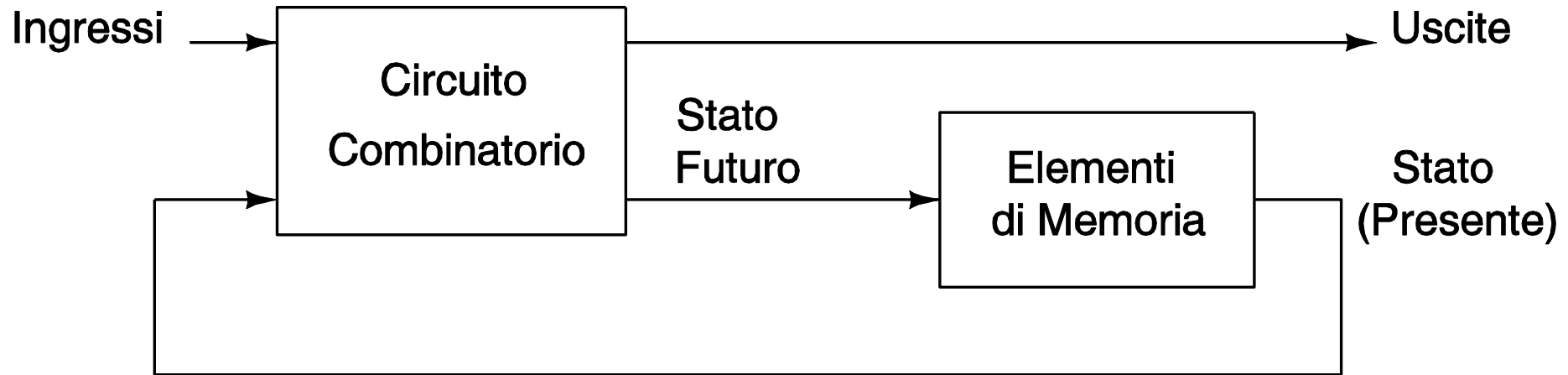


A.A. 2021-2022

Elementi di Elettronica (INF)

Prof. Paolo Crippa

Circuiti Sequenziali



■ Logica Combinatoria

- *Funzione stato futuro*
 $\text{Stato Futuro} = f(\text{Ingressi}, \text{Stato})$
- *Funzione di uscita (Mealy)*
 $\text{Uscite} = g(\text{Ingressi}, \text{Stato})$
- *Funzione di uscita (Moore)*
 $\text{Uscite} = h(\text{Stato})$

- Il tipo di funzione di uscita dipende dalle specifiche e influenza il progetto in maniera significativa

- Dipende dagli istanti di tempo in cui:
 - gli elementi di memoria osservano i loro ingressi, e
 - gli elementi di memoria cambiano il loro stato
- **Sincroni**
 - comportamento definito dalla conoscenza dei segnali in istanti di tempo discreti
 - gli elementi di memoria osservano gli ingressi e possono cambiare stato solo in relazione ad un segnale di temporizzazione (impulsi di clock da un clock)
- **Asincroni**
 - Il comportamento è definito dalla conoscenza degli ingressi ad ogni istante di tempo (continuo) e dal modo in cui essi evolvono nel tempo
 - Se si considera il clock come un altro ingresso, allora tutti i circuiti sono asincroni!
 - Tuttavia, l'astrazione di circuiti sincroni rende trattabili progetti altrimenti complessi!

Elementi di Memoria

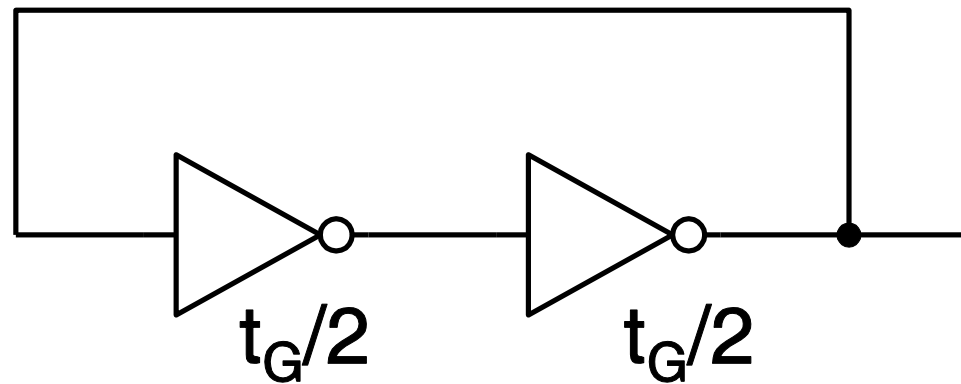
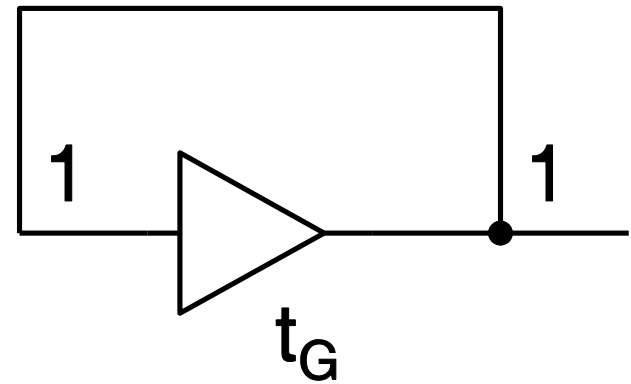
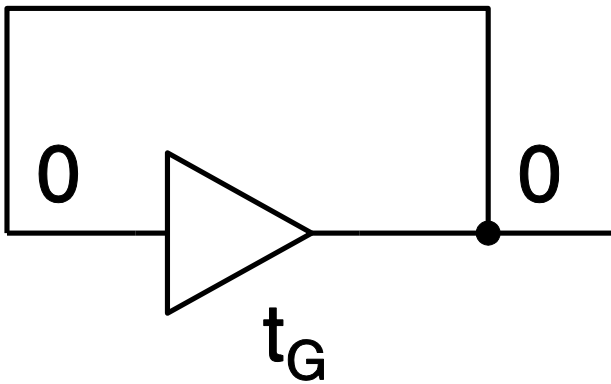
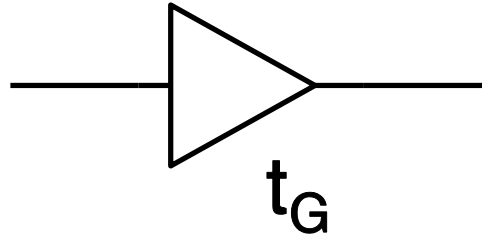


Diagramma a Blocchi

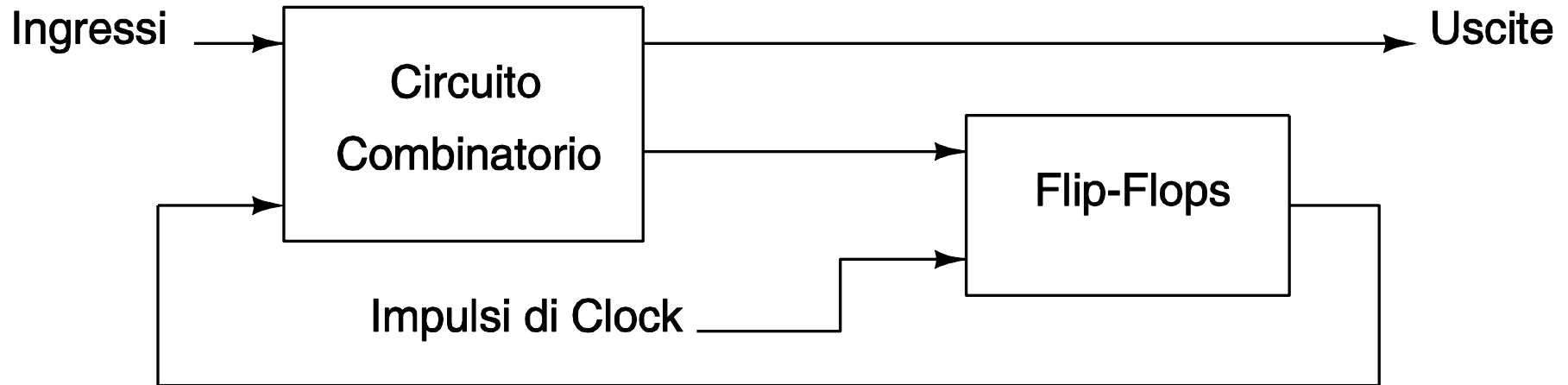
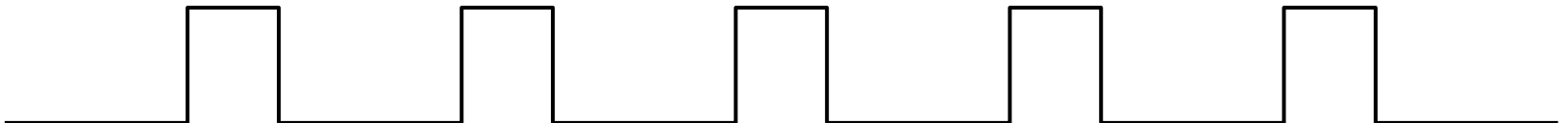
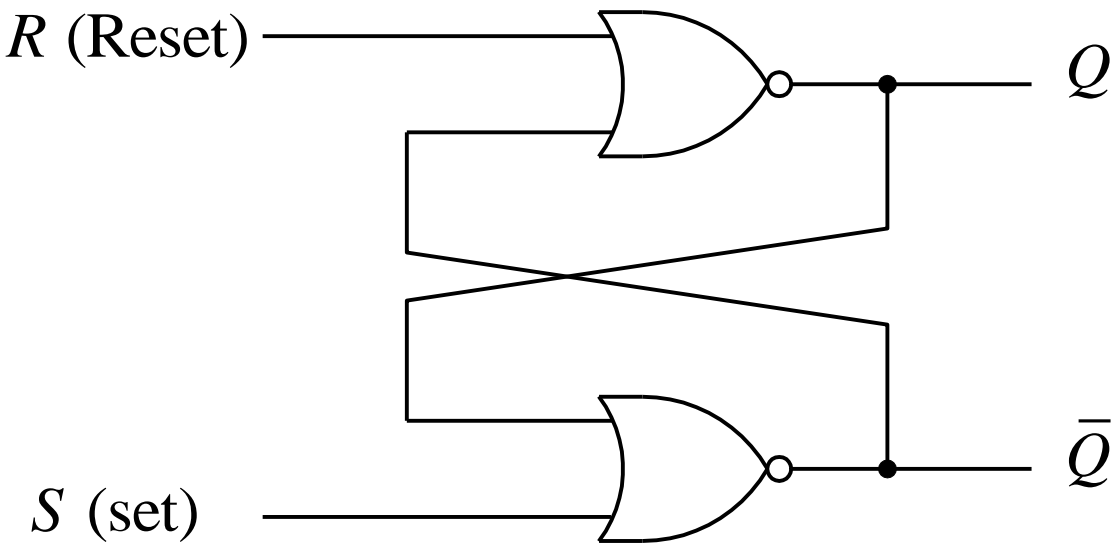


Diagramma di temporizzazione del segnale di clock



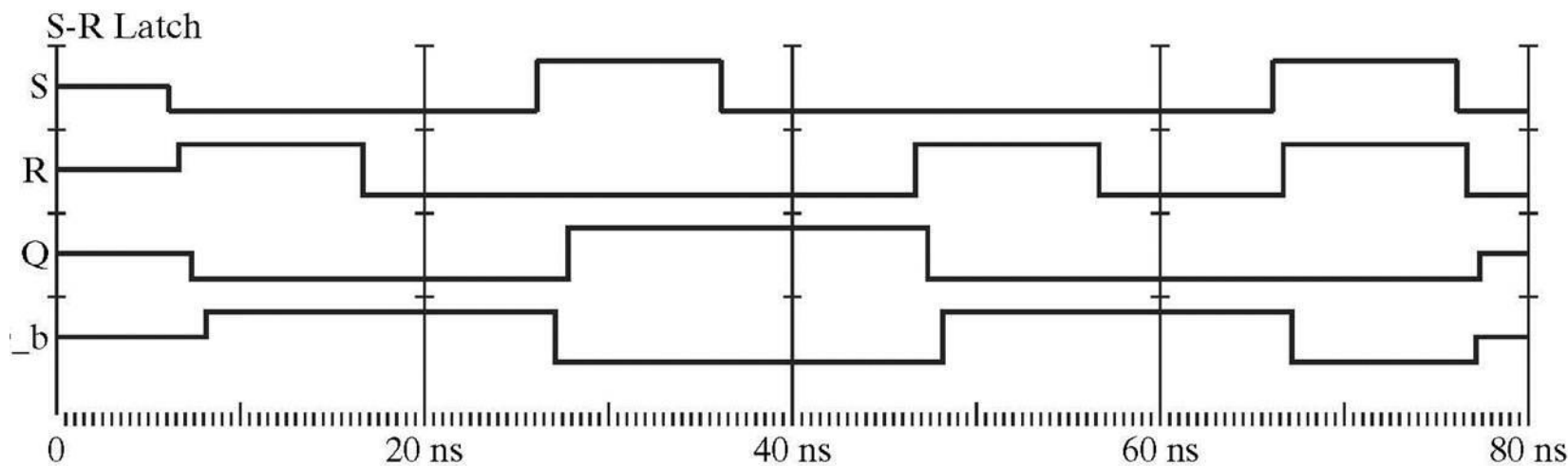
Latch S – R (con Porte NOR)



S	R	Q	\bar{Q}	
1	0	1	0	Stato di Set
0	0	1	0	
0	1	0	1	Stato di Reset
0	0	0	1	
1	1	0	0	Non Ammesso

Diagramma Logico

Tabella Funzionale



Simulazione Logica

Latch \overline{S} – \overline{R} (con Porte NAND)

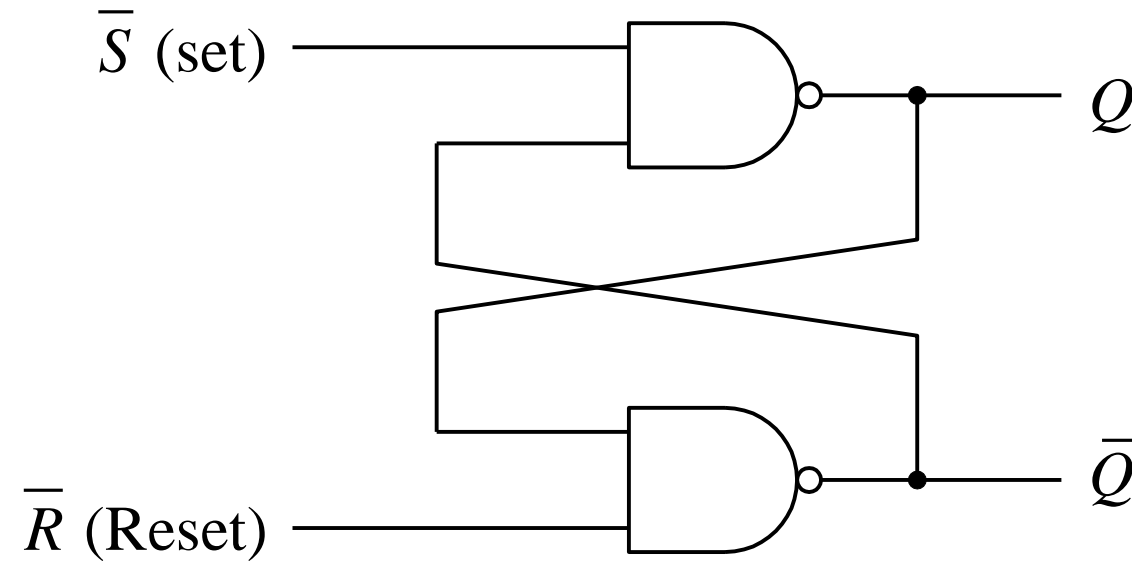


Diagramma Logico

\overline{S}	\overline{R}	Q	\overline{Q}	
0	1	1	0	Stato di Set
1	1	1	0	
1	0	0	1	Stato di Reset
1	1	0	1	
0	0	1	1	Non Ammesso

Tabella Funzionale

Latch S – R con Ingresso di Controllo

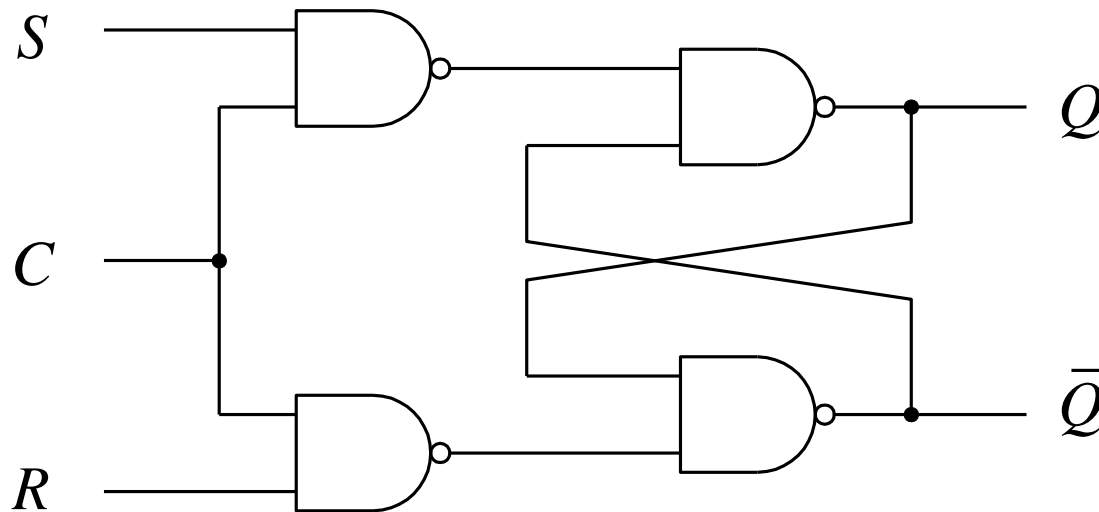


Diagramma Logico

C	S	R	Stato futuro di Q
0	X	X	No variazione
1	0	0	No variazione
1	0	1	Q=0, Stato di reset
1	1	0	Q=1, Stato di set
1	1	1	Non ammesso

Tabella Funzionale

- Aggiungendo due porte NAND al latch \overline{S} - \overline{R} a NAND si ottiene il cosiddetto **latch S – R sincrono (clocked S – R latch)**
- Ha un comportamento nel tempo simile al latch S-R eccetto che gli ingressi S e R sono osservati solo quando la linea C è al valore alto.
- C significa “control” o “clock”.

Latch D Sincrono

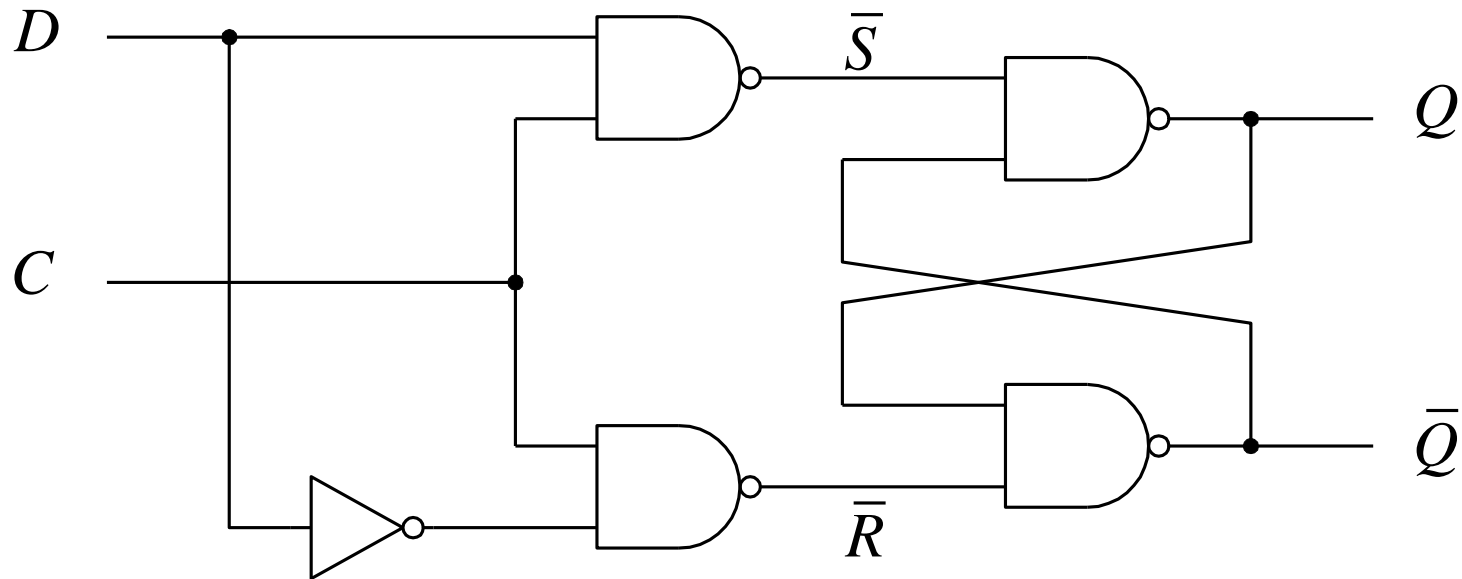
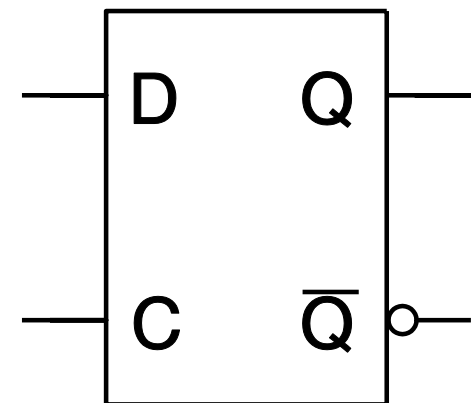


Diagramma Logico

C	D	Stato futuro di Q
0	X	No variazione
1	0	Q=0, Stato di reset
1	1	Q=1, Stato di set

Tabella Funzionale



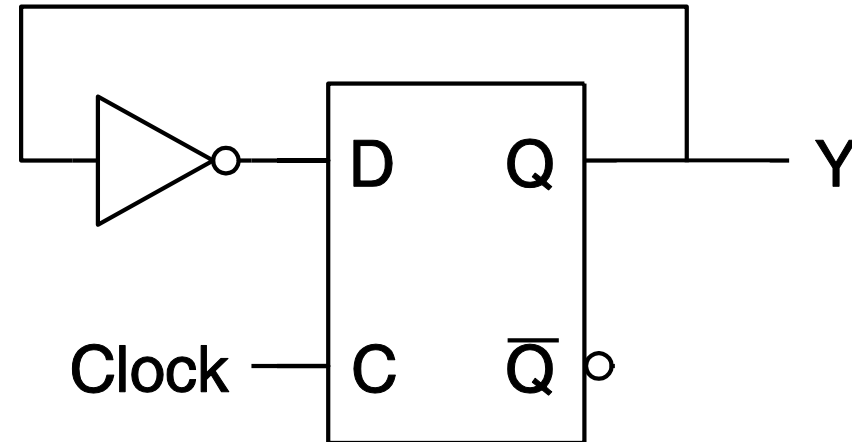
Simbolo Grafico

- Il latch timing problem
- Master-slave flip-flop
- Edge-triggered flip-flop
- Simboli standard per gli elementi di memoria
- Ingressi diretti per flip-flops

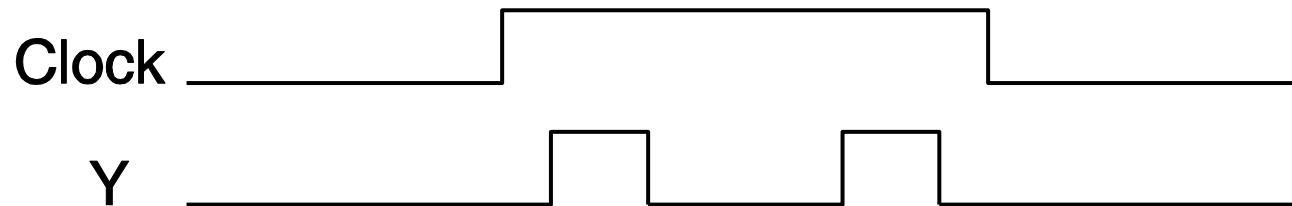
- In un circuito sequenziale possono esistere dei cammini all'interno della logica combinatoria:
 - Da un elemento di memoria all'altro
 - Da un elemento di memoria all'indietro a se stesso
- La logica combinatoria fra un'uscita di un latch e un ingresso di un latch può essere una semplice interconnessione.
- Per un latch D sincrono, l'uscita Q dipende dall'ingresso D allorché l'ingresso di clock C vale 1.

Il Problema del “Latch Timing” - II

- Consideriamo il seguente circuito:



- Supponiamo che inizialmente $Y = 0$.

