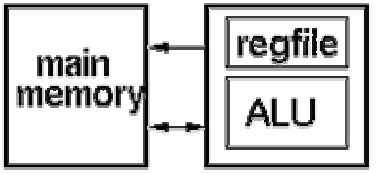
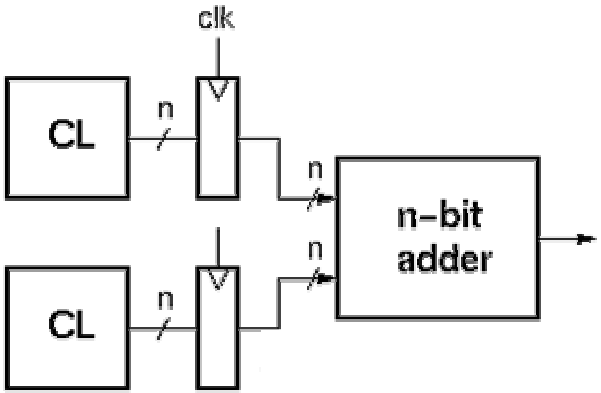
Elementi di memoria

Nella realizzazione di un sistema digitale è necessario utilizzare degli elementi di memoria per:

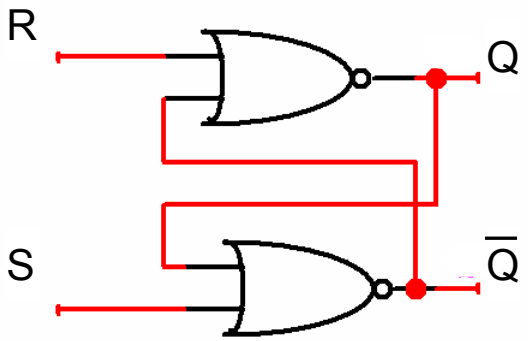
- memorizzazione di dati (registri per istruzioni/dati, buffers, control/status, flag)
- sincronizzazione tra dati

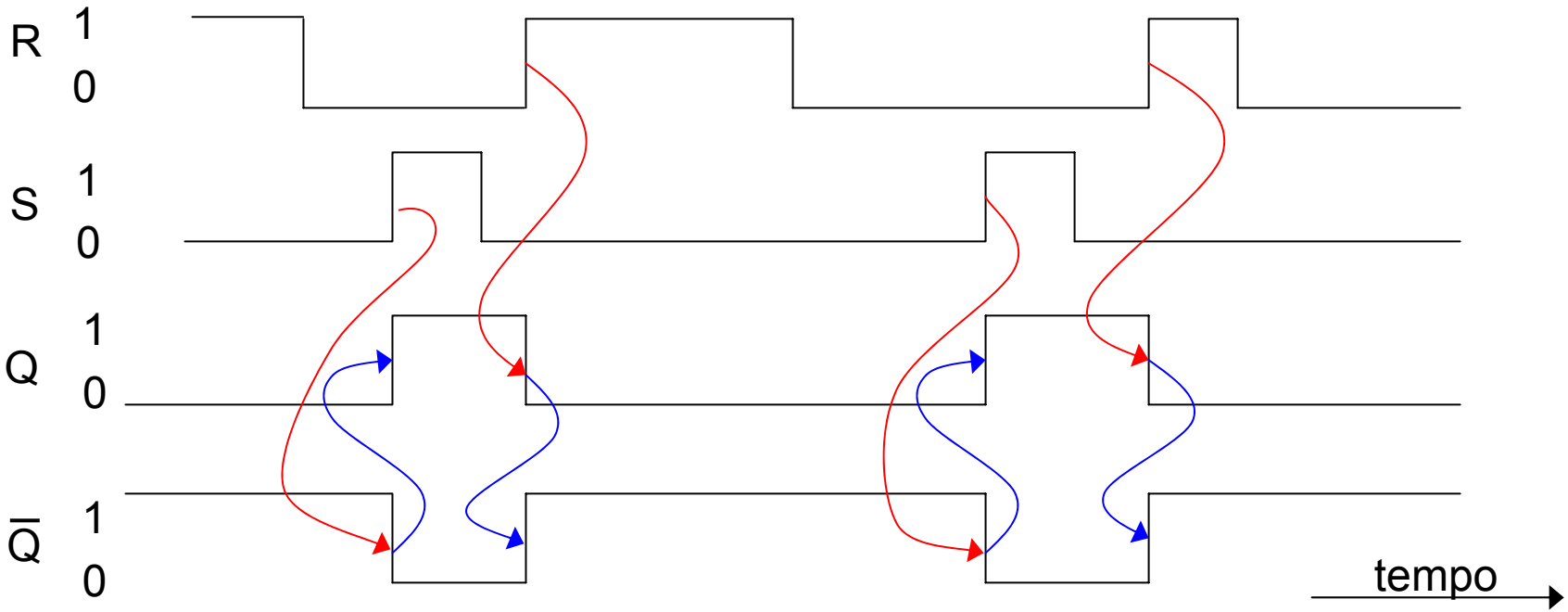




Elementi di memoria: RS-latch

S	R	Q	$\overline{Q}$
0	0	0/1	1/0
0	1	0	1
1	0	1	0
1	1	--	--





Elementi di memoria: D-latch

Un D-latch è un elemento che permette la memorizzazione di un valore binario.
Ha due ingressi:

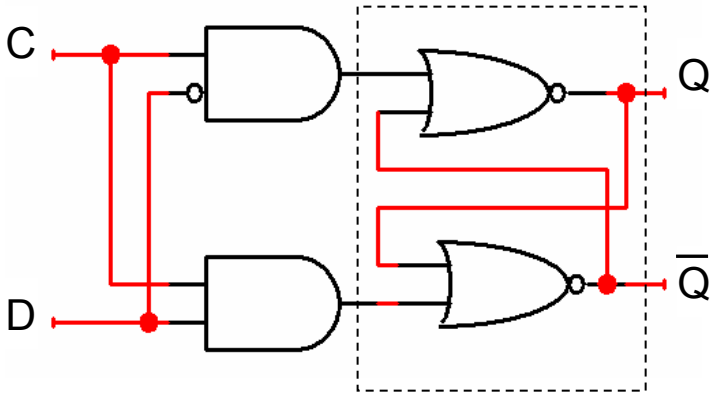
- D (*Data*): il valore da memorizzare
- C (*Clock*): un segnale di abilitazione alla lettura

In uscita presenta il valore memorizzato Q ed il suo complemento $\overline{Q}$

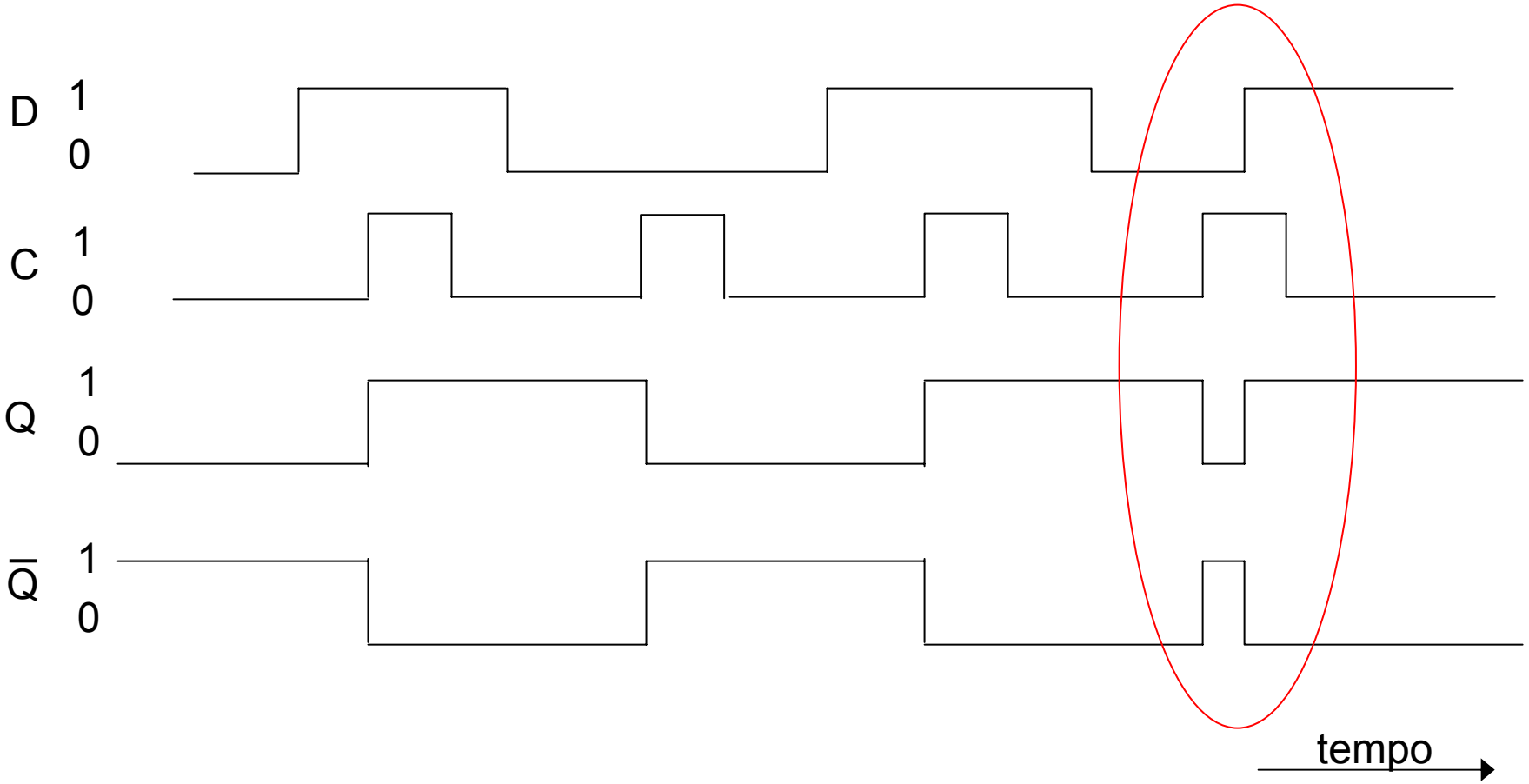
C	D	Q	$\overline{Q}$
0	0	0/1	1/0
0	1	0/1	1/0
1	0	0	1
1	1	1	0

