



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Fonaments de computadors

TEMA 4. CIRCUITS SEQÜENCIALS

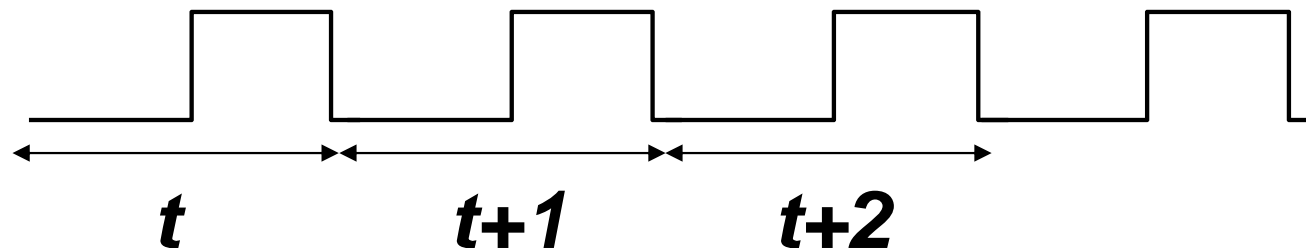
- Estudiar els circuits seqüencials més senzills.
- Introduir el concepte de cronograma.
- Comprendre el funcionament bàsic dels biestables.
- Conèixer els sistemes seqüencials bàsics més importants.
 - Registres, Banc de Registres, Memòria, Comptador

- Introducció
 - Circuits seqüencials, rellotge, cronogrames, símbols lògics.
- Biestables
 - Biestable S-R
 - Biestable D
 - Actiu per nivell
 - Actiu per flanc
 - Amb entrades asíncrones
 - Biestable J-K
 - Biestable T

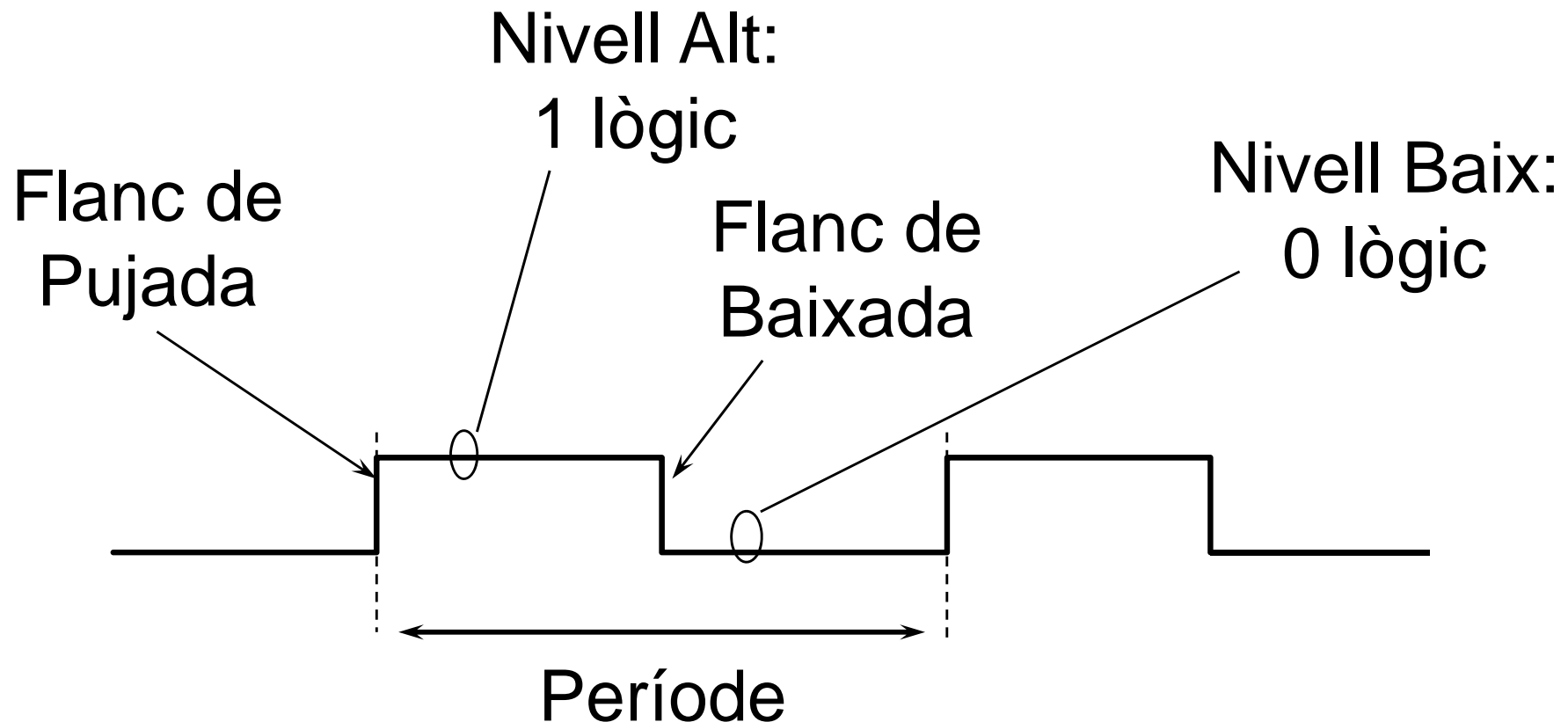
- Blocs seqüencials bàsics
 - Registres d'emmagatzemament
 - Banc de registres, Memòria
 - Registres de desplaçament
 - Comptadors
- Anàlisi de SS per cronograma

- Circuits seqüencials:
 - Les eixides del circuit en l'instant actual $S(t)$ depenen no tan sols del valor actual de les entrades del circuit $E(t)$, sinó també de la “memòria” o “estat emmagatzemat” $Q(t)$
 - Els circuit seqüencials estan formats per:
 - un bloc combinacional
 - un bloc d'elements de memòria per emmagatzemar l'estat Q
 - una entrada de rellotge que decideix quan passem del temps t al temps $t+1$, i que marca quan s'emmagatzema interiorment l'estat del sistema

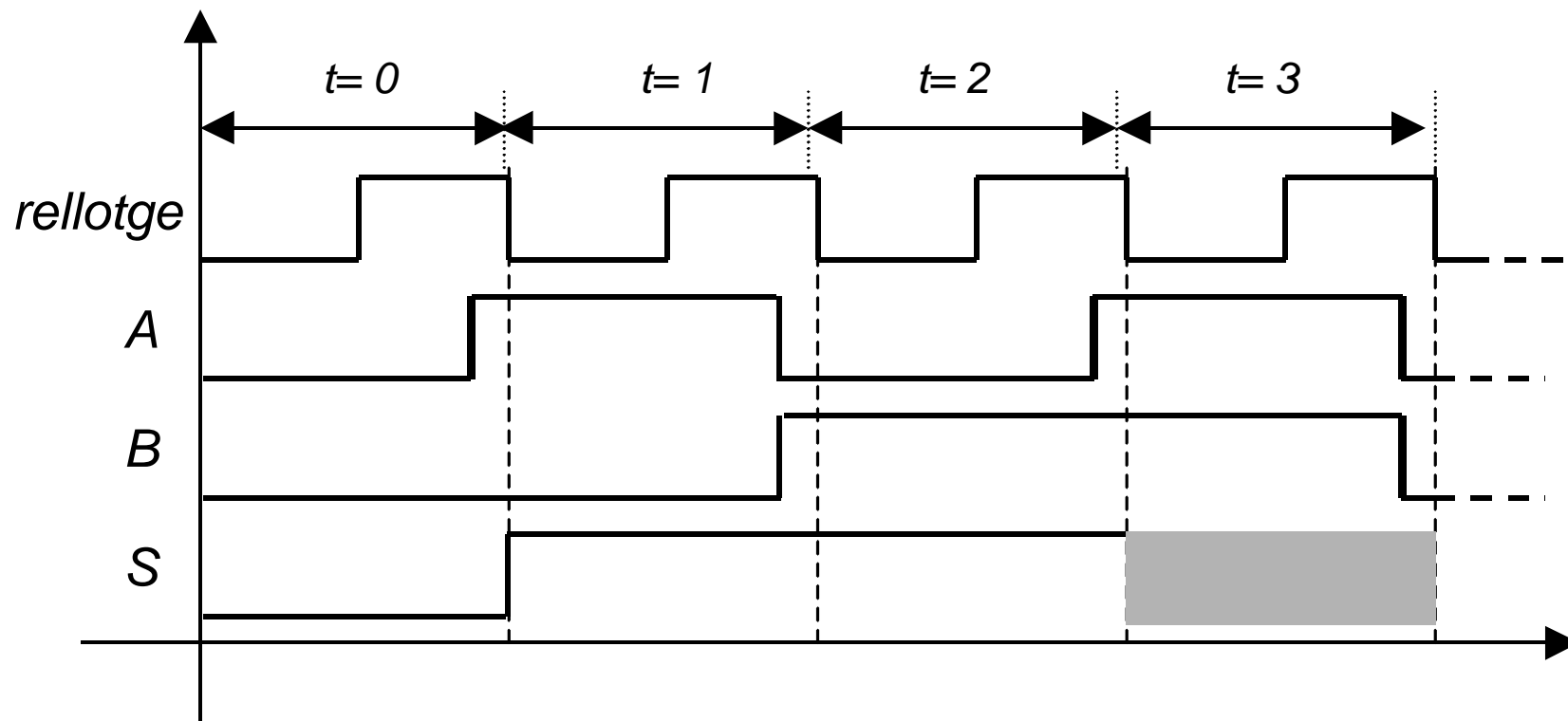
Rellotge



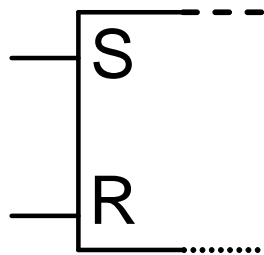
- El senyal de rellotge indica als elements de memòria quan cal que canvien el seu estat



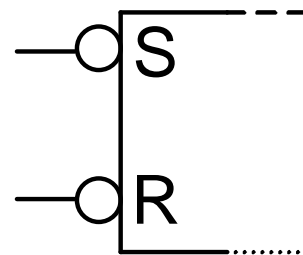
- Cronograma: Representació de l'evolució temporal de les entrades i eixides d'un circuit.
 - El valor desconegut es representa amb una ombra