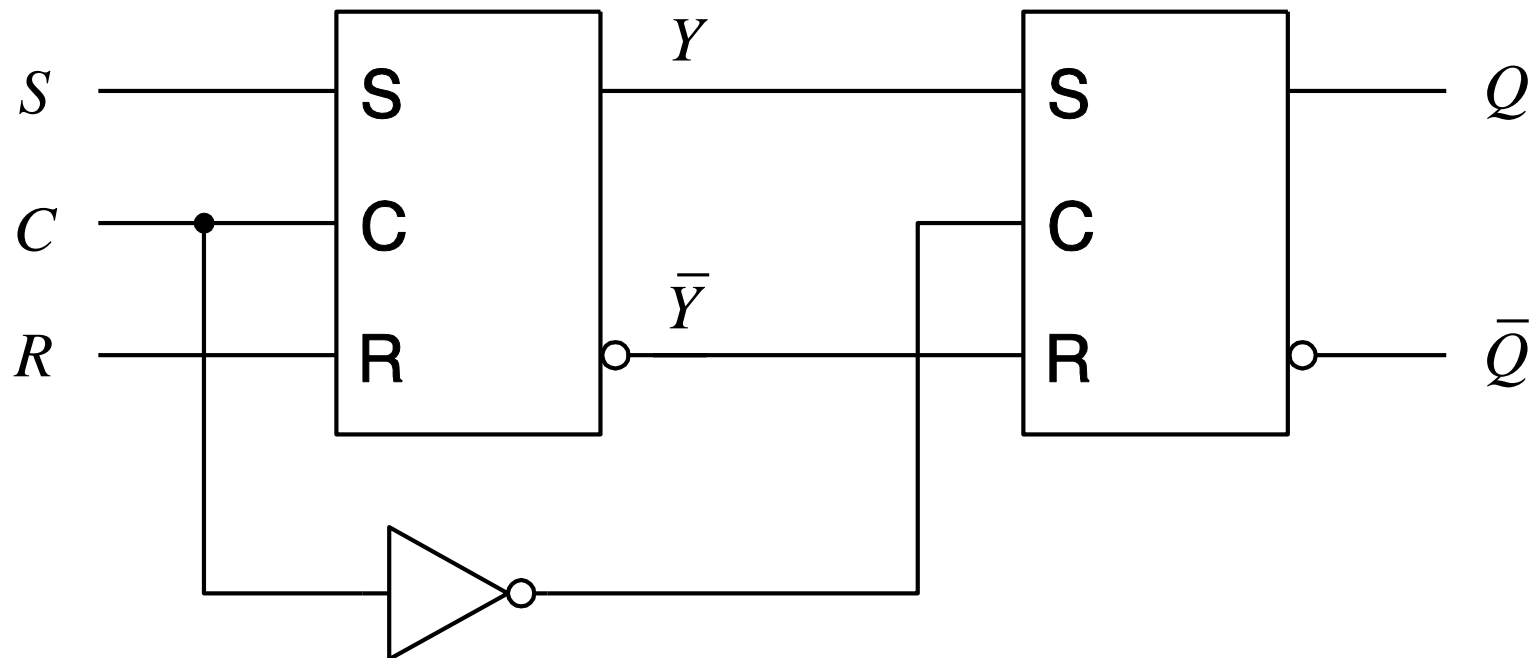


- Fintanto che $C = 1$, il valore di Y continua a cambiare!
- I cambi sono dovuti al ritardo (delay) presente nel percorso di retroazione da Y a Y
- Questo comportamento è ovviamente inaccettabile
- Comportamento desiderato: Y cambia una volta sola ad ogni impulso di clock

Il Problema del “Latch Timing” - III

- Una soluzione al problema del latch timing è quella di interrompere il percorso chiuso (anello di retroazione) da Y a Y all'interno dell'elemento di memoria
- La soluzione comunemente usata per interrompere il percorso di retroazione è quella di sostituire il latch D sincrono con :
 - **un flip-flop master-slave**
 - **un flip-flop edge-triggered**

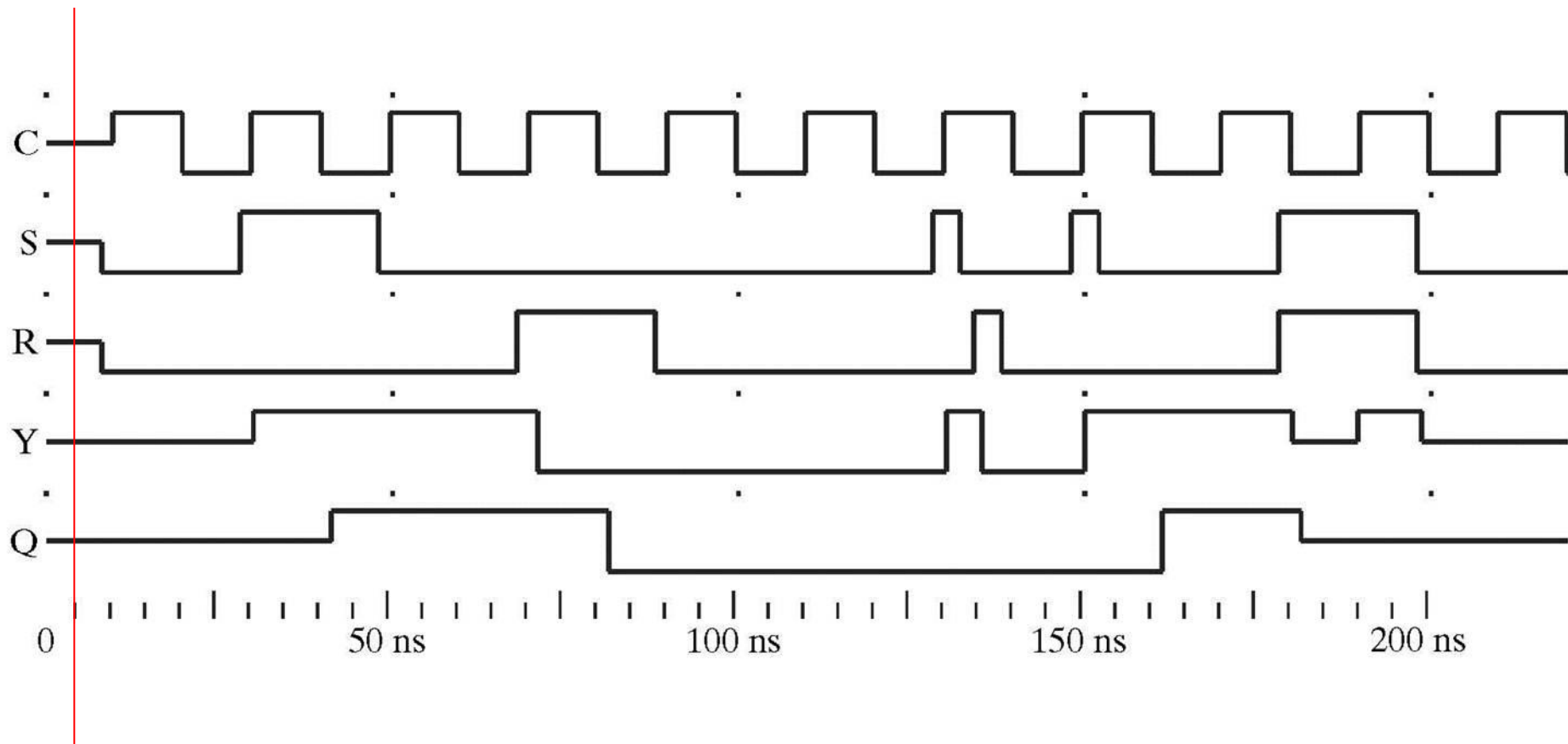
Flip Flop S – R Master Slave



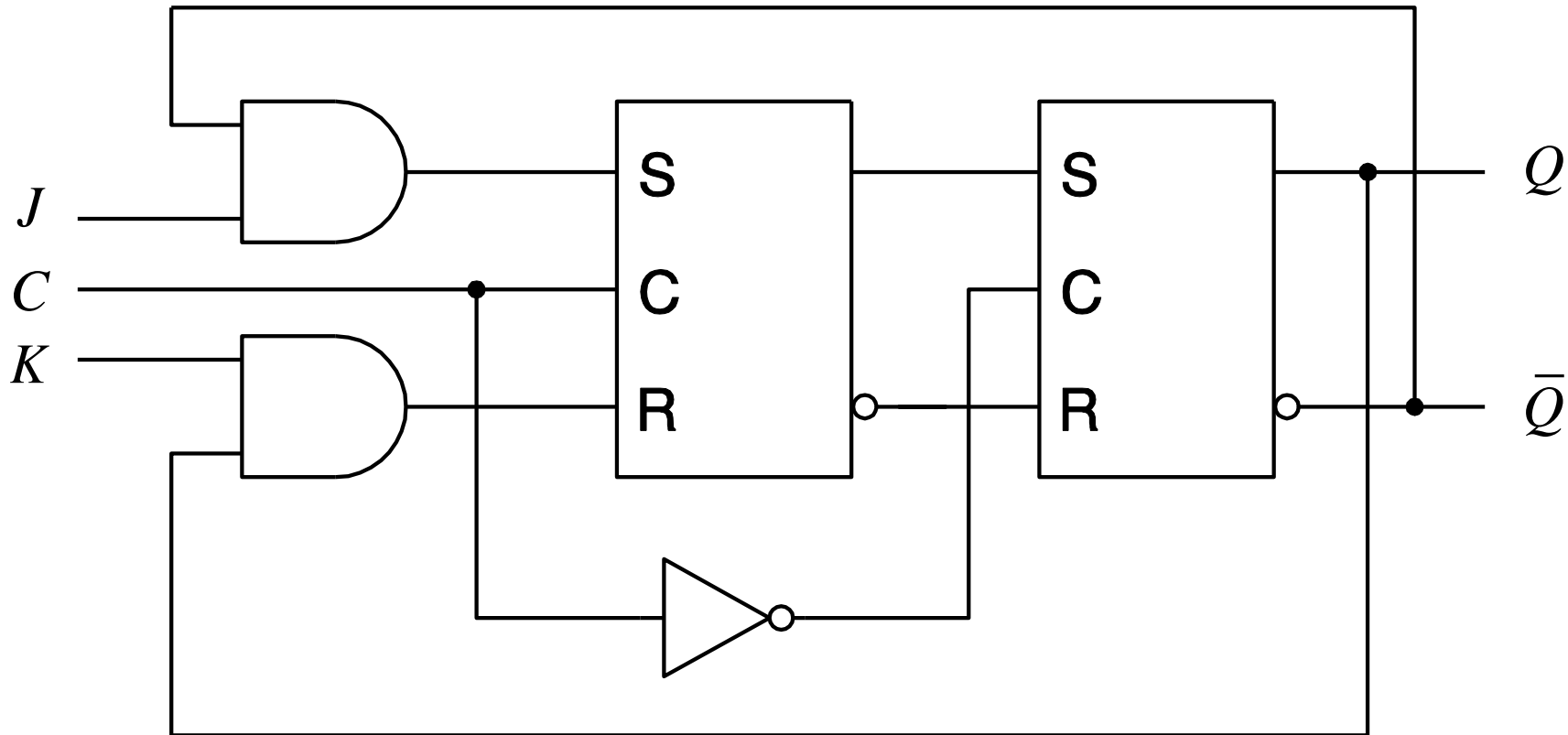
- E' costituito da due latch S-R sincroni in serie con il clock del secondo latch invertito.
- L'ingresso è osservato dal primo latch quando $C = 1$
- L'uscita è cambiata dal secondo latch quando $C = 0$
- Il percorso ingresso-uscita è interrotto dalla differenza nei valori di clock ($C = 1$ e $C = 0$).

Flip Flop S – R Master Slave

Simulazione Logica



Flip Flop J – K Master Slave



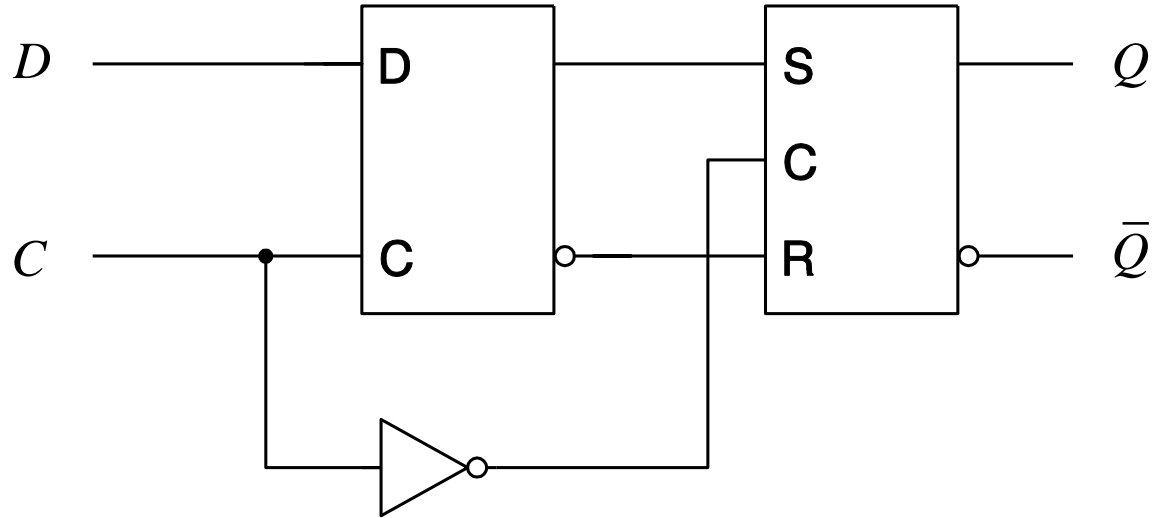
J	K	Stato Futuro Q
0	0	Q
0	1	0
1	0	1
1	1	\bar{Q}

- La variazione dell'uscita del flip-flop è ritardata della larghezza dell'impulso che rende il circuito più lento, oppure
- S e/o R possono cambiare mentre $C = 1$
 - Supponiamo che $Q = 0$ e S vada a 1 e poi ritorni a 0 con R che rimane a 0
 - Il latch master si setta a 1
 - Un 1 viene trasferito allo slave
 - Supponiamo che $Q = 0$ e S vada a 1 e poi ritorni a 0 e R vada a 1 e poi di nuovo a 0
 - Il latch master si setta (a 1) e poi si resetta (a 0)
 - Uno 0 viene trasferito allo slave
 - Questo comportamento è detto “*1s catching*”

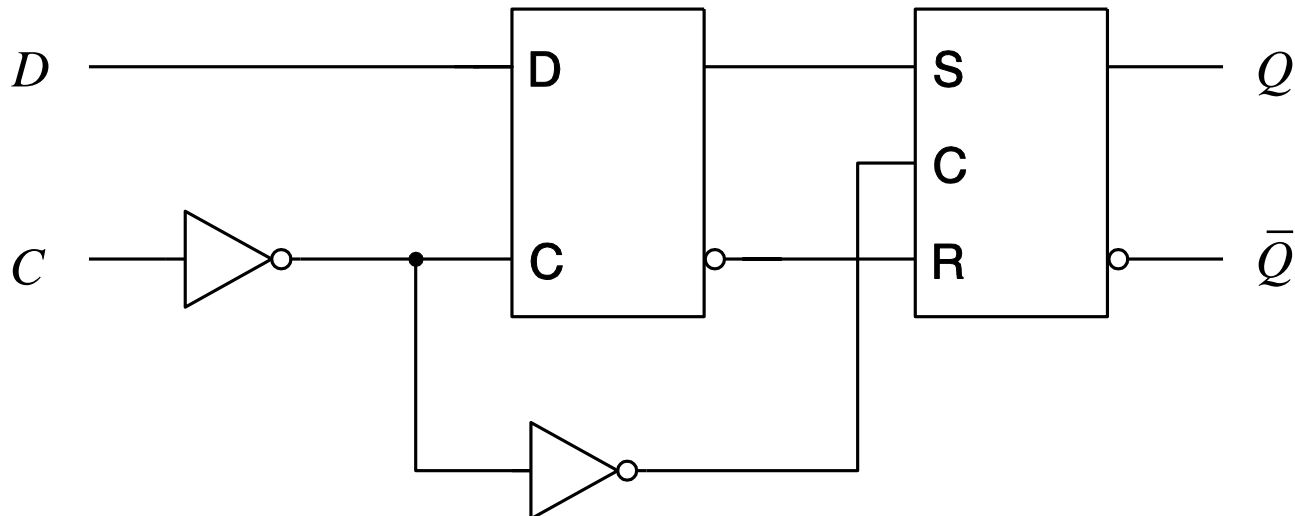
- Si utilizza un flip-flop ***edge-triggered*** invece di un flip-flop master-slave.
- Un flip-flop edge-triggered ignora l'impulso mentre è ad un livello costante e commuta solo durante una transizione del segnale di clock.
- I flip-flop edge-triggered si possono costruire direttamente a livello di circuito elettronico, oppure
- si può usare un flip flop D master-slave che presenta anche un comportamento edge-triggered.

Flip-Flop D Edge-Triggered

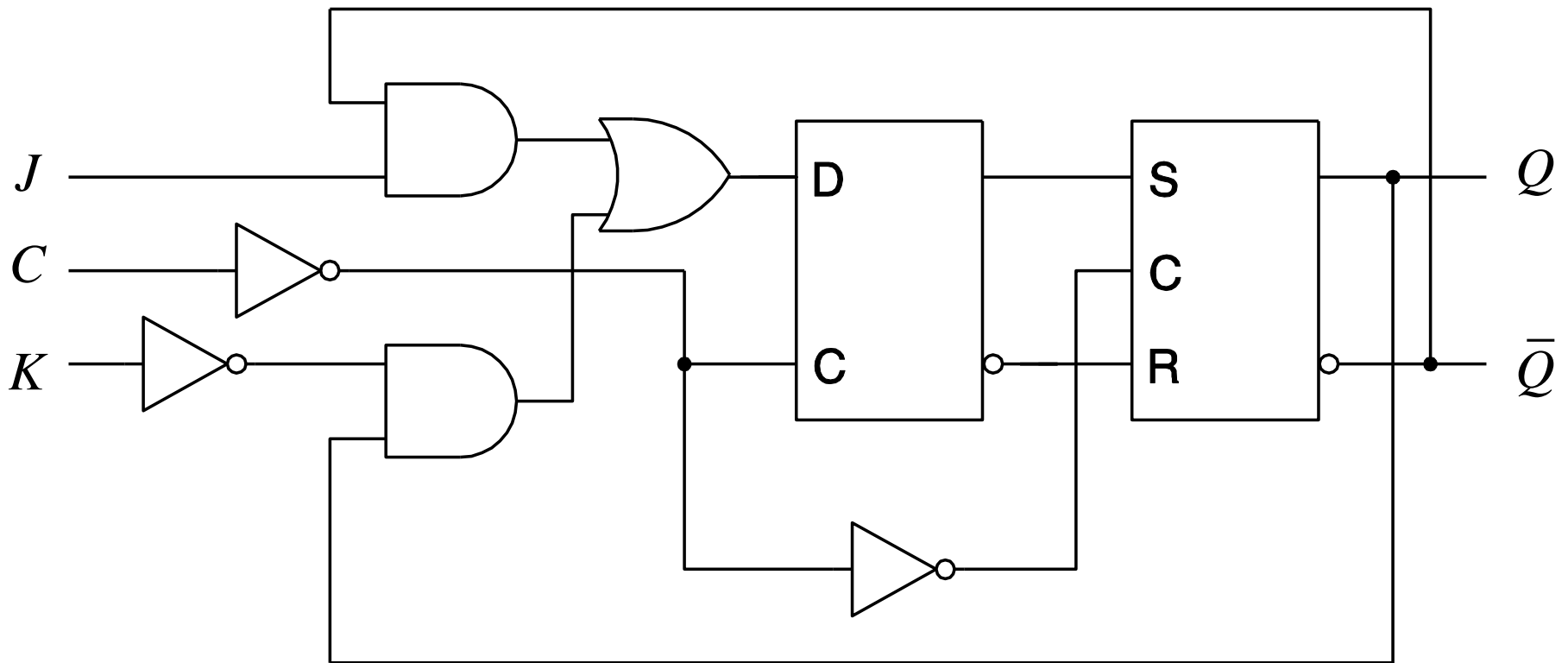
Flip-flop di tipo D sensibile ai fronti di discesa (negative-edge triggered)



Flip-flop di tipo D sensibile ai fronti di salita (positive-edge triggered)



**Elementi di
Elettronica
(INF)
A.A. 2021-22**



Flip-Flop: Tabelle Caratteristiche

Flip-Flop S-R

S	R	Q(t+1)	Operazione
0	0	Q(t)	Quiescenza
0	1	0	Reset
1	0	1	Set
1	1	?	Indefinito

Flip-Flop J-K

J	K	Q(t+1)	Operazione
0	0	Q(t)	Quiescenza
0	1	0	Reset
1	0	1	Set
1	1	$\overline{Q(t)}$	Complemento

Flip-Flop D

D	Q(t+1)	Operazione
0	0	Reset
1	1	Set

Flip-Flop T

T	Q(t+1)	Operazione
0	Q(t)	Quiescenza
1	$\overline{Q(t)}$	Complemento

Flip-Flop S-R: Equazione Caratteristica

Flip-Flop S-R

S	R	Q(t+1)	Operazione
0	0	Q(t)	Quiescenza
0	1	0	Reset
1	0	1	Set
1	1	?	Indefinito

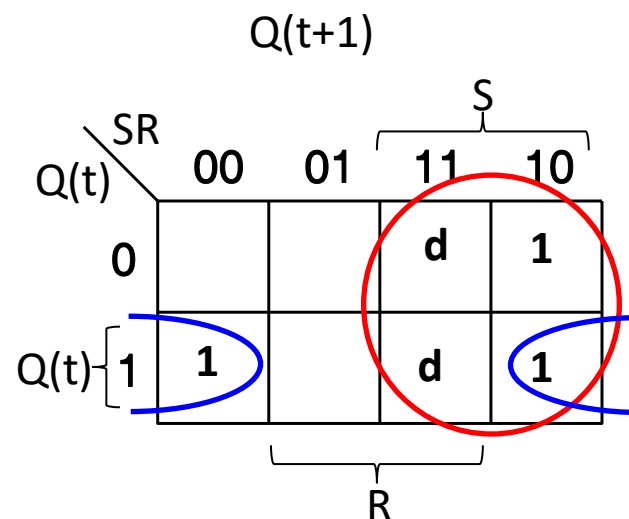
Equazione Caratteristica

$$Q(t+1) = S(t) + \bar{R}(t) \cdot Q(t)$$



Tabella di verità

S	R	Q(t)	Q(t+1)
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	d
1	1	1	d



Flip-Flop J-K: Equazione Caratteristica

Flip-Flop J-K

J	K	Q(t+1)	Operazione
0	0	Q(t)	Quiescenza
0	1	0	Reset
1	0	1	Set
1	1	$\overline{Q(t)}$	Complemento

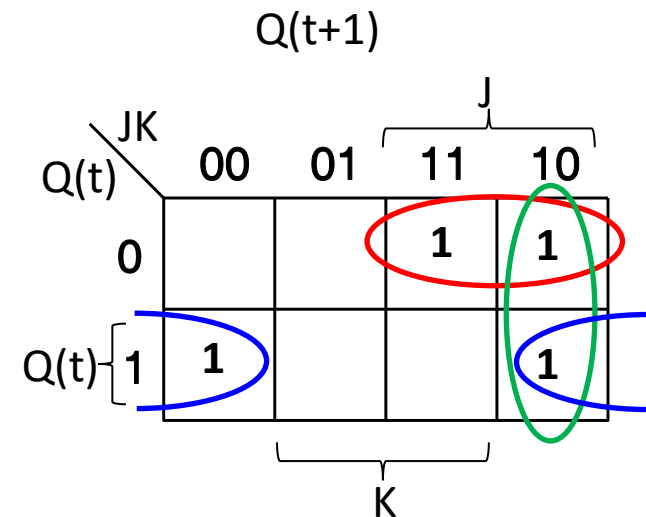
Equazione Caratteristica

$$Q(t+1) = J(t) \cdot \overline{Q}(t) + \overline{K}(t) \cdot Q(t)$$



Tabella di verità

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



Flip-Flop D: Equazione Caratteristica

Flip-Flop D

D	Q(t+1)	Operazione
0	0	Reset
1	1	Set

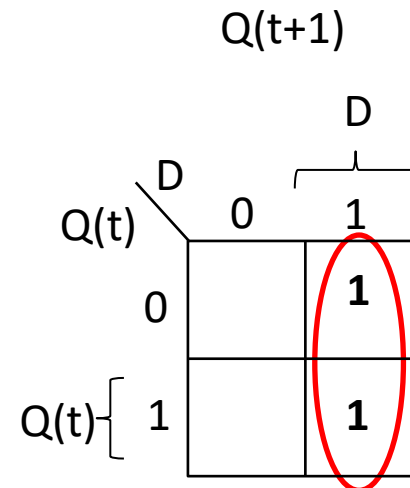
Equazione Caratteristica

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



Tabella di verità

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



Flip-Flop T: Equazione Caratteristica

Flip-Flop T		
T	Q(t+1)	Operazione
0	$\underline{Q(t)}$	Quiescenza
1	$\overline{Q(t)}$	Complemento

Equazione Caratteristica

$$Q(t+1) = \overline{T}(t) \cdot Q(t) + T(t) \cdot \overline{Q}(t) \\ = T(t) \oplus Q(t)$$



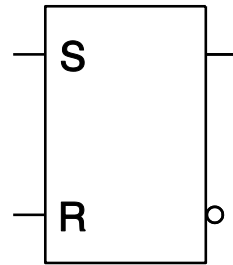
Tabella di verità		
T	Q(t)	Q(t+1)
0	0	0
0	1	1
1	0	1
1	1	0



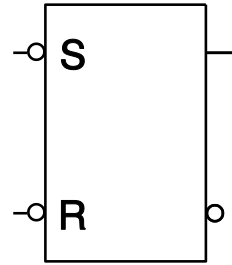
		Q(t+1)	
		T	
Q(t)	T	0	1
	0		1
Q(t)	1	1	

Simboli Grafici Standard per Latch e Flip-Flop

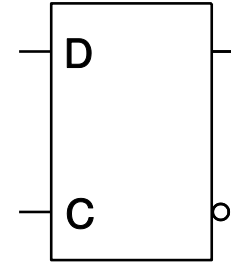
Latch



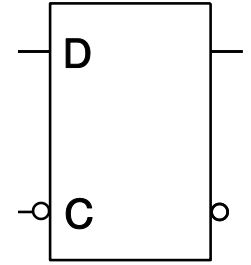
SR



$\overline{S}\overline{R}$

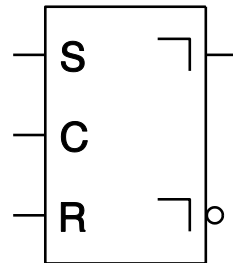


D con controllo
attivo alto

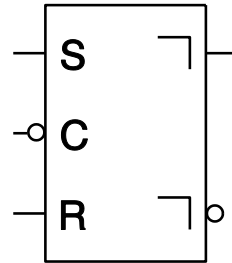


D con controllo
attivo basso

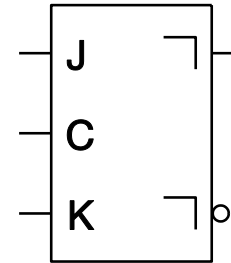
Flip-flop master-slave



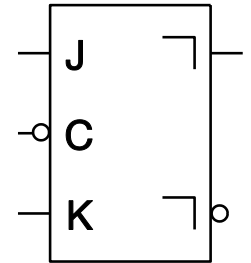
 Triggered SR



 Triggered SR

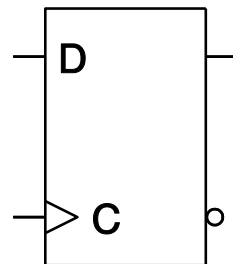


 Triggered JK

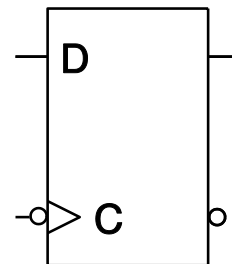


 Triggered JK

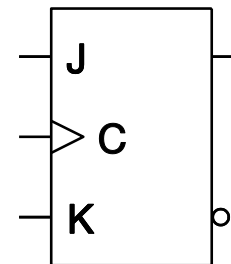
Flip-flop sensibili ai fronti (edge triggered)



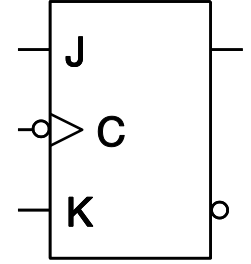
 Triggered D



 Triggered D



 Triggered JK



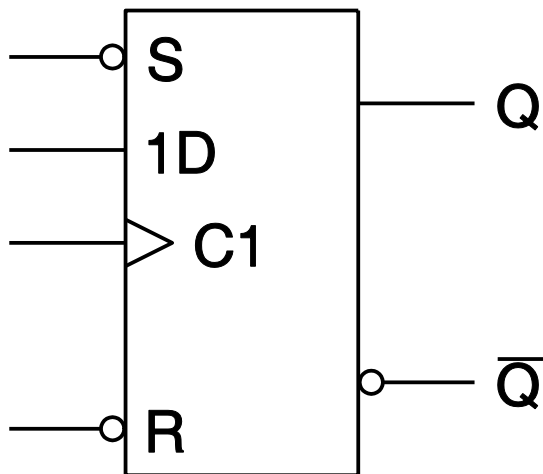
 Triggered JK

Flip-Flop: Ingressi Diretti

Ingressi “speciali”:

- Ingresso di impostazione asincrono: direct set (preset)
- Ingresso di azzeramento asincrono: direct reset (clear)

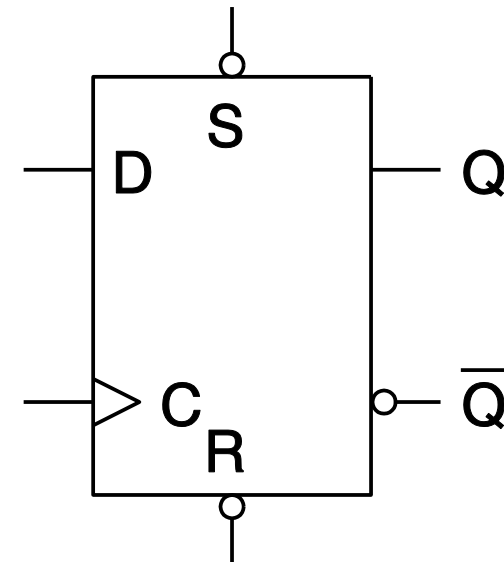
Flip-Flop di tipo D con ingressi di Set e Reset diretti



Simbolo Grafico

S	R	C	D	Q	\bar{Q}
0	1	d	d	1	0
1	0	d	d	0	1
0	0	d	d	Indefinito	
1	1	↑	0	0	1
1	1	↑	1	1	0

