

Capítulo VIII

Registadores de Deslocamento e Contadores

1 Introdução

• Vimos no capítulo anterior que *flip-flops* são dispositivos capazes de “memorizar” o seu estado (SET ou RESET).

• Neste capítulo estudaremos dois circuitos digitais que constituem aplicações fundamentais de *flip-flops*:

- (I) **Registrador de Deslocamento:** Circuito digital cujo objetivo é converter dados binários entre o formato paralelo e o formato serial.
- (II) **Contador:** Circuito digital cujo objetivo é gerar uma seqüência numérica .

2 Registradores de Deslocamento (*shift register*)

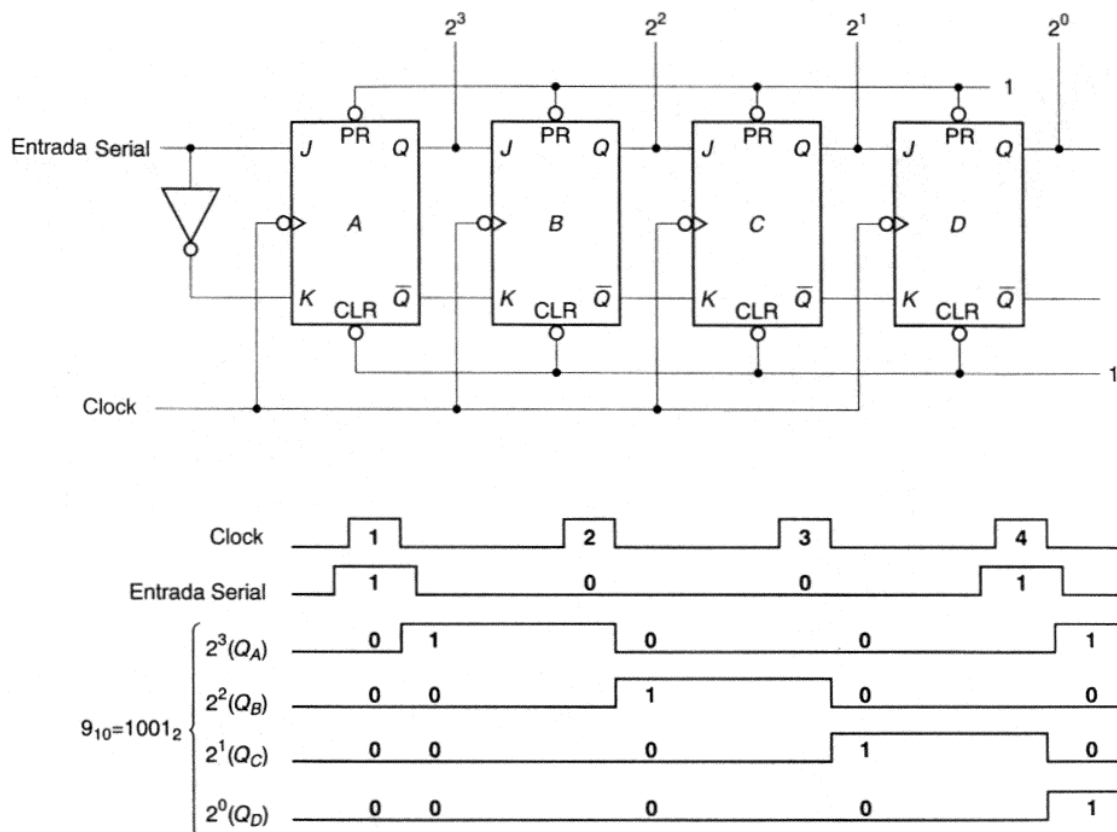


Figura 1: Registrador de deslocamento de 4 bits com entrada serial e saída paralela feito com *flip-flops* JK.

- Note na Figura 1 a porta NOT entre as entradas J e K do *flip-flop* A ($K = \bar{J}$), fazendo com que o valor lógico nas entradas J e $K = \bar{J}$ seja transferido respectivamente às saídas Q e \bar{Q} no instante em que ocorre a borda de descida do *clock* (comporta-se semelhantemente a um *flip-flop* D).
- Note que as saídas Q e \bar{Q} de cada *flip-flop* são interligadas com as entradas J e K do *flip-flop* seguinte. Por causa disto, a cada borda de descida do *clock* as saídas Q e \bar{Q} de cada *flip-flop* são transferidas para a saídas do *flip-flop* seguinte.

Exemplo 1: Determine a saída do *shift register* da Figura 1 quando a seqüência de bits representativa do número 5 é aplicada na entrada serial.

Solução:

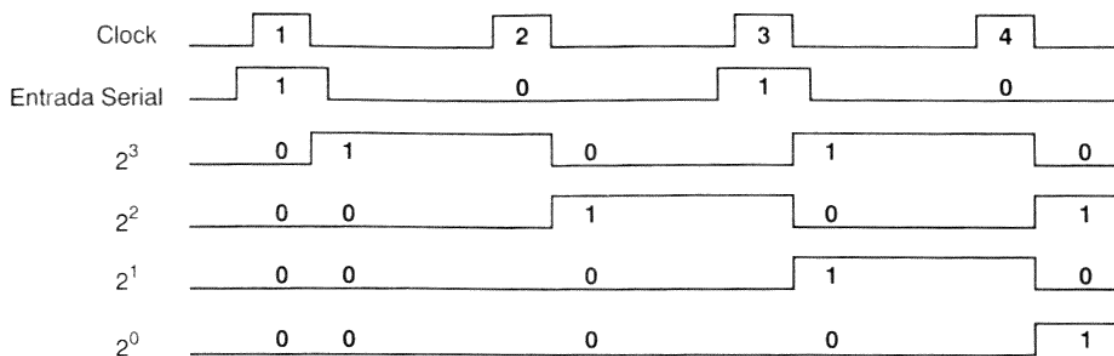


Figura 2: Formas de onda para um registrador de deslocamento de 4 bits com entrada serial e saída paralela com seqüência de bits representativa do número 5 aplicada na entrada serial.

