

Curso: Engenharia de Computação

Sistemas Digitais

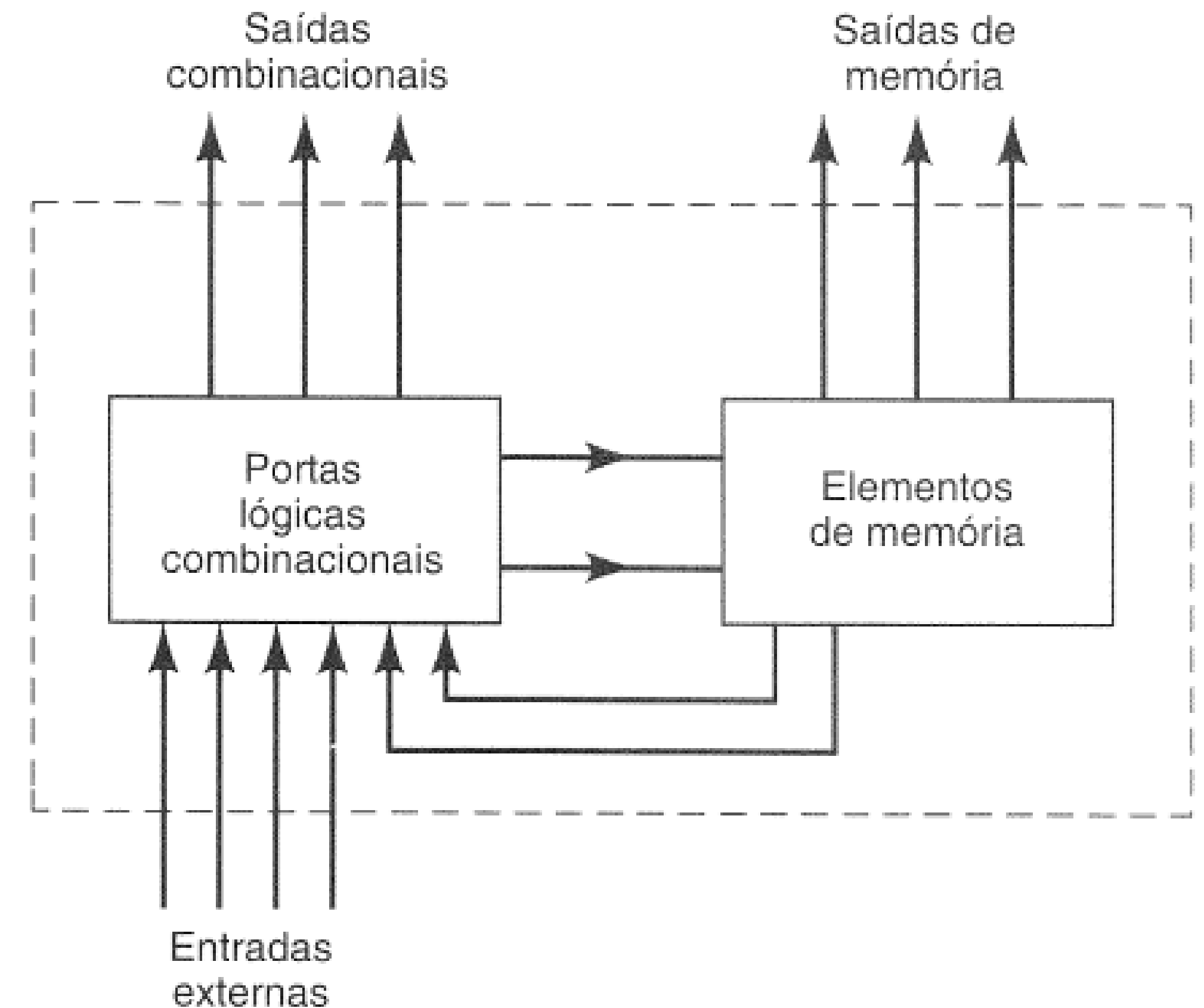
Prof. Clayton J A Silva, MSc
clayton.silva@professores.ibmec.edu.br



Circuitos sequenciais

Circuitos sequenciais

- São aqueles cuja saída em um determinado instante depende das **entradas no instante** considerado e das entradas e saídas em **instantes anteriores**.
- A resposta atual depende da **memória**, ou seja, dos dados do passado.
- Os sistemas digitais utilizam tanto circuitos combinacionais quanto circuitos de memória.



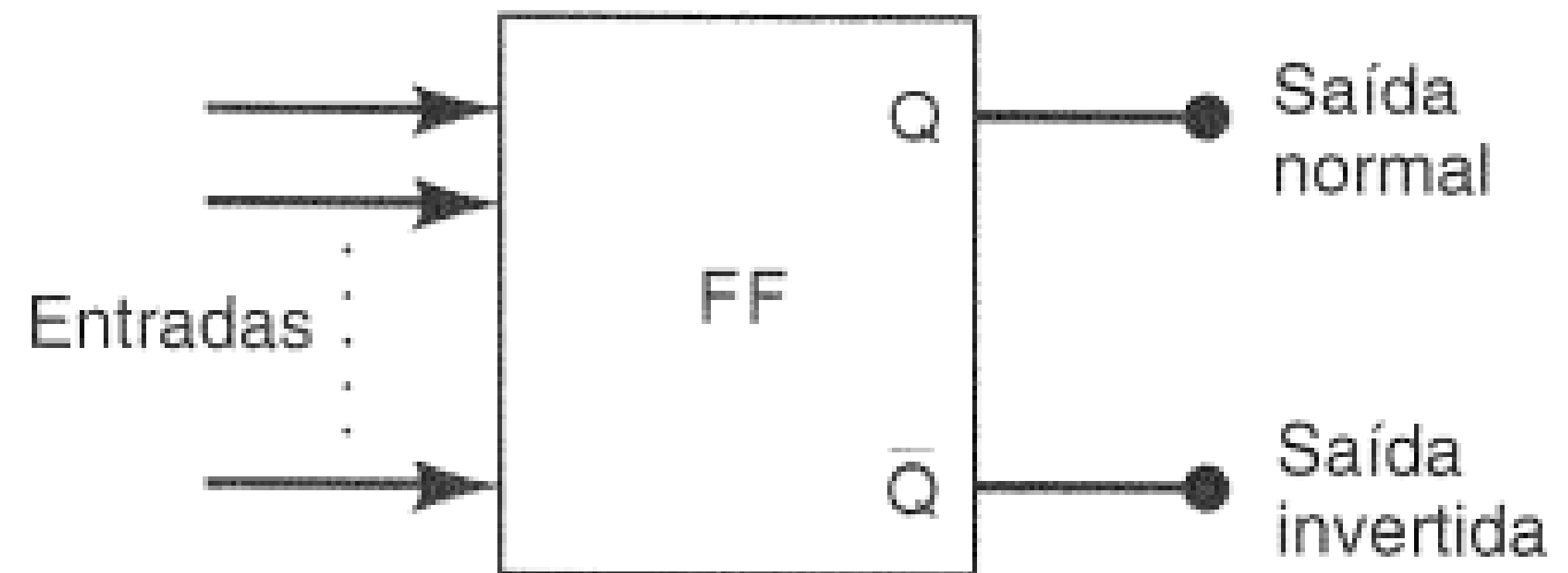
Flip Flop

- Embora uma porta lógica isolada não tenha capacidade de armazenamento, várias portas conectadas de forma adequada podem proporcionar memória.
- Flip Flop genérico – também chamado de *latch*

