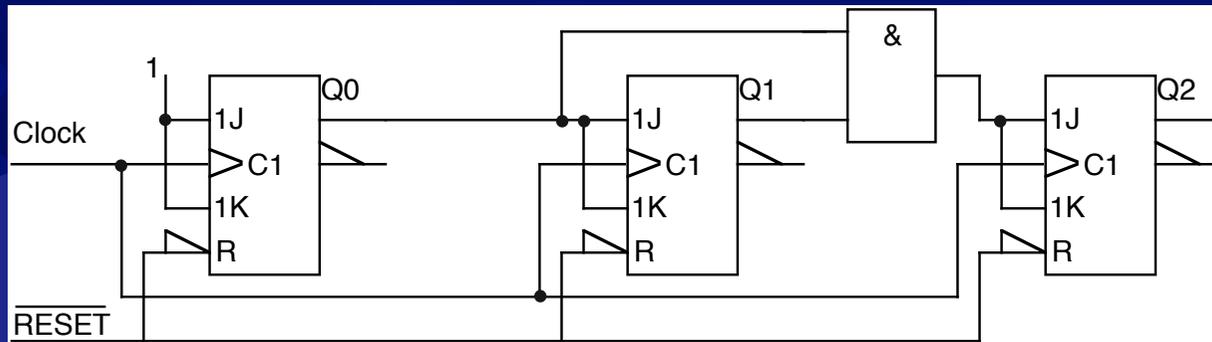
Contadores Síncronos (Estados Instáveis II)

- ✦ Nos contadores síncronos, os estados instáveis têm uma duração cuja ordem de grandeza é dada pelas diferenças entre os tempos de propagação dos FF que mudam de estado (valores por volta dos 5ns serão comuns em muitos contadores).
 - Num contador assíncrono, a ordem de grandeza será sempre superior à dos tempos de propagação dos FF (e aumentará com o n° de FF)
- ✦ De notar que num contador síncrono os estados instáveis não são previsíveis, nem em termos de sequência, nem em valor (enquanto que num assíncrono são – como vimos anteriormente)

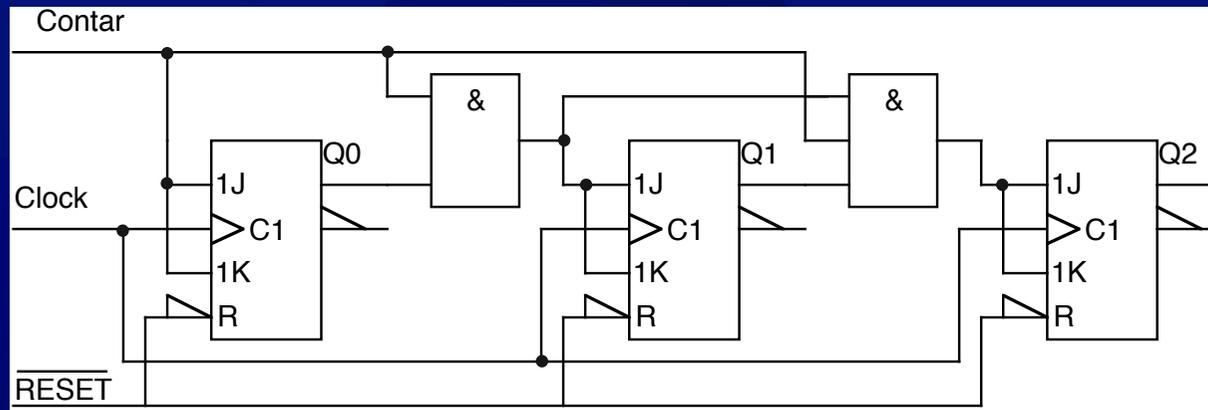


Contadores Síncronos com Reset e Enable

- Reset: Permite reiniciar o contador (i.e., colocá-lo a zero)



