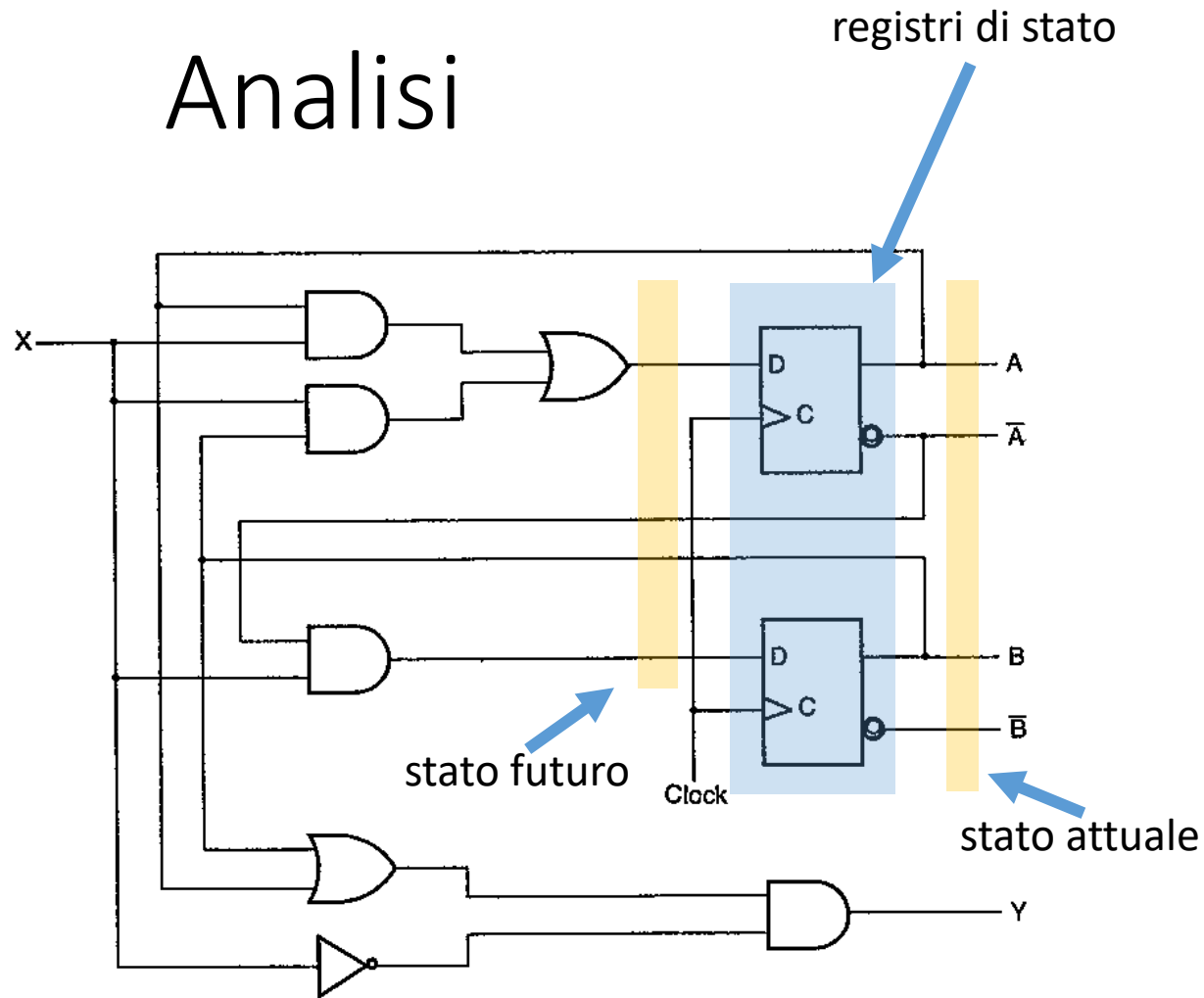


11- Sequential Circuits

Bibliografia



Analisi



A_t	B_t	X	A_{t+1}	B_{t+1}	Y
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

- Anche l'analisi di circuiti sequenziali *pulse-mode* avviene tramite *tabelle di transizione*, che permettono di calcolare stato futuro e uscite in funzione di stato presente e ingressi

Analisi

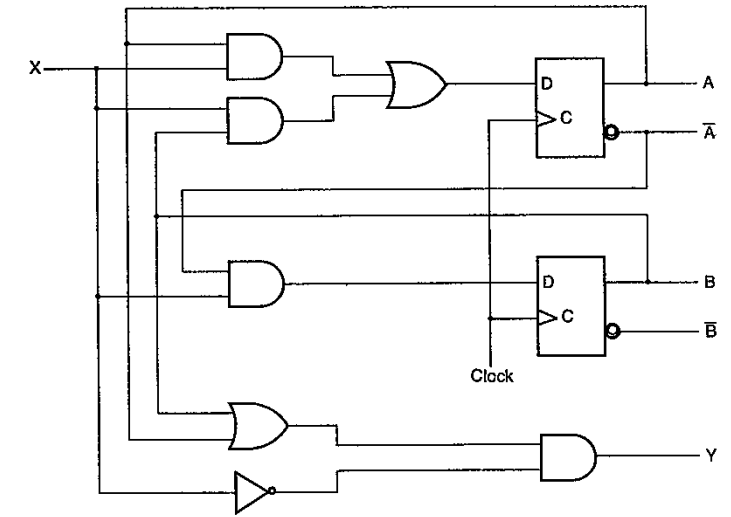
N bit di stato
(uscite dei FF)

N funzioni
(ingressi dei FF)

M ingressi

Stato presente		Ingresso	Stato futuro		Uscita
A	B	X	A	B	Y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

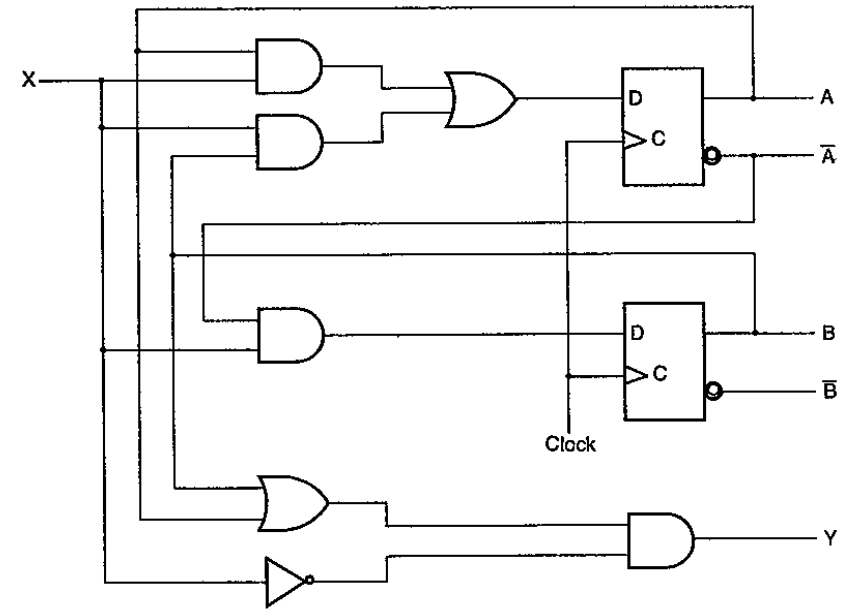
$2^{(N+M)}$ righe



- Con N bit di stato e M ingressi, la tabella possiede **$2^{(N+M)}$ righe**
- Se i registri di stato sono realizzati con D-FF, la tabella specifica N funzioni combinatorie, una per ogni ingresso dei FF
- Le uscite, in generale possono dipendere sia dagli ingressi, sia dai bit di stato

La tabella di transizione

Stato presente		Ingresso	Stato futuro		Uscita
A	B	X	A	B	Y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