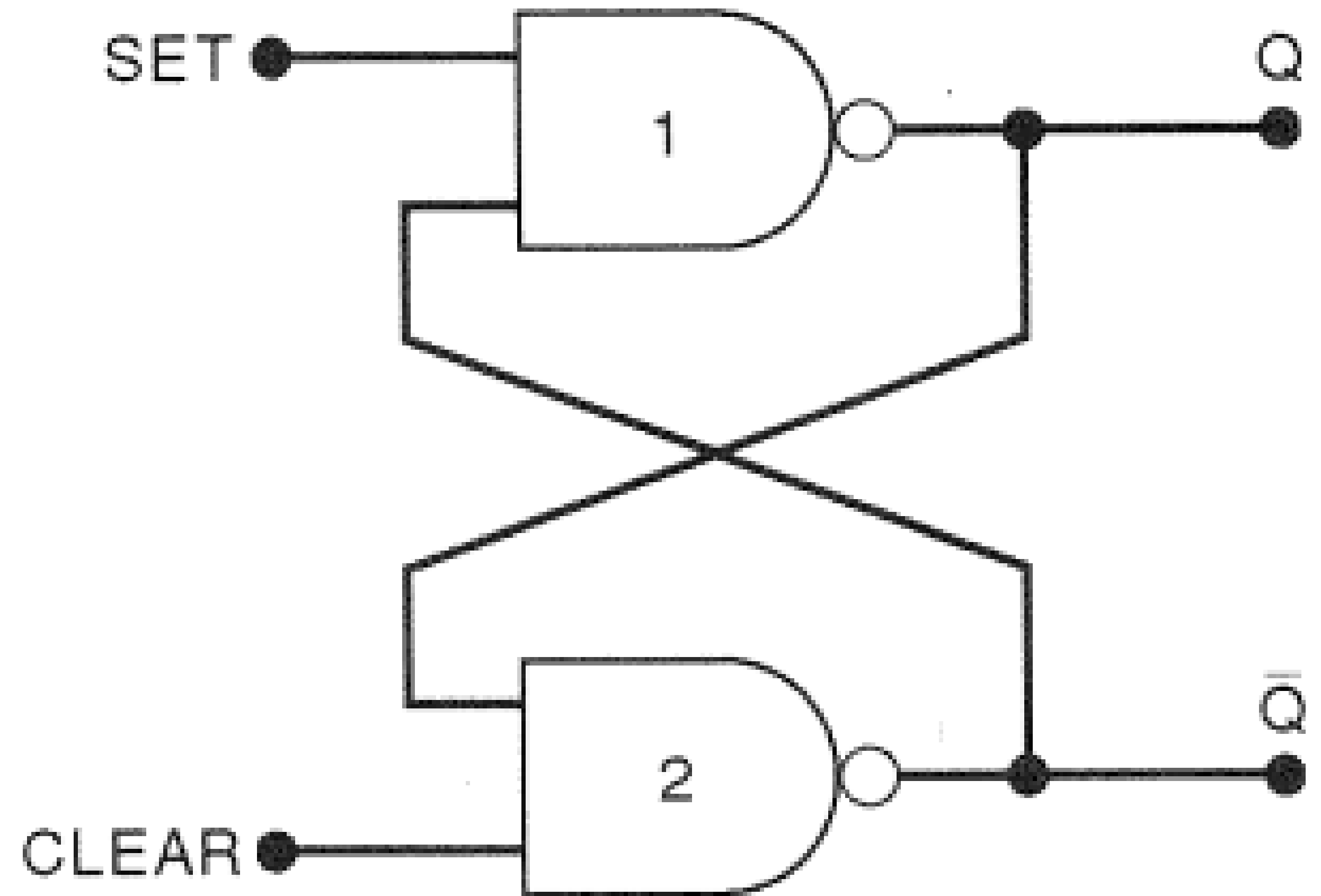
ESTADOS DE SAÍDA

$Q = 1, \bar{Q} = 0$: Estado SET

$Q = 0, \bar{Q} = 1$: Estado CLEAR ou RESET



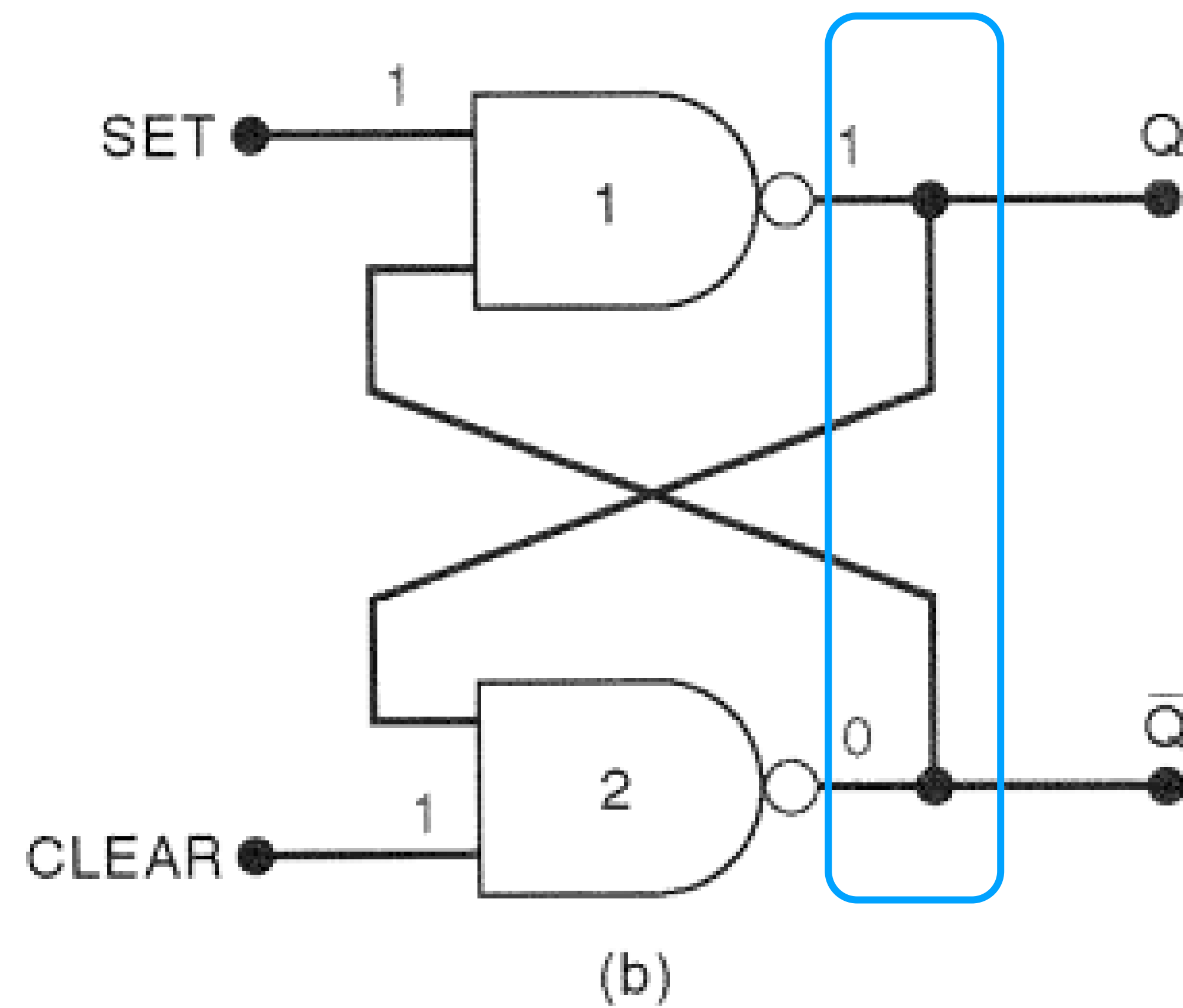
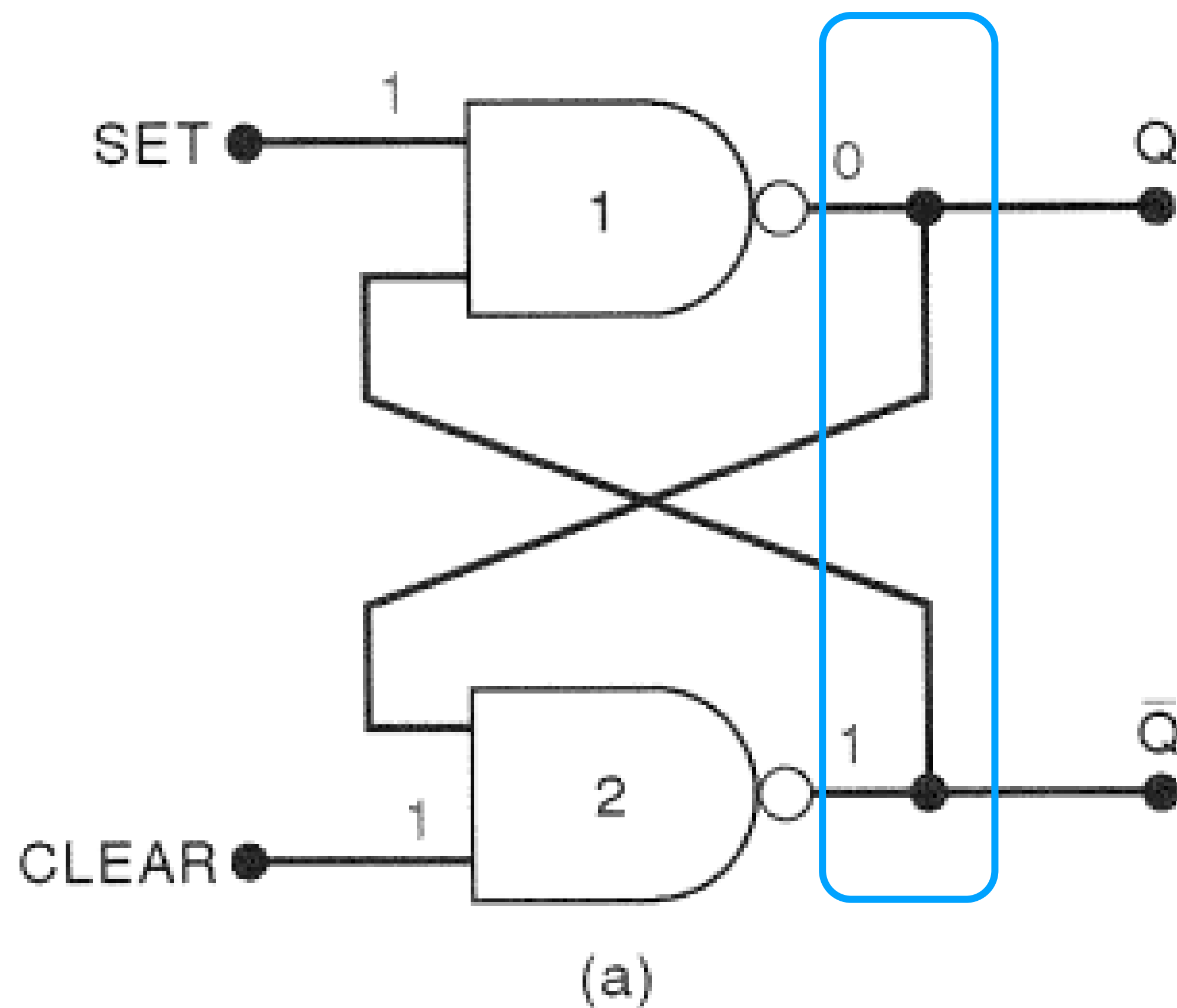
Latch com
portas
NAND



