Corso di Calcolatori Elettronici I
A.A. 2012-2013

Reti sequenziali notevoli: registri, registri a scorrimento, contatori

ing. Alessandro Cilardo

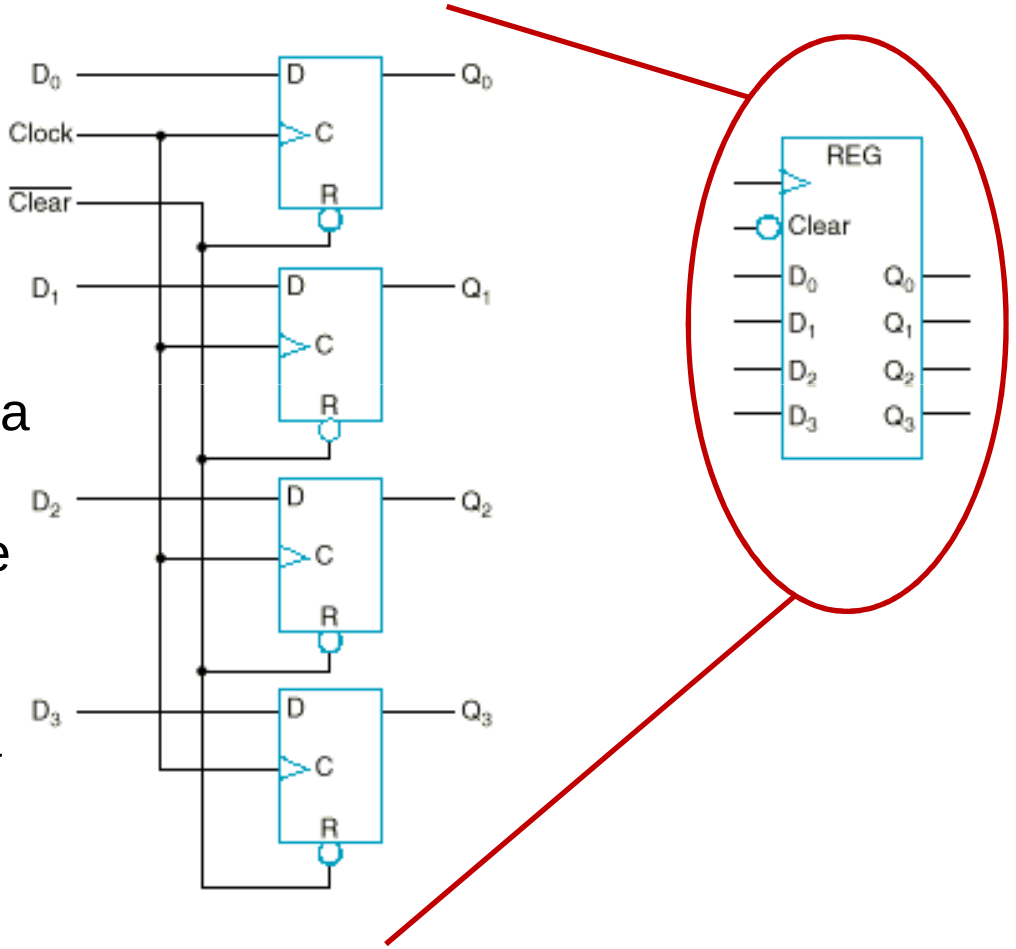
Accademia Aeronautica di Pozzuoli
Corso Pegaso V “GArn Elettronici”

Registri

- Memorizzano *n* differenti bit
- Possono essere realizzati come macchine sincrone o asincrone
- 2^n stati diversi
  - → non vengono progettati con il metodo generale
  - → si ricorre a schemi circuitali noti

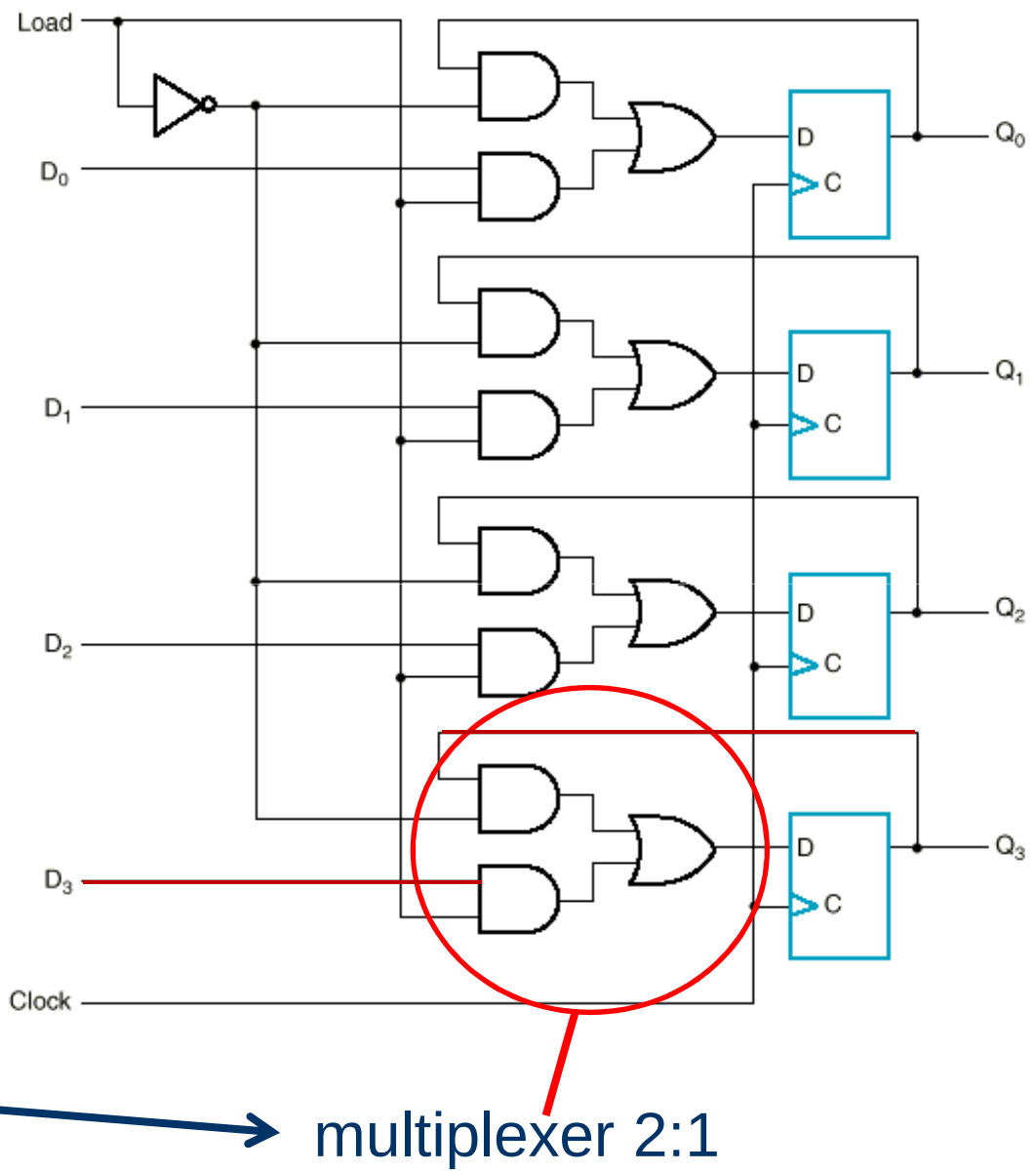
Registri

- Realizzati con *n* flip-flop
- Esempio in figura:
  - registro a sincronizzazione esterna
  - *n* = 4 bit
  - edge-triggered su fronte di salita
  - ingresso di clear (**R**) attivo basso: se si porta **R** a **0**, il registro viene resettato e conterrà quindi tutti bit pari **0**