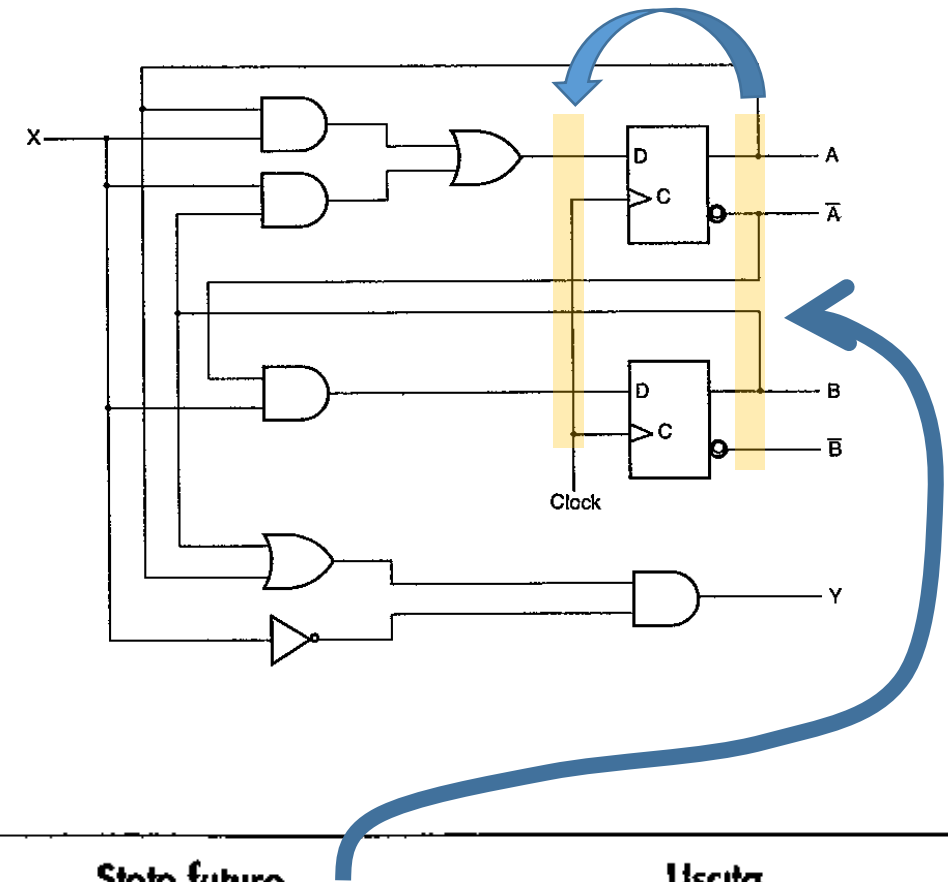


- La tabella può essere riformulata in modo da esplicitare la dipendenza dello stato futuro e le uscite dagli ingressi e dallo stato presente
- Gli indici di riga sono gli stati, quelli di colonna gli ingressi

Stato presente		Stato futuro				Uscita	
		X = 0		X = 1		X = 0	X = 1
A	B	A	B	A	B	Y	Y
0	0	0	0	0	1	0	0
0	1	0	0	1	1	1	0
1	0	0	0	1	0	1	0
1	1	0	0	1	0	1	0

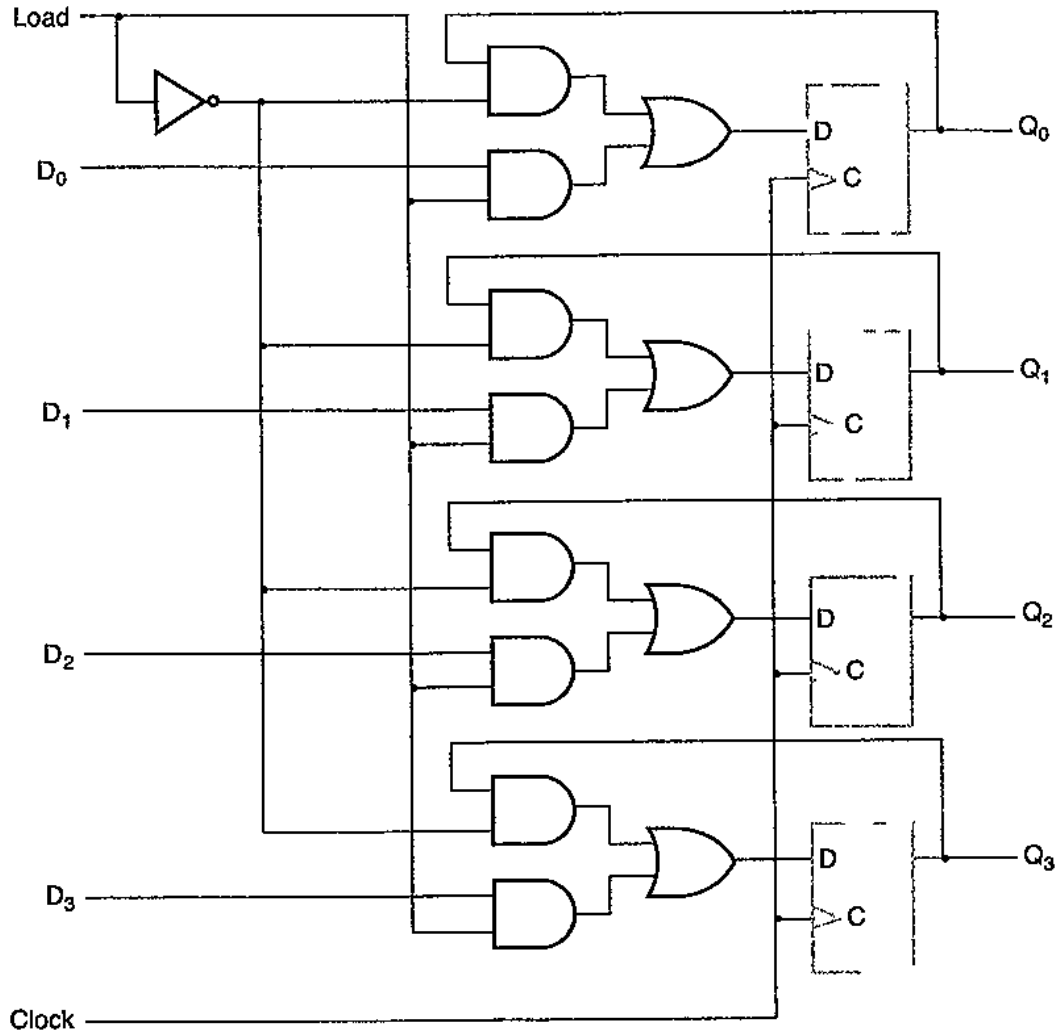
Sintesi di circuiti sequenziali

- Per effettuare la sintesi del circuito, occorre calcolare le funzioni di prossimo stato a partire dai valori presenti nella tabella delle transizioni
- Serve quindi calcolare i valori da porre in ingresso ai FF, in modo da ottenere la transizione/commutazione desiderata
- A tal scopo, si usano le tabelle di eccitazione dei FFs usati