Latch com portas NAND

Estado de repouso

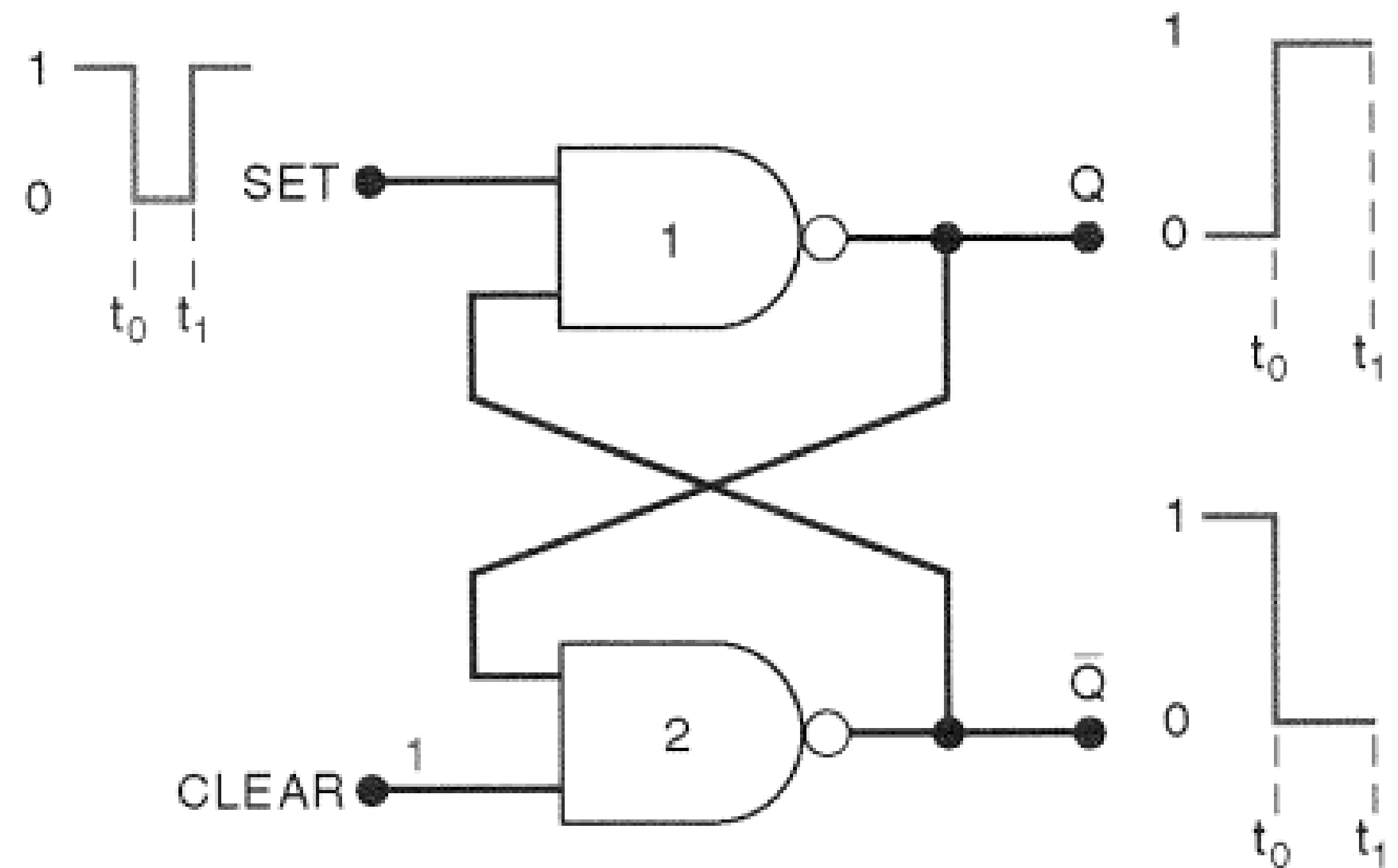
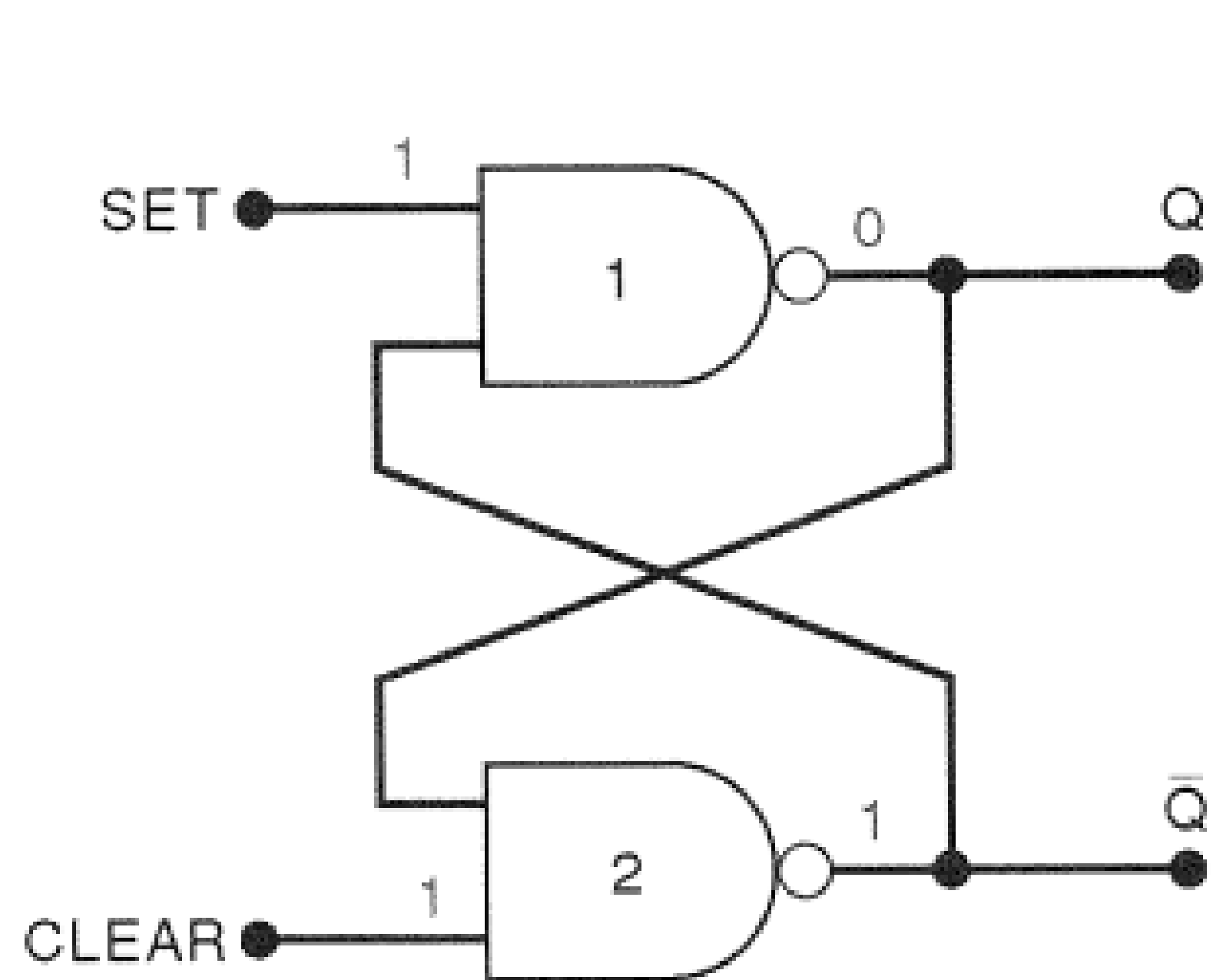
SET = CLEAR = 1