- Enable: Controla se o contador “conta” ou se mantém o estado presente



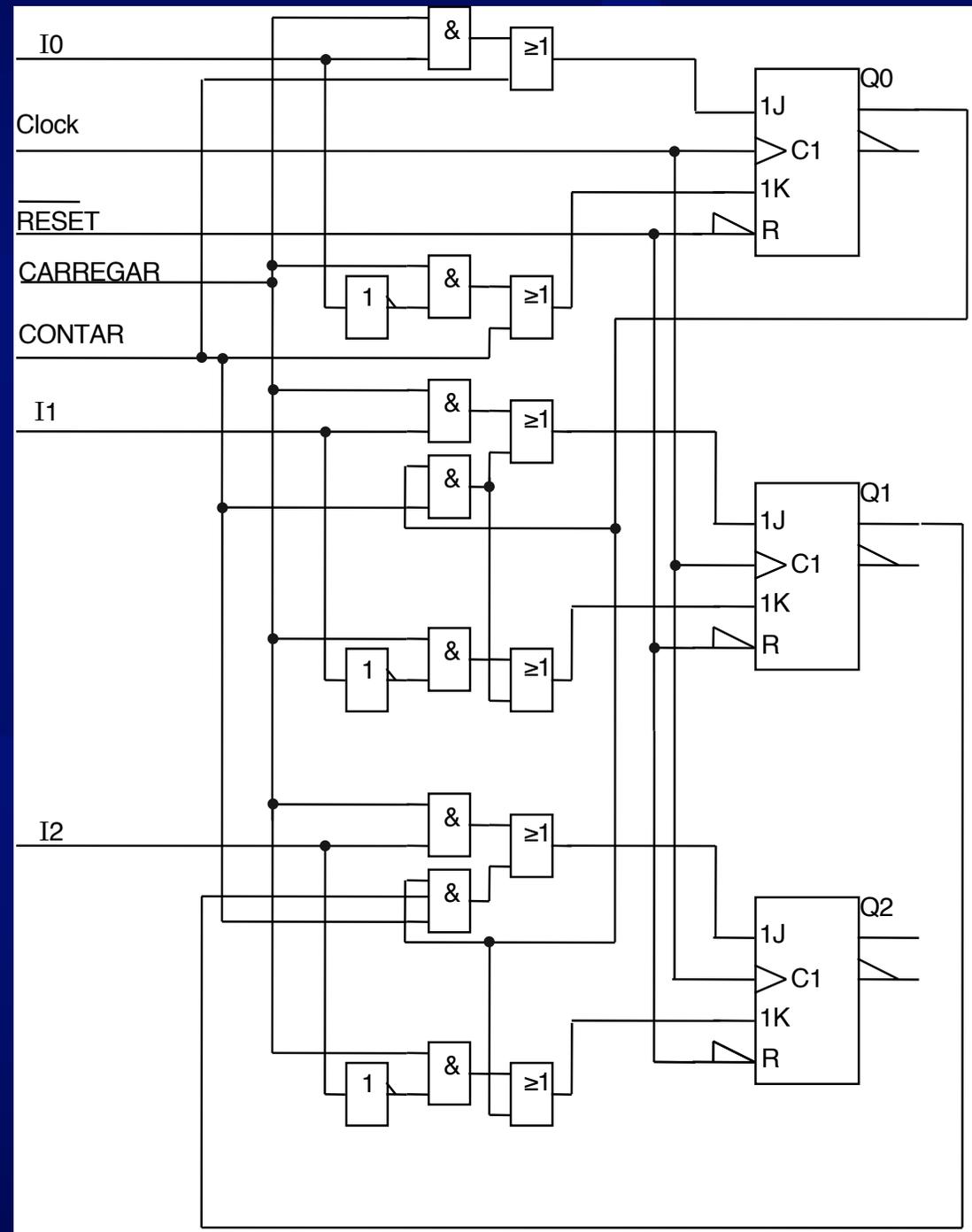
Contadores c/ Carregamento Paralelo

- A capacidade de poder inicializar um contador com um valor à escolha é por vezes muito importante
- Os contadores com Carregamento em Paralelo (LOAD) têm essa funcionalidade
- Para adicionar a funcionalidade de Carregamento em Paralelo a um contador, é necessário adicionar uma variável de controlo que comande o modo de funcionamento, e alterar a lógica que actua as entradas dos FFs
- Exemplo – Alterar um contador de módulo 8 para realizar carregamento paralelo:
 - Adicionamos uma linha de controlo *CARREGAR* que procede ao carregamento quando está a 1, e permite os outros modos quando a 0
 - Repare-se que para que um FF JK seja carregado com o valor de uma linha *I*, as suas entradas deverão ser $J = I, K = \bar{I}$
 - A lógica de J_0 e K_0 será determinada pela necessidade de carregar a entrada I_0 sempre que *CARREGAR* seja 1, e por ter de ser o valor da linha *CONTAR* quando *CARREGAR* for 0:
$$J_0 = \text{CARREGAR} \cdot I_0 + \text{CONTAR}$$
$$K_0 = \text{CARREGAR} \cdot \bar{I}_0 + \text{CONTAR}$$
 - As restantes linhas devem ser mais ou menos óbvias...



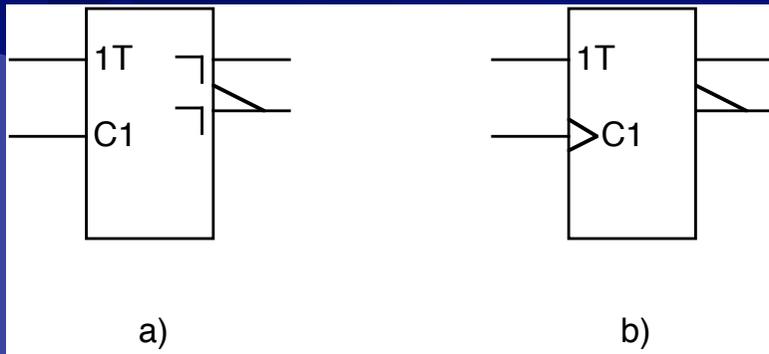
Contadores c/ Carregamento Paralelo (II)

