
Tema 4. Circuitos Secuenciales. Biestables

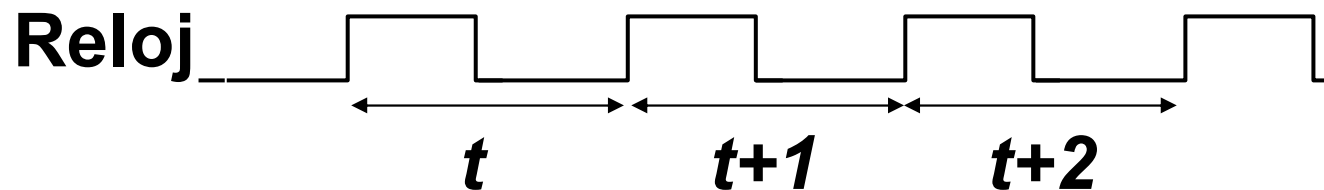
Fundamentos de los Computadores



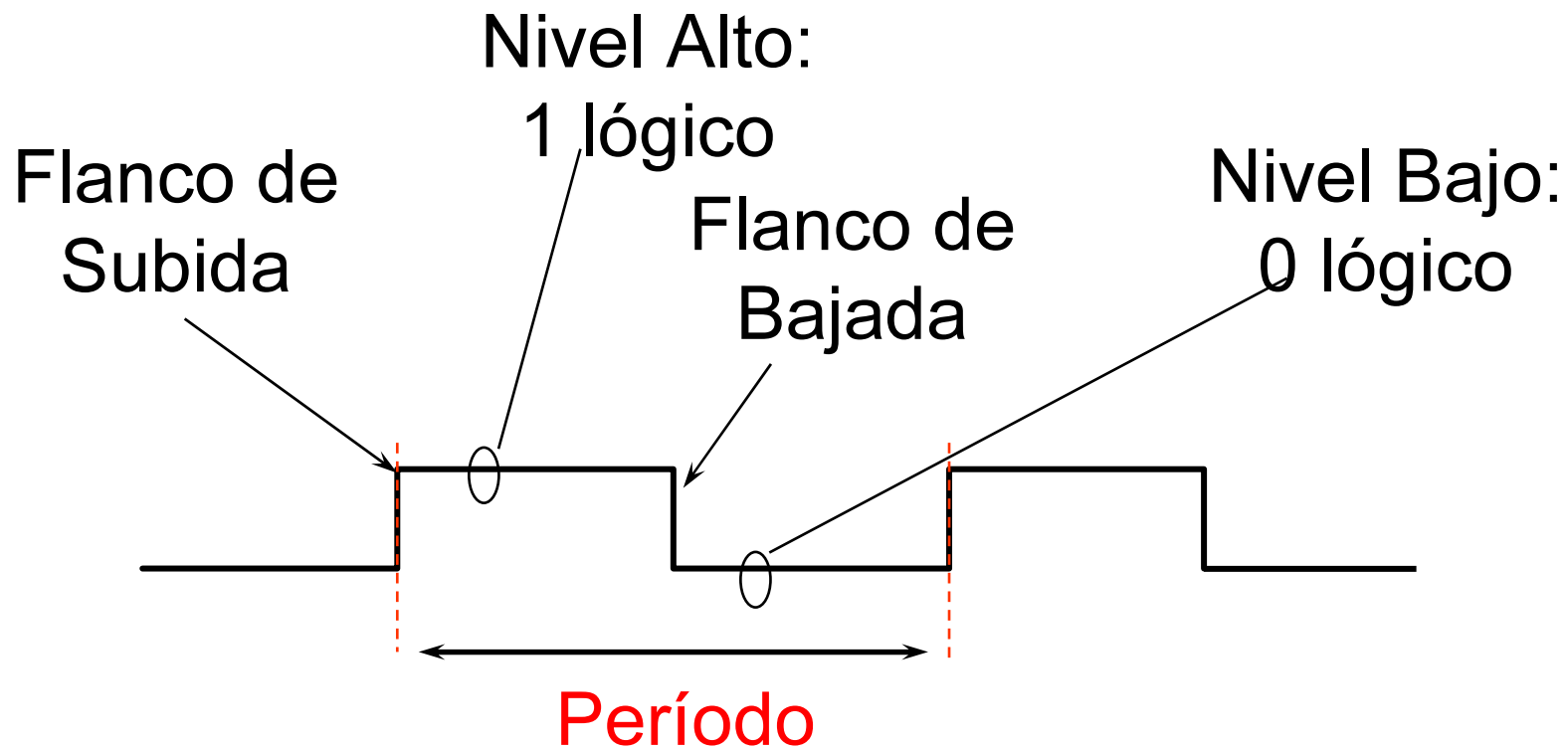
- Objetivos:
 - Estudiar los circuitos secuenciales más sencillos
 - Introducir el concepto de cronograma
 - Comprender el funcionamiento básico de los biestables

- Introducción
- El biestable S - R
 - Con puertas NOR
 - Con puertas NAND
- Entradas de habilitación
- Entradas asíncronas
- El biestable D
- El biestable D activo por flanco
- El biestable J - K
- El biestable T

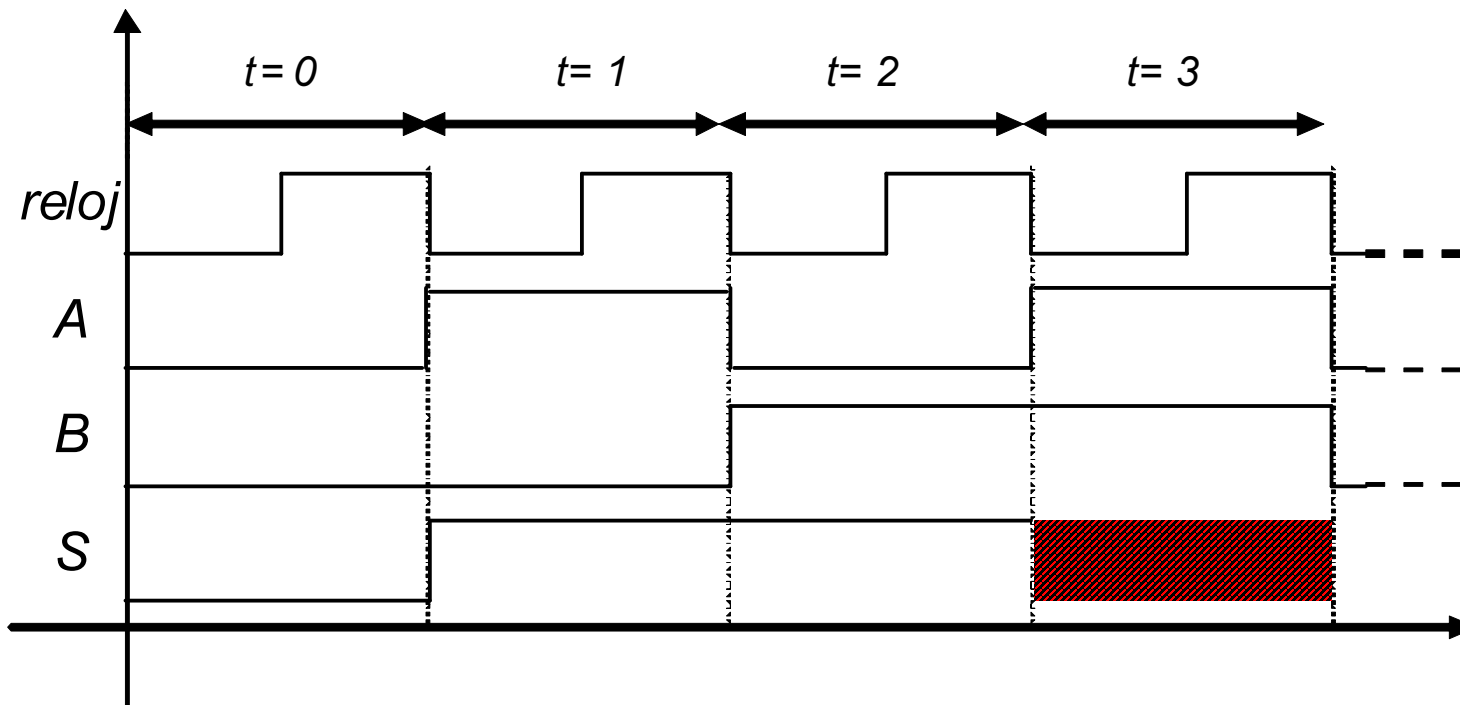
- Circuitos secuenciales:
 - las salidas del circuito en el instante actual $S(t)$ dependen no sólo del valor actual de las entradas del circuito $E(t)$, sino también de su “memoria” o “estado almacenado” $Q(t)$
 - Están formados por
 - un bloque combinacional
 - un bloque de elementos de memoria para almacenar el estado Q
 - una entrada de reloj que decide cuando pasamos del tiempo t al $t+1$, y que marca cuando se almacena interiormente el estado del sistema



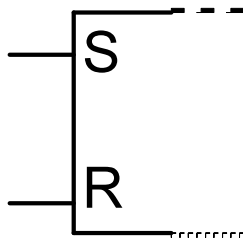
- La señal de reloj indica a los elementos de memoria cuándo deben cambiar su estado



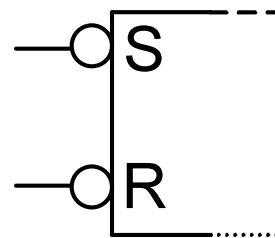
- **Cronograma:** Representación de la evolución temporal de las entradas y salidas de un circuito.
 - El valor indefinido se representa **sombreado**



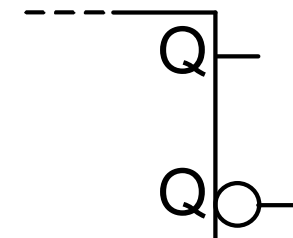
- Símbolos lógicos
 - Entradas y salidas



Entradas activas a nivel alto

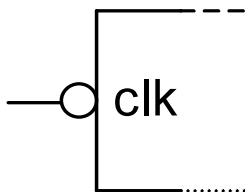


Entradas activas a nivel bajo

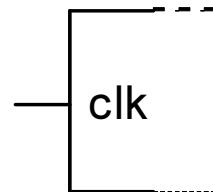


Salidas Q y /Q

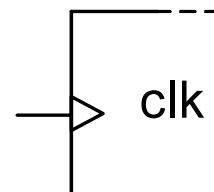
- Señal de reloj



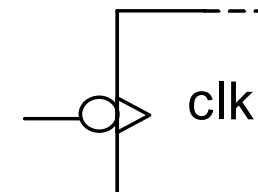
Activa a nivel bajo



Activa a nivel alto



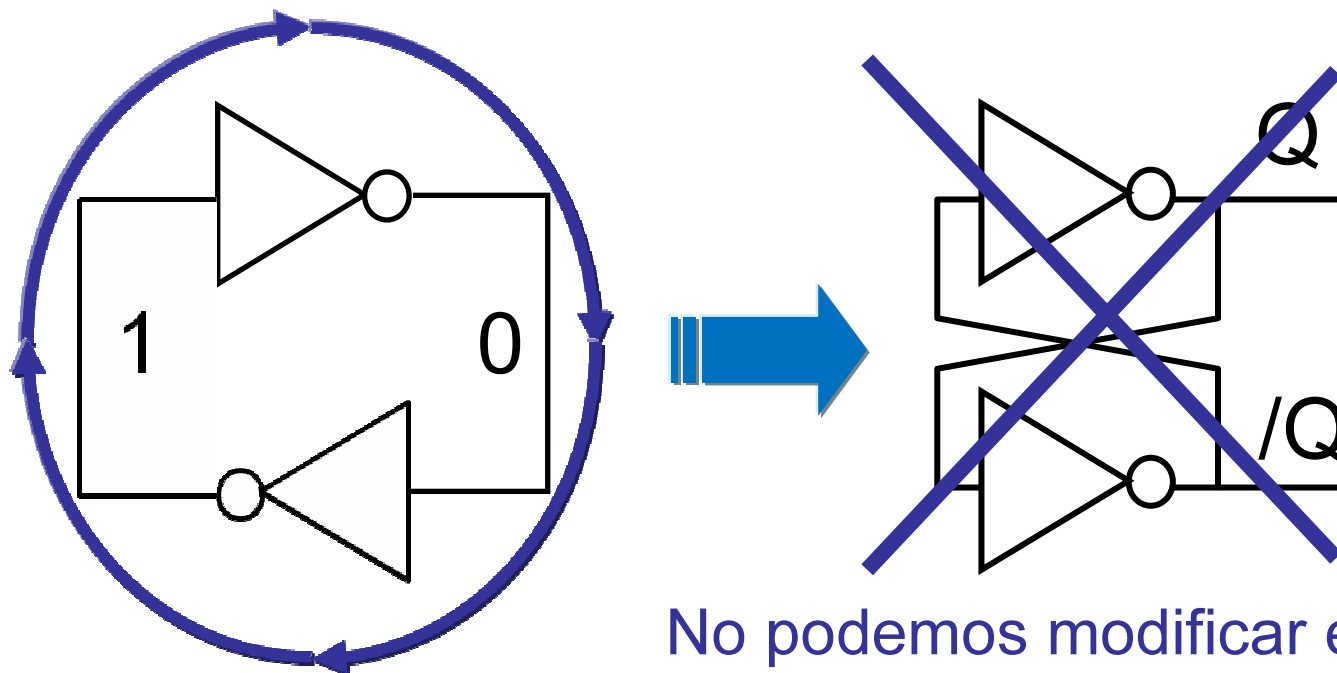
Activa por flanco de subida



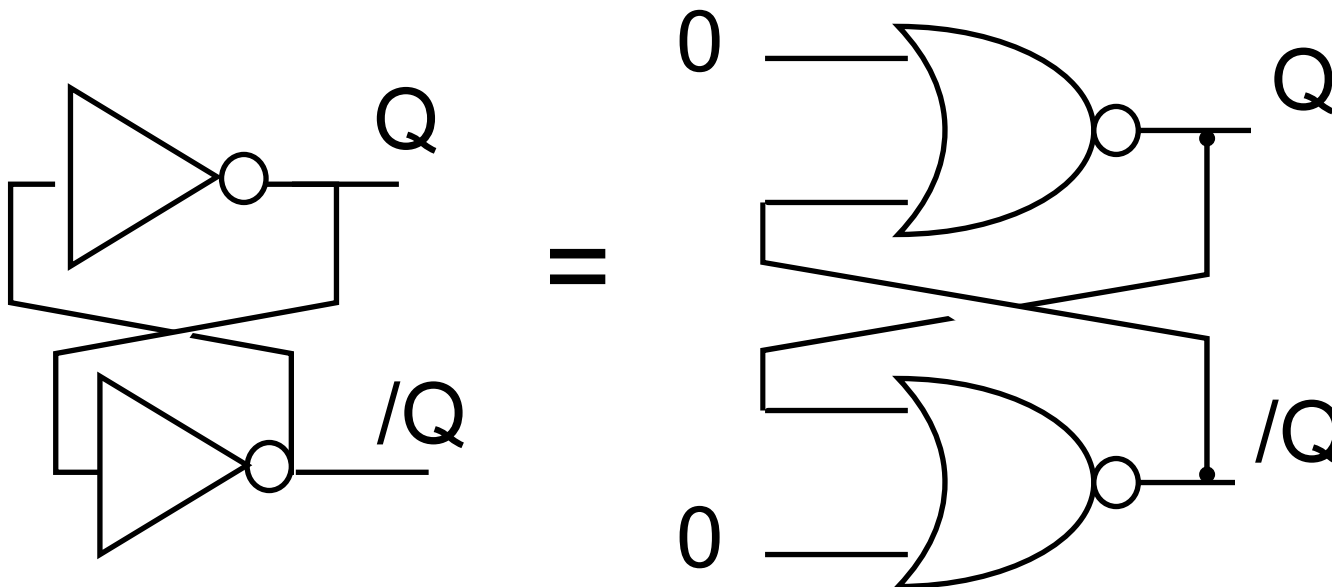
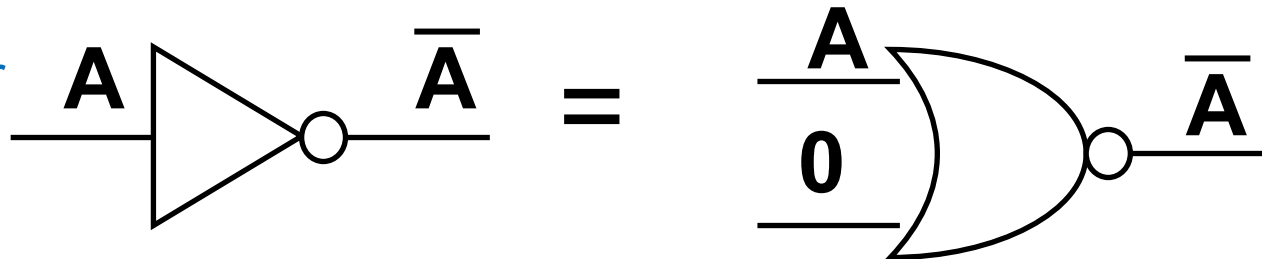
Activa por flanco de bajada

- Biestable
 - Circuito secuencial con dos estados estables (0 y 1)

Podemos almacenar un bit en un circuito haciendo que de vueltas y vueltas, de manera que la información recircule indefinidamente



¿Cómo cambiar el estado del biestable?



BIESTABLE S-R

- Biestable S-R con puertas NOR
 - Activar entrada S para hacer $Q = 1$
 - Activar entrada R para hacer $Q = 0$

- Si $S=R=0$:

$$Q = \overline{\overline{Q} + 0} = \overline{\overline{Q}}$$

$$\overline{Q} = \overline{Q + 0} = \overline{\overline{Q}}$$

el estado se mantiene

- Si $S=0, R=1$: $Q = \overline{\overline{Q} + 1} = \overline{1} = 0$
 $\overline{Q} = \overline{Q + 0} = \overline{Q} = 1$
 el estado se pone a 0

- Si $S=1, R=0$: $\overline{Q} = \overline{Q + 1} = \overline{1} = 0$
 $Q = \overline{\overline{Q} + 0} = \overline{0} = 1$
 el estado se pone a 1

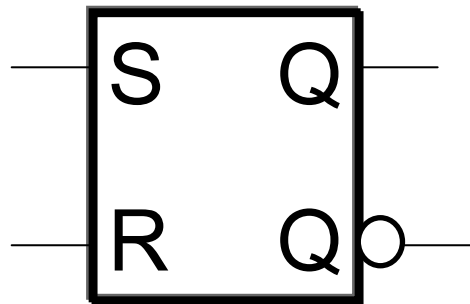
- Si se activan al mismo tiempo S y R, llegamos a una situación no deseada

Es el único caso en el que $\overline{Q} \neq Q$,

- Si **S=R=1**: $Q = \overline{\overline{Q} + 1} = \overline{1} = 0$
 $\overline{Q} = \overline{Q + 1} = \overline{1} = 0$

y si después hacemos S=R=0, dado que $\overline{Q} \neq Q$,
 el biestable evolucionará a un **estado indefinido**

Símbolo lógico

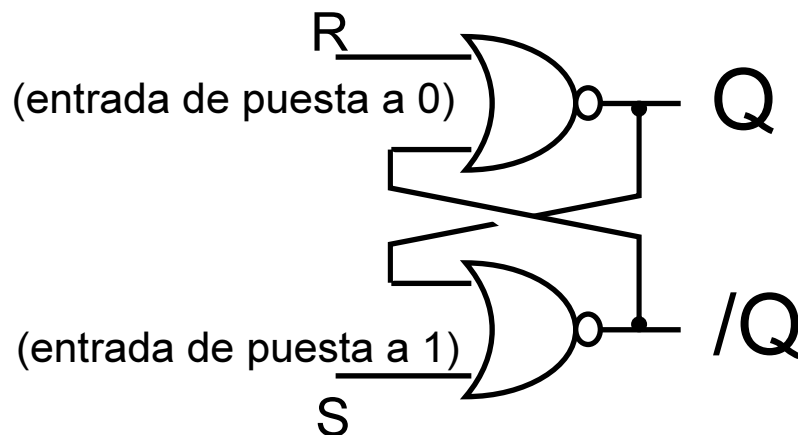


S	R	Q(t)	Q(t+1)	/Q(t+1)
0	0	0	0	1
0	0	1	1	0
0	1	0	0	1
0	1	1	0	1
1	0	0	1	0
1	0	1	1	0
1	1	0	0*	0*
1	1	1	0*	0*