Registri

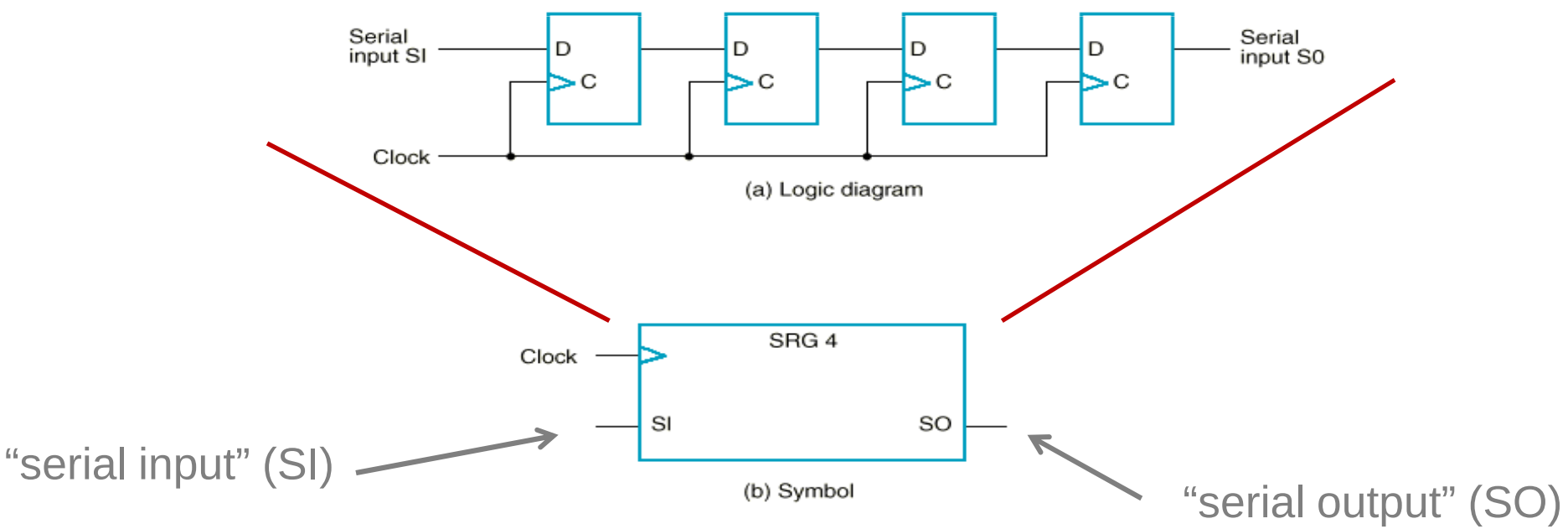
- Registro a caricamento parallelo
  - **Load**=**1** trasferisce i 4 bit di ingresso nei flip-flop, *in parallelo*
  - **Load**=**0** lascia il contenuto del registro inalterato



Sulla base dell'ingresso di selezione **Load**, il multiplexer 2:1 posto in ingresso a ciascun flip-flop riporta in ingresso il bit memorizzato $\mathbf{Q_i}$ (**Load**=**0**), o il bit letto dall'ingresso esterno $\mathbf{D_i}$ (**Load**=**1**)

Registri a scorrimento (*shift registers*)

- Formati da una catena di flip-flop connessi *in cascata*
- Ad ogni colpo di clock, ogni flip-flop passa il proprio contenuto al flip-flop successivo



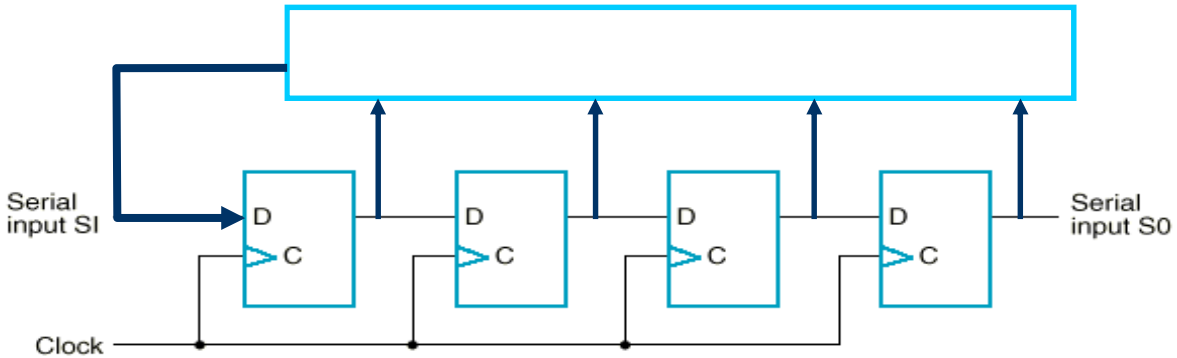
Registri a scorrimento con retroazione

Materiale facoltativo

- Una funzione dello stato è usata in ingresso

  → retroazione (*feedback*)

$$I = f(A_{n-1}, A_{n-2}, \ldots A_0)$$



Registri a scorrimento con retroazione

Materiale facoltativo

- In particolare, se $I = A_{n-1}$ (registro a scorrimento circolare) si possono realizzare contatori modulari, inizializzando opportunamente il registro
- Es.: se $n=4$, si possono ottenere contatori modulo 4 (inizializzandolo con 1000 o 1100) e modulo 2 (usando 0101)

1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1
1	0	0	0

1	1	0	0
0	1	1	0
0	0	1	1
1	0	0	1
1	1	0	0

0	1	0	1
1	0	1	0
0	1	0	1

Registri a scorrimento con retroazione

Materiale facoltativo

- Altro esempio:
  - $I = \text{not}(A_{n-1})$

0	0	0	0
1	0	0	0
1	1	0	0
1	1	1	0
1	1	1	1
0	1	1	1
0	0	1	1
0	0	0	1
0	0	0	0

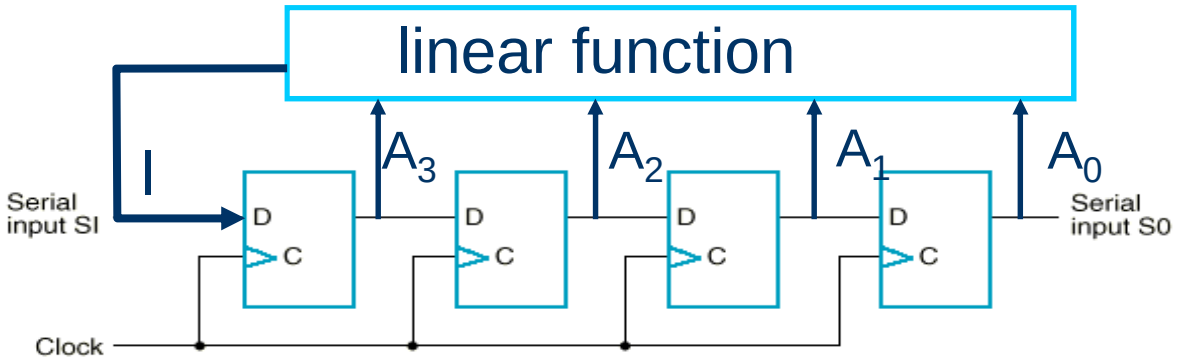
Contatore modulo 8

Registri a scorrimento lineari

Materiale facoltativo

- Normalmente chiamati *Linear Feedback Shift Register* (LFSR)
- Una combinazione lineare dello stato è posta in ingresso

$$I = f(A_{n-1}, A_{n-2}, \ldots A_0) =$$
$$= k_{n-1}A_{n-1} \textbf{ xor } k_{n-2}A_{n-2} \textbf{ xor } \ldots \textbf{ xor } \quad k_0A_0$$