- Contador síncrono de módulo 8 com Carregamento Paralelo e Reset:



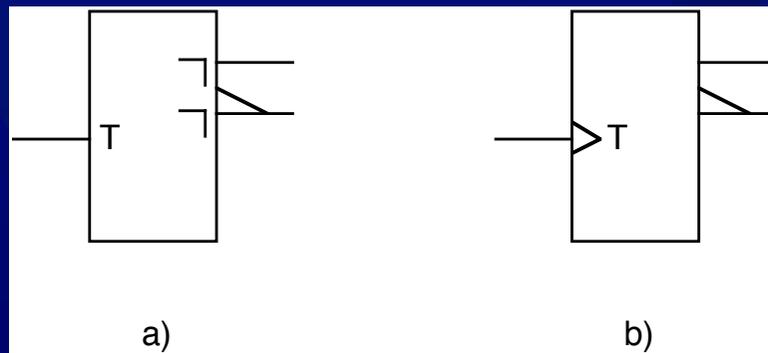
Flip-Flops do tipo T

- Os contadores dos slides anteriores poderiam ter sido implementados com FF do tipo T, que são equivalentes a um FF JK com as entradas ligadas entre si:



CP	T	$Q_{t+\Delta}$
	H	$\overline{Q_t}$
	L	Q_t
	X	Q_t
L	X	Q_t
H	X	Q_t

- Versão simplificada (com a entrada T sempre a H):



CP	$Q_{t+\Delta}$
	$\overline{Q_t}$
	Q_t
L	Q_t
H	Q_t

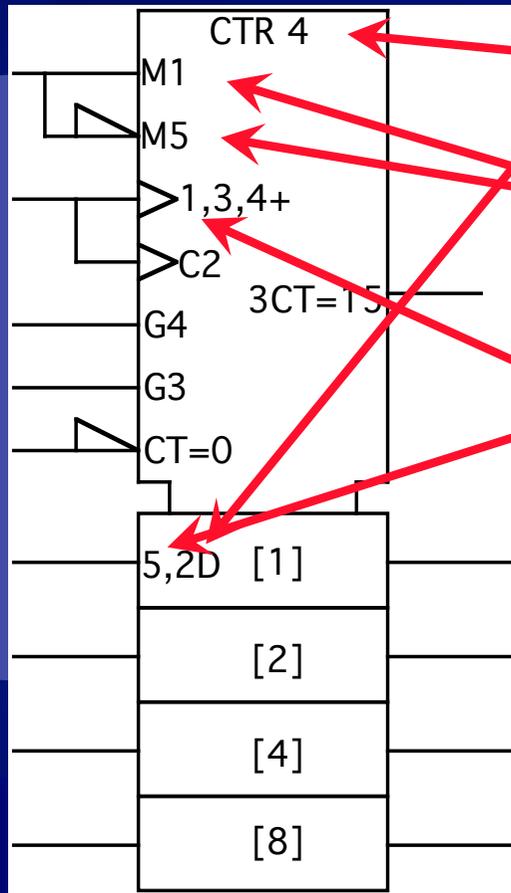


Contadores com vários modos

- ✦ É possível obter contadores com vários modos de funcionamento:
 - Contar;
 - Manter o estado;
 - Fazer *RESET* assíncrono;
 - Fazer carregamento em paralelo (*LOAD*) um determinado valor.
- ✦ Outro exemplo de modo de funcionamento é o que está associado aos contadores bidireccionais, *UP/DOWN*, i.e., que contam de forma ascendente ou descendente:
 - *UP* – 000, 001, 010, 011, 100, 101, 110, 111, 000, 001, ...
 - *DOWN* – 111, 110, 101, 100, 011, 010, 001, 000, 111, 110, ..., etc.
- ✦ Existem no mercado circuitos integrados que possuem todas estas funcionalidades. É necessário saber a simbologia de um contador para perceber o seu funcionamento.



Contadores Complexos (Simbologia)