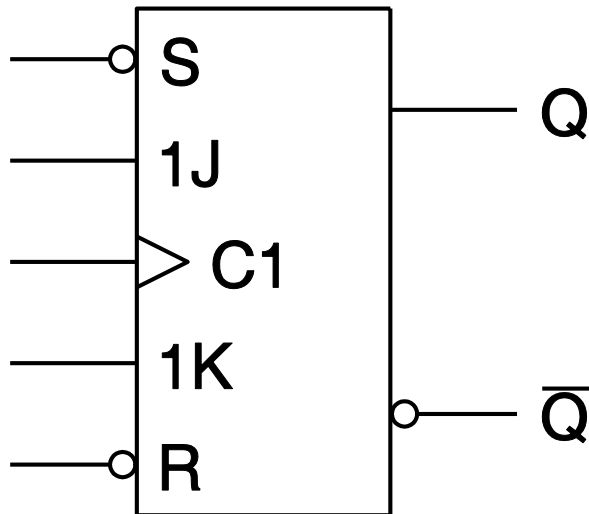
Tabella Funzionale



Simbolo
Semplificato

Flip-Flop: Ingressi Diretti

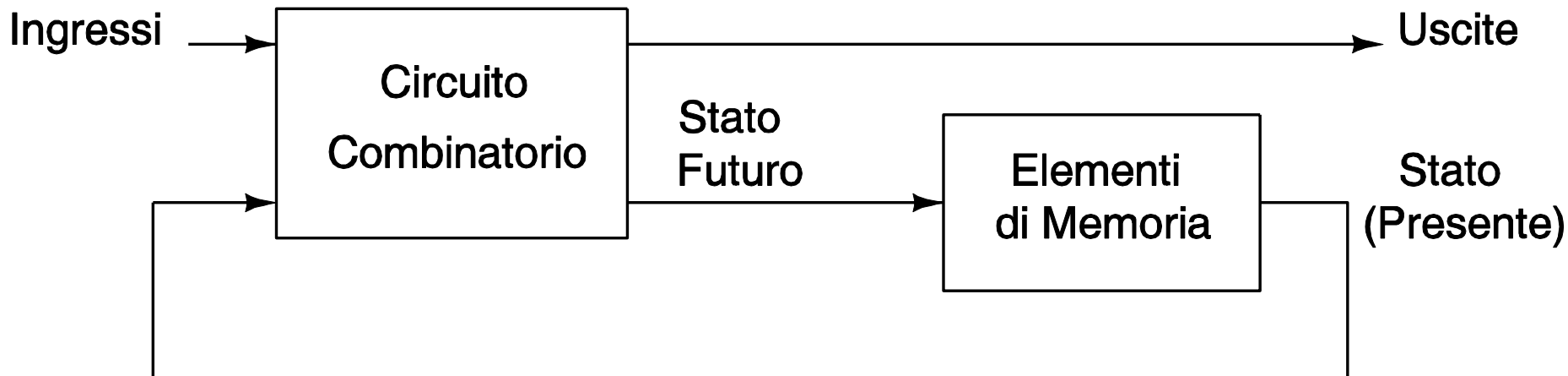
Flip-Flop di tipo J – K con ingressi di Set e Reset diretti



Simbolo Grafico

S	R	C	J	K	Q	\bar{Q}
0	1	d	d	d	1	0
1	0	d	d	d	0	1
0	0	d	d	d	Non ammesso	
1	1	↑	0	0	Nessun cambiamento	
1	1	↑	0	1	1	0
1	1	↑	1	0	0	1
1	1	↑	1	1	Complemento	

Tabella Funzionale

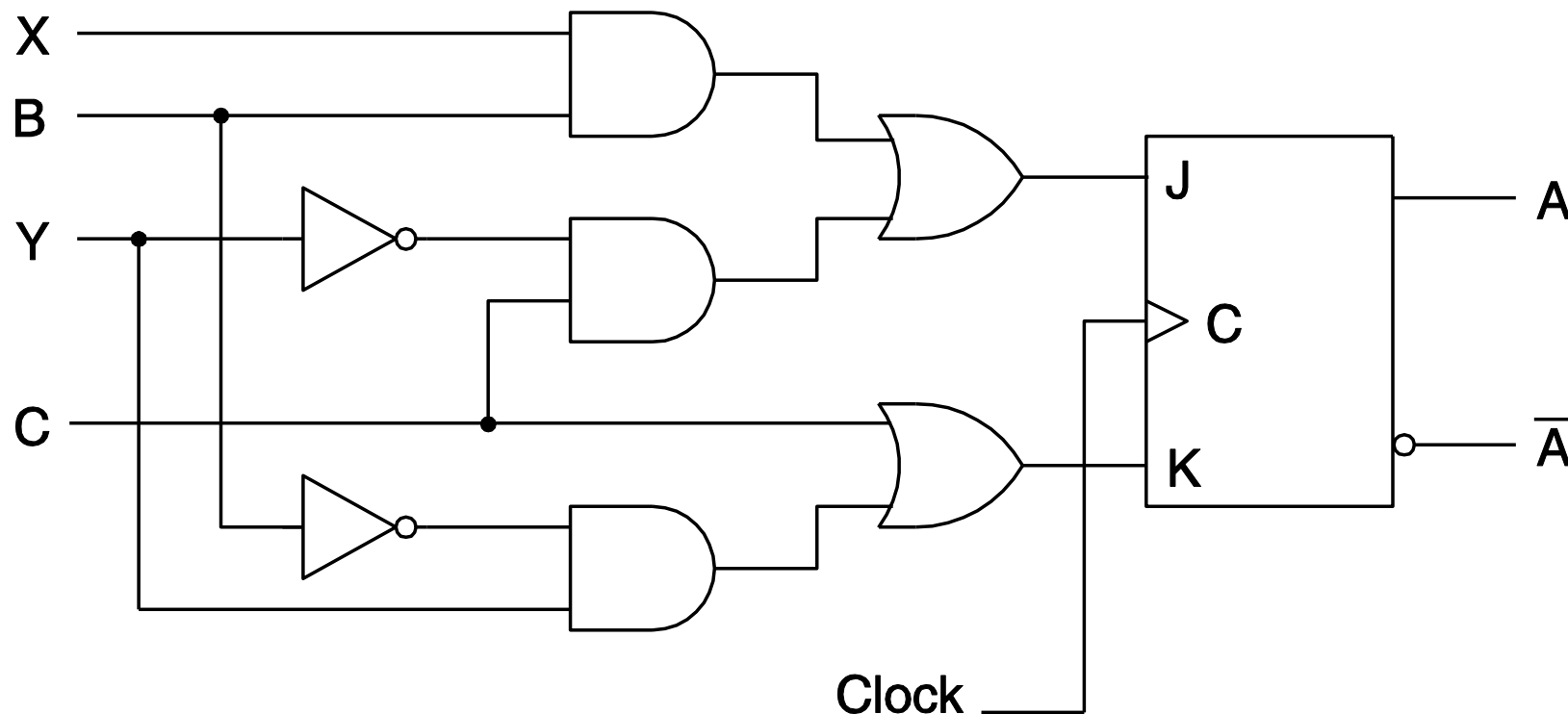


- **Modello Generale**

- Lo Stato al tempo (t) è immagazzinato in un array di flip-flop.
- Lo Stato Futuro al tempo $(t+1)$ è una funzione Booleana dello Stato (t) e degli Ingressi (t) .
- Le Uscite al tempo (t) sono funzioni Booleane dello Stato (t) e (a volte) degli Ingressi (t) .

- **Un circuito sequenziale è sincrono se è composto da flip-flop dotati di ingresso di clock.**

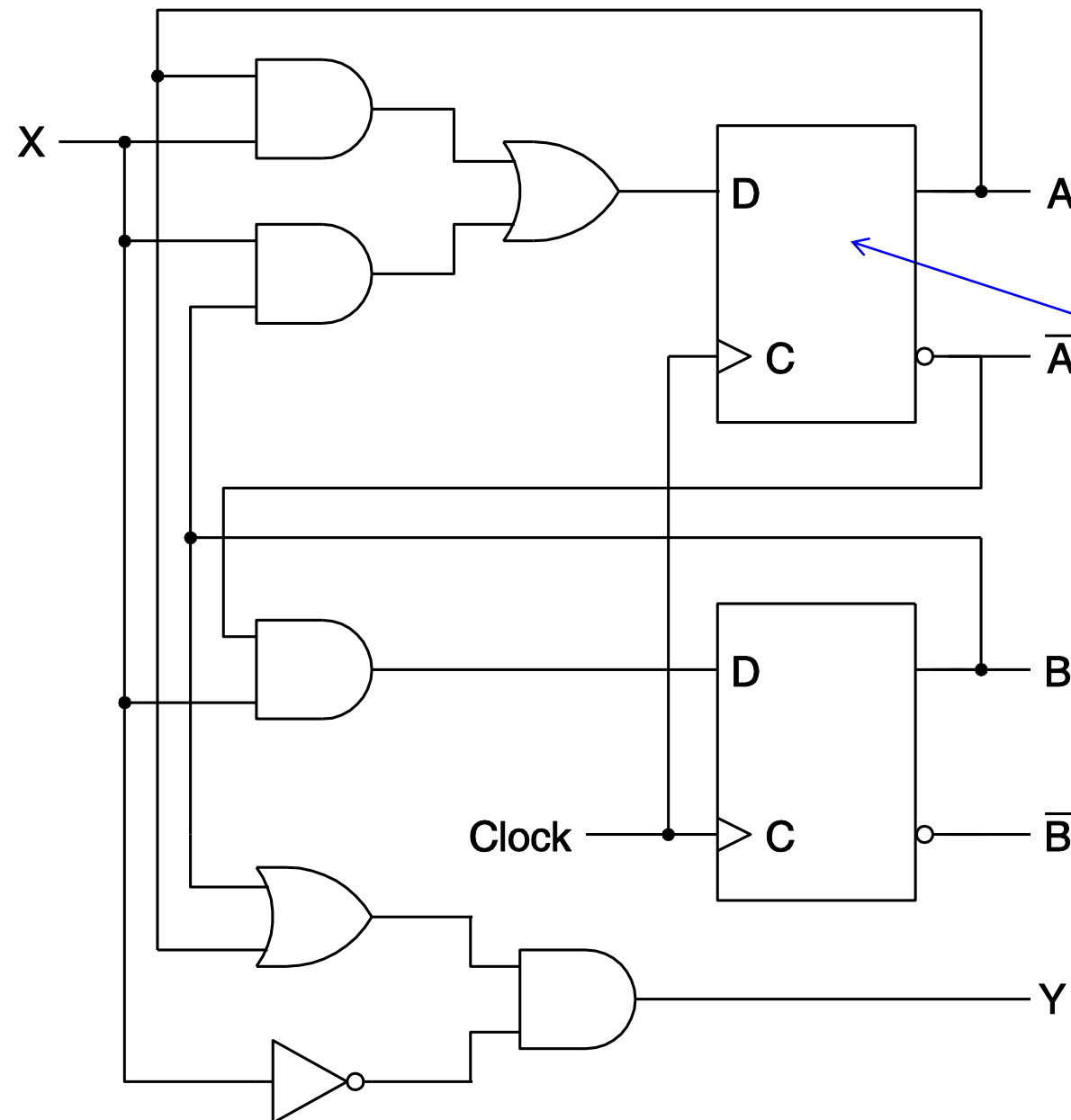
Equazioni di Ingresso ai Flip-Flop (J-K)



$$J_A = (X \cdot B + \bar{Y} \cdot C)$$

$$K_A = (Y \cdot \bar{B} + C)$$

Equazioni di Ingresso ai Flip-Flop (D)



Ingresso del F-F "A"

$$D_A = (A \cdot X + B \cdot X)$$

F-F "A"

Uscita del F-F "A"

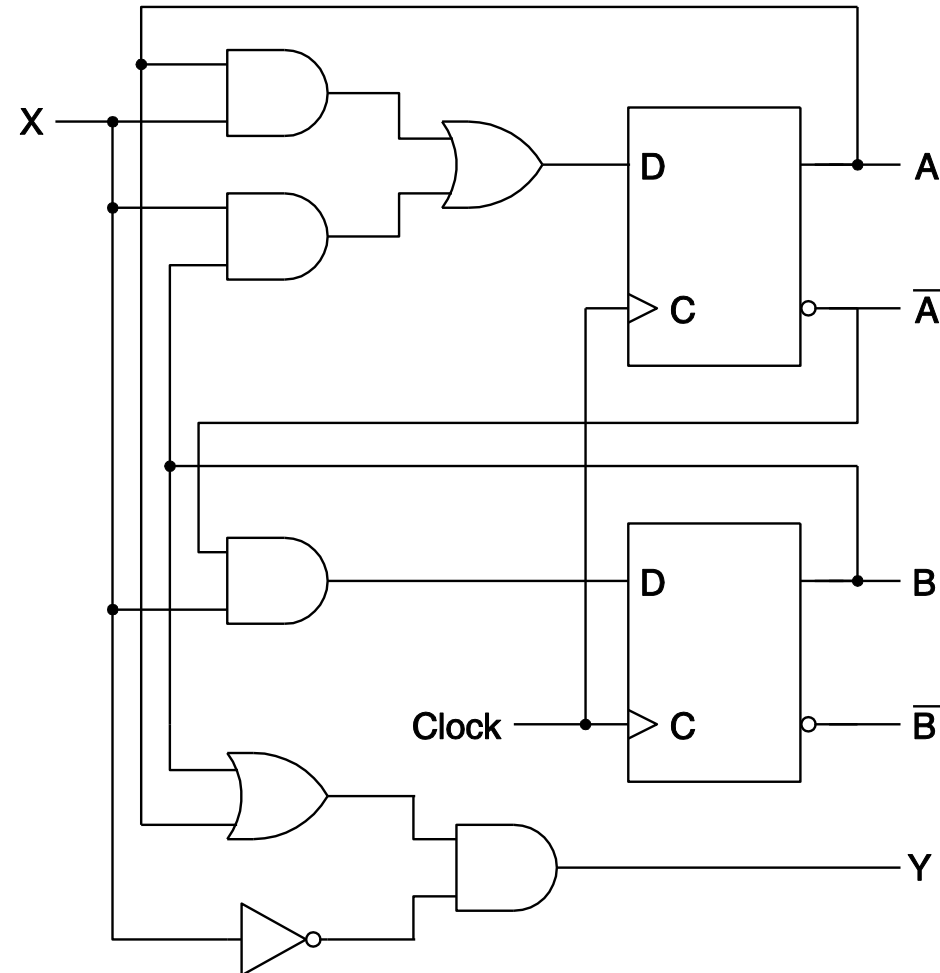
$$D_B = \bar{A} \cdot X$$

Equazioni di Uscita

$$Y = (A + B) \cdot \bar{X}$$

Analisi di Circuiti Sequenziali: Esempio

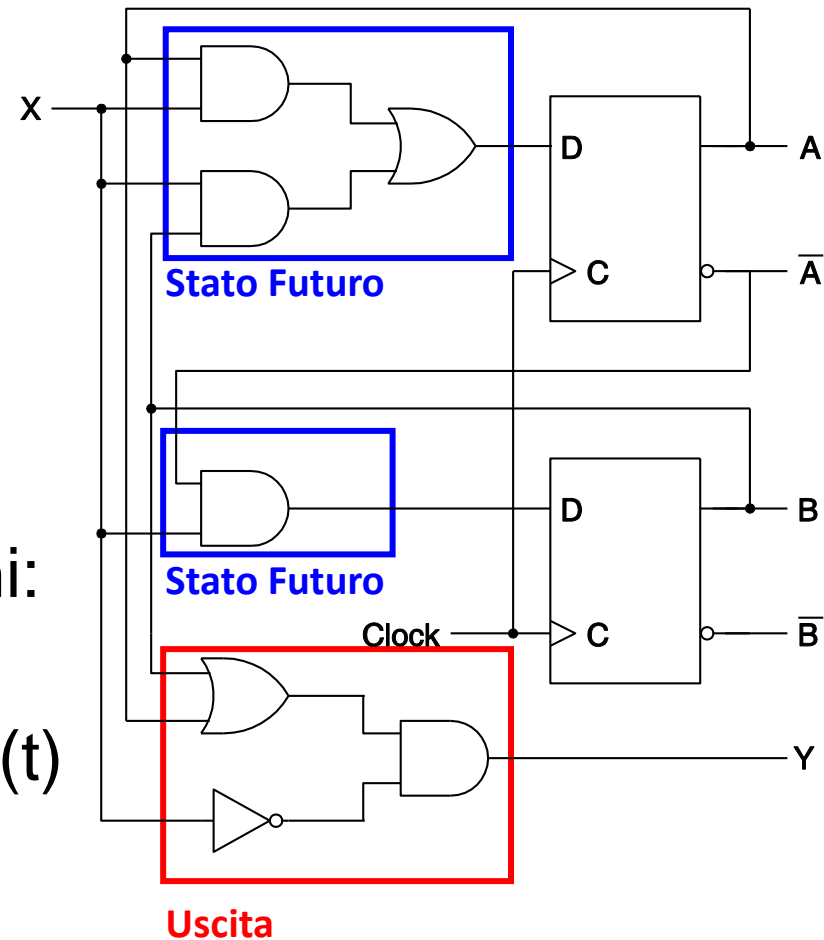
- Ingresso: $x(t)$
- Uscita: $y(t)$
- Stato: $(A(t), B(t))$



- Qual'è la Funzione di Uscita ?
- Qual'è la Funzione Stato Futuro ?

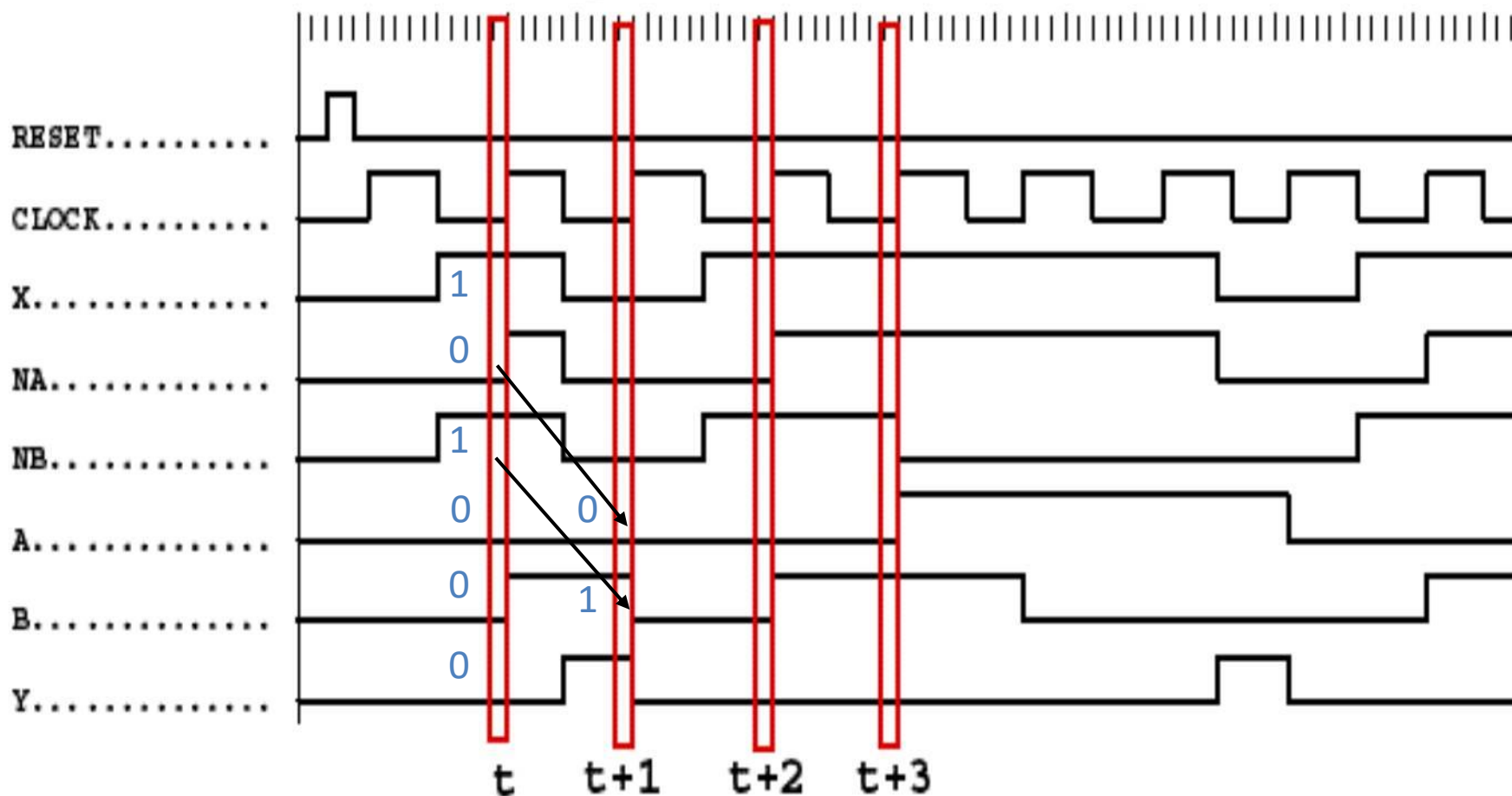
- $A(t+1) = D_A(t)$
- $B(t+1) = D_B(t)$

- $A(t+1) = D_A(t) = A(t)x(t) + B(t)x(t)$
- $B(t+1) = D_B(t) = \bar{A}(t)x(t)$
- $y(t) = \bar{x}(t)(B(t) + A(t))$



Analisi di Circuiti Sequenziali: Esempio

- Dove sono definiti nel tempo gli ingressi, le uscite e gli stati?



- **Tabella di stato** – è una tabella a variabili multiple costituita dalle seguenti quattro sezioni:
 - **Stato Presente** – I valori delle variabili di stato per ogni stato permesso.
 - **Ingresso** – le combinazioni di ingresso permesse.
 - **Stato Futuro** – il valore dello stato al tempo $(t+1)$ sulla base dello stato presente e dell'ingresso.
 - **Uscita** – il valore dell'uscita come funzione dello stato presente e (a volte) dell'ingresso.
- Dal punto di vista di una tabella di verità:
 - gli ingressi sono l'Ingresso e lo Stato Presente
 - le uscite sono l'Uscita e lo Stato Futuro

La Tabella di Stato: Esempio

- La tabella di stato si può riempire utilizzando le **equazioni dello stato futuro** e le **equazioni di uscita**:
- $A(t+1) = A(t)x(t) + B(t)x(t)$ $B(t+1) = \bar{A}(t)x(t)$
- $y(t) = \bar{x}(t)(B(t) + A(t))$

Stato Presente		Ingresso $x(t)$	Stato Futuro		Uscita $y(t)$
$A(t)$	$B(t)$		$A(t+1)$	$B(t+1)$	
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 di Stato: Esempio

- La tabella di stato si può riempire utilizzando le **equazioni dello stato futuro** e le **equazioni di uscita**:
- $A(t+1) = D_A(t) = A(t)x(t) + B(t)x(t)$ $B(t+1) = D_B(t) = A(t)x(t)$
- $y(t) = \bar{x}(t)(B(t) + A(t))$

Stato Presente		Ingresso	Ingressi ai FF		Stato Futuro		Uscita
A(t)	B(t)	x(t)	D _A (t)	D _B (t)	A(t+1)	B(t+1)	y(t)
0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0
0	1	0	0	0	0	0	1
0	1	1	1	1	1	1	0
1	0	0	0	0	0	0	1
1	0	1	1	0	1	0	0
1	1	0	0	0	0	0	1
1	1	1	1	0	1	0	0

La Tabella di Stato Bidimensionale: Esempio

- Tabella Bidimensionale che si adatta bene ad una K-map. Righe dello Stato Presente e colonne di Ingresso in ordine secondo il codice di Gray.
- $A(t+1) = A(t)x(t) + B(t)x(t)$
- $B(t+1) = \bar{A}(t)x(t)$
- $y(t) = \bar{x}(t)(B(t) + A(t))$

Stato Presente		Stato Futuro				Uscita	
		x(t)=0		x(t)=1		x(t)=0	x(t)=1
A(t)	B(t)	A(t+1)	B(t+1)	A(t+1)	B(t+1)	y(t)	y(t)
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

- Nel caso del flip-flop D, i valori dello stato futuro sono ottenuti direttamente dalle equazioni d'ingresso.
- **L'analisi con flip-flop JK invece richiede più passaggi per determinare lo stato futuro**

Procedura in due passi per determinare lo stato futuro

- 1) Ricavare da ciascuna equazione di ingresso la combinazione binaria dello stato presente e degli ingressi al flip-flop.
- 2) Utilizzare le tabelle caratteristiche del flip-flop corrispondente per determinare lo stato futuro.

Analisi di Circuiti Sequenziali con Flip Flop J-K

Es. Circuito con 2 F-F J – K, indicati con A e B, un ingresso X, e le seguenti equazioni di ingresso ai F-F:

$$J_A = B \quad K_A = B \cdot \bar{X} \quad J_B = \bar{X} \quad K_B = A \cdot \bar{X} + \bar{A} \cdot X$$



Tabella di stato

Stato Presente		Ingresso	Ingressi ai Flip Flop				Stato Futuro	
A	B	X	J _A	K _A	J _B	K _B	A	B
0	0	0	0	0	1	0	0	1
0	0	1	0	0	0	1	0	0
0	1	0	1	1	1	0	1	1
0	1	1	1	0	0	1	1	0
1	0	0	0	0	1	1	1	1
1	0	1	0	0	0	0	1	0
1	1	0	1	1	1	1	0	0
1	1	1	1	0	0	0	1	1

- I circuiti sequenziali o macchine sequenziali sono anche chiamati *Macchine a Stati Finiti* o *Finite State Machines* (FSM).
- Esistono due modelli formali:
 - Modello di Moore
 - dal nome di E.F. Moore
 - le uscite sono funzione SOLO degli stati
 - di solito specificate negli stati.
 - Modello di Mealy
 - dal nome di G. Mealy
 - le uscite sono funzione degli ingressi E degli stati
 - di solito specificate sugli archi di transizione di stato.

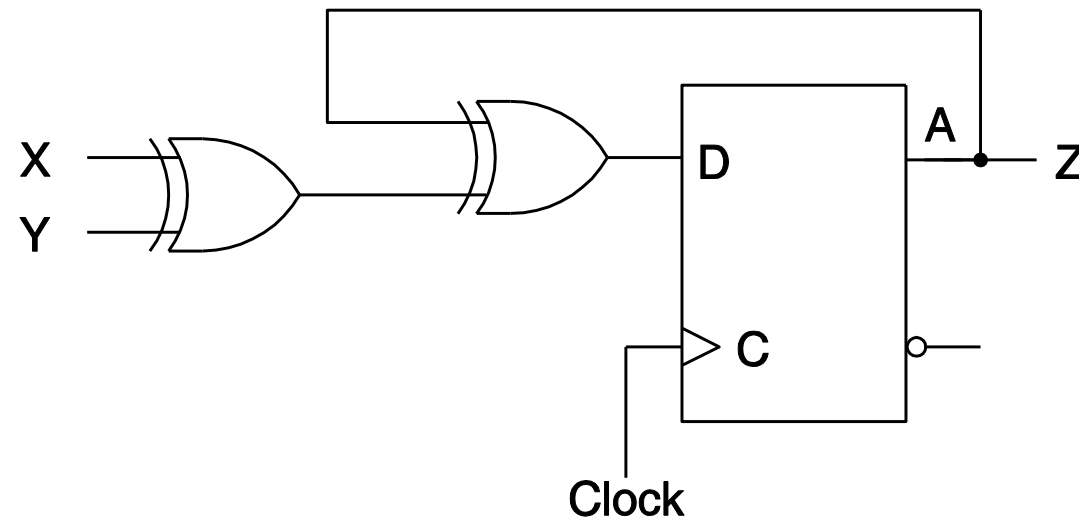
OSS. Un circuito di tipo Moore può essere sempre ricondotto ad uno di tipo Mealy nel quale le uscite del circuito sono direttamente connesse alle uscite dei flip-flop

I Modelli di Moore e di Mealy: Esempio

- Esempio: ricavare il diagramma logico e la tabella di stato di un circuito sequenziale specificato dalla seguente:
 - Equazione di ingresso $D_A = A \oplus X \oplus Y$
 - Equazione di uscita $Z = A$

■ Sol. :

Modello di Moore



Stato Presente	Ingressi		Stato Futuro	Uscita
A	X	Y	A	Z
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

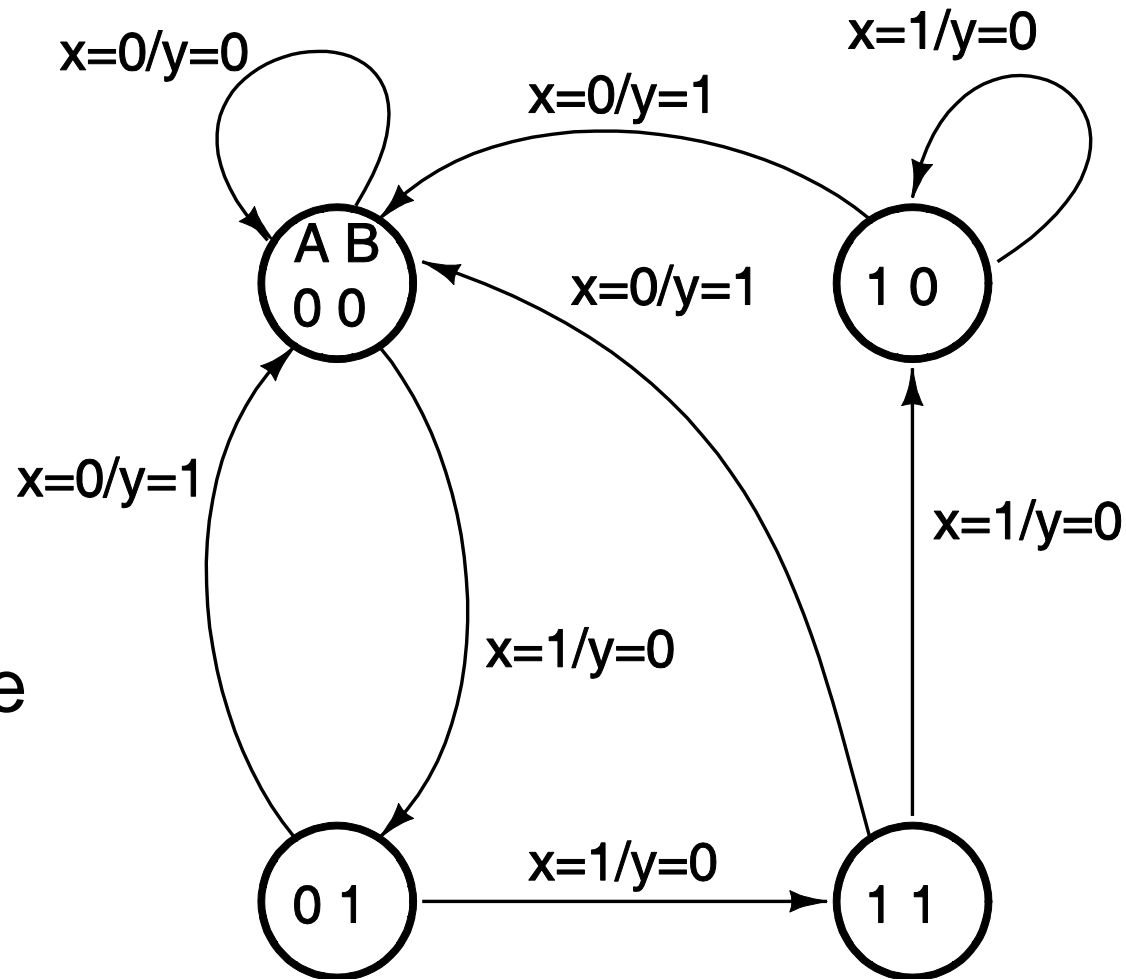
Il Diagramma di Stato - I

- La funzione di un circuito sequenziale si può rappresentare in forma grafica come un diagramma di stato con i seguenti componenti:
 - Un cerchio con il nome dello stato per ogni stato.
 - Un arco orientato dallo Stato Presente allo Stato Futuro per ogni transizione di stato.
 - Un'etichetta (label) su ogni arco orientato con i valori di Ingresso che causano le transizioni di stato.
 - Un'etichetta:
 - in ogni cerchio con il valore dell'Uscita prodotta, oppure
 - su ogni arco orientato con il valore dell'Uscita prodotta.