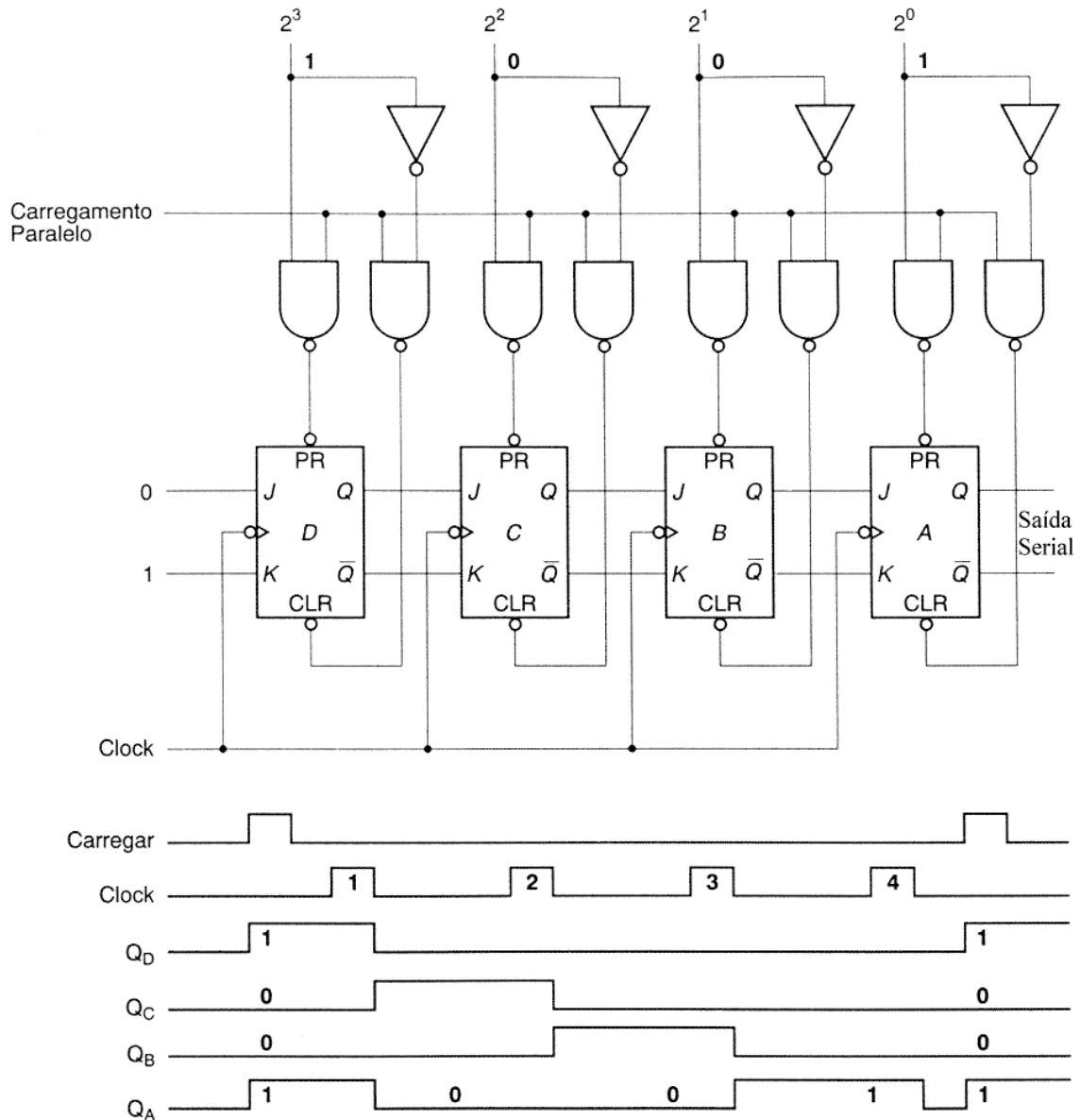


Figura 3: Registrador de deslocamento de 4 bits com entrada paralela e saída serial. Note que após a borda de descida do primeiro *clock* o nível lógico 0 na entrada *J* do flip-flop A é transferido para sua saída, “seguindo” a palavra binária de entrada que é deslocada para a direita neste registrador.

Exemplo 2: Determine a saída do *shift register* da Figura 3 quando a palavra binária representativa do número 10 é aplicada na entrada paralela.

Solução:

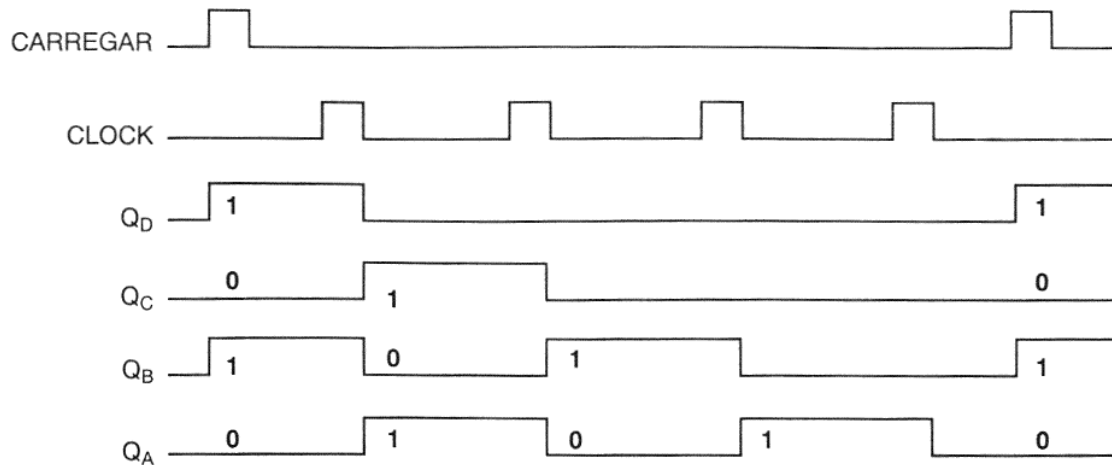


Figura 4: Formas de onda para um registrador de deslocamento de 4 bits com entrada paralela e saída serial com a palavra binária representativa do número 10 aplicada na entrada paralela.

- A comunicação entre dois microcomputadores através de suas portas seriais é baseada em registradores de deslocamento. Em cada porta serial, um registrador de deslocamento transforma os dados do barramento interno do microcomputador (que estão em formato paralelo) no formato serial adequado para a transmissão através do cabo que interliga os dois computadores.