CTR 4 – Contador de 4 bits construído a partir de FF de tipo D

M1, M5 – Modos de funcionamento

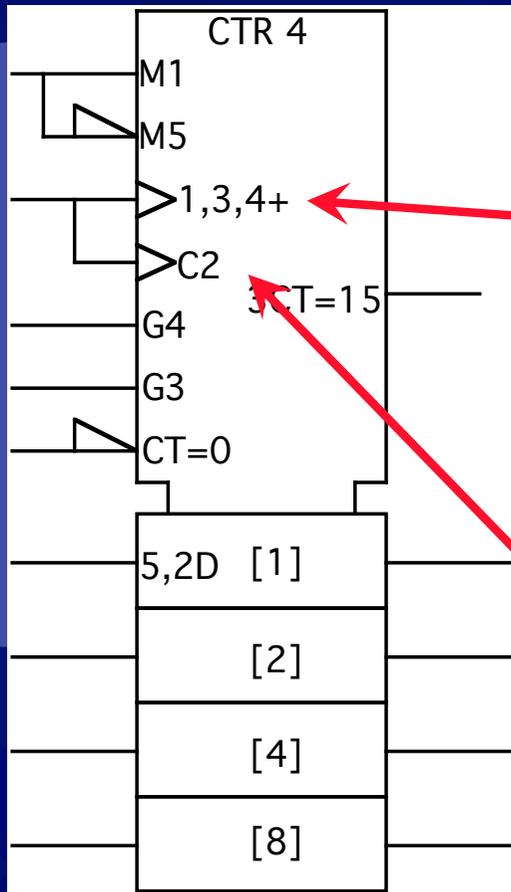
- M1 – conta (a linha está a H)
- M5 – carrega em paralelo (a linha está a L)

A linha de Modo define dois modos diferentes. Os valores 1 e 5 apenas servem para indicar quais as linhas que dependem desta linha. Para se saber o que faz cada modo é necessário analisar todo o símbolo e procurar os valores 1 e 5



Contadores Complexos (Simbologia II)

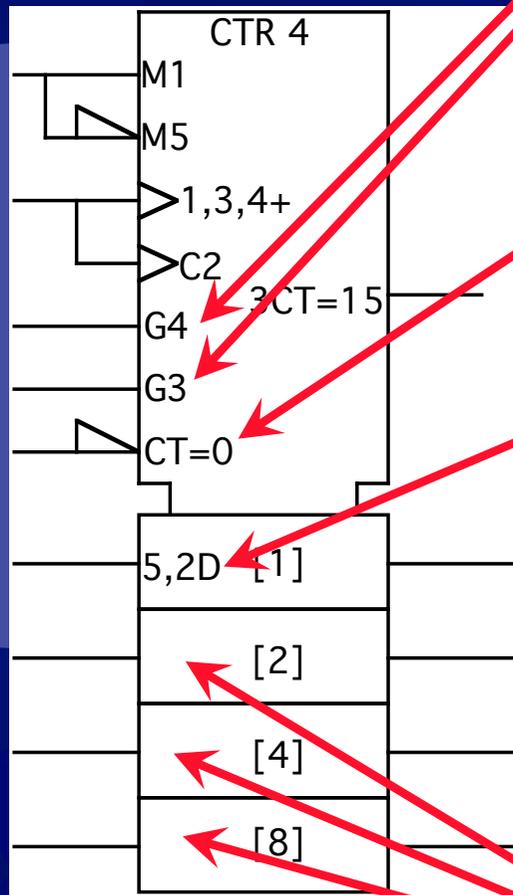
- ☀ Linha de relógio: A linha é dividida em duas para facilidade de leitura do símbolo. Isso pode ser interpretado como tendo a linha duas funções (tal como acontece com a linha de Modo)



- 1,3,4+: Indica que, a verificarem-se as condições 1, 3 e 4, o contador conta ascendentemente (o sinal '+' significa contagem ascendente e o '-' significa contagem descendente). Portanto, quando houver um flanco ascendente no relógio (o triângulo), e as linhas 1, 3 e 4 estiverem activas, o contador conta.
- C2: É uma linha de clock que serve para despoletar outras operações descritas algures no símbolo (o carregamento paralelo, como veremos)



Contadores Complexos (Simbologia III)



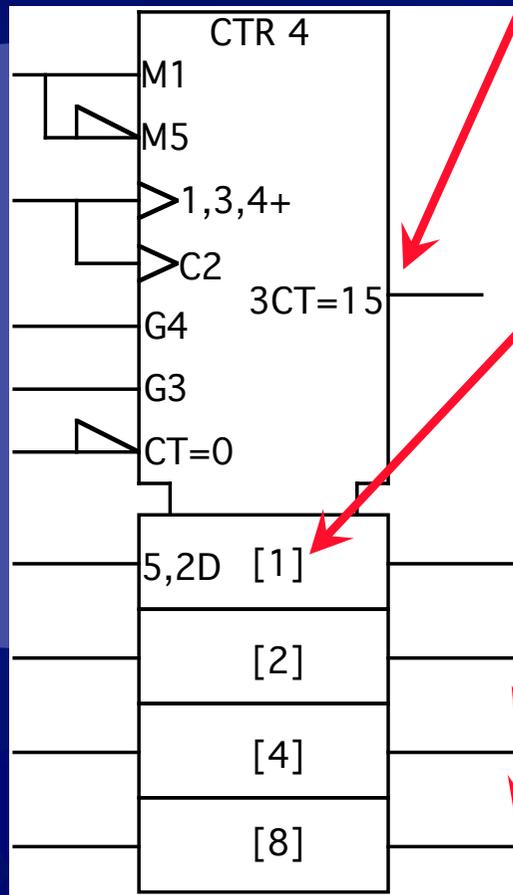
G3 e G4: Enables de contagem. A diferença entre os dois está em que um deles influencia a saída (de que já trataremos) enquanto que o outro, não.