- **Formato dell'etichetta:**
 - Nel cerchio con inclusa l'uscita:
 - stato/uscita
 - l'uscita del tipo Moore dipende solo dallo stato
 - Sull'arco orientato con inclusa l'uscita:
 - ingresso/uscita
 - l'uscita del tipo Mealy dipende dallo stato e dall'ingresso

Diagrammi di Stato: Esempio 1

- Quale tipo?
- Il diagramma fa confusione per grandi circuiti
- Per piccoli circuiti, di solito è di più facile comprensione della tabella di stato



- I circuiti sequenziali o macchine sequenziali sono anche chiamati *Macchine a Stati Finiti* o *Finite State Machines* (FSM).
- Esistono due modelli formali:

1) Modello di Moore

- dal nome di E.F. Moore;
- le uscite sono funzione SOLO degli stati,
- di solito specificate negli stati.

2) Modello di Mealy

- dal nome di G. Mealy;
- le uscite sono funzione degli ingressi E degli stati,
- di solito specificate sugli archi di transizione di stato.

Tabelle di Stato (Moore e Mealy)

- La tabella di stato del Modello di Moore mappa lo stato sulle uscite

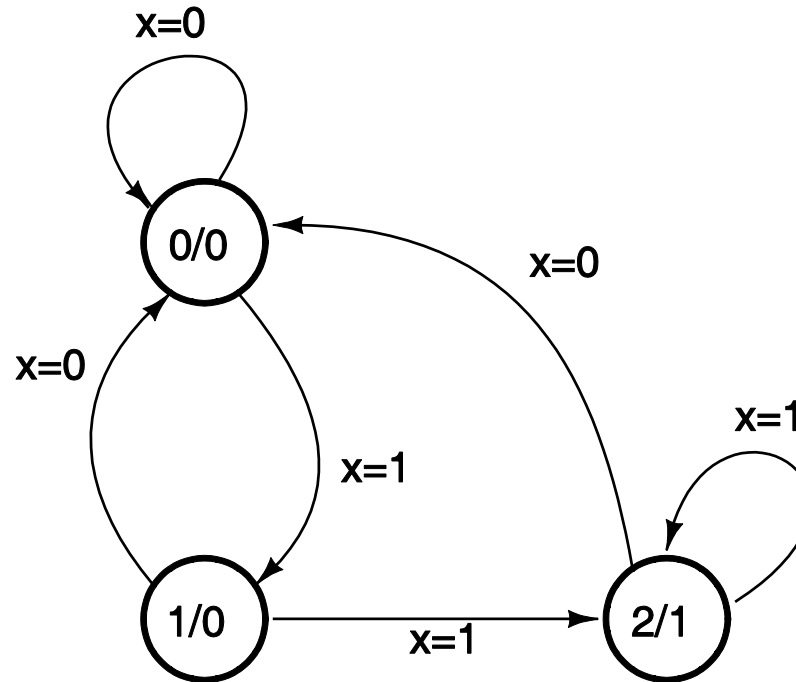
Stato Presente	Stato Futuro		Uscita
	$x(t)=0$	$x(t)=1$	
0	0	1	0
1	0	2	0
2	0	2	1

- La tabella di stato del Modello di Mealy mappa gli ingressi e lo stato sulle uscite

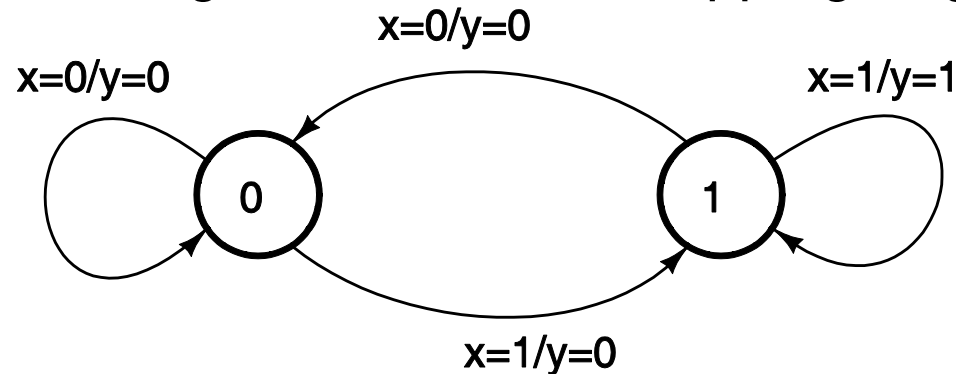
Stato Presente	Stato Futuro		Uscita	
	$x(t)=0$	$x(t)=1$	$x(t)=0$	$x(t)=1$
0	0	1	0	0
1	0	1	0	1

Diagrammi di Stato (Moore e di Mealy)

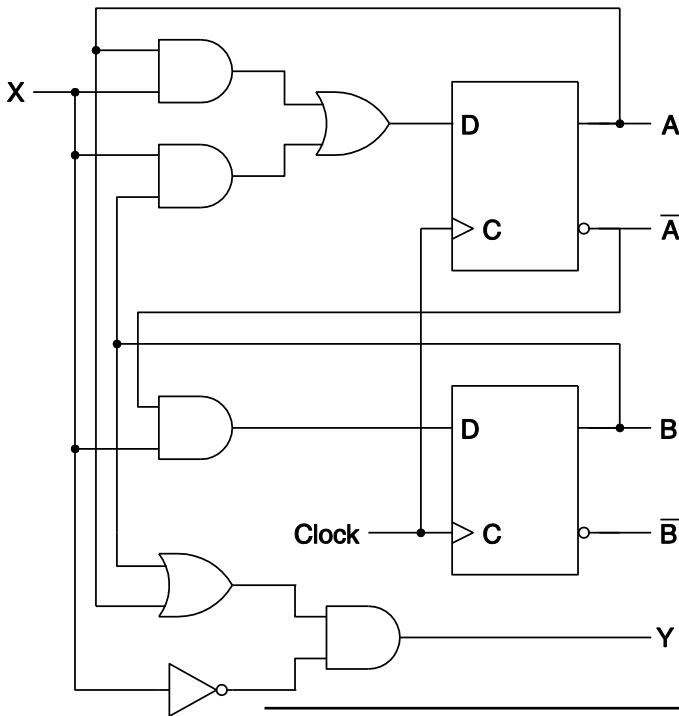
- Il modello di Moore del diagramma di stato mappa lo stato sulle uscite



- Il modello di Mealy del diagramma di stato mappa gli ingressi e lo stato sulle uscite

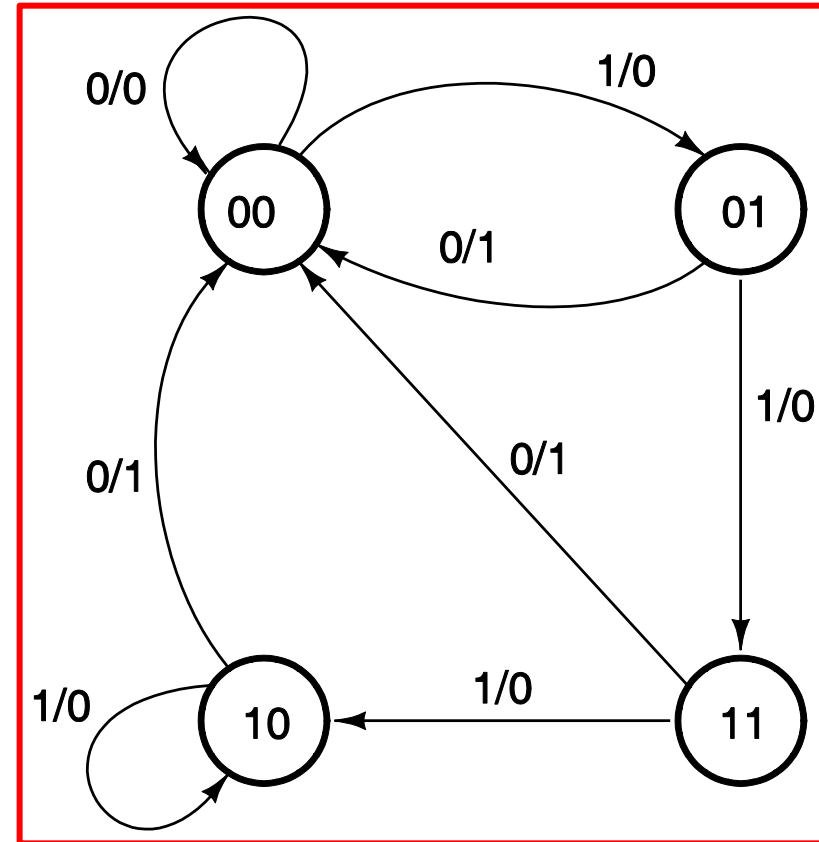


Diagrammi di Stato: Esempi Precedenti



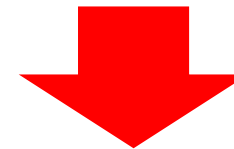
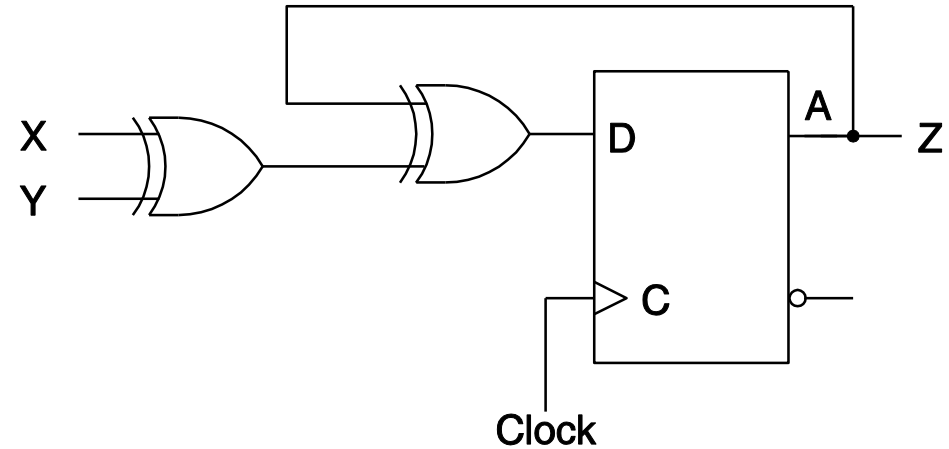
Stato Presente		Ingresso	Stato Futuro		Uscita
A	B	X	A	B	Y
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	0	1
0	1	1	1	1	0
1	0	0	0	0	1
1	0	1	1	0	0
1	1	0	0	0	1
1	1	1	1	0	0

Mealy

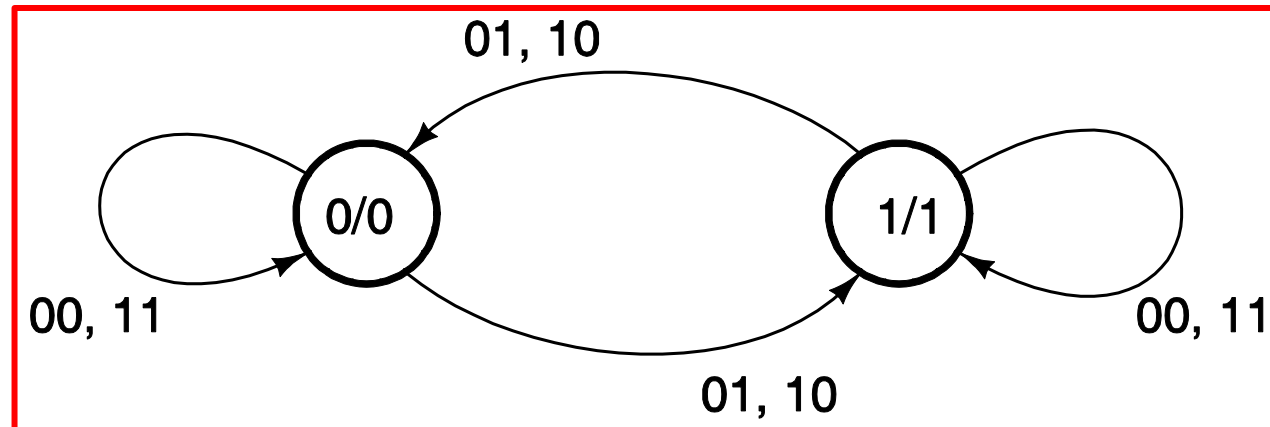


Diagrammi di Stato: Esempi Precedenti

Stato Presente	Ingressi		Stato Futuro	Uscita
A	X	Y	A	Z
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1



Moore

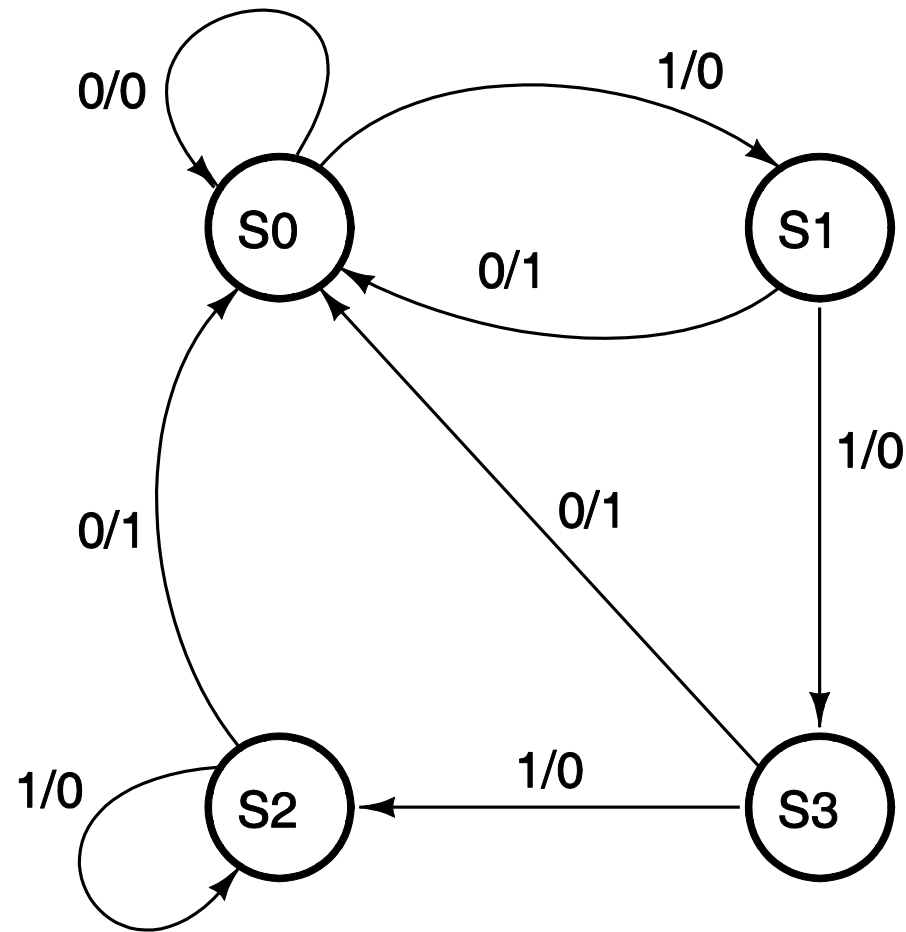


Definizione di Stati Indistinguibili (Equivalenti)

- Due stati sono *indistinguibili* o *equivalenti* se mostrano la stessa risposta per ogni possibile combinazione degli ingressi (se la loro risposta è una sequenza di uscita identica per ogni possibile sequenza di ingresso).
- Alternativamente, due stati sono *indistinguibili* o *equivalenti* se per ogni combinazione degli ingressi generano la stessa uscita e transitano negli stessi stati o in stati equivalenti (se le loro uscite prodotte per ogni simbolo di ingresso sono identiche e i loro stati futuri per ogni simbolo di ingresso sono gli stessi o sono equivalenti).

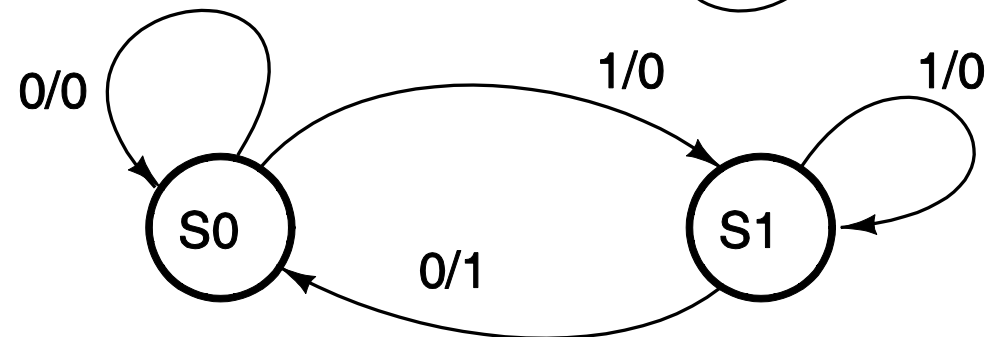
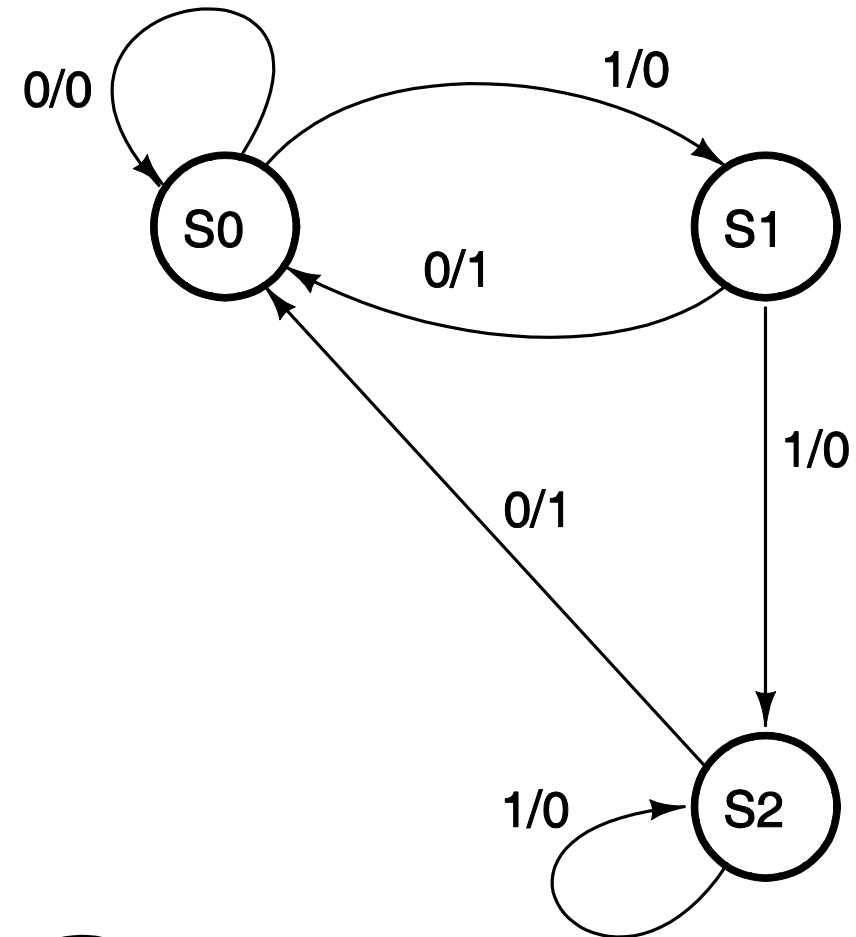
Esempio di Stato Equivalente

- Per gli stati S3 e S2,
 - L'uscita per l'ingresso 0 è 1 e per l'ingresso 1 è 0, e
 - Lo stato futuro per l'ingresso 0 è S0 e per l'ingresso 1 è S2.
 - Dalla definizione alternativa, gli stati S3 e S2 sono equivalenti.



Esempio di Stato Equivalente

- Sostituendo S3 e S2 con un singolo stato produce il diagramma di stato:
- Esaminando il nuovo diagramma, gli stati S1 e S2 sono equivalenti poichè
 - le loro uscite per l'ingresso 0 è 1 e per l'ingresso 1 è 0, e
 - il loro stato futuro per l'ingresso 0 è S0 e per l'ingresso 1 è S2,
- Sostituendo S1 e S2 con un singolo stato produce il diagramma di stato:



Procedura di Progetto - I

1. Specifica - Si determinino le specifiche del progetto.
2. Sintesi - Ricavare il diagramma di stato o la tabella di stato dalla descrizione del problema. Se è disponibile solo il diagramma di stato, si ricavi la tabella di stato.
3. Assegnazione dello stato - Si assegnino i codici binari agli stati.
4. Determinazione delle equazioni di ingresso ai F-F - Si scelga il tipo o i tipi di F-F da impiegare e si derivino le equazioni di ingresso ai F-F dalle colonne relative allo stato futuro della tabella di stato codificata.

OSS. n Flip-Flop possono rappresentare 2^n stati binari

Procedura di Progetto - II

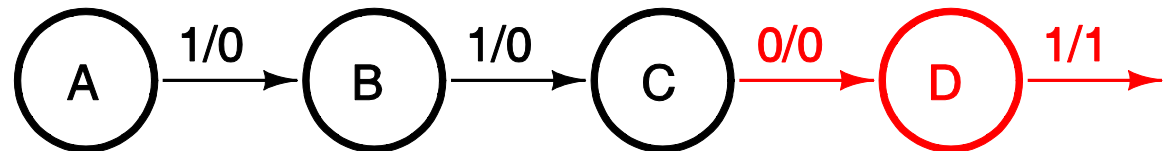
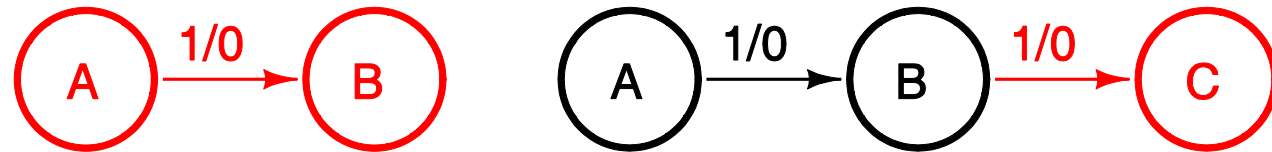
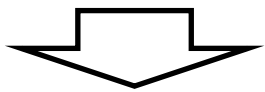
5. Determinazione delle equazioni di uscita - Si derivino le equazioni di uscita dalle colonne relative alle uscite della tabella di stato.
6. Ottimizzazione - Si semplifichino le equazioni di ingresso ai F-F e le equazioni di uscita.
7. Implementazione tecnologica - Si disegni il diagramma logico utilizzando F-F e porte combinatorie così come specificato dalle equazioni di ingresso ai F-F e dalle equazioni di uscita. Si modifichi il diagramma in base alla tecnologia disponibile.
8. Verifica - Si verifichi la correttezza del circuito finale ottenuto.

Es. 1 Progettare un riconoscitore di sequenze (1101)

- Costruzione del diagramma di stato (Mealy o Moore?)