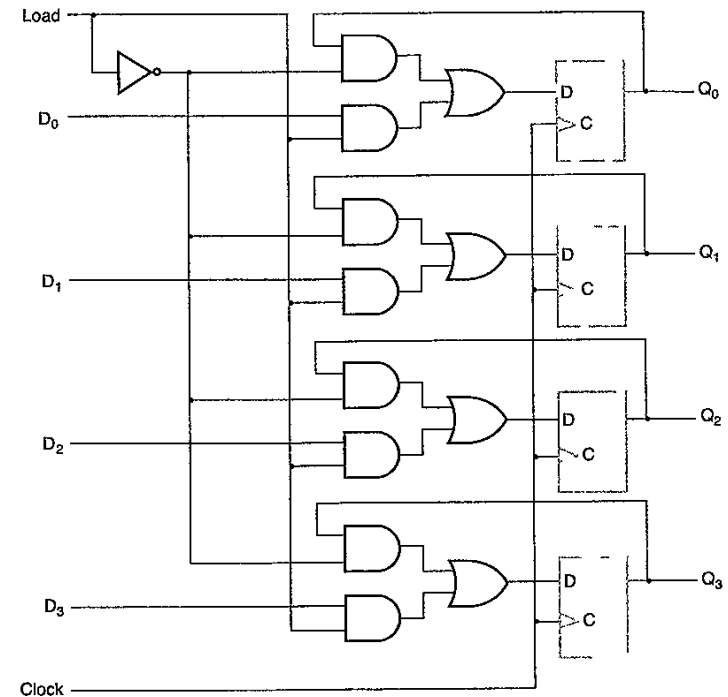
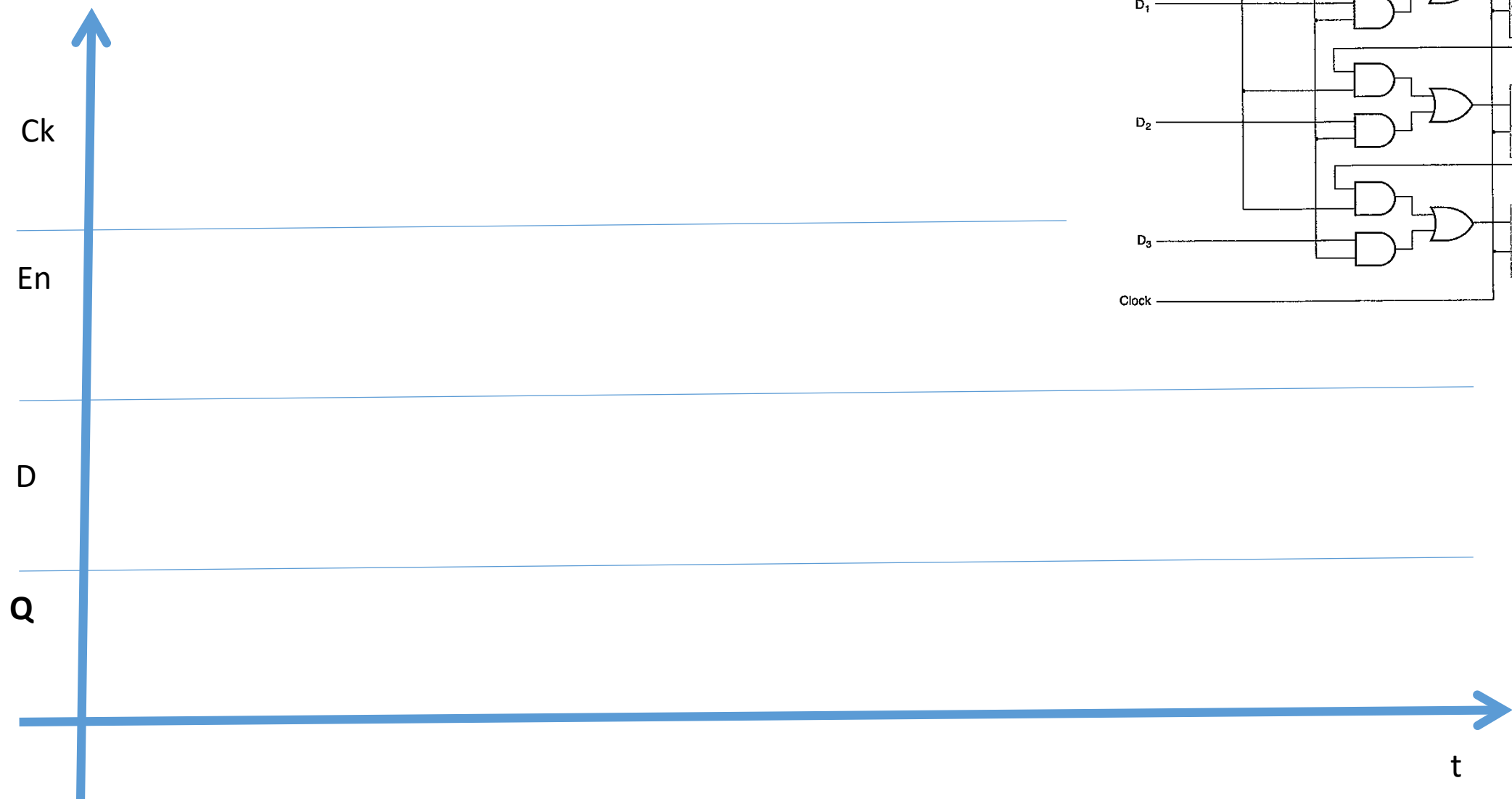
Stato presente	Stato futuro				Uscita	
	X = 0		X = 1		X = 0	X = 1
A B	A	B	A	B	Y	Y
0 0	0	0	0	1	0	0
0 1	0	0	1	1	1	0
1 0	0	0	1	0	1	0
1 1	0	0	1	0	1	0

Registri con Load



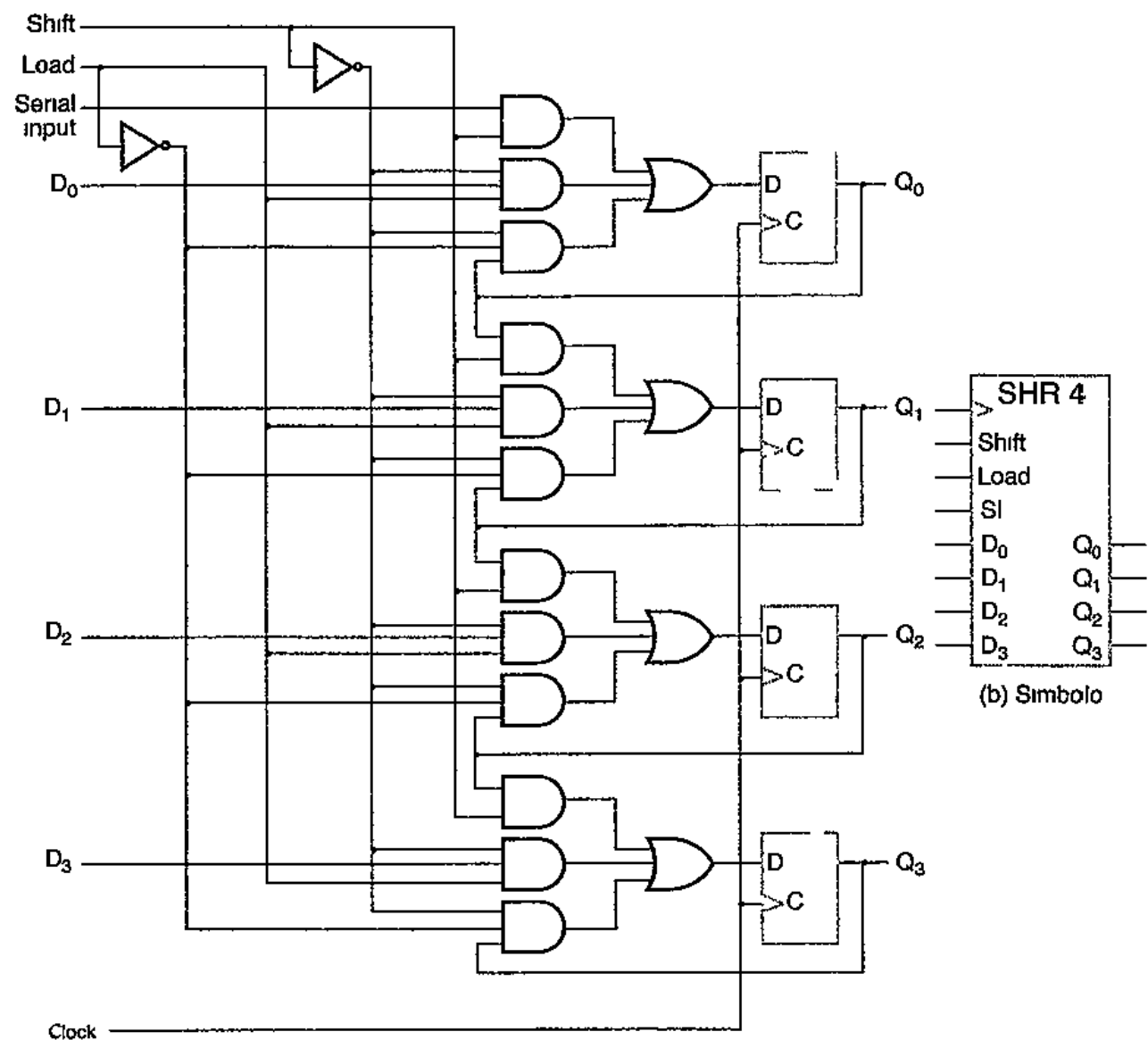
- Costituiscono l'elemento base di progettazione di qualunque sistema sincrono
- Il segnale di Load permette di caricare nel registro un nuovo pattern di valori di ingresso
- Il pattern è stabile fino al prossimo caricamento
- Sia il segnale di Load, sia i D_i devono rispettare le specifiche timing dei FFs

Timing Diagram



Shift Registers /1

- Utilizzati per molteplici scopi:
 - Generazione di ritardi
 - Conversione seriale/parallela e parallela/seriale
 - Moltiplicazione/divisione per potenze del 2
 - Selezione di particolari campi di bits
 -
- Tutti gli ingressi devono rispettare la disciplina sincrona