CT = 0: Linha de Reset assíncrono do contador. Quando accionada coloca a contagem a 0 (CT=0).

5,2D: Entrada de carregamento paralelo de cada um dos FF. O carregamento dá-se quando a linha 5 está activa e quando surge um flanco activo no relógio (C2). No caso de ser 5D, isso significaria que o carregamento seria assíncrono, isto é, que se verificava logo que a linha de modo era activada, não dependendo do relógio. A notação é semelhante para cada um dos 4 FFs, mas não está repetida



Contadores Complexos (Simbologia IV)



☀ 3CT=15: Linha de saída que indica que a contagem do contador atingiu o último estado de contagem (15). A linha fica a High enquanto o contador estiver no estado 15 (1111). A saída só é activada quando o enable 3 está activo

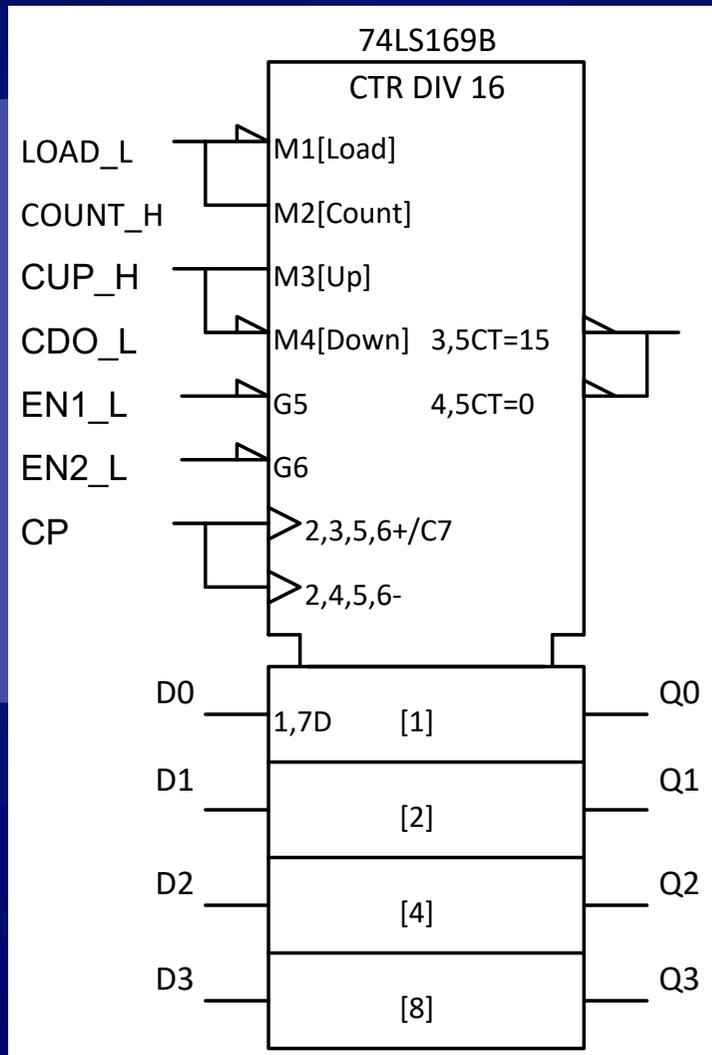
☀ [1], [2], [4] e [8]: Comentários (tudo o que estiver dentro de parêntesis rectos é comentário). Indicam qual o peso dos diversos flip-flops na contagem

☀ Saídas (é onde se vê a contagem...)



Contadores Complexos (Simbologia V)

Exemplo de contador Up/Down de 4 bits:



- Reparar na existência de modos de contagem ascendente (2,3+), descendente (2,4-) e carregamento paralelo síncrono (1,7)
- G5 e G6 são Enables
- M1 e M2 indicam se há contagem ou carregamento paralelo
- M3 e M4 definem a direcção da contagem
- O contador assinala quando chega a 15 quando está em modo ascendente, e a 0 quando está em modo descendente