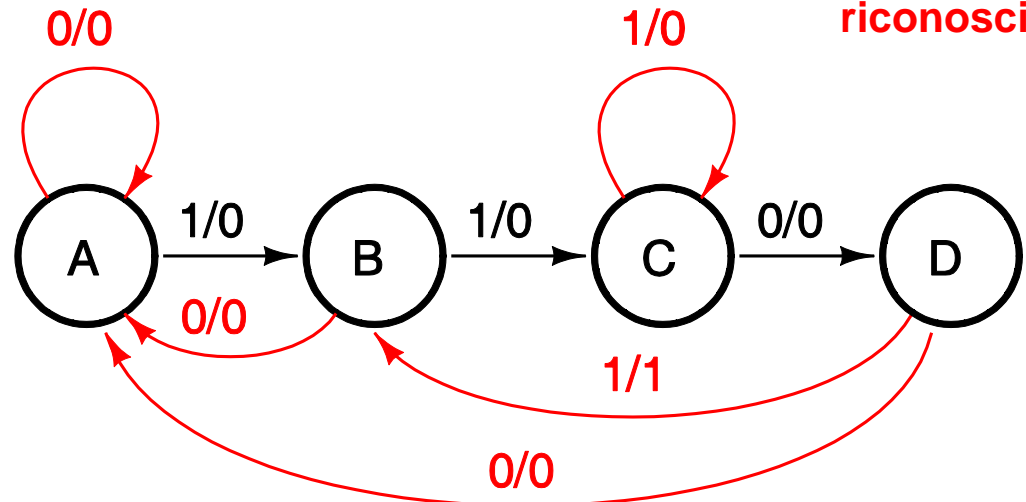
1101 \rightarrow 1

X \rightarrow Z



sequenza
riconosciuta!

diagramma di stato
(modello di Mealy)



- **Costruzione della tabella di stato**

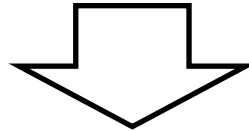


tabella di stato

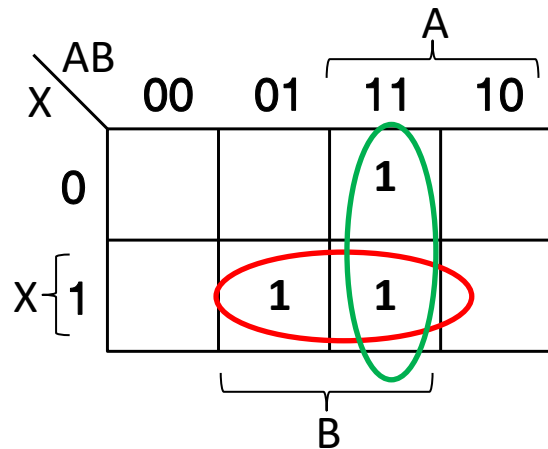
Stato Presente	Stato Futuro		Uscita Z	
	X = 0	X = 1	X = 0	X = 1
A	A	B	0	0
B	A	C	0	0
C	D	C	0	0
D	A	B	0	1

- **Codifica binaria degli stati**

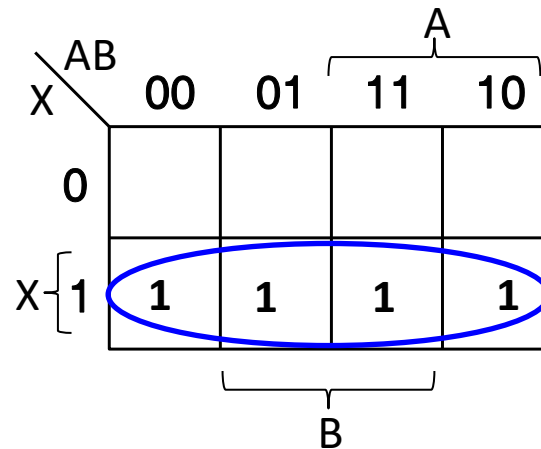
tabella di stato rappresentando gli stati con un codice di Gray a 2 bit

Stato Presente		Stato Futuro		Uscita Z	
AB		X = 0	X = 1	X = 0	X = 1
00		00	01	0	0
01		00	11	0	0
11		10	11	0	0
10		00	01	0	1

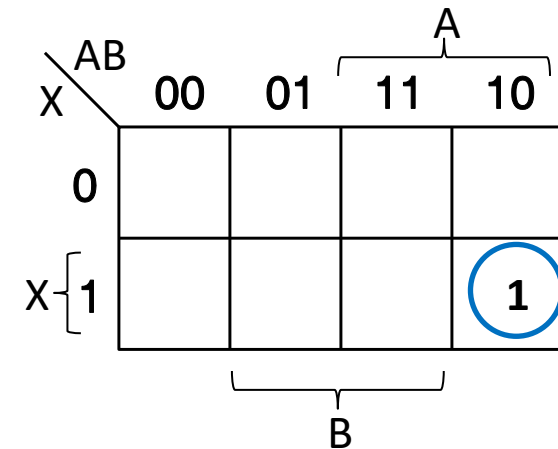
- Determinazione (e semplificazione) delle equazioni di ingresso ai flip-flop e delle equazioni di uscita**



$$D_A = A \cdot B + B \cdot X$$

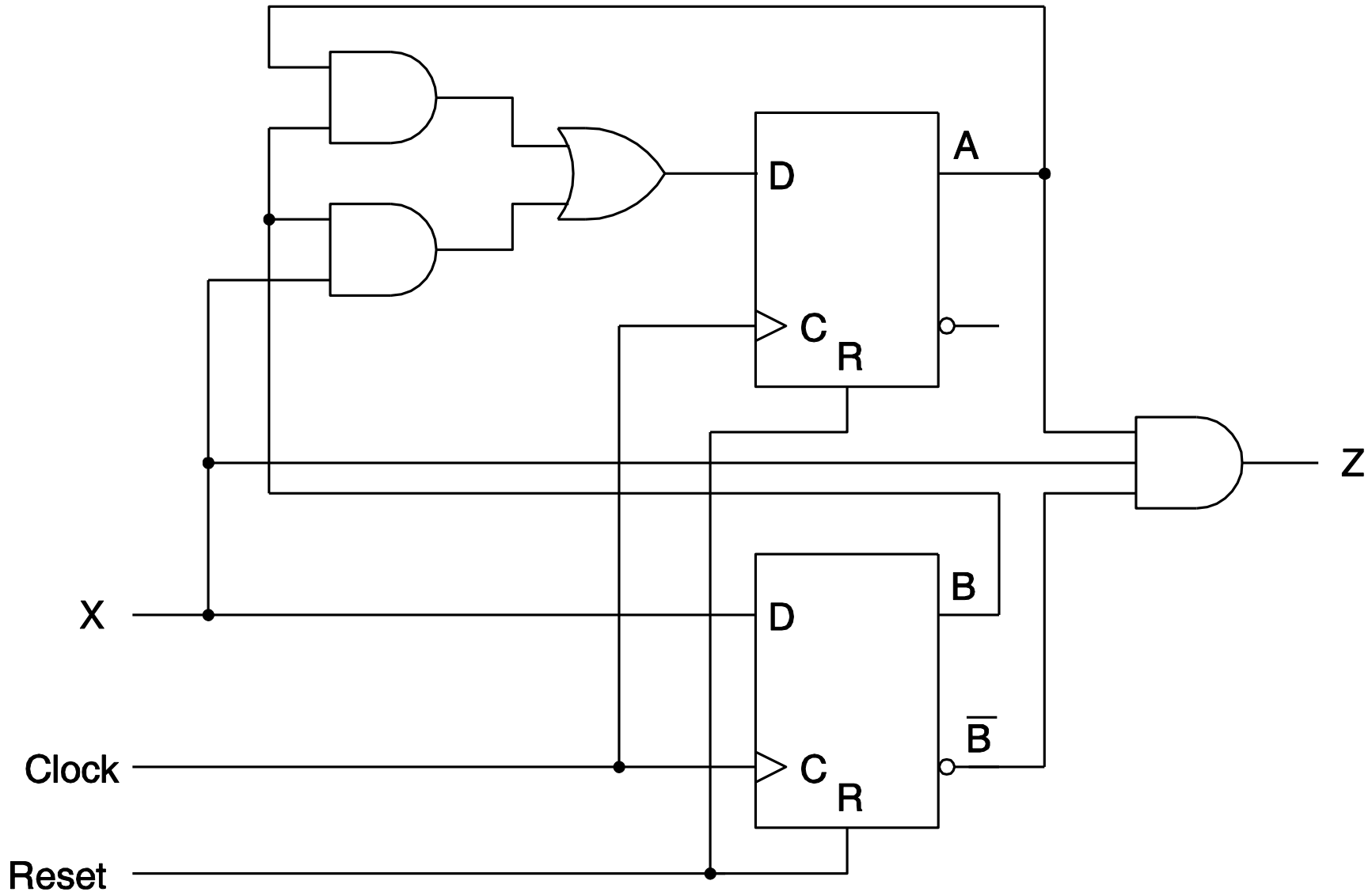


$$D_B = X$$



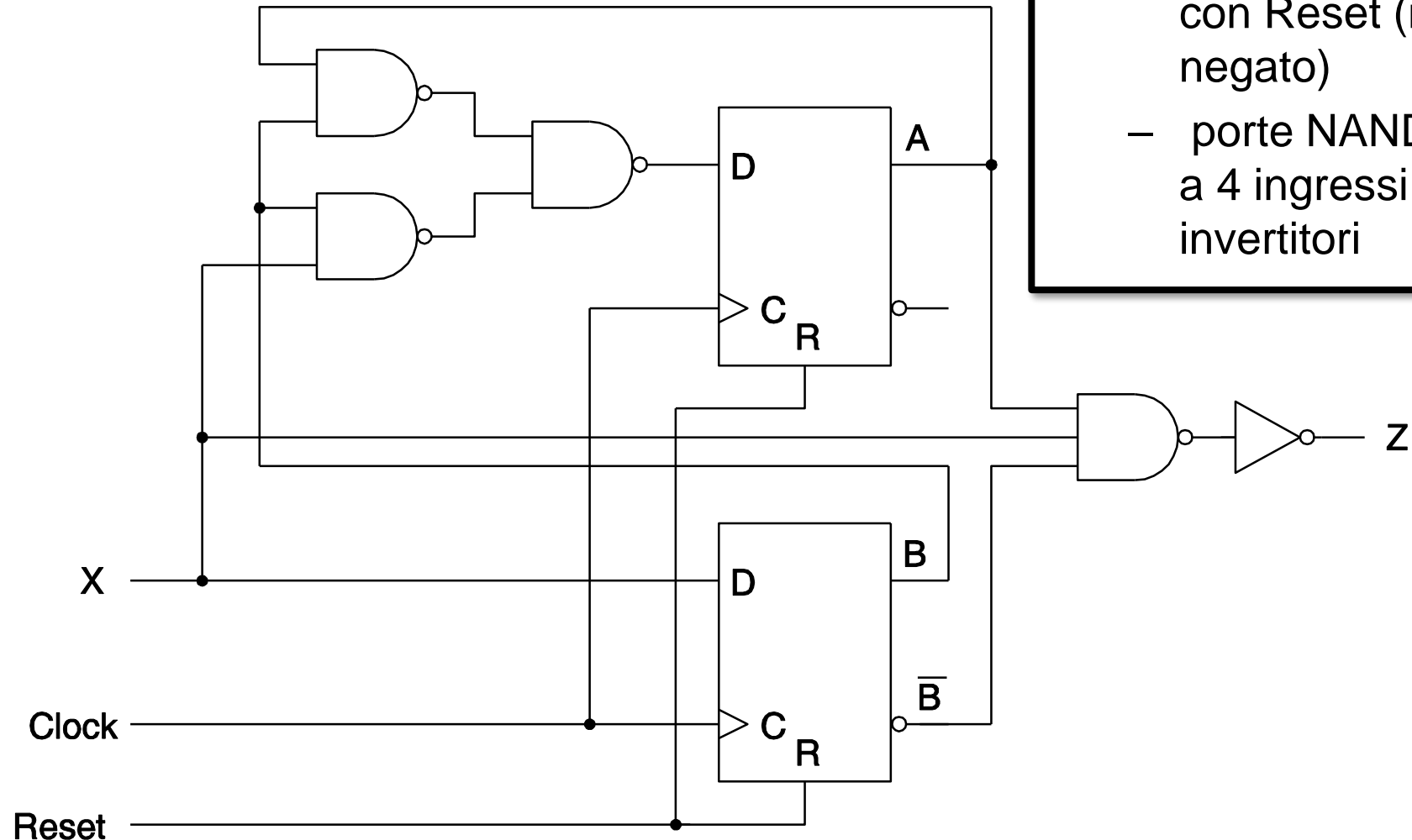
$$Z = A \cdot \bar{B} \cdot X$$

- Diagramma Logico



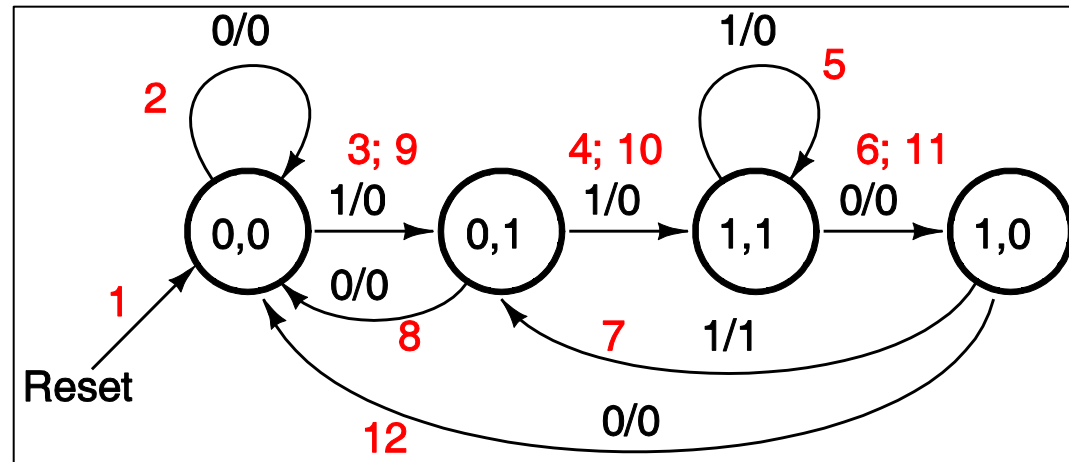
- Implementazione tecnologica

- Libreria:
 - Flip-flop di tipo D con Reset (non negato)
 - porte NAND fino a 4 ingressi e invertitori

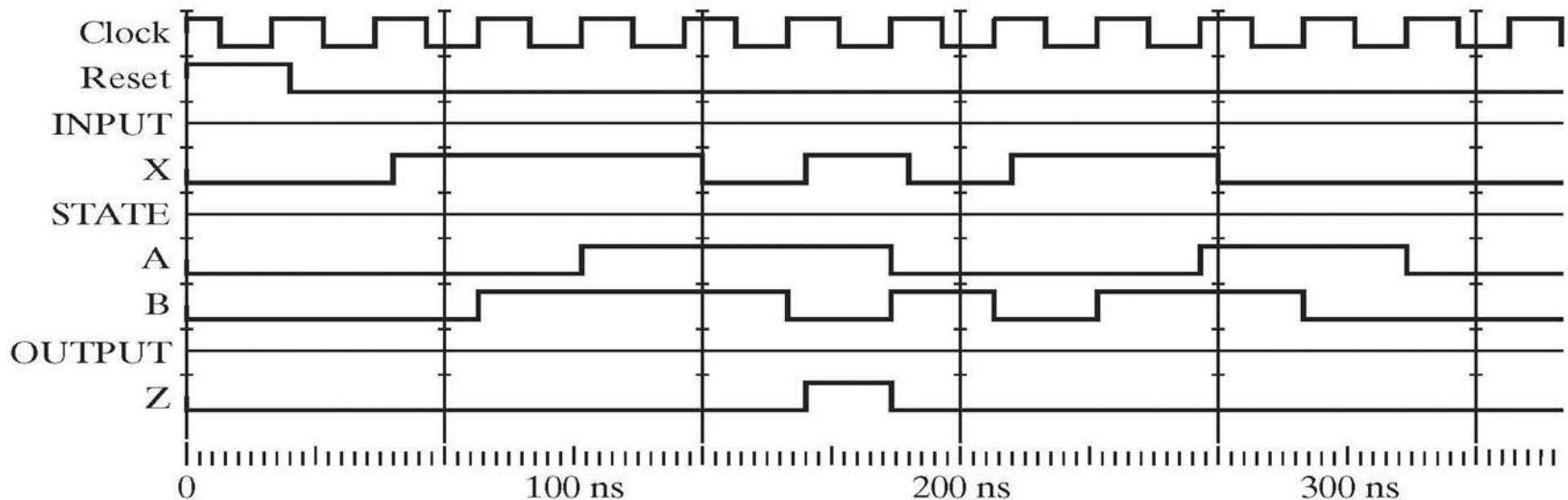


Progetto di Circuiti Sequenziali

• Verifica



Clock Edge:	0	1	2	3	4	5	6	7	8	9	10	11	12	13
Input R:	X	1	0	0	0	0	0	0	0	0	0	0	0	
Input X:	X	0	0	1	1	1	0	1	0	1	1	0	0	
State (A,B):	X,X	0,0*	0,0	0,0	0,1	1,1	1,1	1,0	0,1	0,0	0,1	1,1	1,0	0,0
Output Z:	X	0	0	0	0	0	0	1	0	0	0	0	0	



- **Codifica binaria degli stati (versione alternativa)**

**tabella di stato rappresentando gli stati con un codice
1-out-of-4 a 4 bit**

Stato Presente		Stato Futuro		Uscita Z	
ABCD		X = 0	X = 1	X = 0	X = 1
1000		1000	0100	0	0
0100		1000	0010	0	0
0010		0001	0010	0	0
0001		1000	0100	0	1

Progetto di Circuiti Sequenziali

- Diagramma Logico
(versione alternativa)**

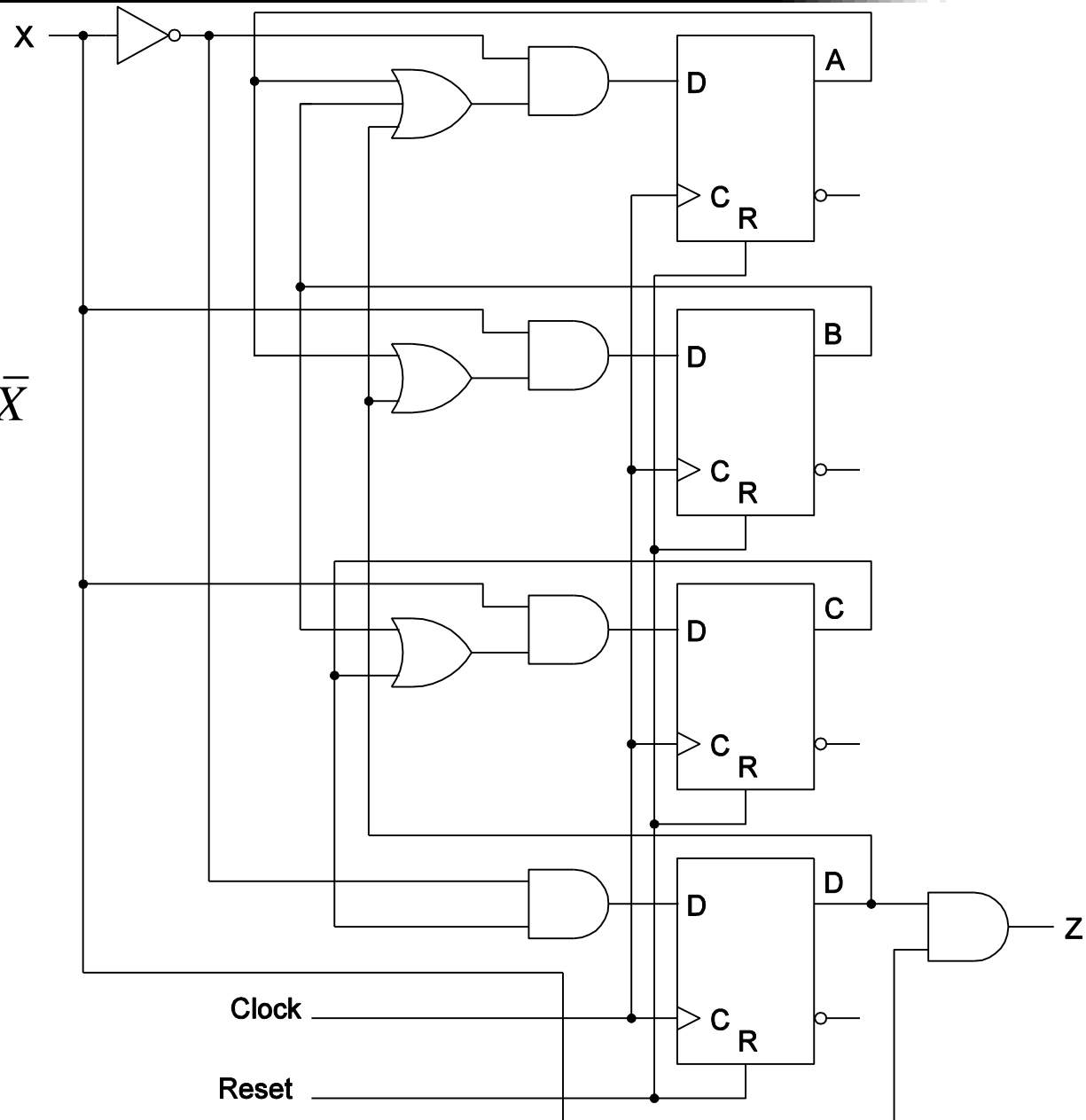
$$A(t+1) = D_A = (A + B + D) \bar{X}$$

$$B(t+1) = D_B = (A + D) X$$

$$C(t+1) = D_C = (B + C) X$$

$$D(t+1) = D_D = C \bar{X}$$

$$Z = DX$$



Es. 2. Costruire il diagramma di stato per un decodificatore BCD – Eccesso-3

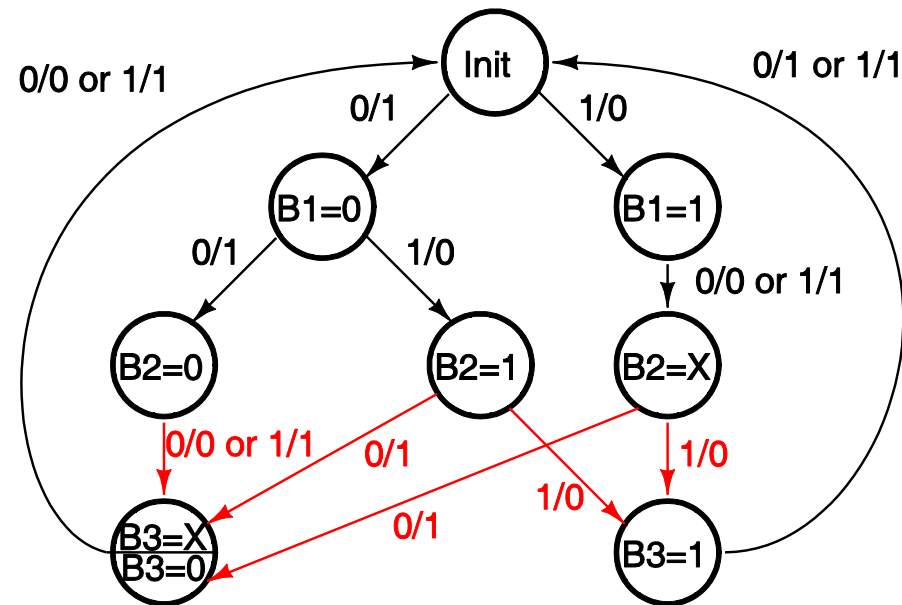
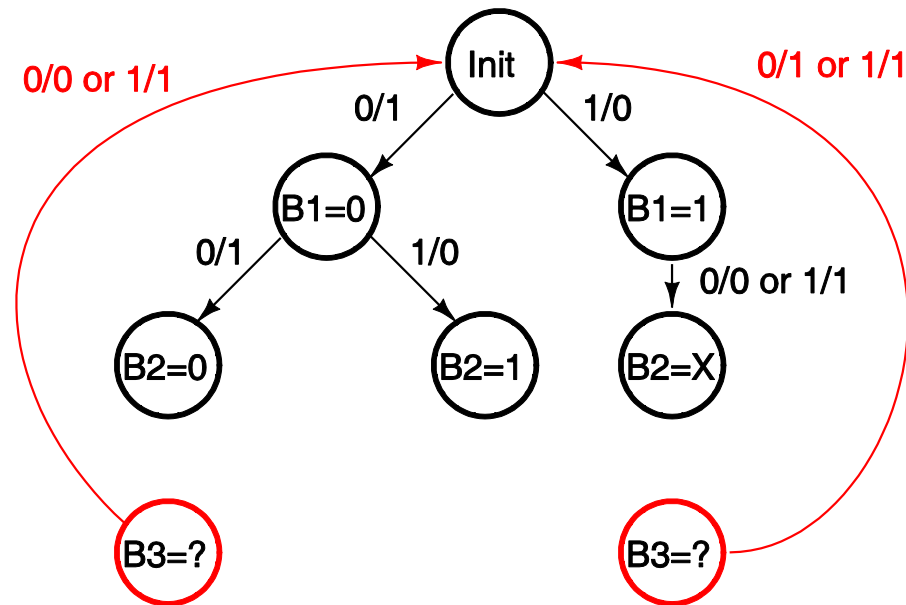
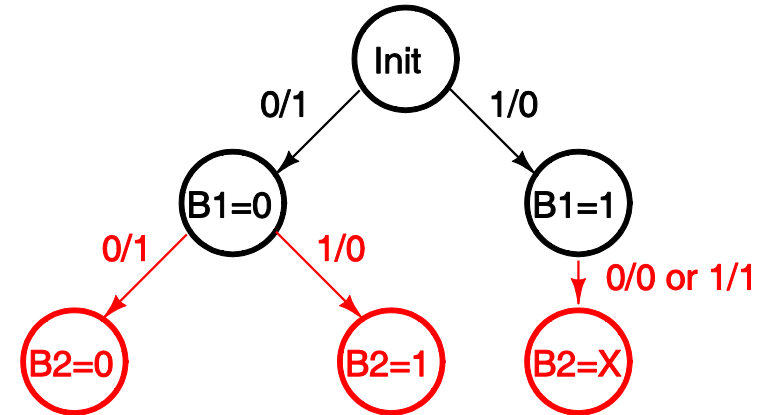
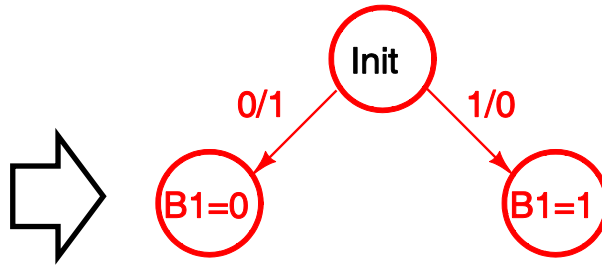
Tabella delle Sequenze

Sequenze in ordine di presentazione delle cifre (1 -> LSB)								Sequenze in ordine di prefisso comune (1 -> MSB)							
Ingresso BCD				Uscita Eccesso-3				Ingresso BCD				Uscita Eccesso-3			
1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
0	0	0	0	1	1	0	0	0	0	0	0	1	1	0	0
1	0	0	0	0	0	1	0	0	0	0	1	1	1	0	1
0	1	0	0	1	0	1	0	0	0	1	0	1	1	1	0
1	1	0	0	0	1	1	0	0	1	0	0	1	0	1	0
0	0	1	0	1	1	1	0	0	1	1	0	1	0	0	1
1	0	1	0	0	0	0	1	1	0	0	0	0	0	1	0
0	1	1	0	1	0	0	1	1	0	0	1	0	0	1	1
1	1	1	0	0	1	0	1	1	0	1	0	0	0	0	1
0	0	0	1	1	1	0	1	1	1	0	0	0	1	1	0
1	0	0	1	0	0	1	1	1	1	1	0	0	1	0	1

Progetto di Circuiti Sequenziali

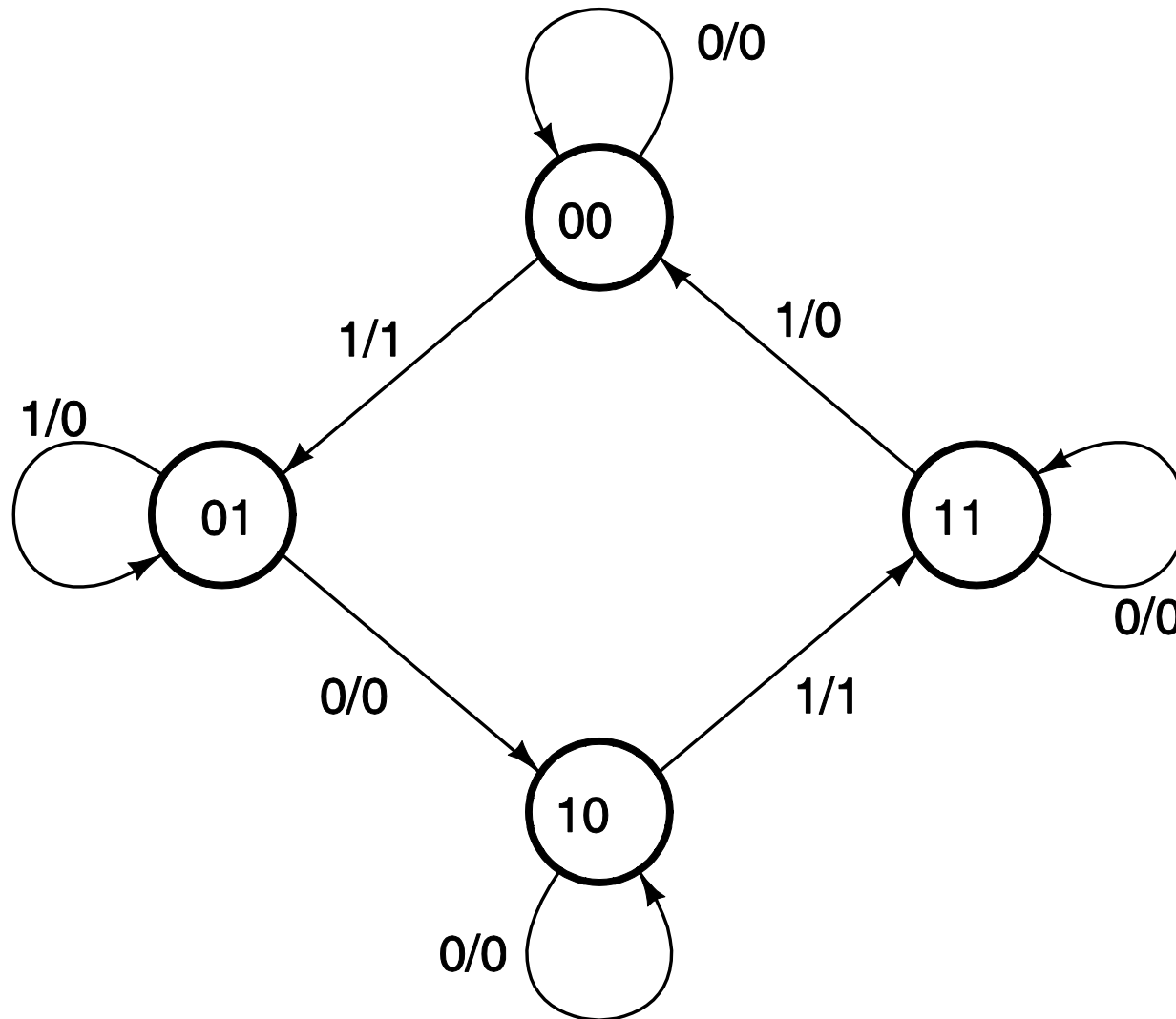
Es. 2. Costruire il diagramma di stato per un decodificatore BCD – Eccesso-3

diagramma
di stato



Progettare con Flip-Flop di Tipo D

Es. 3. Progettare un circuito sequenziale sincrono il cui funzionamento è descritto dal seguente diagramma di stato