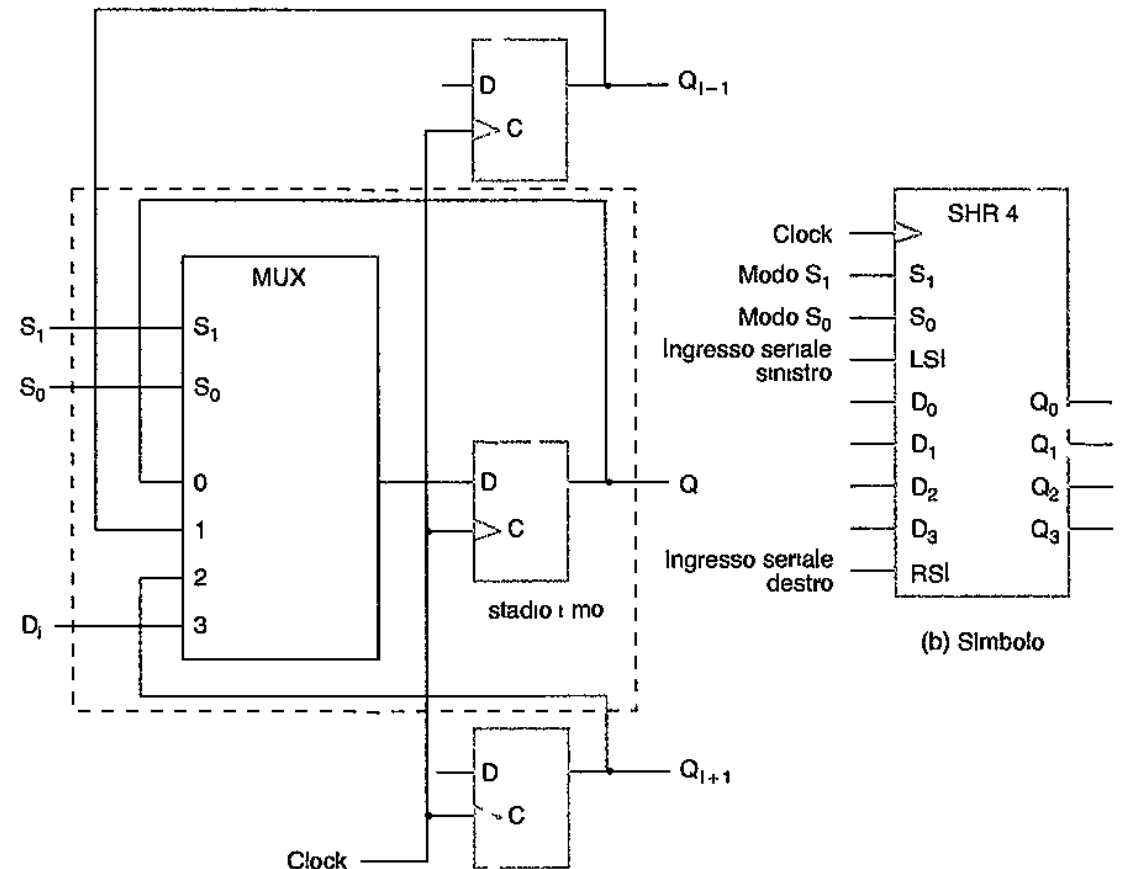


Shift	Load	Azione
0	0	Nessun cambiamento
0	1	Caricamento parallelo dei dati
1	x	Scorrimento da Q ₀ a Q ₃

Shift Registers /2

- Lo shift dei dati può diventare bidirezionale espandendo le funzionalità del multiplexer
- Il flusso dati può quindi essere parallelo o antiparallelo a quello del clock

Controllo modo		Azione
S_1	S_0	
0	0	Nessun cambiamento
0	1	Scorrimento in avanti
1	0	Scorrimento indietro
1	1	Caricamento parallelo dei dati



Da D a JK

Transition Table

	J-K Input		Outputs		D Input
	J	K	Qp	Qp+1	
idle	0	0	0	0	0
	0	0	1	1	1
reset	0	1	0	0	0
	0	1	1	0	0
set	1	0	0	1	1
	1	0	1	1	1
toggle	1	1	0	1	1
	1	1	1	0	0



ingressi

stato presente

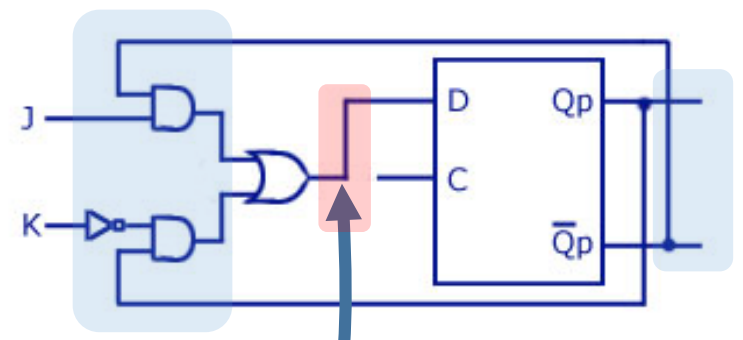
stato futuro

K-map

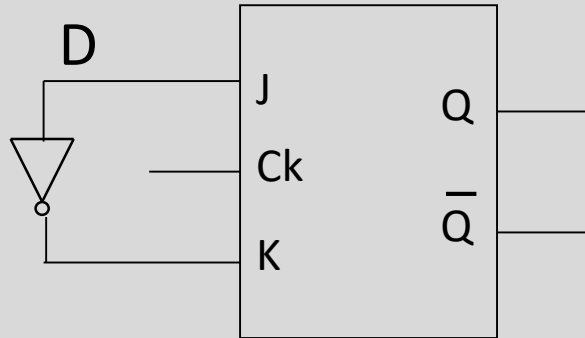
J \ KQp	00	01	11	10
1	0 ⁰	1 ¹	0 ³	0 ²
1	1 ⁴	1 ⁵	0 ⁷	1 ⁶

$D = J\bar{Q}_p + \bar{K}Q_p$

Logic Diagram



Flip-Flop tipo D e T

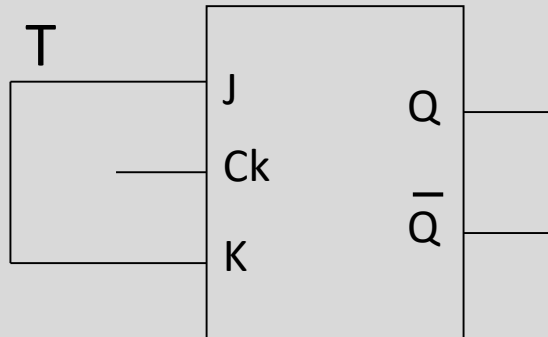


D_n	Q_{n+1}
1	1
0	0

Flip-Flop tipo **D**elay

J e K sono sempre uno l'opposto dell'altro

Il dato presente in ingresso
viene trasferito
in uscita quando il clock è alto



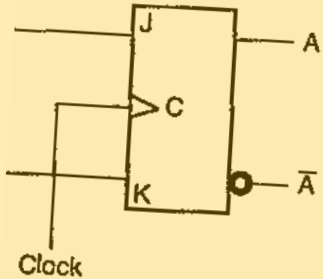
T_n	Q_{n+1}
1	$\bar{Q}_n$
0	Q_n

Flip-Flop tipo **T**oggle switch

J e K sono uguali

Se $T=0$ stato rimane costante
Se $T=1$ cambia stato ad ogni impulso
di clock

Tabelle di eccitazione dei FFs



(a) Flip flop JK

$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

(b) Flip flop SR

$Q(t)$	$Q(t+1)$	S	R
0	0	0	X
0	1	1	0
1	0	0	1
1	1	X	0

(c) Flip flop D

$Q(t)$	$Q(t+1)$	D
0	0	0
0	1	1
1	0	0
1	1	1



(d) Flip flop T

$Q(t)$	$Q(t+1)$	T
0	0	0
0	1	1
1	0	1
1	1	0

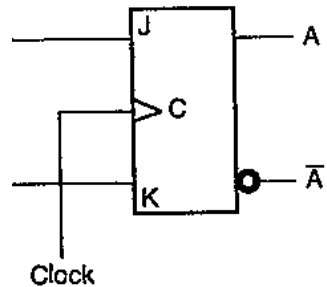
- Le tabelle di eccitazione dei FF permettono di calcolare gli ingressi da applicare per ottenere una determinata commutazione dell'uscita
- La tabella dei D-FF è immediata, quella degli altri FFs va determinata a partire dalla relativa tabella delle transizioni

Contatori

- Un contatore associa ad ogni periodo di clock un pattern binario, in base ad un codice predefinito (Gray, BCD, arbitrario, ...)
- Il contatore binario associa a ciascun periodo di clock un valore crescente o decrescente nell'intervallo $[0, 2^{N-1}]$
- La sequenza scelta ha, in generale, un impatto significativo sulla frequenza di funzionamento

Sequenza di conteggio crescente				Sequenza di conteggio decrescente			
Q ₃	Q ₂	Q ₁	Q ₀	Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0	1	1	1	1
0	0	0	1	1	1	1	0
0	0	1	0	1	1	0	1
0	0	1	1	1	1	0	0
0	1	0	0	1	0	1	1
0	1	0	1	1	0	1	0
0	1	1	0	1	0	0	1
0	1	1	1	1	0	0	0
1	0	0	0	0	1	1	1
1	0	0	1	0	1	1	0
1	0	1	0	0	1	0	1
1	0	1	1	0	1	0	0
1	1	0	0	0	0	1	1
1	1	0	1	0	0	1	0
1	1	1	0	0	0	0	1
1	1	1	1	0	0	0	0