Latch com portas NAND

Setar o latch

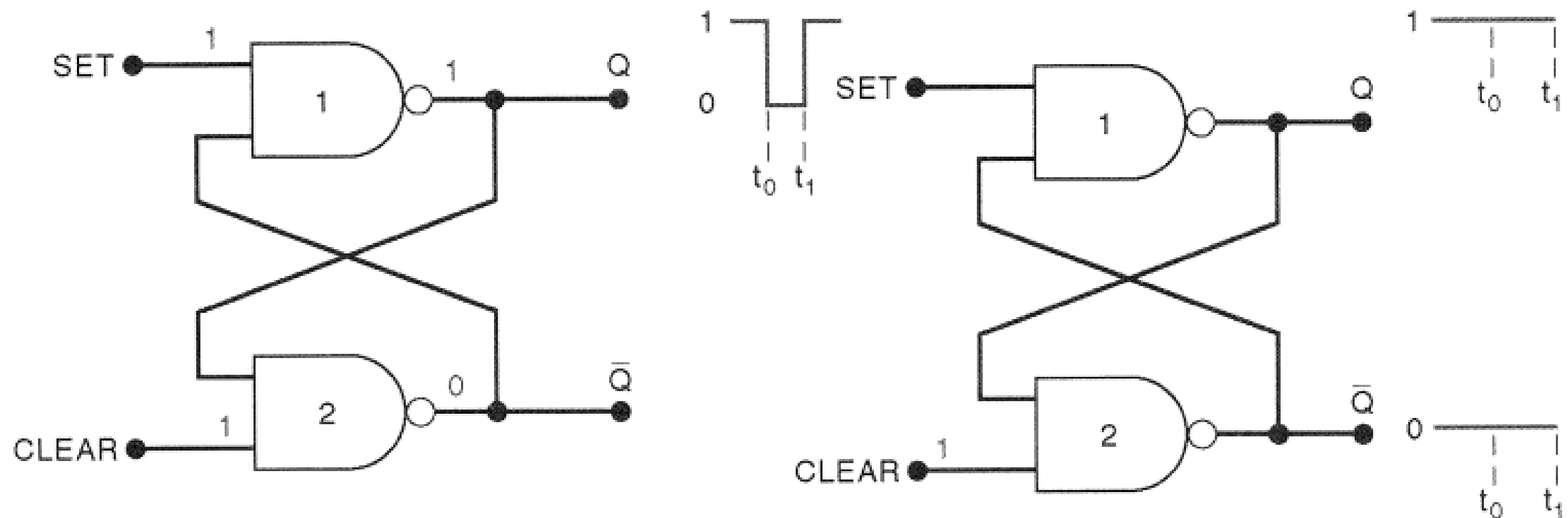
SET = 1 \rightarrow 0



Latch com portas NAND

Setar o latch

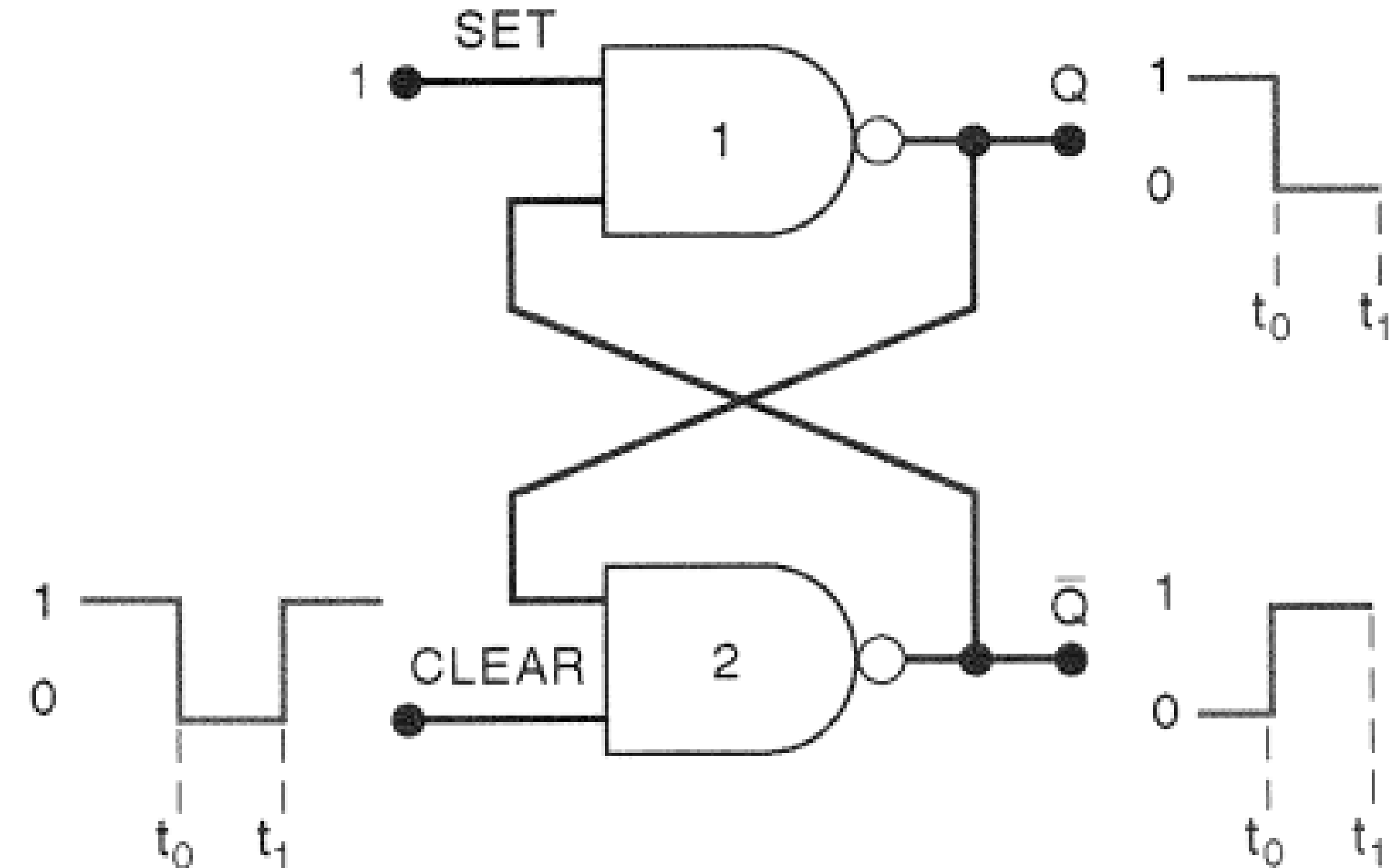
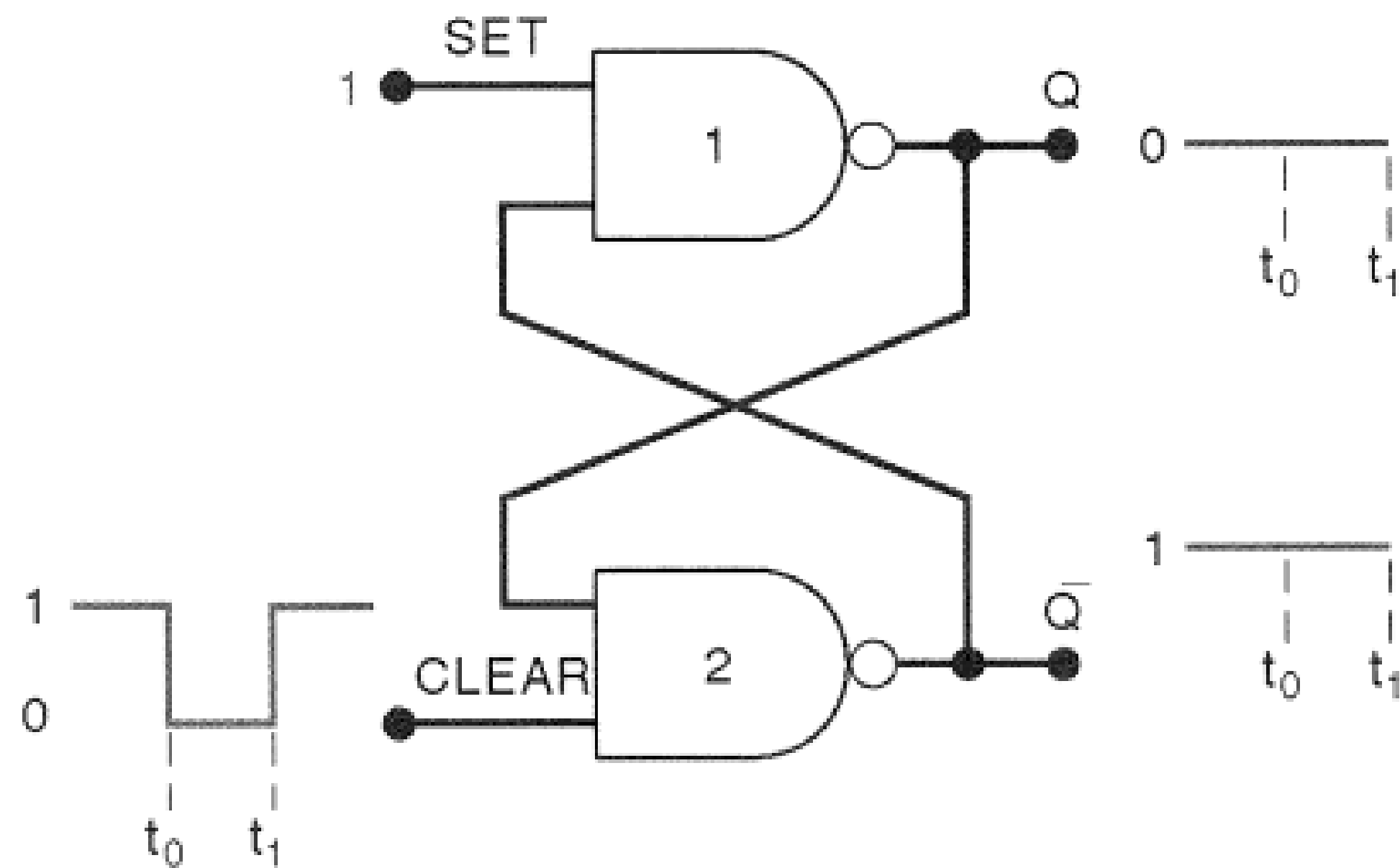
SET = 1 \rightarrow 0



Latch com portas NAND

Resetar o latch

RESET = 1 \rightarrow 0



Latch com portas NAND

1. **SET = CLEAR = 1.** Esta condição é o estado normal de repouso e não tem nenhum efeito sobre o estado de saída. As saídas Q e \bar{Q} permanecerão com os mesmos valores que estavam antes desta condição de entrada.
2. **SET = 0, CLEAR = 1.** Isto sempre faz a saída ir para o estado no qual $Q = 1$, onde permanecerá mesmo após SET retornar para ALTO. Isto é denominado setar o latch.
3. **SET = 1, CLEAR = 0.** Isto sempre produz o estado $Q = 0$, onde a saída permanecerá mesmo após CLEAR retornar para ALTO. Isto é denominado limpar ou ressetar o latch.
4. **SET = CLEAR = 0.** Esta condição tenta setar e limpar o latch ao mesmo tempo e pode produzir resultados ambíguos. Não deve ser usada.

Latch com portas NAND

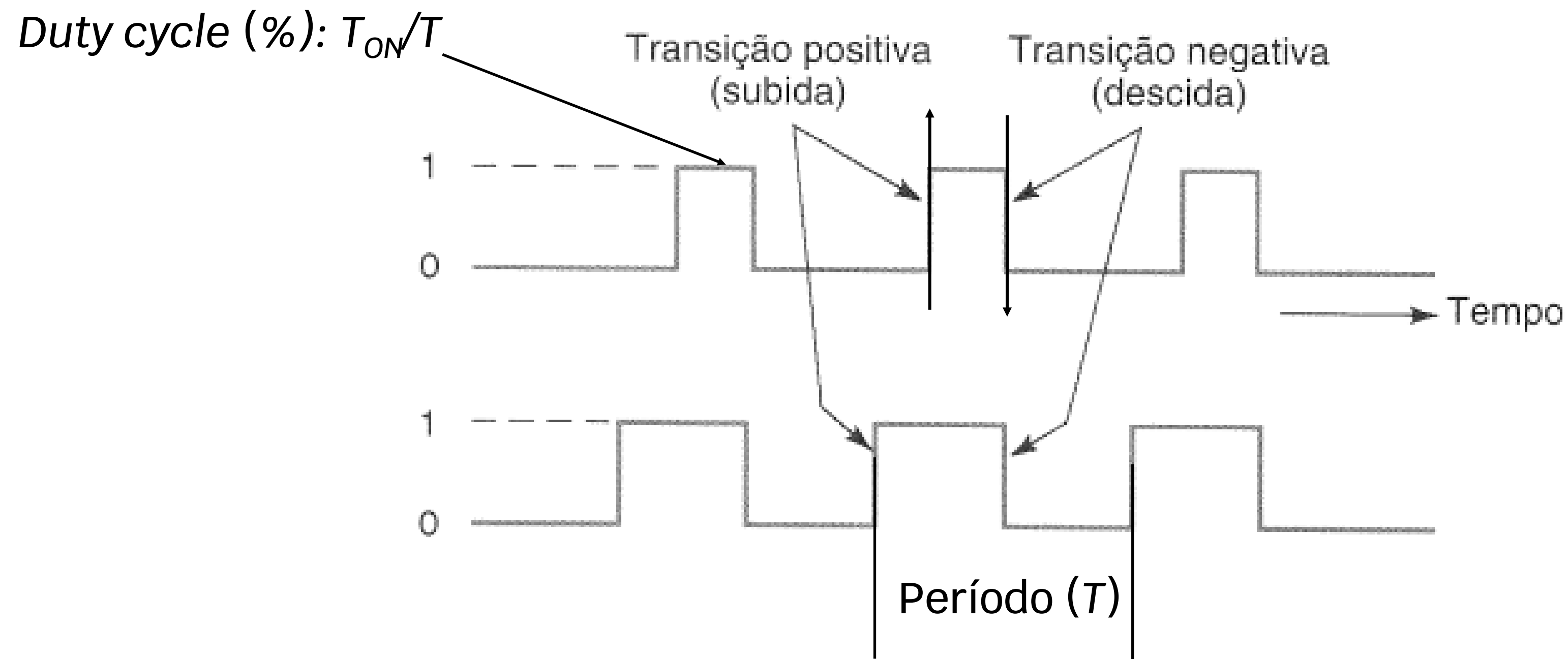
Set	Clear	Saída
1	1	Não muda
0	1	$Q = 1$
1	0	$Q = 0$
0	0	Inválido*

* produz $Q = \bar{Q} = 1$

1. **SET = CLEAR = 1.** Esta condição é o estado normal de repouso e não tem nenhum efeito sobre o estado de saída. As saídas Q e \bar{Q} permanecerão com os mesmos valores que estavam antes desta condição de entrada.
2. **SET = 0, CLEAR = 1.** Isto sempre faz a saída ir para o estado no qual $Q = 1$, onde permanecerá mesmo após SET retornar para ALTO. Isto é denominado setar o latch.
3. **SET = 1, CLEAR = 0.** Isto sempre produz o estado $Q = 0$, onde a saída permanecerá mesmo após CLEAR retornar para ALTO. Isto é denominado limpar ou ressetar o latch.
4. **SET = CLEAR = 0.** Esta condição tenta setar e limpar o latch ao mesmo tempo e pode produzir resultados ambíguos. Não deve ser usada.