Progettare con Flip-Flop di Tipo D

Tabella di Stato

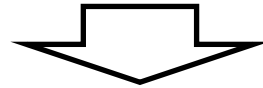
Stato Presente		Ingresso	Stato Futuro		Uscita
A	B	X	A	B	Y
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	1	1	0
1	1	1	0	0	0

Progettare con Flip-Flop di Tipo D

$$A(t+1) = D_A(A, B, X) = \sum_{ABX} m(2, 4, 5, 6)$$

$$B(t+1) = D_B(A, B, X) = \sum_{ABX} m(1, 3, 5, 6)$$

$$Y(A, B, X) = \sum_{ABX} m(1, 5)$$



AB		A			
		00	01	11	10
X	0		1	1	1
	1				1
		B			

AB		A			
		00	01	11	10
X	0			1	
	1	1	1		1
		B			

AB		A			
		00	01	11	10
X	0				
	1	1			1
		B			

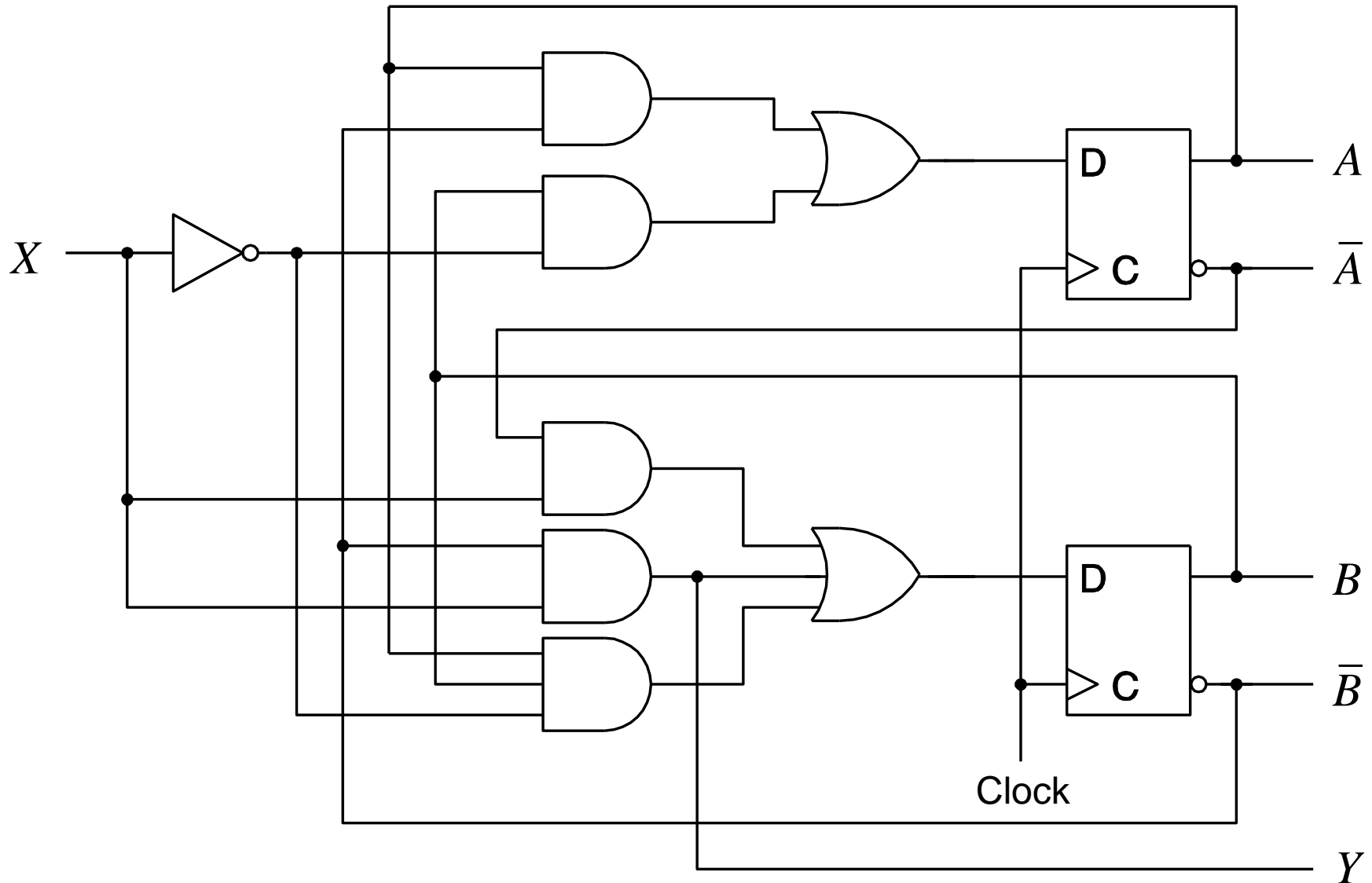
$$D_A = A \cdot \bar{B} + B \cdot \bar{X}$$

$$D_B = \bar{A} \cdot X + \bar{B} \cdot X + A \cdot B \cdot \bar{X}$$

$$Y = \bar{B} \cdot X$$

Progettare con Flip-Flop di Tipo D

Diagramma Logico



Progettare con Stati Non Utilizzati

Stato Presente			Ingresso	Stato Futuro		
A	B	C		A	B	C
0	0	1	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	0	1	1
0	1	0	1	1	0	0
0	1	1	0	0	0	1
0	1	1	1	1	0	0
1	0	0	0	1	0	1
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	1	0	0

3 variabili di stato ma solo 5 stati utilizzati!

Progettare con Stati Non Utilizzati

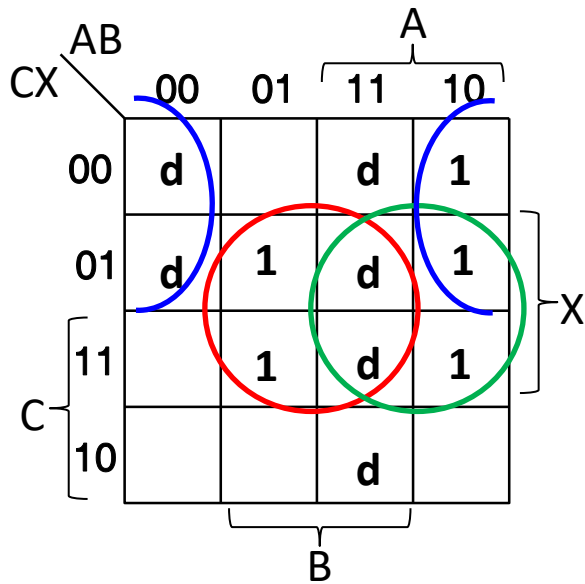
Stato Presente			Ingresso	Stato Futuro		
A	B	C	X	A	B	C
0	0	0	0	d	d	d
0	0	0	1	d	d	d
0	0	1	0	0	0	1
0	0	1	1	0	1	0
0	1	0	0	0	1	1
0	1	0	1	1	0	0
0	1	1	0	0	0	1
0	1	1	1	1	0	0
1	0	0	0	1	0	1
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	1	0	0
1	1	0	0	d	d	d
1	1	0	1	d	d	d
1	1	1	0	d	d	d
1	1	1	1	d	d	d

stati non utilizzati

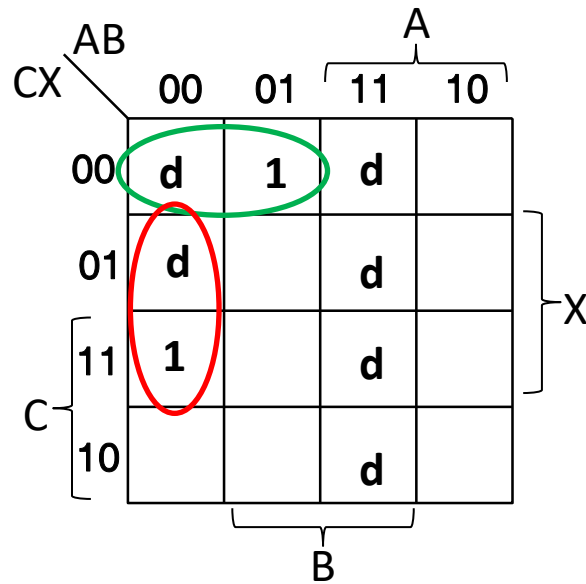
stati non utilizzati

stati non utilizzati

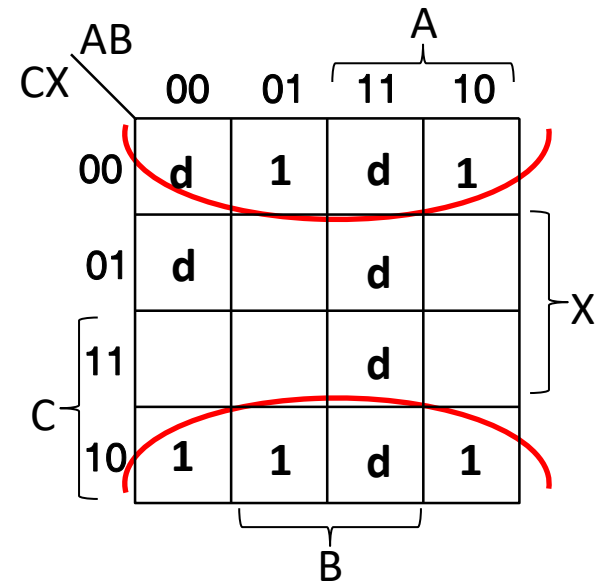
Progettare con Stati Non Utilizzati



D_A



D_B

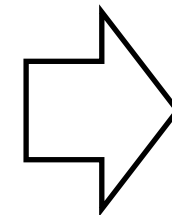


D_C

$$A(t+1) = D_A(A, B, C, X) = A \cdot X + B \cdot X + \bar{B} \cdot \bar{C}$$

$$B(t+1) = D_B(A, B, C, X) = \bar{A} \cdot \bar{C} \cdot \bar{X} + \bar{A} \cdot \bar{B} \cdot X$$

$$C(t+1) = D_C(A, B, C, X) = \bar{X}$$



**Diagramma
Logico**

Progettare con Flip-Flop di Tipo Generico

Tabelle di Eccitazione dei Flip Flop

Flip-Flop J-K

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

Flip-Flop D

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

Flip-Flop S-R

Q(t)	Q(t+1)	S	R
0	0	0	d
0	1	1	0
1	0	0	1
1	1	d	0

Flip-Flop T

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

Progettare con Flip-Flop di Tipo Generico

Flip Flop: Tabella Riassuntiva

Tipo	Tabella Caratteristica				Equazione Caratteristica	Tabella di Eccitazione				
D	D		Q(t+1)	Operazione	$Q(t+1) = D(t)$	Q(t+1)		D		Operazione
	0		0	Reset		0		0		Reset
	1		1	Set		1		1		Set
SR	S	R	Q(t+1)	Operazione	$Q(t+1) = S(t) + \bar{R}(t) \cdot Q(t)$	Q(t)	Q(t+1)	S	R	Operazione
	0	0	Q(t)	No Variazione		0	0	0	d	No Variazione
	0	1	0	Reset		0	1	1	0	Set
	1	0	1	Set		1	0	0	1	Reset
	1	1	?	Indefinito		1	1	d	0	No Variazione
JK	J	K	Q(t+1)	Operazione	$Q(t+1) = J(t) \cdot \bar{Q}(t) + \bar{K}(t) \cdot Q(t)$	Q(t)	Q(t+1)	J	K	Operazione
	0	0	Q(t)	No Variazione		0	0	0	d	No Variazione
	0	1	0	Reset		0	1	1	d	Set
	1	0	1	Set		1	0	d	1	Reset
	1	1	$\bar{Q}(t)$	Complemento		1	1	d	0	No Variazione
T	T		Q(t+1)	Operazione	$Q(t+1) = T(t) \oplus Q(t)$	Q(t+1)		T		Operazione
	0		Q(t)	No Variazione		Q(t)		0		No Variazione
	1		$\bar{Q}(t)$	Complemento		$\bar{Q}(t)$		1		Complemento

Progettare con Flip-Flop di Tipo J – K

Esempio - si progetti con flip flop J-K il circuito sequenziale le cui specifiche sono riportate sottoforma della seguente tabella di stato

Stato Presente		Ingresso	Stato Futuro		Uscita
A	B		A	B	
0	0	0	0	0	0
0	0	1	0	1	1
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	1	0	0
1	0	1	1	1	1
1	1	0	1	1	0
1	1	1	0	0	0

Progettare con Flip-Flop di Tipo J – K

Tabella di stato contenente
gli ingressi per i Flip-Flop J-K

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

Stato Presente		Ingresso	Stato Futuro		Ingressi Flip Flop				Uscita
A	B	X	A	B	J _A	K _A	J _B	K _B	Y
0	0	0	0	0	0	d	0	d	0
0	0	1	0	1	0	d	1	d	1
0	1	0	1	0	1	d	d	1	0
0	1	1	0	1	0	d	d	0	0
1	0	0	1	0	d	0	0	d	0
1	0	1	1	1	d	0	1	d	1
1	1	0	1	1	d	0	d	0	0
1	1	1	0	0	d	1	d	1	0

Progettare con Flip-Flop di Tipo J – K

Equazioni di Ingresso ai Flip Flop

X \ AB		A			
		00	01	11	10
X	0		1	d	d
	1			d	d

B

$$J_A = B \cdot \bar{X}$$

X \ AB		A			
		00	01	11	10
X	0	d	d		
	1	d	d	1	

B

$$K_A = B \cdot X$$

X \ AB		A			
		00	01	11	10
X	0		d	d	
	1	1	d	d	1

B

$$J_B = X$$

X \ AB		A			
		00	01	11	10
X	0	d	1		d
	1	d		1	d

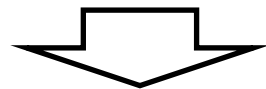
B

$$K_B = A \cdot X + \bar{A} \cdot \bar{X} = \overline{A \oplus X}$$

Progettare con Flip-Flop di Tipo J – K

Equazioni di Uscita

$$Y(A, B, X) = \sum_{ABX} m(1, 5)$$



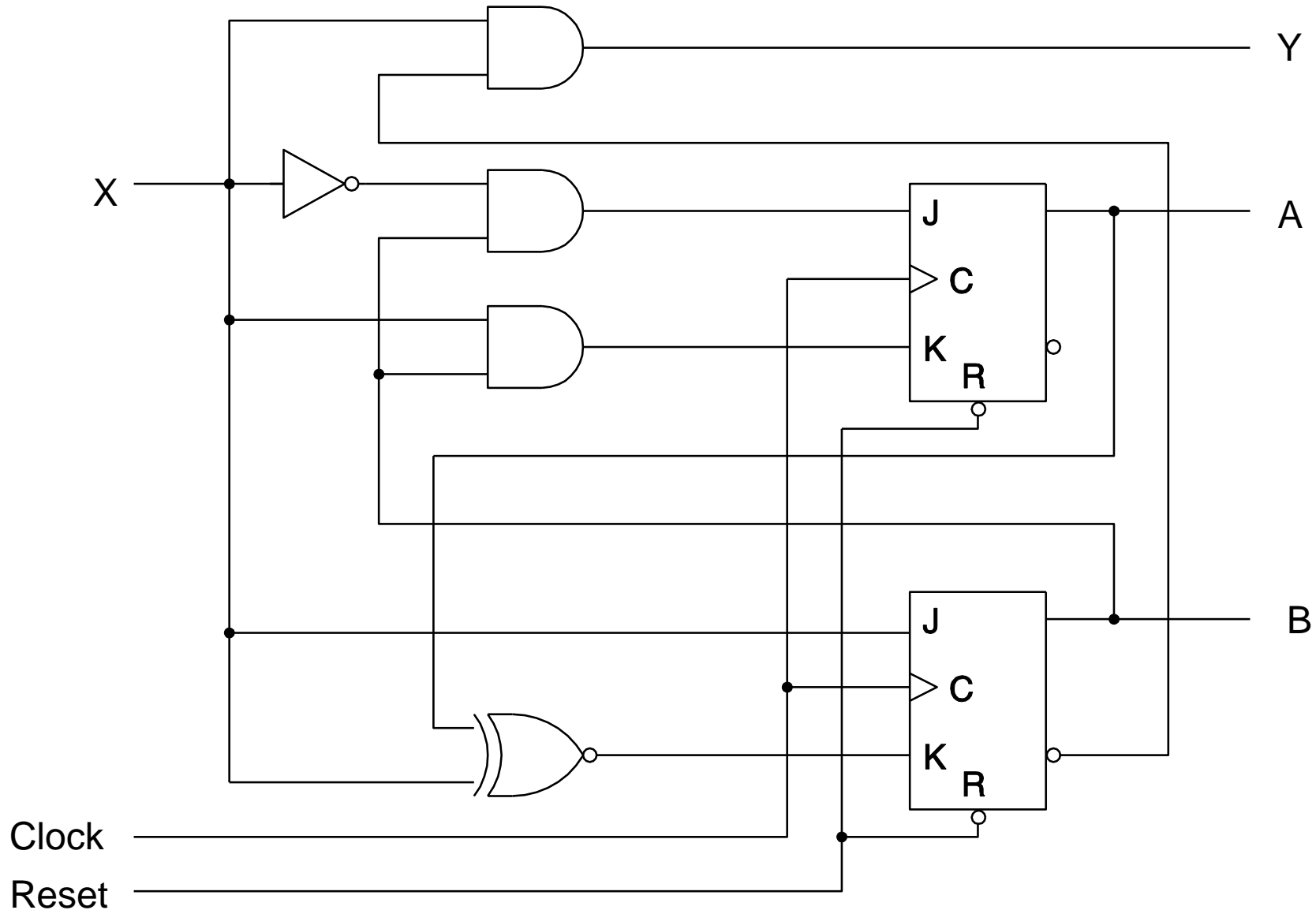
AB \ X		A			
		00	01	11	10
X	0				
	1	1			1

B

$$Y = \bar{B} \cdot X$$

Progettare con Flip-Flop di Tipo J – K

Diagramma Logico



Progettare con Flip-Flop di Tipo J – K

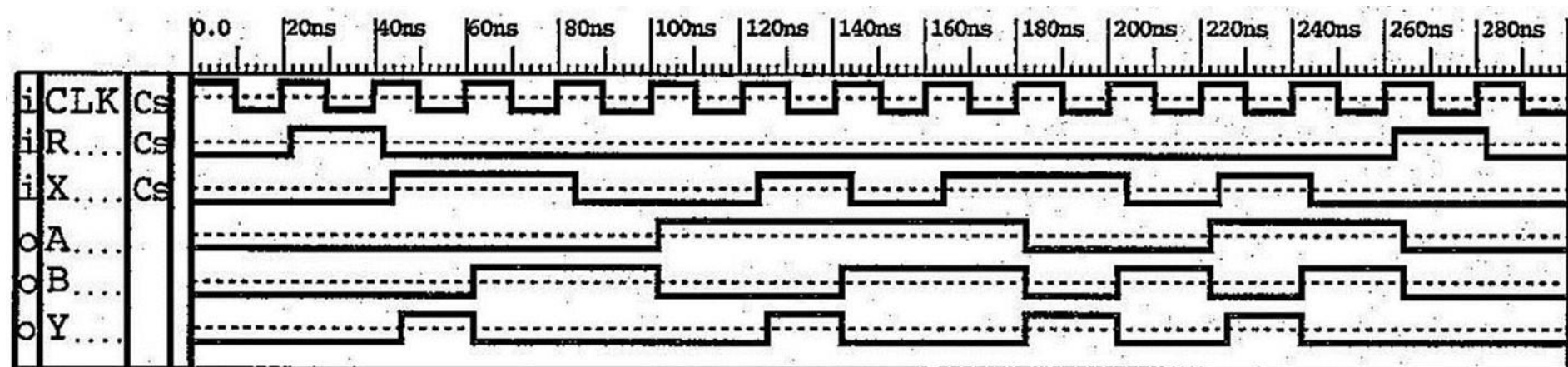
Verifica Funzionale Tramite Simulazione

Test circuitale e risultati attesi

R:	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0
X:	0	0	0	1	1	0	0	1	0	1	1	0	1	0	0	0
A:	X	X	0*	0	0	0	1	1	1	1	0	0	1	1	0*	0
B:	X	X	0*	0	1	1	0	0	1	1	0	1	0	1	0*	0
Y:	0	0	0*	1	0	0	0	1	0	0	1	0	1	0	0*	0

* Queste risposte sono asincrone e pertanto non aspettano il prossimo impulso di clock

Simulazione



Sono circuiti sequenziali in grado di gestire i dati immagazzinandoli.

Sono formati da una serie di flip flop e da un insieme di porte logiche (circuito combinatorio) che implementano opportune transizioni di stato.

- *I flip flop immagazzinano i dati.*
- *Il circuito combinatorio determina i dati che devono essere trasferiti nei flip flop*

Registri a Caricamento Parallelo - I

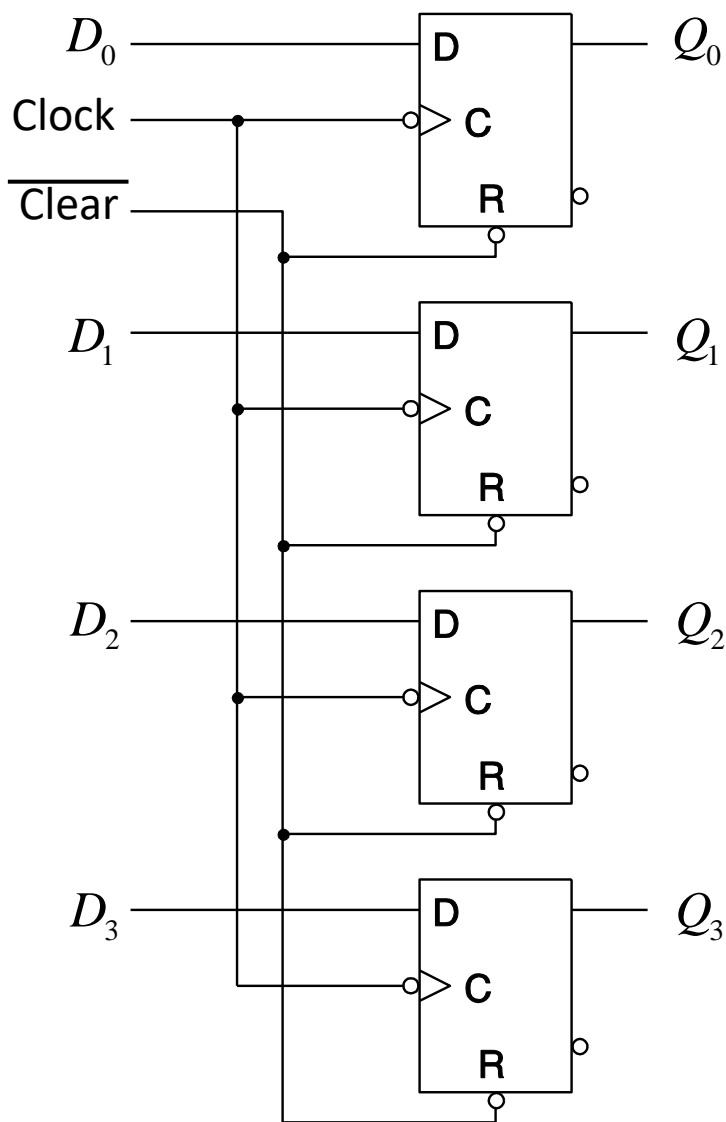
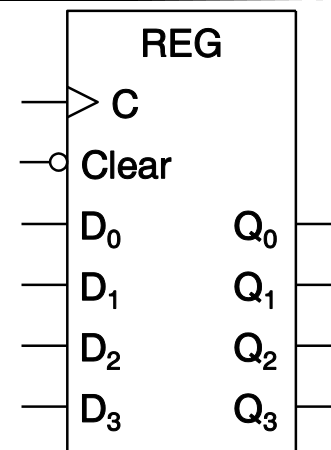


Diagramma logico

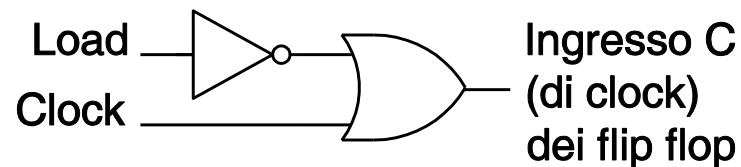
Registro a caricamento parallelo a 4 bit



Simbolo

Controllo degli ingressi

Necessita del
clock gating



$C = \overline{\text{Load}} + \text{Clock}$ Ingresso controllo di caricamento

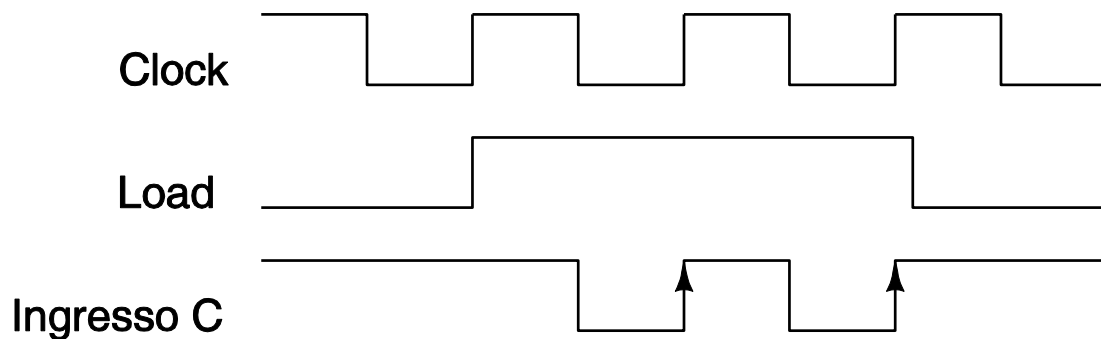


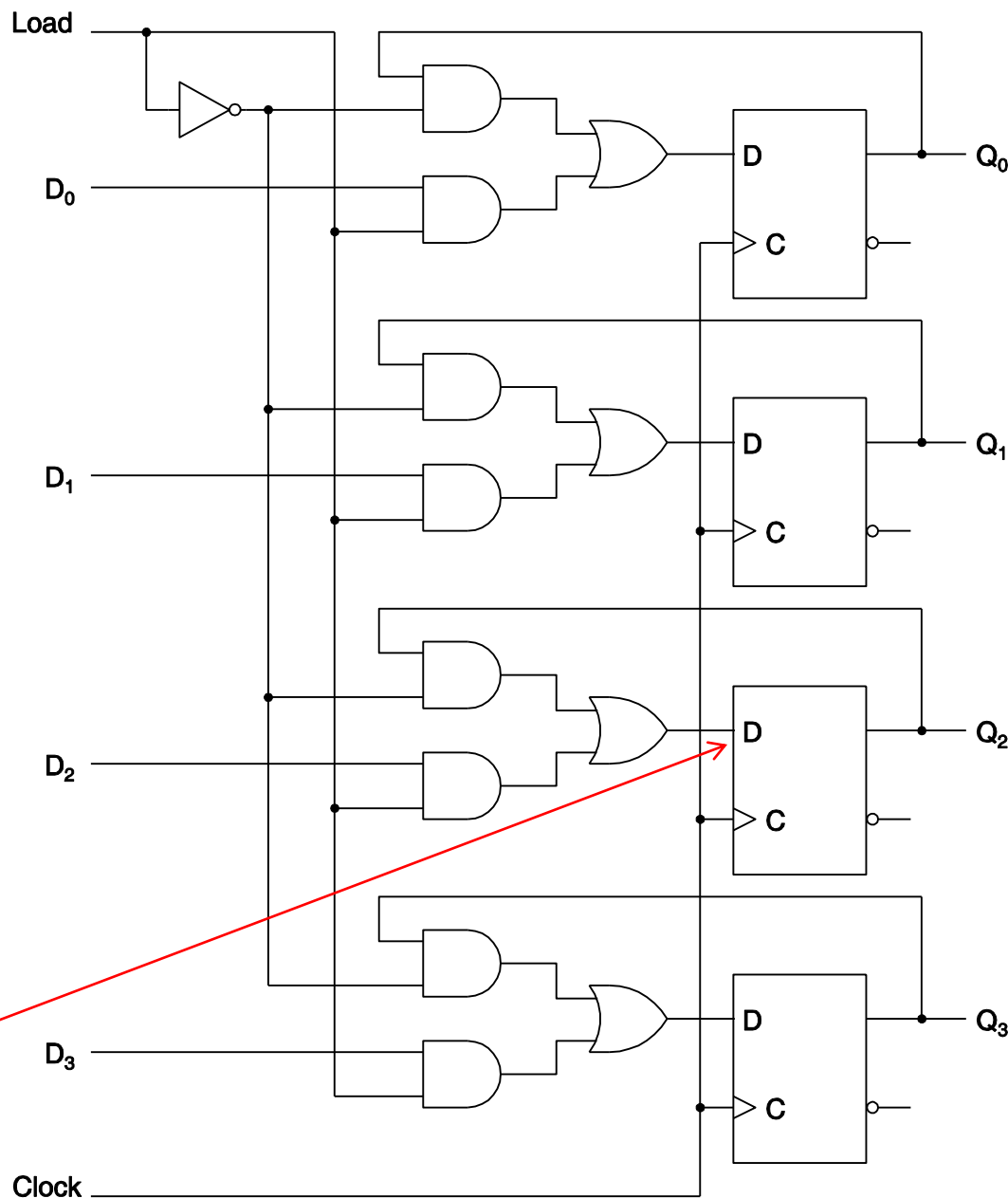
Diagramma di temporizzazione

Registri a Caricamento Parallelo - II

**Registro a
caricamento
parallelo a 4 bit**

**Controllo degli ingressi
senza clock gating !**

$$D = Q_i \cdot \overline{Load} + D_i \cdot Load$$



Registri a Scorrimento (Shift Registers)