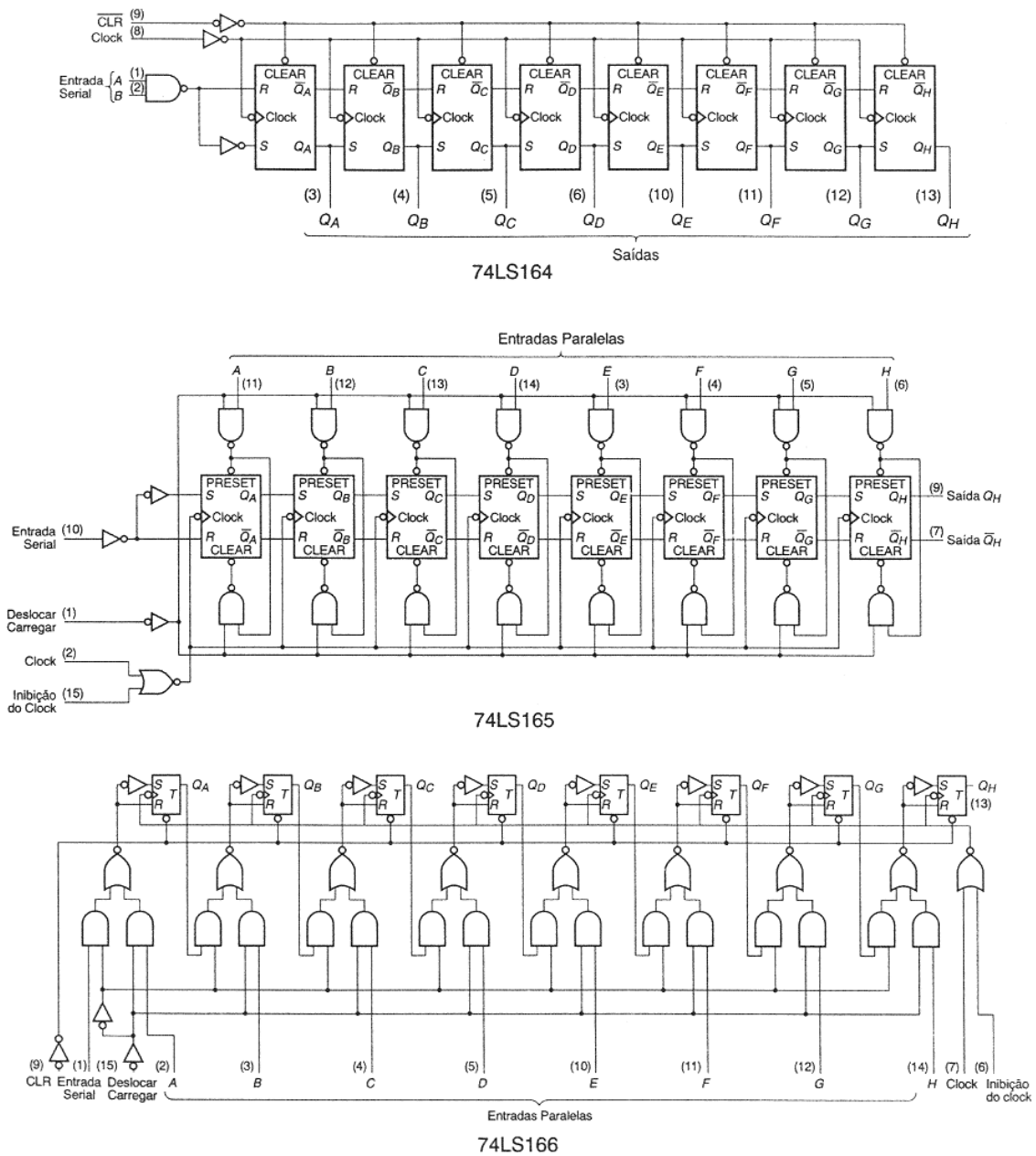


Figura 5a: Registradores de deslocamento comercialmente disponíveis na família TTL.

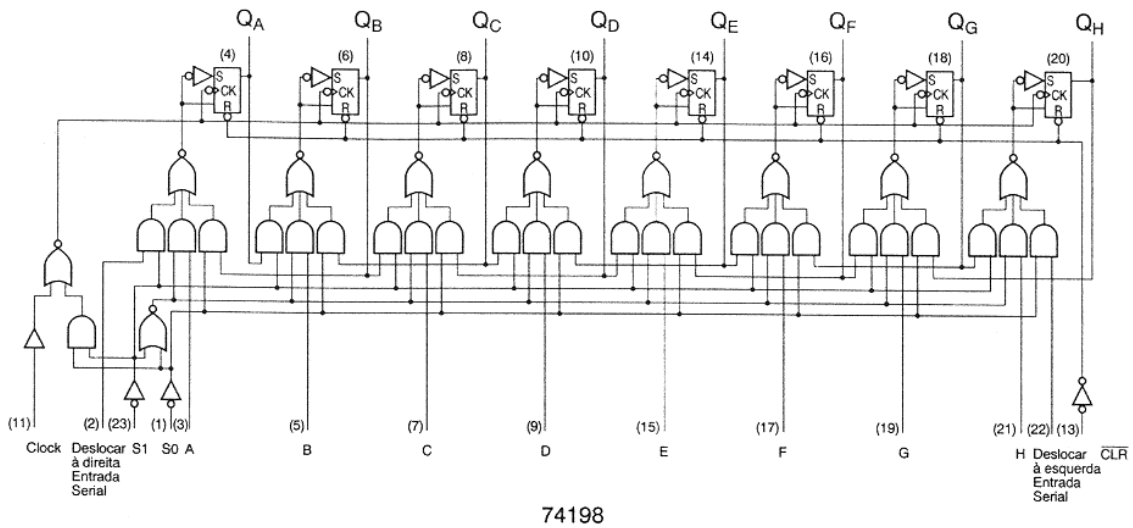


Figura 5b: Registradores de deslocamento comercialmente disponíveis na família TTL.

3 Contadores Assíncronos (*ripple counter*)

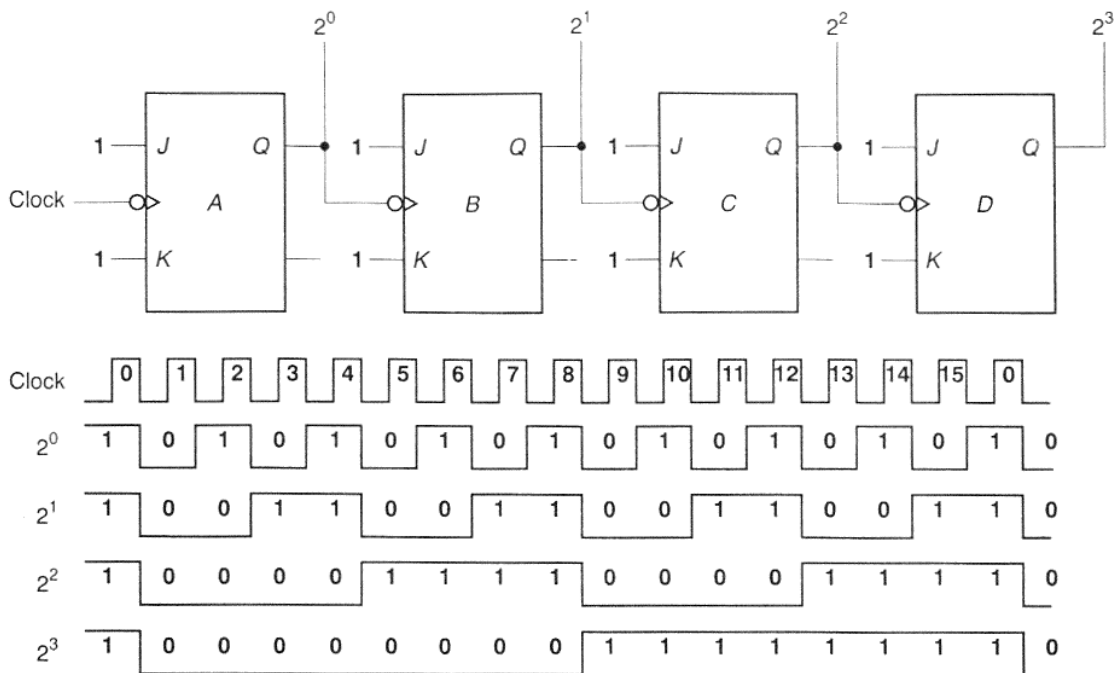


Figura 6: Contador assíncrono de 4 bits (conta até $2^4 = 16$). Note que somente o *flip-flop* A recebe o sinal de *clock*. Todos os demais recebem o *clock* da saída *Q* do *flip-flop* à esquerda, e, por isso, o contador é denominado assíncrono. Toda vez que ocorre a descida do *clock* na entrada de um *flip-flop*, o *flip-flop* muda de estado (*toggle*) e incrementa a contagem do dígito binário correspondente à posição do *flip-flop* no contador.