Tabla de verdad

* indica situación no deseada

Diagrama lógico

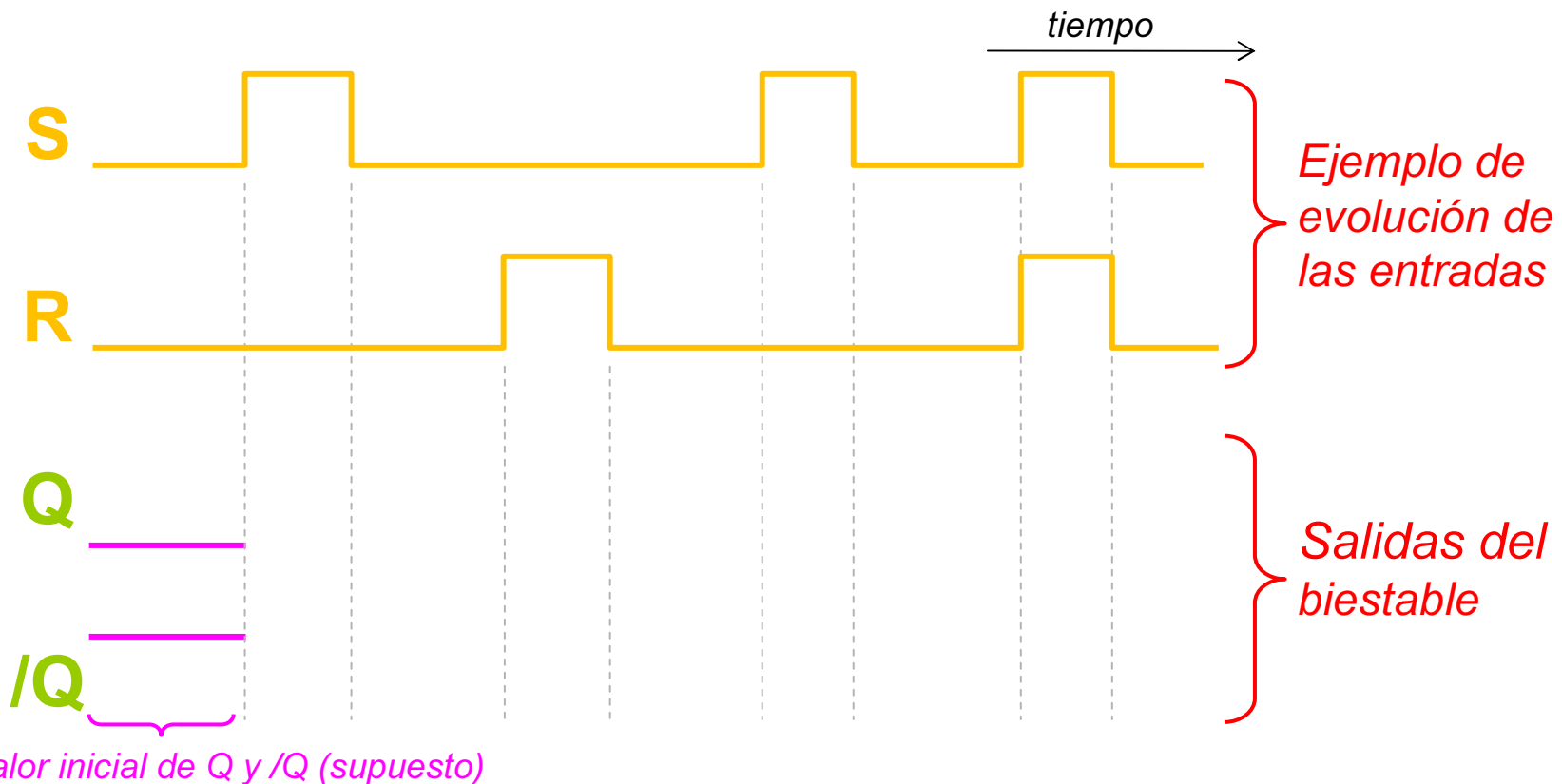


S	R	Q(t+1)	/Q(t+1)
0	0	Q(t)	/Q(t)
0	1	0	1
1	0	1	0
1	1	0*	0*

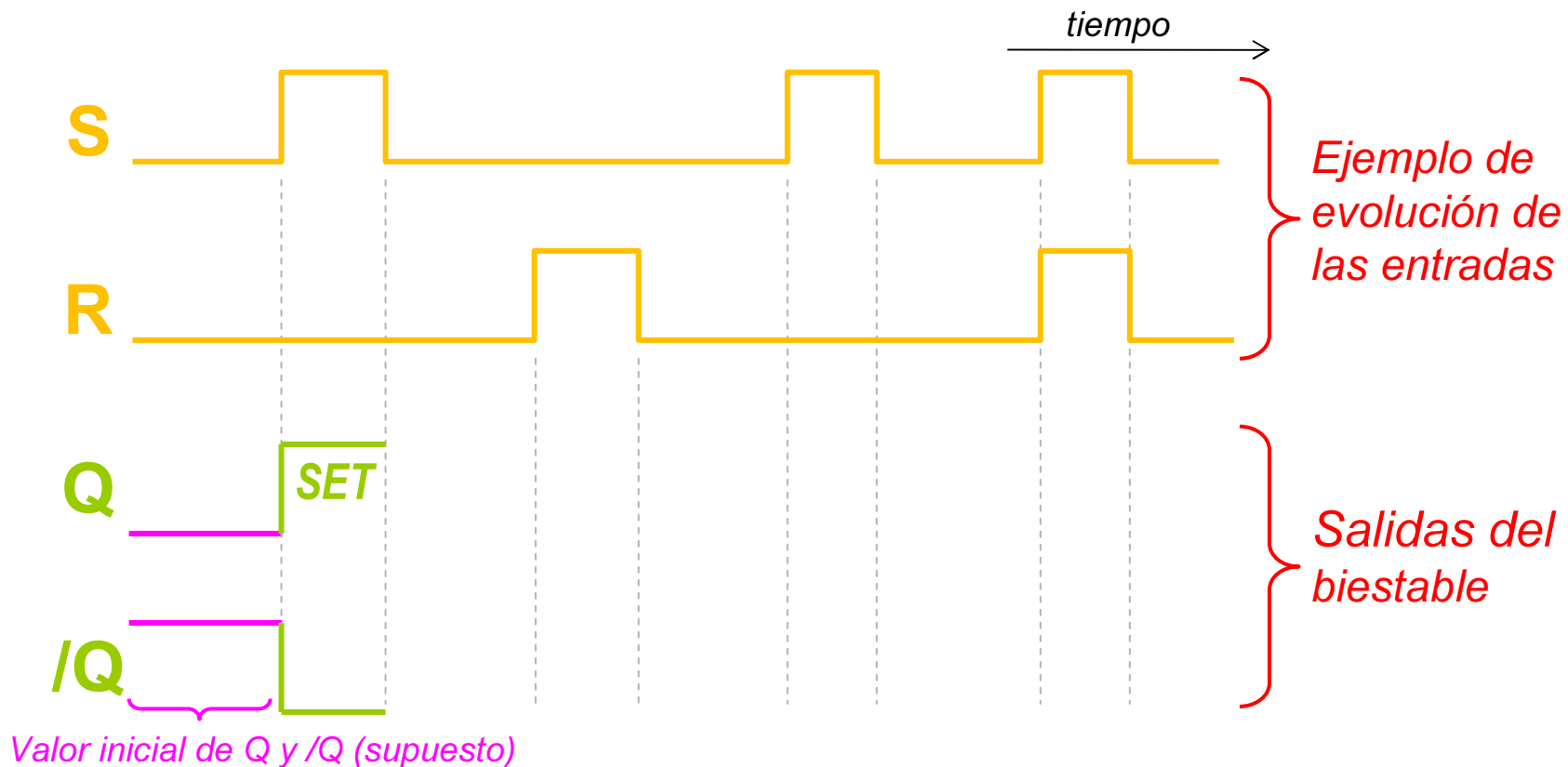
Tabla de verdad reducida

* indica situación no deseada

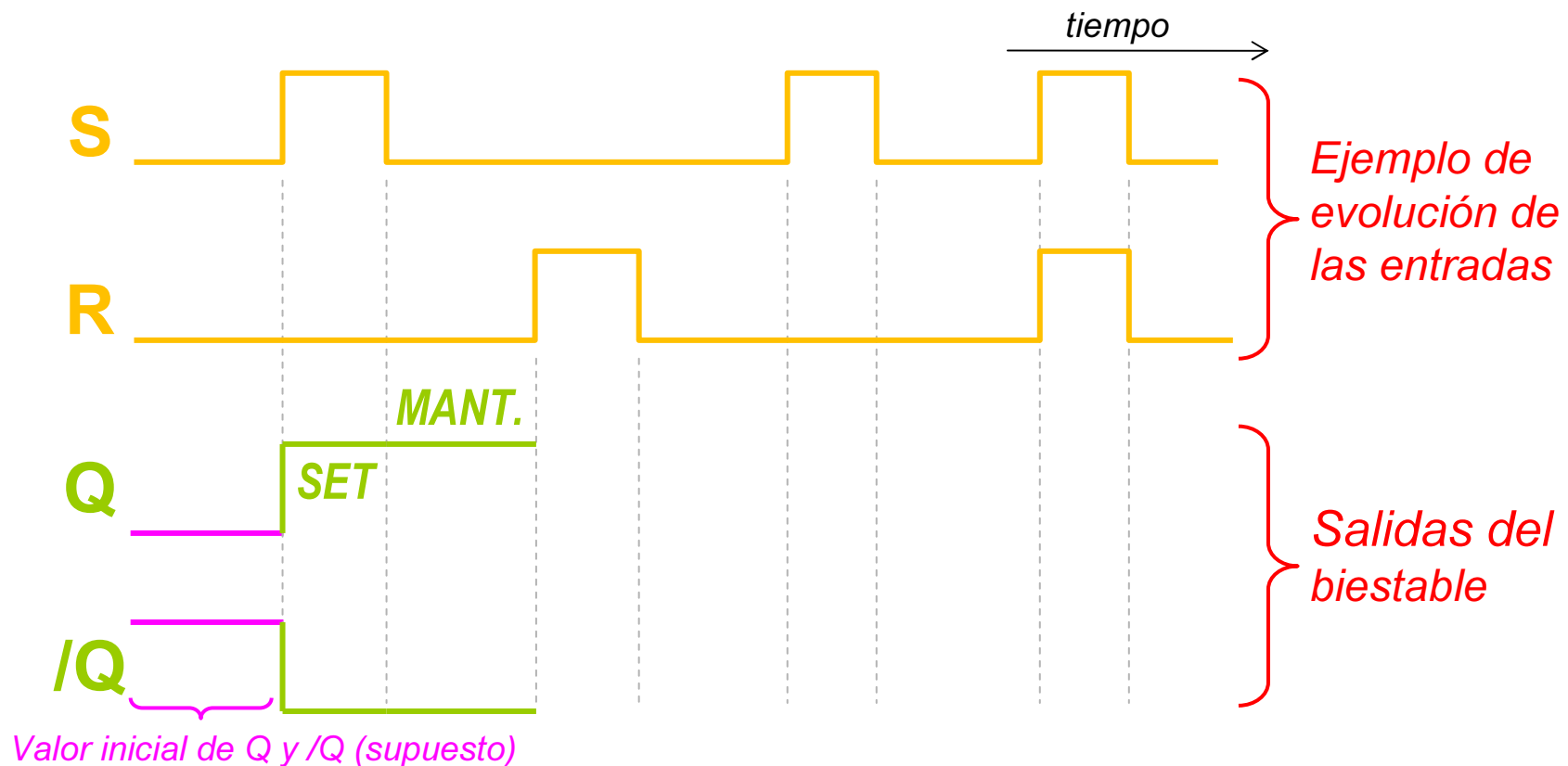
- El cronograma se utiliza para conocer la evolución temporal del estado del biastable cuando cambian sus entradas



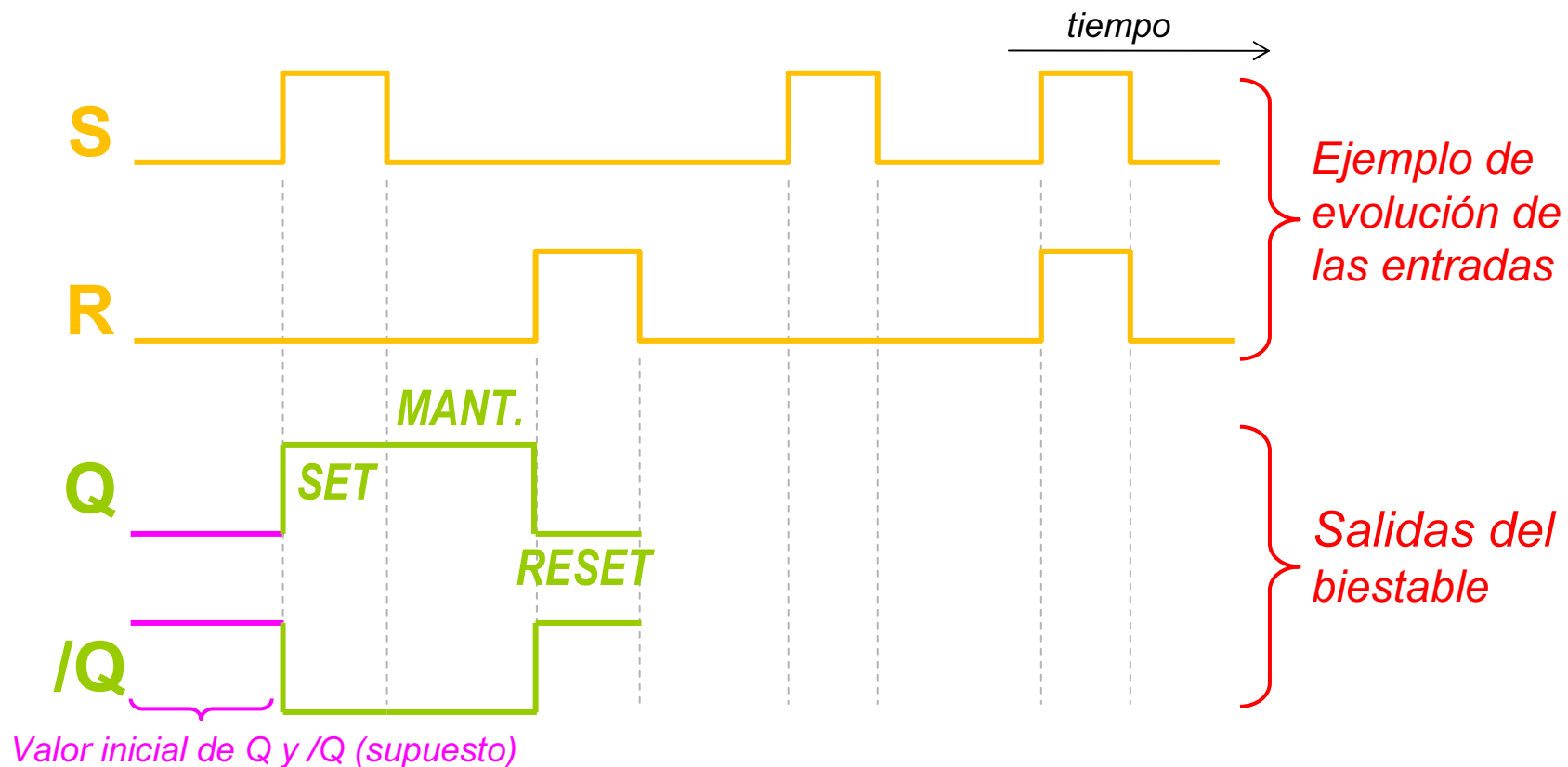
- El cronograma se utiliza para conocer la evolución temporal del estado del biestable cuando cambian sus entradas



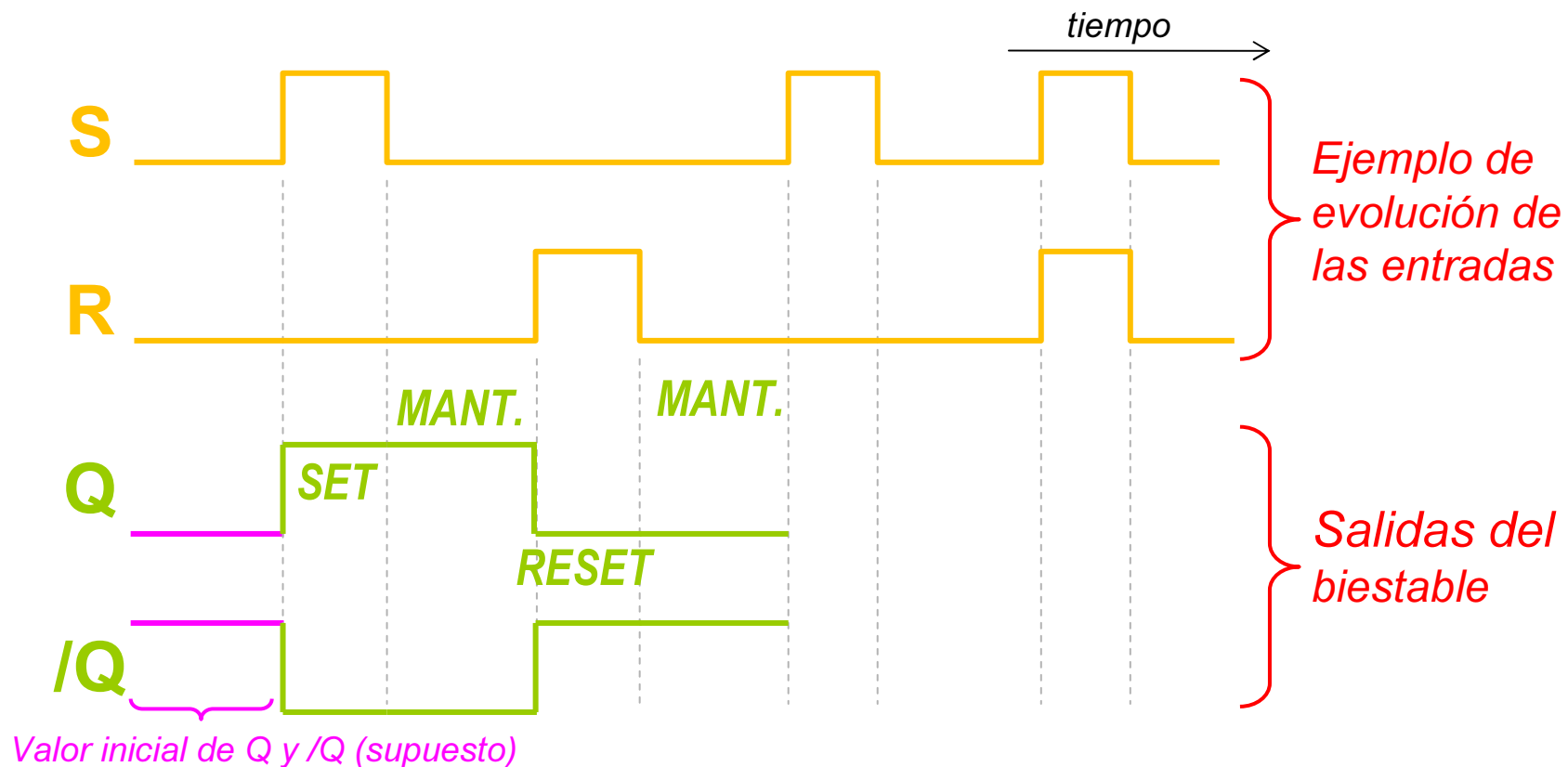
- El cronograma se utiliza para conocer la evolución temporal del estado del biastable cuando cambian sus entradas



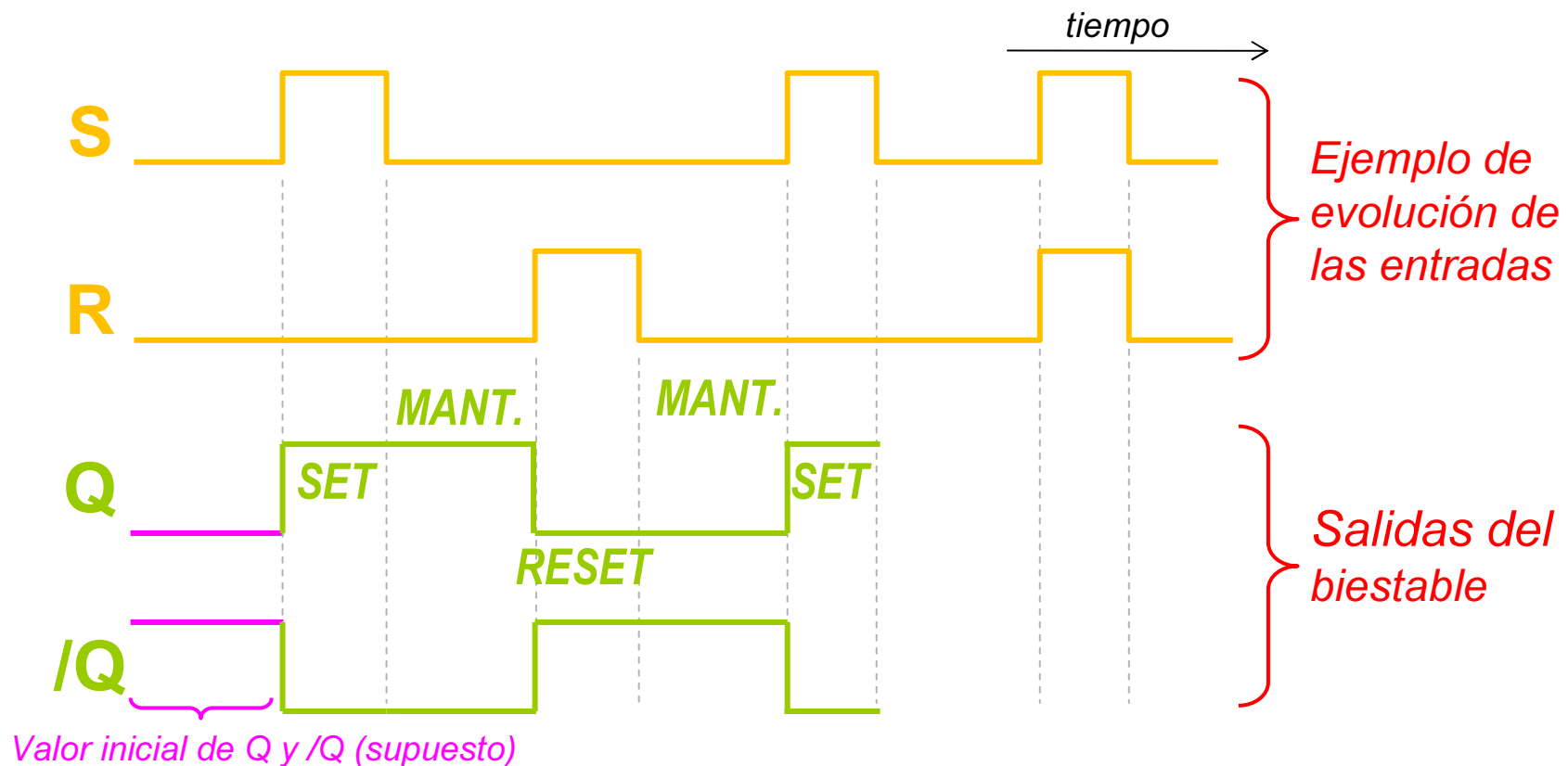
- El cronograma se utiliza para conocer la evolución temporal del estado del biestable cuando cambian sus entradas