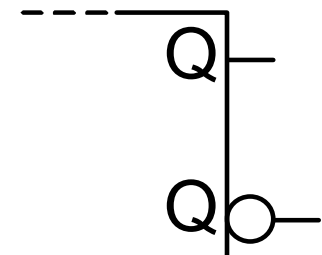
- Símbols lògics
 - Entrades i eixides



Entrades actives a nivell alt

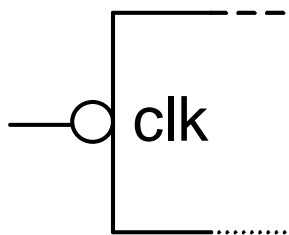


Entrades actives a nivell baix

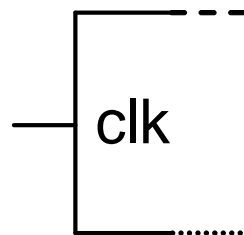


Eixides Q i /Q

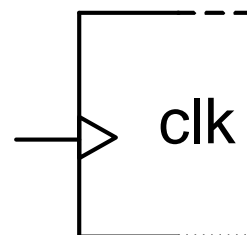
- Senyal de rellotge



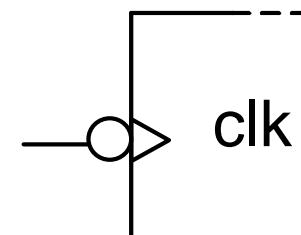
Actiu a
nivell baix



Actiu a
nivell alt



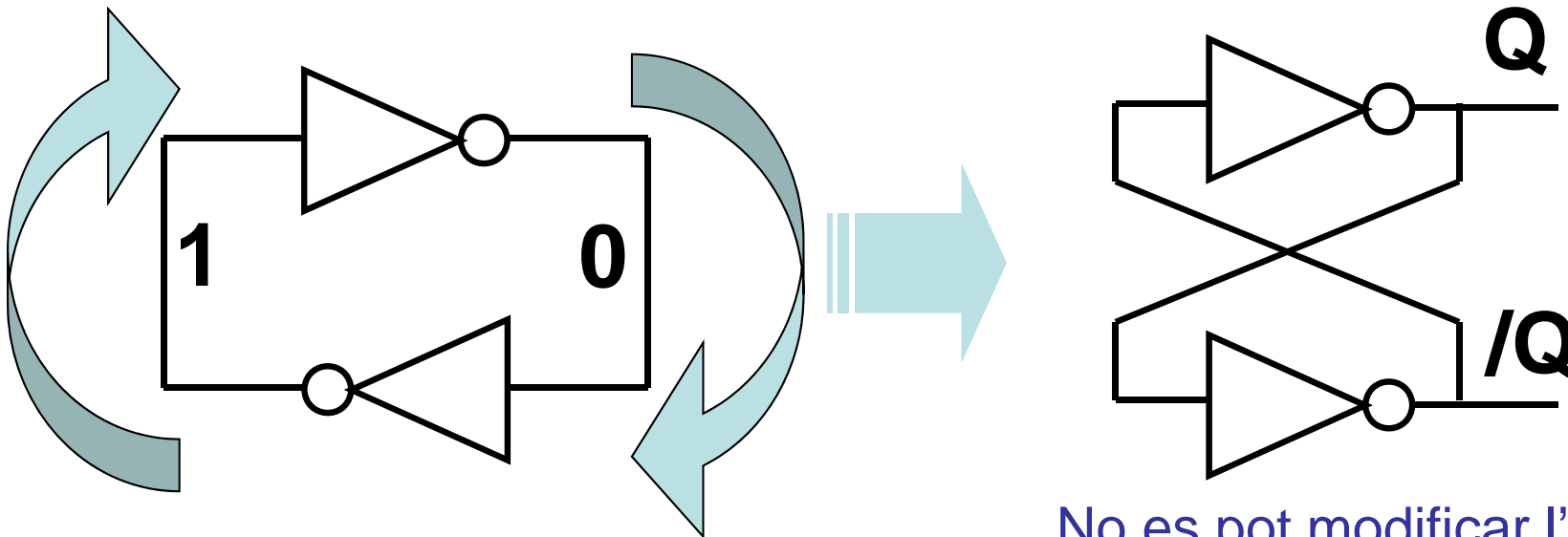
Actiu per
flanc de pujada



Actiu per
flanc de baixada

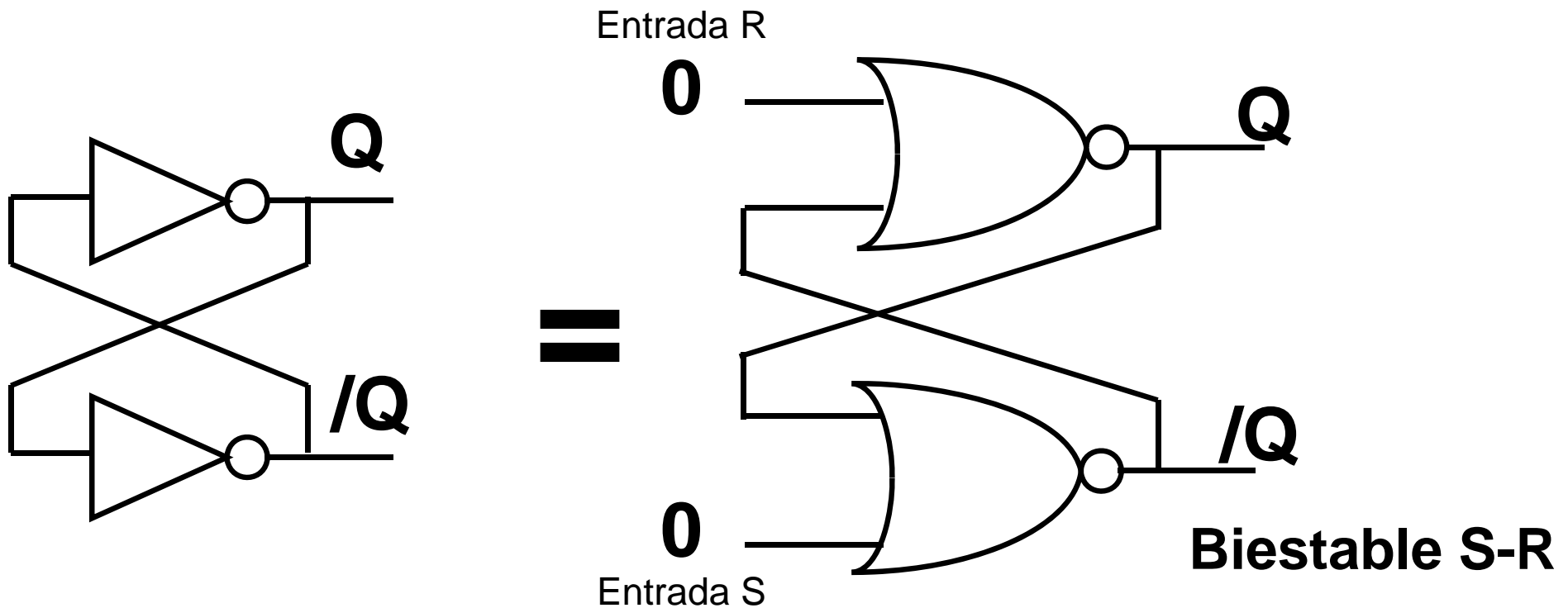
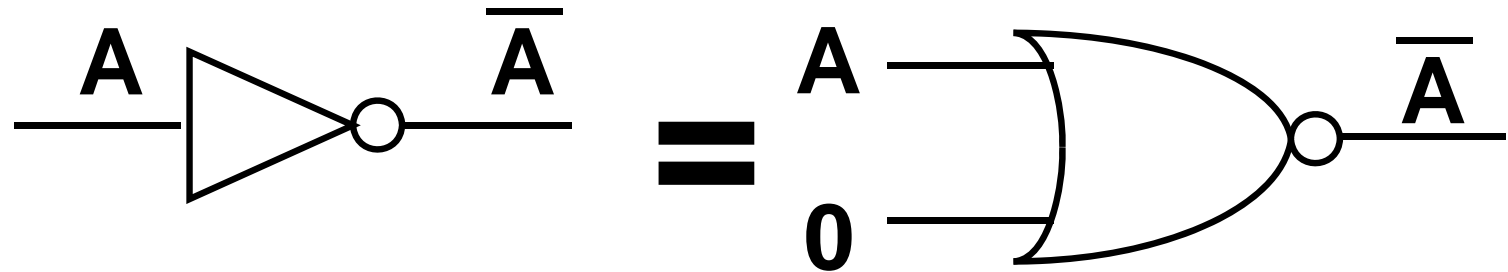
- Biestable: Circuit seqüencial amb dos estats estables (0 i 1)

Podem emmagatzemar un bit en el circuit i fer que done voltes i voltes, de manera que la informació recircule indefinidament.



No es pot modificar l'estat

Com canviar
l'estat del
biestable?



- Bistable S-R amb portes NOR
 - Com es pot determinar el seu comportament?
 1. Obtenint una taula amb tots els casos.
 - Des de les equacions (sense realimentació)
 - Les entrades externes i l'estat actual (t) són les entrades, i l'estat següent ($t+1$) són les eixides
 2. Anàlisi de la taula amb tots el casos.
 - a) Els casos s'agrupen per a cadascuna de les combinacions de valors de les entrades externes (sense l'estat actual)
 - b) Obtenim les eixides aplicant la taula una i altra vegada fins que l'estat es manté estable, es a dir, estat següent és l'estat actual.
 - Aquest mètode serveix per a qualsevol bistable actiu per nivell.

- Taula: equacions $Q(t+1) = R + \overline{Q(t)}$ i $\overline{Q(t+1)} = S + Q(t)$



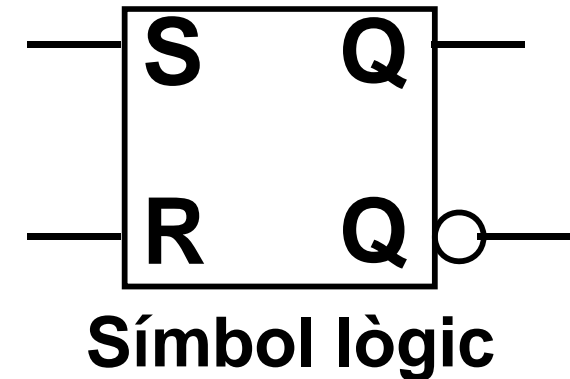
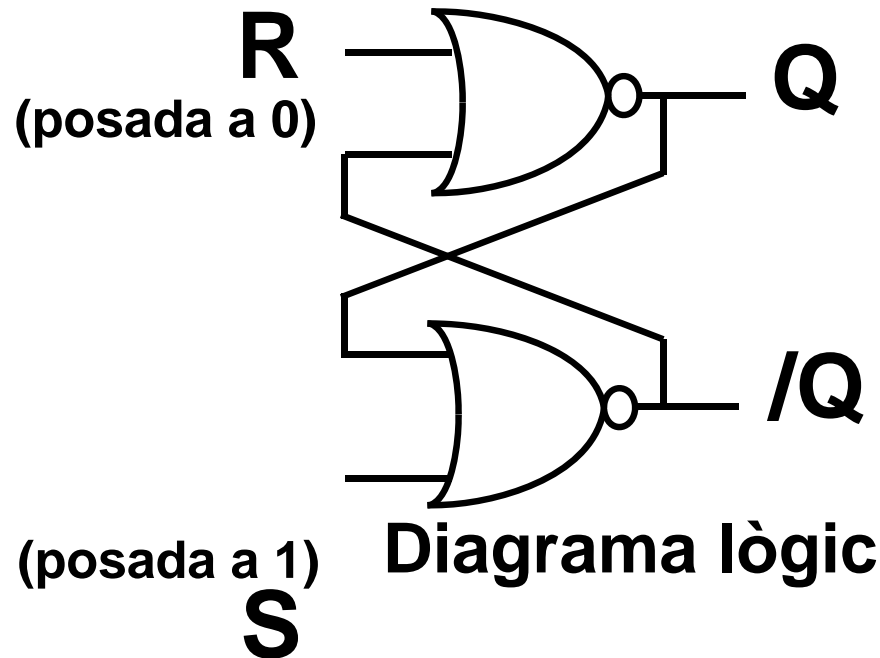
S	R	Q(t)	$\overline{Q(t)}$	Q(t+1)	$\overline{Q(t+1)}$
0	0	0	0	1	1
0	0	0	1	0	1
0	0	1	0	1	0
0	0	1	1	0	0
0	1	0	0	0	1
0	1	0	1	0	1
0	1	1	0	0	0
0	1	1	1	0	0
1	0	0	0	1	0
1	0	0	1	0	0
1	0	1	0	1	0
1	0	1	1	0	0
1	1	0	0	0	0
1	1	0	1	0	0
1	1	1	0	0	0
1	1	1	1	0	0

- Anàlisi: fem grups segons els valors d'S i R

	S	R	Q(t)	/Q(t)	Q(t+1)	/Q(t+1)		Q(t+1)	/Q(t+1)
A0	0	0	0	0	1	1	→ A3 (Oscil·la)	Q(t)	/Q(t)
A1	0	0	0	1	0	1	→ A1 (Estable)		
A2	0	0	1	0	1	0	→ A2 (Estable)		
A3	0	0	1	1	0	0	→ A0 (Oscil·la)		
B0	0	1	0	0	0	1	→ B1	0	1
B1	0	1	0	1	0	1	→ B1 (Estable)		
B2	0	1	1	0	0	0	→ B0		
B3	0	1	1	1	0	0	→ B0		
C0	1	0	0	0	1	0	→ C2	1	0
C1	1	0	0	1	0	0	→ C0		
C2	1	0	1	0	1	0	→ C2 (Estable)		
C3	1	0	1	1	0	0	→ C0		
D0	1	1	0	0	0	0	→ D0 (Estable)	0*	0*
D1	1	1	0	1	0	0	→ D0		
D2	1	1	1	0	0	0	→ D0		
D3	1	1	1	1	0	0	→ D0		

- Bistable S-R amb portes NOR
 - Comentaris als valors de resum de l'anàlisi
 1. Quan l'estat següent (els valors de $Q(t+1)$ i $/Q(t+1)$) és el mateix per tots els casos agrupats, resumir és fàcil
 - Indiquem el valor numèric (casos Bn , Cn i Dn)
 - El cas (Dn) en el qual els valors resum són $Q(t+1) = /Q(t+1)$ indiquem amb un * que la situació és “**no volguda**” perquè **no** es compleix la condició habitual $/Q(t+1) = \overline{Q(t+1)}$
 2. Quan els valors d'estat següent són diferents per els casos agrupats, (casos An), el resum és més difícil
 - Expressem $Q(t+1)$ i $/Q(t+1)$ a partir dels valors de $Q(t)$ i $/Q(t)$
 - **L'oscil·lació** (l'estat següent passa de 00 a 11 i de 11 a 00 de manera indefinida) **no apareix en el resum**(cassos $A0$ i $A3$)