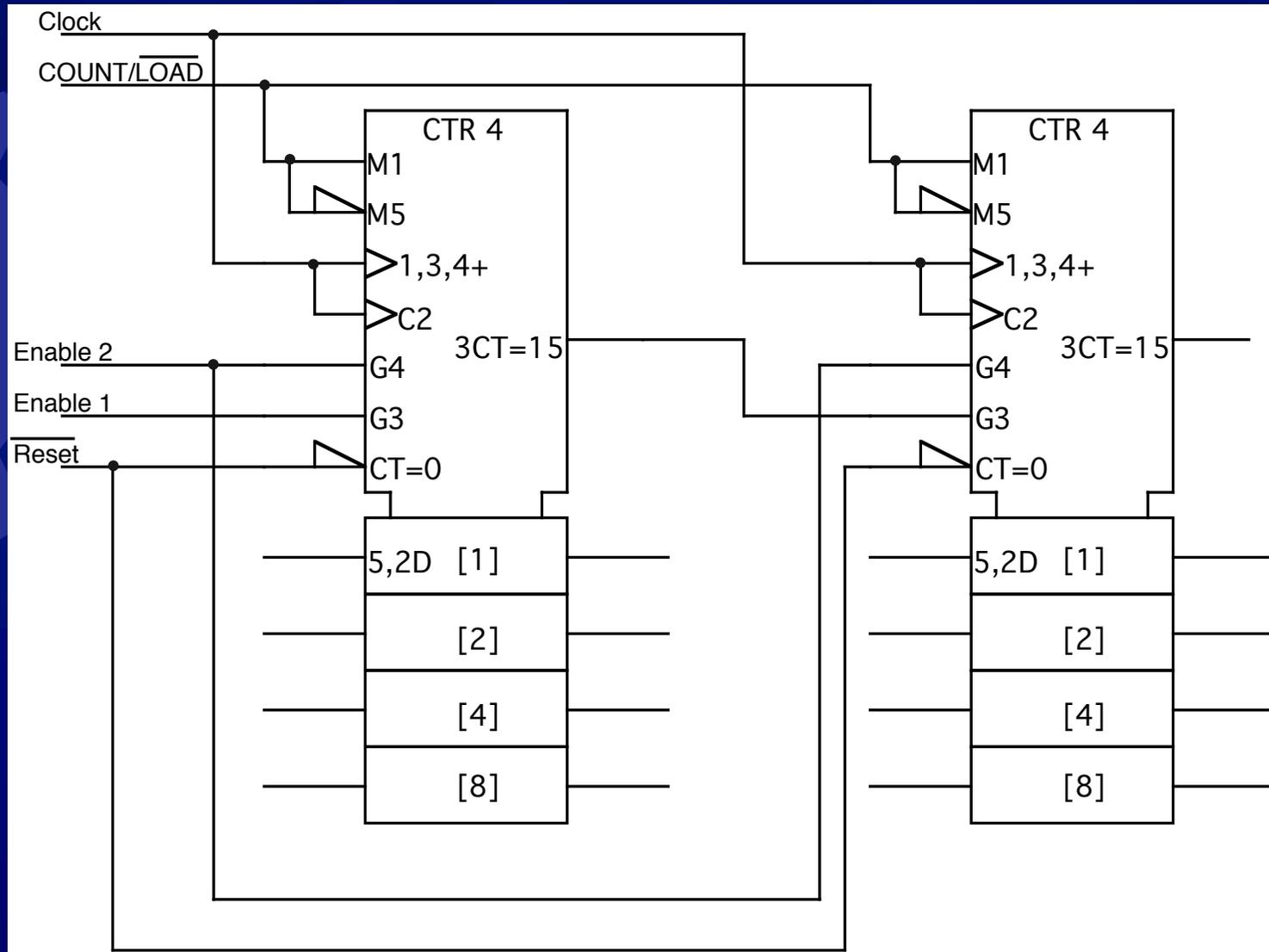
Interligação de Contadores integrados

- ✦ Muitas vezes é necessário proceder à interligação entre vários contadores integrados para aumentar o módulo de contagem (por exemplo, utilizar contadores de 4 bits para fazer um contador de módulo 100...)
- ✦ Uma solução óbvia é interligar os contadores de modo que um dos contadores seja o menos significativo e os seguintes contem apenas quando todos os anteriores chegam ao último estado de contagem e passam para o primeiro estado
- ✦ Afim de manter o carácter síncrono, a solução passa por interligar os contadores de forma a que o relógio seja comum e que o enable de cada um dependa do conjunto dos menos significativos terem chegado ao último estado de contagem



Interligação de Contadores integrados (II)

- Exemplo: Interligação de 2 contadores de 4 bits. O módulo de contagem é $16 \times 16 = 256$