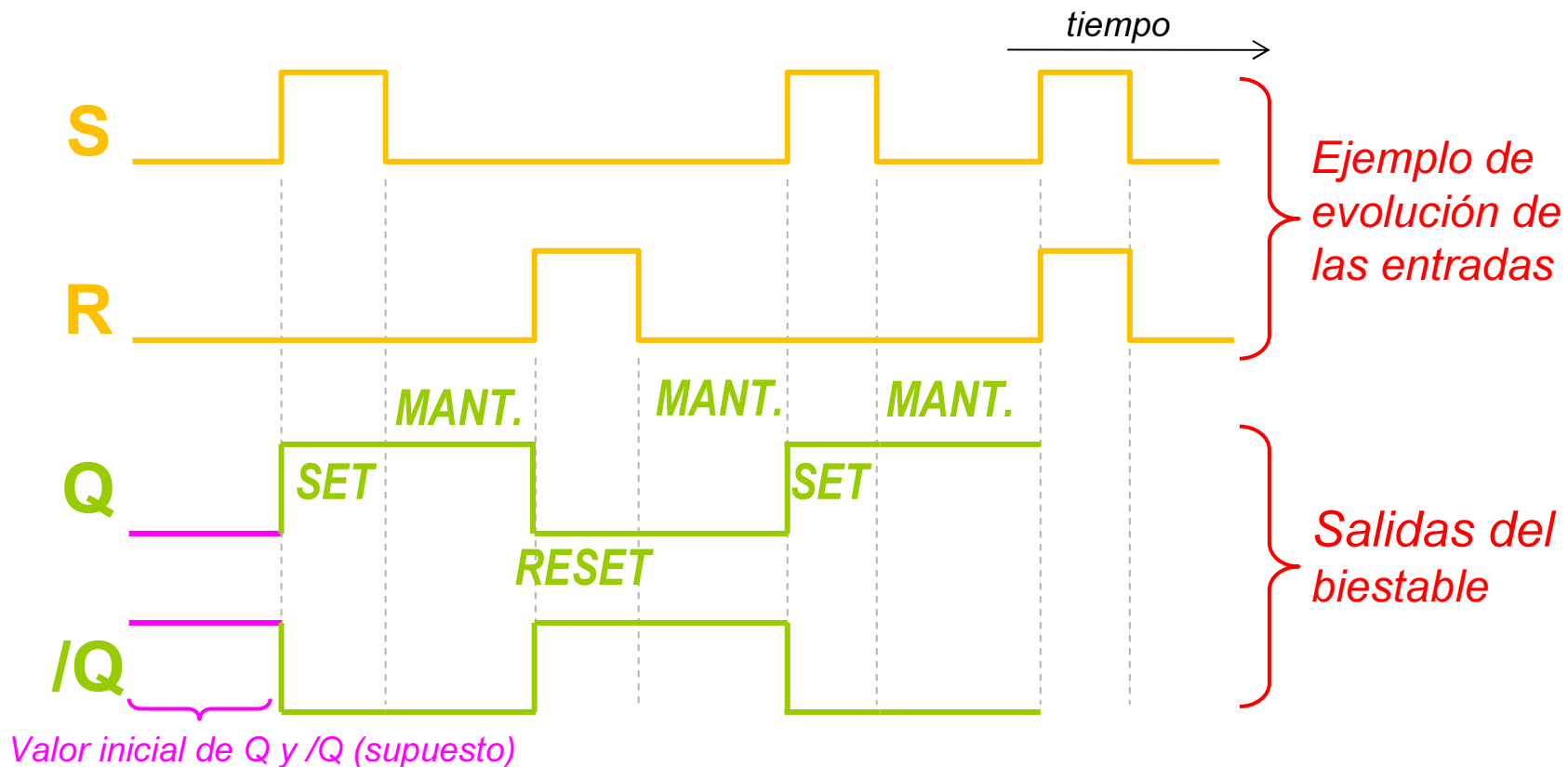
- El cronograma se utiliza para conocer la evolución temporal del estado del biestable cuando cambian sus entradas



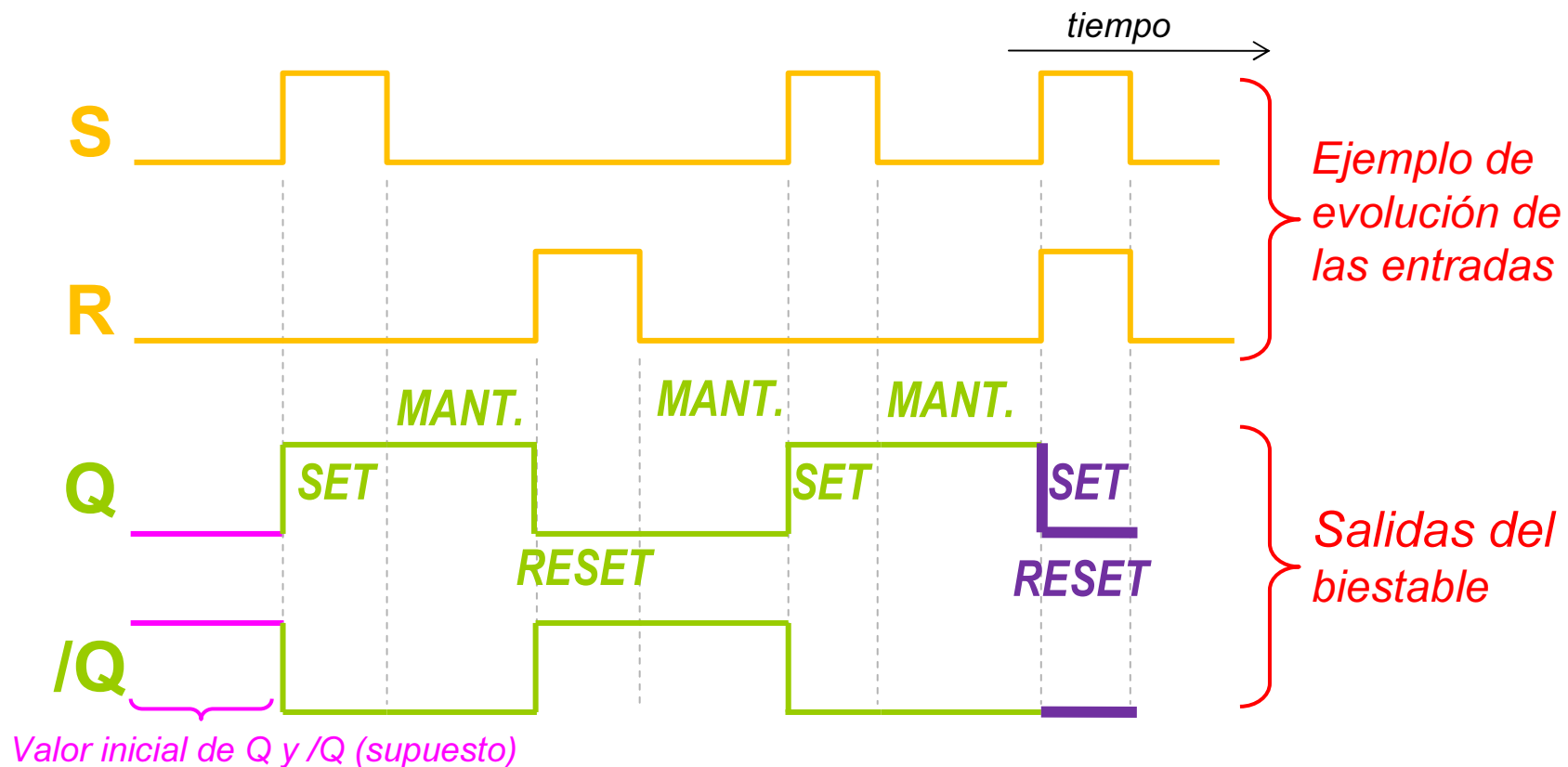
- El cronograma se utiliza para conocer la evolución temporal del estado del biastable cuando cambian sus entradas



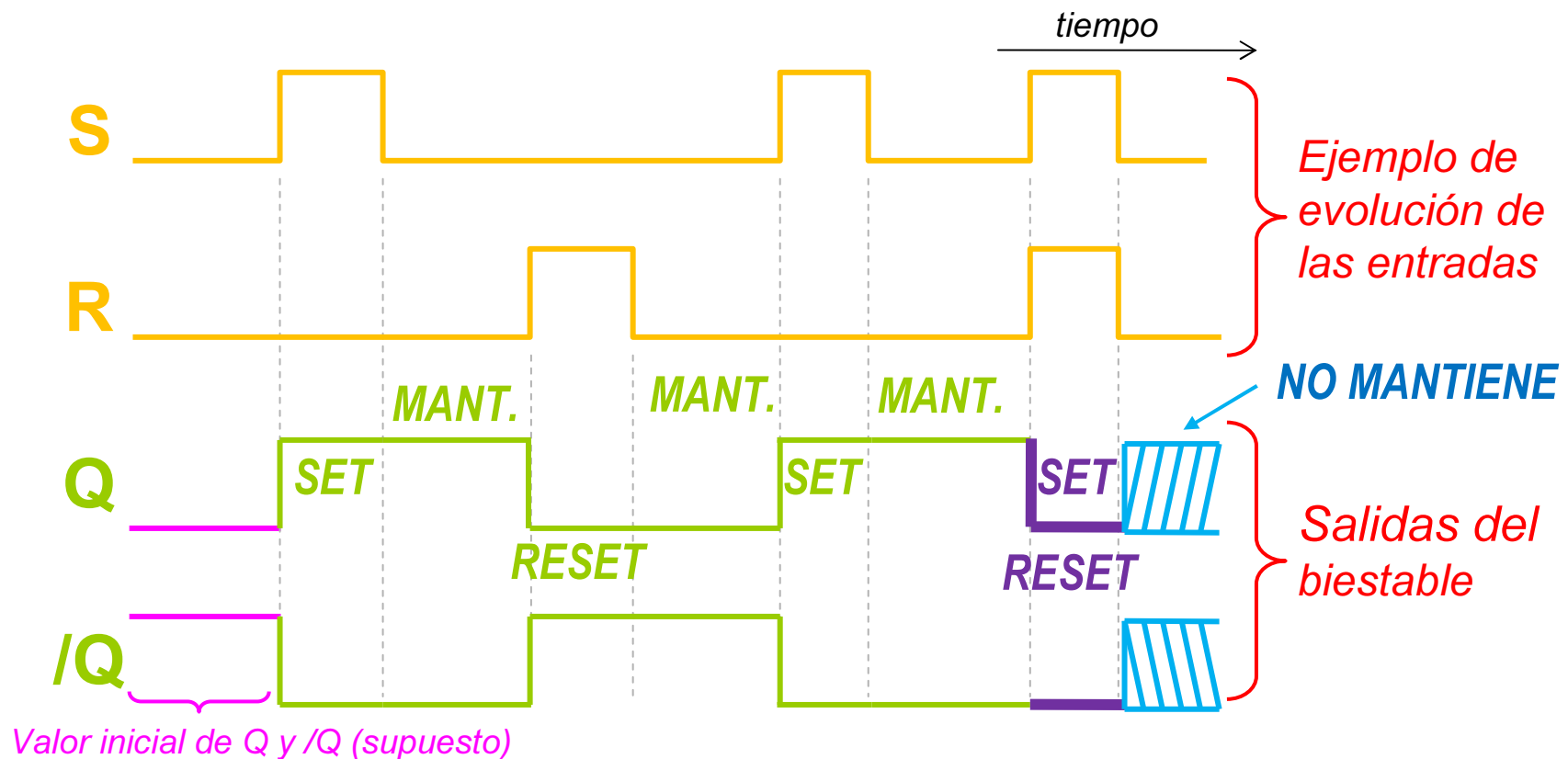
- El cronograma se utiliza para conocer la evolución temporal del estado del biestable cuando cambian sus entradas



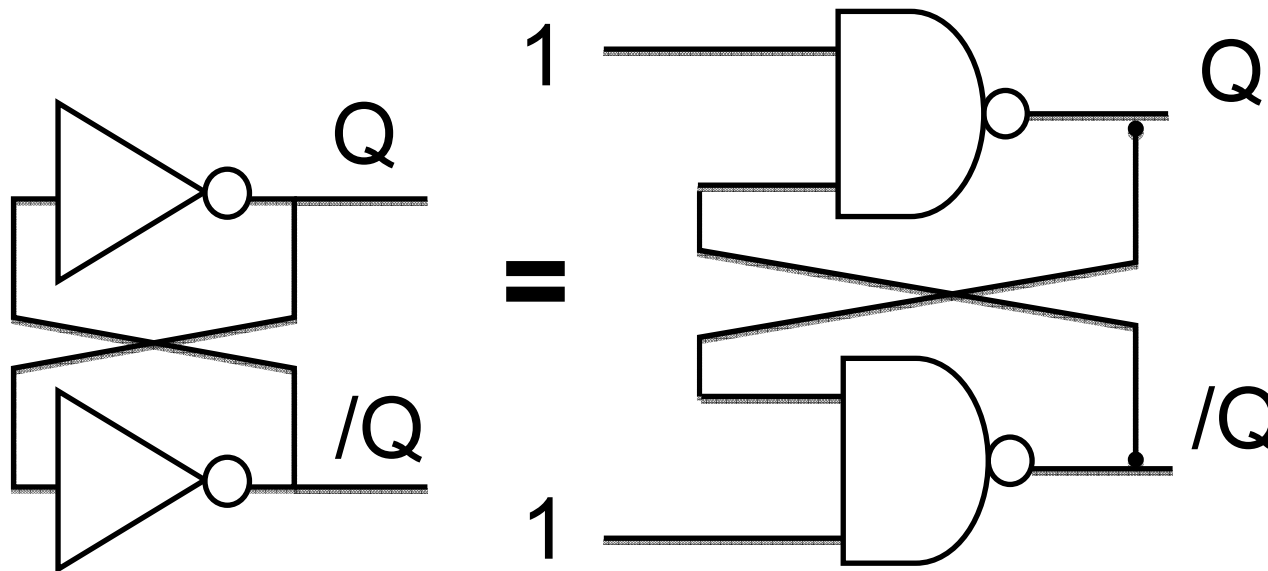
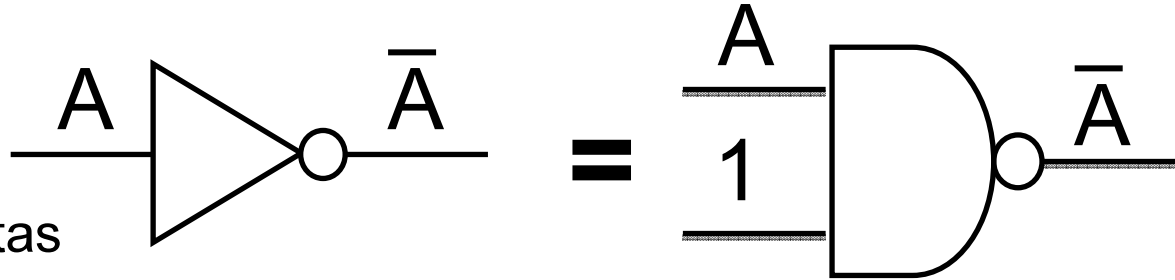
- El cronograma se utiliza para conocer la evolución temporal del estado del biestable cuando cambian sus entradas



- El cronograma se utiliza para conocer la evolución temporal del estado del biestable cuando cambian sus entradas



Se podría hacer un biastable con otro tipo de puertas



BIESTABLE S-R

- Las entradas S y R se activan en este caso a nivel bajo:

Si $(/S)=(/R)=1$
el estado se mantiene

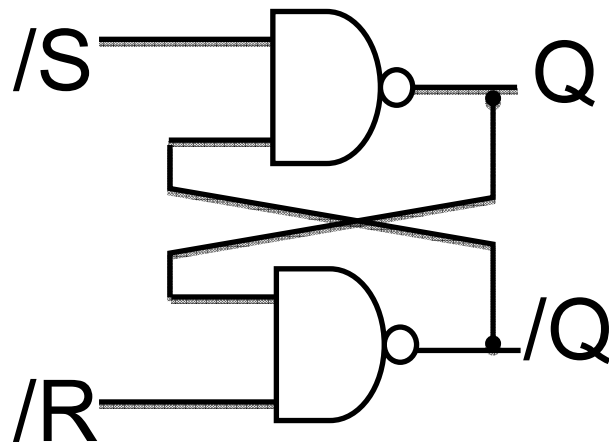
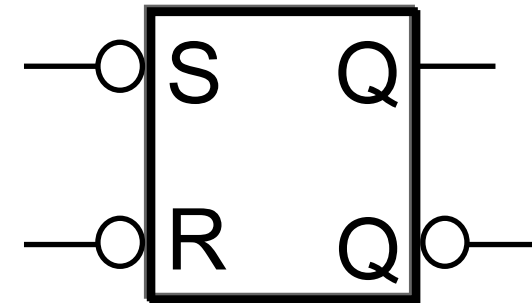