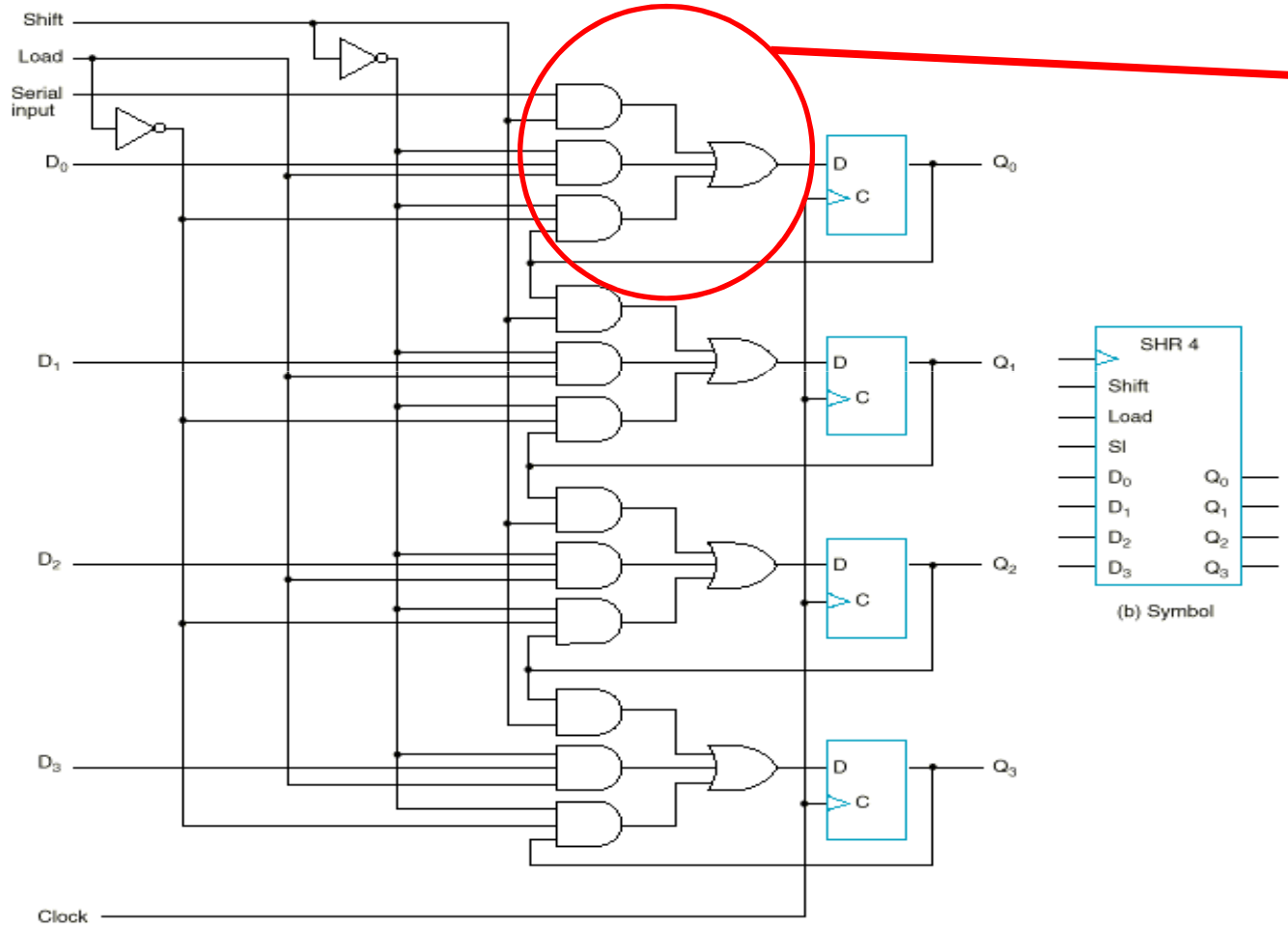
Registri a scorrimento lineari

Materiale facoltativo

- In generale, è possibile scegliere ***f*** in maniera tale da ottenere sequenze di valori che soddisfino certe proprietà
  - → sequenze pseudo-casuali
  - → sequenze a ciclo massimo (lunghe 2^n-1)

    ottenute le sa funzione *f* è un polinomio irriducibile su GF(2)
- I LFSR possono vedersi come sostituti dei contatori (più efficienti in termini realizzativi)

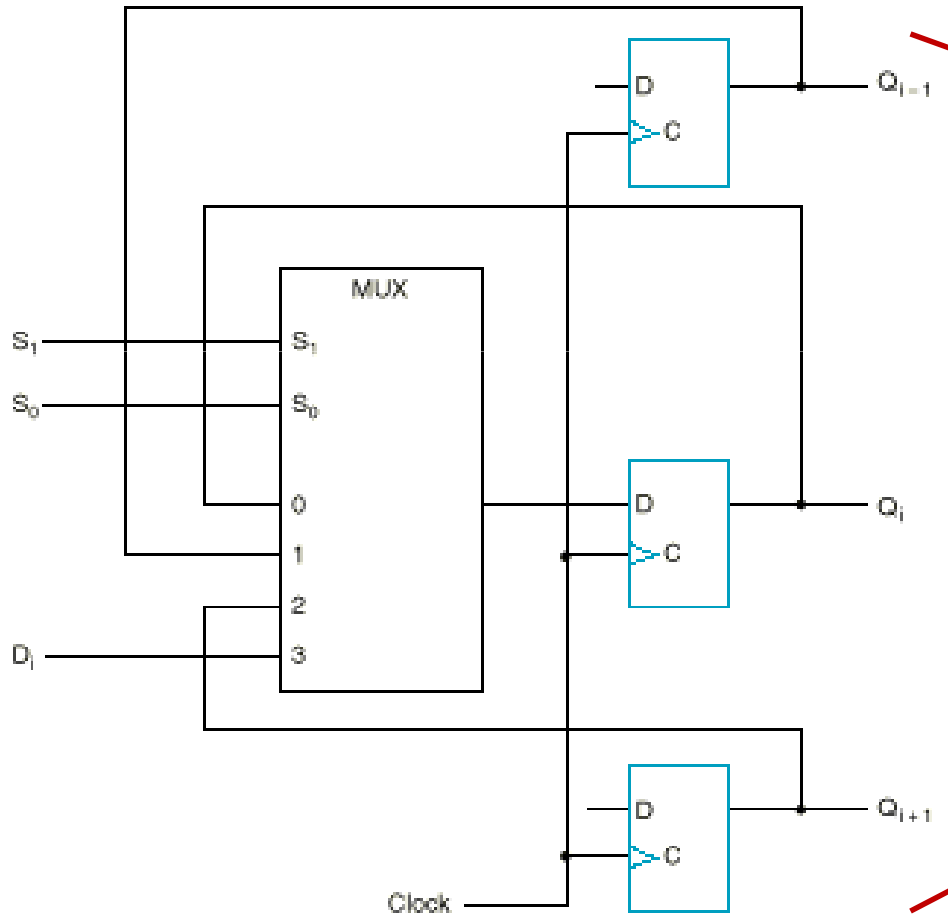
Registro a scorrimento con caricamento parallelo

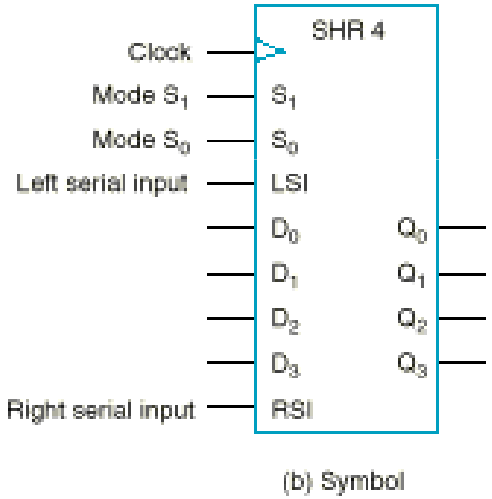


multiplexer 3:1

il multiplexer 3:1 consente di caricare in ciascun flip-flop il bit nella stessa posizione $\mathbf{Q_i}$, quello nella posizione precedente $\mathbf{Q_{i-1}}$, il bit proveniente dall'esterno $\mathbf{D_i}$

Registro a scorrimento bidirezionale





(b) Symbol

il multiplexer (MUX) consente di caricare in ciascun flip-flop il bit nella stessa posizione $\mathbf{Q_i}$, quello nella posizione precedente $\mathbf{Q_{i-1}}$, o quello nella posizione successiva $\mathbf{Q_{i+1}}$ a seconda del valore dei segnali di selezione $\mathbf{s_1}$, $\mathbf{s_0}$