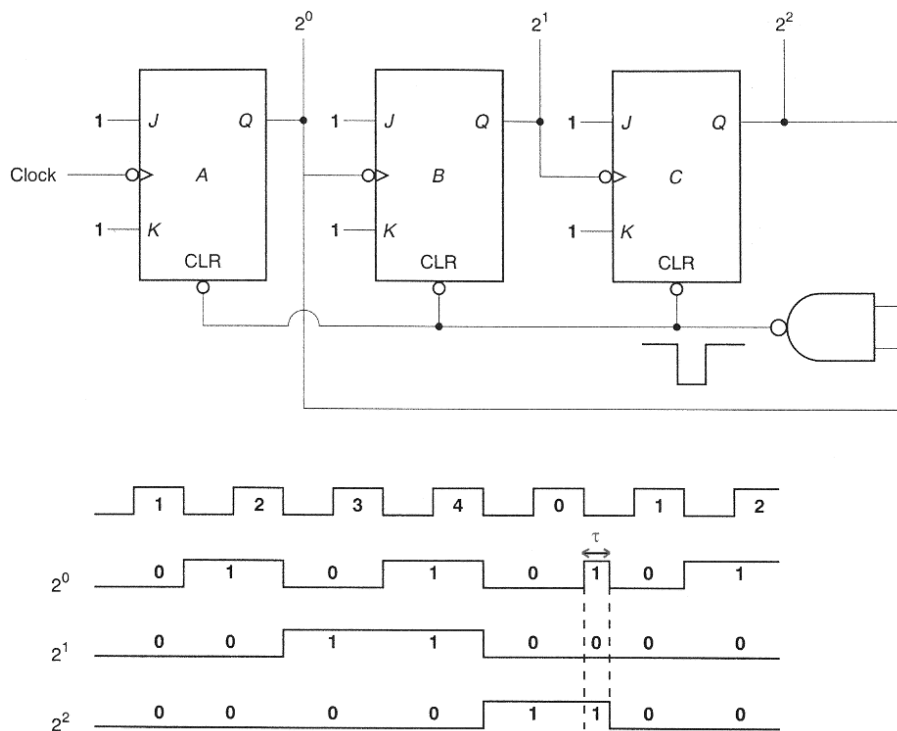


Figura 7: Contador assíncrono para contagem até 5. O número $5_{10} = 101_2$ é decodificado pela porta NAND, efetuando um CLEAR em todos os *flip-flops*, reiniciando a contagem. Note que o número $7_{10} = 111_2$ também é detectado pela porta NAND, mas a contagem é reiniciada antes de atingir este valor. Note que a duração τ em que o número 101 está presente nas saídas Q é muito pequeno (um *spike* da ordem do tempo de propagação de uma porta), e isto pode gerar problema se o sinal nas saídas for utilizado como *clock* para algum outro circuito.

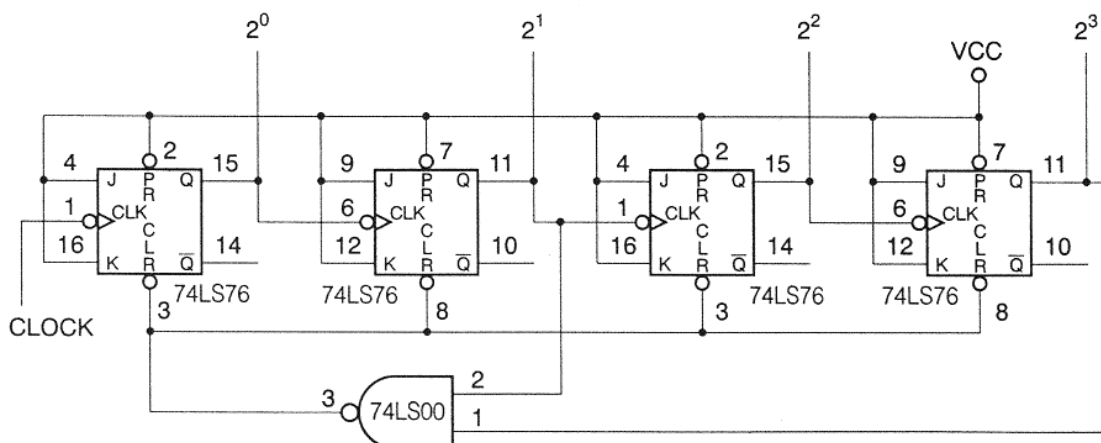


Figura 8: Contador assíncrono para contagem até 10. O número $10_{10} = 1010_2$ é decodificado pela porta NAND, efetuando um CLEAR em todos os *flip-flops*, reiniciando a contagem. Note que os números, $11_{10} = 1011_2$, $14_{10} = 1110_2$ e $15_{10} = 1111_2$ também são detectados pela porta NAND, mas a contagem é reiniciada antes de atingir este valor.

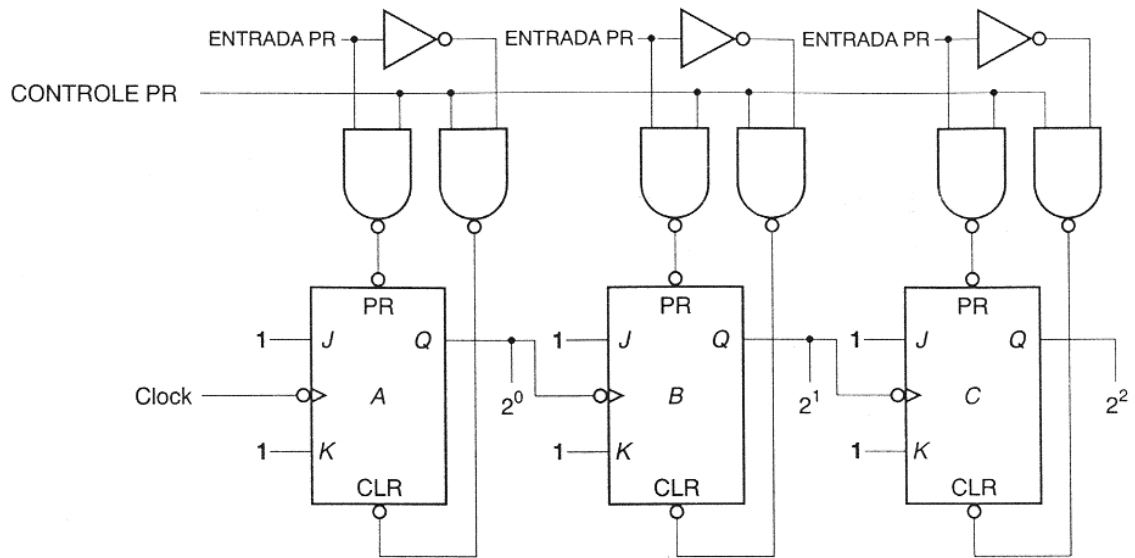


Figura 9: Contador assíncrono presetável de 4 bits (conta até $2^3 = 8$). As entradas PR recebem a palavra binária na qual se deseja inicializar a contagem. Durante a inicialização o CONTROLE_PR deve receber o nível lógico 1.