Diagrama lógico



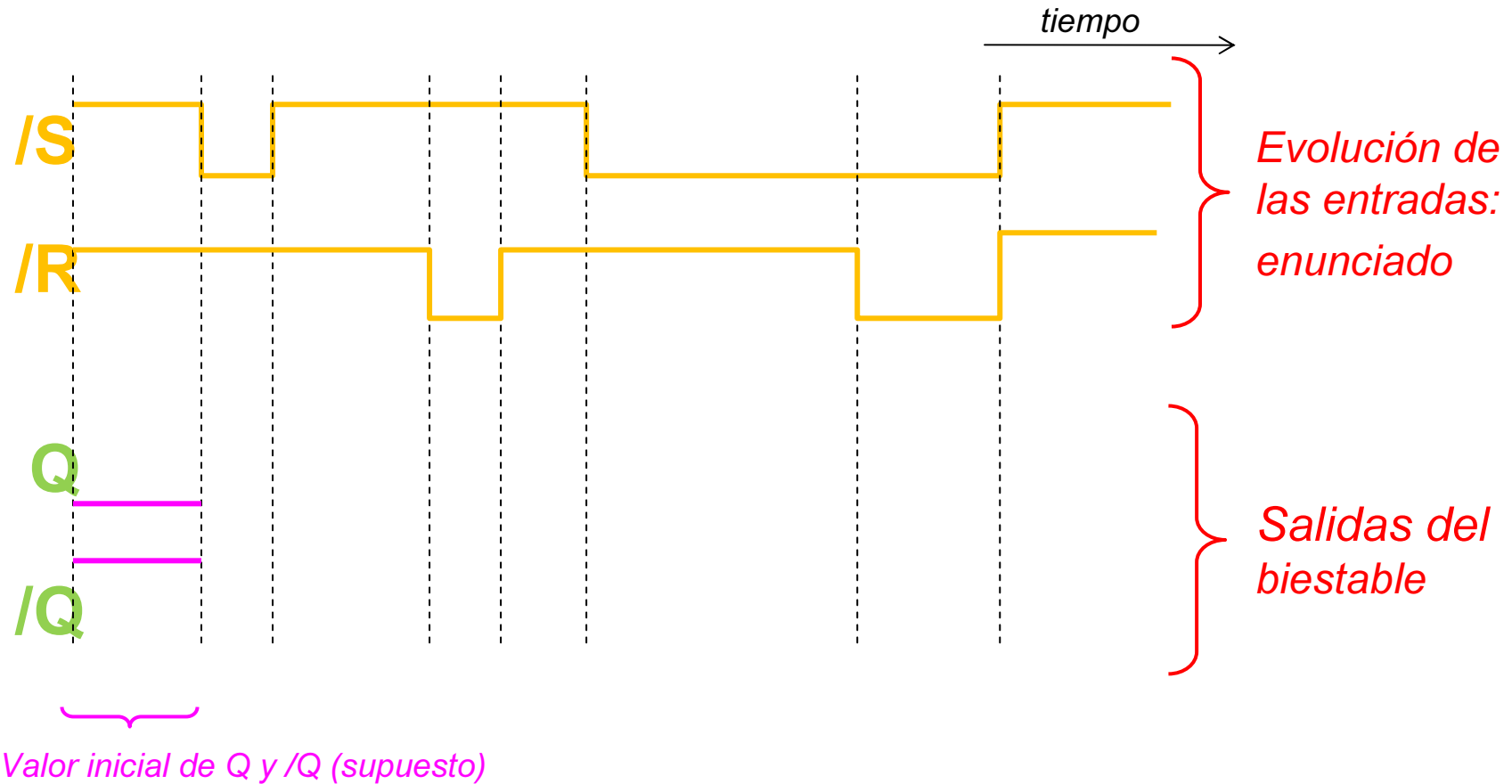
Símbolo lógico

$/S$	$/R$	$Q(t+1)$	$/Q(t+1)$
0	0	1*	1*
0	1	1	0
1	0	0	1
1	1	$Q(t)$	$/Q(t)$

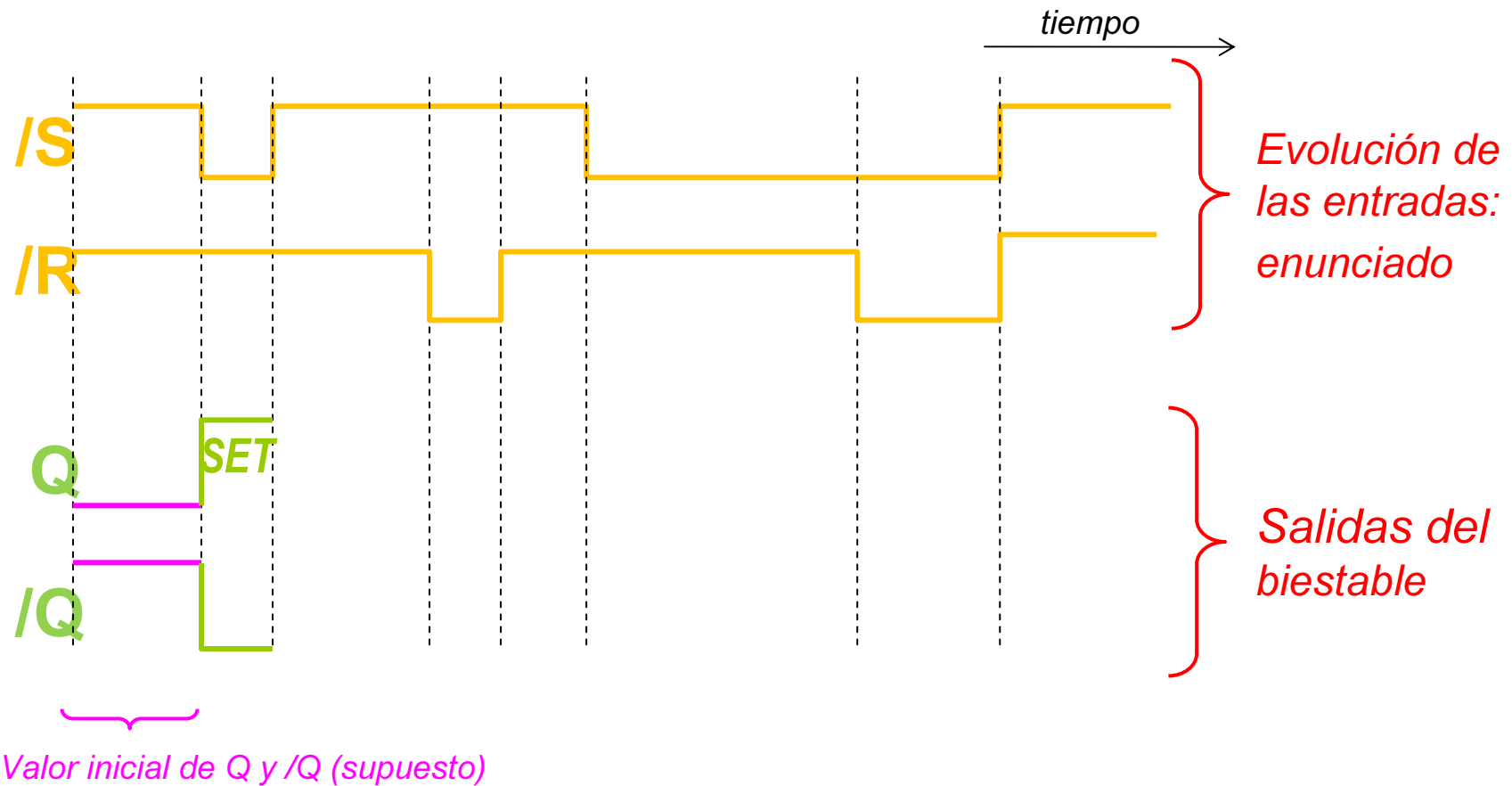
Tabla de verdad reducida

* indica situación no deseada

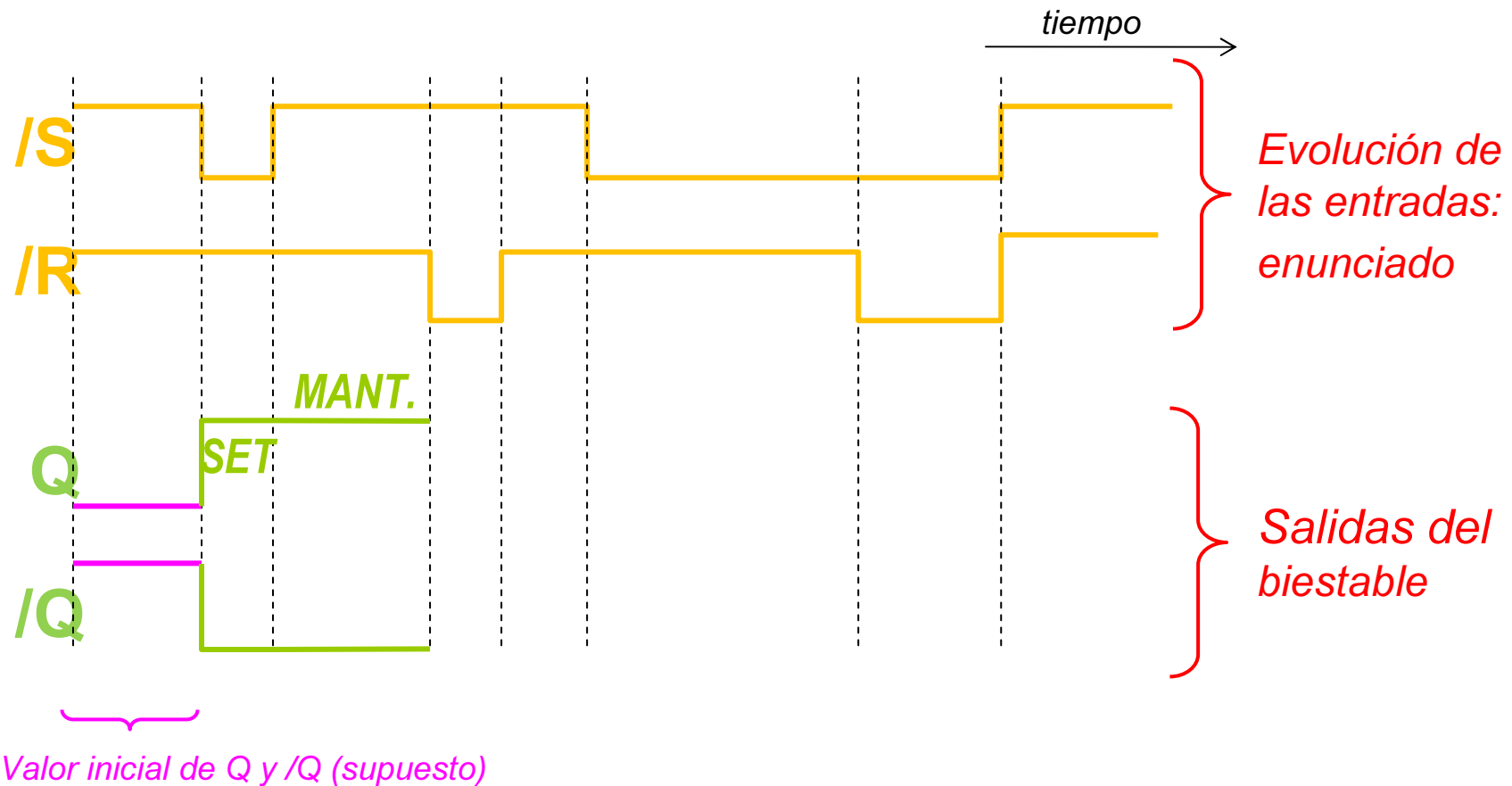
- Ejemplo de funcionamiento (cronograma)



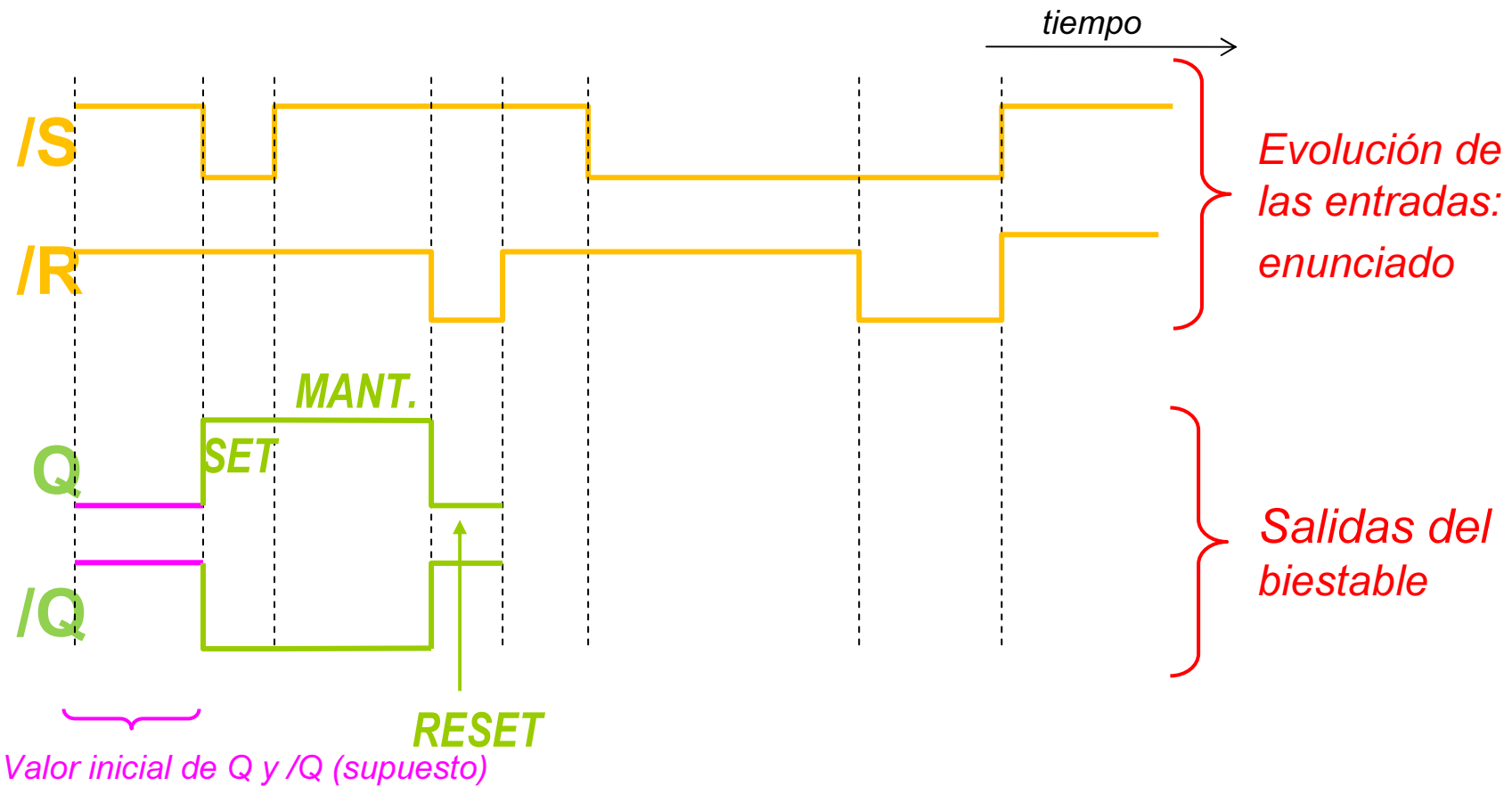
- Ejemplo de funcionamiento (cronograma)



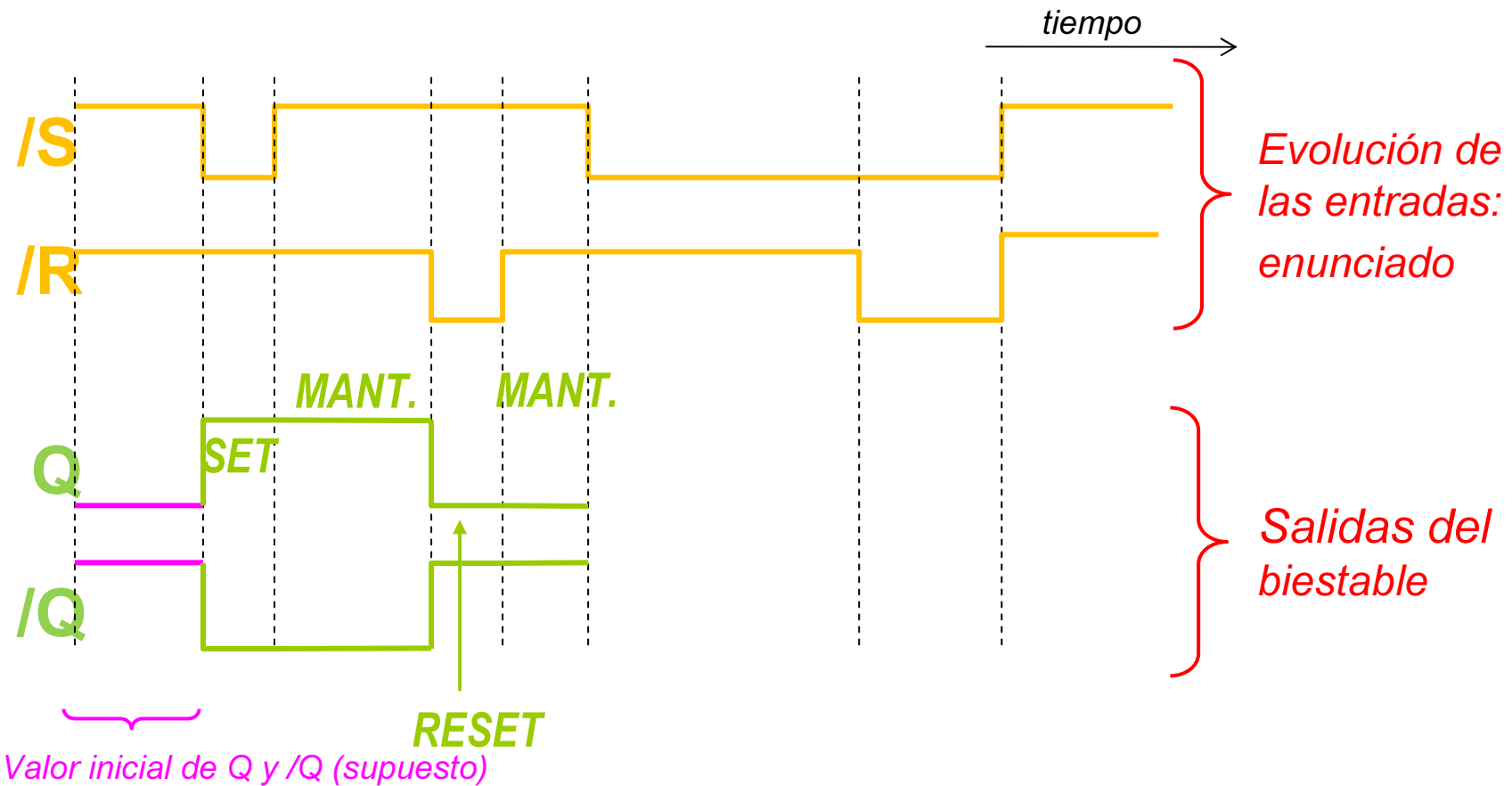
- Ejemplo de funcionamiento (cronograma)



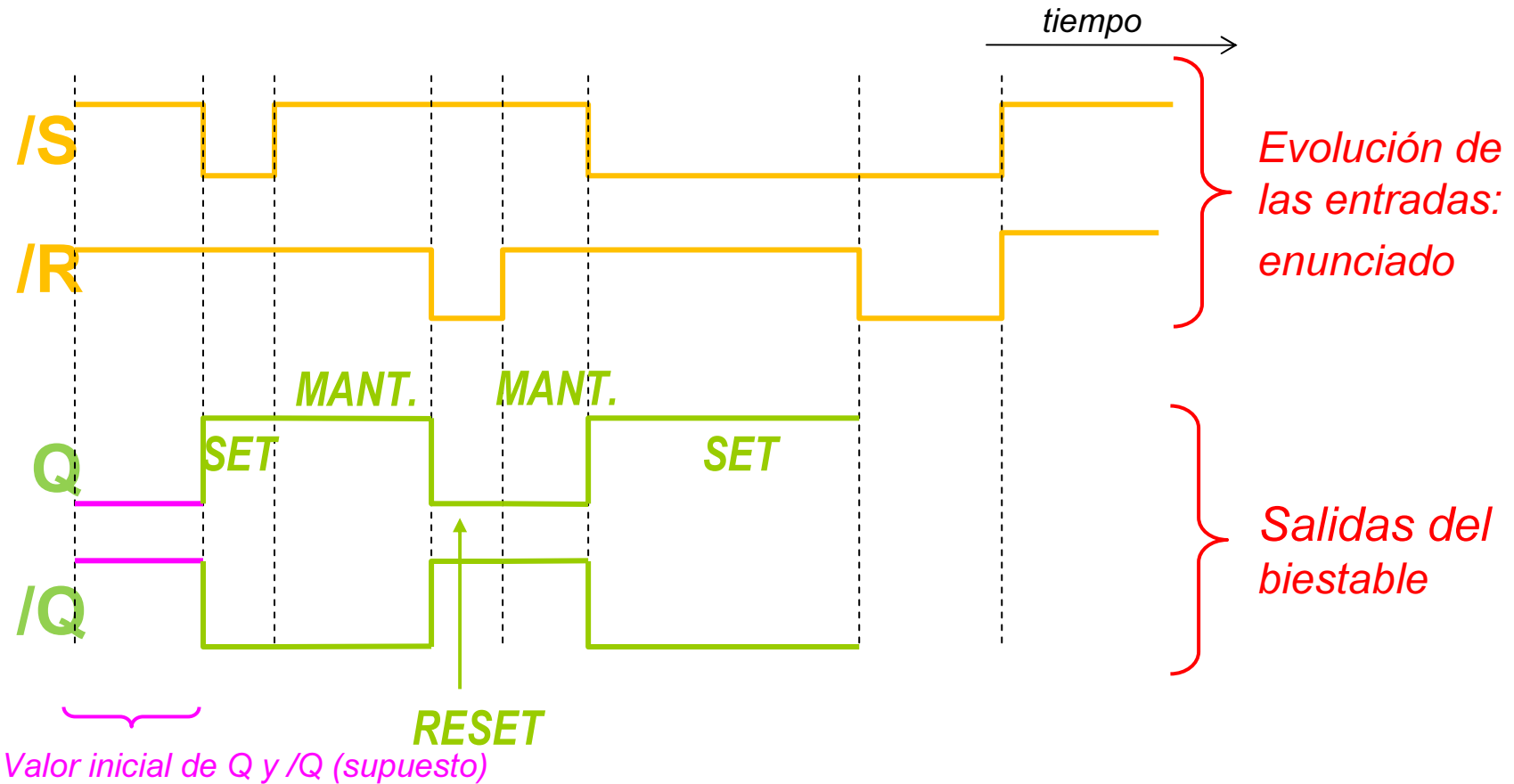
- Ejemplo de funcionamiento (cronograma)



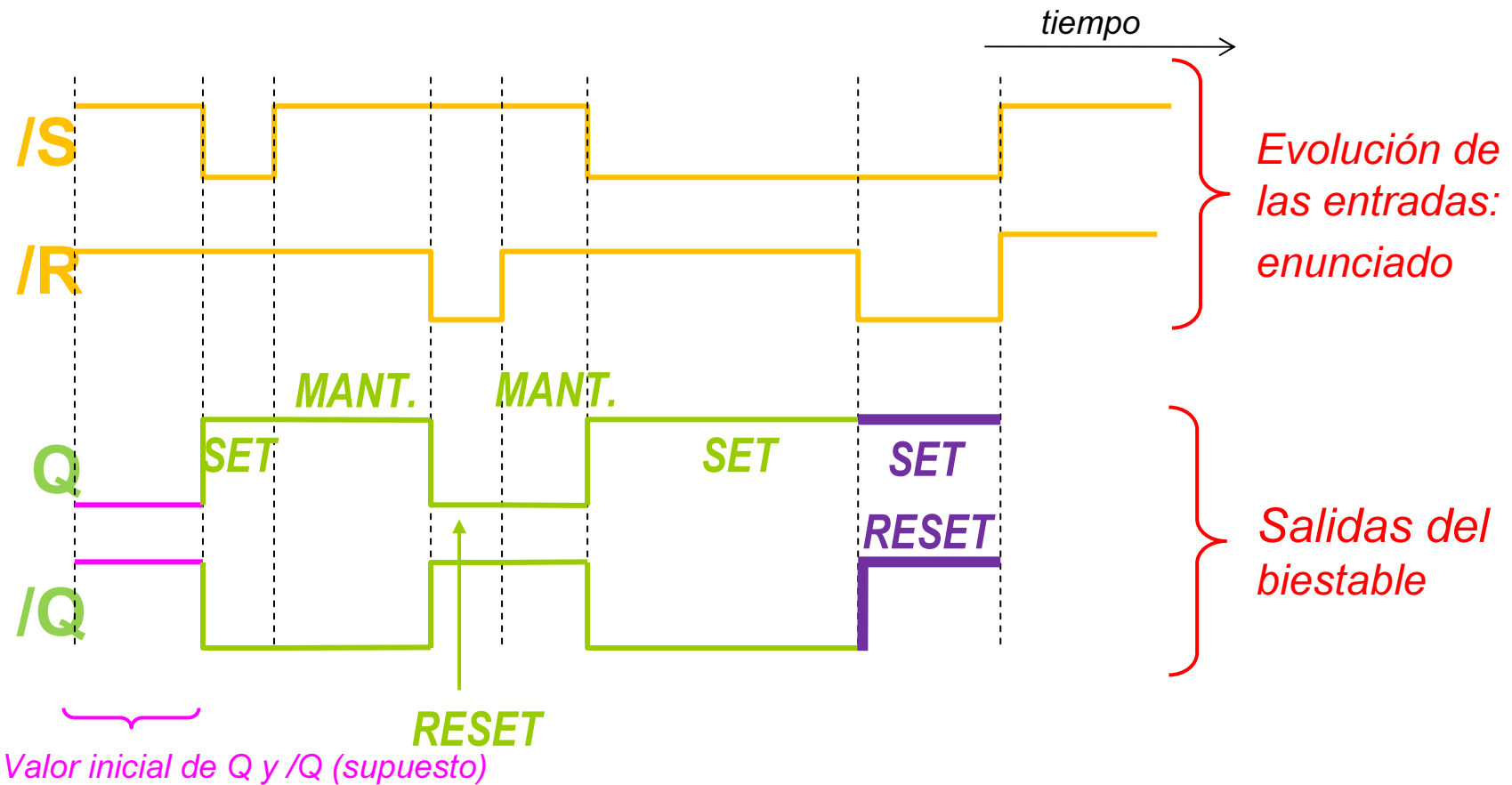
- Ejemplo de funcionamiento (cronograma)