- Bistable S-R amb portes NOR

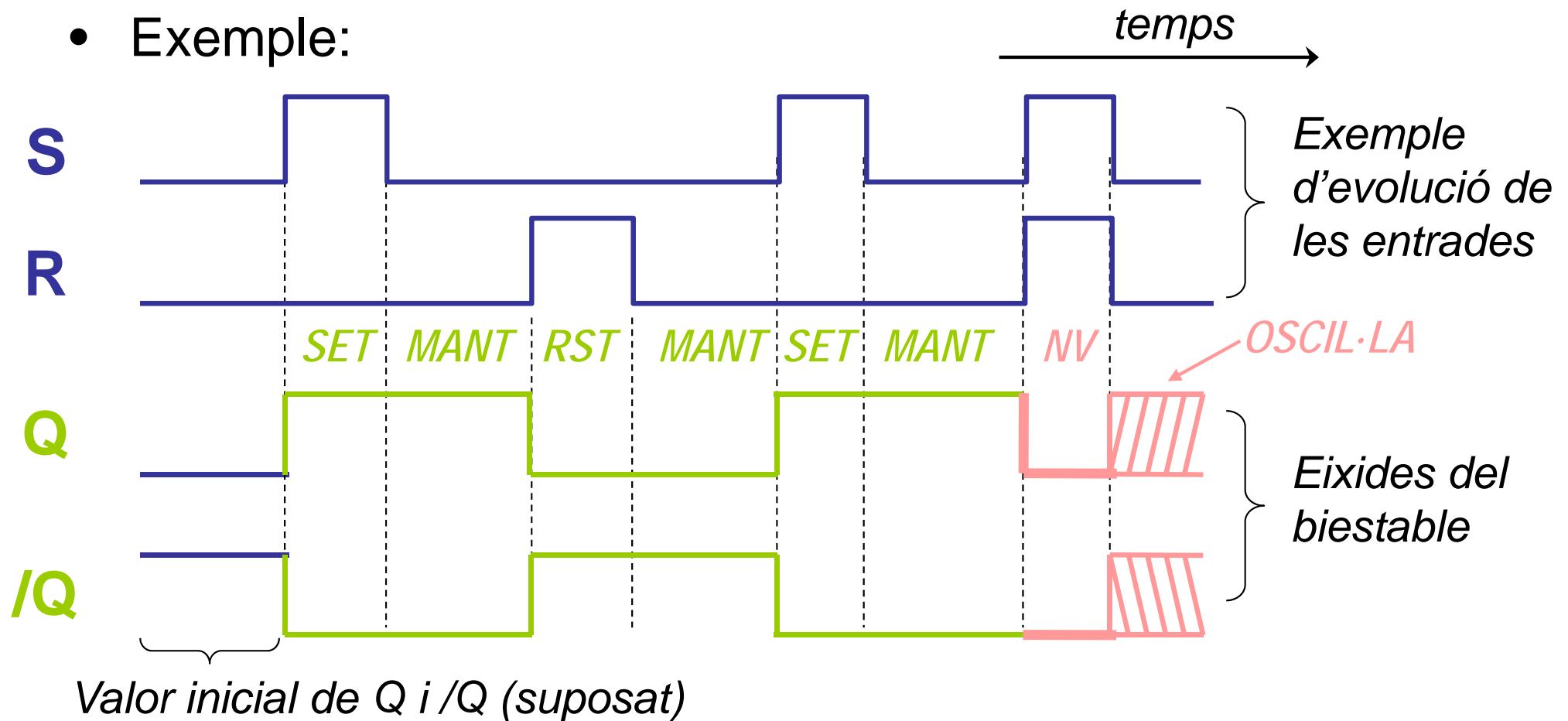


S	R	Q(t+1)	/Q(t+1)
0	0	Q(t)	/Q(t)
0	1	0	1
1	0	1	0
1	1	0*	0*

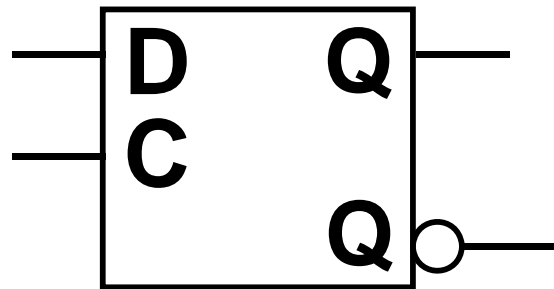
Taula de funcionament

* =situació no volguda

- El cronograma s'utilitza per conèixer l'evolució temporal de l'estat quan canvien les entrades
- Exemple:



- S'utilitzen per la implementació d'elements de memòria, amb la finalitat d'emmagatzemar el valor d'una línia d'informació (un bit)

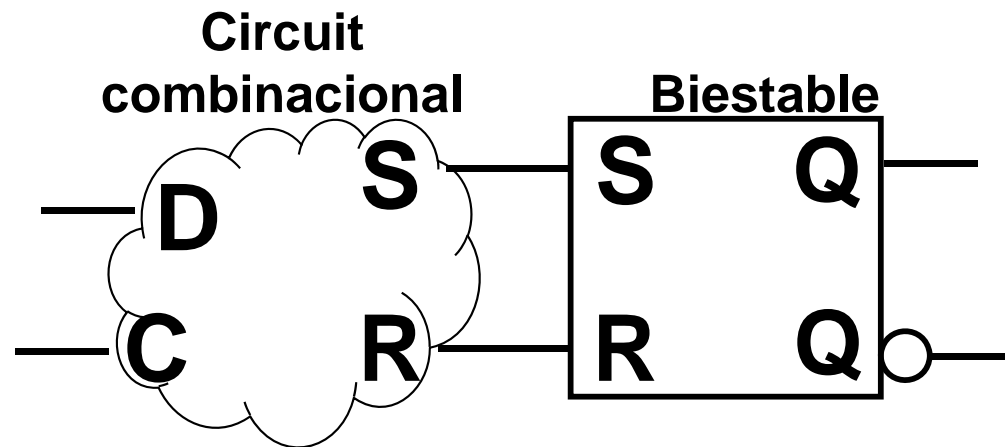


Símbol lògic

C	D	$Q(t+1)$	$/Q(t+1)$
0	X	$Q(t)$	$/Q(t)$
1	1	1	0
1	0	0	1

Taula de funcionament

- Podem construir un biastable D partint d'un S-R.



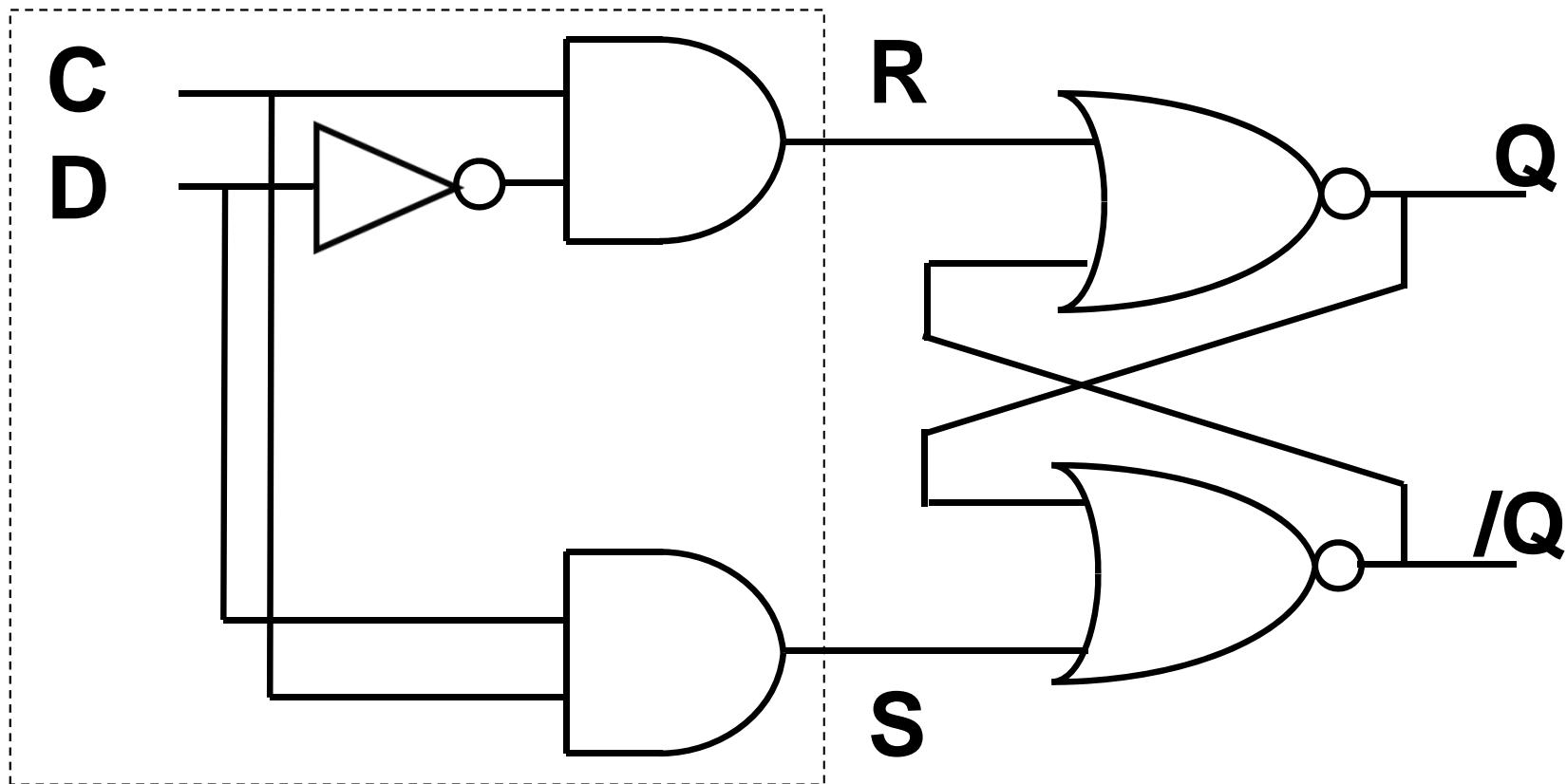
- Especificació del circuit combinacional

- Si $C=0$, volem que $Q(t+1) = Q(t)$, per tant l'eixida cal que siga $S=R=0$
- Si $C=1$ i $D=0$, volem que $Q(t+1) = 0$, per tant l'eixida cal que siga $S=0$ y $R=1$
- Si $C=1$ y $D=1$, volem que $Q(t+1) = 1$, per tant l'eixida cal que siga $S=1$ y $R=0$

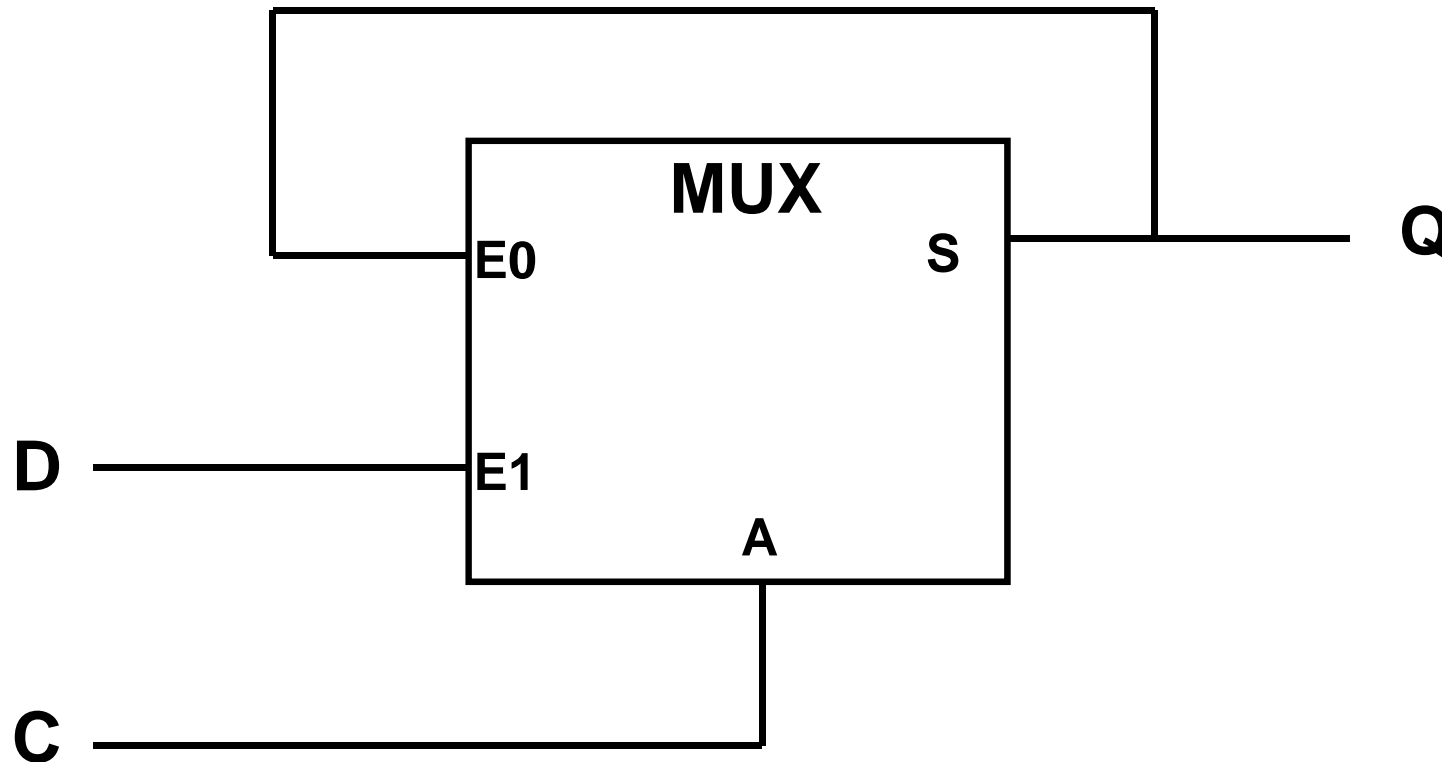
Entrades		eixides	
C	D	S	R
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	0

Efecte en S-R	
$Q(t+1)$	$/Q(t+1)$
$Q(t)$	$/Q(t)$
$Q(t)$	$/Q(t)$
0	1
1	0

- De la taula del circuit combinacional s'obté $S=C \cdot D$ i $R=C \cdot \bar{D}$



- Altra possible implementació:

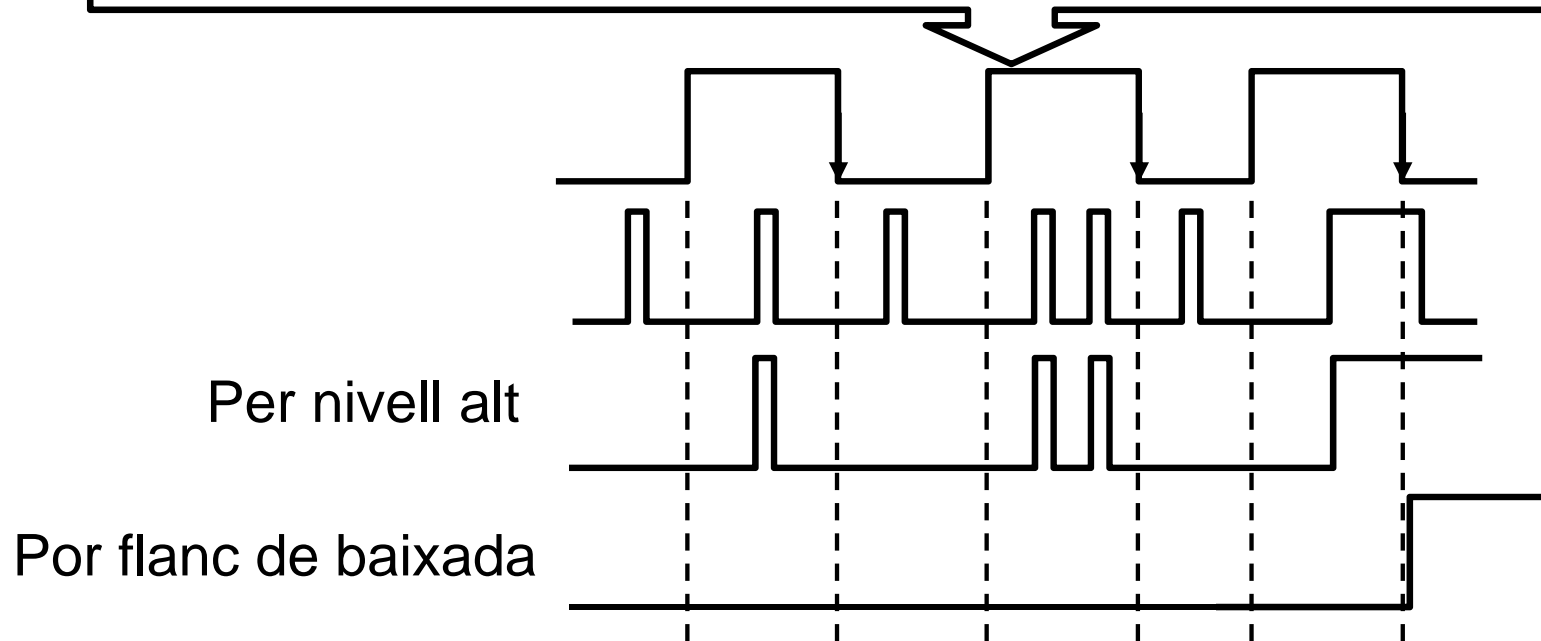


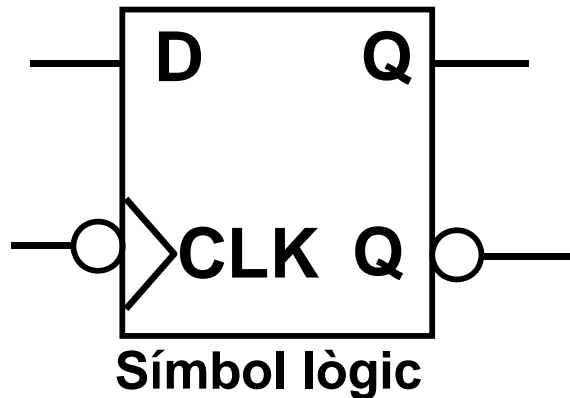
Si el biestable és actiu per nivell
l'es produeix un impuls no
volgut en l'entrada de dades...



L'impuls no volgut
es traslladarà a l'eixida

Podem fer que el senyal de rellotge treballe per flanc

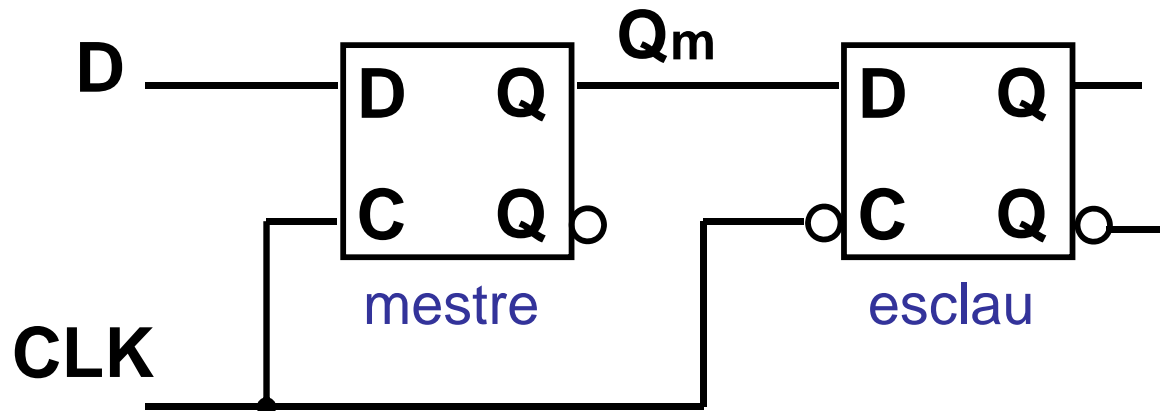




CLK	D	Q(t+1)	/Q(t+1)
0	X	Q(t)	/Q(t)
1	X	Q(t)	/Q(t)
↓	1	1	0
↓	0	0	1

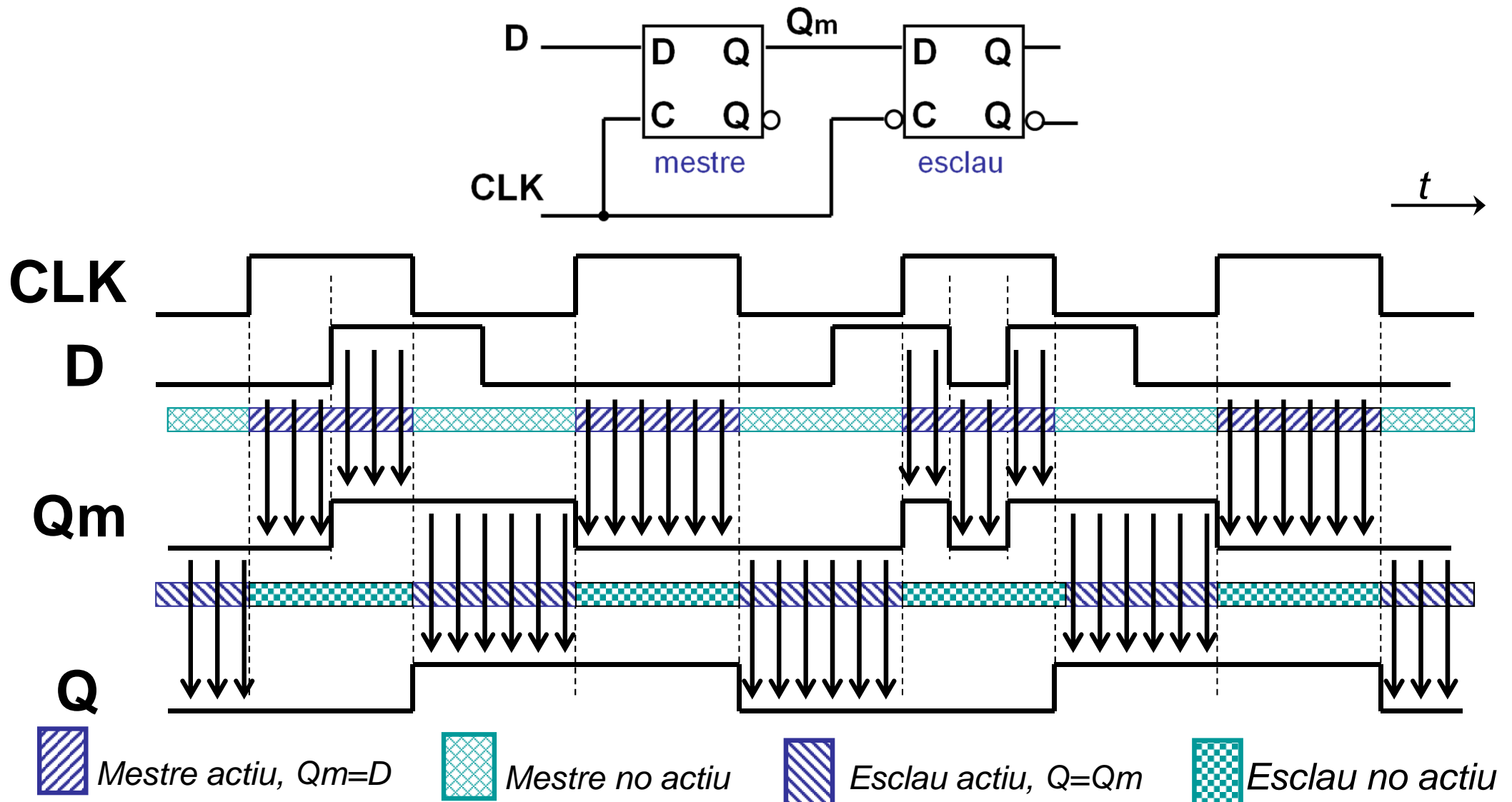
Taula de funcionament

- Per implementar biestables que s'activen per flanc s'utilitzen freqüentment dos biestables actius per nivell en una configuració mestre-esclau (MASTER-SLAVE)

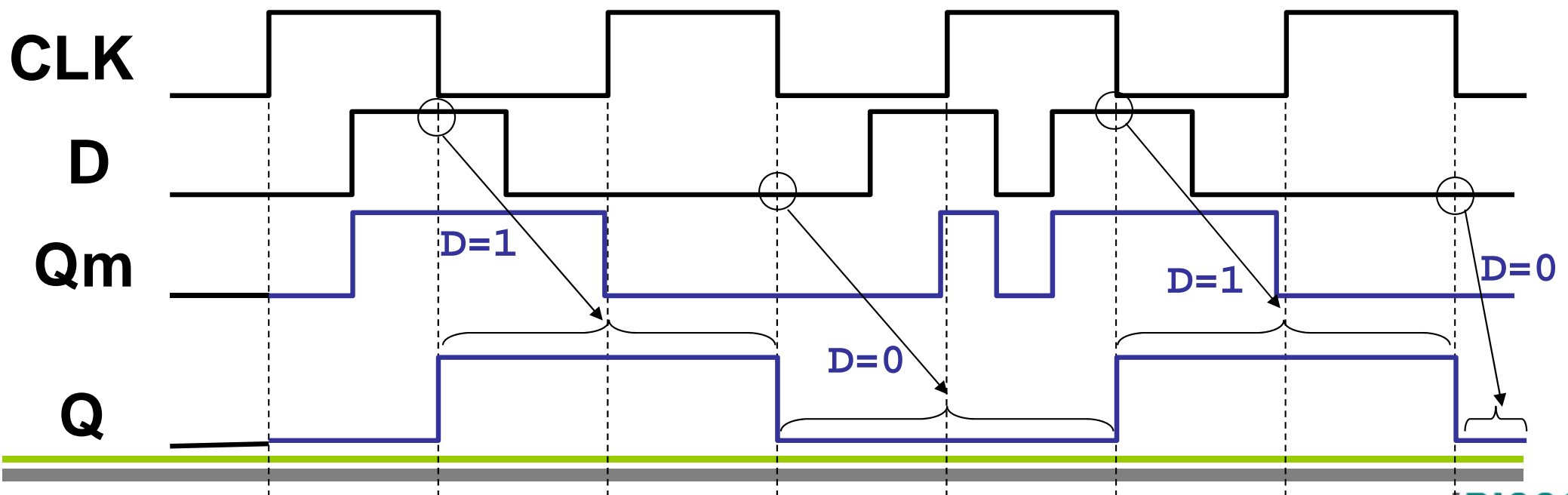


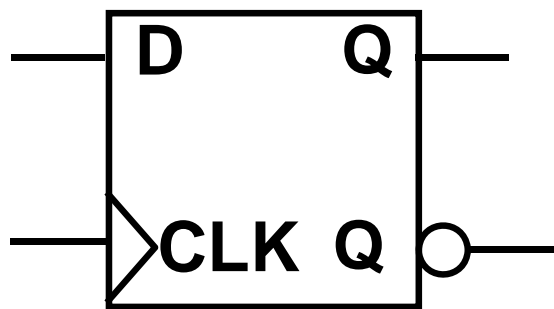
Biestable D actiu per flanc de baixada

FCO



- Com es pot observar:
 - La línia interna Qm canvia quan CLK = 1, seguint l'evolució de l'entrada D
 - L'eixida Q només canvia en els flancs de CLK
- *L'efecte final de la configuració mestre-esclau és que l'últim valor de l'entrada D just abans del flanc és el valor que apareix en l'eixida Q*