C	D	R	S
0	0	0	0
0	1	0	0
1	0	1	0
1	1	0	1

Il D-latch può essere realizzato tramite un RS-latch



Andamento temporale del D-latch

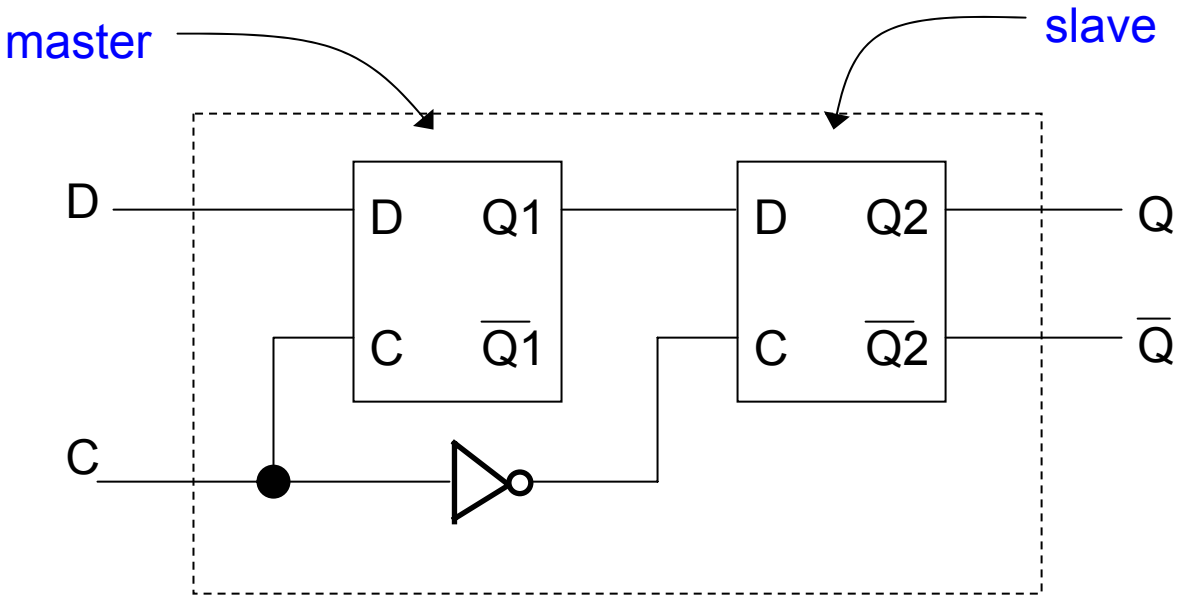


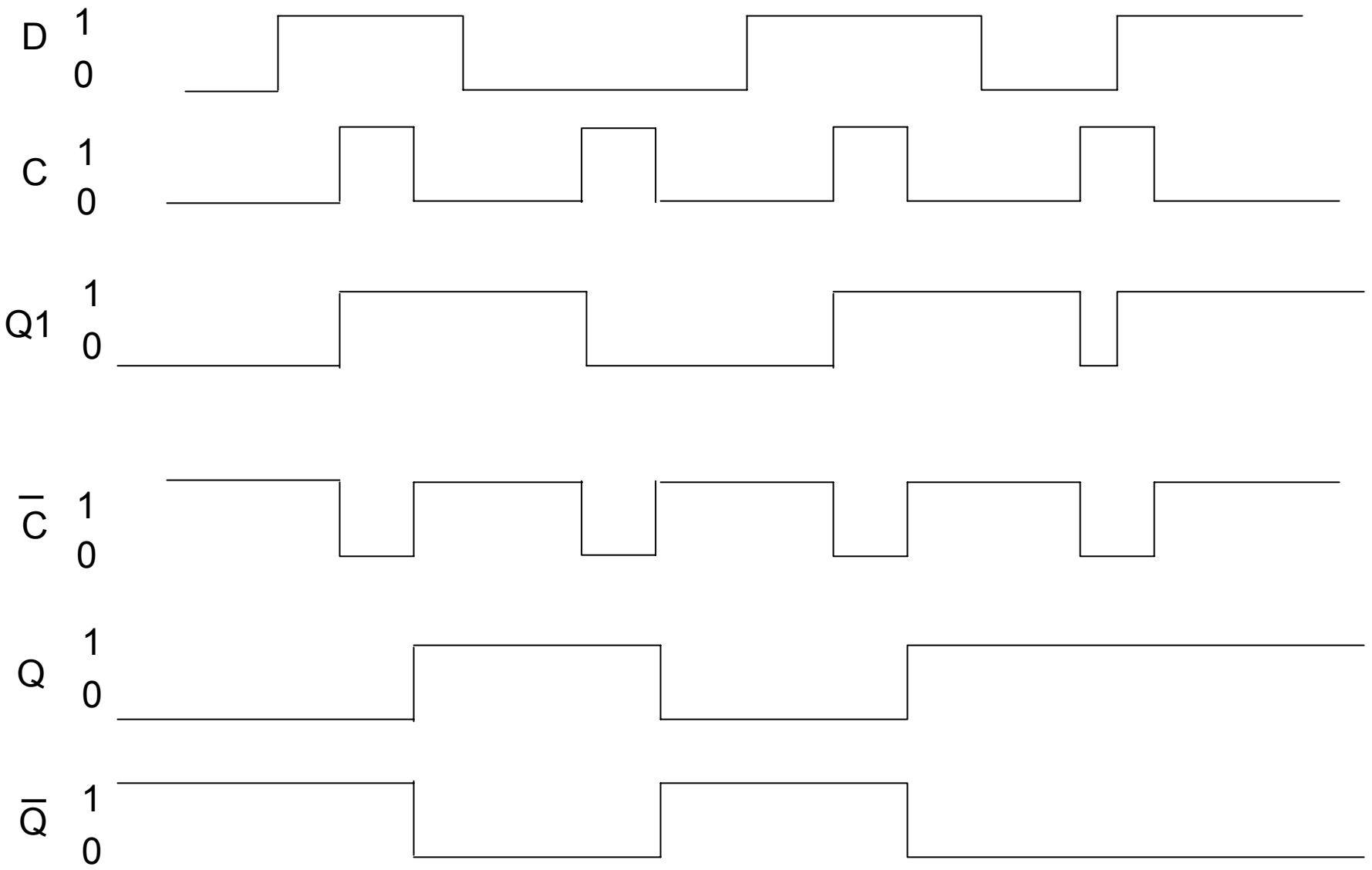
Eventuali variazioni di D mentre C è alto vengono seguite

Elementi di memoria: flip flop D edge triggered

Gli elementi di memoria edge-triggered modificano il loro stato in corrispondenza del fronte di salita o di discesa del clock. In tal modo si evitano i problemi dei latch.