Registro a scorrimento a 4 bit

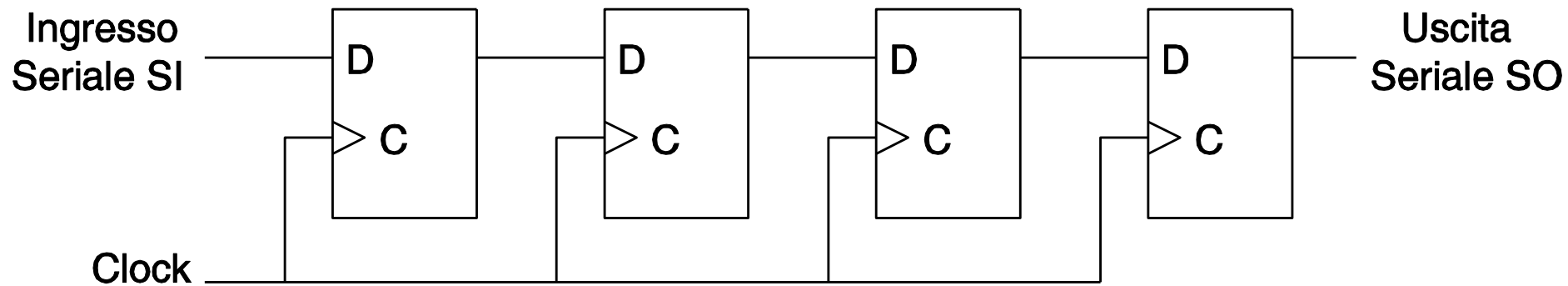
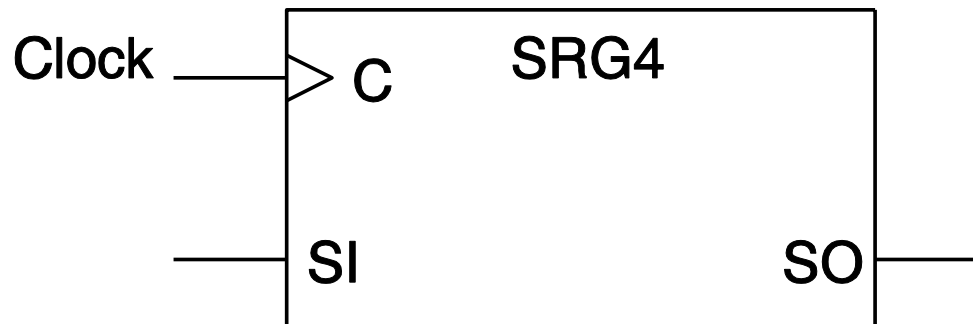


Diagramma logico



Simbolo

Registri a Scorrimento: Trasferimento Seriale

Diagramma
a blocchi

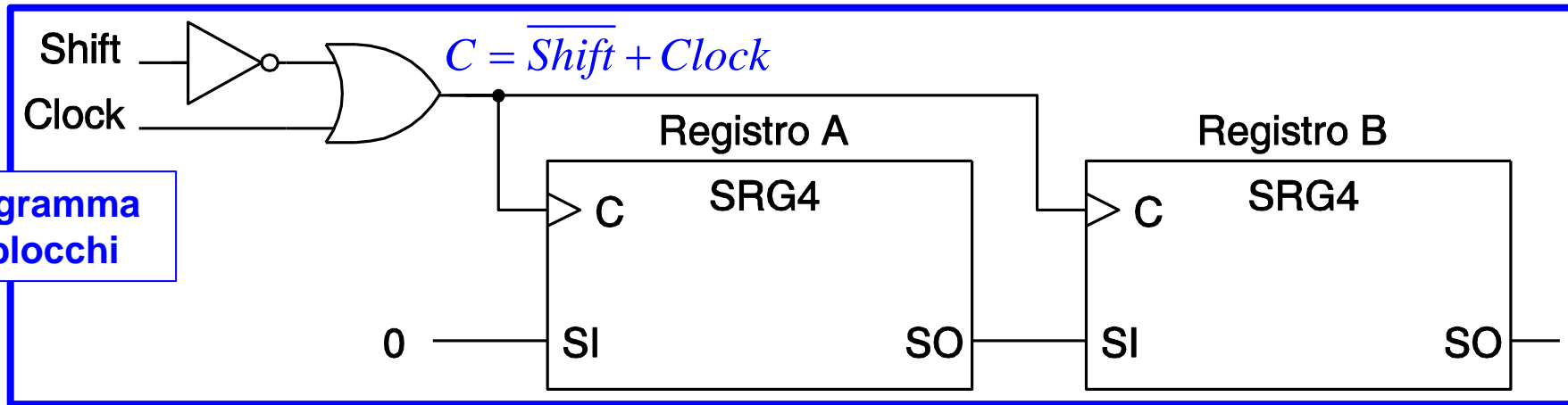
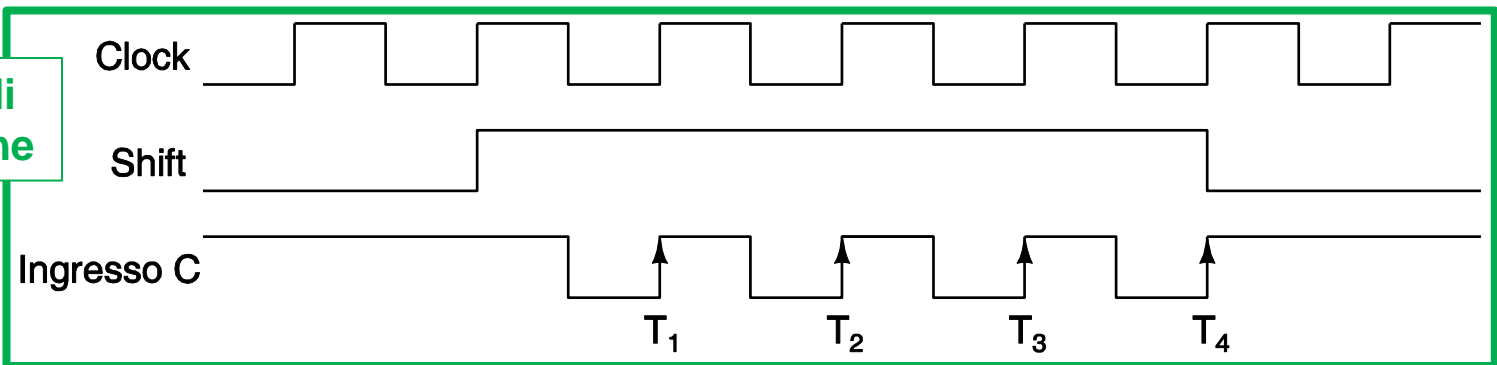


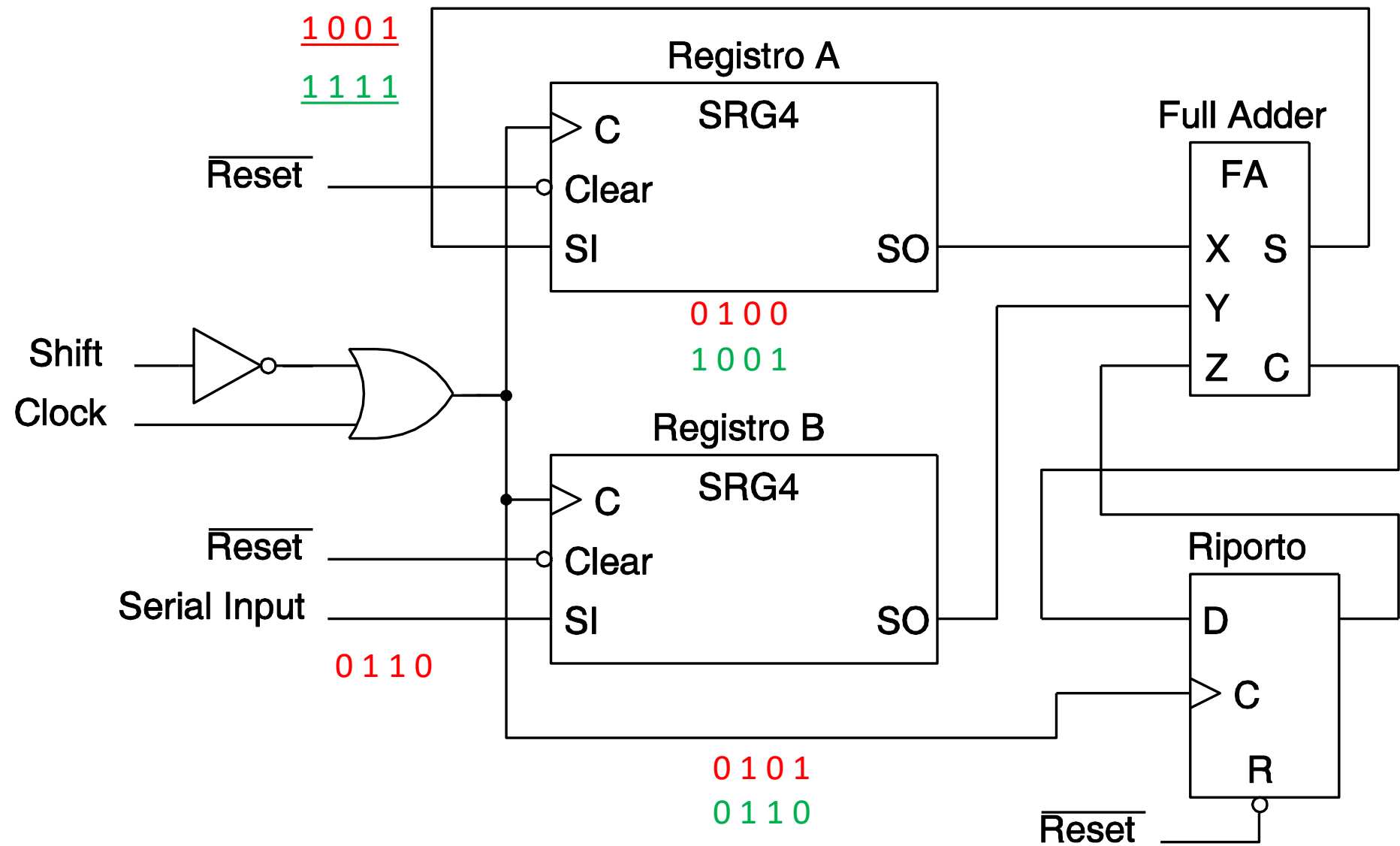
Diagramma di
temporizzazione



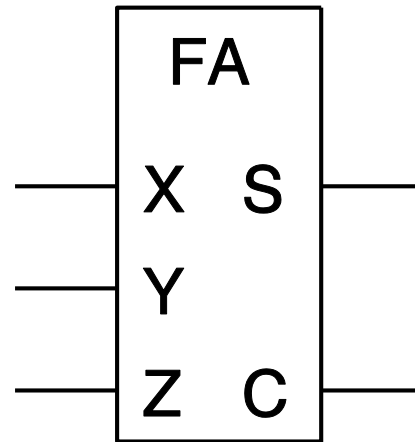
Configurazioni
intermedie dei
registri

Impulso di temporizzazione	Registro a scorrimento A				Registro a scorrimento B			
Valore Iniziale	1	0	1	1	0	0	1	0
Dopo T_1	0	1	0	1	1	0	0	1
Dopo T_2	0	0	1	0	1	1	0	0
Dopo T_3	0	0	0	1	0	1	1	0
Dopo T_4	0	0	0	0	1	0	1	1

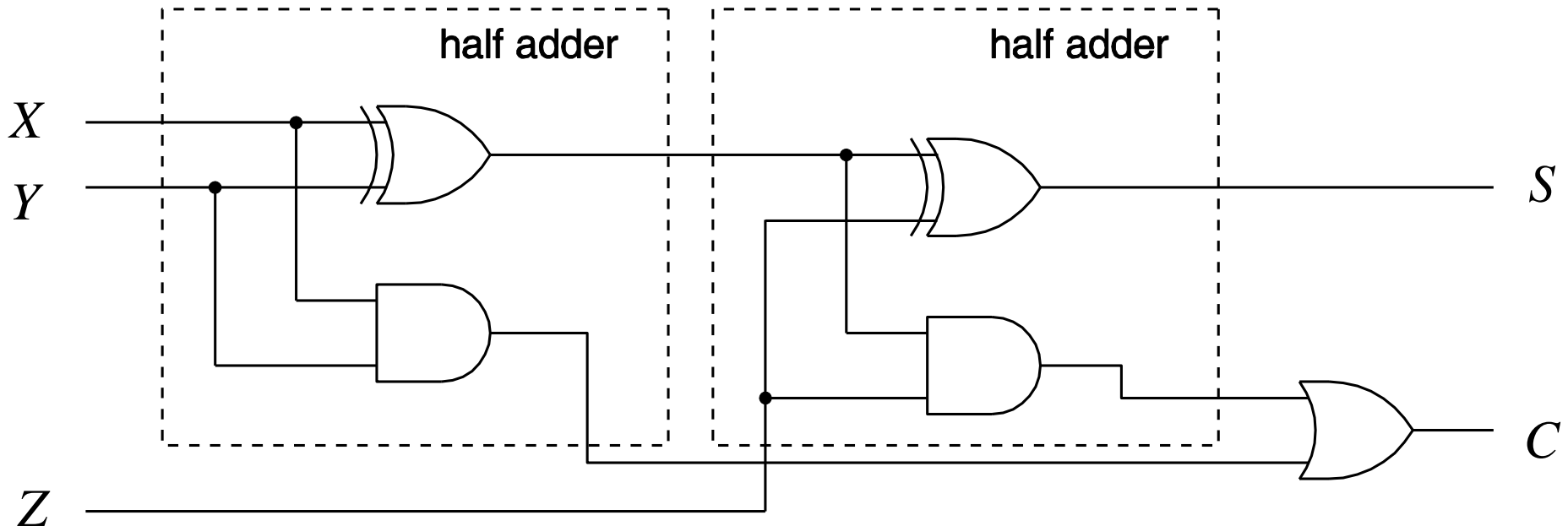
Sommatore Seriale



Full Adder: Circuito Logico

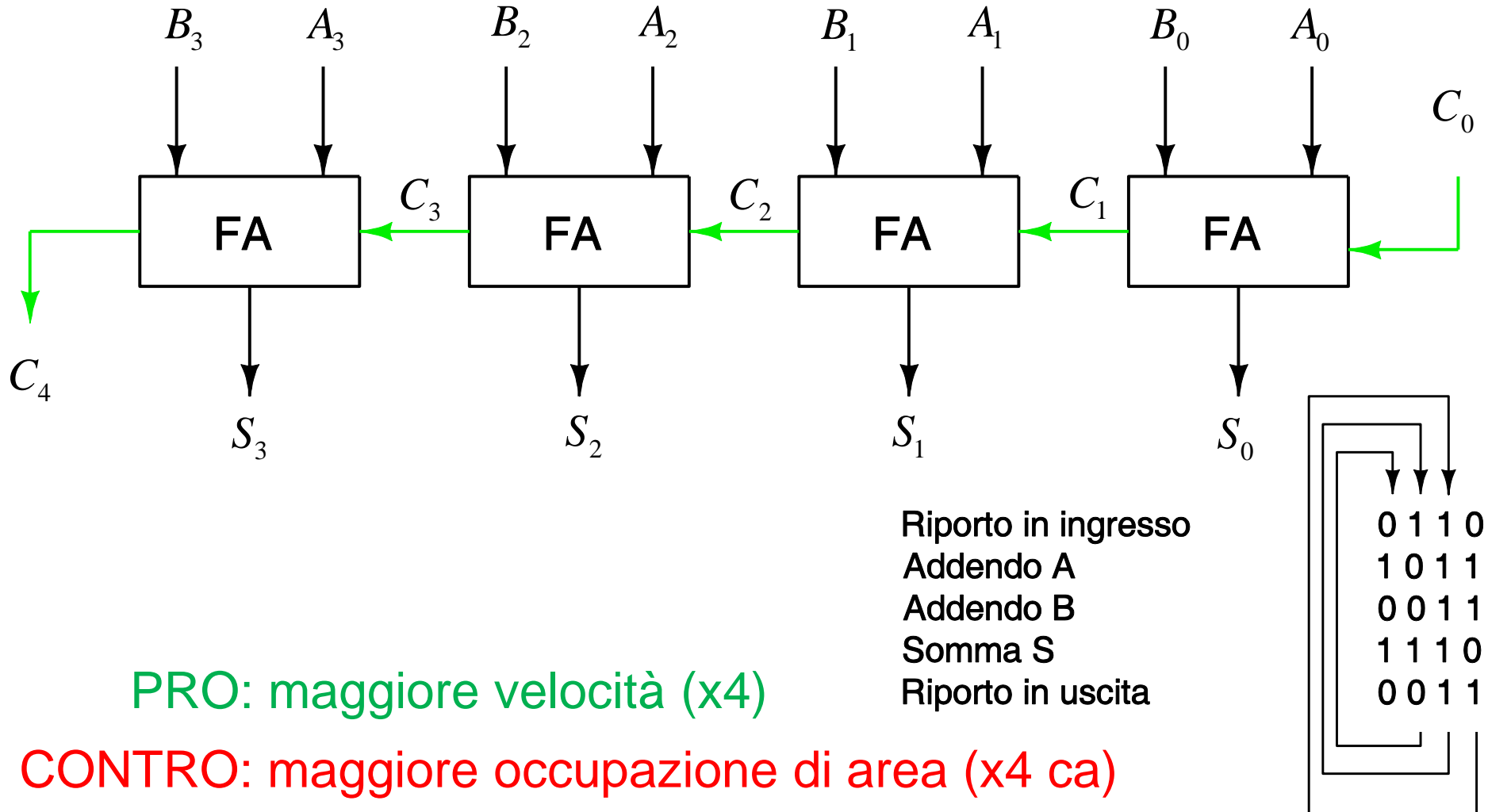


Full Adder

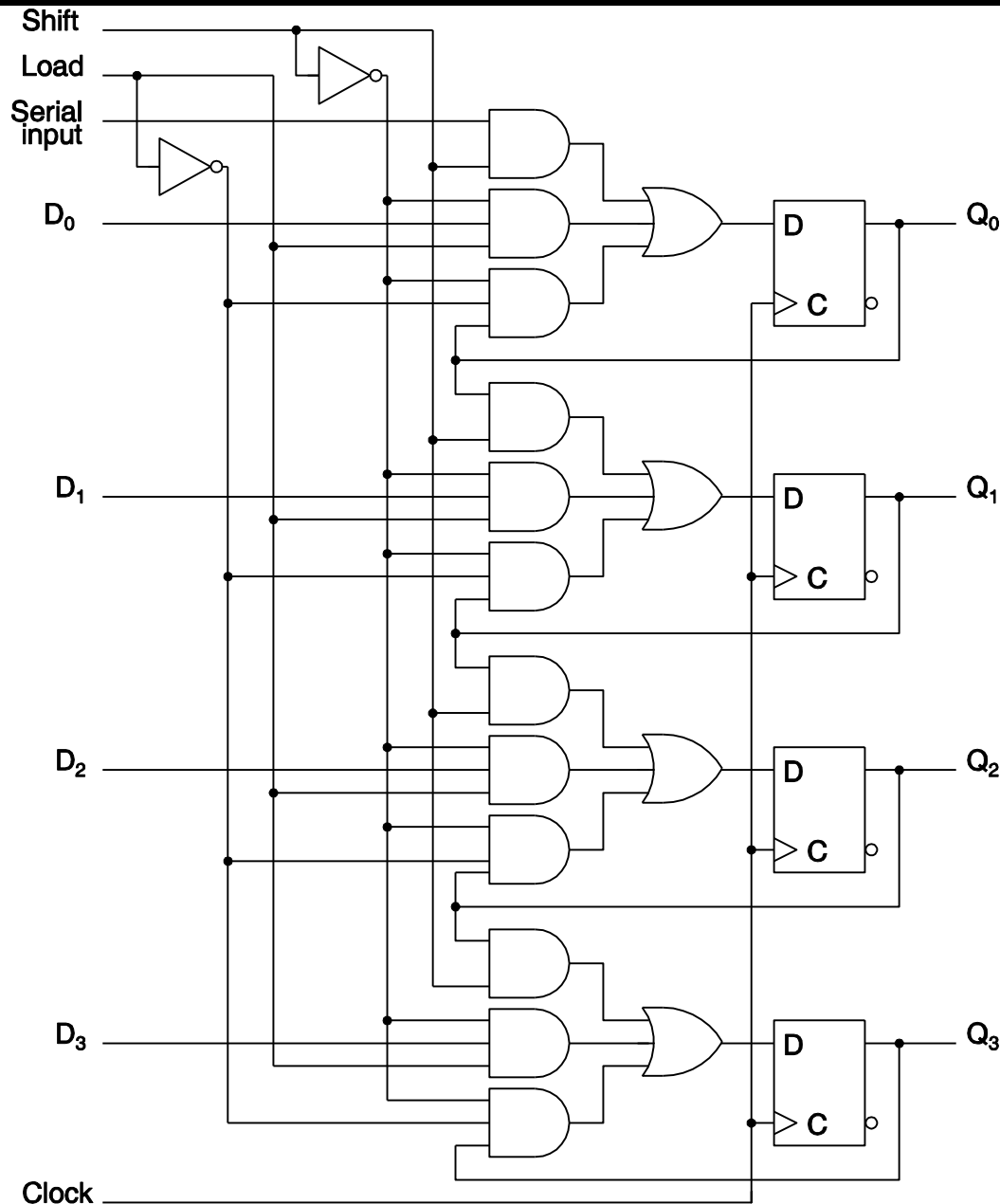


Sommatore Parallelo

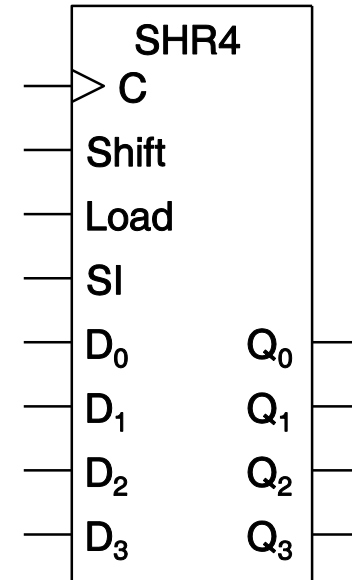
Sommatore a 4-bit con riporto in cascata (4-bit Ripple Carry Adder) realizzato con quattro Full Adder:



Registri a Scorrimento a Caricamento Parallelo



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



**Simbolo
(4-bit shift-register)**

Registri a Scorrimento Bidirezionali

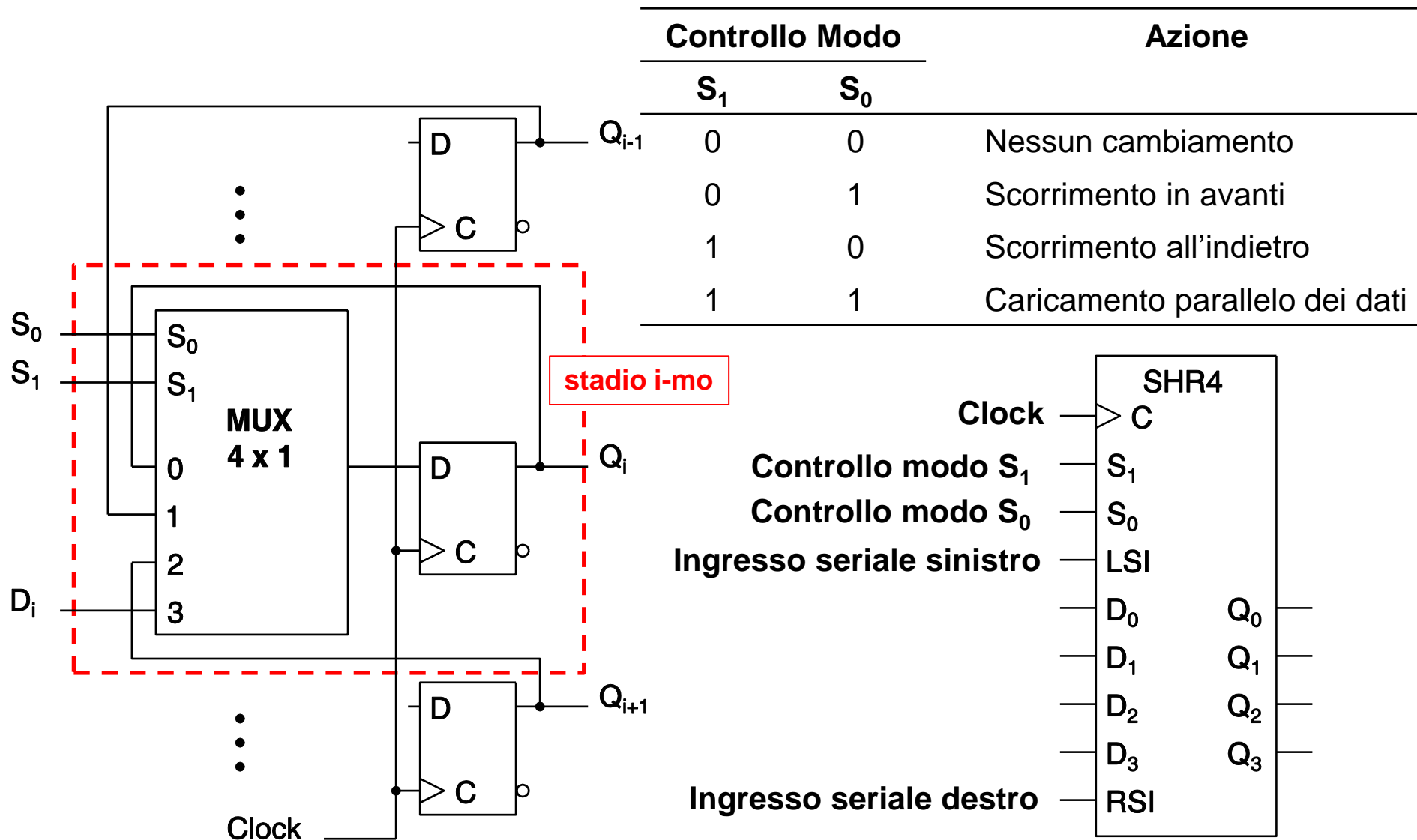


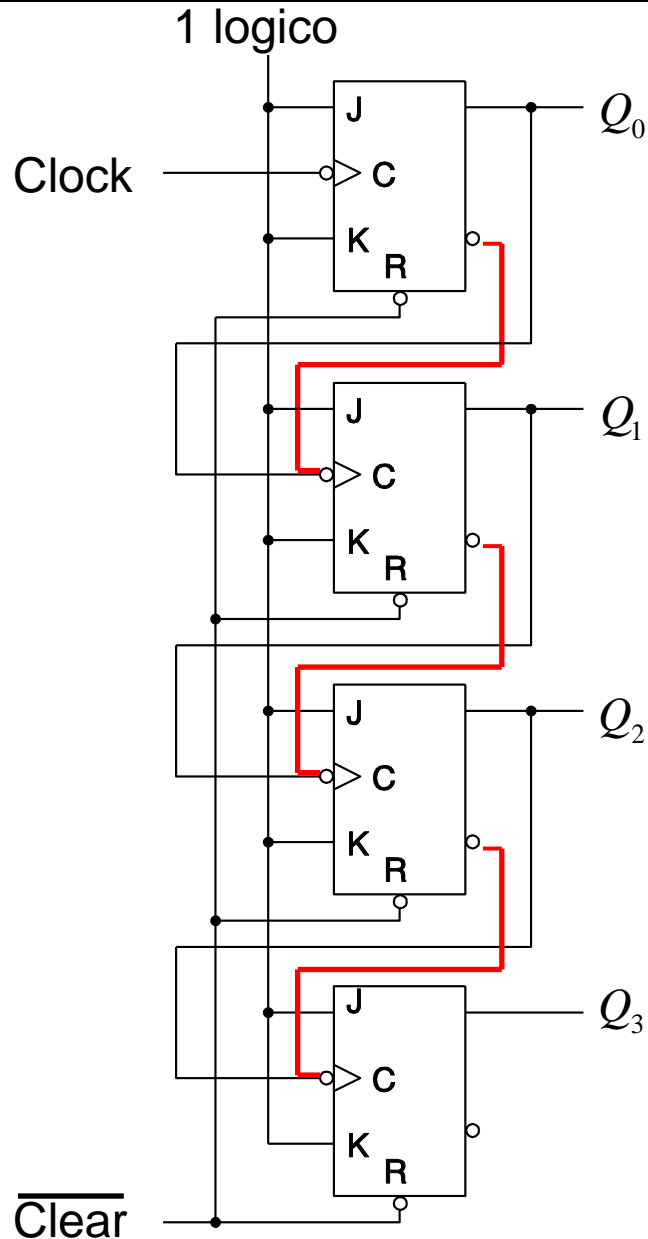
Diagramma logico stadio i-mo

Simbolo
(4-bit bidirectional shift-register)