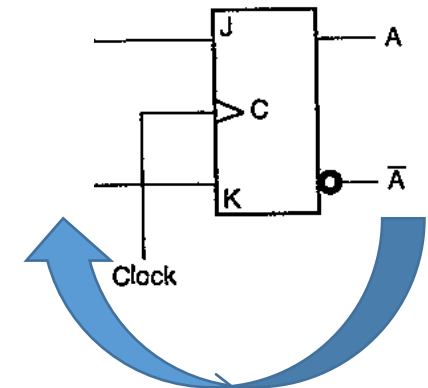
Sintesi di Contatori Binari: JK-FFs /1



Stato presente				Stato futura			
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0



(a) Flip flop JK			
$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0



- La regolarità della successione degli stati permette di affrontare agevolmente una sintesi del circuito
- Gli ingressi dei JK-FFs devono essere determinati attraverso la relativa tabella di eccitazione

Sintesi di Contatori Binari: JK-FFs /2

Q (t)	Q (t + 1)	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

[illegible]

Sintesi di Contatori Binari: JK-FFs /3

- Si possono a questo punto determinare le 8 funzioni combinatorie che pilotano gli ingressi dei 4 JK-FFs

Stato presente				Stato futura				Ingressi ai flip flop							
Q ₃	Q ₂	Q ₁	Q ₀	Q ₃	Q ₂	Q ₁	Q ₀	J _{Q3}	K _{Q3}	J _{Q2}	K _{Q2}	J _{Q1}	K _{Q1}	J _{Q0}	K _{Q0}
0	0	0	0	0	0	0	1	0	x	0	x	0	x	1	x
0	0	0	1	0	0	1	0	0	x	0	x	1	x	x	1
0	0	1	0	0	0	1	1	0	x	0	x	x	0	1	x
0	0	1	1	0	1	0	0	0	x	1	x	x	1	x	1
0	1	0	0	0	1	0	1	0	x	x	0	0	x	1	x
0	1	0	1	0	1	1	0	0	x	x	0	1	x	x	1
0	1	1	0	0	1	1	1	0	x	x	0	x	0	1	x
0	1	1	1	1	0	0	0	1	x	x	1	x	1	x	1
1	0	0	0	1	0	0	1	x	0	0	x	0	x	1	x
1	0	0	1	1	0	1	0	x	0	0	x	1	x	x	1
1	0	1	0	1	0	1	1	x	0	0	x	x	0	1	x
1	0	1	1	1	1	0	0	x	0	1	x	x	1	x	1
1	1	0	0	1	1	0	1	x	0	x	0	0	x	1	x
1	1	0	1	1	1	1	0	x	0	x	0	1	x	x	1
1	1	1	0	1	1	1	1	x	0	x	0	x	0	1	x
1	1	1	1	0	0	0	0	x	1	x	1	x	1	x	1

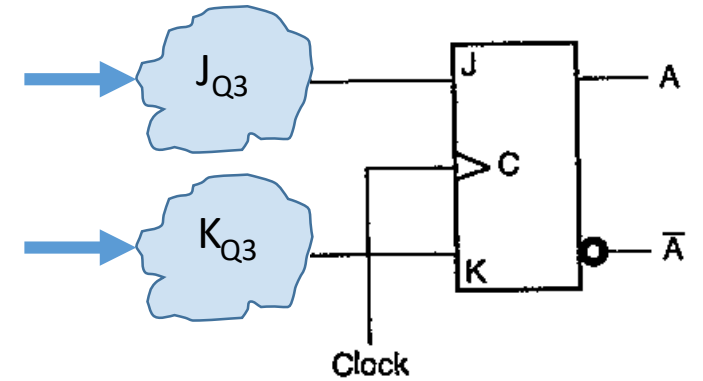
Sintesi di Contatori Binari: J/K₃

- $J_{Q3} = J_{Q3}(Q_3, Q_2, Q_1, Q_0)$



Q ₁ Q ₀	00	01	11	10
Q ₃ Q ₂				
00				
01				
11				
10				

Stato presente					
Q ₃	Q ₂	Q ₁	Q ₀	J _{Q3}	K _{Q3}
0	0	0	0	0	×
0	0	0	1	0	×
0	0	1	0	0	×
0	0	1	1	0	×
0	1	0	0	0	×
0	1	0	1	0	×
0	1	1	0	0	×
0	1	1	1	1	×
1	0	0	0	×	0
1	0	0	1	×	0
1	0	1	0	×	0
1	0	1	1	×	0
1	1	0	0	×	0
1	1	0	1	×	0
1	1	1	0	×	0
1	1	1	1	×	1



- $K_{Q3} = K_{Q3}(Q_3, Q_2, Q_1, Q_0)$



Q ₁ Q ₀	00	01	11	10
Q ₃ Q ₂				
00				
01				
11				
10				

Sintesi di Contatori Binari: J/K₂

- $J_{Q2} = J_{Q2}(Q_3, Q_2, Q_1, Q_0)$



Q ₁ Q ₀	00	01	11	10
Q ₃ Q ₂				
00				
01				
11				
10				

Stato presente					
Q ₃	Q ₂	Q ₁	Q ₀	J _{Q2}	K _{Q2}
0	0	0	0	0	×
0	0	0	1	0	×
0	0	1	0	0	×
0	0	1	1	1	×
0	1	0	0	×	0
0	1	0	1	×	0
0	1	1	0	×	0
0	1	1	1	×	1
1	0	0	0	0	×
1	0	0	1	0	×
1	0	1	0	0	×
1	0	1	1	1	×
1	1	0	0	×	0
1	1	0	1	×	0
1	1	1	0	×	0
1	1	1	1	×	1

- $K_{Q2} = K_{Q2}(Q_3, Q_2, Q_1, Q_0)$



Q ₁ Q ₀	00	01	11	10
Q ₃ Q ₂				
00				
01				
11				
10				

Sintesi di Contatori Binari: J/K₁

- $J_{Q1} = J_{Q1}(Q_3, Q_2, Q_1, Q_0)$



Q1 Q0	00	01	11	10
Q3 Q2				
00				
01				
11				
10				

Stato presente					
Q ₃	Q ₂	Q ₁	Q ₀	J _{Q1}	K _{Q1}
0	0	0	0	0	×
0	0	0	1	1	×
0	0	1	0	×	0
0	0	1	1	×	1
0	1	0	0	0	×
0	1	0	1	1	×
0	1	1	0	×	0
0	1	1	1	×	1
1	0	0	0	0	×
1	0	0	1	1	×
1	0	1	0	×	0
1	0	1	1	×	1
1	1	0	0	0	×
1	1	0	1	1	×
1	1	1	0	×	0
1	1	1	1	×	1

- $K_{Q1} = K_{Q1}(Q_3, Q_2, Q_1, Q_0)$



Q1 Q0	00	01	11	10
Q3 Q2				
00				
01				
11				
10				

Sintesi di Contatori Binari: J/K₀

- $J_{Q0} = J_{Q0}(Q_3, Q_2, Q_1, Q_0)$



Q1 Q0	00	01	11	10
Q3 Q2				
00				
01				
11				
10				

Stato presente			
Q ₃	Q ₂	Q ₁	Q ₀
0	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
0	1	1	1
1	0	0	0
1	0	0	1
1	0	1	0
1	0	1	1
1	1	0	0
1	1	0	1
1	1	1	0
1	1	1	1