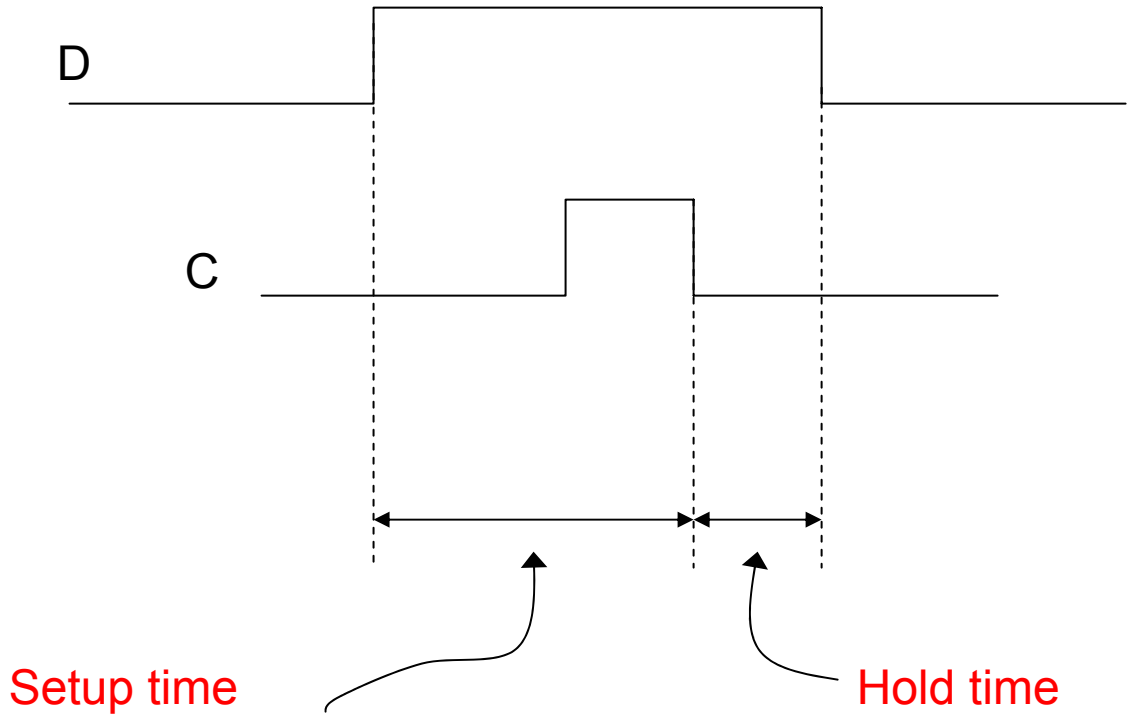
Realizzazione *Master-Slave*




D
1
0
C
1
0
Q1
1
0
$\overline{C}$
1
0
Q
1
0
$\overline{Q}$
1
0

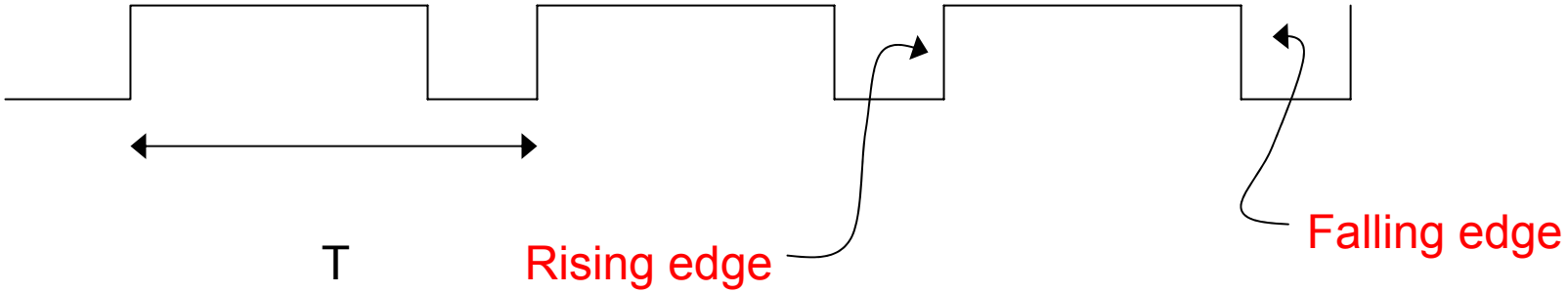
L’ingresso D deve essere stabile in un intervallo di tempo intorno al fronte del segnale C, altrimenti l’uscita del flip flop non è affidabile

Sono quindi definiti dei tempi minimi di durata del segnale D prima del fronte (*setup time*) e dopo il fronte (*hold time*)



Clock

Il clock è un segnale periodico, definito da una propria frequenza f=1/T



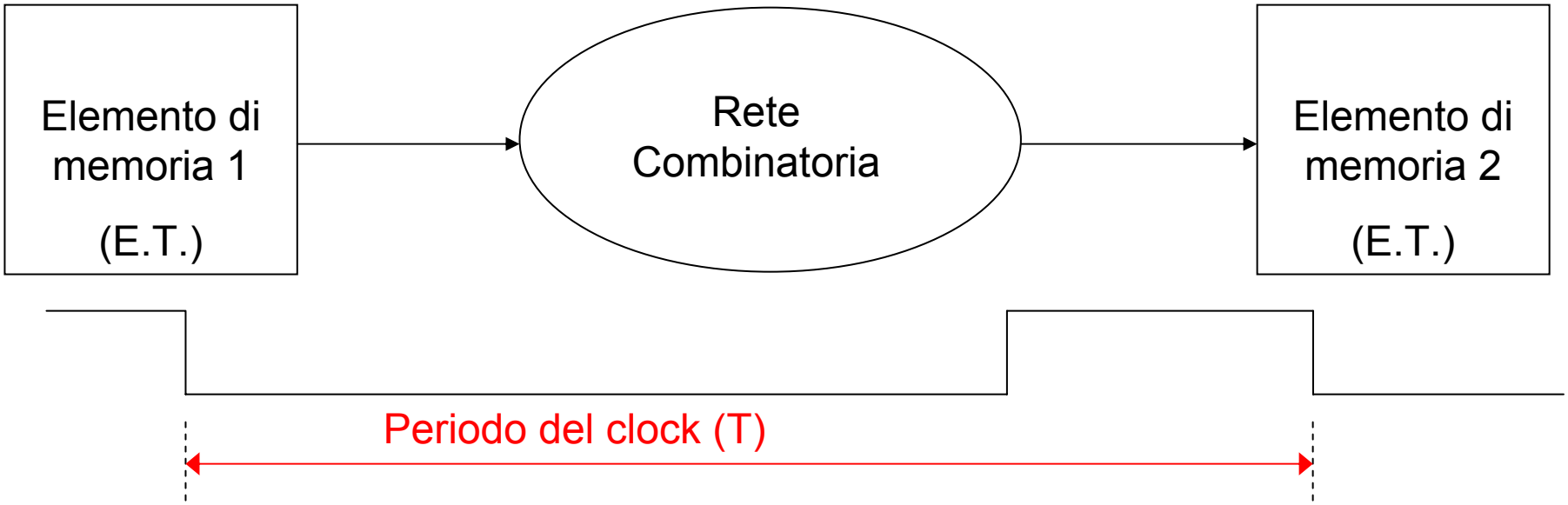
Può assumere solo due valori (clock alto o clock basso). La transizione tra un valore ed un altro definisce un fronte di salita (*rising edge*) o di discesa (*falling edge*).

In un sistema digitale *sincrono* il segnale di clock viene utilizzato per determinare i cambiamenti di stato negli elementi di memoria.

Quando ciò avviene in corrispondenza di un fronte si parla di *sincronizzazione edge triggered* (*edge triggered clocking*)

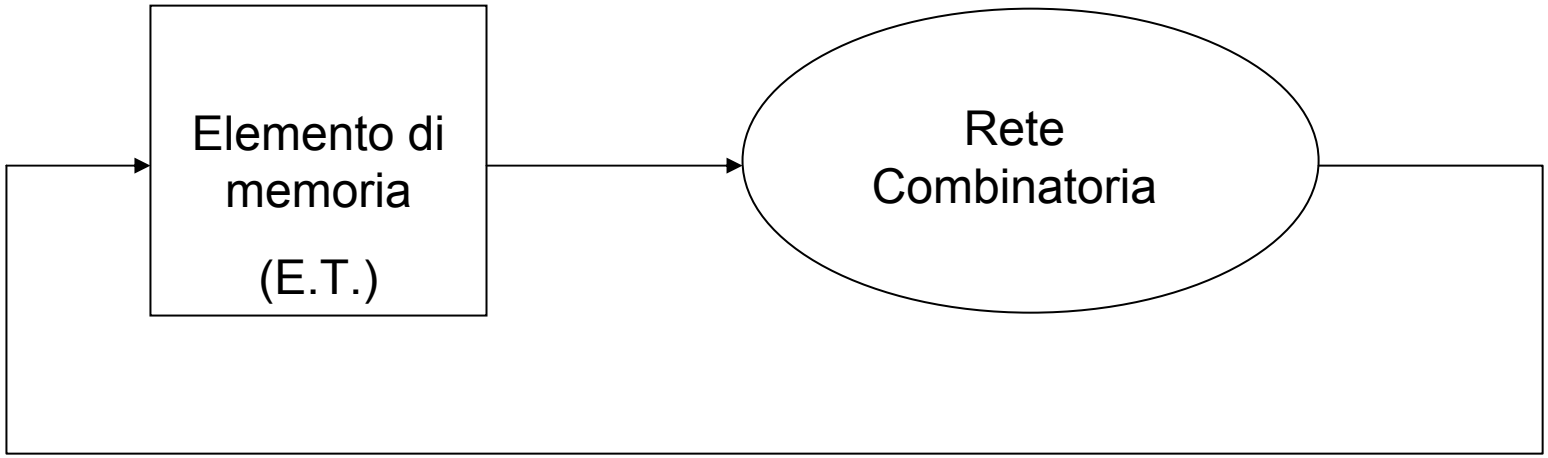
Il Clock in un sistema digitale sequenziale (1)

I segnali che sono ingresso di elementi di memoria devono essere stabili quando si verifica il fronte del clock.



Per assicurare la stabilità dei valori letti all'uscita della rete combinatoria, il periodo del clock deve essere sufficientemente lungo. Se t_d è il ritardo legato al percorso critico della rete combinatoria, allora $T > t_d$.

Il Clock in un sistema digitale sequenziale (2)

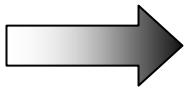


Con una sincronizzazione di tipo edge triggered è possibile realizzare delle retroazioni sugli stessi elementi di memoria.

Elementi di memoria: flip flop JK

Il flip flop JK coincide con il flip flop RS tranne che per la combinazione J=K=1, in cui lo stato prossimo è definito come il complemento dello stato presente (contatore modulo 2).