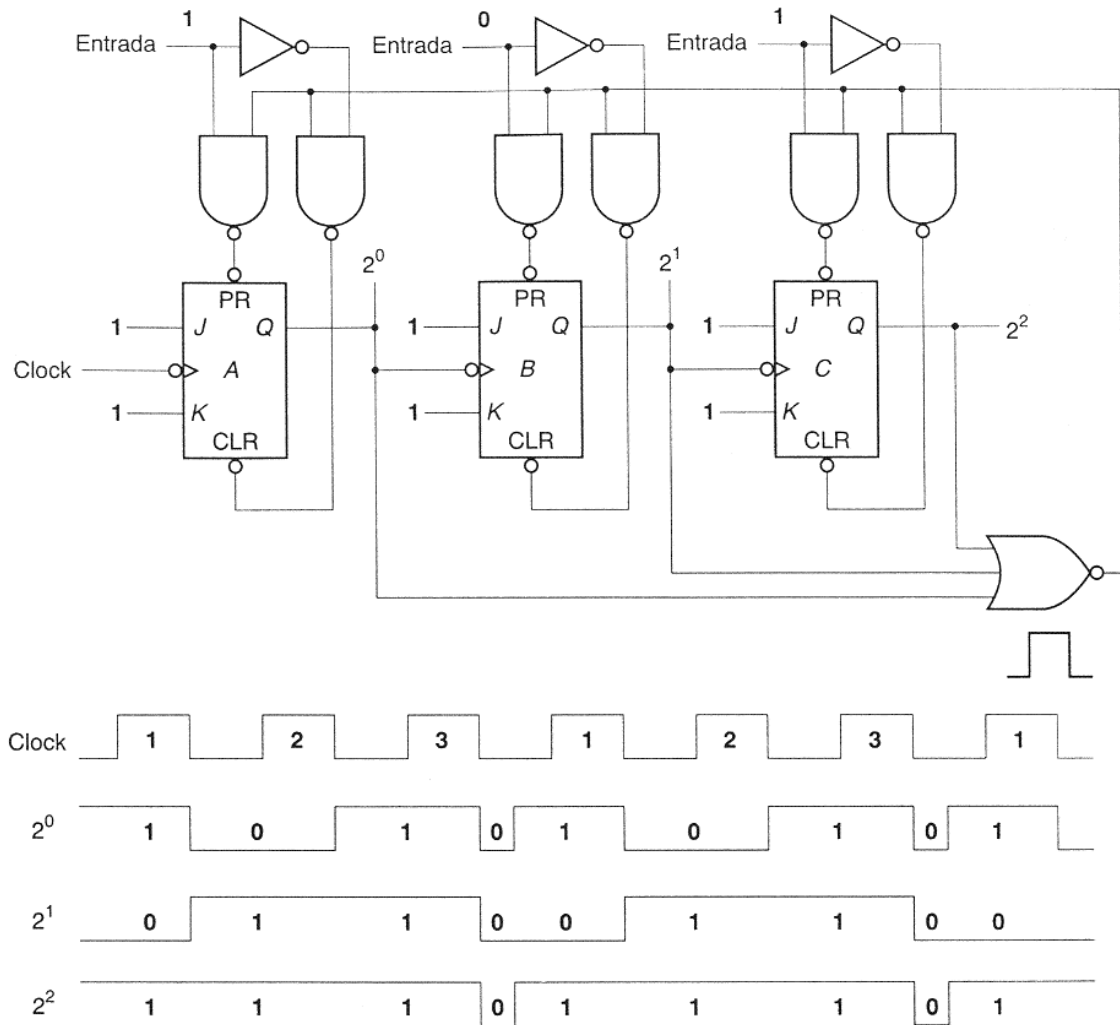


Figura 10: Contador assíncrono presetável para contagem em módulo 3. A seqüência gerada é 5,6,7,5...

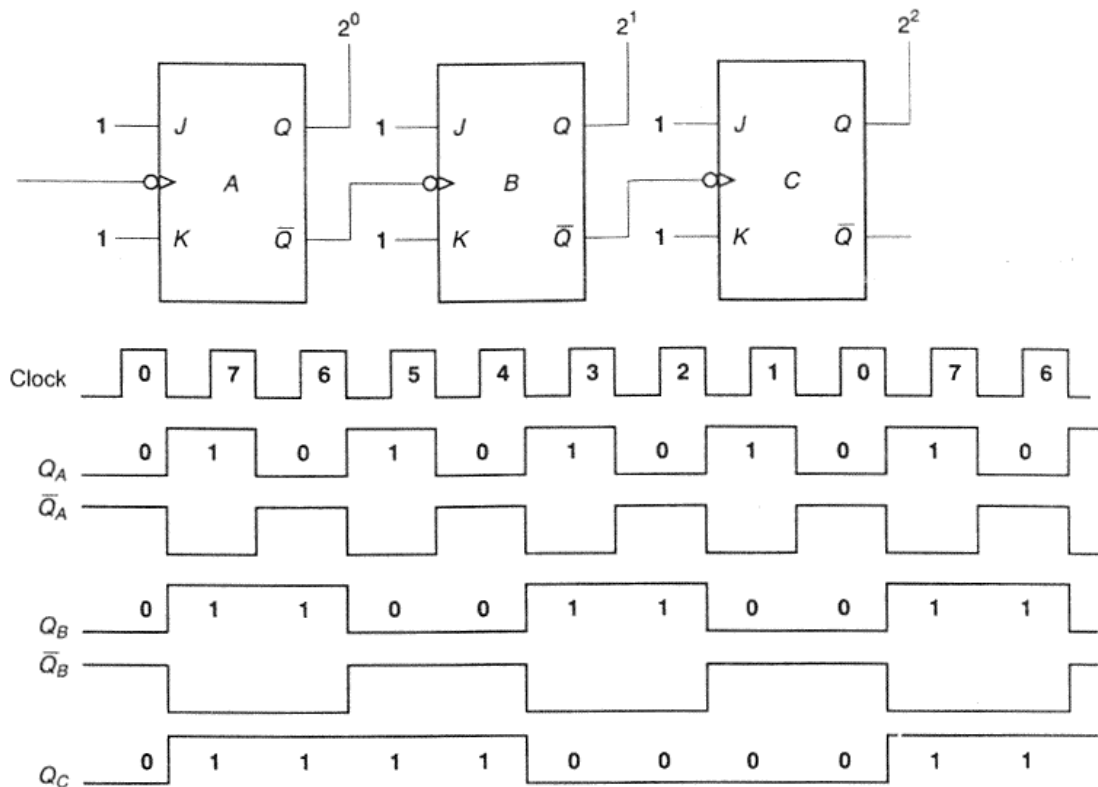


Figura 11: Contador assíncrono decrescente. Para contagem decrescente liga-se a entrada de *clock* de cada *flip-flop* (exceto o 1°) à saída a \bar{Q} do *flip-flop* à esquerda.