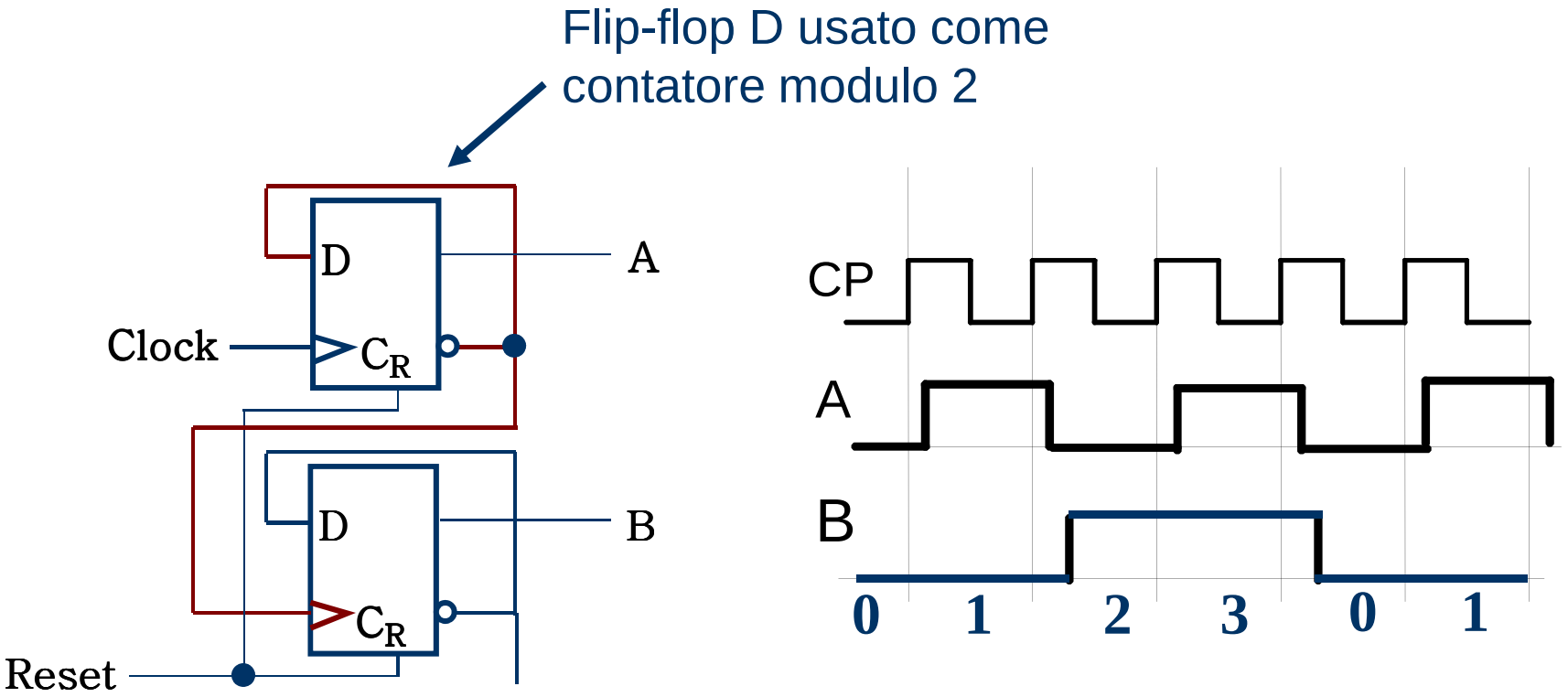
Contatori

- Posseggono *n* stati ordinati
- Un unico (*count*) ingresso fa progredire la rete attraverso la sequenza di stati ordinati
- un’opportuna codifica delle uscite rende questa macchina un contatore
- Se il primo stato coincide con lo stato successivo dell’ultimo stato, si realizza un contatore modulo *n*

Contatori

- Contatori a cascata (ripple-counter)
  - l'uscita di un elemento di memoria attiva l'elemento successivo
  - essenzialmente, sono realizzati connettendo più contatori modulo 2
- Contatori sincroni
  - il segnale di attivazione è comune agli elementi con memoria
  - una rete combinatoria "calcola" il prossimo valore memorizzato
- Contatori asincroni
  - attivati da transizioni sull'ingresso di conteggio
  - esempio: il flip-flop T è un contatore asincrono modulo 2

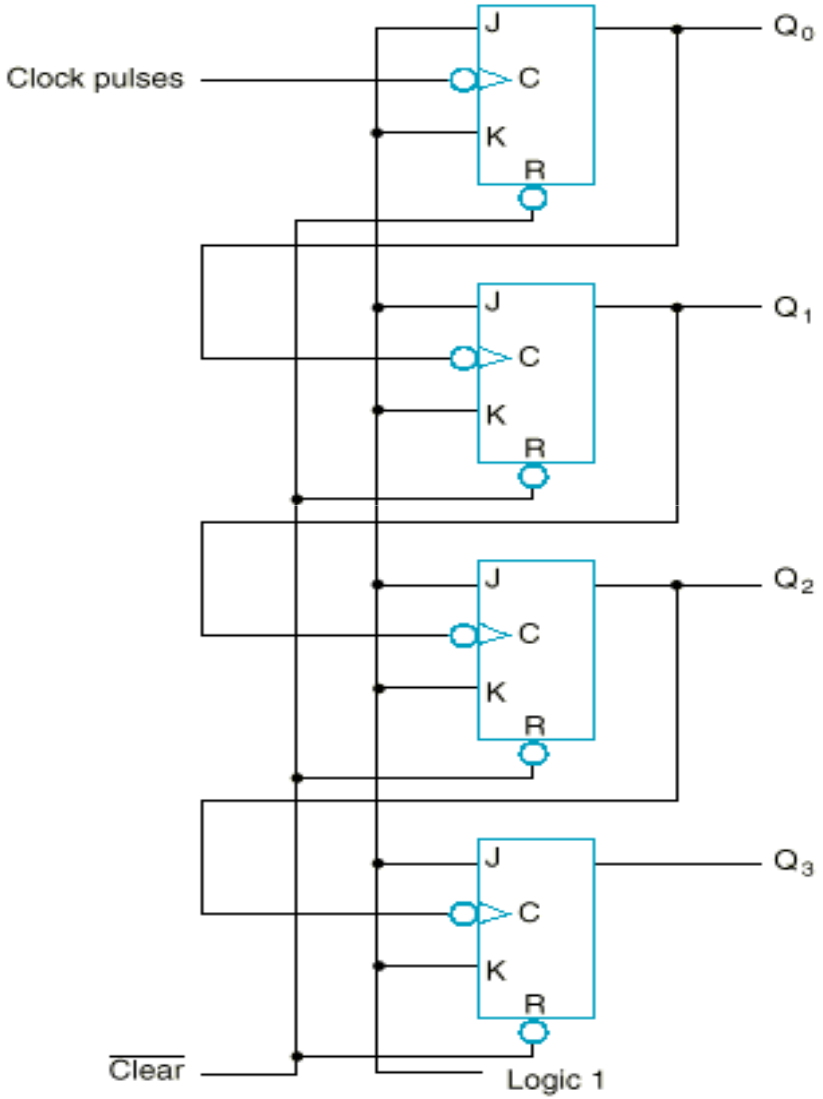
Contatori a Cascata



un contatore ad ***n*** bit può essere composto connettendo come sopra ***n*** flip-flop

Contatori a Cascata

- contatore a crescere con flip-flop JK (in modalità *Toggle*)



Contatori a Cascata

- sono di semplice realizzazione
- vengono attivate solo le parti del circuito interessate alle transizioni