J	K	Q_{n+1}
0	0	Q_n
0	1	0
1	0	1
1	1	$\overline{Q_n}$



$$\begin{cases} R = K\,Q_n \\ S = J\,\overline{Q_n} \end{cases}$$

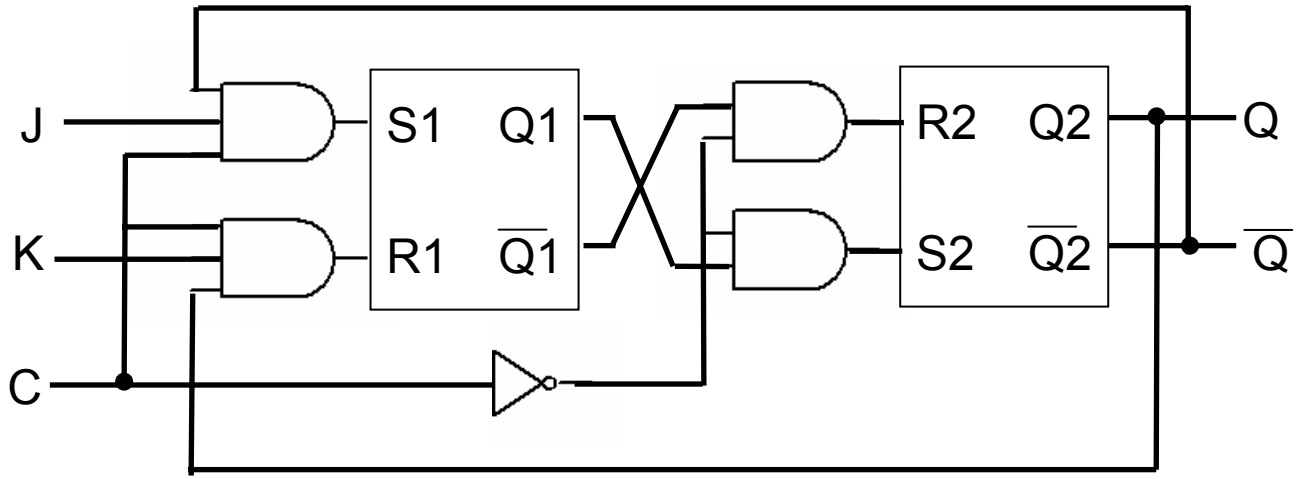
Retroazione delle uscite

La retroazione diretta delle uscite non è realizzabile perché potrebbe condurre ad uno stato diverso da quello previsto o indurre un processo di oscillazione (se J=K=1).

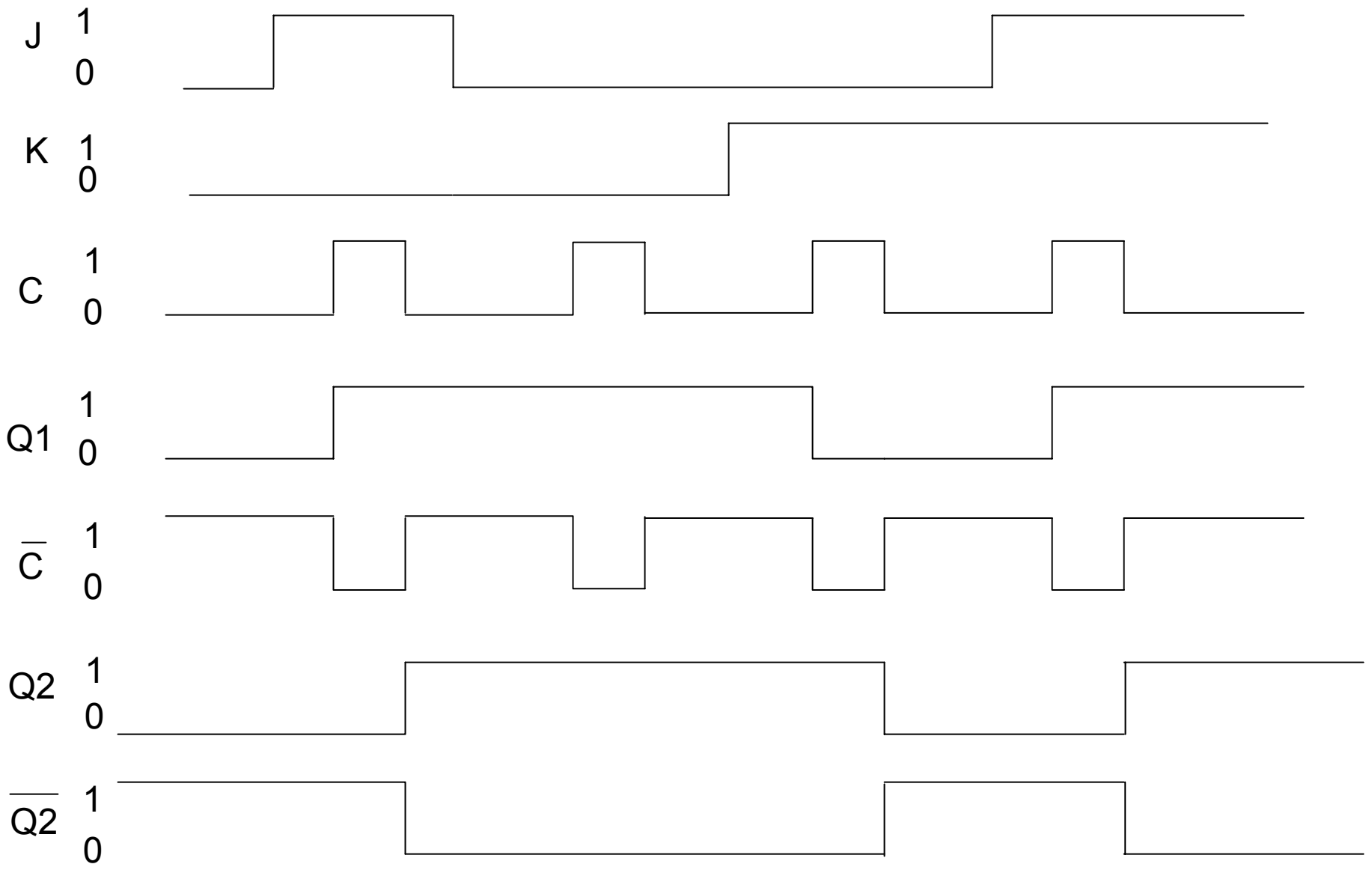
Per avere una realizzazione efficiente è necessario separare logicamente le uscite correnti dagli ingressi. **Soluzione ?**

Realizzazione di flip flop JK

Una possibile realizzazione si ottiene tramite uno schema master-slave con bistabili RS abilitati.

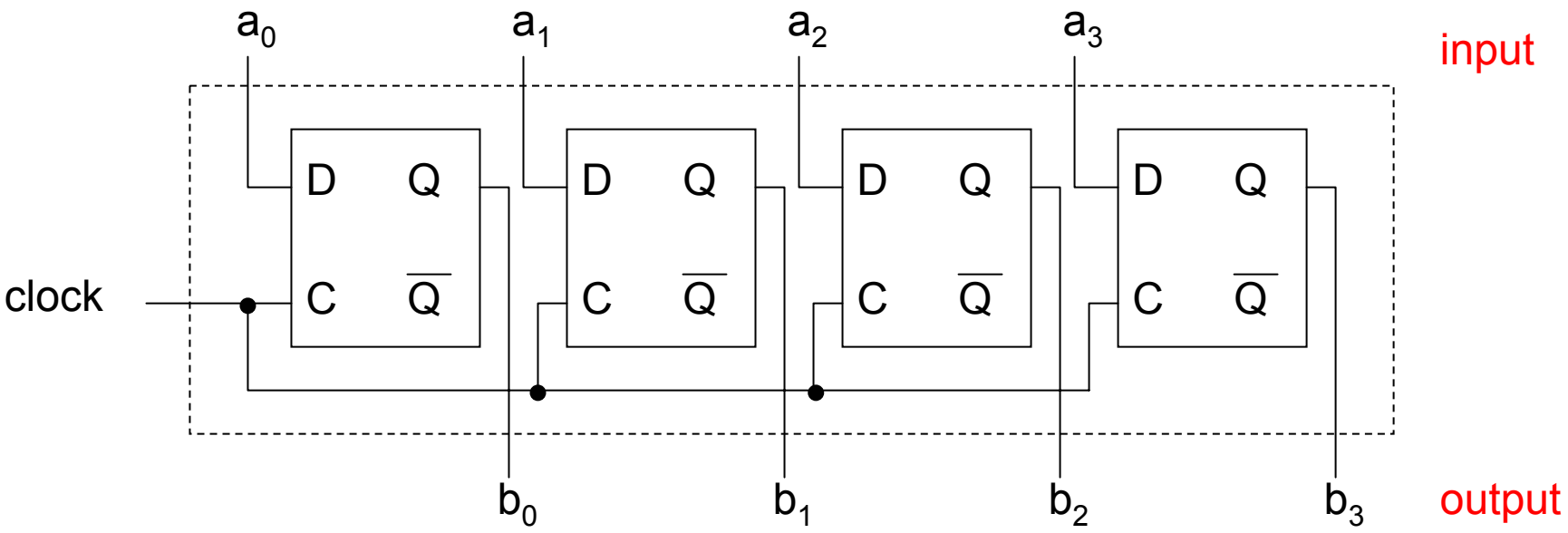


Il primo stadio separa logicamente le uscite correnti (Q2,$\overline{Q2}$) dagli ingressi (R2,S2) dello stadio finale che opera la commutazione dello stato dell'intero flip flop.