4 Contadores Síncronos

- Em um contador síncrono, todos os *flip-flops* compartilham o mesmo sinal de *clock* e, portanto, todos os *flip-flops* ficam habilitados no mesmo instante para que ocorra a troca de estado.
- A vantagem do contador síncrono sobre o contador assíncrono reside na inexistência de atraso de propagação entre os pulsos *clock* dos *flip-flops*, evitando *spikes* e *glitches* (pulsos de curta duração resultantes de atrasos de propagação distintos).

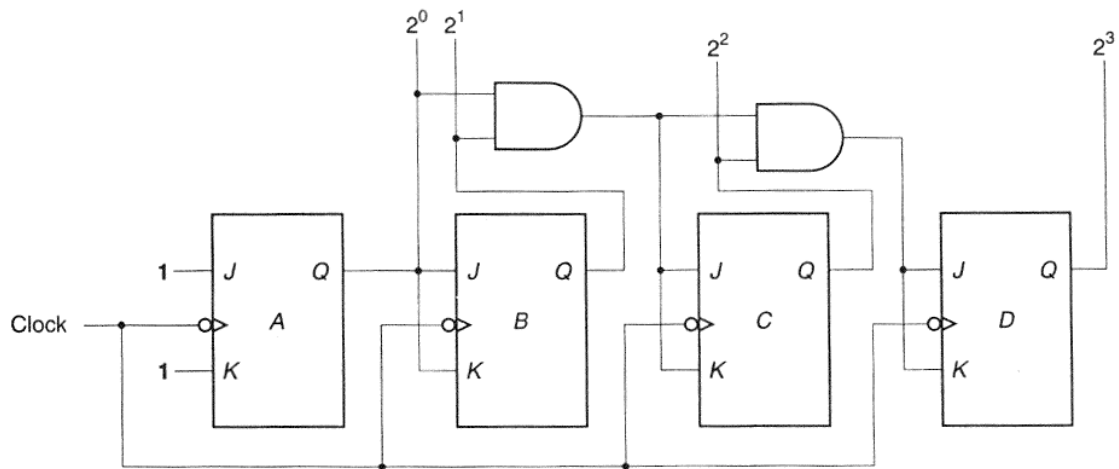


Figura 12: Contador assíncrono de 4 bits (conta até $2^4 = 16$). Note que todos os *flip-flops* recebem o mesmo sinal de *clock*. As portas AND determinam a seqüência numérica gerada, que, no caso, é a seqüência de contagem 0000_2 , 0001_2 , ..., 1111_2 . Note que qualquer seqüência numérica pode ser gerada com um contador síncrono.

- A melhor maneira de compreender a operação de um contador síncrono é projetando o contador para uma seqüência numérica desejada.
- Os exemplos de projeto de contadores que seguem serão baseados em *flip-flops* JK disparados pela borda de descida do *clock*, cuja Tabela-Verdade é:

(a)

\overline{PR}	\overline{CL}	J	K	C	Q	\overline{Q}
0	1	X	X	X	1	0
1	0	X	X	X	0	1
0	0	X	X	X	1	1
1	1	0	1	↯	0	1
1	1	1	0	↯	1	0
1	1	0	0	X	Q	Q
1	1	1	1	↯	Toggle	

Estado Inválido

(b)

Antes do Clock			Após o Clock			Antes do Clock	
Q			Q			J	K
0	0	0	0	0	0	X	
0	0	1	0	1	1	X	
1	0	0	1	0	X	1	
1	0	1	1	1	X	0	

Figura 13: (a) Tabela-Verdade de um *flip-flop* JK disparado pela borda de descida do *clock*. (b) Tabela de Transição de Estado do *flip-flop* JK.

Exemplo 3: Projete um contador síncrono que gere a seqüência numérica CBA = 000, 001, 010, 011, 100, 000 ...

Solução:

● Primeiramente, vamos construir a Tabela de Estados do contador, com base na seqüência desejada e na Tabela de Transição de Estado do *flip-flop* JK (Figura 13b). Sejam C , B e A as saídas Q dos *flip-flops* responsáveis respectivamente pela representação dos bits 2^2 (MSB), 2^1 e 2^0 (LSB) da seqüência CBA:

Tabela de Estados do Contador											
Q antes do <i>clock</i>			Q depois do <i>clock</i>			C		B		A	
C	B	A	C	B	A	J	K	J	K	J	K
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	0	0	0	X	1	0	X	0	X

- A seguir, determina-se os Mapas de Karnaugh das variáveis J e K dos *flip-flops* C , B e A em função dos valores das variáveis C , B e A na coluna “ Q antes do *clock*” na Tabela de Estados do Contador:

		BA			
		00	01	11	10
C	0	0	0	1	0
	1	X	X	X	X

Figura 14: $J_C = AB$.

		BA			
		00	01	11	10
C	0	X	X	X	X
	1	1	X	X	X

Figura 15: $K_C = 1$.

		BA			
		00	01	11	10
C	0	0	1	X	X
	1	0	X	X	X

Figura 16: $J_B = A$.

		BA			
		00	01	11	10
C	0	X	X	1	0
	1	X	X	X	X

Figura 17: $K_B = A$.

C		BA			
		00	01	11	10
0		1	X	X	1
1		0	X	X	X

Figura 18: $J_A = \bar{C}$.

C		BA			
		00	01	11	10
0		X	1	1	X
1		X	X	X	X

Figura 19: $K_A = 1$.

- Portanto, das figuras 14 a 19, o circuito resultante é:

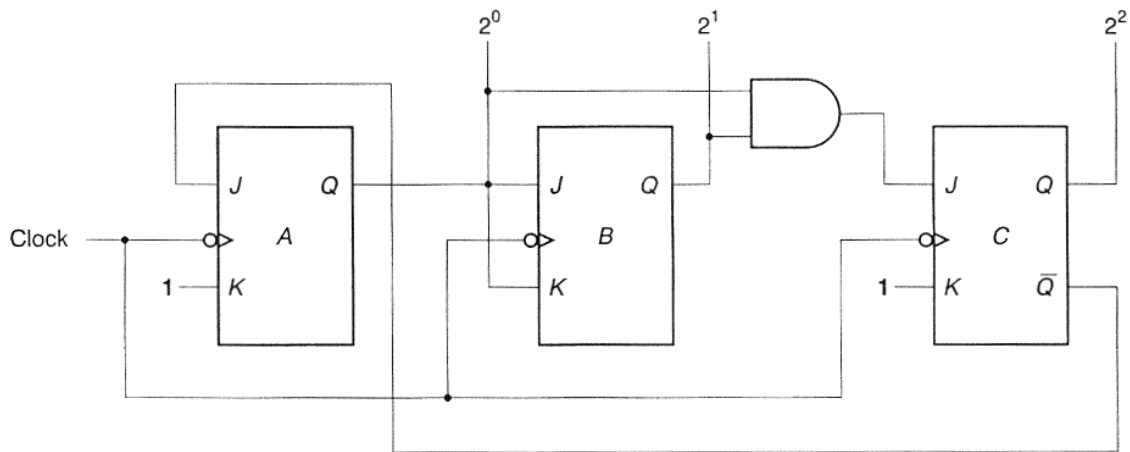


Figura 20: Contador síncrono gerador da seqüência numérica 000, 001, 010, 011, 100, 000 ...