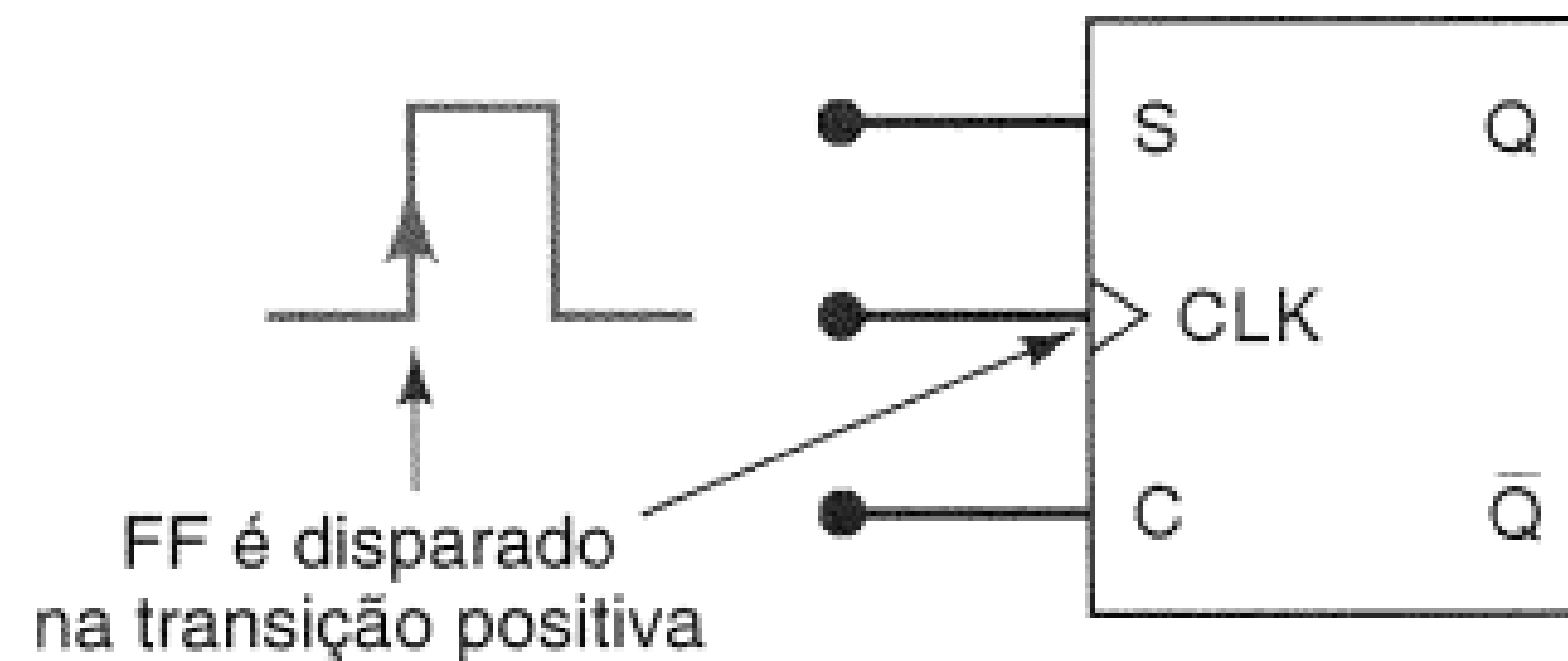
Modos síncrono e assíncrono

- Circuitos assíncronos mudam de estados por um sinal que independe de um relógio de referência.
- Circuitos síncronos: o sinal de *clock*



Flip Flop com *clock*

Flip Flop SC (*SET-CLEAR*) ou SR (*SET-RESET*)

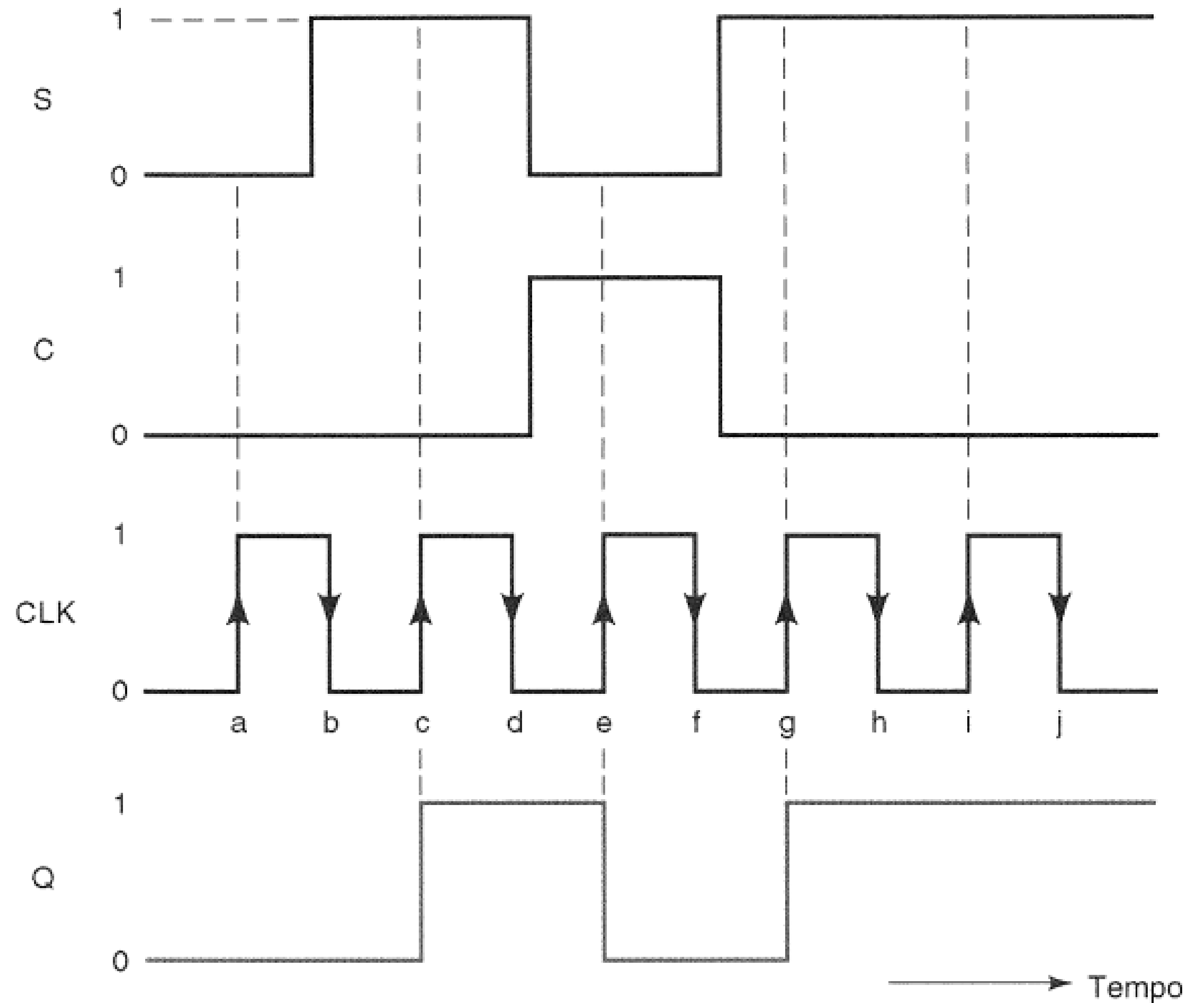


Entradas			Saída	
S	C	CLK		Q
0	0	↑		Q_0 (não muda)
1	0	↑		1
0	1	↑		0
1	1	↑		Ambígua

Flip Flop SC (*SET-CLEAR*) ou
SR (*SET-RESET*)

Flip Flop com *clock*

Entradas			Saída	
S	C	CLK		Q
0	0	↑		Q_0 (não muda)
1	0	↑		1
0	1	↑		0
1	1	↑		Ambígua



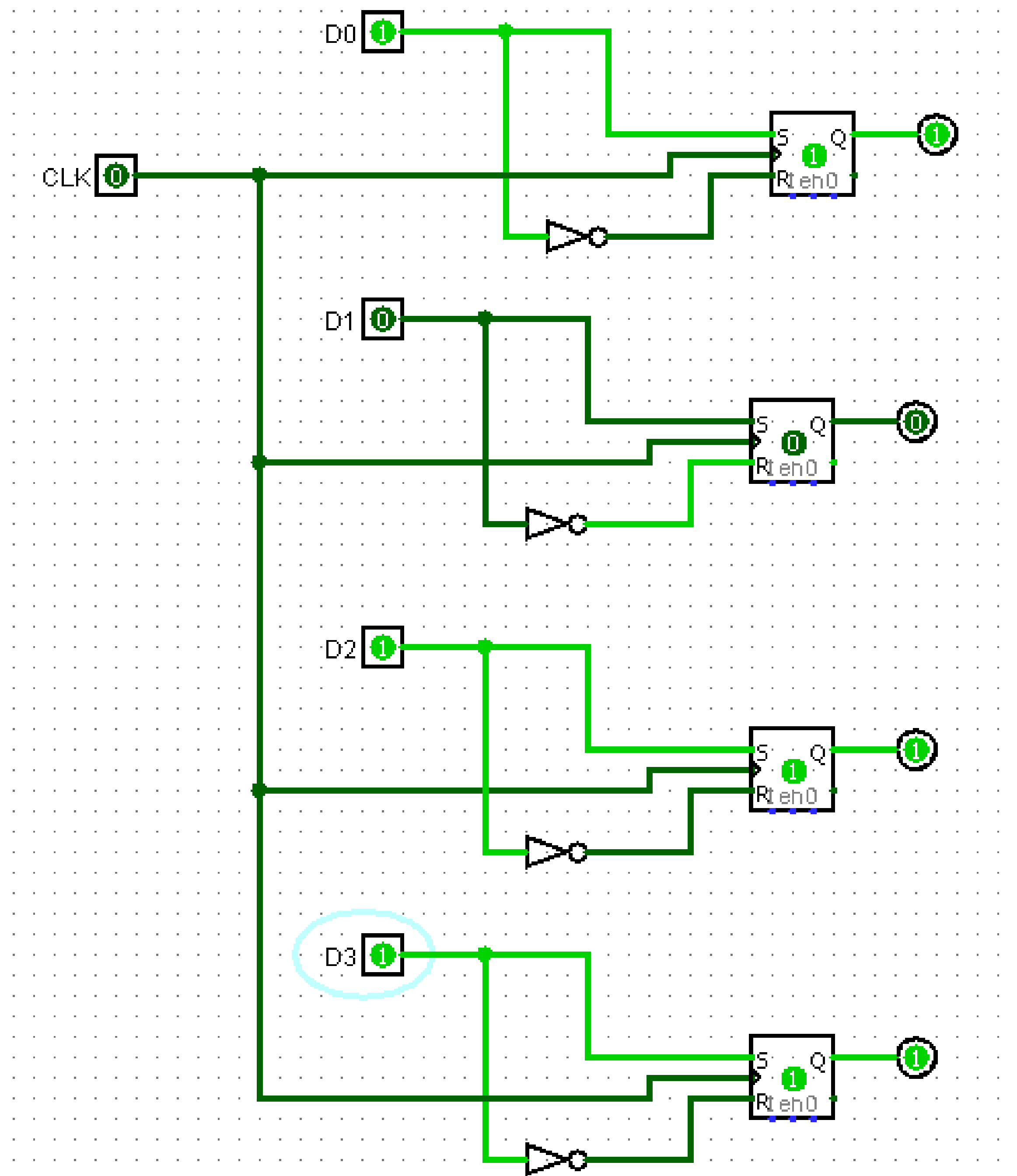
Flip Flop SC (*SET-CLEAR*) ou
SR (*SET-RESET*)