J
1
0
K
1
0
C
1
0
Q1
1
0
$\overline{C}$
1
0
Q2
1
0
$\overline{Q2}$
1
0

Registri

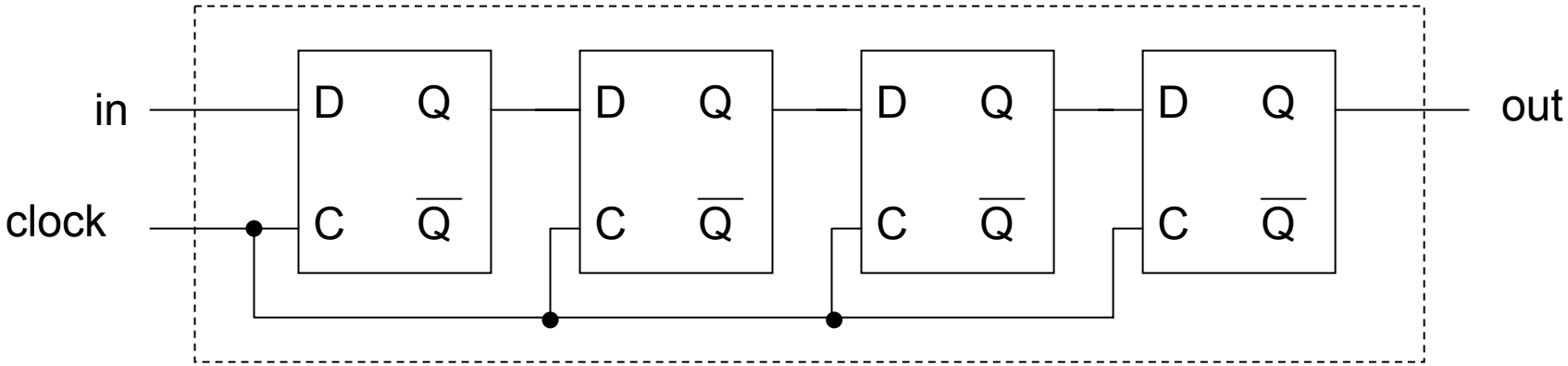
Un singolo bistabile è sufficiente per memorizzare un bit. Per memorizzare dati rappresentati su più bit è conveniente organizzare più bistabili in una struttura comune, il *registro*.



I bistabili sono sincronizzati da un clock comune in modo da caricare (o leggere) i dati in tutti i bistabili contemporaneamente.

Registri a scorrimento (1)

In diversi tipi di elaborazioni è necessario operare delle traslazioni (shift) o delle rotazioni sui contenuti di una registro. Un supporto hardware che rende possibile questa operazione è fornito dal registro a scorrimento (shift register).

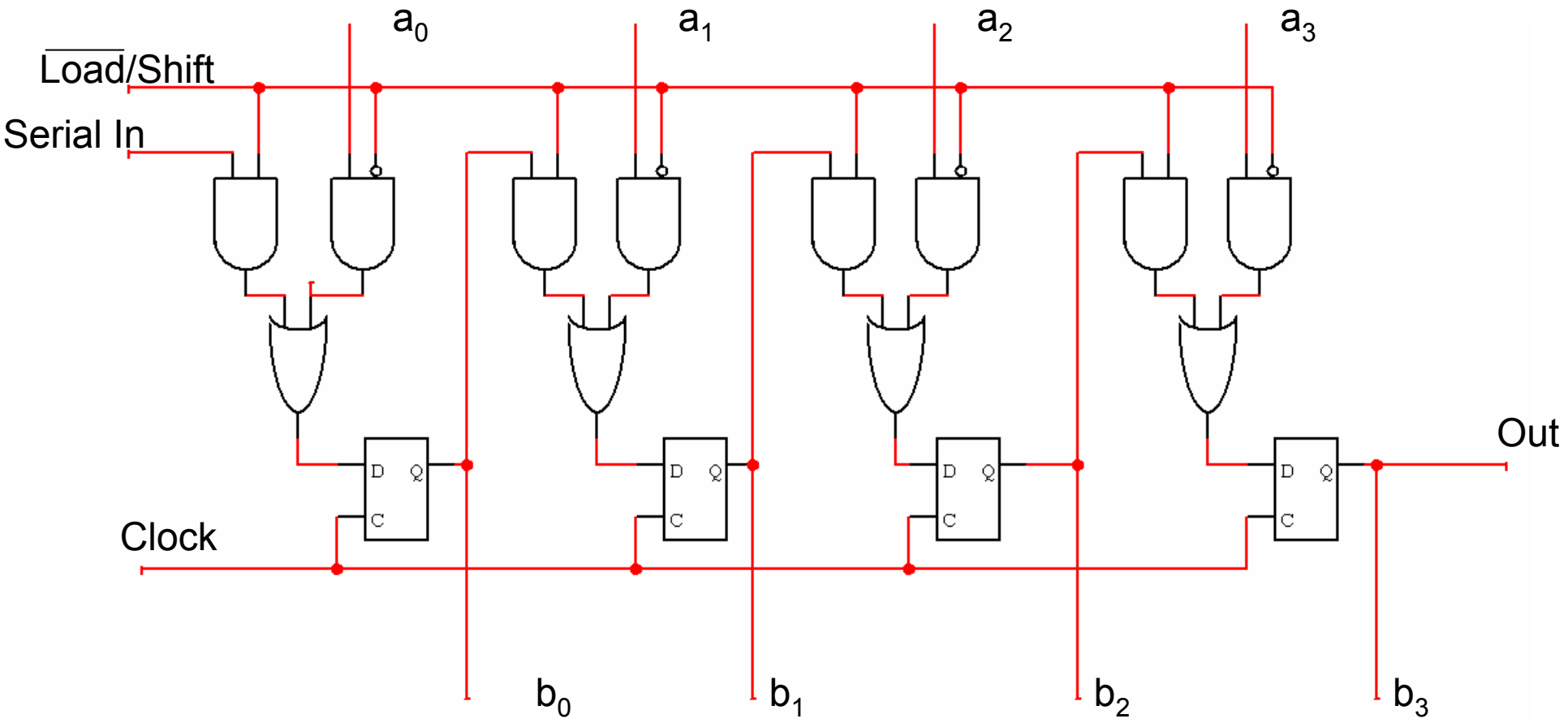


Ad ogni impulso di clock il contenuto del registro trasla di una posizione verso destra.

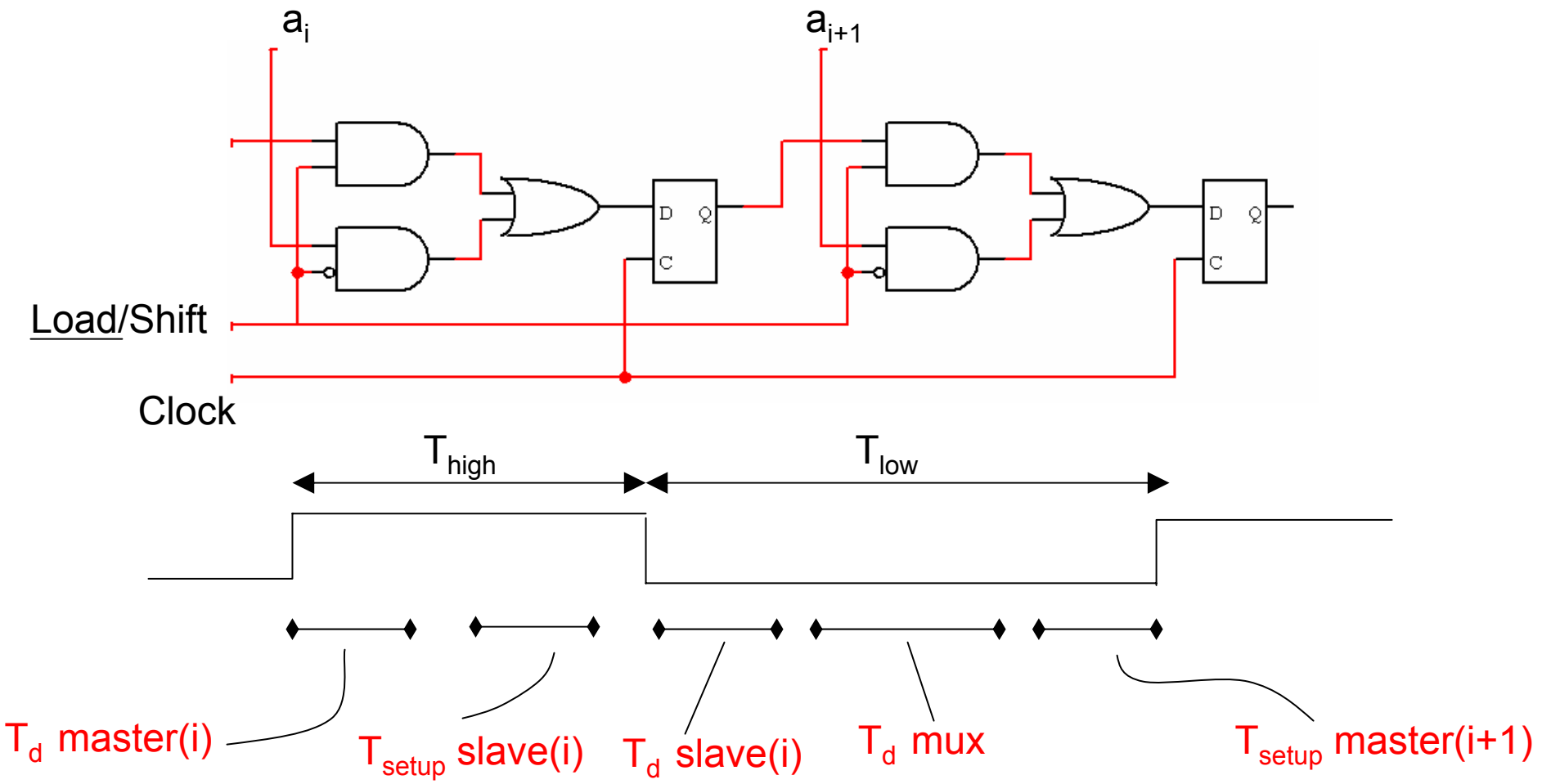
Quali sono i vincoli sulla tempificazione dei bistabili ?