J _{Q0}	K _{Q0}
1	×
×	1
1	×
×	1
1	×
×	1
1	×
×	1
1	×
×	1
1	×
×	1
1	×
×	1
1	×
×	1

- $K_{Q0} = K_{Q0}(Q_3, Q_2, Q_1, Q_0)$



Q1 Q0	00	01	11	10
Q3 Q2				
00				
01				
11				
10				

Mappe

$Q_3Q_2 \backslash Q_1Q_0$		Q_1			
		00	01	11	10
Q_3	00				
	01			1	
	11	X	X	X	X
	10	X	X	X	X
		Q_0			
		Q_2			

$$J_{Q3} = Q_0 Q_1 Q_2$$

X	X	X	X
X	X	X	X
		1	

$$K_{Q3} = Q_0 Q_1 Q_2$$

	1	X	X
	1	X	X
	1	X	X
	1	X	X

$$J_{Q1} = Q_0$$

X	X	1	
X	X	1	
X	X	1	
X	X	1	

$$K_{Q1} = Q_0$$

		1	
X	X	X	X
X	X	X	X
		1	

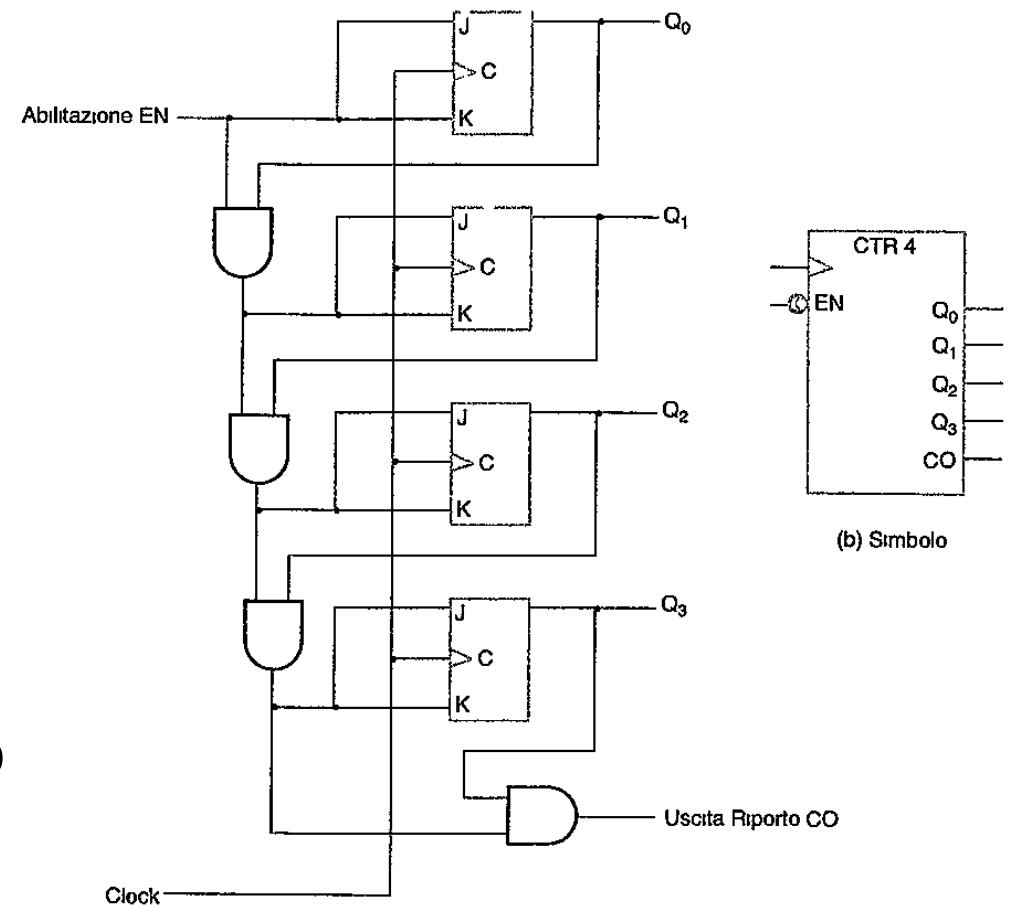
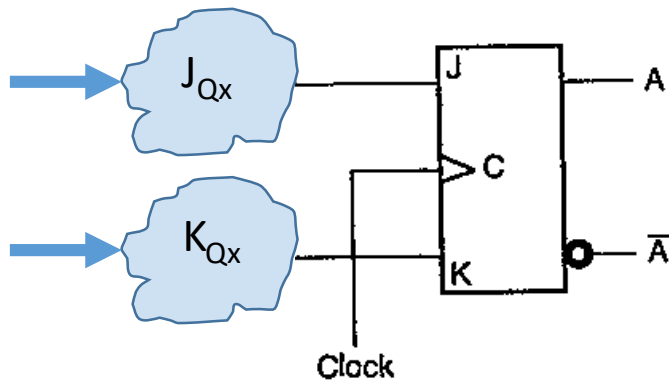
$$J_{Q2} = Q_0 Q_1$$

X	X	X	X
		1	
		1	
X	X	X	X

$$K_{Q2} = Q_0 Q_1$$

Circuito e funzioni di prossimo stato

- Per ogni JK-FF risulta $J_{Qx} = K_{Qx}$
- E' possibile fermare il conteggio aggiungendo un prodotto in ogni equazione
- Se $En=0$, allora $J_{Qx} = K_{Qx} = 0$ e il FF rimane nello stato precedente



$$\begin{aligned}
 J_{Q0} &= K_{Q0} = EN \\
 J_{Q1} &= K_{Q1} = Q_0 \quad EN \\
 J_{Q2} &= K_{Q2} = Q_0 \quad Q_1 \quad EN \\
 J_{Q3} &= K_{Q3} = Q_0 \quad Q_1 \quad Q_2 \quad EN
 \end{aligned}$$

Simmetrie della tabella di transizione

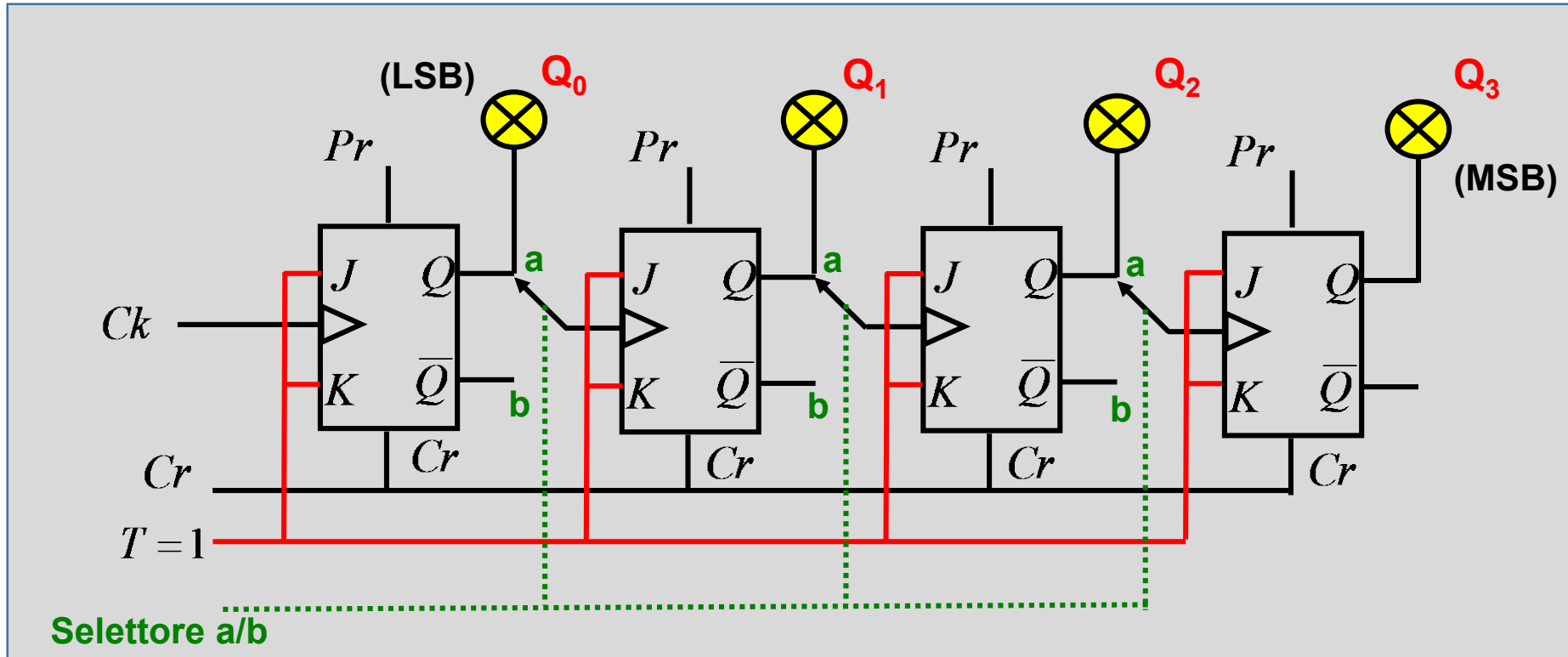
Stato presente				Stato futura			
Q ₃	Q ₂	Q ₁	Q ₀	Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

Stato presente				Stato futura			
Q ₃	Q ₂	Q ₁	Q ₀	Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