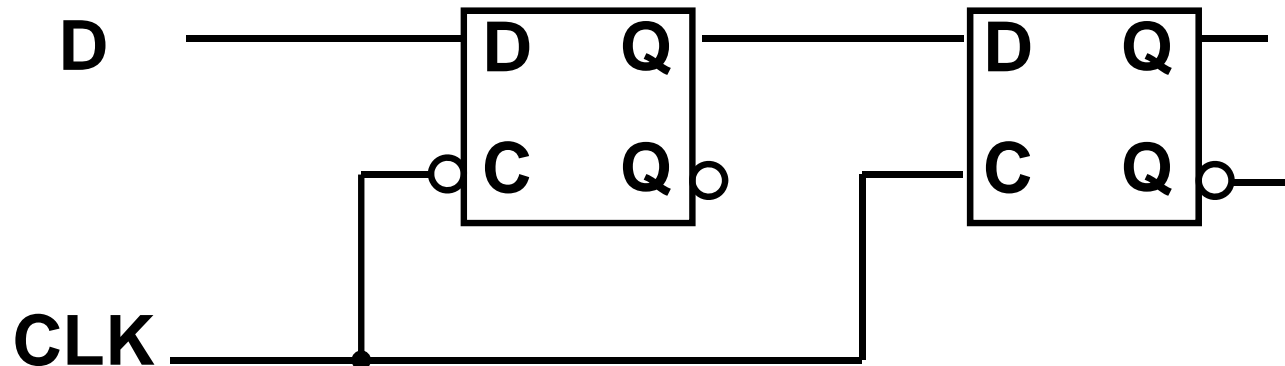


Símbol lògic

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

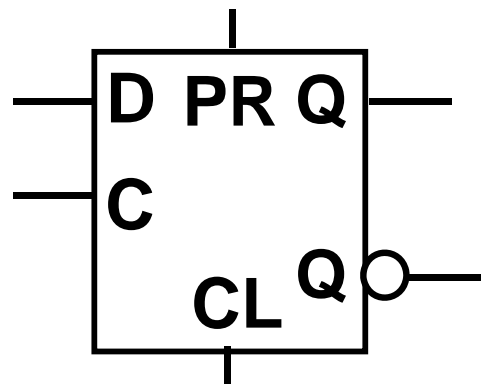
Taula de funcionament

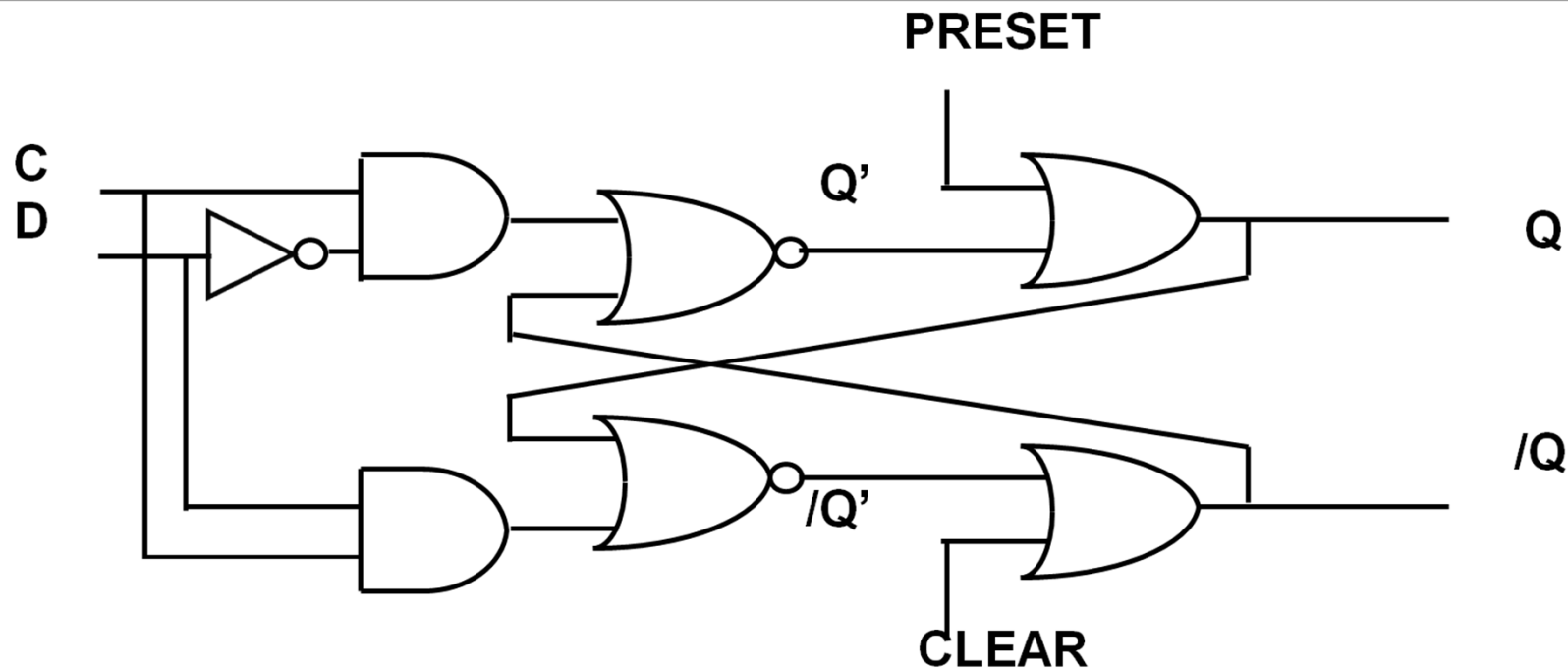
- Implementació (mestre actiu a nivell baix, esclau a nivell alt)



- Al biastable D actiu per nivell (alt o baix) se li poden afegir entrades asíncrones del tipus:
 - CLEAR (CL): posada a zero asíncrona
 - PRESET (PR): posada a un asíncrona
- Les entrades asíncrones:
 - Tenen prioritat sobre els altres senyals
 - Permeten canviar l'estat del biastable en qualsevol moment, independentment del valor de la resta de les entrades

Símbol lògic





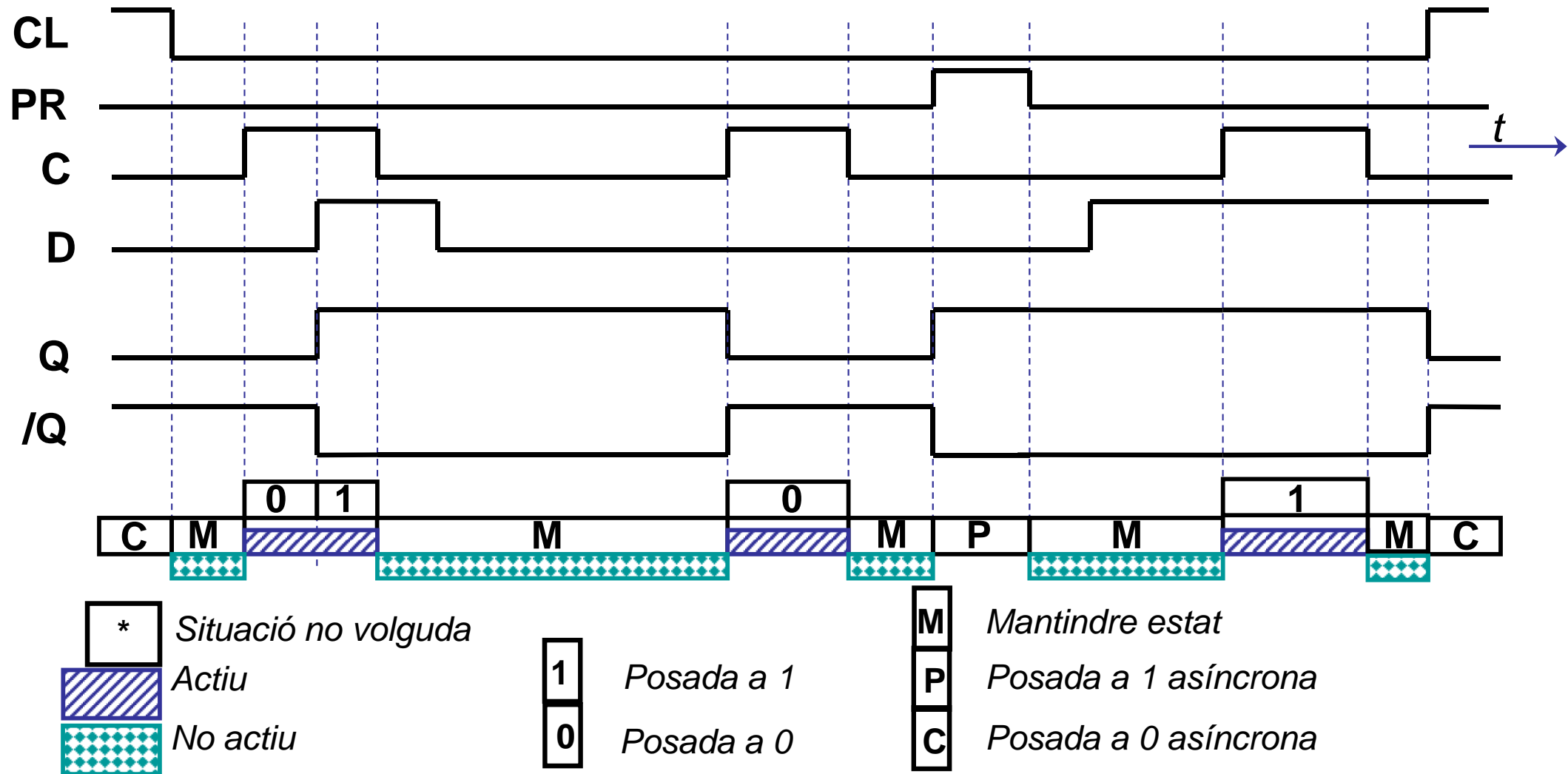
- Si $CLEAR=PRESET=0 \Rightarrow Q = Q', /Q = /Q'$
- Si $CLEAR=1, PRESET=0 \Rightarrow /Q = 1 \Rightarrow Q' = 0 \Rightarrow Q = 0$
- Si $CLEAR=0, PRESET=1 \Rightarrow Q = 1 \Rightarrow /Q' = 0 \Rightarrow /Q = 0$
- Si $CLEAR=1, PRESET=1 \Rightarrow Q=/Q=1^*$

- Taula de funcionament

PR	CL	C	D	$Q(t+1)$	$/Q(t+1)$
0	1	X	X	0	1
1	0	X	X	1	0
1	1	X	X	1*	1*
0	0	1	1	1	0
0	0	1	0	0	1
0	0	0	X	$Q(t)$	$/Q(t)$

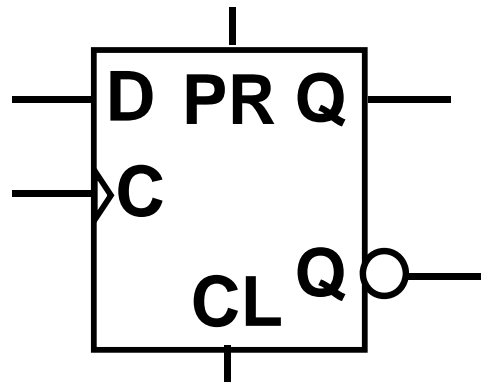
Biestable D amb entrades asíncrones

FCO

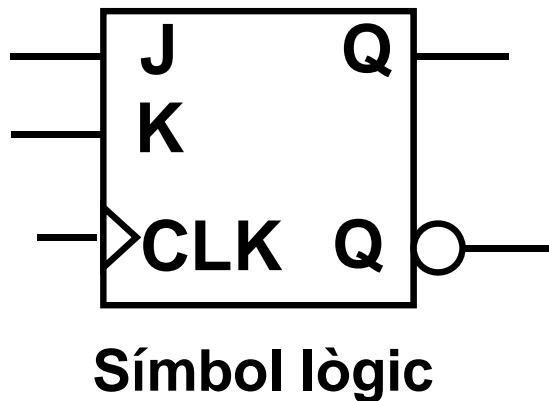


- Al biastable D actiu per flanc (pujada o baixada) també se li poden afegir entrades asíncrones:
 - En aquest cas, tant el mestre com l'esclau han de tenir entrades asíncrones
- Exemple: Biastable D actiu per flanc de pujada amb entrades asíncrones de PRESET i CLEAR

Símbol lògic



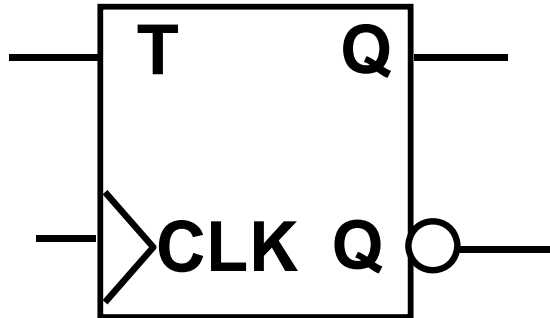
- El biestable S-R presenta problemes en activar-se simultàniament les dues entrades S i R
- El biestable J-K té un funcionament semblant al S-R, però evita el problema anterior invertint l'estat quan J i K estan actives simultàniament



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)

Taula de funcionament

- Només té una entrada anomenada T (*toggle*)
- Manté l'estat (si $T=0$) o el canvia (si $T=1$) cada vegada que arriba un flanc actiu de rellotge
- No es construeix comercialment, però es pot construir fàcilment amb un J-K



Símbol lògic

Clk	T	Q(t)	Q(t+1)
↑	0	0	0
↑	0	1	1
↑	1	0	1
↑	1	1	0
↓/0/1	X	Q(t)	Q(t)

Taula de funcionament

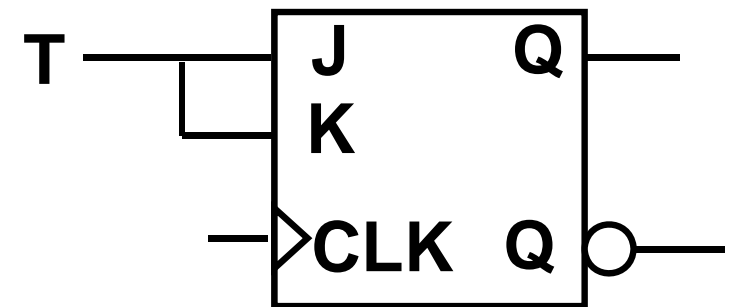


Diagrama lògic

- De manera anàloga, es pot construir un biestable T actiu per flanc de baixada emprant un biestable J-K actiu per flanc de baixada