  → adatti a circuiti che richiedono basso consumo di potenza

Ma…

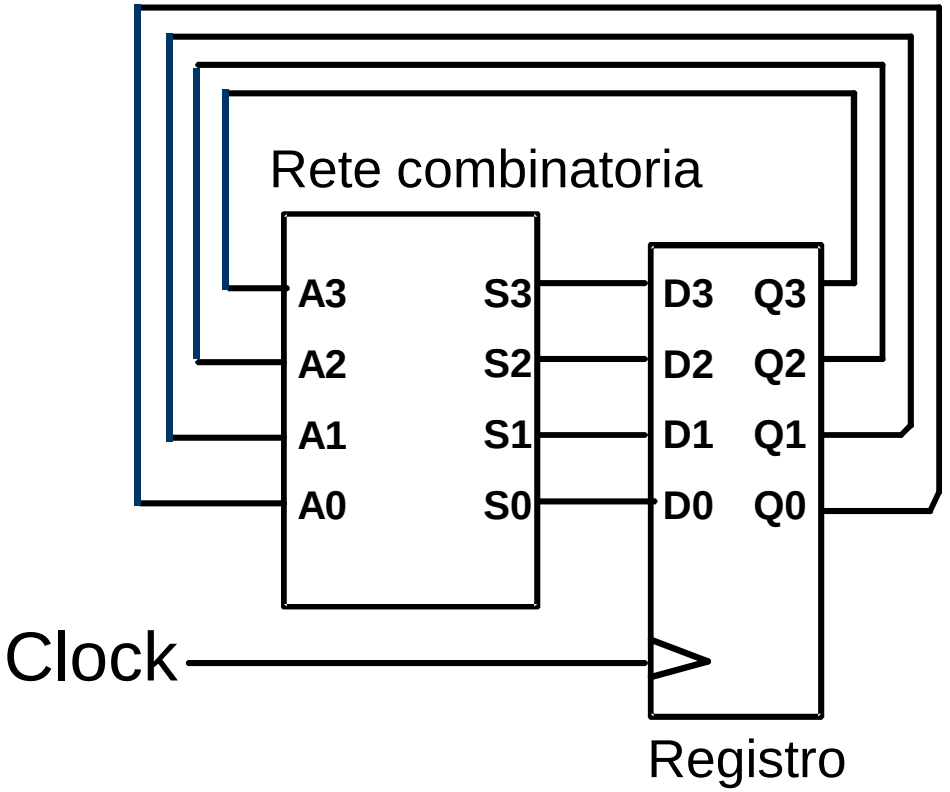
- particolarmente lenti per un alto numero di bit
- più critici da dimensionare a causa della dipendenza dei ritardi e del *clock skew*

Contatori sincroni

- segnale di sincronizzazione comune a tutti i flip-flop
  - → elemina l'effetto "ripple" (propagazione lenta del riporto)
  - → prestazioni migliori
- La rete combinatoria che "aggiorna" lo stato è progettata come per una qualsiasi macchina sequenziale sincrona
  - può essere realizzata con un adder
- Possibili ingressi ausiliari:
  - increment / decrement
  - reset
  - load

Contatori sincroni

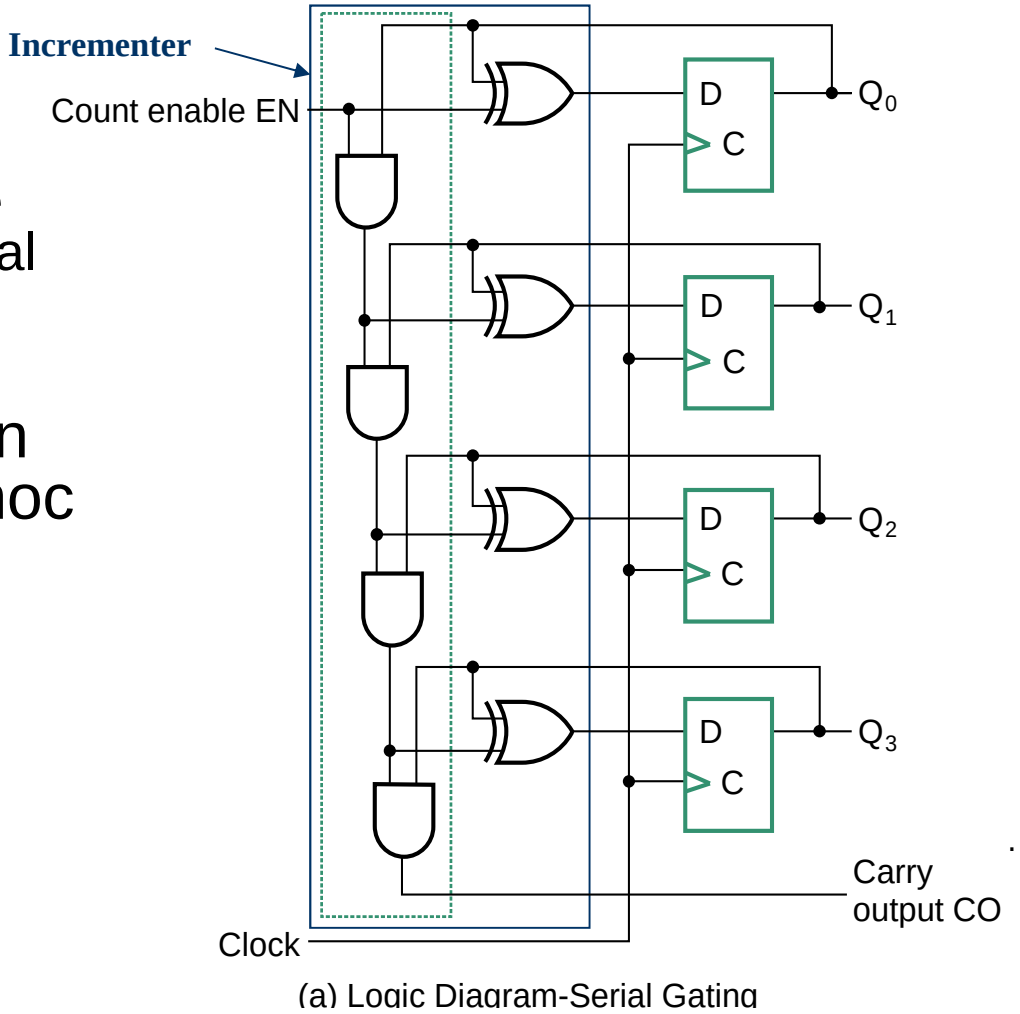
- Nel modello generale, una rete combinatoria definisce la dipendenza dello stato prossimo dallo stato corrente
- Determina la sequenza di stati percorsi durante il conteggio



Contatori sincroni

Materiale facoltativo

- “Catene di carry”:
  - serie di AND attraverso le quali scorre il riporto (serial gating)
- E’ possibile sostituire la rete di propagazione con una rete progettata ad hoc (parallel gating)
  - ad es. carry lookahead

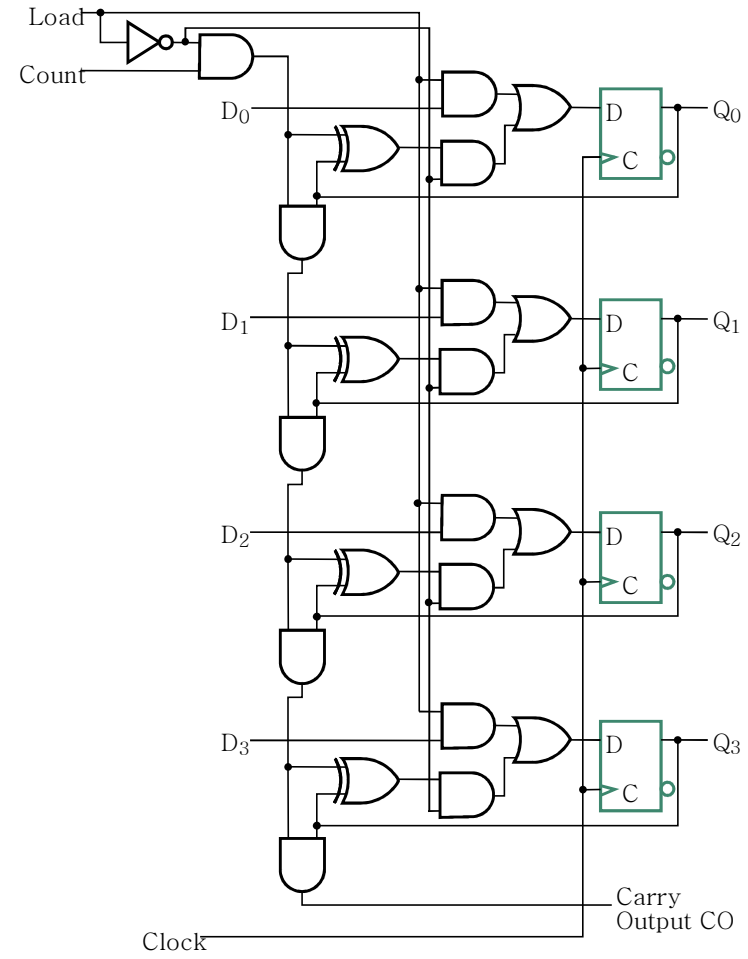


(a) Logic Diagram-Serial Gating

Contatori sincroni

Materiale facoltativo

- Contatore con caricamento parallelo
  - I multiplexer in ingresso ai flip-flop determinano se caricare in parallelo, o far evolvere il conteggio
- Esempio di *clock-gating*