Stato presente				Stato futura			
Q ₃	Q ₂	Q ₁	Q ₀	Q ₃	Q ₂	Q ₁	Q ₀
0	0	0	0	0	0	0	1
0	0	0	1	0	0	1	0
0	0	1	0	0	0	1	1
0	0	1	1	0	1	0	0
0	1	0	0	0	1	0	1
0	1	0	1	0	1	1	0
0	1	1	0	0	1	1	1
0	1	1	1	1	0	0	0
1	0	0	0	1	0	0	1
1	0	0	1	1	0	1	0
1	0	1	0	1	0	1	1
1	0	1	1	1	1	0	0
1	1	0	0	1	1	0	1
1	1	0	1	1	1	1	0
1	1	1	0	1	1	1	1
1	1	1	1	0	0	0	0

Contatore Asincrono modulo 16

- Un Flip-Flop di tipo T è un divisore di frequenza per 2
→ possiamo realizzare un contatore con Flip-Flop T in cascata

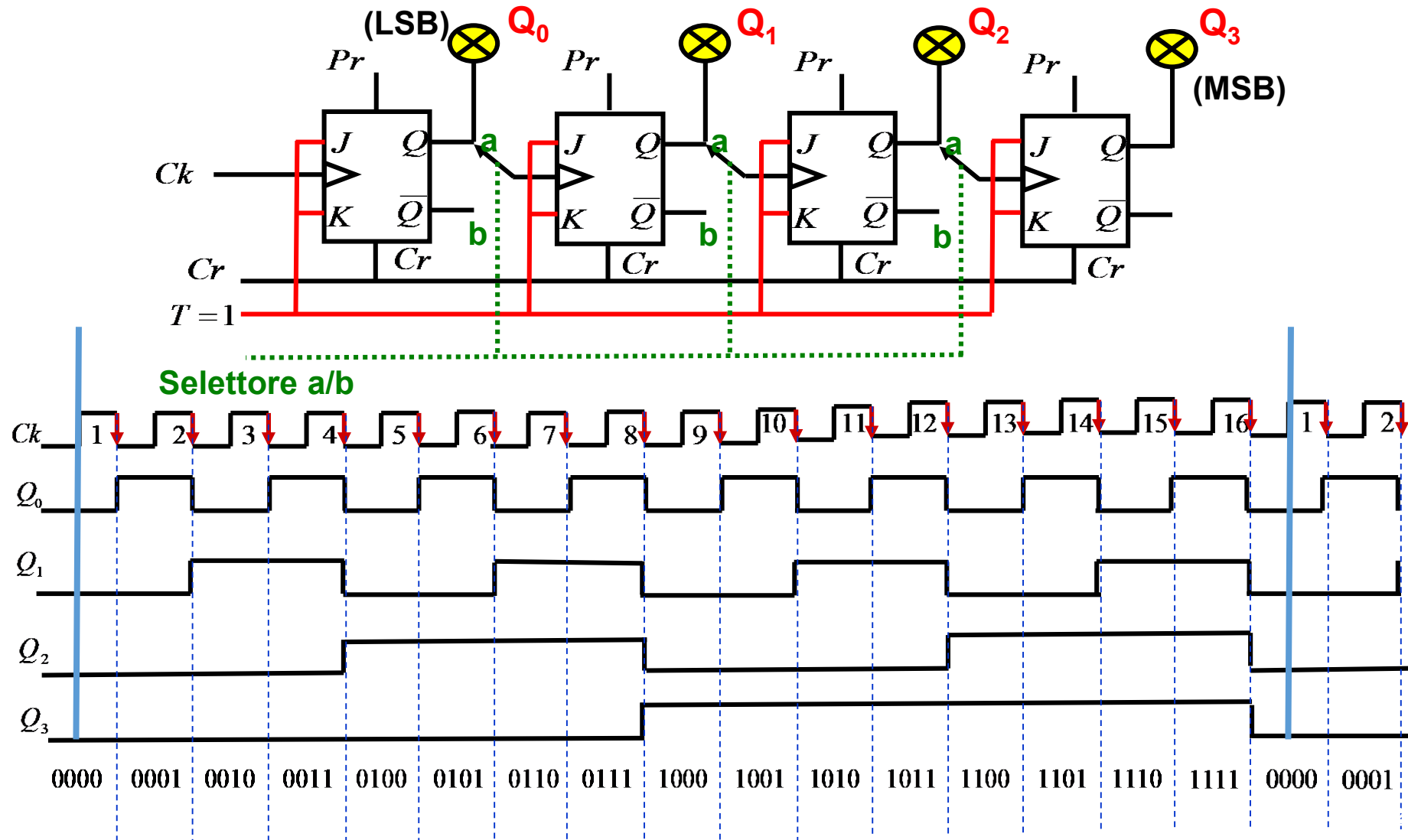


Il master cambia stato quando il clock passa da 0 a 1

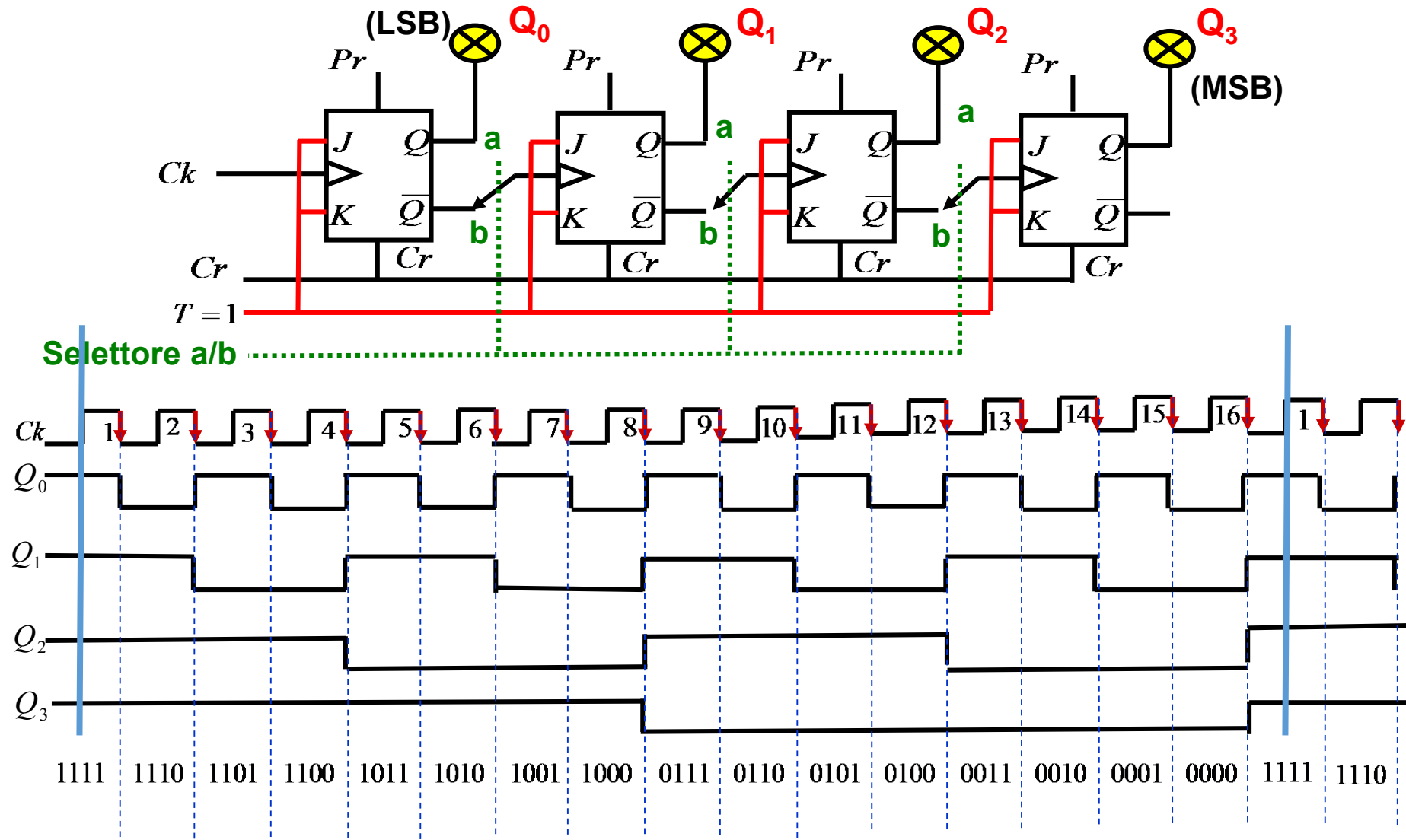
Lo stato del master è trasferito allo slave quando il clock passa da 1 a 0