Uma aplicação básica

Entradas			Saída
S	C	CLK	Q
0	0	↑	Q_0 (não muda)
1	0	↑	1
0	1	↑	0
1	1	↑	Ambígua

Flip Flop SC (*SET-CLEAR*) ou
SR (*SET-RESET*)

Uma
aplicação
básica:
registrador
de 4 bits



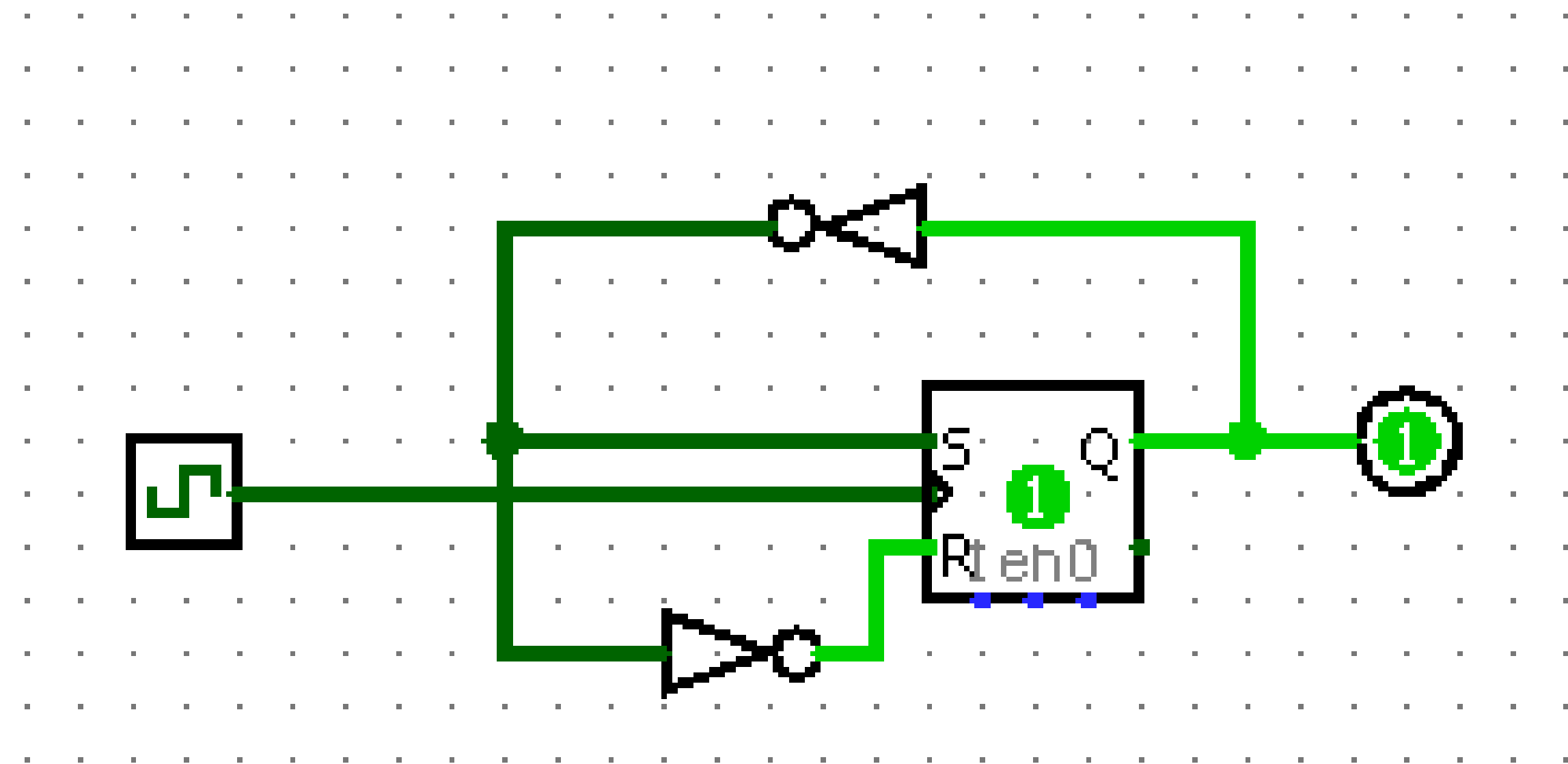
Flip Flop SC (*SET-CLEAR*) ou
SR (*SET-RESET*)

Outra aplicação básica

Entradas			Saída
S	C	CLK	Q
0	0	↑	Q_0 (não muda)
1	0	↑	1
0	1	↑	0
1	1	↑	Ambígua

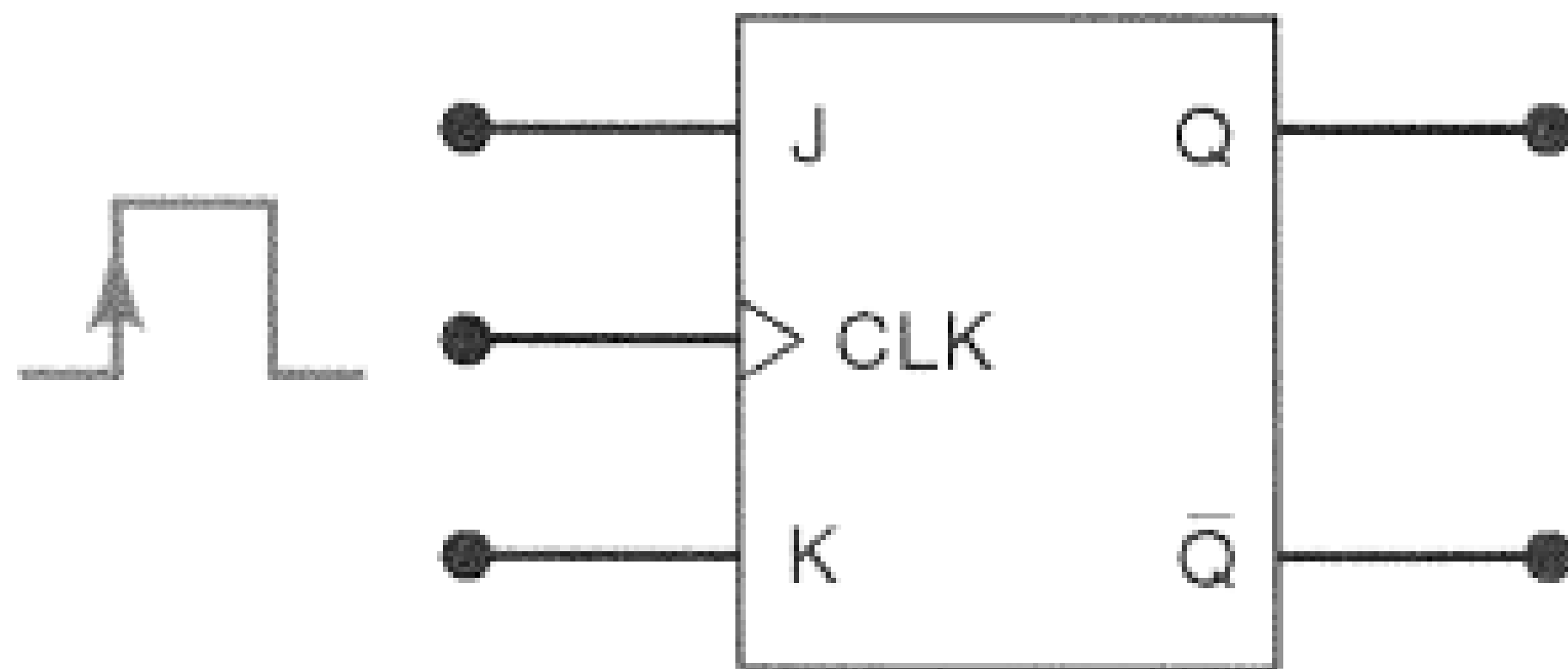
Flip Flop SC (*SET-CLEAR*) ou
SR (*SET-RESET*)