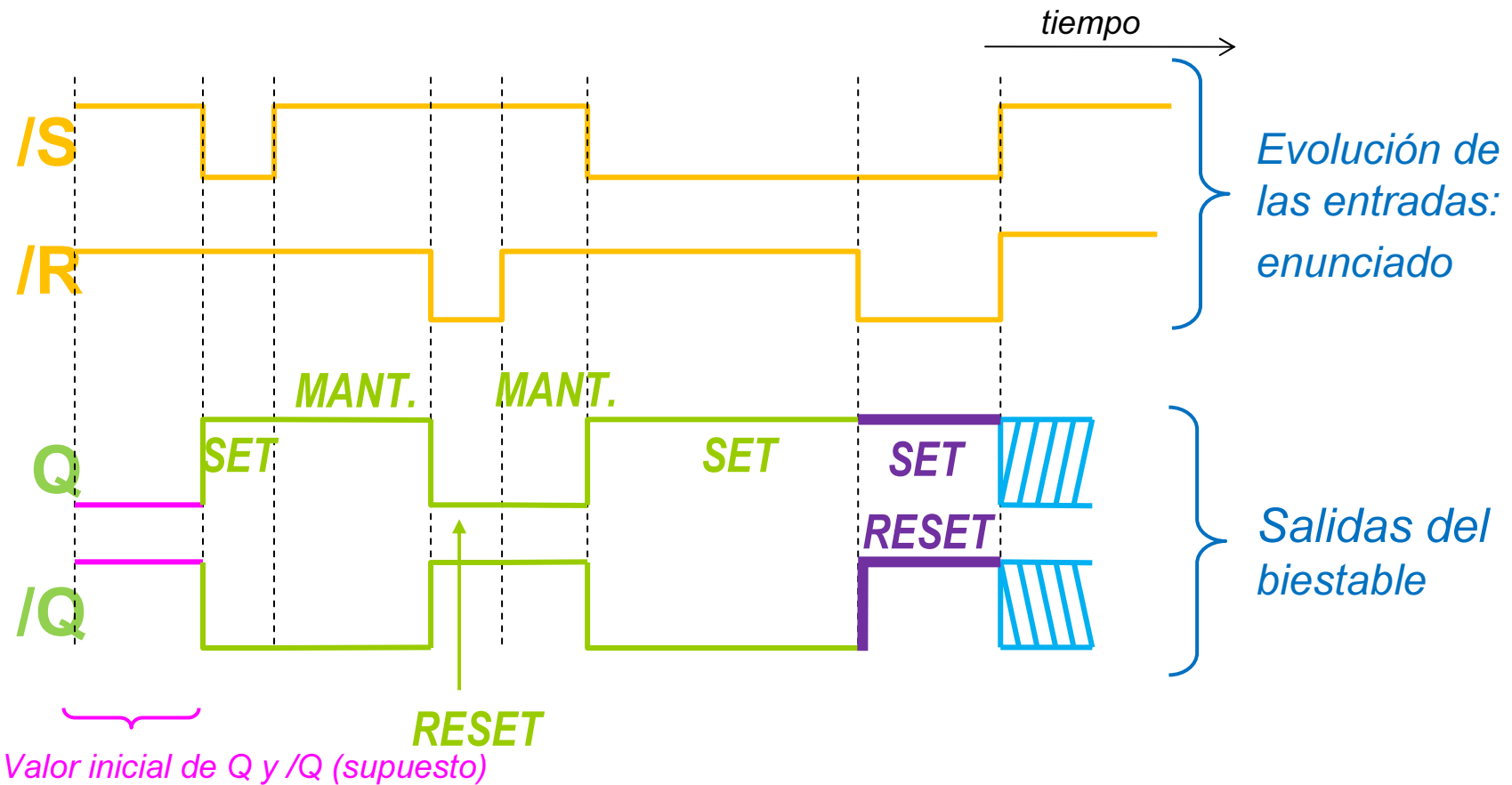
- Ejemplo de funcionamiento (cronograma)



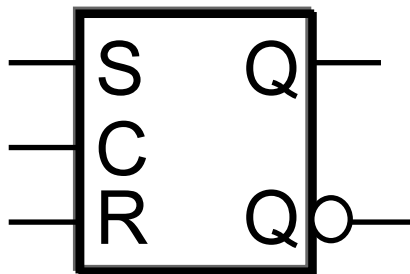
- Ejemplo de funcionamiento (cronograma)



- Ejemplo de funcionamiento (cronograma)



- Bistable S-R con puertas NAND y entrada de habilitación



Símbolo lógico

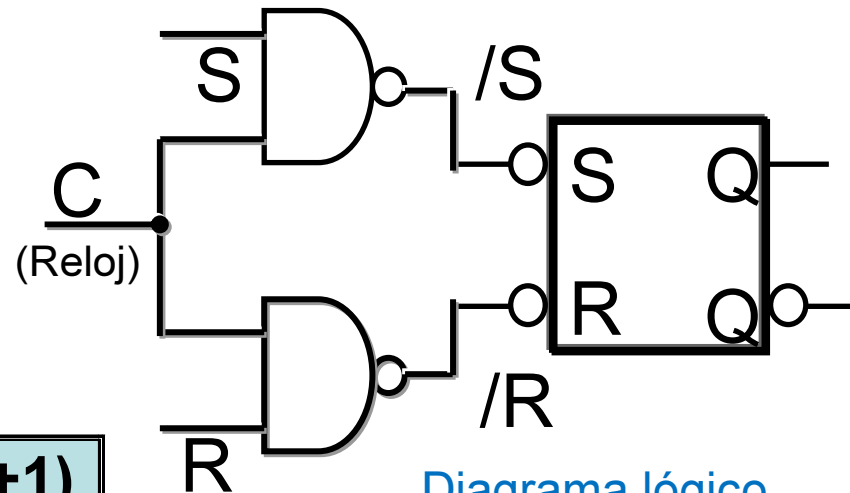


Diagrama lógico

C	S	R	/S	/R	Q(t+1)	/Q(t+1)
1	0	0	1	1	Q(t)	/Q(t)
1	0	1	1	0	0	1
1	1	0	0	1	1	0
1	1	1	0	0	1*	1*
0	X	X	1	1	Q(t)	/Q(t)

Tabla de verdad reducida * indica situación no deseada

Este caso es 1* y no 0* porque las puertas son NAND. Si utilizáramos puertas NOR en el lazo de realimentación la condición sería 0*

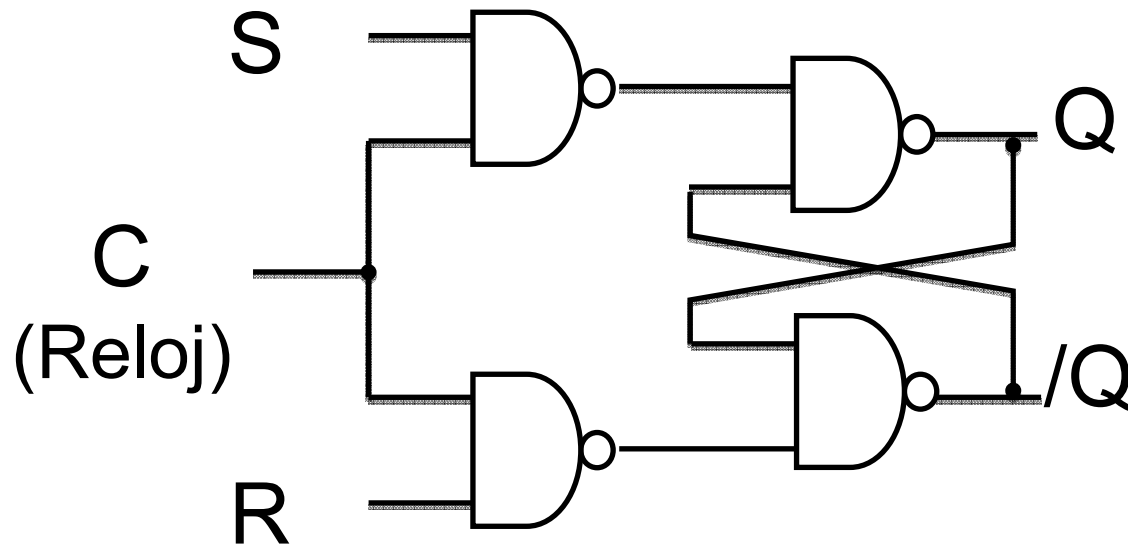
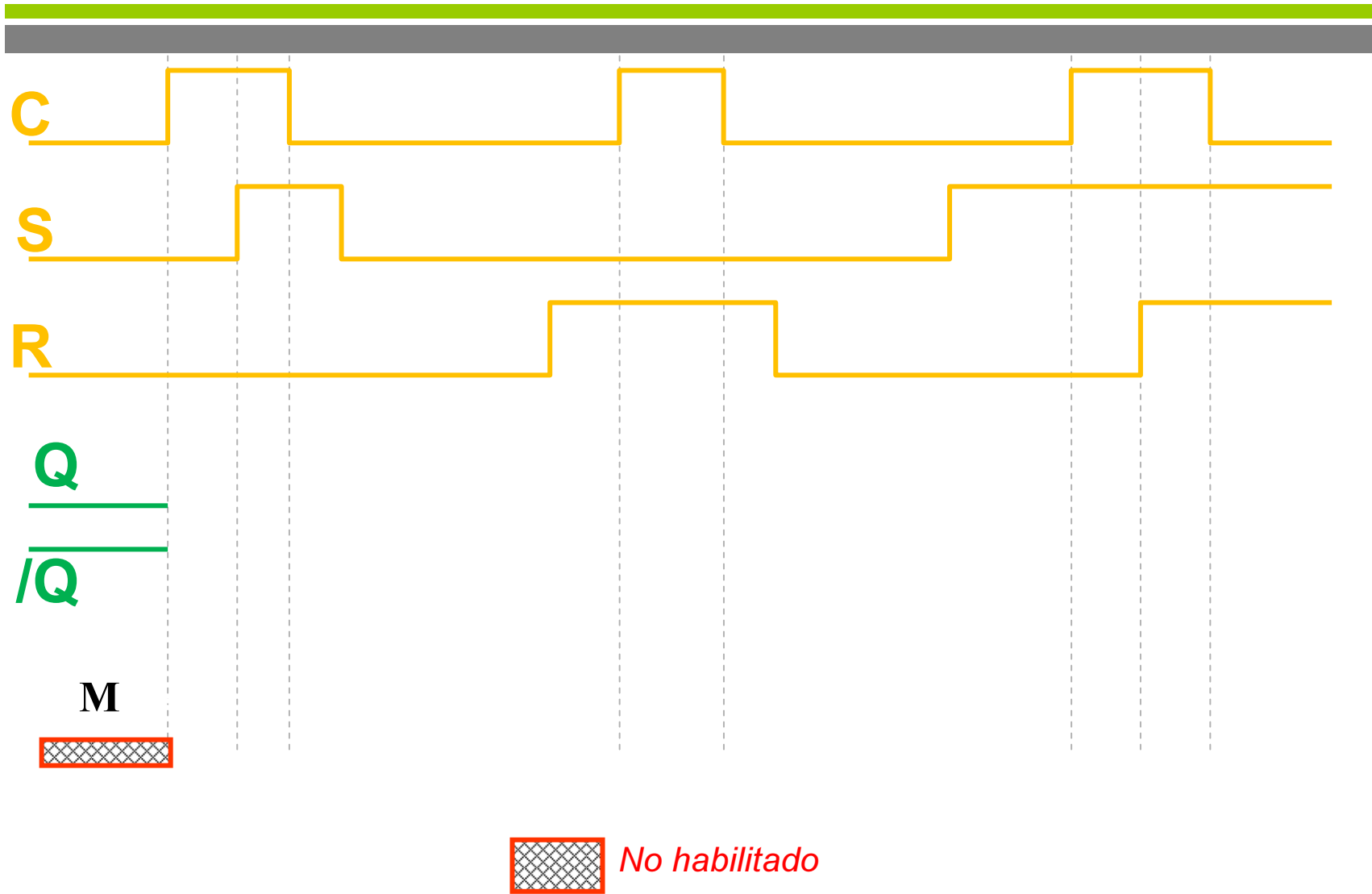
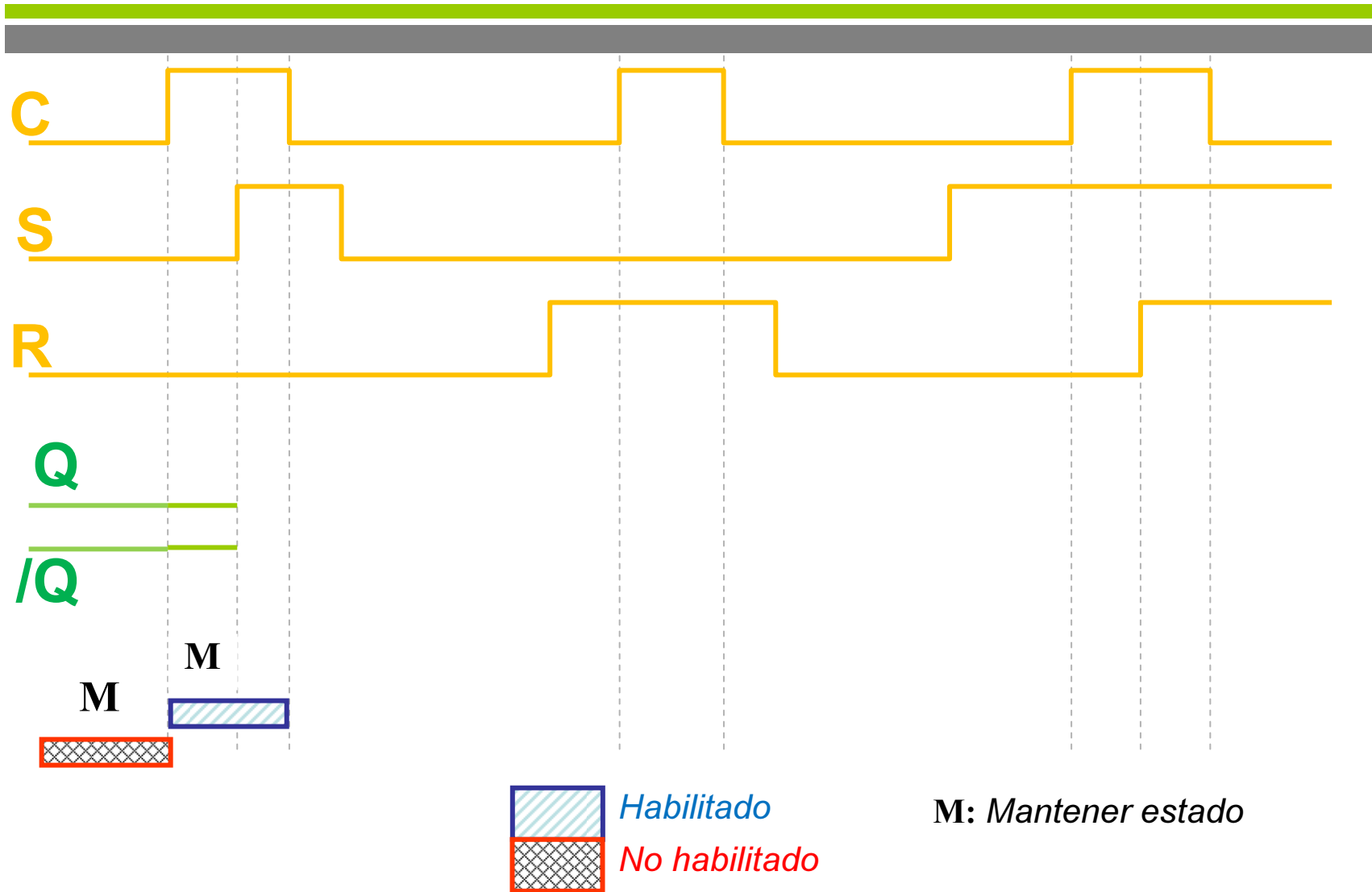


Diagrama lógico: detalle

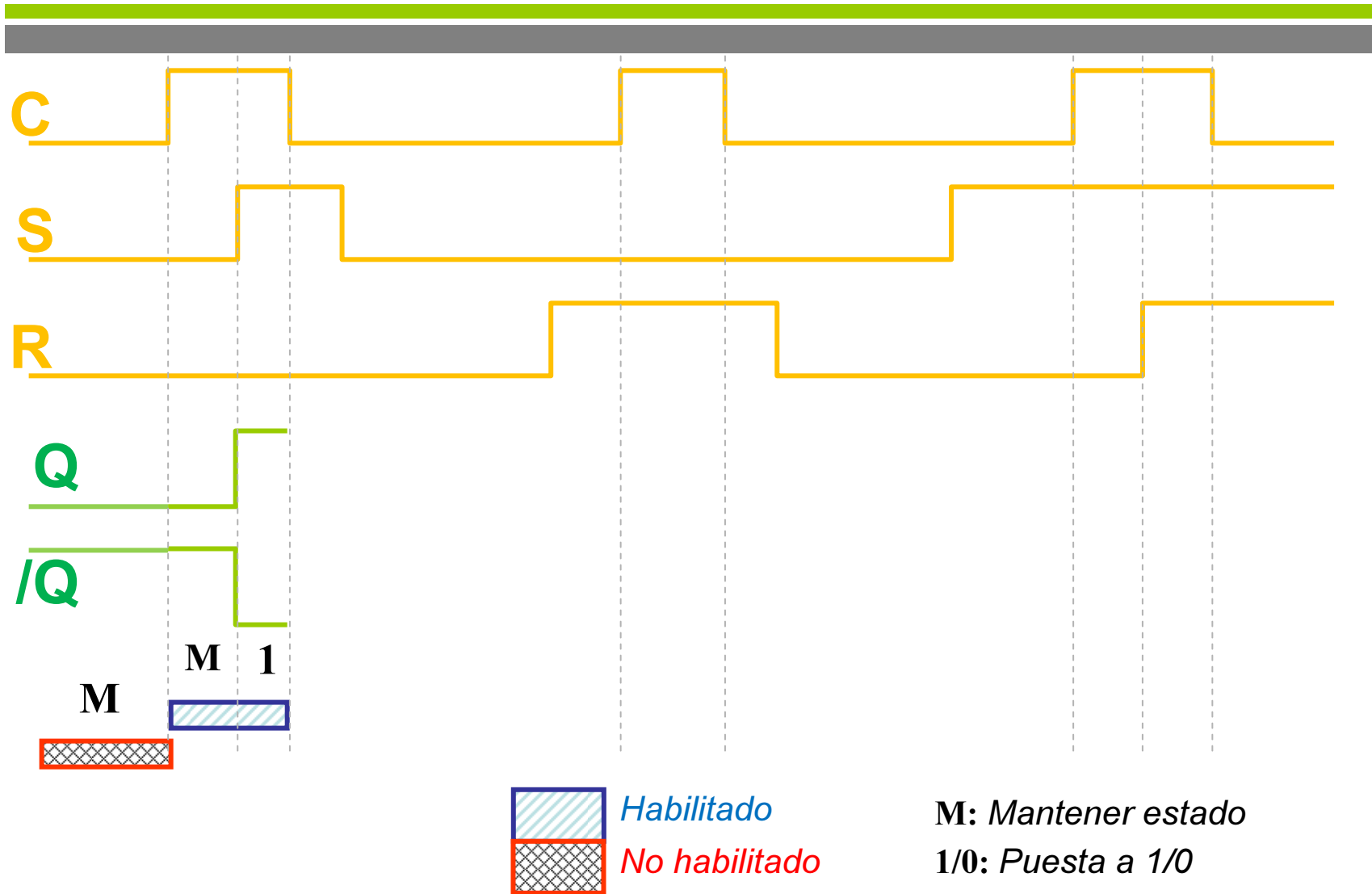
Entrada de habilitación (iii)