Registri a scorrimento (2)

Un tipo particolarmente utile di registro a scorrimento è quello che consente la scrittura e la lettura del contenuto del registro anche in parallelo.



Definizione del periodo di clock



$T_{high} > T_d\text{master(i)} + T_{setup}\text{slave(i)}$

$T_{low} > T_d\text{slave(i)} + T_d\text{mux} + T_{setup}\text{master(i+1)}$

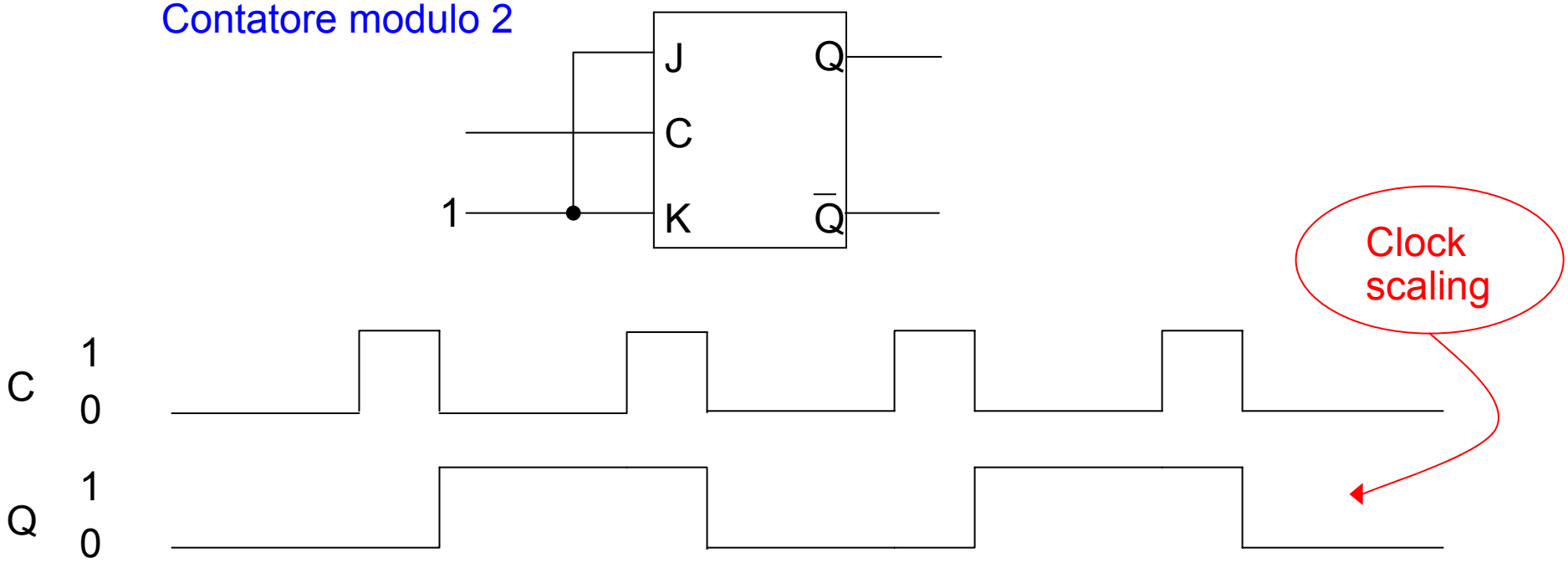
Contatori (1)

Circuiti fondamentali nella realizzazione di sistemi di elaborazione.

Forniscono, in maniera codificata, il numero delle volte che si è verificato un evento (es. falling edge di un segnale in ingresso).

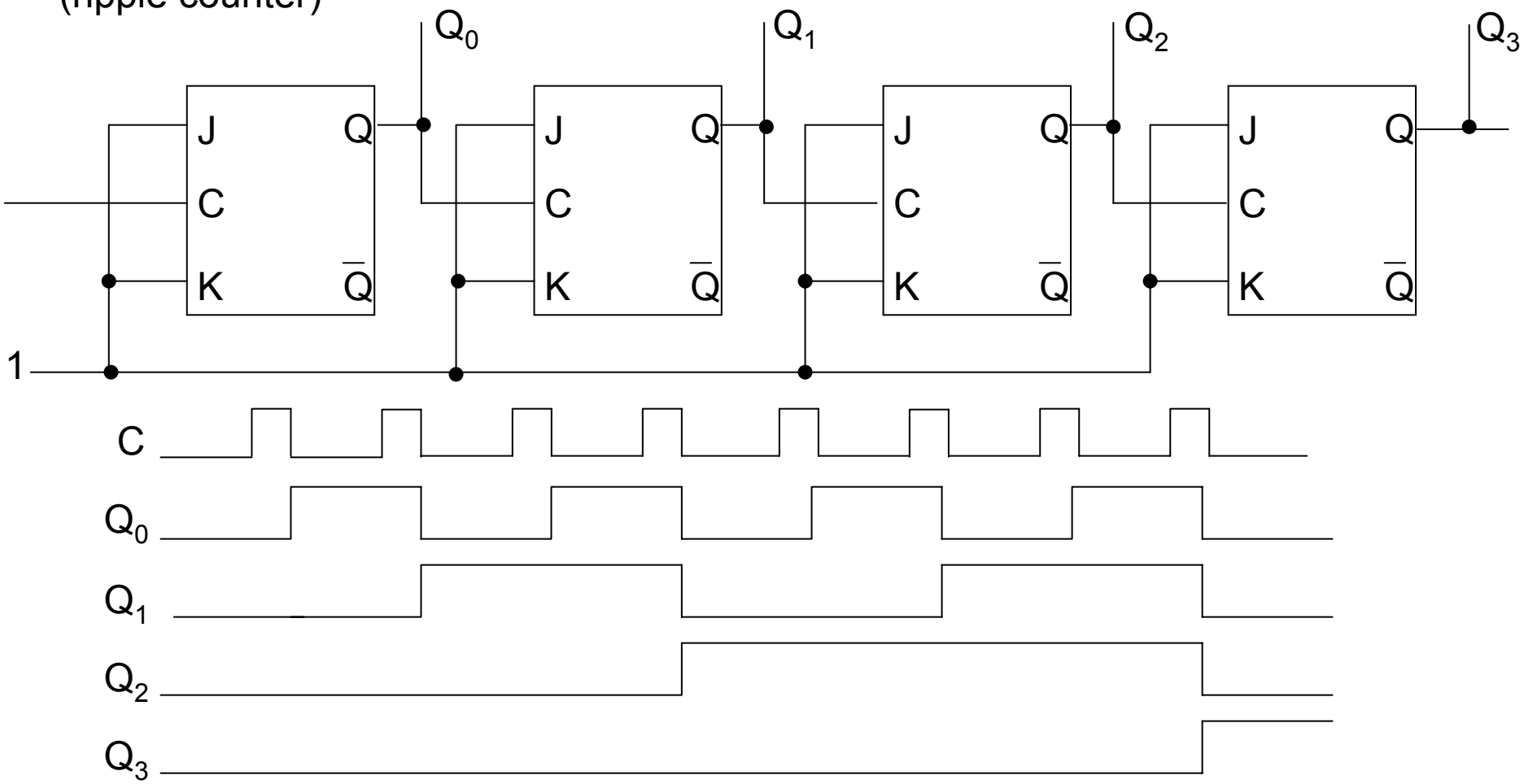
Il conteggio avviene mod 2^k, se l'uscita è su k bit.