Contatori sincroni

- Esempio di tabella degli stati per un contatore sincrono a 4 bit (16 stati)
- Non sono presenti ingressi:

  → nell'esempio, le transizioni sono implicitamente attivate dal clock

Present state				Next state			
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

Contatori sincroni: esempio

Present state				Next state				Flip-flop inputs							
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	×	0	×	0	×	1	×
0	0	0	1	0	0	1	0	0	×	0	×	1	×	×	1
0	0	1	0	0	0	1	1	0	×	0	×	×	0	1	×
0	0	1	1	0	1	0	0	0	×	1	×	×	1	×	1
0	1	0	0	0	1	0	1	0	×	×	0	0	×	1	×
0	1	0	1	0	1	1	0	0	×	×	0	1	×	×	1
0	1	1	0	0	1	1	1	0	×	×	0	×	0	1	×
0	1	1	1	1	0	0	0	1	×	×	1	×	1	×	1
1	0	0	0	1	0	0	1	×	0	0	×	0	×	1	×
1	0	0	1	1	0	1	0	×	0	0	×	1	×	×	1
1	0	1	0	1	0	1	1	×	0	0	×	×	0	1	×
1	0	1	1	1	1	0	0	×	0	1	×	×	1	×	1
1	1	0	0	1	1	0	1	×	0	×	0	0	×	1	×
1	1	0	1	1	1	1	0	×	0	×	0	1	×	×	1
1	1	1	0	1	1	1	1	×	0	×	0	×	0	1	×
1	1	1	1	0	0	0	0	×	1	×	1	×	1	×	1

usiamo flip-flop JK:

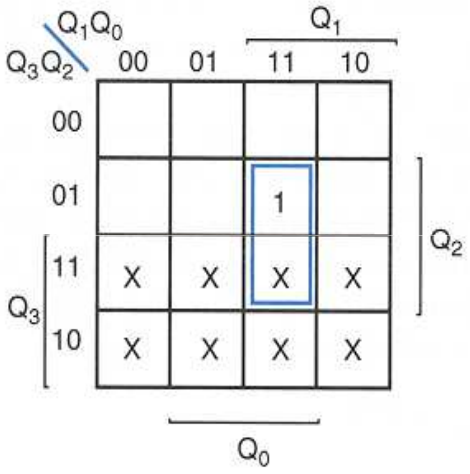
Flip-Flop Excitation Tables

(a) *JK* Flip-Flop

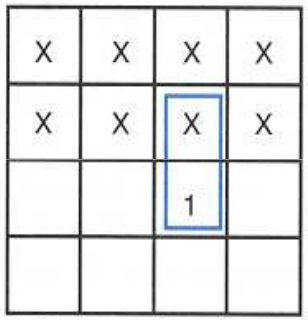
$Q(t)$	$Q(t+1)$	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

Contatori sincroni: esempio

Present state				Next state					
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	J_{Q3}	K_{Q3}
0	0	0	0	0	0	0	1	0	X
0	0	0	1	0	0	1	0	0	X
0	0	1	0	0	0	1	1	0	X
0	0	1	1	0	1	0	0	0	X
0	1	0	0	0	1	0	1	0	X
0	1	0	1	0	1	1	0	0	X
0	1	1	0	0	1	1	1	0	X
0	1	1	1	1	0	0	0	1	X
1	0	0	0	1	0	0	1	X	0
1	0	0	1	1	0	1	0	X	0
1	0	1	0	1	0	1	1	X	0
1	0	1	1	1	1	0	0	X	0
1	1	0	0	1	1	0	1	X	0
1	1	0	1	1	1	1	0	X	0
1	1	1	0	1	1	1	1	X	0
1	1	1	1	0	0	0	0	X	1



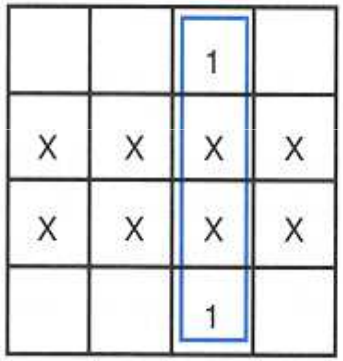
$J_{Q3} = Q_0 Q_1 Q_2$



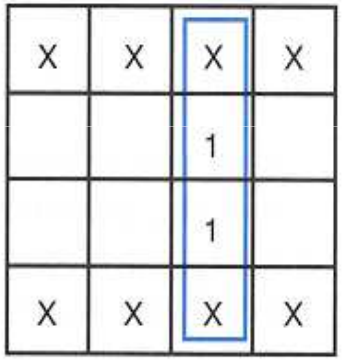
$K_{Q3} = Q_0 Q_1 Q_2$

Contatori sincroni: esempio

Present state				Next state				Flip-flo	
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	J_{Q2}	K_{Q2}
0	0	0	0	0	0	0	1	0	X
0	0	0	1	0	0	1	0	0	X
0	0	1	0	0	0	1	1	0	X
0	0	1	1	0	1	0	0	1	X
0	1	0	0	0	1	0	1	X	0
0	1	0	1	0	1	1	0	X	0
0	1	1	0	0	1	1	1	X	0
0	1	1	1	1	0	0	0	X	1
1	0	0	0	1	0	0	1	0	X
1	0	0	1	1	0	1	0	0	X
1	0	1	0	1	0	1	1	0	X
1	0	1	1	1	1	0	0	1	X
1	1	0	0	1	1	0	1	X	0
1	1	0	1	1	1	1	0	X	0
1	1	1	0	1	1	1	1	X	0
1	1	1	1	0	0	0	0	X	1



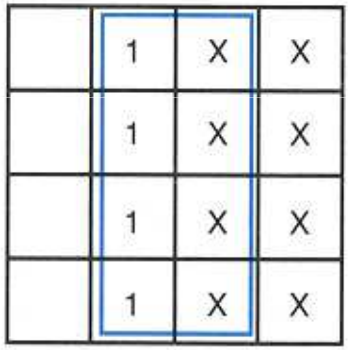
$J_{Q2} = Q_0Q_1$



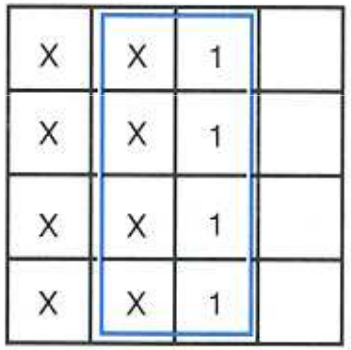
$K_{Q2} = Q_0Q_1$

Contatori sincroni: esempio

Present state				Next state				p inputs	
Q_3	Q_2	Q_1	Q_0	Q_3	Q_2	Q_1	Q_0	J_{Q1}	K_{Q1}
0	0	0	0	0	0	0	1	0	X
0	0	0	1	0	0	1	0	1	X
0	0	1	0	0	0	1	1	X	0
0	0	1	1	0	1	0	0	X	1
0	1	0	0	0	1	0	1	0	X
0	1	0	1	0	1	1	0	1	X
0	1	1	0	0	1	1	1	X	0
0	1	1	1	1	0	0	0	X	1
1	0	0	0	1	0	0	1	0	X
1	0	0	1	1	0	1	0	1	X
1	0	1	0	1	0	1	1	X	0
1	0	1	1	1	1	0	0	X	1
1	1	0	0	1	1	0	1	0	X
1	1	0	1	1	1	1	0	1	X
1	1	1	0	1	1	1	1	X	0
1	1	1	1	0	0	0	0	X	1