Outra
aplicação
básica:
divisor de
frequência



Flip Flop com *clock*

Flip Flop JK

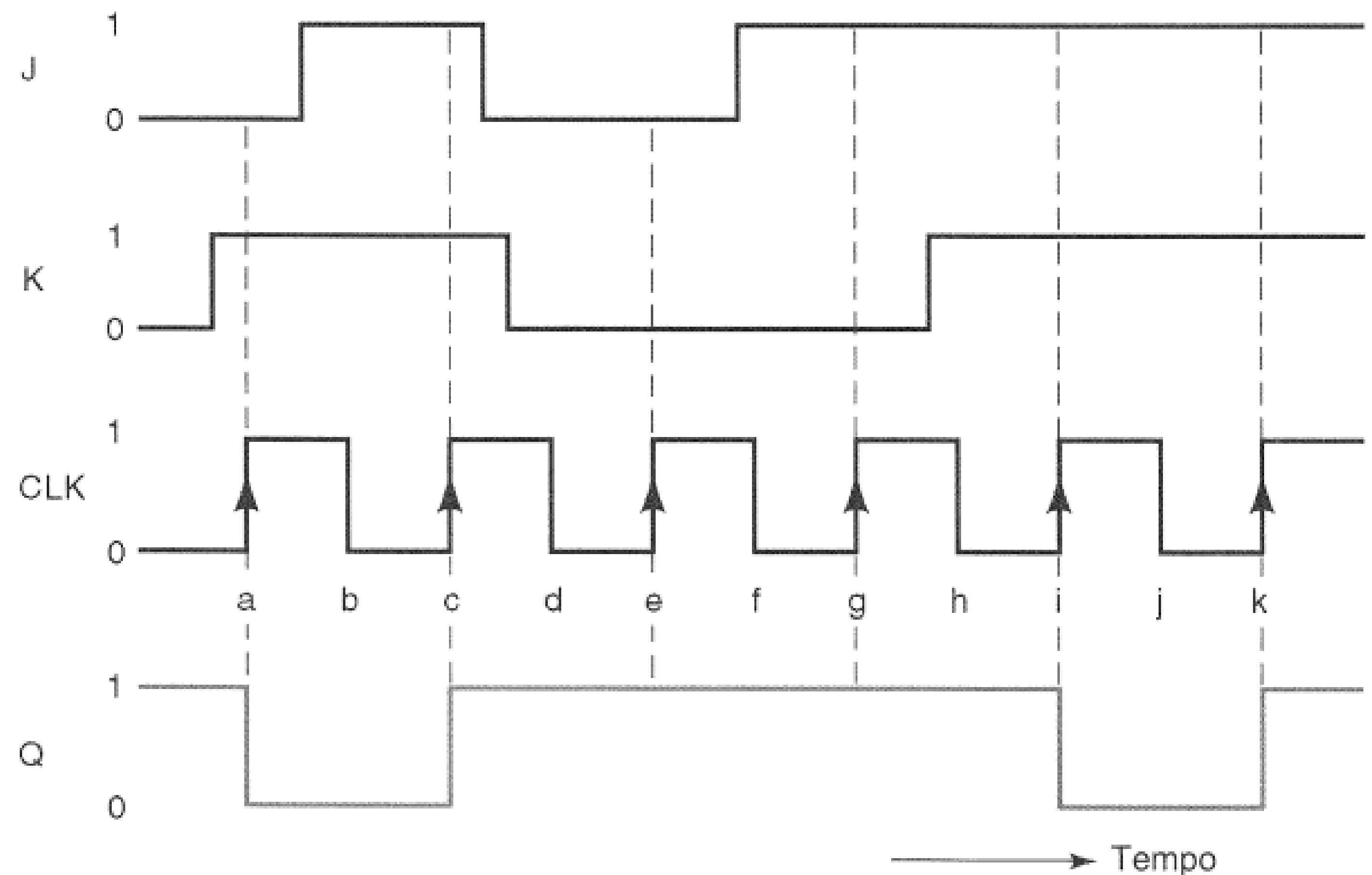


J	K	CLK	Q
0	0	\uparrow	Q_0 (não muda)
1	0	\uparrow	1
0	1	\uparrow	0
1	1	\uparrow	\bar{Q}_0 (comuta)

Flip Flop com *clock*

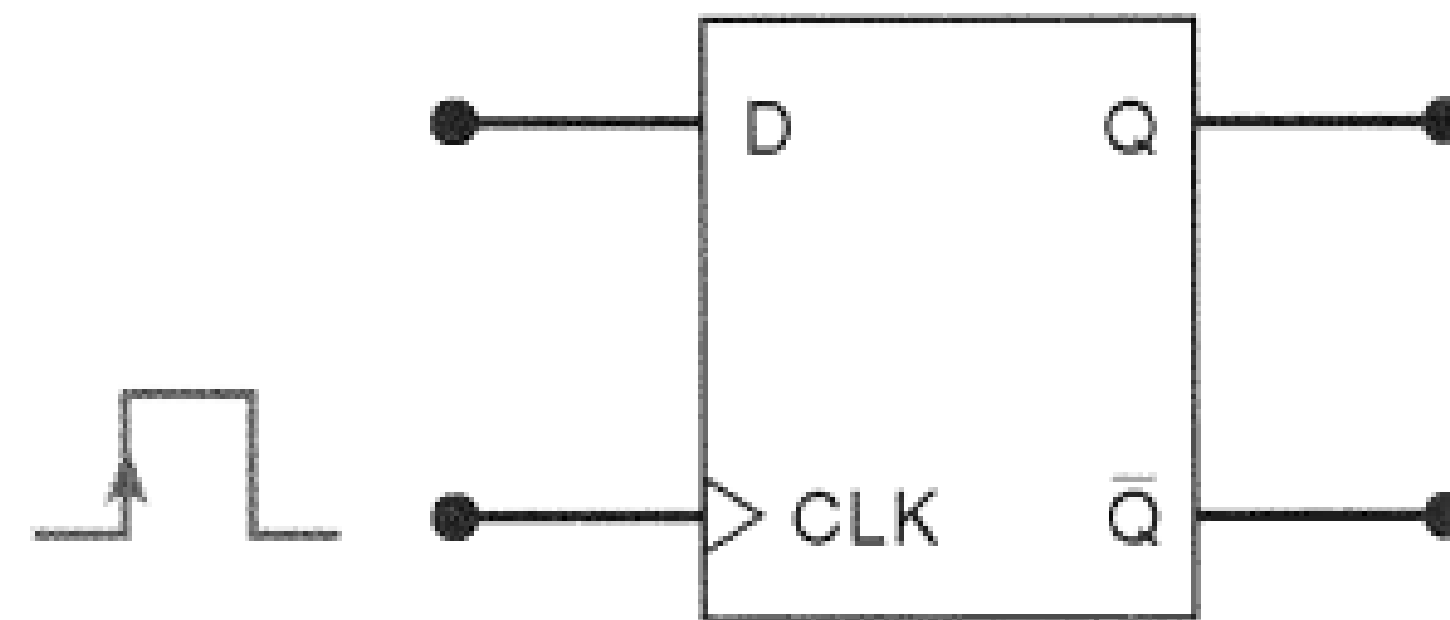
Flip Flop JK

J	K	CLK	Q
0	0	↑	Q_0 (não muda)
1	0	↑	1
0	1	↑	0
1	1	↑	$\overline{Q_0}$ (comuta)



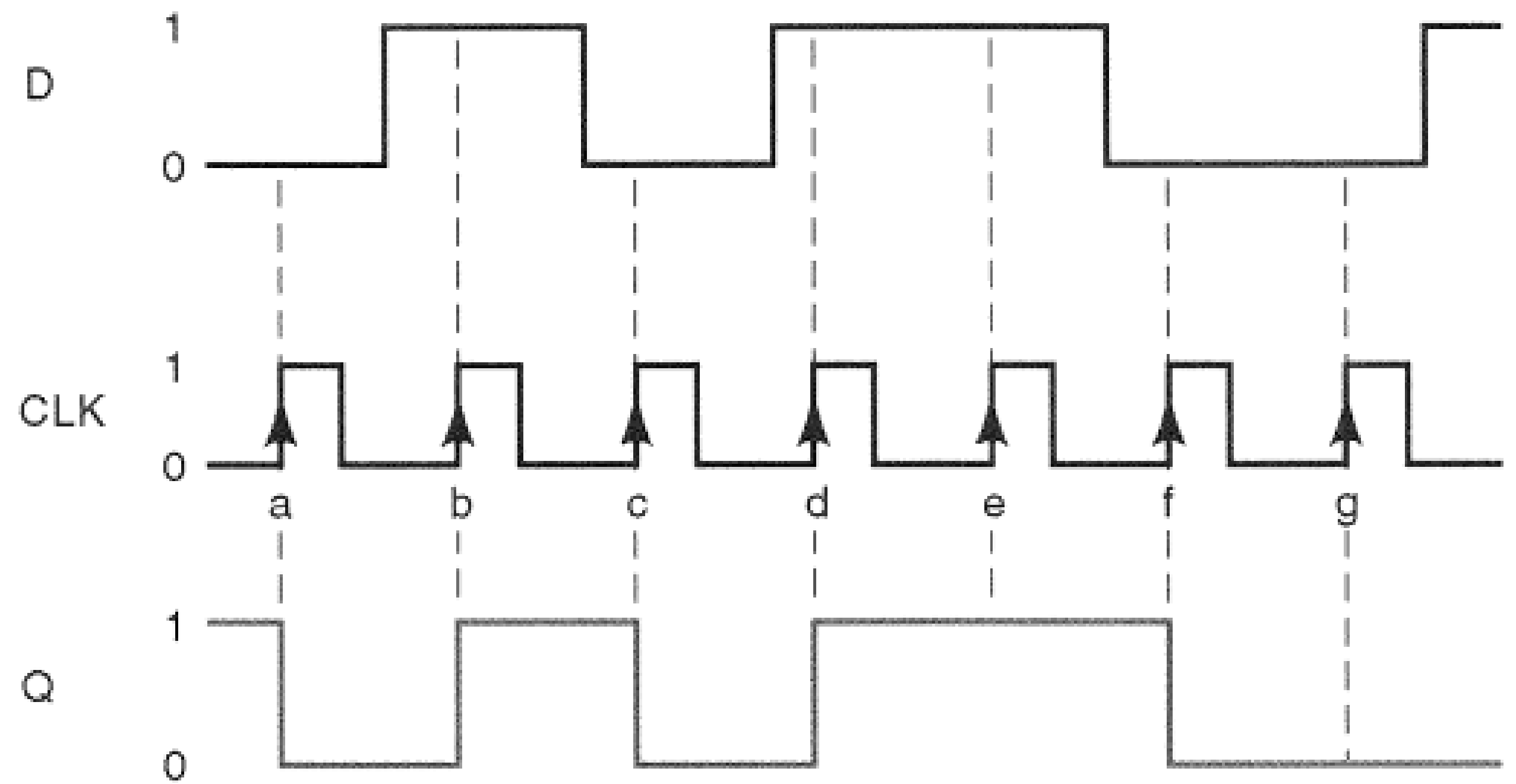
Flip Flop com *clock*

Flip Flop D



(a)