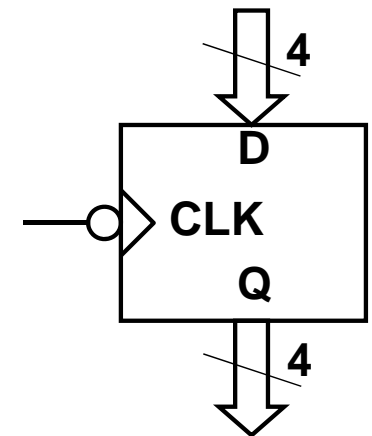
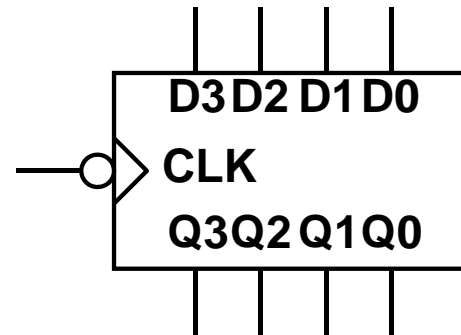
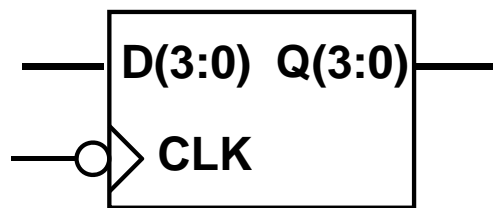
- Hi ha sistemes seqüencials tan importants i d'ús tan comú que se'ls ha donat nom
 - Registre d'emmagatzemament
 - Emmagatzema una dada d' N bits
 - Banc de Registres i Memòria
 - Agrupació de diversos registres d'emmagatzemament
 - Mateixa funcionalitat bàsica. Difereixen fonamentalment en aspectes de capacitat, velocitat de funcionament i disseny

- Hi ha sistemes seqüencials tan importants i d'ús tan comú que se'ls ha donat nom (cont.)
 - Registre de desplaçament
 - Emmagatzema una dada d' N bits. Es necessiten N cicles de rellotge perquè la informació entre (escriptura) i/o isca (lectura)
 - Comptador
 - Circuit que canvia de valor de forma autònoma en cada cicle de rellotge que segueix una seqüència de valors predeterminada
 - Generalment el compte és binari (ascendent o descendent)

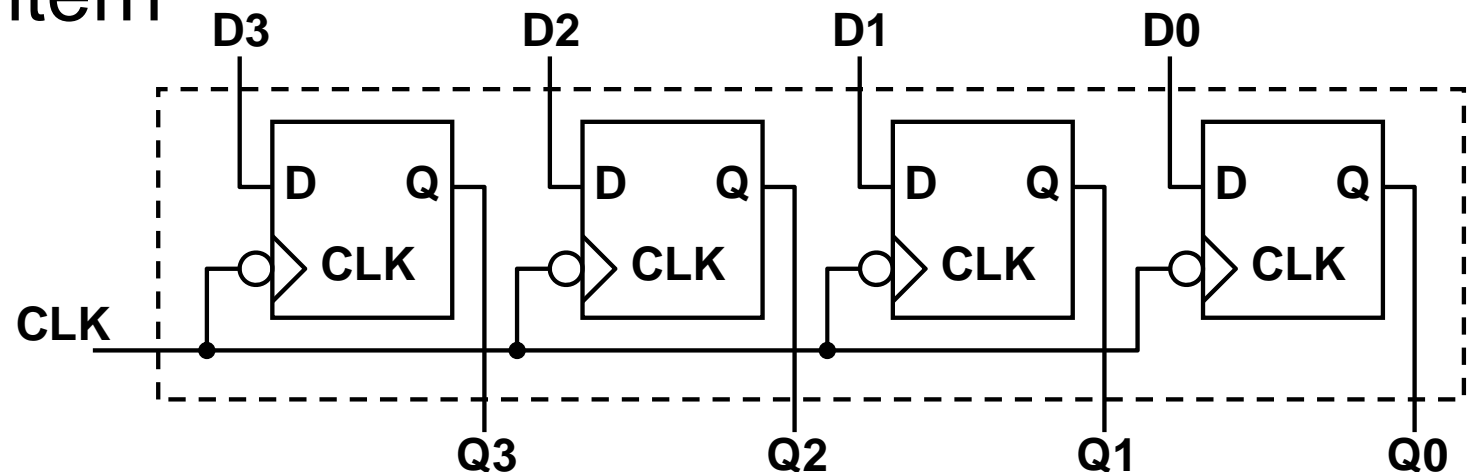
- Agrupació síncrona de biestables D actius per flanc
 - Tants biestables D com a bits vulguem emmagatzemar
 - Cada biestable D emmagatzema un bit
 - Les entrades/eixides de dades del registre coincideixen amb les dels biestables D
 - Un únic rellotge (sistema síncron) interconnecta les entrades de rellotge de tots els biestables
 - Tots els biestables han de ser actius en el mateix flanc de rellotge

- Exemple: Registre d'emmagatzemament de 4 bits actiu per flanc de baixada

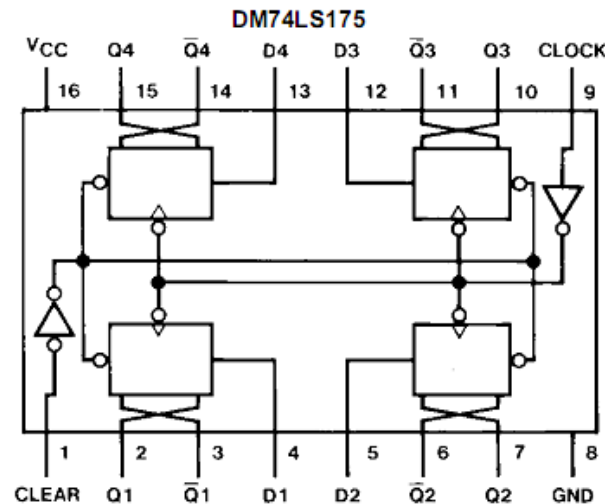
- Símbols lògics possibles



- Esquema intern



- Circuit integrat '175
- 4 biestables tipus D, actius per flanc de pujada amb entrada asíncrona de posada a 0



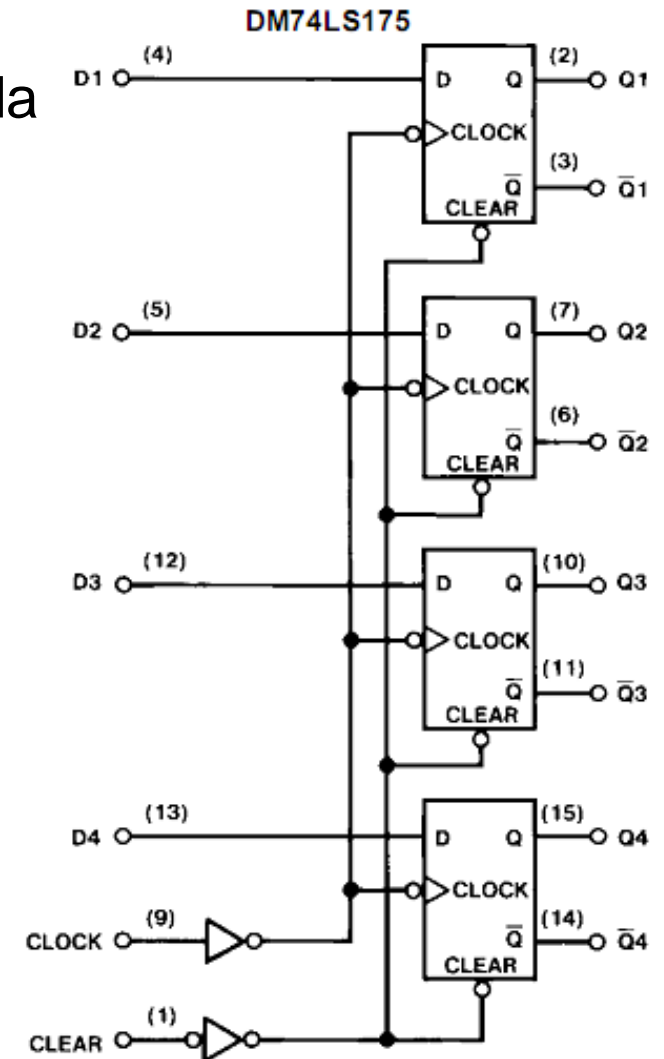
Function Table

(Each Flip-Flop)

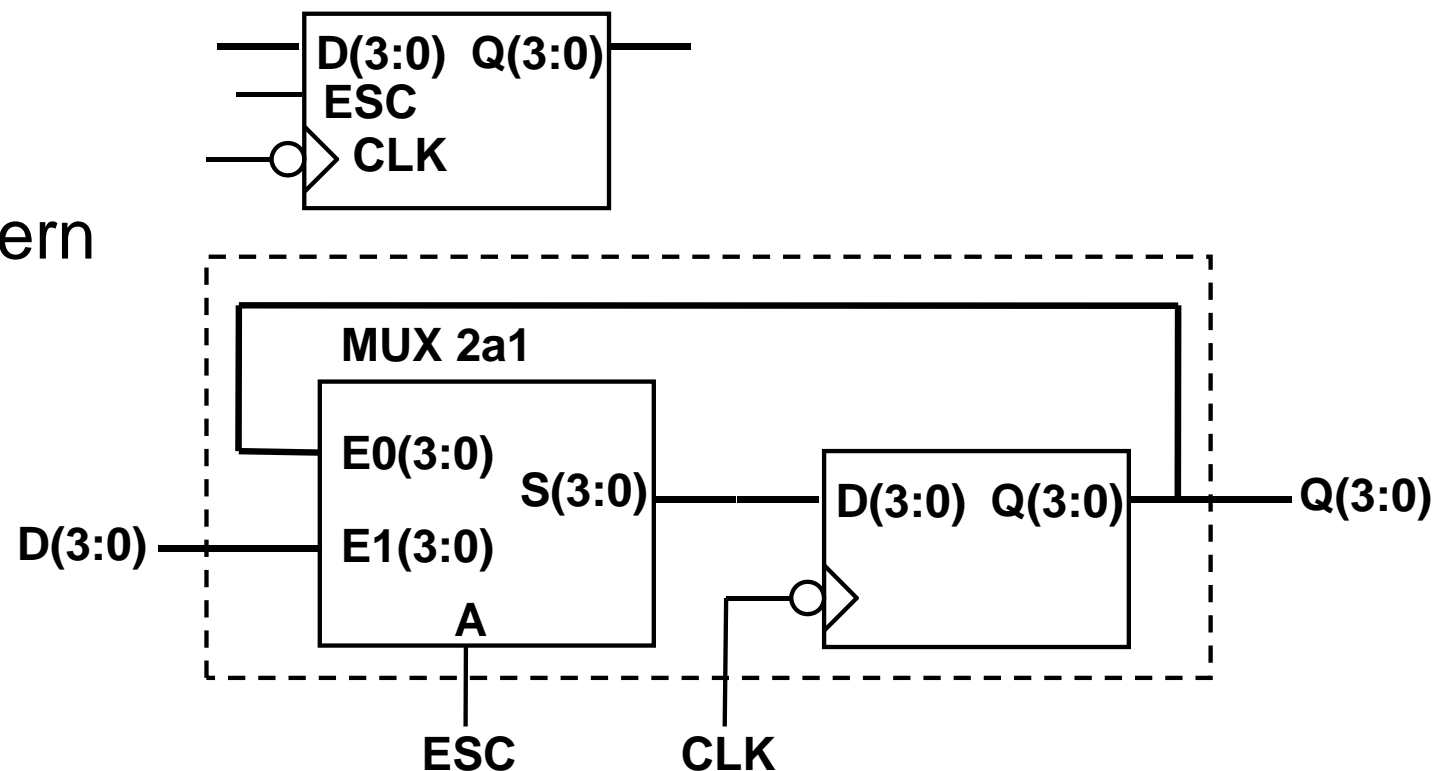
Inputs			Outputs	
Clear	Clock	D	Q	\bar{Q} †
L	X	X	L	H
H	↑	H	H	L
H	↑	L	L	H
H	L	X	Q_0	\bar{Q}_0

H = HIGH Level (steady state)

L = LOW Level (steady state)