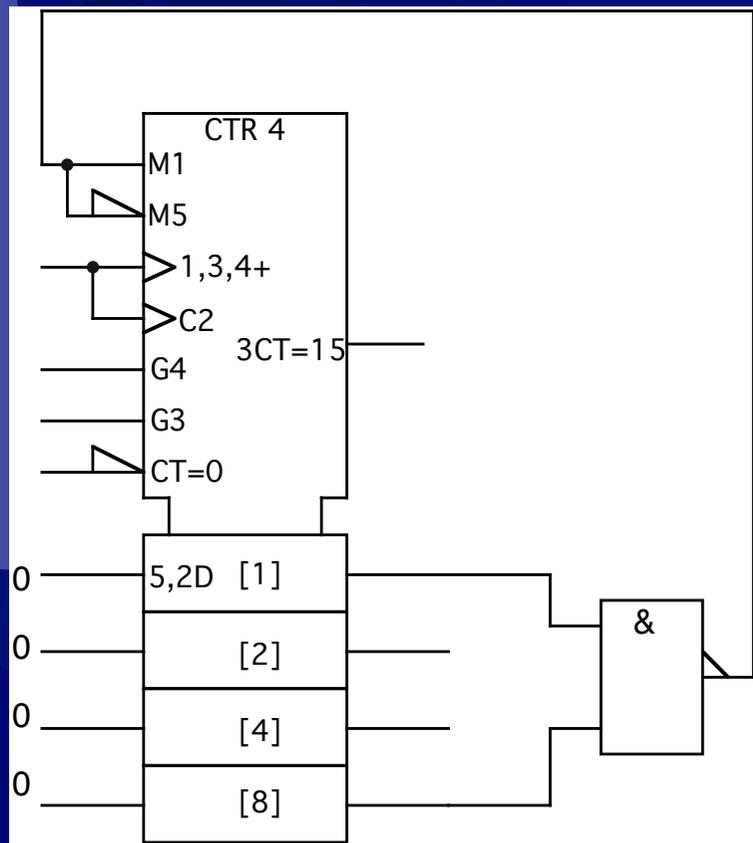
Utilização do carregamento paralelo para realizar diferentes módulos de contagem

- ✦ O carregamento paralelo dos contadores tem ainda a utilidade de permitir modificar os módulos de contagem
- ✦ Suponhamos que pretendemos transformar um contador de módulo 16 num de módulo 10. A solução está em detectar o último estado de contagem pretendido e usar essa linha para activar o carregamento do contador
- ✦ As linhas de carregamento paralelo serão utilizadas para impor o número zero. Note-se que não pode ser usado o Reset uma vez que ele é assíncrono e, supostamente, pretendemos manter o contador síncrono



Utilização do carregamento paralelo para realizar diferentes módulos de contagem (II)

- Exemplo: Obter um contador de módulo 10 (conta de 0000 a 1001) a partir de um contador integrado de módulo 16



- A primeira vez que, na sequência de contagem, se encontram os dois bits dos extremos a 1, é o número 9, pelo que ao detectarmos apenas essa situação, detectamos o 9
- Poderia ter-se usado outro valor para carregar nas entradas de carregamento paralelo se a sequência pretendida a isso obrigasse. Se se pretendesse, por exemplo, a sequência4,5,6,7,8,9,4,... então detectar-se-ia o 9 da mesma forma, mas em vez de carregar o número 0, carregar-se-ia o número 4

Concepção de um contador de módulo m genérico

- ✦ Pretende-se conceber um contador síncrono utilizando um procedimento sistemático de síntese
- ✦ Este método permite conceber contadores genéricos independentemente de o módulo de contagem ser ou não potência de 2
- ✦ Exemplo: contador síncrono de módulo 6 (conta de 0 a 5)

Estado Presente

Estado Seguinte

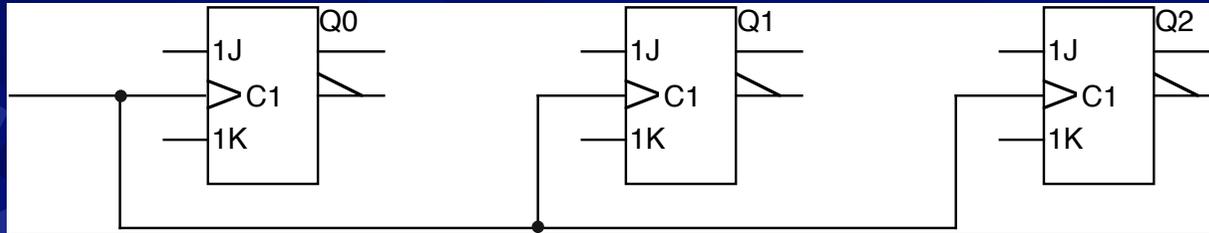
E.P.			E.S.		
Q2	Q1	Q0	Q2	Q1	Q0
000			001		
001			010		
010			011		
011			100		
100			101		
101			000		

- ✦ Começa-se por determinar o número de FFs necessário (3)
- ✦ Depois preenche-se uma tabela em que se indica qual o estado que sucede cada estado
- ✦ Note-se que as combinações 110 e 111 foram deliberadamente deixadas de fora por estarem fora da contagem



Concepção de um contador de módulo m genérico (II)

- Para contar até 5 são necessários 3 FFs JK:



- A tarefa de projectar o circuito resume-se a decidir que funções lógicas utilizar para atacar as entradas J e K de cada um dos FFs
- Existem sempre 2 opções:
 - Pretende-se colocar a saída do FF a L;
 - Pretende-se colocar a saída do FF a H.
- Para cada opção é necessário saber que valores se devem colocar nas entradas. Para isso existe a chamada “Tabela de Excitação” do FF JK:

Q-Transição		JK
de	para	
0	0	0X
0	1	1X
1	0	X1
1	1	X0

Concepção de um contador de módulo m genérico (III)