$J_{Q1} = Q_0$



$K_{Q1} = Q_0$

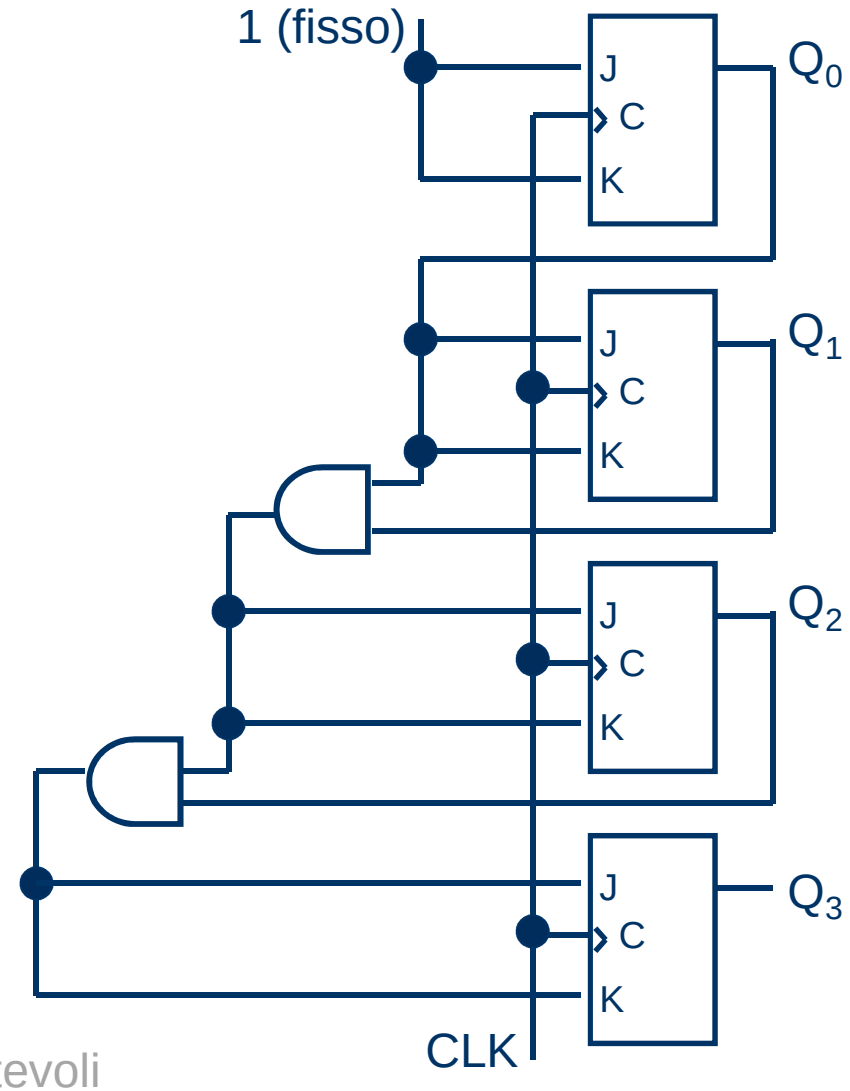
Contatori sincroni: esempio

$J_{Q0} = 1$
$K_{Q0} = 1$

$J_{Q1} = Q_0$
$K_{Q1} = Q_0$

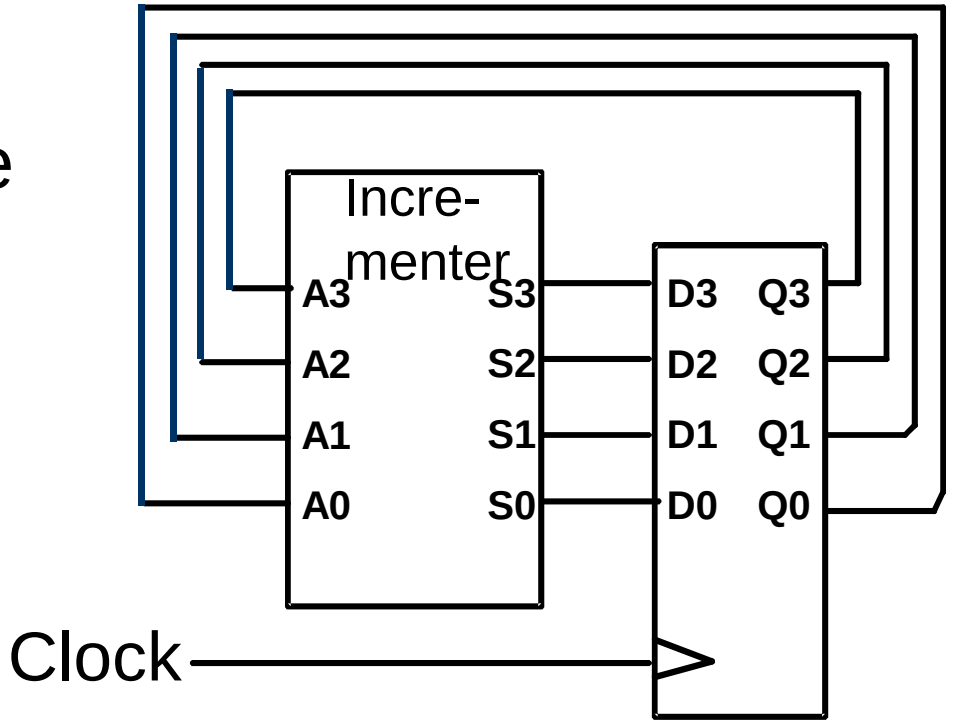
$J_{Q2} = Q_0\,Q_1$
$K_{Q2} = Q_0\,Q_1$

$J_{Q3} = Q_0\,Q_1\,Q_2$
$K_{Q3} = Q_0\,Q_1\,Q_2$



Contatori sincroni

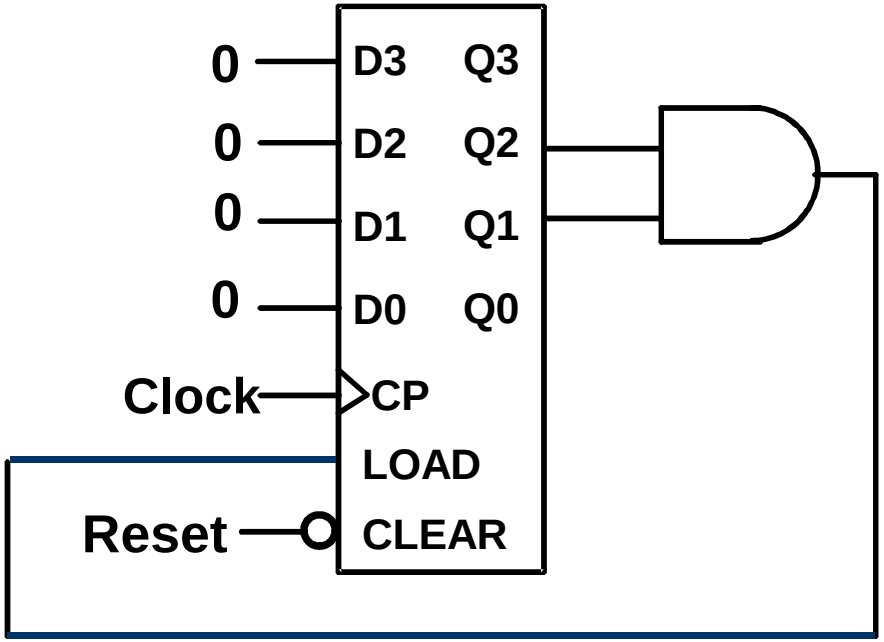
- La rete di aggiornamento può essere realizzata come un addizionatore, o meglio come un “incrementatore”