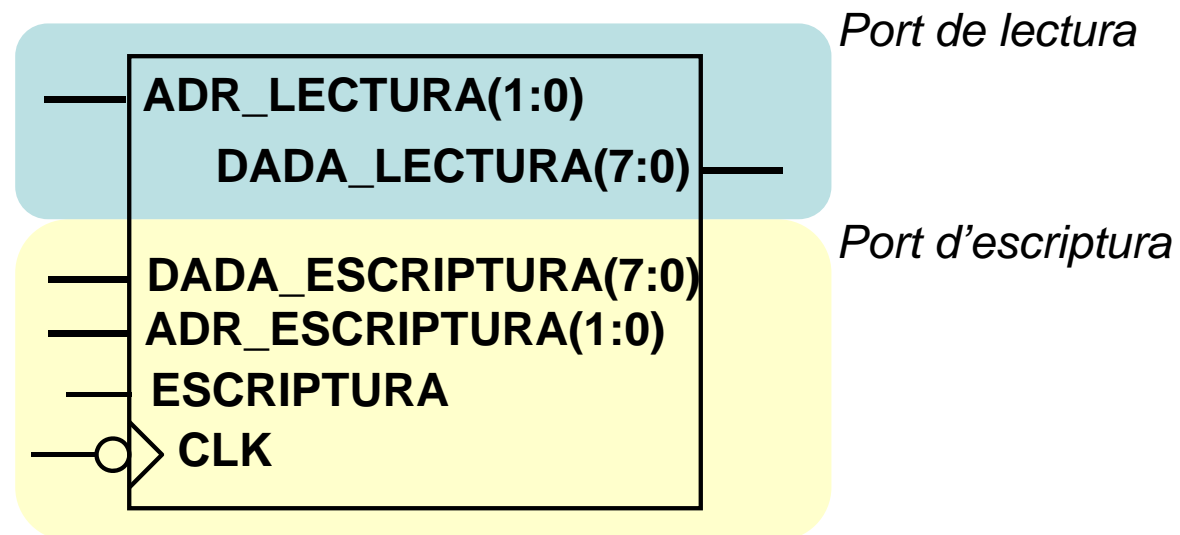
- Senyal d'escriptura en paral·lel (*parallel load*)
 - Permet habilitar o deshabilitar l'escriptura en el registre durant el flanc actiu de rellotge
 - Símbol lògic
 - Esquema intern



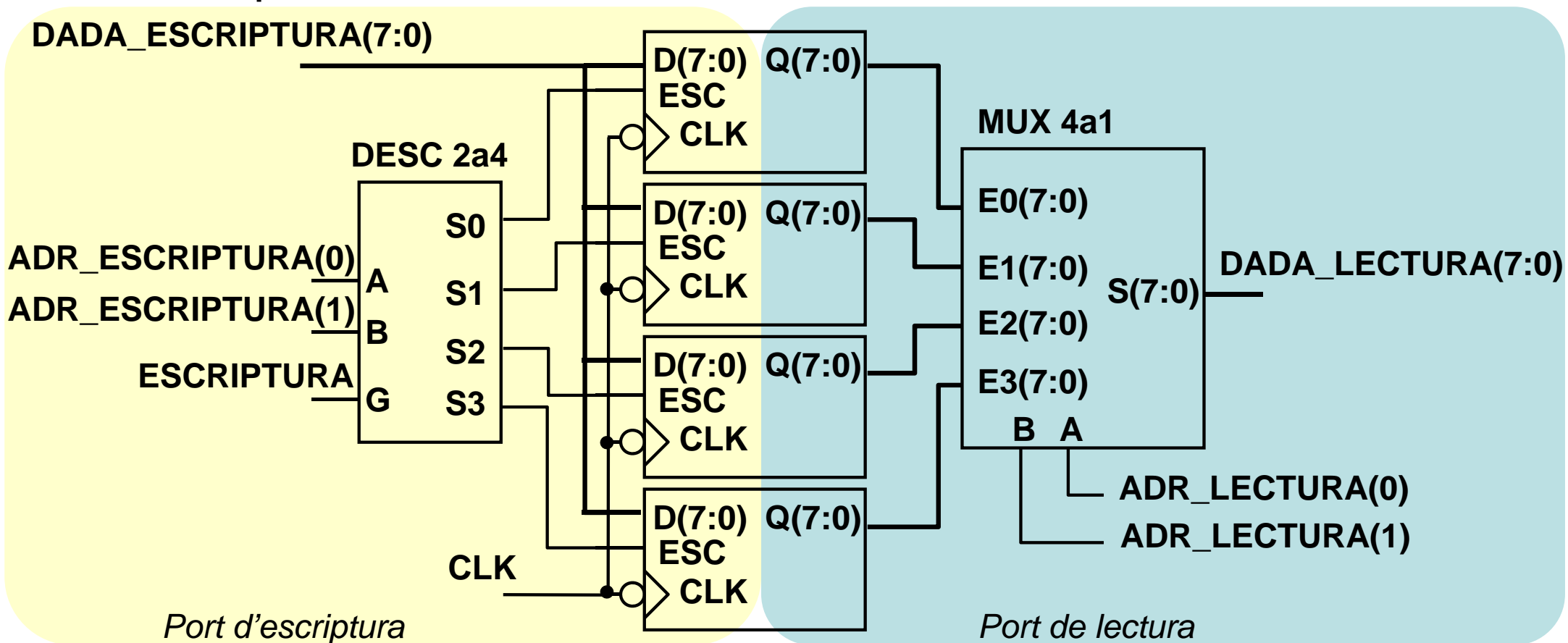
- Operació d'escriptura
 - Operació destructiva
 - La dada (prèviament) emmagatzemada desapareix i és sobreescrita amb la dada de l'operació d'escriptura
 - Operativa
 - 1) Establir el valor de les entrades D dels biestables
 - 2) Activar el senyal d'escriptura
 - 3) L'escriptura es farà efectiva en el flanc actiu del senyal de rellotge
- Operació de lectura
 - Operació no destructiva
 - La dada emmagatzemada roman inalterada
 - Operativa
 - Examinar les eixides Q dels biestables

- Agrupació de registres
 - Permet l'escriptura d'una dada sobre un registre
 - Cal seleccionar quin registre volem escriure
 - Aquesta informació és l'adreça **d'escriptura**
 - Internament, un descodificador selecciona quin registre treballa
 - Permet la lectura d', almenys, una dada emmagatzemada
 - Cal seleccionar quin registre volem llegir
 - Aquesta informació és l'adreça **de lectura**
 - Internament, un multiplexor selecciona quina dada s'obté
 - Per a permetre la lectura simultània de dues o més dades és necessari disposar de tantes entrades d'adreça de lectura i multiplexors interns com a dades es vulguen llegir

- Exemple: Banc de registres de 4 registres de 8 bits amb un port (via d'accés) d'escriptura i un altre de lectura
 - 4 registres => 2 Bits d'adreça
 - Dades d'entrada/eixida de 8 bits
 - Símbol lògic



- Exemple (cont):
 - Esquema intern



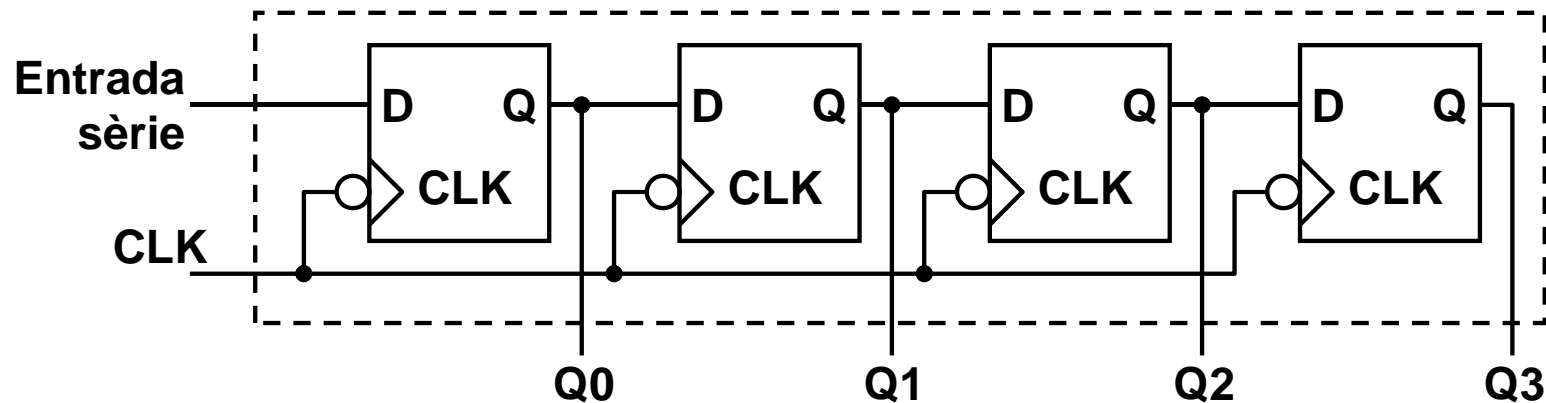
- Operació d'escriptura
 - Operativa
 - 1) Establir el valor de la dada d'escriptura
 - 2) Establir el valor de l'adreça d'escriptura
 - 3) Activar el senyal d'escriptura
 - 4) L'escriptura es farà efectiva en el flanc actiu del senyal de rellotge
- Operació de lectura
 - Operativa
 - 1) Establir el valor de l'adreça de lectura
 - 2) Examinar el valor de la dada de lectura

- Mateixa funcionalitat que un Banc de Registres, però
 - Molta més capacitat (Kbytes, Mbytes, Gbytes, ...)
 - Molt més lenta
 - Tecnologia diferent
- Una única operació (lectura/escriptura) en un moment donat
 - Una única entrada d'adreça i una altra de dada
 - Línies d'ordres de lectura i escriptura
 - Perquè la memòria sàpiga què fer en cada moment
 - Llegir, escriure o no res (si no s'activa ni lectura ni escriptura)



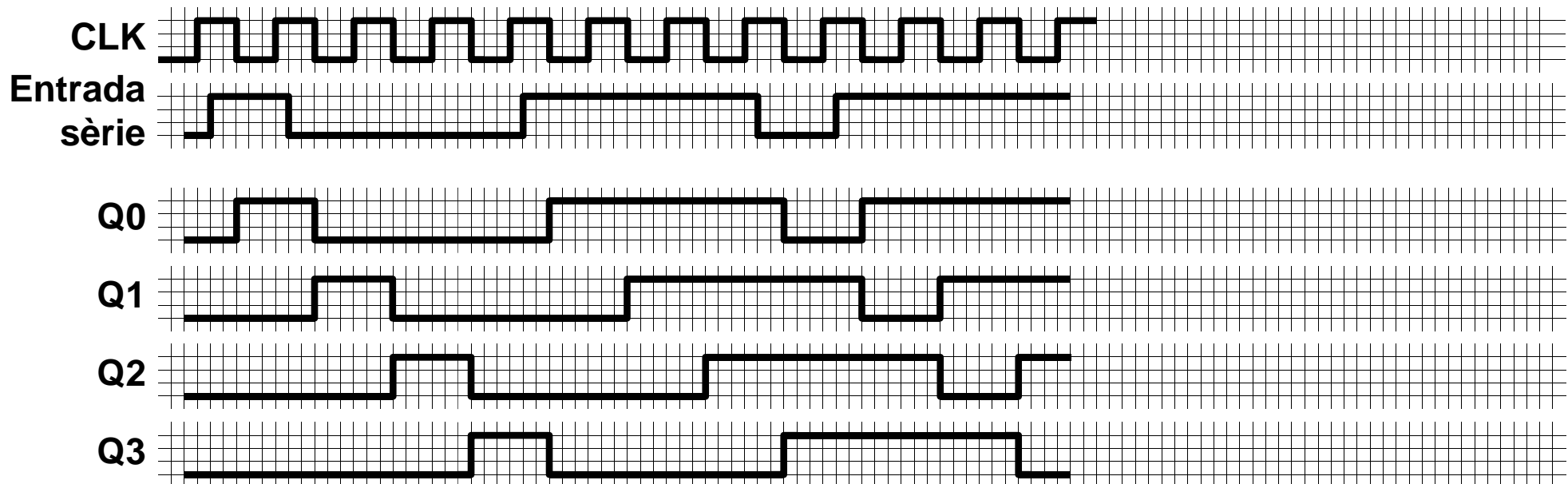
- Agrupació síncrona de biestables D per flanc
 - Tants biestables D com bits vulguem emmagatzemar
 - La informació necessita diversos cicles de rellotge per a entrar (escriptura) o eixir (lectura)
 - Es coneixen com a entrada sèrie i eixida **sèrie**, respectivament
 - Quan tots els bits entren o ixen en el mateix cicle de rellotge es diu que el registre té entrada o eixida **paral·lela**
 - Estructura amb entrada sèrie
 - El primer biestable connecta la seua entrada amb l'única entrada de dades
 - La resta, cada entrada amb l'eixida del biestable anterior
 - Estructura amb eixida sèrie
 - Només és accessible l'eixida de l'últim biestable

- Entrada sèrie, eixida paral·lela
 - Esquema intern

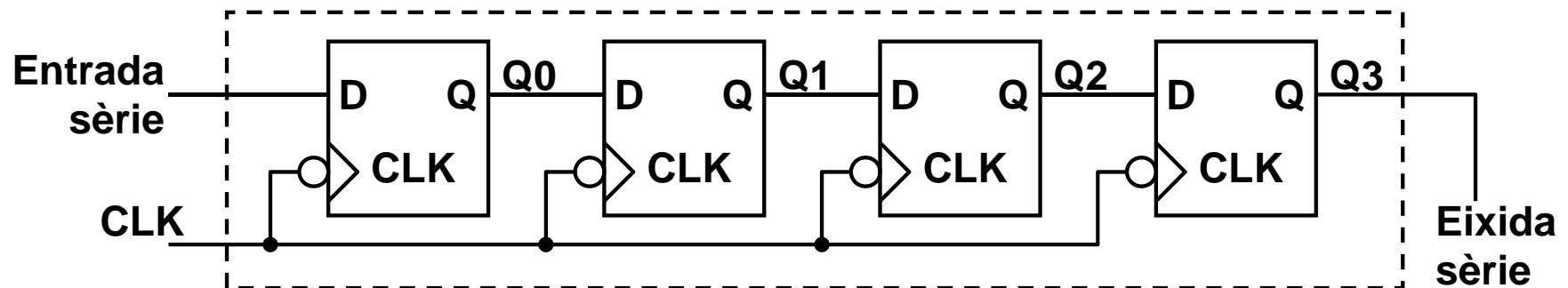


- Funcionament
 - Un bit (el valor de l'entrada sèrie) entra per l'esquerra i desplaça la informació emmagatzemada una posició cap a la dreta
 - El bit emmagatzemat en l'extrem dret es perd