- ✦ A partir da tabela de excitação dos FFs e da tabela de estados, obtém-se a tabela de excitação das entradas:

Q-Transição		JK
de	para	
0	0	0X
0	1	1X
1	0	X1
1	1	X0

E.P.			E.S.		
Q ₂	Q ₁	Q ₀	Q ₂	Q ₁	Q ₀
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	0	0
1	0	0	1	0	1
1	0	1	1	1	0

E.P.			E.S.		
Q ₂	Q ₁	Q ₀	J ₂ K ₂	J ₁ K ₁	J ₀ K ₀
0	0	0	0 X	0 X	1 X
0	0	1	0 X	1 X	X 1
0	1	0	0 X	X 0	1 X
0	1	1	1 X	X 1	X 1
1	0	0	X 0	0 X	1 X
1	0	1	X 1	0 X	X 1

- ✦ Cada coluna J₀, K₀, J₁, etc. corresponde a uma função que pode ser simplificada através dos métodos usuais (mapa de Karnaugh...)
- ✦ Essa função de excitação deverá ser colocada na entrada respectiva



Concepção de um contador de módulo m genérico (cont. IV)

- Simplificando as funções, obtém-se:

		J2				J1			
		Q1	Q0			Q1	Q0		
Q2		00	01	11	10	00	01	11	10
0		0	0	1	0	0	1	X	X
1		X	X	X	X	0	0	X	X

		K2				K1			
		Q1	Q0			Q1	Q0		
Q2		00	01	11	10	00	01	11	10
0		X	X	X	X	X	X	1	0
1		0	1	X	X	X	X	X	X

$$J2 = Q1 Q0$$

$$K2 = Q0$$

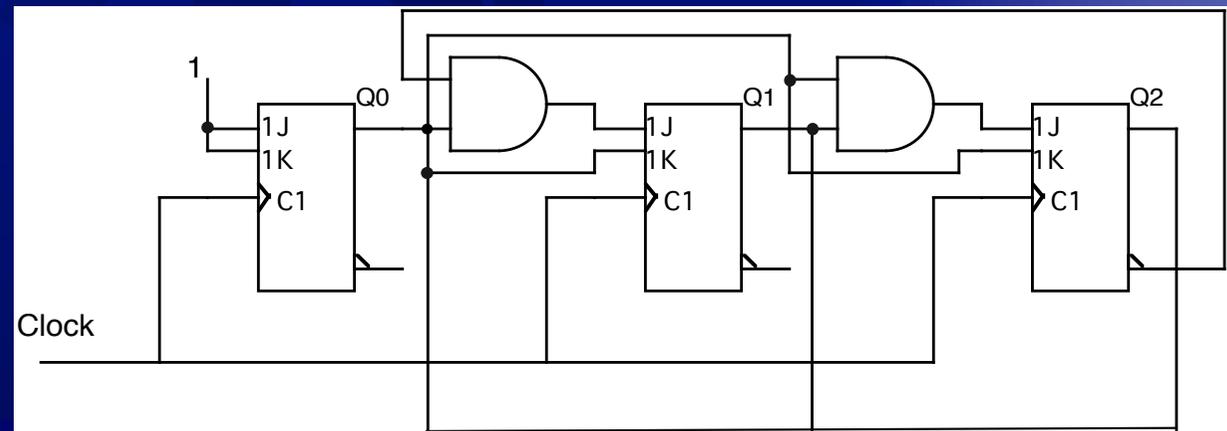
$$J1 = Q0 \overline{Q2}$$

$$K1 = Q0$$

$$J0 = 1$$

$$K0 = 1$$

- O circuito final é:



Projecto de Contadores Bidireccionais

- ✦ É possível projectar contadores bidireccionais, UP/DOWN, i.e., que contam de forma ascendente ou descendente
- ✦ O método de projecto é semelhante ao que vimos para projectar contadores de módulo genérico
- ✦ Exemplo: Projectar um contador Up/Down de módulo 10 (conta de 0 a 9):

E.P.	E.S.	
	Up/down = 0	Up/down = 1
0000	0001	1001
0001	0010	0000
0010	0011	0001
0011	0100	0010
0100	0101	0011
0101	0110	0100
0110	0111	0101
0111	1000	0110
1000	1001	0111
1001	0000	1000

Estado Presente

Estado Seguinte

- ✦ Up/Down = 0, contagem ascendente
- ✦ Up/Down = 1, contagem descendente
(depois segue-se o processo de síntese anterior, tendo em atenção que as funções J e K dependem também de Up/Down)



Bibliografia

- ✦ Arroz,G., Monteiro,J.C., Oliveira,A.,
“Arquitectura de Computadores, dos
Sistemas Digitais aos Microprocessadores”,
secção 6.6, 2ª Edição, 2009
- ✦ Mano,M., Kime,C. – “Logic and Computer
Design Fundamentals”, Prentice Hall, secções
5.5, 5.7, 6.1 a 6.4