Contatore modulo 2



Contatori (2)

Contatore mod 2^k realizzato tramite k contatori mod 2 montati in cascata (ripple counter)

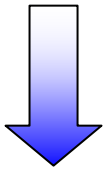


Reti sequenziali

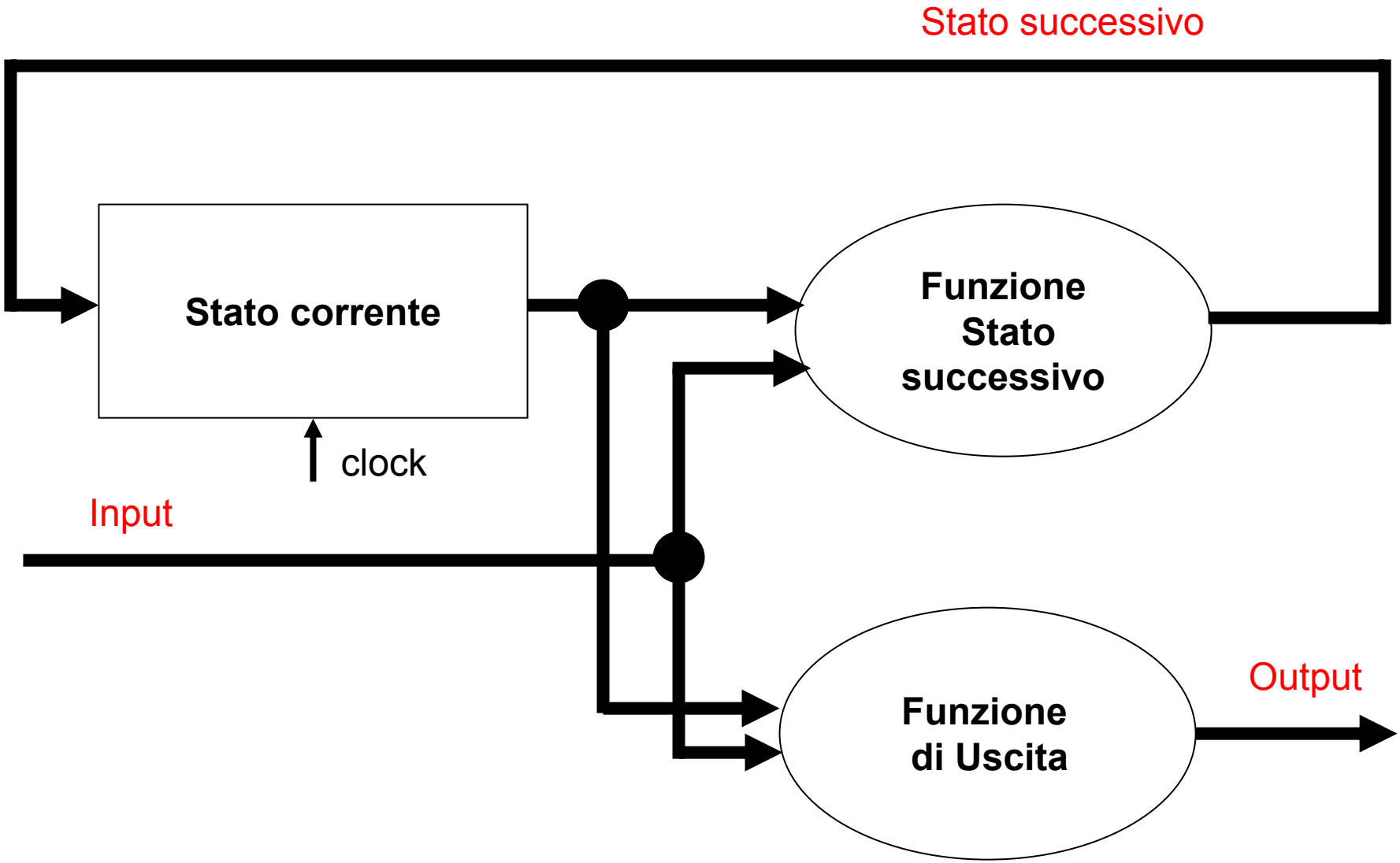
- una rete sequenziale è un circuito logico avente n ingressi (x_1,x_2,…,x_n) ed m uscite (y_1,y_2, …,y_m), ciascuno dei quali assume valori binari (0/1).
- ad ogni istante, il valore delle uscite dipende sia dagli ingressi presenti, sia dalla sequenza degli ingressi precedenti.
- in ogni istante, si può identificare uno *stato S* in cui la rete si trova, in funzione della sequenza degli ingressi precedenti.
- lo stato S è memorizzato su k bit (s_0,s_1,…,s_{k-1})
- ogni uscita può essere definita come una funzione booleana degli ingressi e dello stato presenti $y_i = y_i(x_1,x_2,\ldots,x_n, s_0,s_1,\ldots,s_{k-1})$.
- lo stato successivo è definito come una funzione booleana degli ingressi e dello stato presenti $s_j = s_j(x_1,x_2,\ldots,x_n, s_0,s_1,\ldots,s_{k-1})$.

Reti sequenziali

- una rete sequenziale è un circuito logico avente n ingressi (x_1,x_2,…,x_n) ed m uscite (y_1,y_2, …,y_m), ciascuno dei quali assume valori binari (0/1).
- ad ogni istante, il valore delle uscite dipende sia dagli ingressi presenti, sia dalla sequenza degli ingressi precedenti.
- in ogni istante, si può identificare uno *stato S* in cui la rete si trova, in funzione della sequenza degli ingressi precedenti.
- le uscite possono quindi essere definite come funzioni degli ingressi correnti e dello stato corrente.
- ugualmente, lo stato successivo può essere determinato in funzione dello stato corrente e degli ingressi correnti.



Modello di macchina a stati finiti


Stato successivo
Stato corrente
clock
Input
Funzione
Stato
successivo
Funzione
di Uscita
Output

Esempi di macchine sequenziali

- **Registri (RS, D, JK)**
- **Registri a scorrimento**
- **Contatori**

Come si definisce lo stato ?

Quanti stati diversi può assumere la macchina ?

Come si definisce la funzione stato successivo ?

Come si definisce la funzione di Uscita ?

Registro JK

Stato: valore di Q

Num. di stati: 2

F. di stato successivo:

F. di uscita: valore di Q

J=0 e K=0 ⇨ Q'=Q

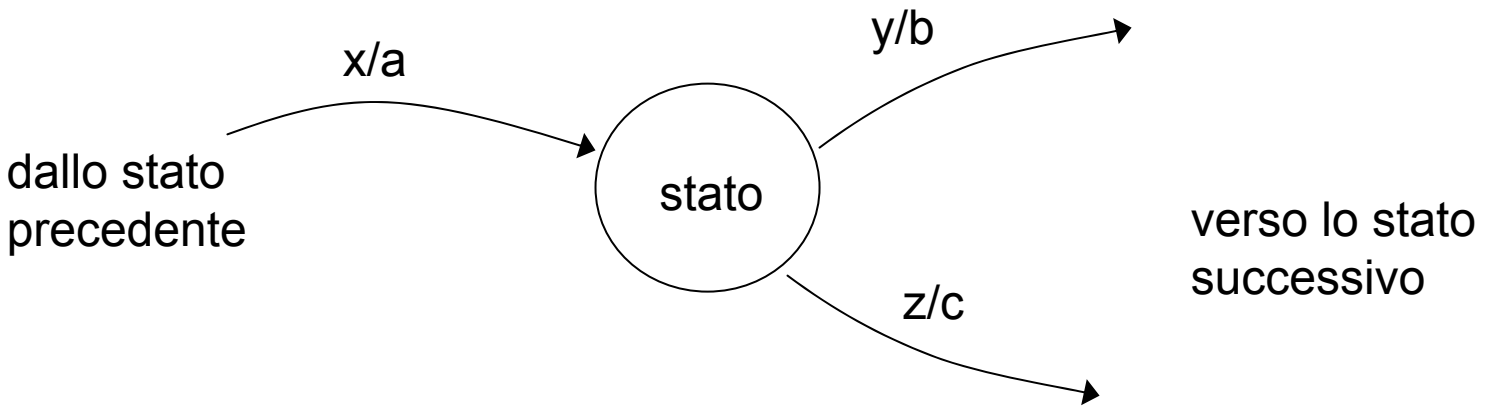
J=1 e K=0 ⇨ Q'=1

J=0 e K=1 ⇨ Q'=0

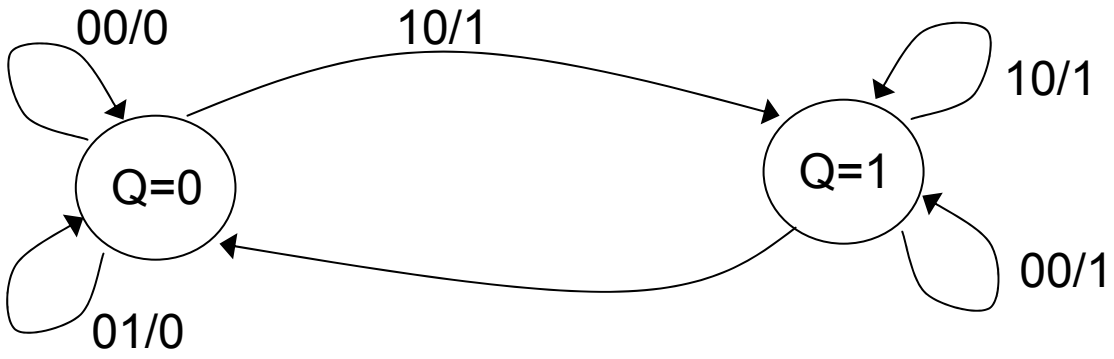
J=1 e K=1 ⇨ Q'=$\overline{Q}$

Diagramma di stato

Strumento per descrivere graficamente una macchina sequenziale.