Exemplo 4: Projete um contador síncrono que gere a seqüência numérica CBA = 000, 001, 010, 011, 100, 101, 000 ...

Solução:

- Primeiramente, obtém-se a Tabela de Estados do contador, com base na seqüência desejada e na Tabela de Transição de Estado do *flip-flop* JK (Figura 13b). Sejam **C**, **B** e **A** as saídas Q dos *flip-flops* responsáveis respectivamente pela representação dos bits 2^2 (MSB), 2^1 e 2^0 (LSB) da seqüência CBA:

Tabela de Estados do Contador											
Q antes do <i>clock</i>			Q depois do <i>clock</i>			C		B		A	
C	B	A	C	B	A	J	K	J	K	J	K
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	0	0	X	1	X	X	1
0	1	0	0	1	1	0	X	X	0	1	X
0	1	1	1	0	0	1	X	X	1	X	1
1	0	0	1	0	1	X	0	0	X	1	X
1	0	1	0	0	0	X	1	0	X	X	1

- A seguir, determina-se os Mapas de Karnaugh das variáveis J e K dos *flip-flops* **C**, **B** e **A** em função dos valores das variáveis **C**, **B** e **A** na coluna “ Q antes do *clock*” na Tabela de Estados do Contador:

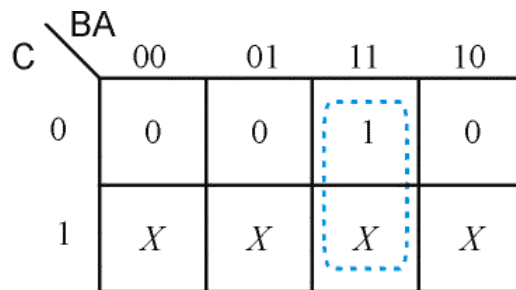


Figura 21: $J_C = AB$.

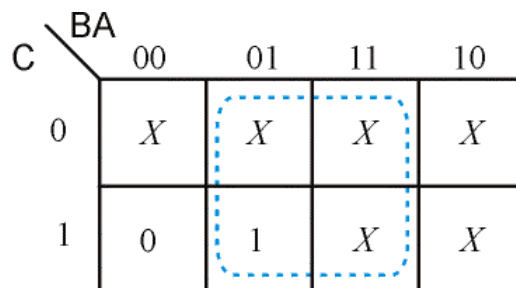


Figura 22: $K_C = A$.

		BA			
		00	01	11	10
C	0	0	1	X	X
	1	0	0	X	X

Figura 23: $J_B = A\bar{C}$.

		BA			
		00	01	11	10
C	0	X	X	1	0
	1	X	X	X	X

Figura 24: $K_B = A$.

		BA			
		00	01	11	10
C	0	1	X	X	1
	1	1	X	X	X

Figura 25: $J_A = 1$.

		BA			
		00	01	11	10
C	0	X	1	1	X
	1	X	1	X	X

Figura 26: $K_A = 1$.

- Portanto, das figuras 21 a 26, o circuito resultante é:

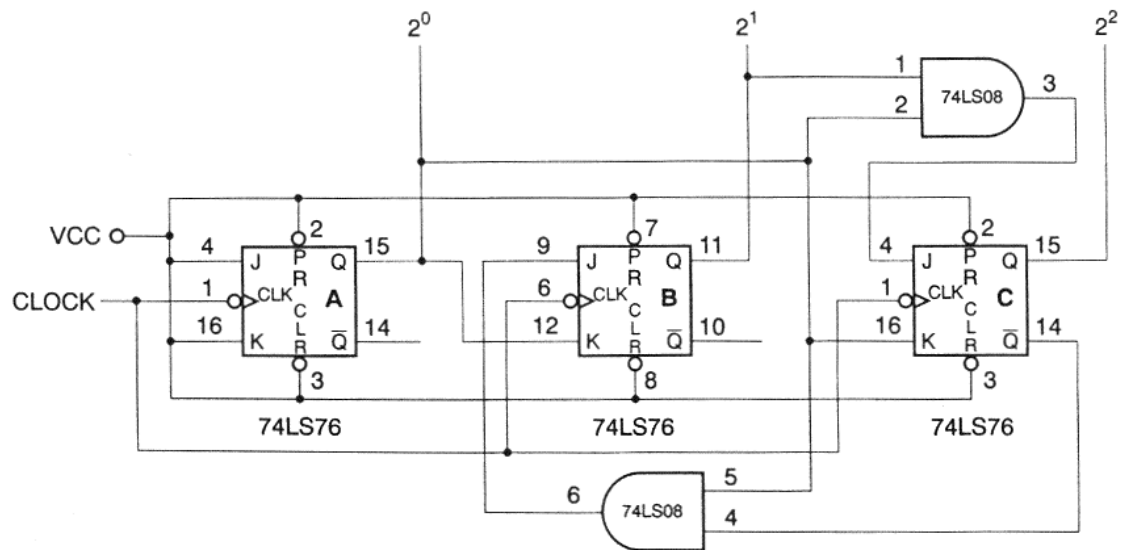


Figura 27: Contador síncrono gerador da seqüência numérica 000, 001, 010, 011, 100, 101, 000 ...

- A Figura 28 mostra contadores comercialmente disponíveis na família TTL.