- Entrada sèrie, eixida paral·lel (cont.)
 - Exemple de funcionament
 - Estat inicial $Q_3Q_2Q_1Q_0 = 0000$
 - L'entrada sèrie pren els valors indicats en el cronograma (com a exemple de seqüència de valors)



- Entrada sèrie, eixida sèrie
 - Mateix circuit intern que amb eixida paral·lela, excepte que l'única eixida disponible és la de l'últim biestable

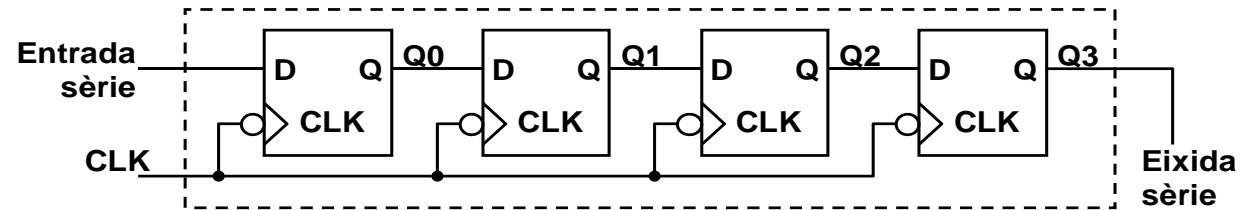


- Entrada paral·lela, eixida sèrie
 - Necessitem una entrada addicional (que podem anomenar, per exemple, LOAD) per a determinar si el sistema ha de carregar la dada d'entrada (LOAD=1) o desplaçar (LOAD=0)

- Registres de desplaçament (resum)
 - Un registre de desplaçament pot desplaçar
 - A esquerreres o a dretes
 - No hi ha unanimitat en el que significa “a esquerreres” o “a dretes”
 - Millor indicar el sentit explícitament
 - Fins i tot pot desplaçar en ambdós sentits
 - No simultàniament, de vegades a esquerreres i de vegades a dretes
 - Amb una entrada de control addicional que indique el sentit
 - En aquest cas pot disposar d’
 - » Una única entrada de dades sèrie
 - » Una entrada de dades sèrie per a cada un dels sentits de desplaçament (entrada sèrie per l'esquerra, entrada sèrie per la dreta)

- Registres de desplaçament (resum, cont.)
 - L'entrada de dades pot ser
 - Sèrie: quan l'entrada de dades és d'un bit per cicle de rellotge
 - Paral·lela: quan tot el registre s'escriu en un únic cicle de rellotge
 - L'eixida pot ser
 - Sèrie: quan per a observar el valor emmagatzemat en el registre necessitem tants cicles de rellotge com biestables
 - Paral·lela: quan podem observar el valor de tots els biestables en un únic cicle de rellotge

- Qüestions:



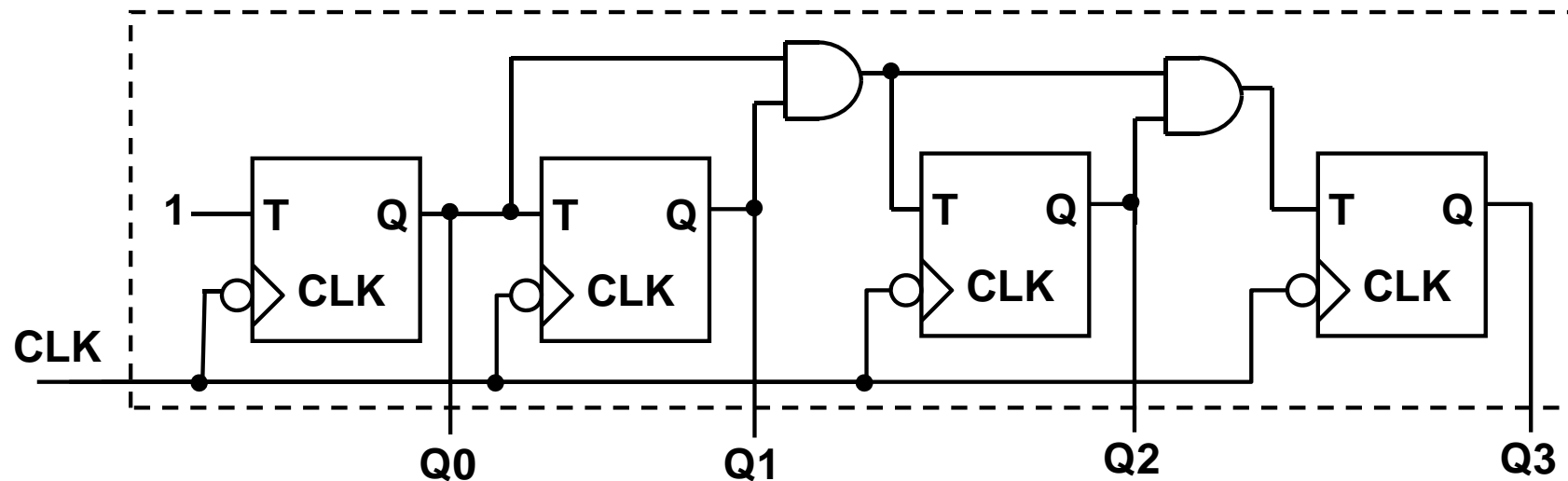
- Com fer un registre de desplaçament amb entrada i eixida sèrie, amb desplaçament de Q3 a Q0?
- I que es pugui triar el sentit del desplaçament amb un senyal anomenat “sentit” i amb dues entrades sèrie?
- Ara, con li afegim càrrega paral·lela?
- I fer-lo cíclic de vegades sí i de vegades no?

¿Multiplexors?

- Comptadors síncrons
 - Circuit que canvia de valor de forma autònoma en cada cicle de rellotge seguint una seqüència, generalment un compte binari (ascendent o descendent)
 - El compte pot ser ascendent o descendent
 - El compte també pot ser ascendent/descendent (reversible)
 - No simultàniament, de vegades ascendent i de vegades descendent
 - Una entrada de control addicional indica el sentit

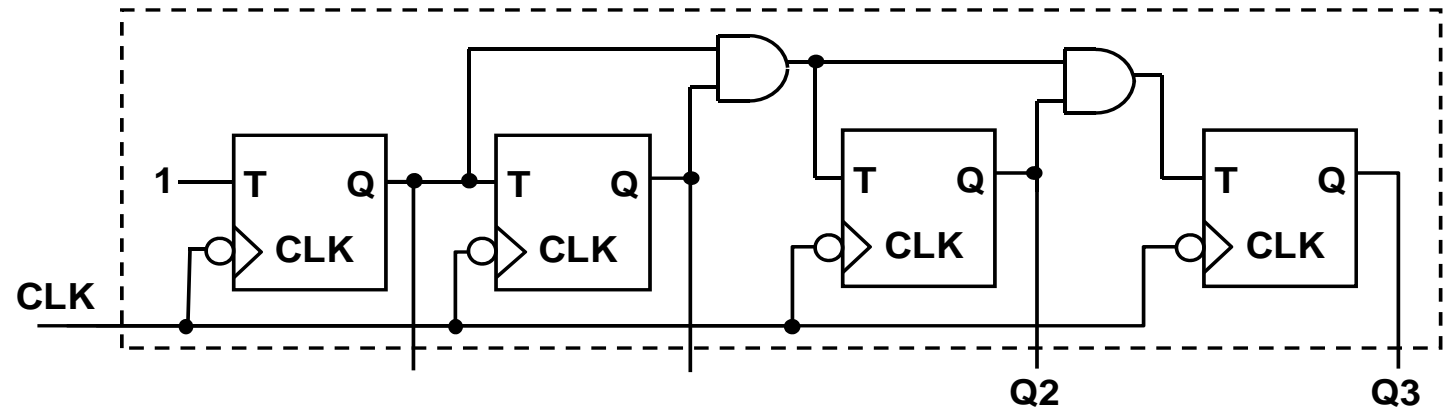
- Comptadors síncrons

- Solen estar construïts amb biestables T o J-K (amb $J=K$)
 - El circuit resultant utilitza menys portes lògiques que si s'utilitzen biestables D
- Exemple: Comptador síncron ascendent de 4 bits
 - Eixida 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 0, ...



- Comptadors: classificació pel tipus de compte
 - Binaris
 - Fan tots els comptes possibles entre 0 i $2^{\text{variables d'estat}} - 1$
 - De mòdul N
 - Fan N comptes distints, sent $N < 2^{\text{variables d'estat}}$
 - Solen incloure el compte 0...00, comptant llavors de 0 a N-1
 - Exemple
 - Un comptador de dècades és un comptador de mòdul 10 que compta de 0 a 9

- Qüestions:



- Com fer un comptador síncron descendent de 4 bits amb biestables T?
- I ara, com fer un comptador síncron de vegades ascendent i de vegades descendent, controlant el sentit amb un senyal anomenat “PUJA”?

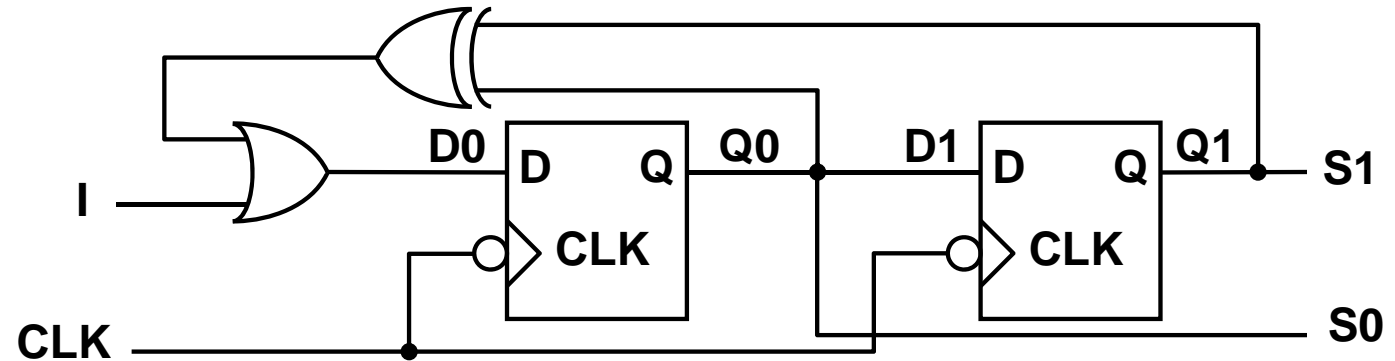
¿Més Multiplexors?

- Anàlisi de sistemes seqüencials
 - Consisteix a obtenir l'eixida del sistema a partir del circuit
 - Es poden emprar diferents mètodes
 - Cronograma
 - Obté l'eixida del sistema per a una seqüència d'entrades particular
 - Altre mètode: al tema següent

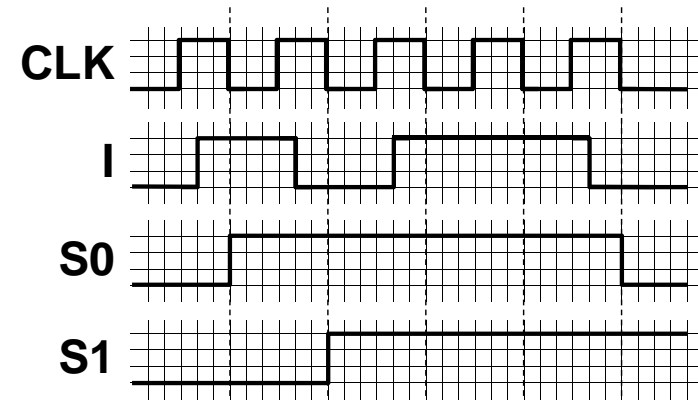
- Anàlisi per cronograma
 - Diagrama temporal
 - Incorpora totes les entrades i eixides del sistema
 - Pot ser d'utilitat per a simplificar l'anàlisi, l'afegir també senyals interns del circuit
 - Necessitem
 - El circuit
 - L'estat inicial del sistema (si no es pot deduir des de les entrades)
 - Una seqüència d'entrades
 - Obtenim
 - La seqüència d'eixides del sistema per a la seqüència d'entrades concreta que hem emprat en l'anàlisi

- Anàlisi per cronograma
 - Per a realitzar l'anàlisi hem de recórrer a les taules de veritat
 - De cada biestable del circuit
 - Per a cadascuna de les combinacions d'entrada que tinga al llarg del temps
 - Hem d'aplicar les combinacions d'entrada una per una en el temps
 - Perquè el nou valor (d'un biestable) pot influir en el comportament futur (del mateix biestable o d'un altre)

- Exemple
 - Analitzar



- Valor inicial $Q1Q0 = 00_2$
- Seqüència d'entrada $I = 1 - 0 - 1 - 1 - 0$
- Cronograma



(S1S0 en decimal) Eixida 0 1 3 3 3 2



UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA



Fonaments de computadors

TEMA 4. CIRCUITS SEQÜENCIALS