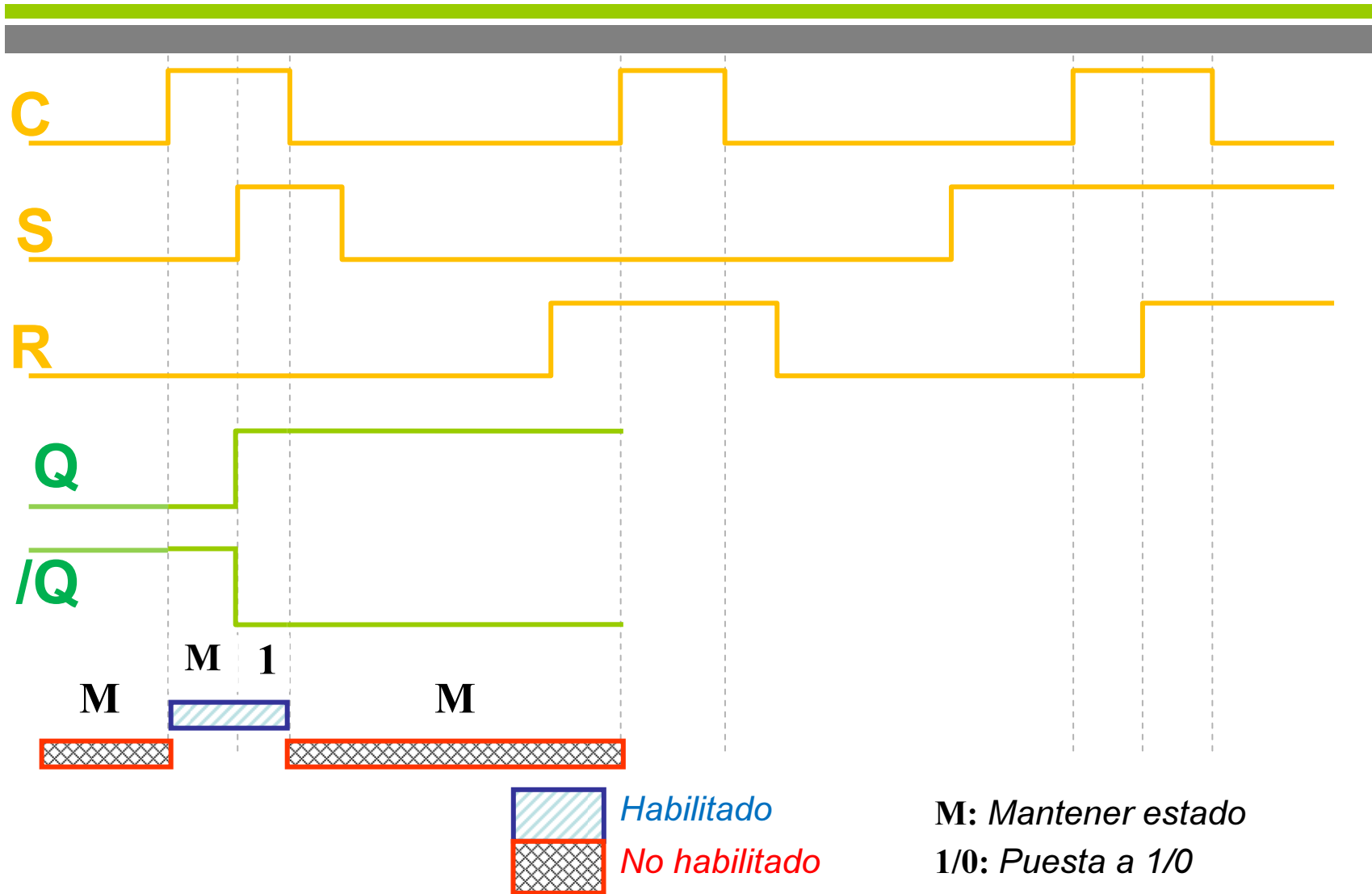
Entrada de habilitación (iv)



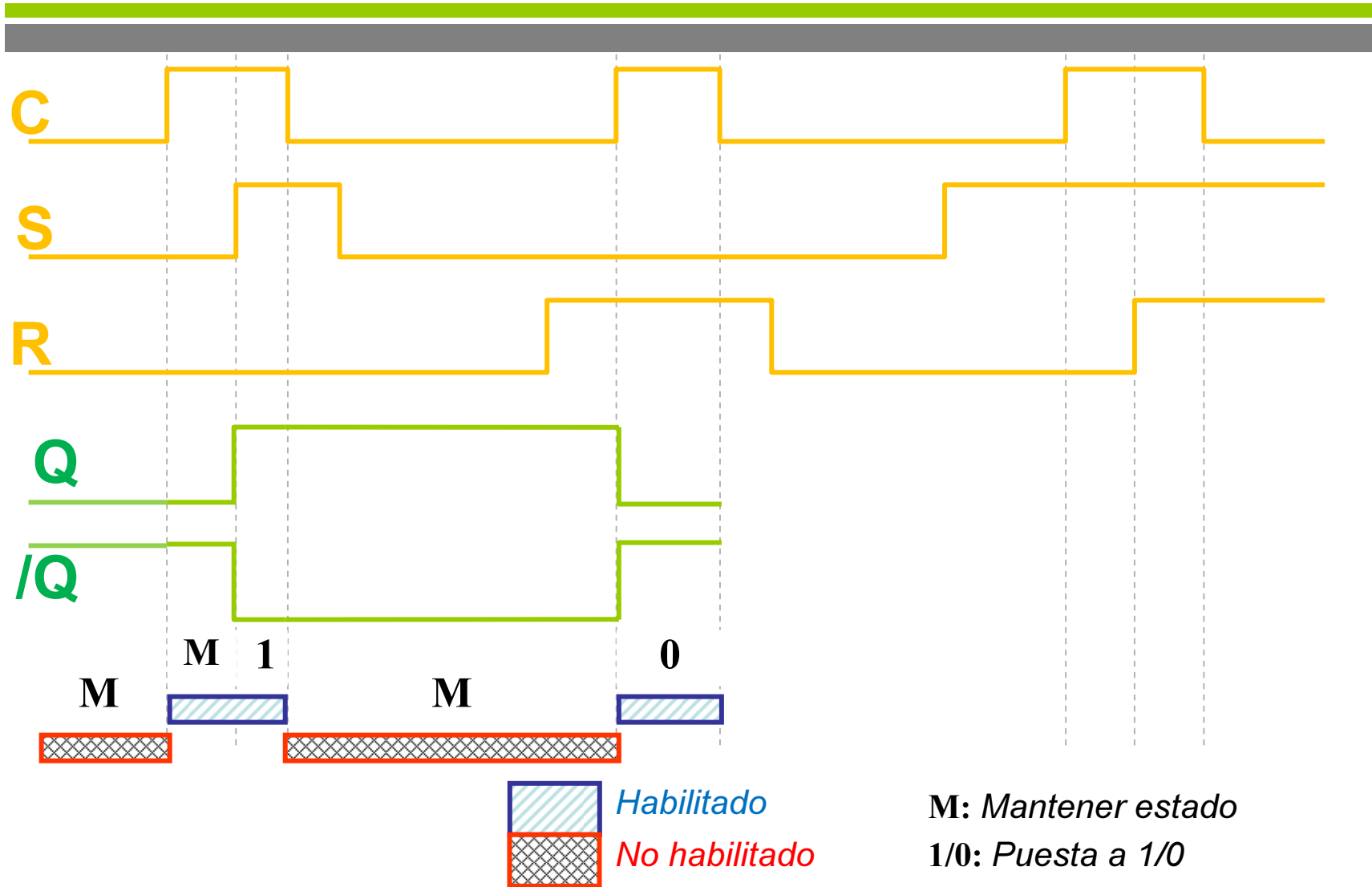
Entrada de habilitación (v)



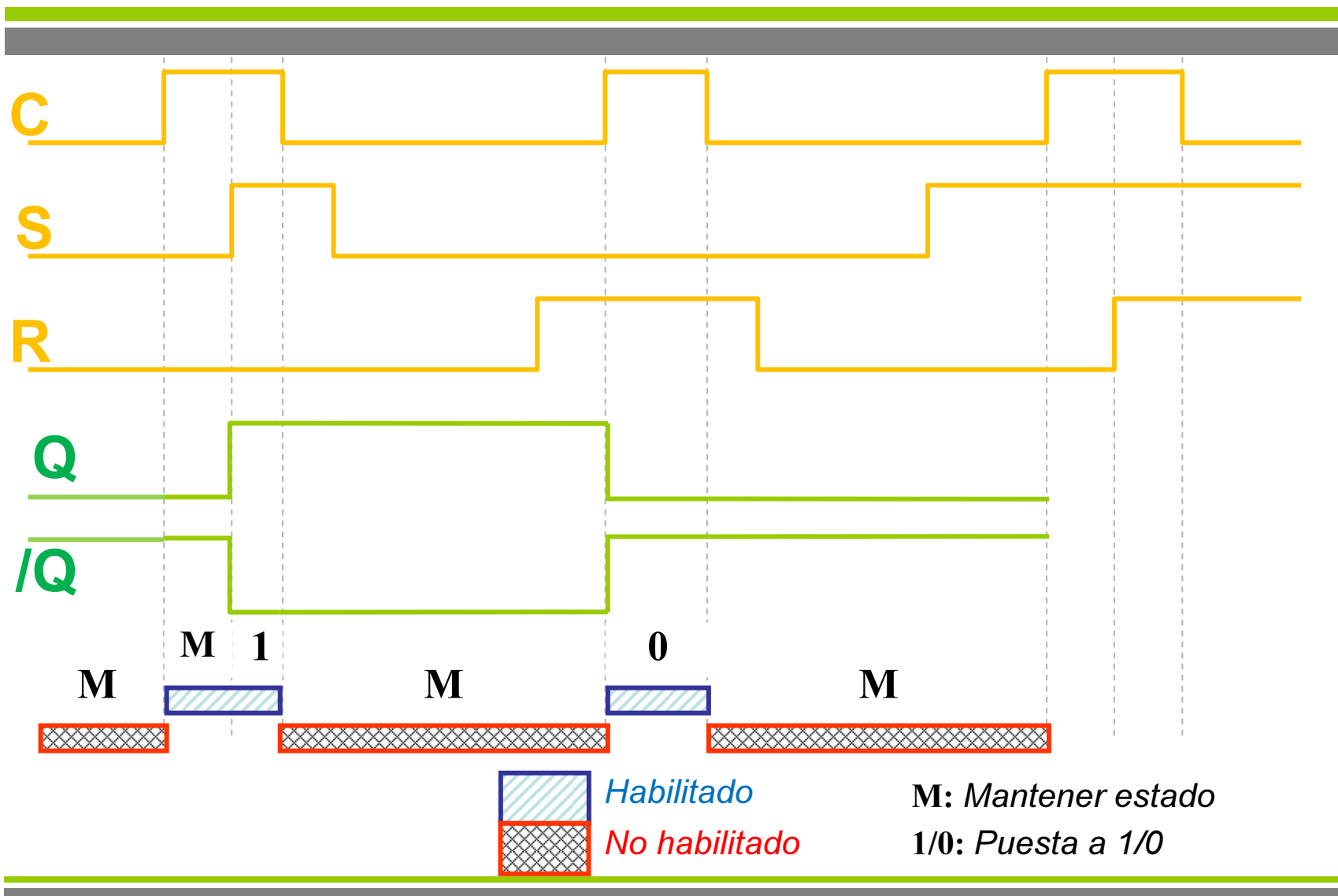
Entrada de habilitación (vi)



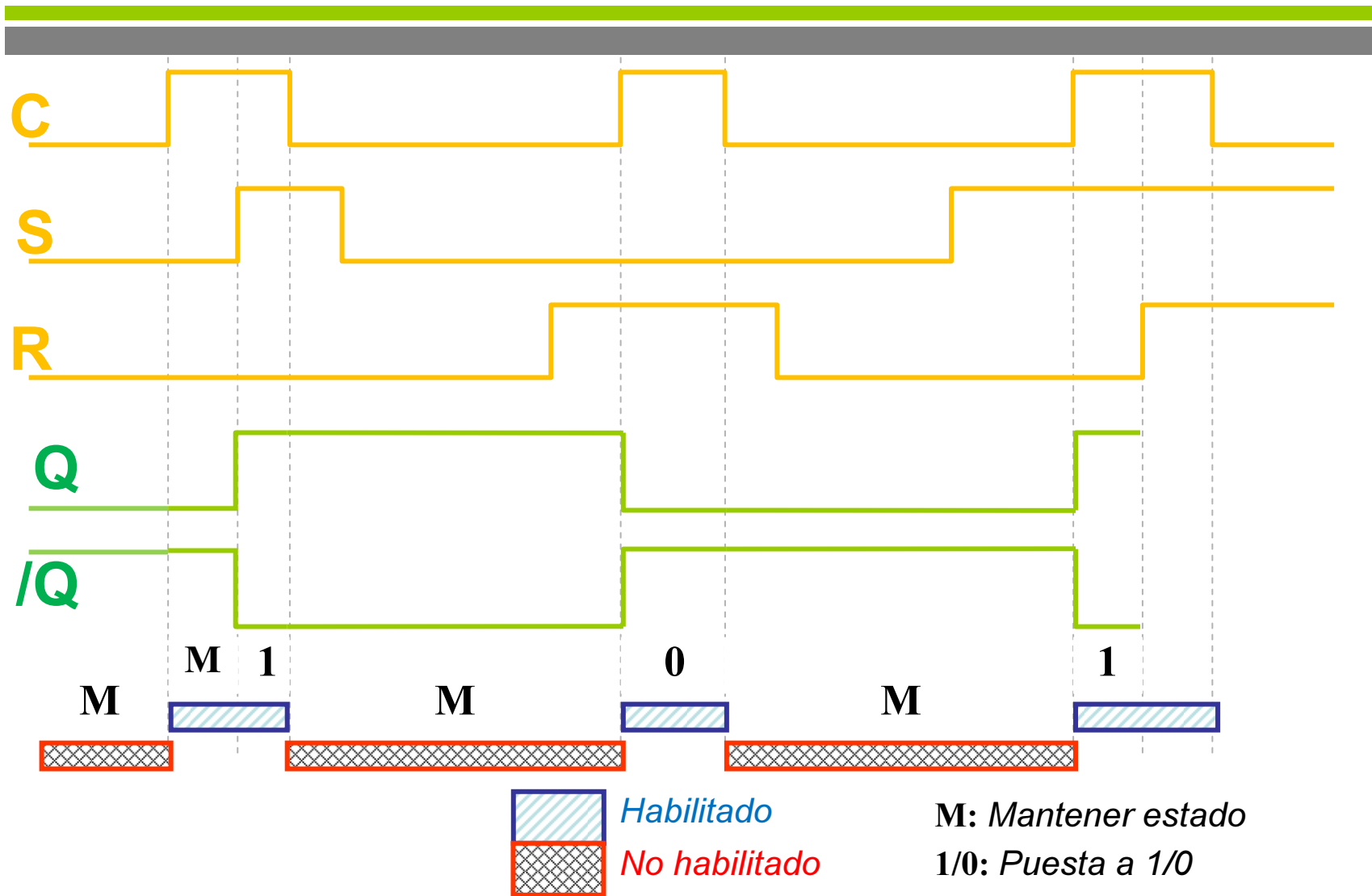
Entrada de habilitación (vii)



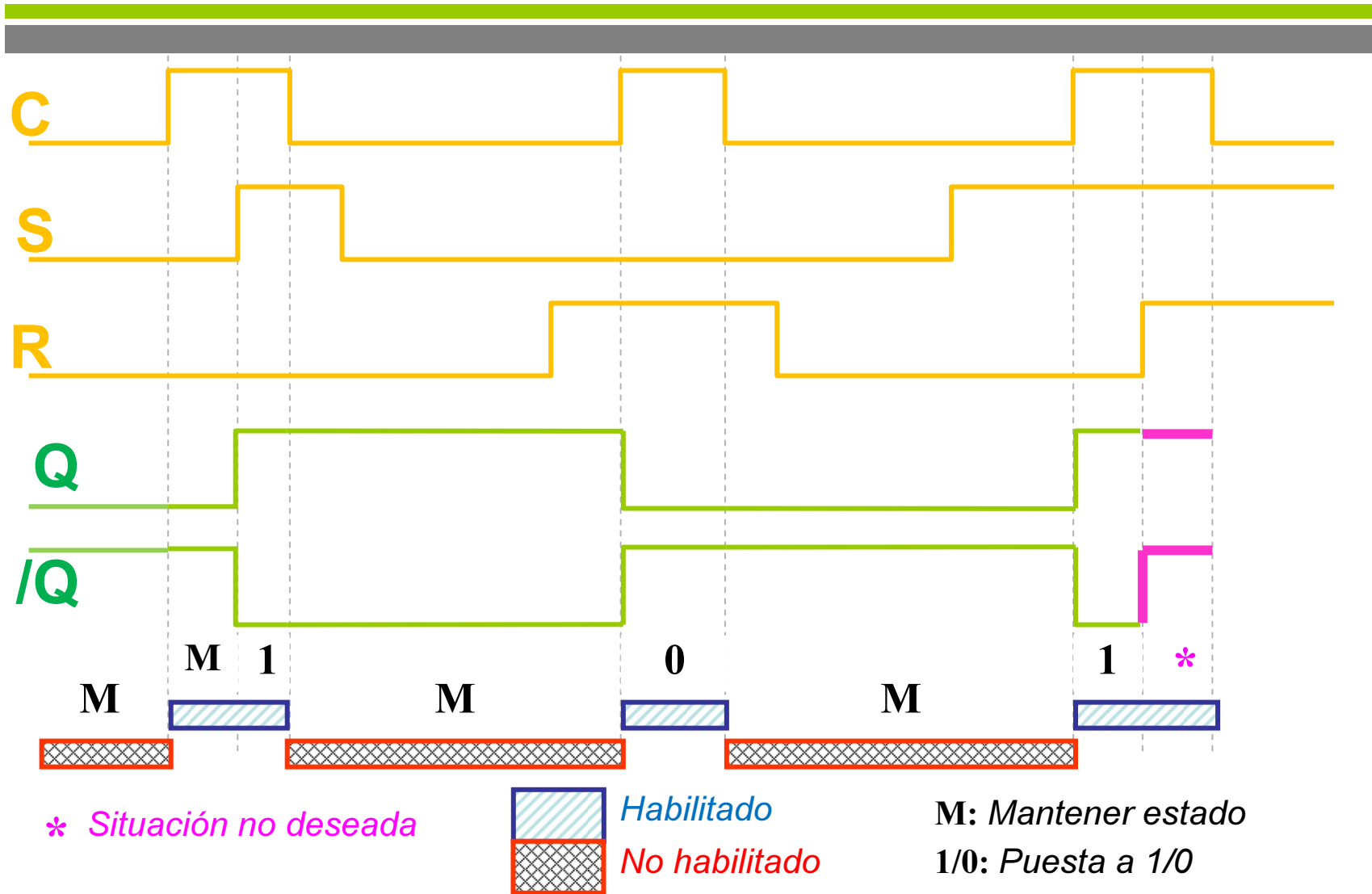
Entrada de habilitación (viii)



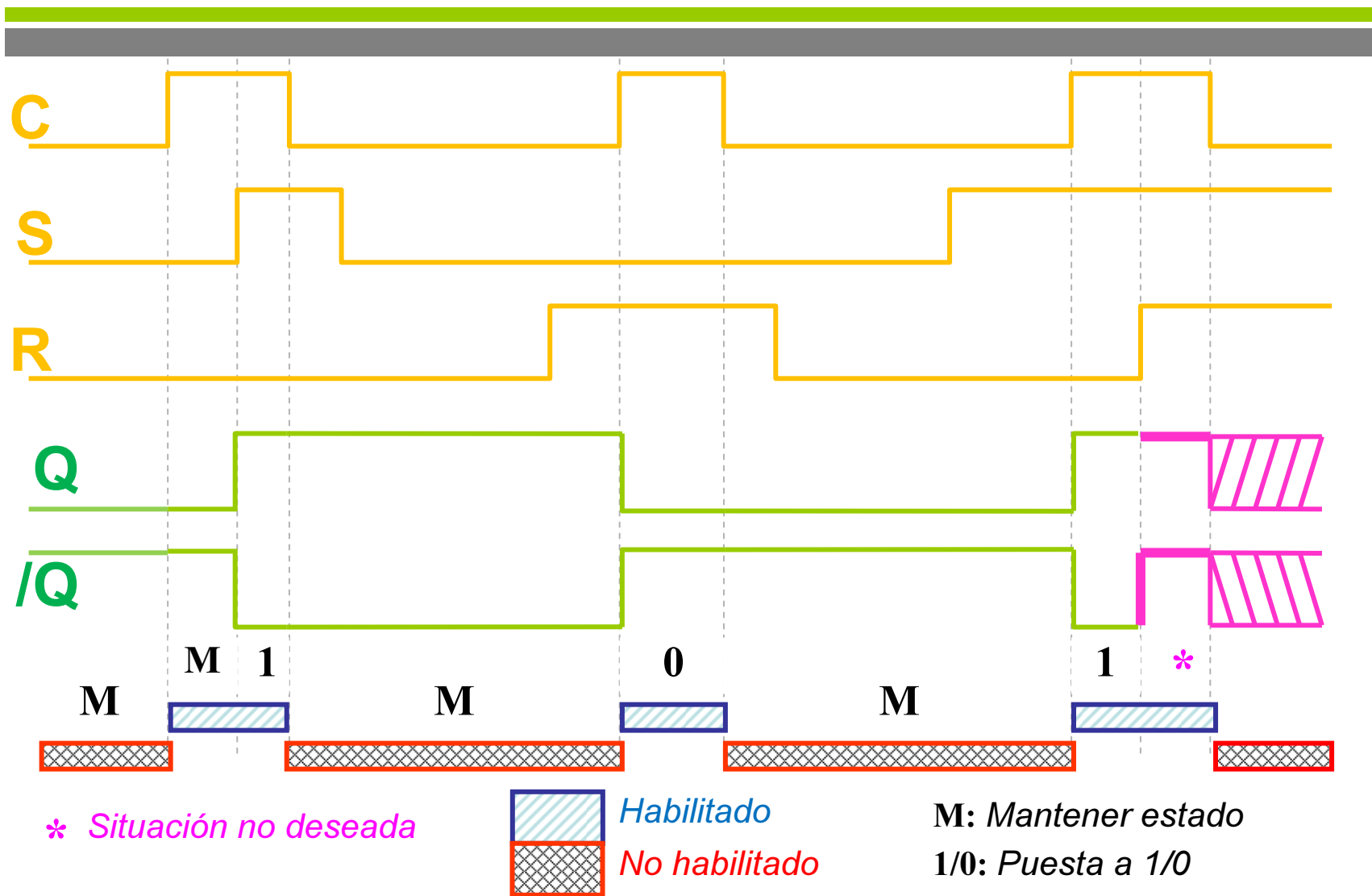
Entrada de habilitación (ix)



Entrada de habilitación (x)

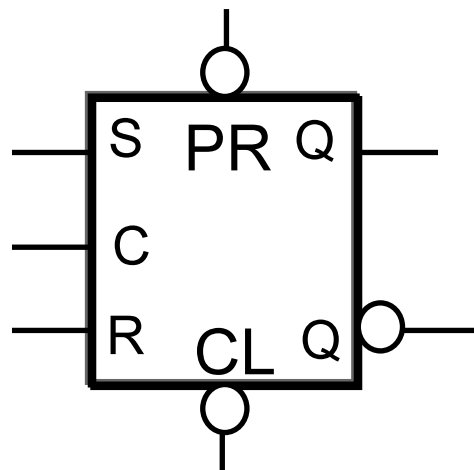


Entrada de habilitación (xi)