Q cambia stato sul fronte di discesa del clock

Contatore Asincrono modulo 16

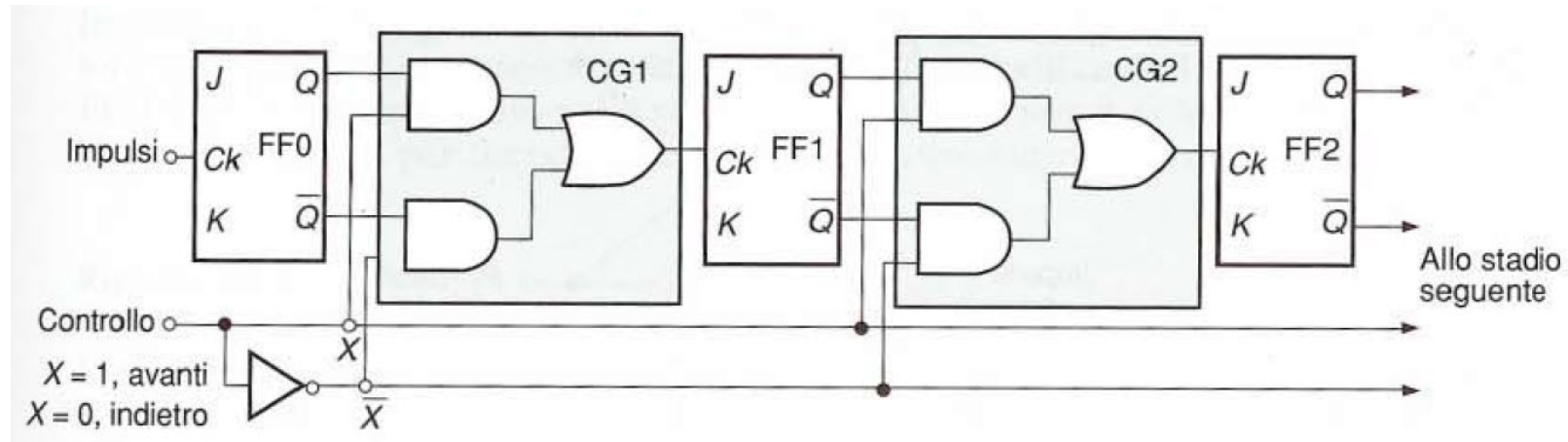


Contatore Asincrono modulo 16 all'indietro



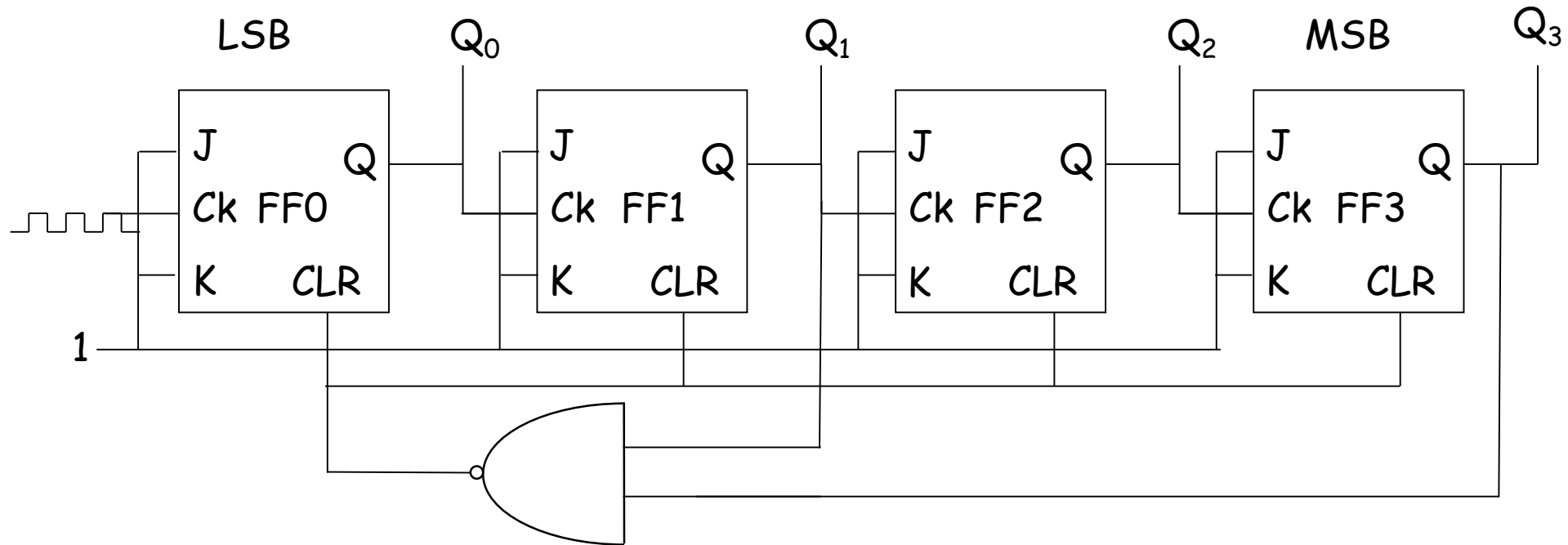
Contatore asincrono avanti-indietro

Un Multiplexer tra ogni coppia di Flip-Flop permette di contare in avanti o all'indietro a seconda del valore della variabile X



Contatore asincrono modulo 10

Normalmente si riazzera a 2^N . Allora:



- ✓ Abilito AND quando output è uguale a 10 (1010 binario)
- ✓ AND manda il CLR ai Flip-Flop
- ✓ Il contatore si riazzera dopo il 10° impulso

Contatori Asincroni: Limitazioni

- La massima frequenza di conteggio è

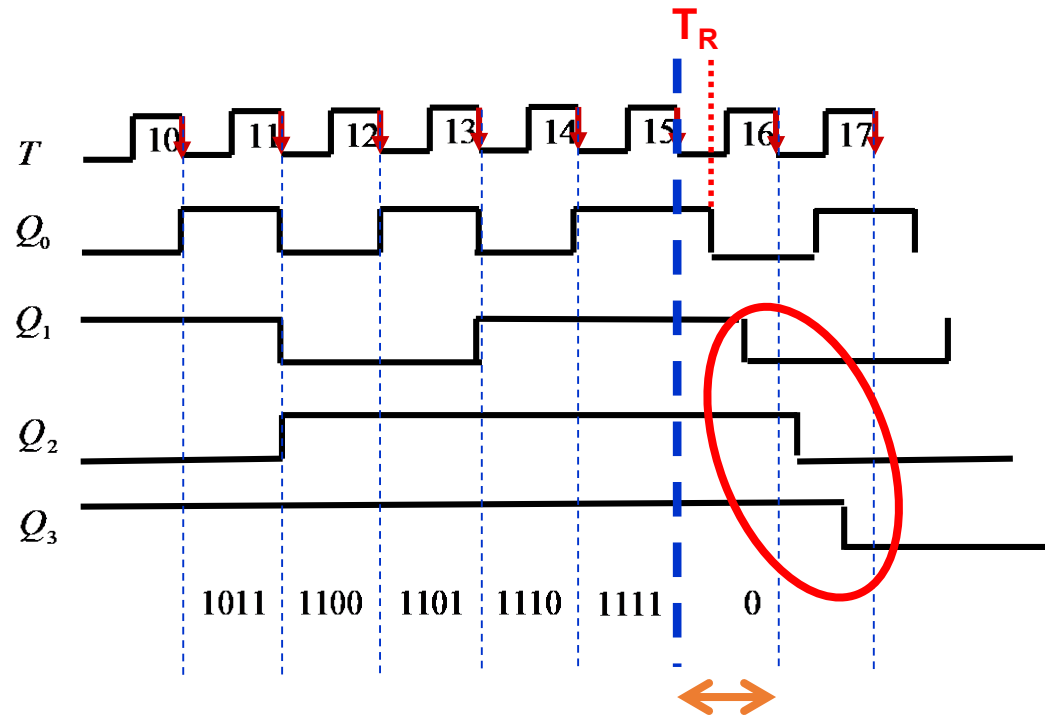
$$f_{\max} = 1/T_{\min} = 1/nT_p$$

Con T_p tempo di propagazione del segnale in un Flip-Flop.

- La condizione peggiore è che debba commutare l'ultimo Flip-Flop (quando tutti gli altri sono a 1).
- I bit hanno un settling time crescente, occorre attendere quello più elevato prima di leggere il risultato corrente

Contatori Asincroni: Limitazioni

- Il tempo di propagazione del Ck (**propagazione del riporto della somma**) tra Flip-Flop successivi può essere un problema, specialmente quando il numero di Flip-Flop è grande.



Se $T_{R_{tot}} > T_{Ck}$ non è possibile leggere il contenuto del contatore