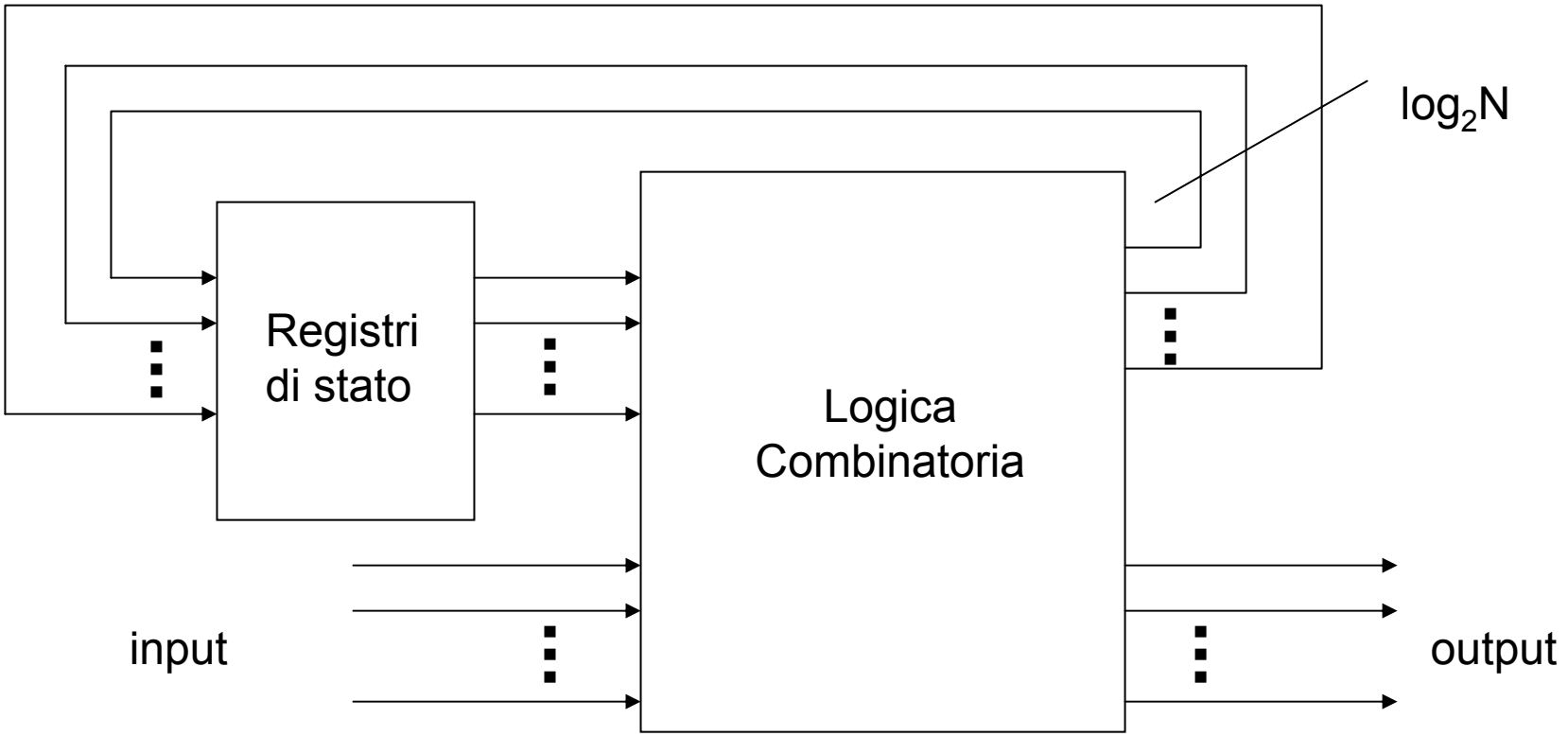


Registro JK

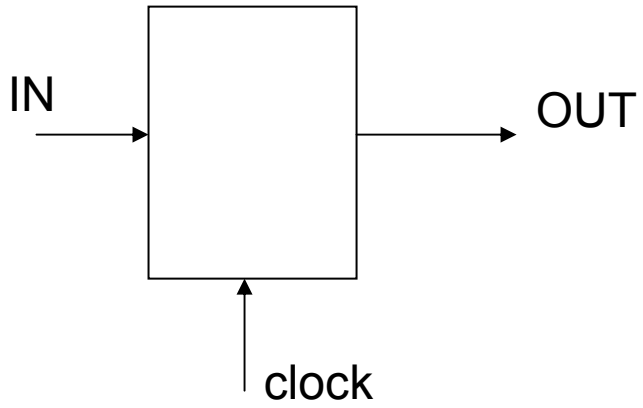


Realizzazione di una macchina a stati finiti

MSF ad N stati ⇨ lo stato è rappresentabile con $\log_2 N$ bit



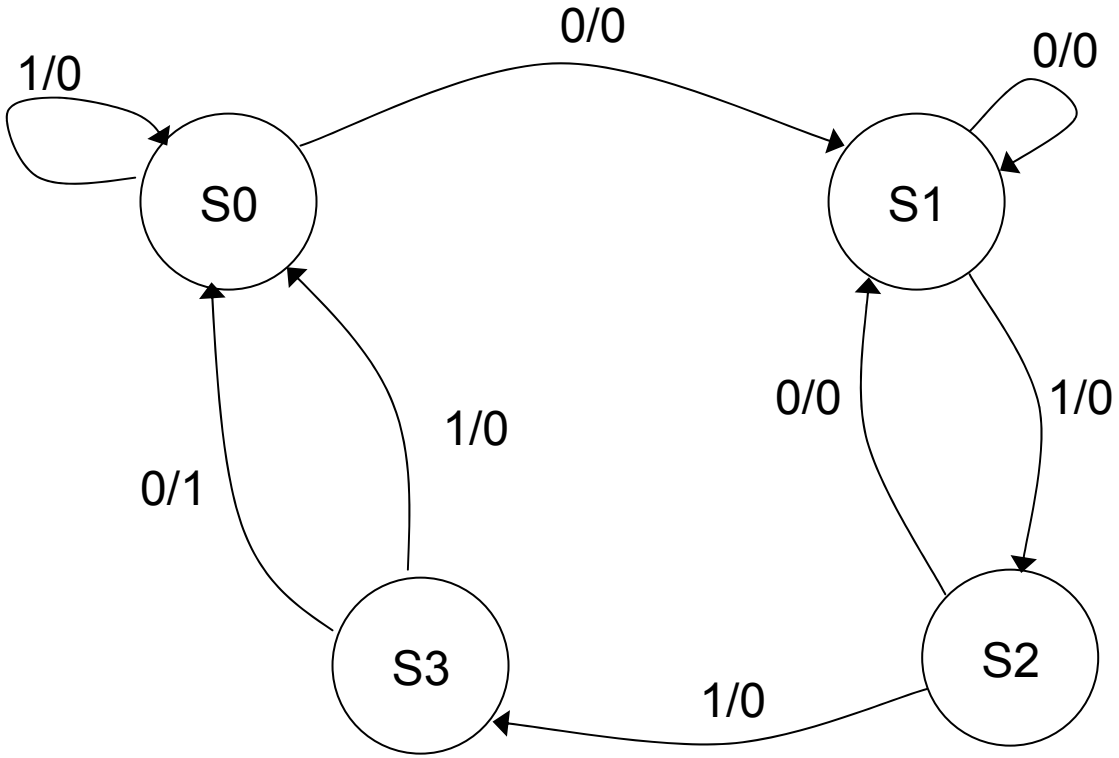
Esempio: riconoscitore di sequenza



Riconoscere una sequenza data (es. 0110)

- viene letta una cifra binaria da IN ad ogni ciclo di clock
- nel caso venga incontrata la sequenza cercata, OUT va a 1 in corrispondenza dell'arrivo dell'ultima cifra, altrimenti resta a 0.

Riconoscitore di sequenza: diagramma di stato



Riconoscitore di sequenza: codifica degli stati e definizione delle funzioni di stato successivo e di uscita

Stato	x0	x1
S0	0	0
S1	0	1
S2	1	0
S3	1	1

Funzione stato successivo

	IN = 0	IN = 1
S0	S1	S0
S1	S1	S2
S2	S1	S3
S3	S0	S0

Funzione uscita

	IN = 0	IN = 1
S0	0	0
S1	0	0
S2	0	0
S3	1	0

Riconoscitore di sequenza: definizione delle funzioni di stato successivo

Funzione stato successivo

	IN = 0	IN = 1
S0	S1	S0
S1	S1	S2
S2	S1	S3
S3	S0	S0

x0	x1	IN	x0'	x1'
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	0

Riconoscitore di sequenza: definizione delle funzioni di stato successivo

x0	x1	IN	x0’	x1’
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	1	1
1	1	0	0	0
1	1	1	0	0

	00	01	11	10
0				
1		1		1

	00	01	11	10
0	1	1		1
1				1

x0’ = !x0*x1*IN + x0*!x1*IN

x1’ = !x0*!IN + x0*!x1

Riconoscitore di sequenza: definizione della funzione di uscita

Funzione uscita

	IN = 0	IN = 1
S0	0	0
S1	0	0
S2	0	0
S3	1	0

x0	x1	IN	OUT
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

OUT = x0*x1*!IN

Circuito realizzato con flip-flop D