- Biestable S-R con habilitación y con entradas asíncronas
 - /CLEAR: puesta a cero asíncrona
 - /PRESET: puesta a uno asíncrona

Tienen prioridad sobre la señal de reloj y permiten poner el estado a uno o a cero



Símbolo lógico

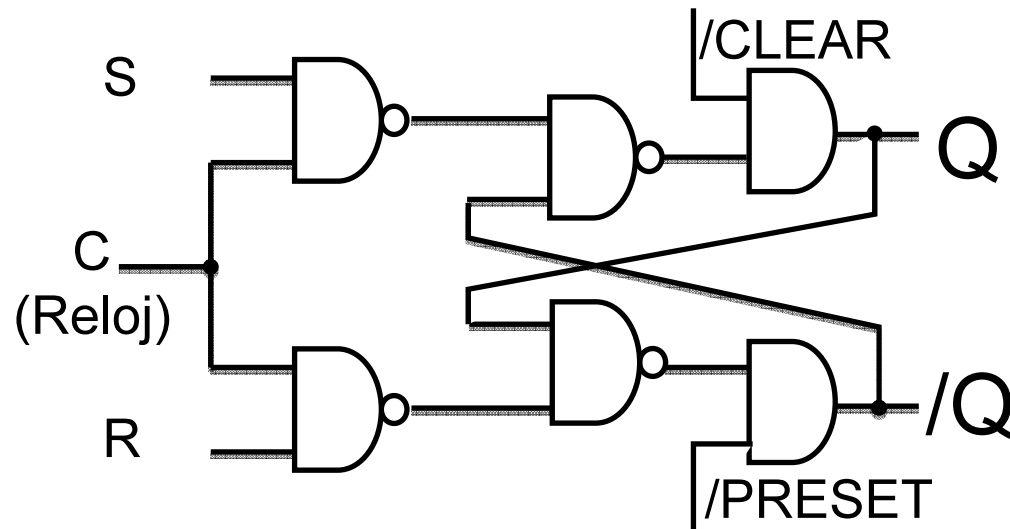
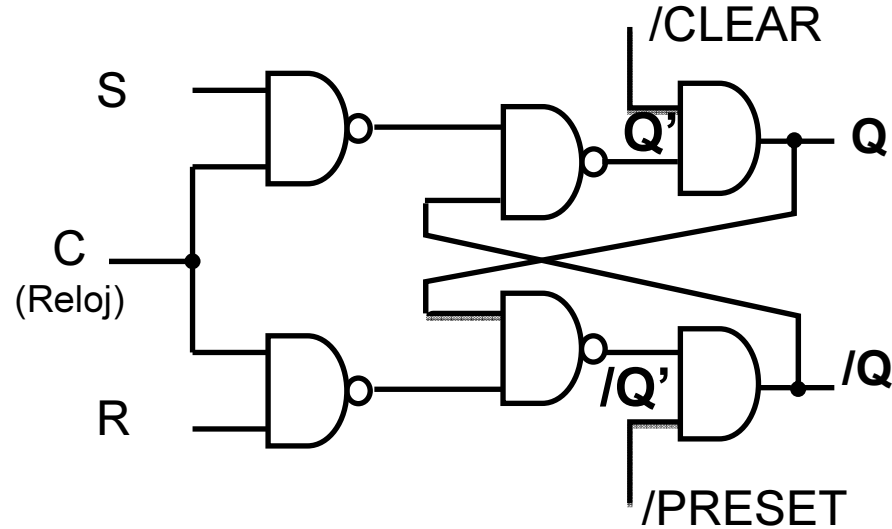


Diagrama lógico



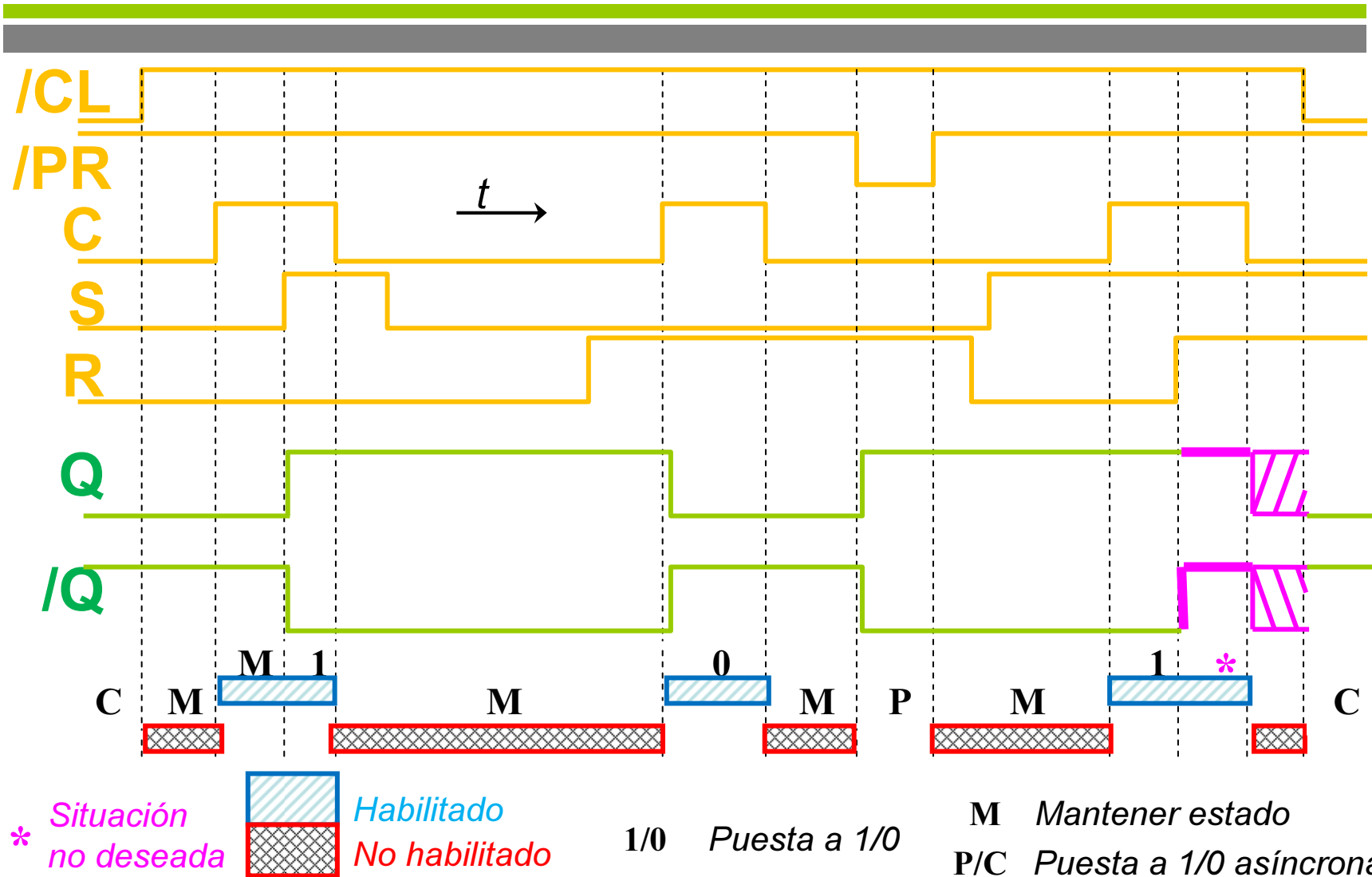
- Si $\text{/CLEAR} = \text{/PRESET} = 1 \Rightarrow Q = Q', \text{/Q} = \text{/Q}'$
- Si $\text{/CLEAR} = 0, \text{/PRESET} = 1 \Rightarrow Q = 0, \text{/Q}' = 1, \text{/Q} = 1$
- Si $\text{/CLEAR} = 1, \text{/PRESET} = 0 \Rightarrow \text{/Q} = 0, Q' = 1, Q = 1$
- Si $\text{/CLEAR} = 0, \text{/PRESET} = 0 \Rightarrow Q = \text{/Q} = 0^*, Q' = \text{/Q}' = 1^*$

- Biestable S-R con habilitación y entradas asíncronas

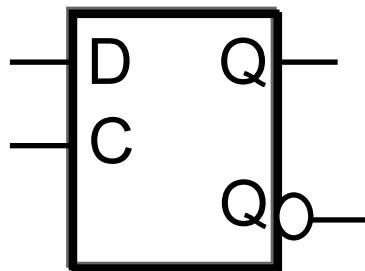
/PR	/CL	C	S	R	Q(t+1)	/Q(t+1)
0	1	X	X	X	1	0
1	0	X	X	X	0	1
0	0	X	X	X	0*	0*
1	1	1	0	0	Q(t)	/Q(t)
1	1	1	0	1	0	1
1	1	1	1	0	1	0
1	1	1	1	1	1*	1*
1	1	0	X	X	Q(t)	/Q(t)

Tabla de verdad

Entradas asíncronas (iv)



- Se utilizan para la implementación de elementos de memoria, cuya única finalidad es almacenar el valor de una línea de información (un bit)



Símbolo lógico

C	D	$Q(t+1)$	$\bar{Q}(t+1)$
0	X	$Q(t)$	$\bar{Q}(t)$
1	1	1	0
1	0	0	1

Tabla de verdad

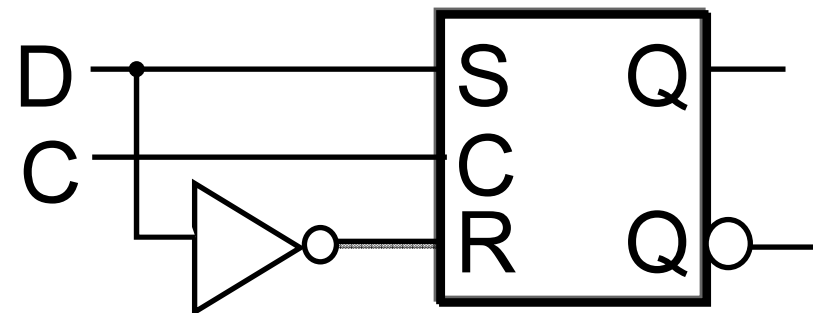
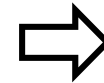
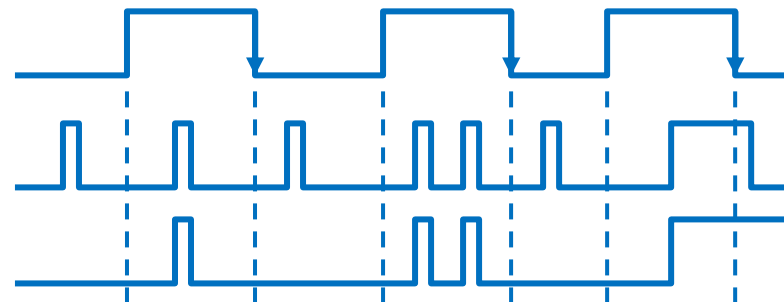


Diagrama lógico

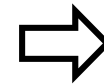
Si se activa el biestable por nivel y hay un pulso no deseado en la entrada de datos



el pulso no deseado se trasladará a la salida

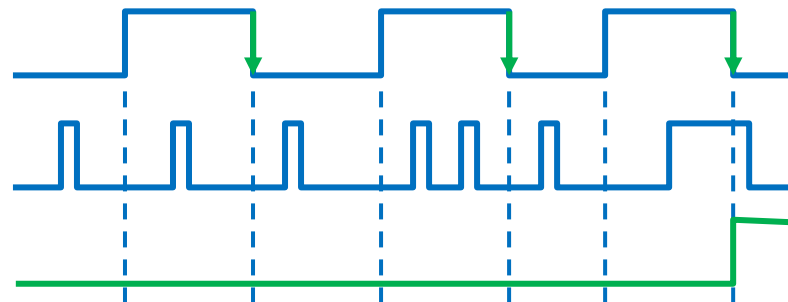


Si se activa el biestable por nivel y hay un pulso no deseado en la entrada de datos

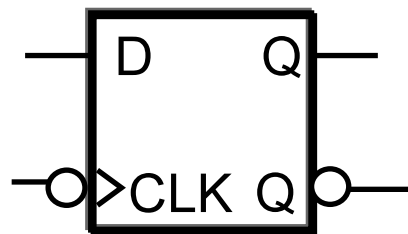


el pulso no deseado se trasladará a la salida

Podemos hacer que la señal de reloj actúe por flanco



- Para implementar biestables que se activen por flanco se utiliza con frecuencia un par de biestables por nivel en configuración MASTER-SLAVE (maestro-esclavo)



Símbolo lógico

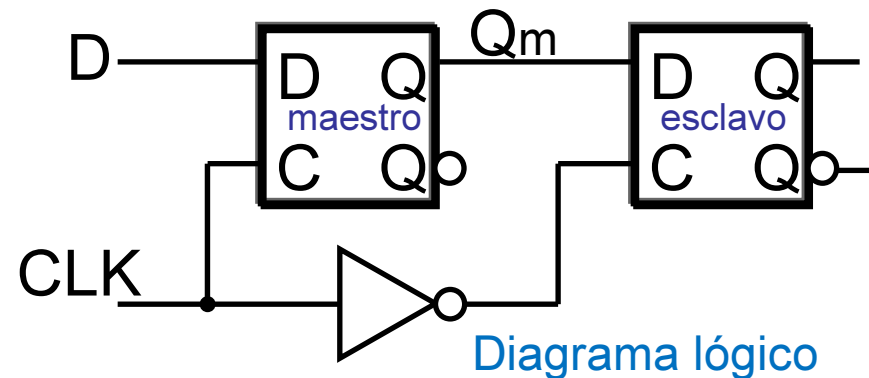
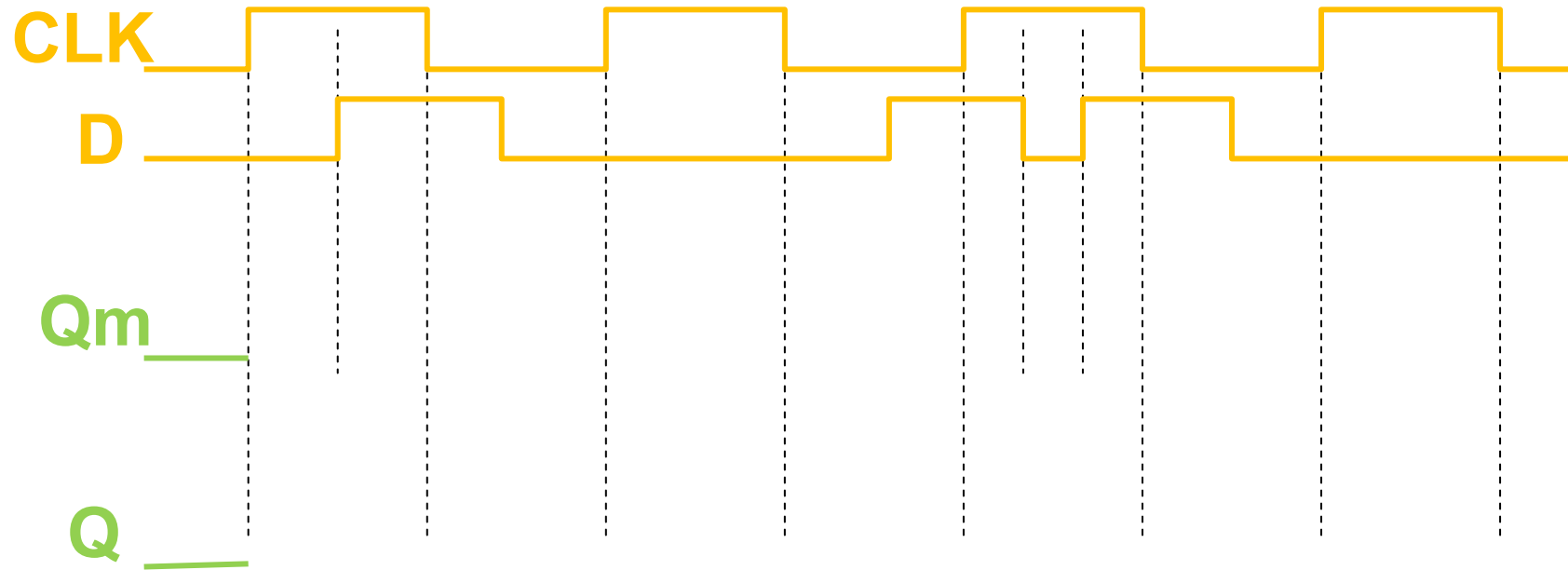


Diagrama lógico

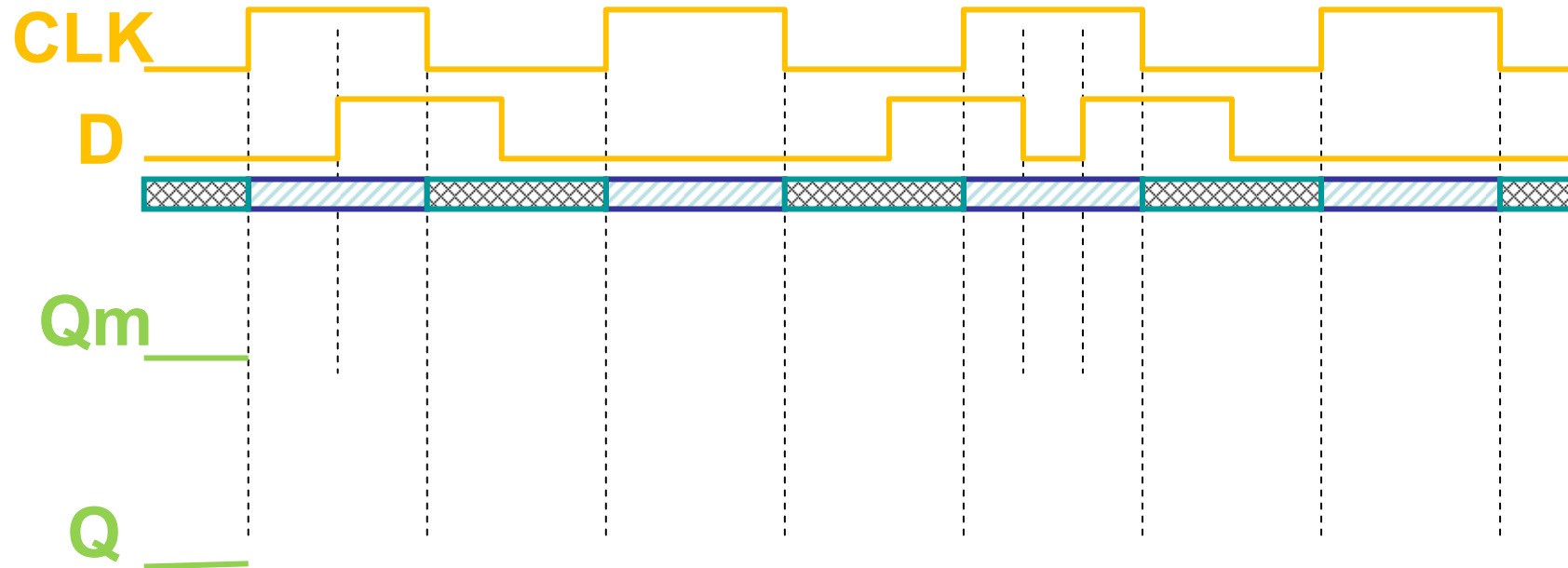
CLK	D	Q(t+1)	/Q(t+1)
0	X	Q(t)	/Q(t)
1	X	Q(t)	/Q(t)
↓	1	1	0
↓	0	0	1

Tabla de verdad

Biestable D por flanco de bajada (ii)



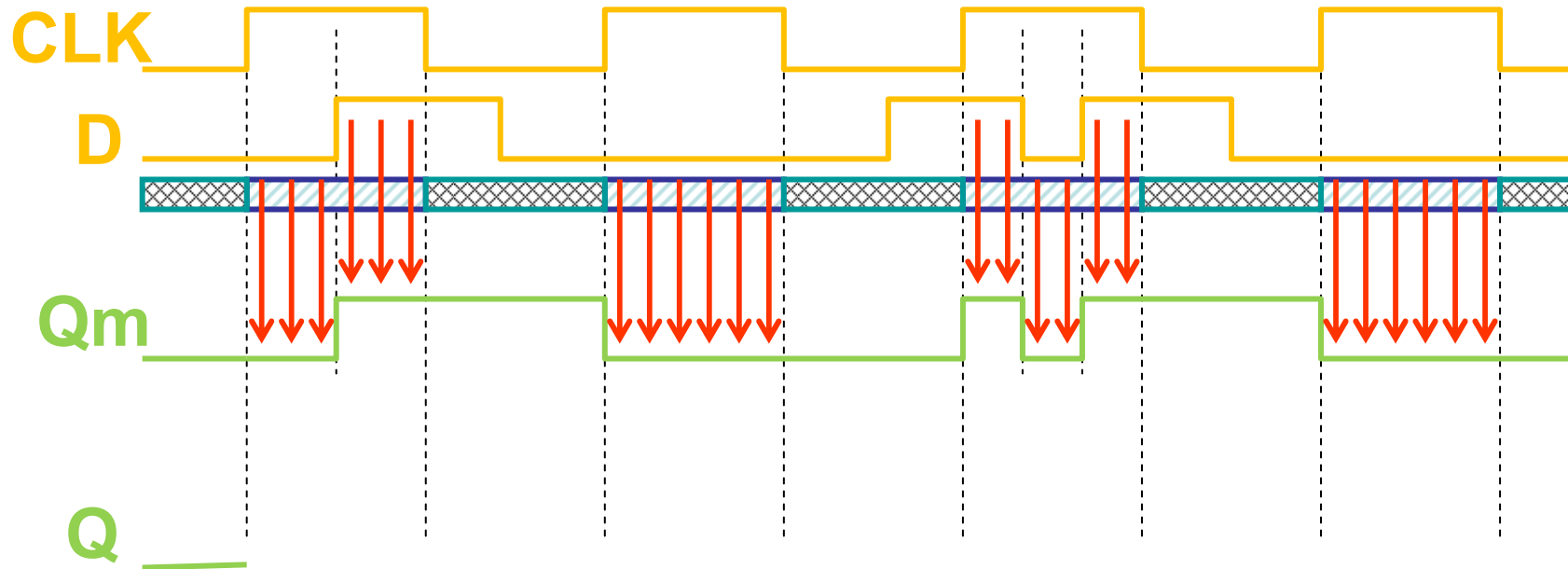
Biastable D por flanco de bajada (iii)