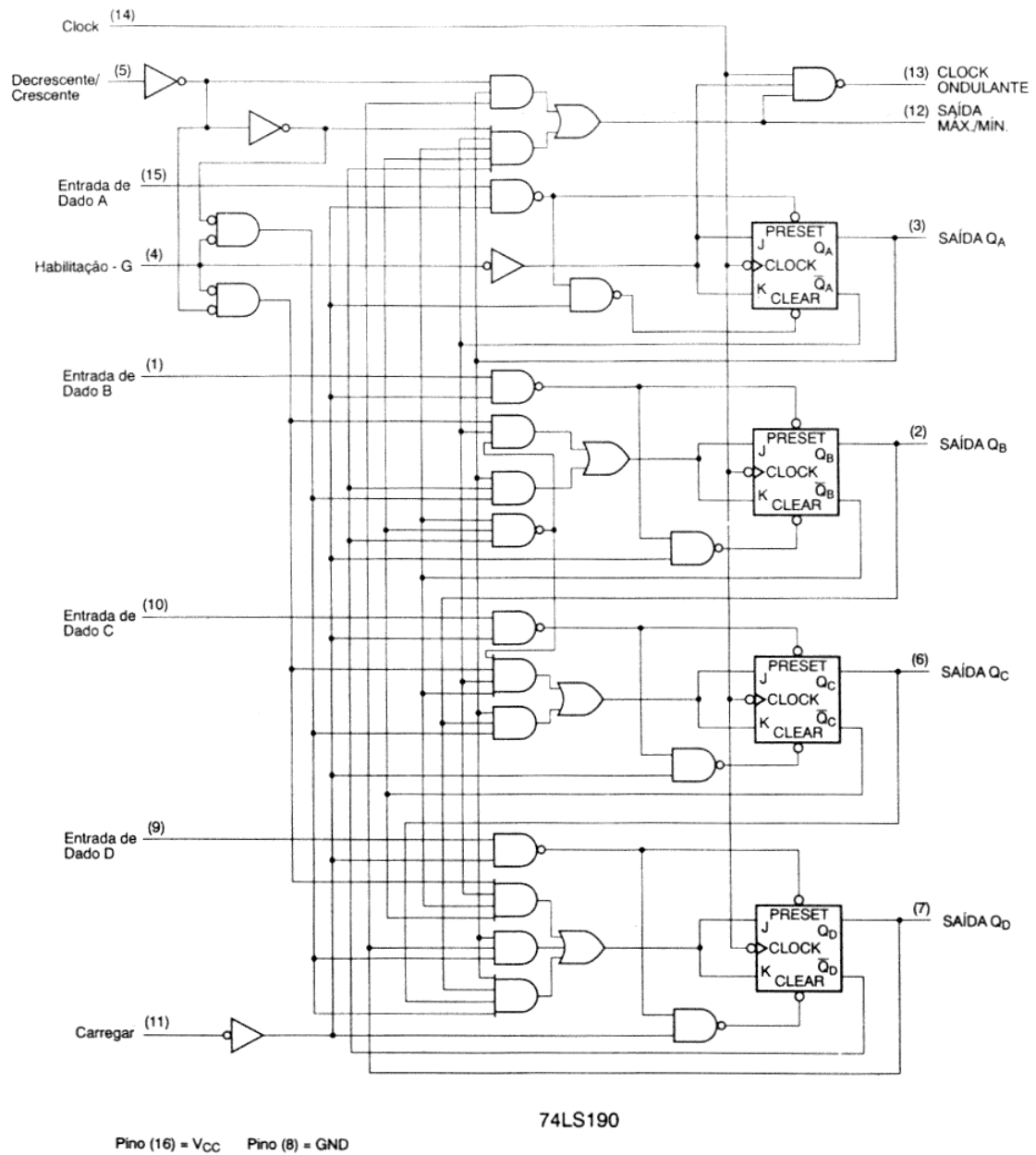


Figura 28a: Alguns contadores síncronos comercialmente disponíveis em CIs da família TTL.

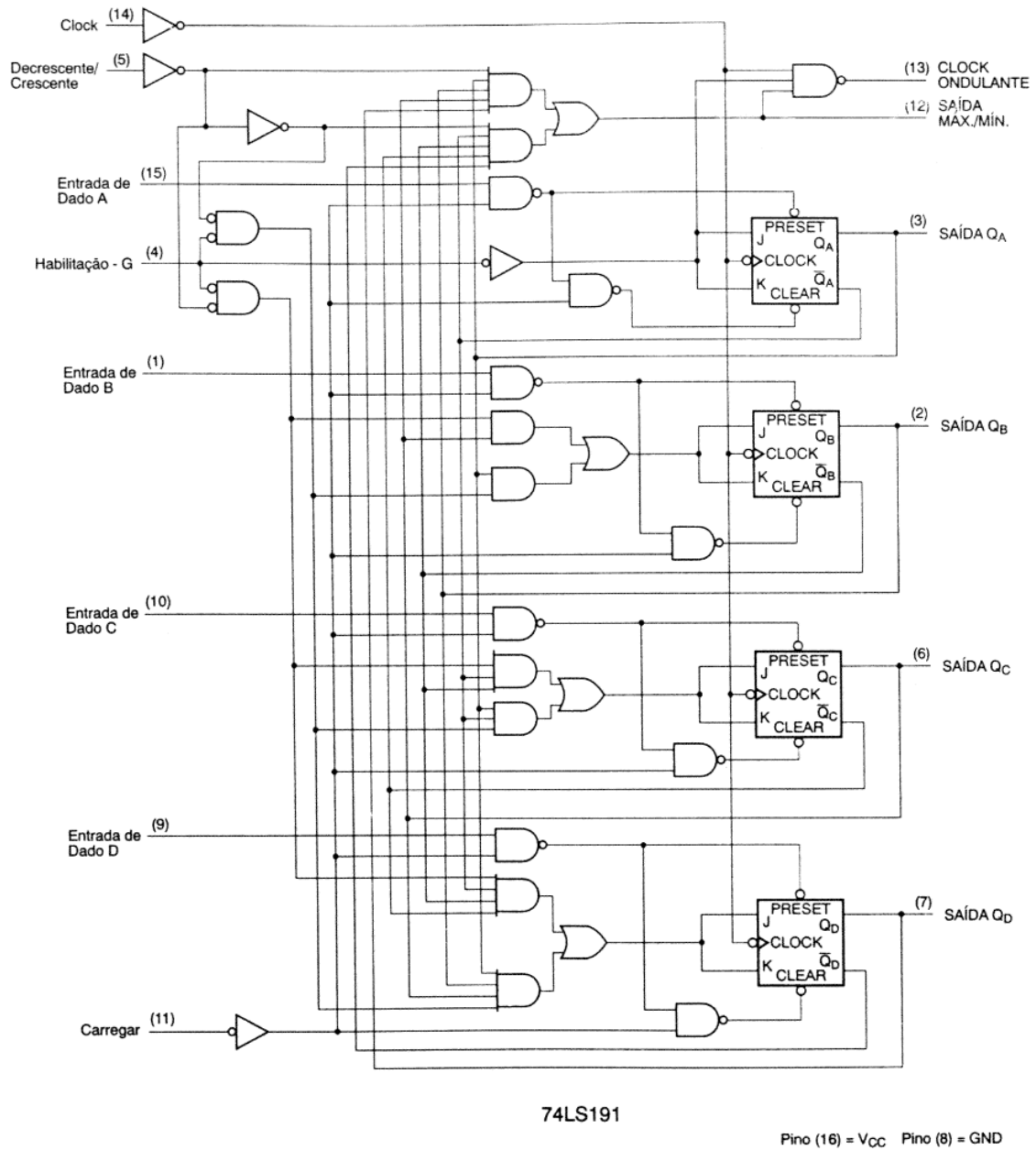


Figura 28b: Alguns contadores síncronos comercialmente disponíveis em CIs da família TTL.