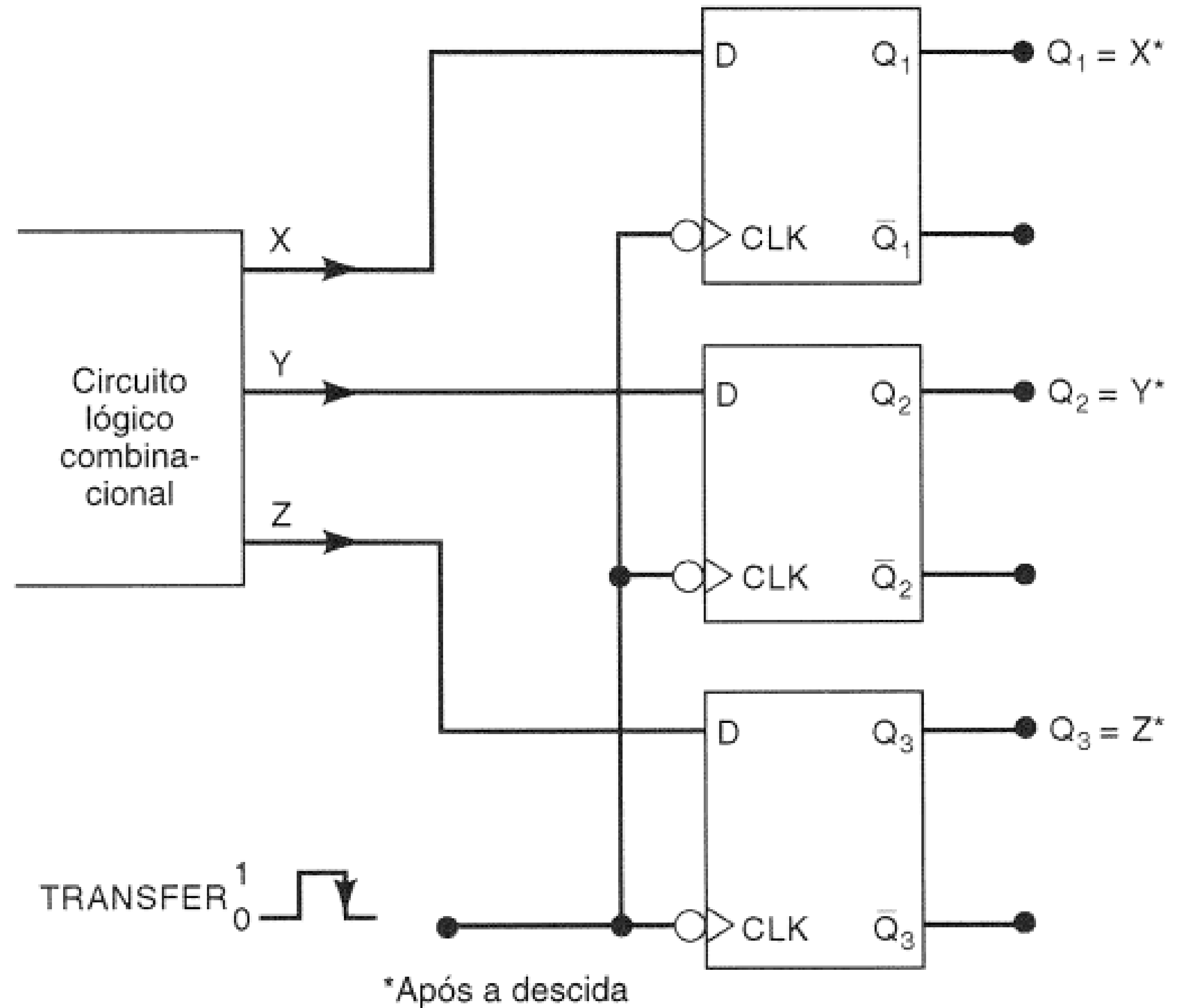
D	CLK	Q
0	↑	0
1	↑	1



Flip Flop D

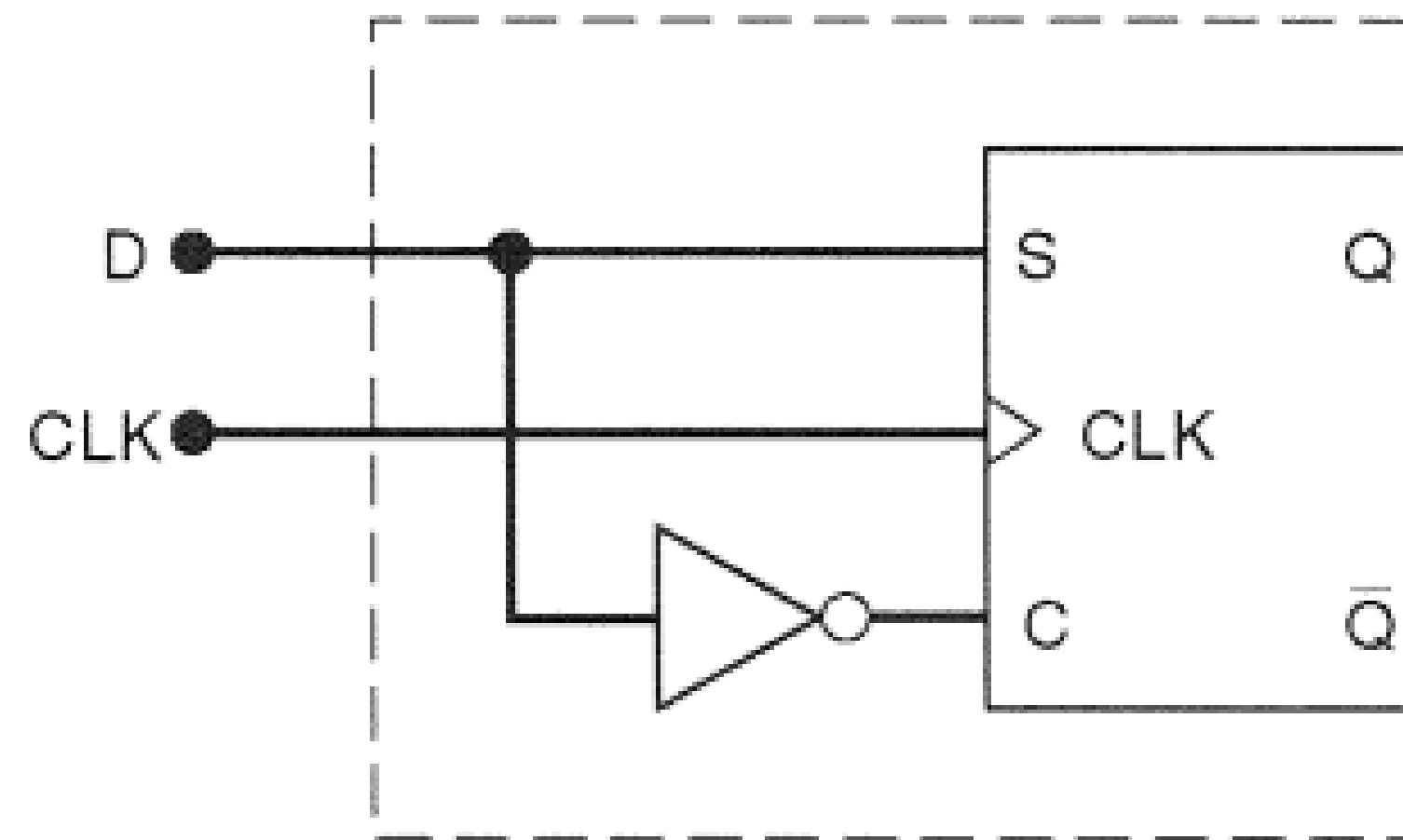
Flip Flop com *clock*

Uma aplicação:
Transferência de
dados em paralelo

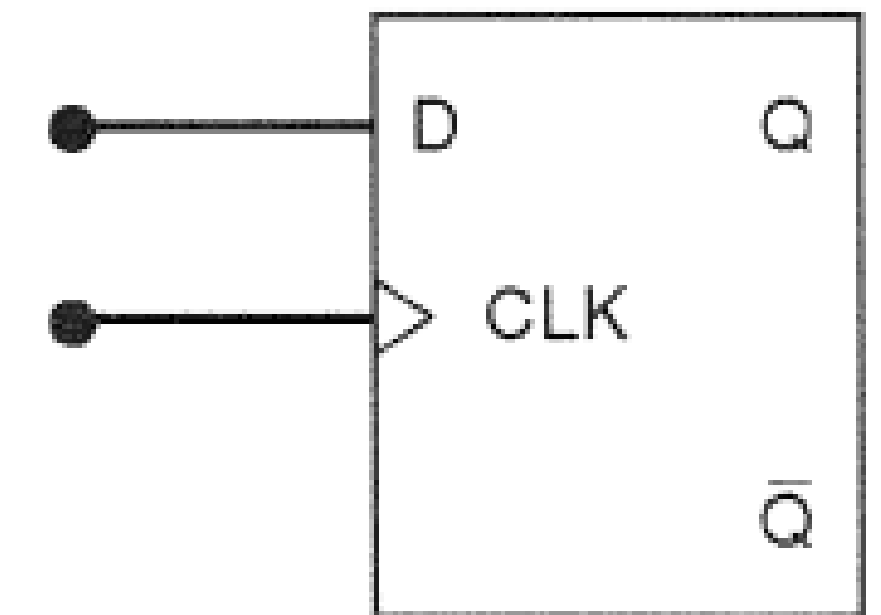


Implementação de FF D com FF SC

	D		SC	
CLK	D	Q	S	C
↑	0	0	0	1
↑	1	1	1	0

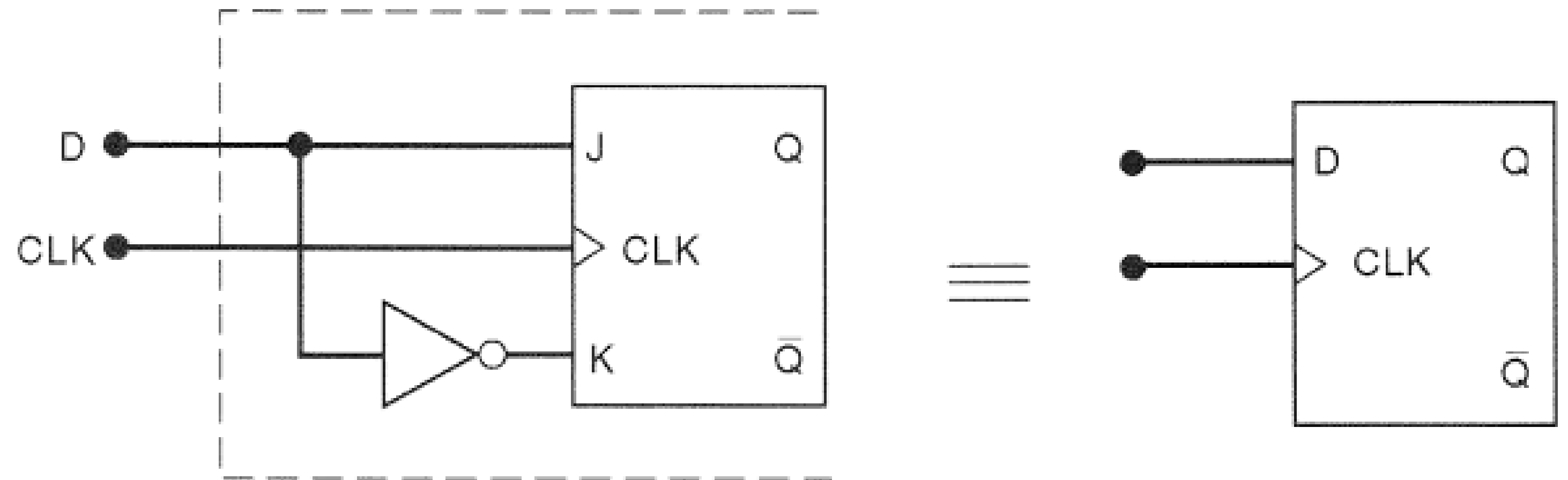


≡



Implementação de FF D com FF JK

	D		JK	
CLK	D	Q	J	K
↑	0	0	0	1
↑	1	1	1	0





IBMEC.BR

 /IBMEC

 IBMEC

 @IBMEC_OFICIAL

 @IBMEC

 **ibmec**