 *Maestro habilitado, $Q_m = D$*

 *Maestro no habilitado*

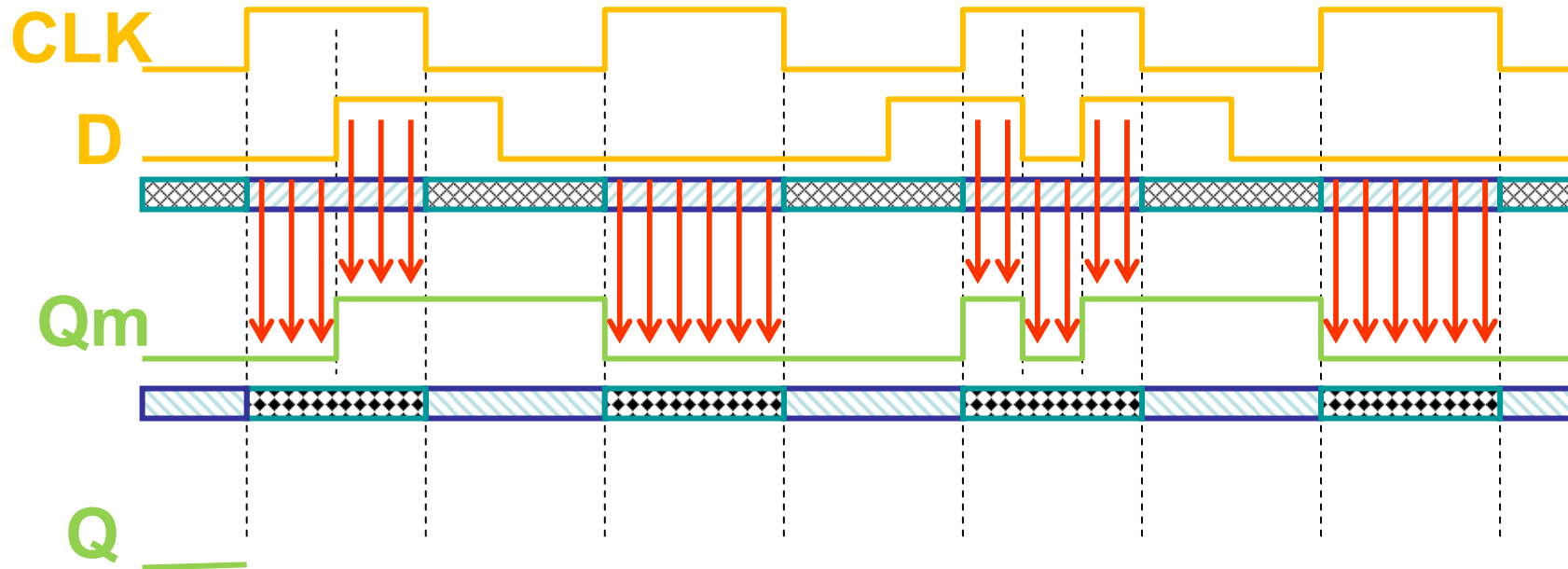
Bi stable D por flanco de bajada (iv)



 *Maestro habilitado, $Q_m=D$*

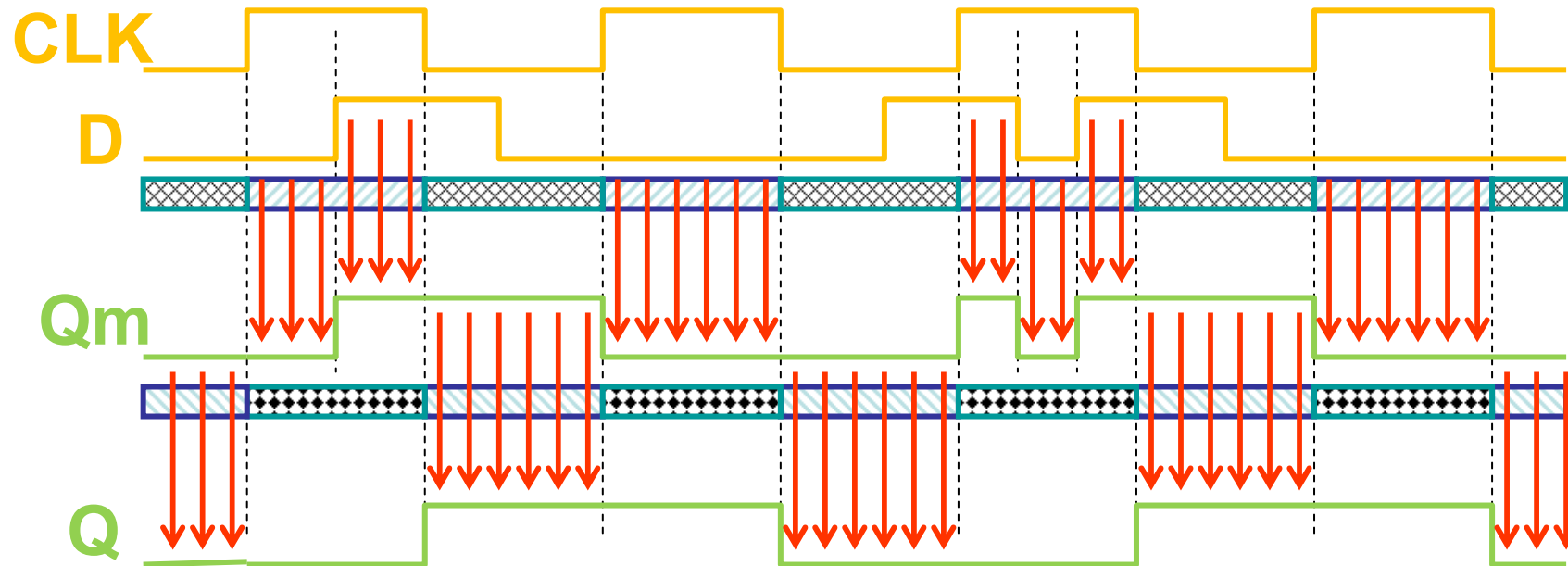
 *Maestro no habilitado*

Bi stable D por flanco de bajada (v)



- | | |
|--|--|
|  <i>Maestro habilitado, $Qm=D$</i> |  <i>Esclavo habilitado, $Q=Qm$</i> |
|  <i>Maestro no habilitado</i> |  <i>Esclavo no habilitado</i> |

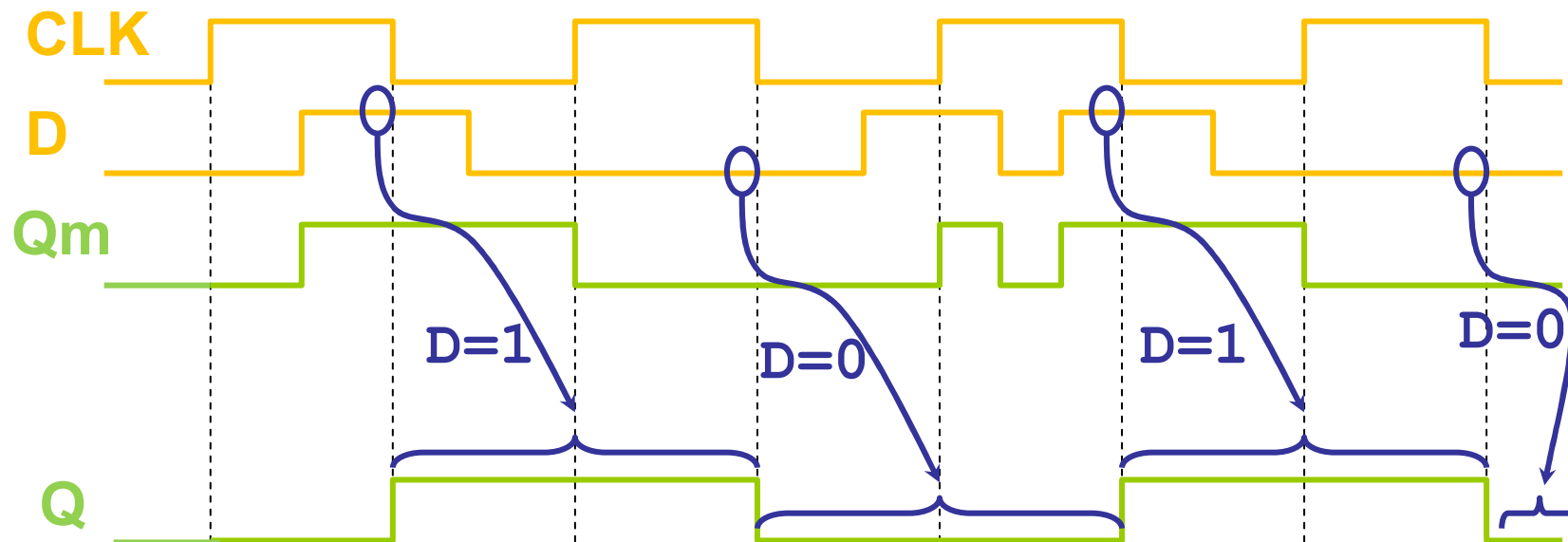
Bi stable D por flanco de bajada (vi)



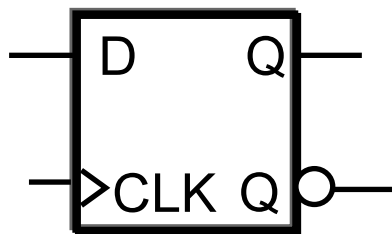
- | | |
|---|---|
|  <i>Maestro habilitado, $Q_m = D$</i> |  <i>Esclavo habilitado, $Q = Q_m$</i> |
|  <i>Maestro no habilitado</i> |  <i>Esclavo no habilitado</i> |

- La línea interna Q_m cambia cuando $CLK = 1$, siguiendo la evolución de la entrada
- La salida Q sólo cambia en los flancos de CLK

El efecto neto de la configuración maestro-esclavo es que el último valor de la entrada D justo antes del flanco es el valor que aparece en la salida Q



Sólo es necesario añadir un inversor a la entrada CLK del biestable D activo por flanco de bajada



Símbolo lógico

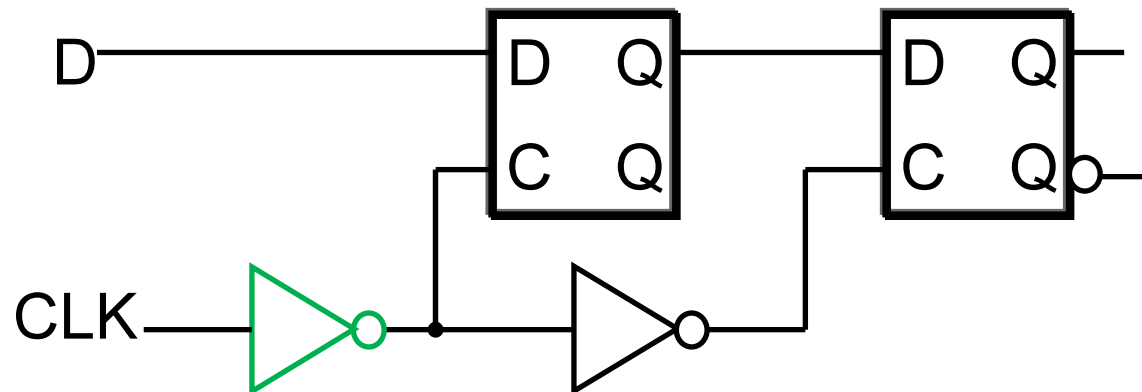


Diagrama lógico

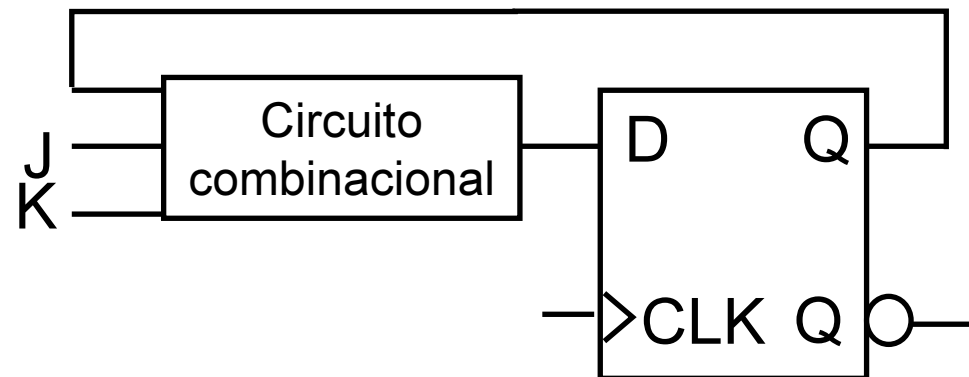
CLK	D	Q(t+1)	/Q(t+1)
0	X	Q(t)	/Q(t)
1	X	Q(t)	/Q(t)
↑	1	1	0
↑	0	0	1

Tabla de verdad

El biestable S-R
presenta problemas
cuando se activan
simultáneamente
las dos entradas S y R



Podemos diseñar
un biestable similar
que no presente problemas
a partir de un biestable D



**Debemos decidir qué hará el circuito cuando se activen
al tiempo las entradas J (puesta a 1) y K (puesta a 0):
En un J-K, invertir el estado anterior**

Biestable J-K activado por flanco de subida

- Biestable D activado por flanco de subida
- Entrada **J** o de puesta a **1**:
cuando se active, $Q(t+1) = 1$
- Entrada **K** o de puesta a **0**:
cuando se active, $Q(t+1) = 0$
- Si **no se activa ninguna**:
mantener el estado $Q(t+1) = Q(t)$
- Si **se activan al tiempo**:
invertir el estado: $Q(t+1) = \overline{Q(t)}$

Biastable J-K (iii)

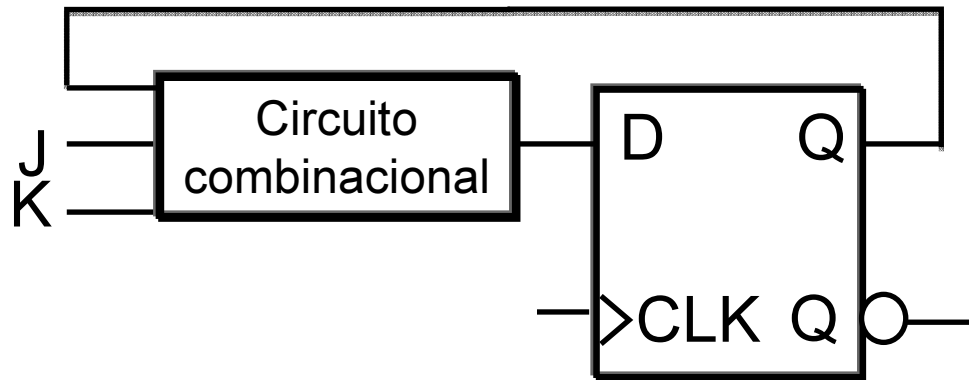


Diagrama lógico

J	K	Q(t)	D
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

Tabla de verdad

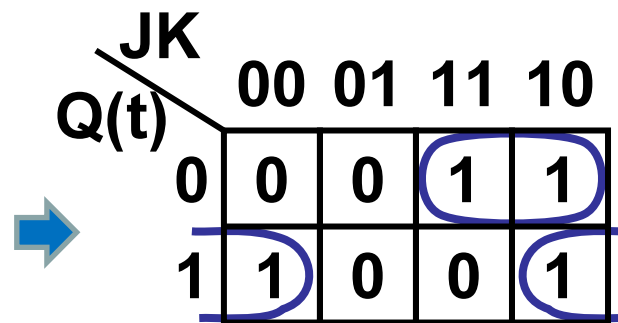
- Utilizando esta tabla de verdad, el biestable almacena el valor de D cuando llegue el flanco de reloj

J	K	Q(t)	D
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0

$J=0, K=0 \Rightarrow D = Q(t) \Rightarrow Q(t+1) = D = Q(t)$
 $J=0, K=1 \Rightarrow D = 0 \Rightarrow Q(t+1) = D = 0$
 $J=1, K=0 \Rightarrow D = 1 \Rightarrow Q(t+1) = D = 1$
 $J=1, K=1 \Rightarrow D = \overline{Q(t)} \Rightarrow Q(t+1) = D = \overline{Q(t)}$

- Implementación del circuito que calcula D en función de J, K y Q(t)

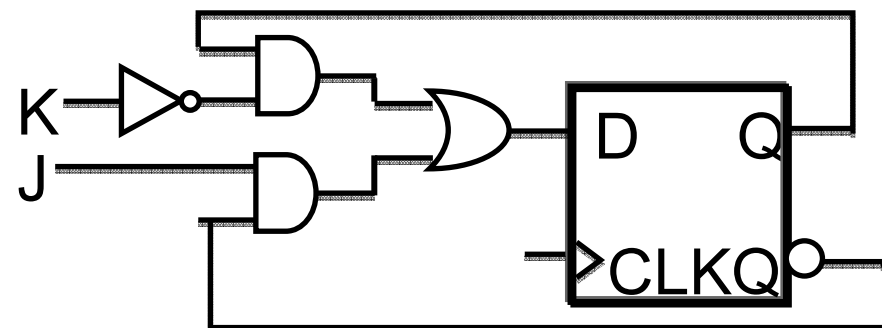
J	K	Q(t)	D
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	0



$$D = J\bar{Q}(t) + \bar{K}Q(t)$$

y aprovechando que $\bar{Q}(t) = \overline{Q(t)}$,

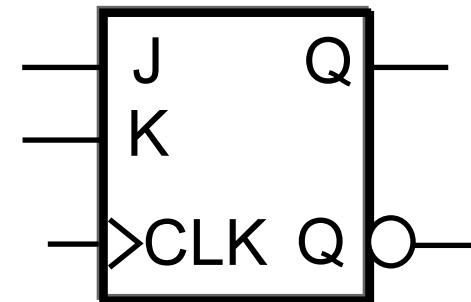
$$D = J/\bar{Q}(t) + \bar{K}Q(t)$$



Biastable J-K (vi)

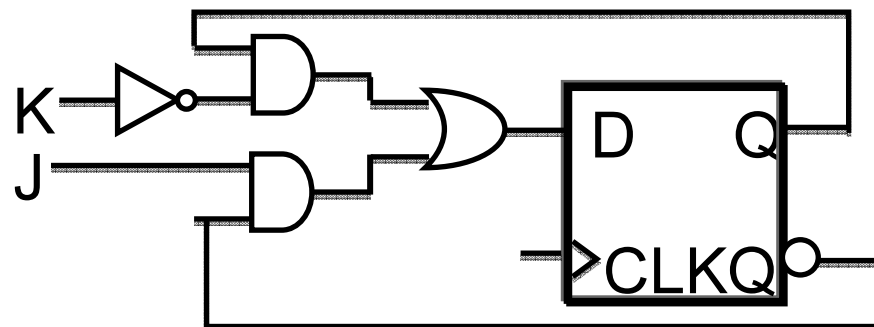
CLK	J	K	Q(t+1)	/Q(t+1)
0	X	X	Q(t)	/Q(t)
1	X	X	Q(t)	/Q(t)
↑	0	0	Q(t)	/Q(t)
↑	0	1	0	1
↑	1	0	1	0
↑	1	1	/Q(t)	Q(t)

Tabla de verdad

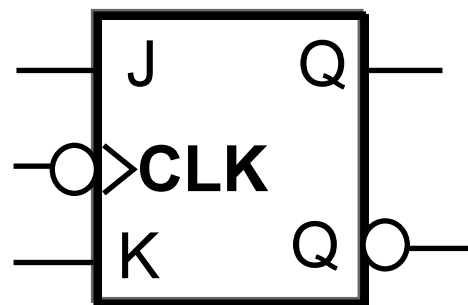


Símbolo lógico

Diagrama lógico



- Biestable J-K activado por flanco de bajada
 - El elemento de memoria es un biestable D activado por flanco de bajada
 - Las puertas lógicas y las conexiones internas serían las mismas que en el caso anterior
 - Igual que en los biestables D, los biestables J-K comerciales pueden disponer de entradas asíncronas



Símbolo lógico

- Sólo tiene una entrada llamada T (*toggle*)
- Mantiene el estado (si $T=0$) o lo cambia (si $T=1$) cada vez que llega un flanco de reloj
- No se construye comercialmente, pero se puede implementar fácilmente utilizando un J-K

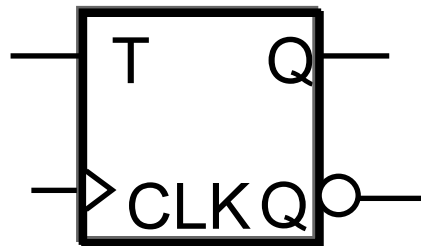
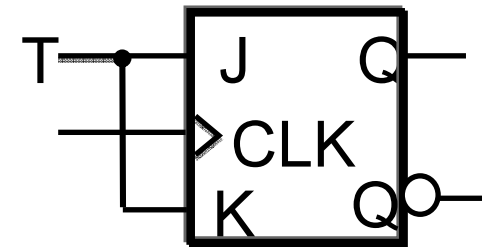


Diagrama lógico

CLK	T	Q(t+1)	/Q(t+1)
0	X	Q(t)	/Q(t)
1	X	Q(t)	/Q(t)
↑	1	/Q(t)	Q(t)
↑	0	Q(t)	/Q(t)

Tabla de verdad



Símbolo lógico

De manera análoga, se puede construir un biastable T activado por flanco de bajada usando un biastable J-K activo por flanco de bajada