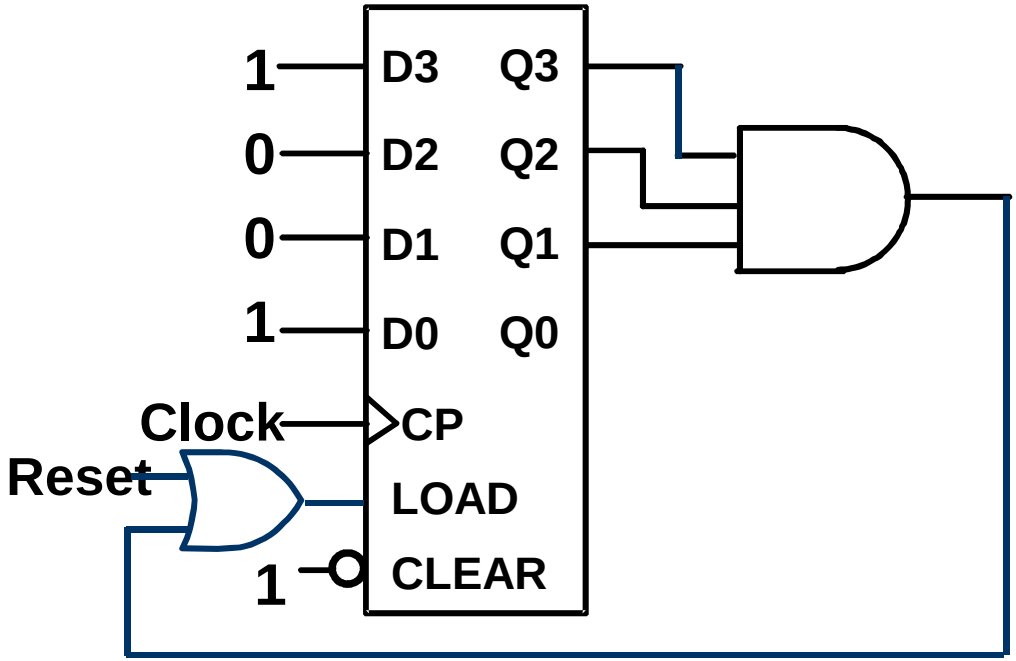
Contatori modulari

- E’ possibile realizzare contatori modulari con un modulo diverso da una potenza di 2
- Basta che il contatore si “auto-resetti” appena ha raggiunto il valore massimo di conteggio

Contatori modulari



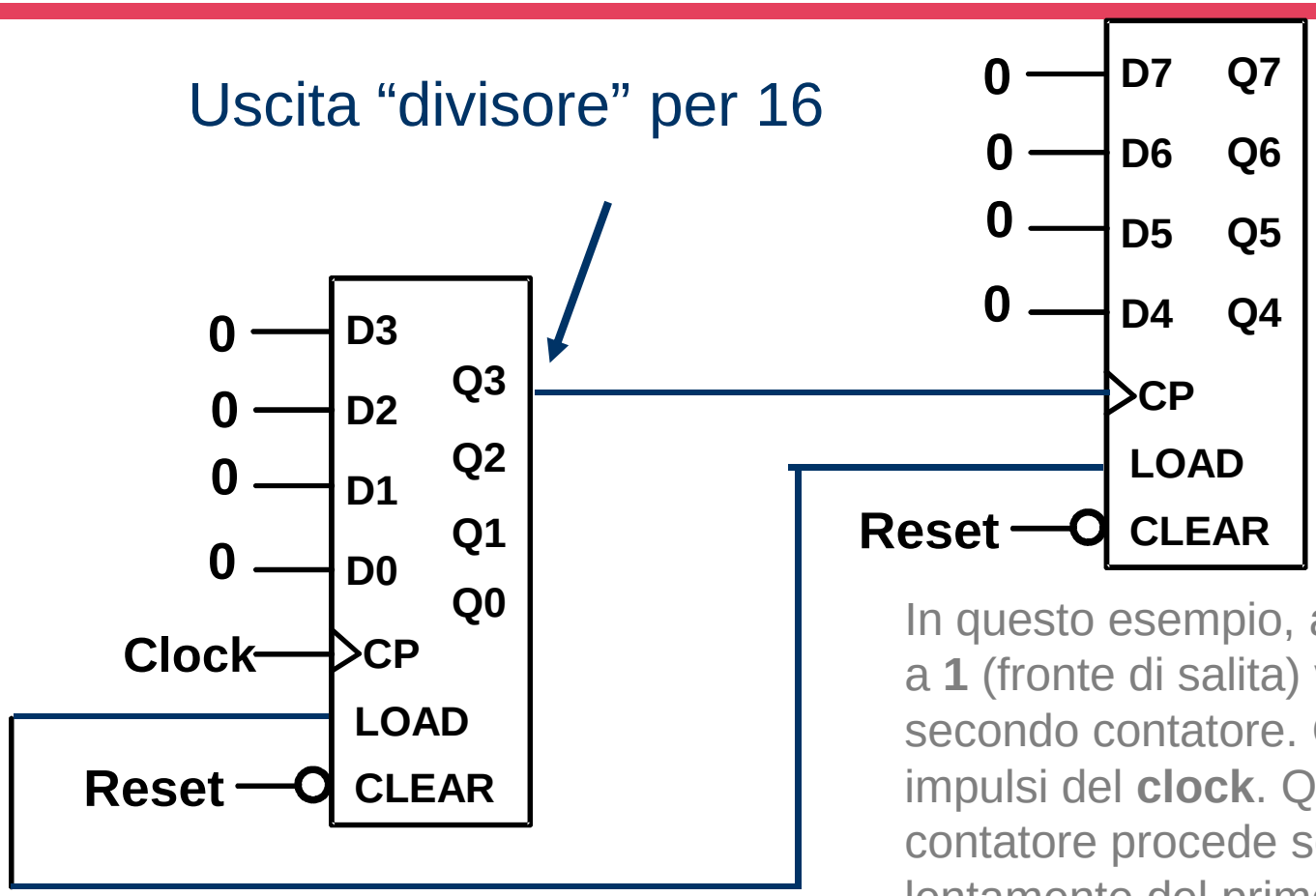
In questo esempio, appena Q_2 e Q_1 diventano insieme **1**, il contatore si resetta e ricomincia da capo. Ciò accade appena il conteggio arriva a $Q_3Q_2Q_1Q_0$ =**0110**. Il componente è un contatore **modulo 7**



In questo esempio, appena Q_3 Q_2 Q_1 diventano insieme **1**, il contatore si resetta e ricomincia dal valore **1001**. Ciò accade appena il conteggio arriva a $Q_3Q_2Q_1Q_0$ =**1110**. Il componente è un contatore **modulo 6**

Comporre contatori

Uscita “divisore” per 16

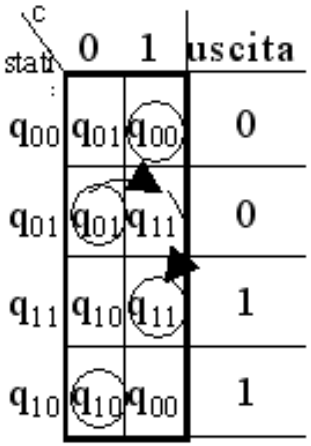


In questo esempio, appena Q_3 passa da **0** a **1** (fronte di salita) viene attivato il secondo contatore. Ciò accade ogni sedici impulsi del **clock**. Quindi il secondo contatore procede sedici volte più lentamente del primo. Complessivamente, le uscite $Q_7\ Q_6\ Q_5\ Q_4\ Q_3\ Q_2\ Q_1\ Q_0$ realizzano un conteggio modulo $2^8=256$

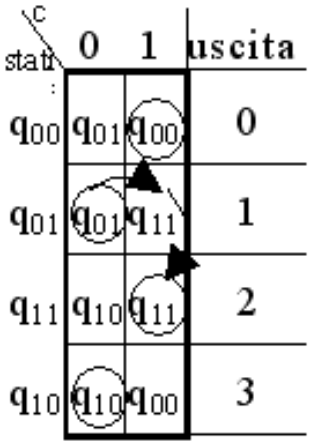
Contatori asincroni

Materiale facoltativo

- Nel modello a stati, sono molto simili al flip-flop T (che è un contatore modulo 2)
  - contano le transizioni dei fronti
  - problemi di alee essenziali



c / stati	0	1	uscita
q_{00}	q_{01}	q_{00}	0
q_{01}	q_{01}	q_{11}	0
q_{11}	q_{10}	q_{11}	1
q_{10}	q_{10}	q_{00}	1



c / stati	0	1	uscita
q_{00}	q_{01}	q_{00}	0
q_{01}	q_{01}	q_{11}	1
q_{11}	q_{10}	q_{11}	2
q_{10}	q_{10}	q_{00}	3