Un registro che si evolve secondo una sequenza predefinita di stati, all'applicazione di impulsi di ingresso, è denominato contatore

1. Contatori Asincroni
2. Contatori Sincroni

Contatori Binari in Cascata (Asincroni)



Sequenze di conteggio di un contatore binario

Conteggio crescente

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

Conteggio decrescente

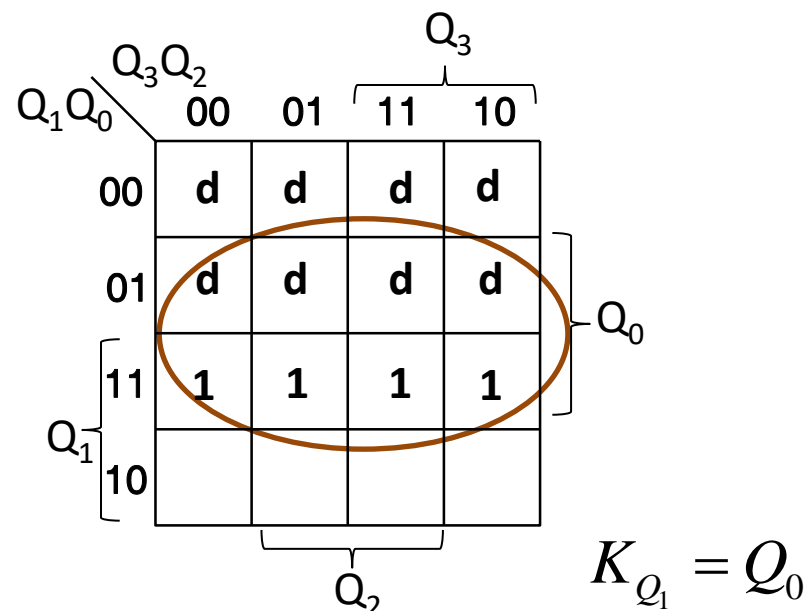
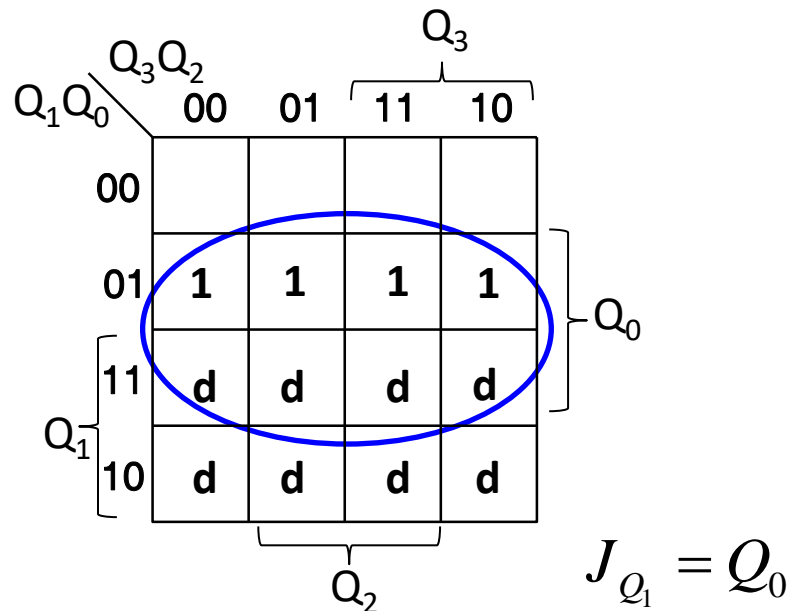
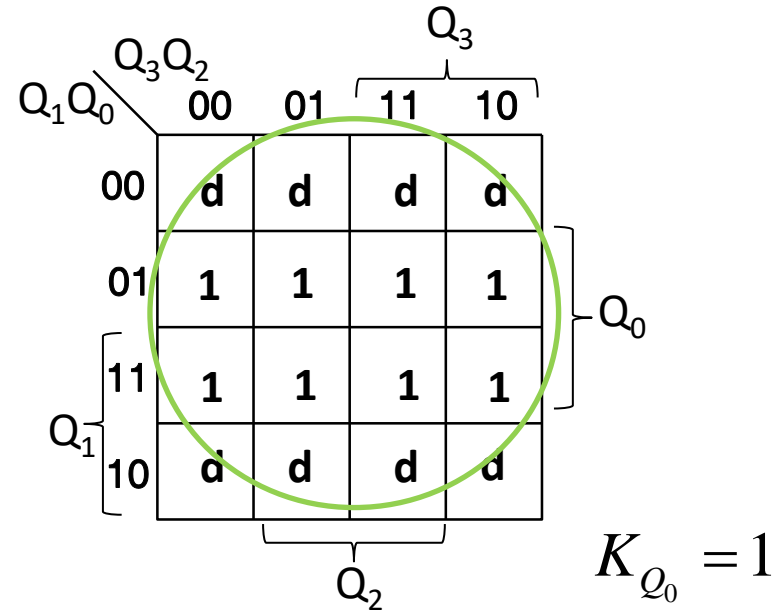
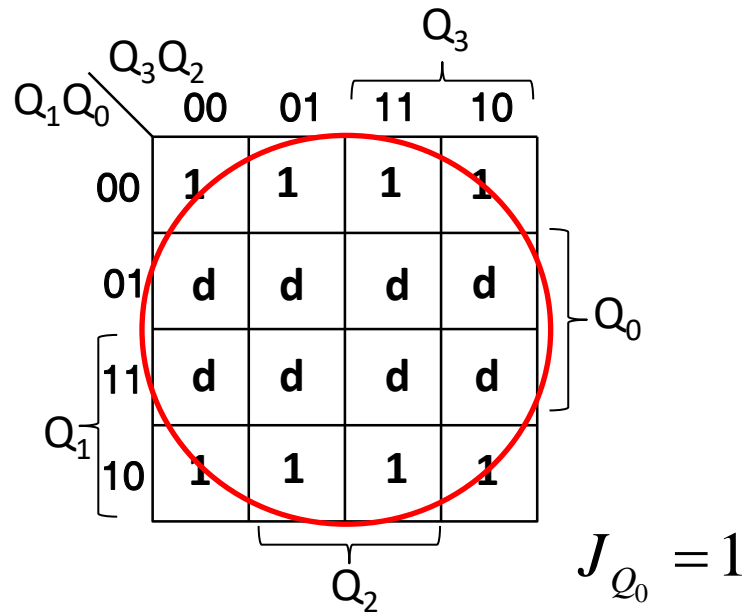
Q_3	Q_2	Q_1	Q_0
1	1	1	1
1	1	1	0
1	1	0	1
1	1	0	0
1	0	1	1
1	0	1	0
1	0	0	1
1	0	0	0
0	1	1	1
0	1	1	0
0	1	0	1
0	1	0	0
0	0	1	1
0	0	1	0
0	0	0	1
0	0	0	0

Contatori Binari Sincroni : Progettazione con flip flop J-K

Tabella di stato e ingressi ai flip flop per il contatore binario a 4 bit

Stato Presente				Stato Futuro				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	d	0	d	0	d	1	d
0	0	0	1	0	0	1	0	0	d	0	d	1	d	d	1
0	0	1	0	0	0	1	1	0	d	0	d	d	0	1	d
0	0	1	1	0	1	0	0	0	d	1	d	d	1	d	1
0	1	0	0	0	1	0	1	0	d	d	0	0	d	1	d
0	1	0	1	0	1	1	0	0	d	d	0	1	d	d	1
0	1	1	0	0	1	1	1	0	d	d	0	d	0	1	d
0	1	1	1	1	0	0	0	1	d	d	1	d	1	d	1
1	0	0	0	1	0	0	1	d	0	0	d	0	d	1	d
1	0	0	1	1	0	1	0	d	0	0	d	1	d	d	1
1	0	1	0	1	0	1	1	d	0	0	d	d	0	1	d
1	0	1	1	1	1	0	0	d	0	1	d	d	1	d	1
1	1	0	0	1	1	0	1	d	0	d	0	0	d	1	d
1	1	0	1	1	1	1	0	d	0	d	0	1	d	d	1
1	1	1	0	1	1	1	1	d	0	d	0	d	0	1	d
1	1	1	1	0	0	0	0	d	1	d	1	d	1	d	1

Contatori Binari Sincroni : Progettazione con flip flop J-K



Contatori Binari Sincroni : Progettazione con flip flop J-K

Q_1Q_0		Q_3			
		Q_3Q_2	00	01	$\overbrace{11 \quad 10}^{Q_2}$
Q_1	00		d	d	
	01		d	d	
	11	1	d	d	1
	10		d	d	

$J_{Q_2} = Q_0 \cdot Q_1$

Q_1Q_0		Q_3			
		Q_3Q_2	00	01	$\overbrace{11 \quad 10}^{Q_2}$
Q_1	00	d			d
	01	d			d
	11	d	1	1	d
	10	d			d

$K_{Q_2} = Q_0 \cdot Q_1$

Q_1Q_0		Q_3			
		Q_3Q_2	00	01	$\overbrace{11 \quad 10}^{Q_2}$
Q_1	00			d	d
	01			d	d
	11	1	d	d	Q_0
	10			d	
		$\underbrace{\hspace{10em}}_{Q_2}$			

J_{Q_3}

Q_1Q_0		Q_3			
		Q_3Q_2	00	01	$\overbrace{11 \quad 10}^{Q_2}$
Q_1	00	d	d		
	01	d	d		
	11	d	d	1	
	10	d	d		

Q_0

Q_2

$K_{Q_3} = Q_0 \cdot Q_1 \cdot Q_2$

Contatori Binari Sincroni : Progettazione con flip flop J-K

Contatore binario a 4 bit



4 flip flop JK
8 equazioni di ingresso
8 mappe di Karnaugh

EN : ingresso di abilitazione



Equazioni di ingresso
dei flip flop

$$J_{Q_0} = K_{Q_0} = EN$$

$$J_{Q_1} = K_{Q_1} = Q_0 \cdot EN$$

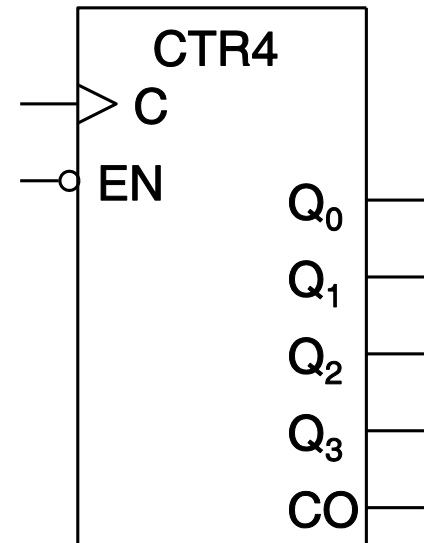
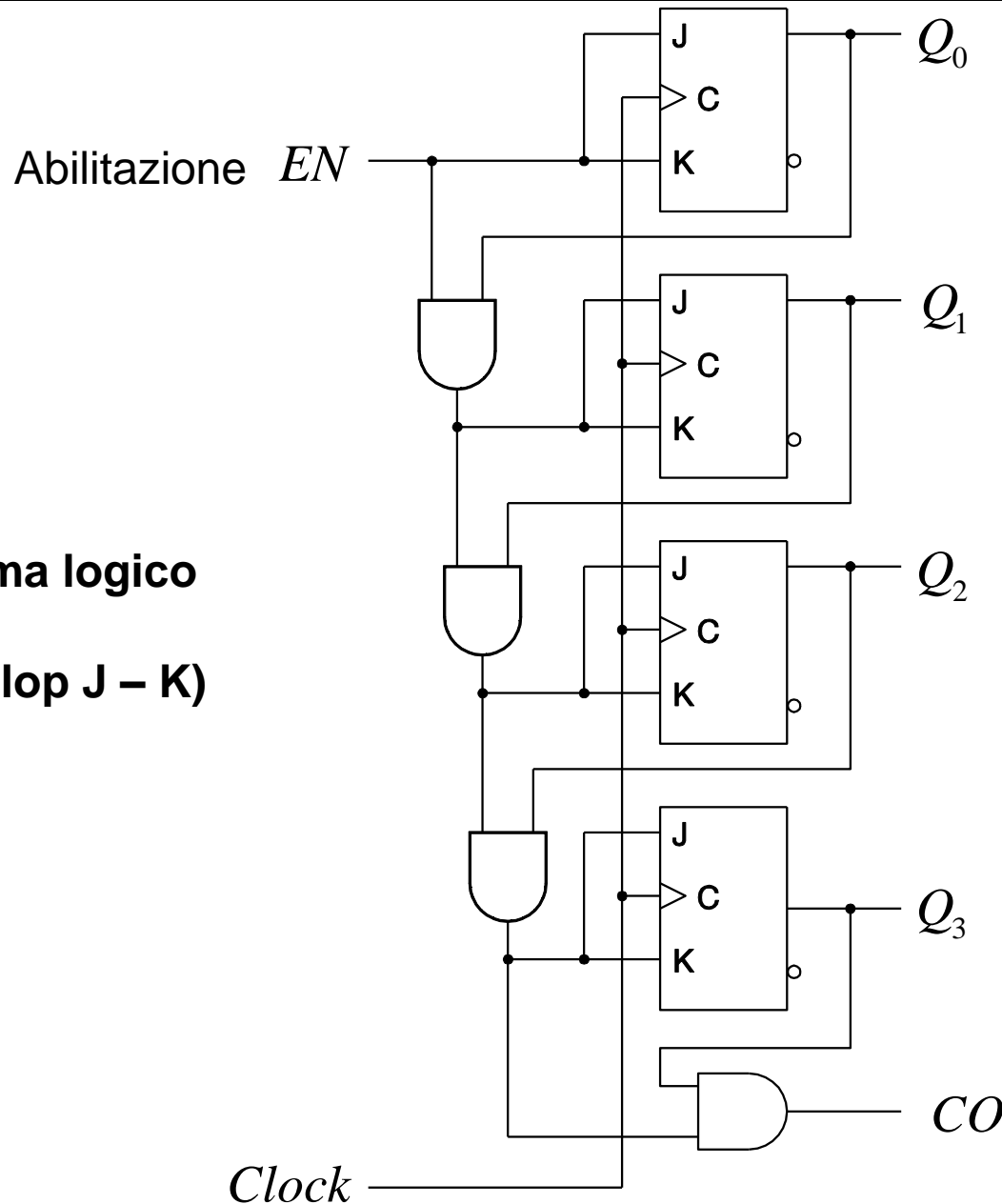
$$J_{Q_2} = K_{Q_2} = Q_0 \cdot Q_1 \cdot EN$$

$$J_{Q_3} = K_{Q_3} = Q_0 \cdot Q_1 \cdot Q_2 \cdot EN$$

Contatore binario a n bit

$$J_{Q_i} = K_{Q_i} = Q_0 \cdot Q_1 \cdot \dots \cdot Q_{i-1} \cdot EN \quad i = 0, \dots, n-1$$

Contatori Binari Sincroni : Progettazione con flip flop J-K



Simbolo

Contatori Binari Sincroni : Progettazione con flip flop D

**Tabella di stato per il
contatore binario a 4 bit**

Stato Presente				Stato Futuro			
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 Binari Sincroni : Progettazione con flip flop D

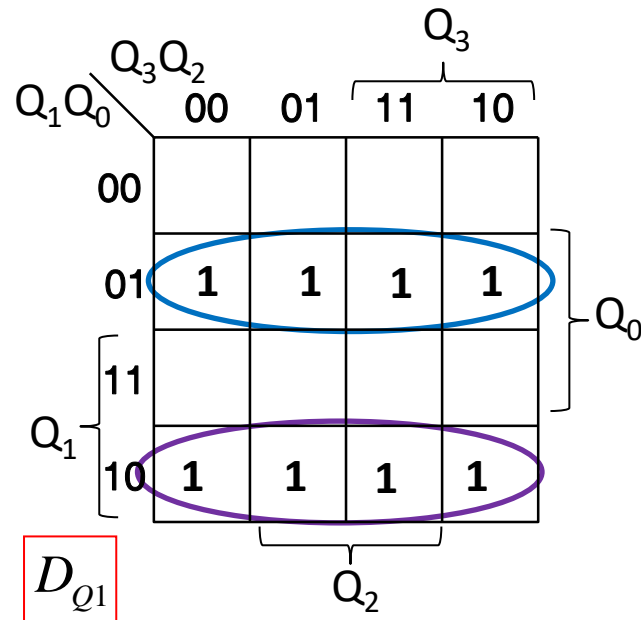
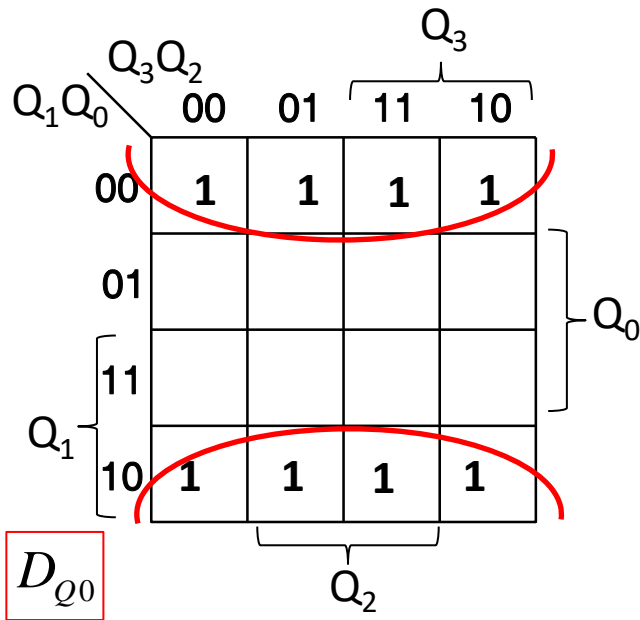
$$D_{Q_0} (Q_3, Q_2, Q_1, Q_0) = \sum_{Q_3 Q_2 Q_1 Q_0} m(0, 2, 4, 6, 8, 10, 12, 14)$$

$$D_{Q_1} (Q_3, Q_2, Q_1, Q_0) = \sum_{Q_3 Q_2 Q_1 Q_0} m(1, 2, 5, 6, 9, 10, 13, 14)$$

$$D_{Q_2} (Q_3, Q_2, Q_1, Q_0) = \sum_{Q_3 Q_2 Q_1 Q_0} m(3, 4, 5, 6, 11, 12, 13, 14)$$

$$D_{Q_3} (Q_3, Q_2, Q_1, Q_0) = \sum_{Q_3 Q_2 Q_1 Q_0} m(7, 8, 9, 10, 11, 12, 13, 14)$$

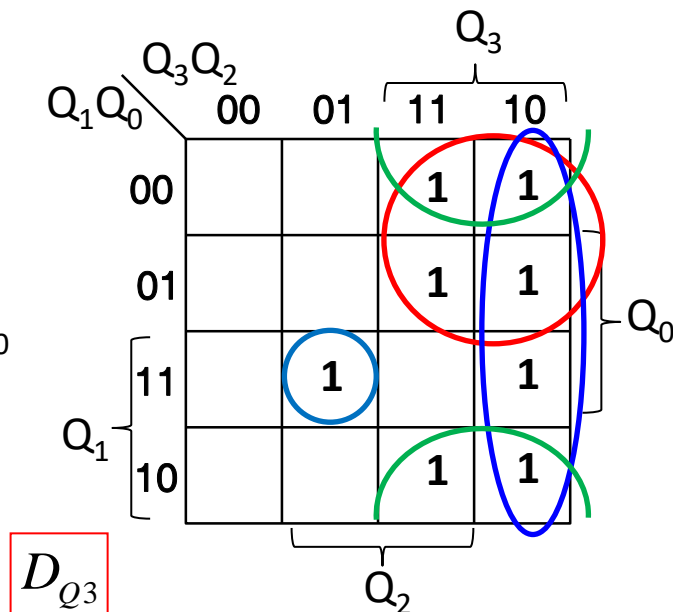
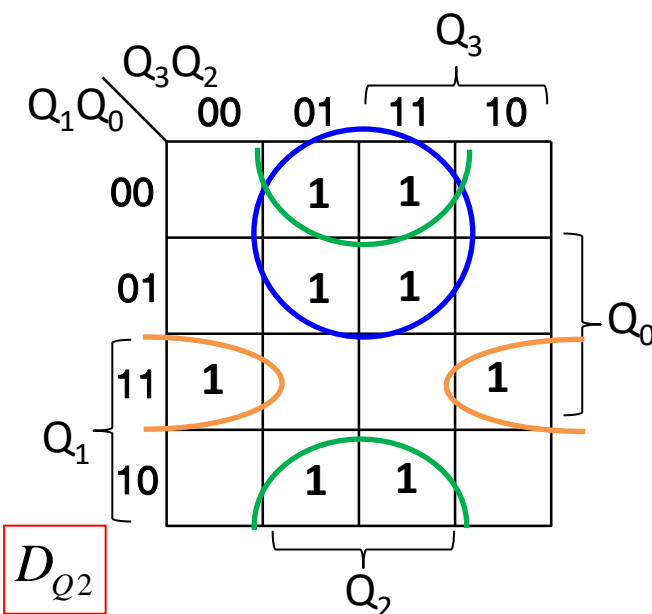
Contatori Binari Sincroni : Progettazione con flip flop D



$$D_{Q0} = \overline{Q_0}$$

$$D_{Q1} = \overline{Q_0} \cdot Q_1 + \overline{Q_1} \cdot Q_0$$

$$D_{Q2} = \overline{Q_0} \cdot Q_2 + \overline{Q_1} \cdot Q_2 + Q_0 \cdot Q_1 \cdot \overline{Q_2}$$



$$D_{Q3} = Q_3 \cdot \overline{Q_2} + Q_3 \cdot \overline{Q_1} + Q_3 \cdot \overline{Q_0} + \overline{Q_3} \cdot Q_2 \cdot Q_1 \cdot Q_0$$

Contatori Binari Sincroni : Progettazione con flip flop D

$$D_{Q_0} = \overline{Q_0}$$

$$D_{Q_1} = Q_0 \oplus Q_1$$

$$D_{Q_2} = Q_2 \oplus (Q_0 \cdot Q_1)$$

$$D_{Q_3} = Q_3 \oplus (Q_2 \cdot Q_1 \cdot Q_0)$$

EN: ingresso di abilitazione

Equazioni di ingresso dei flip flop

$$D_{Q_0} = Q_0 \oplus EN$$

$$D_{Q_1} = Q_1 \oplus (Q_0 \cdot EN)$$

$$D_{Q_2} = Q_2 \oplus (Q_0 \cdot Q_1 \cdot EN)$$

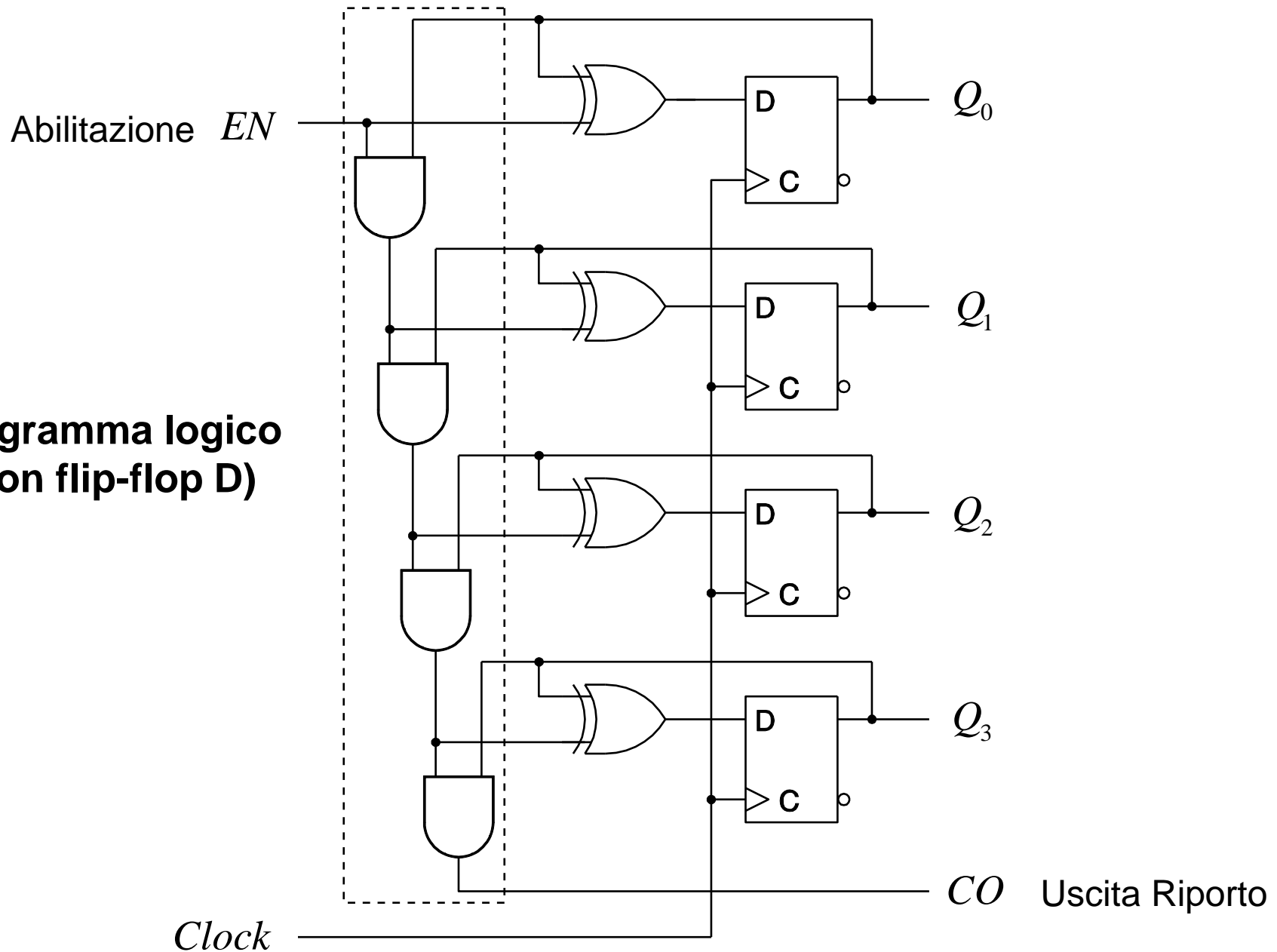
$$D_{Q_3} = Q_3 \oplus (Q_0 \cdot Q_1 \cdot Q_2 \cdot EN)$$

Contatore binario a n bit

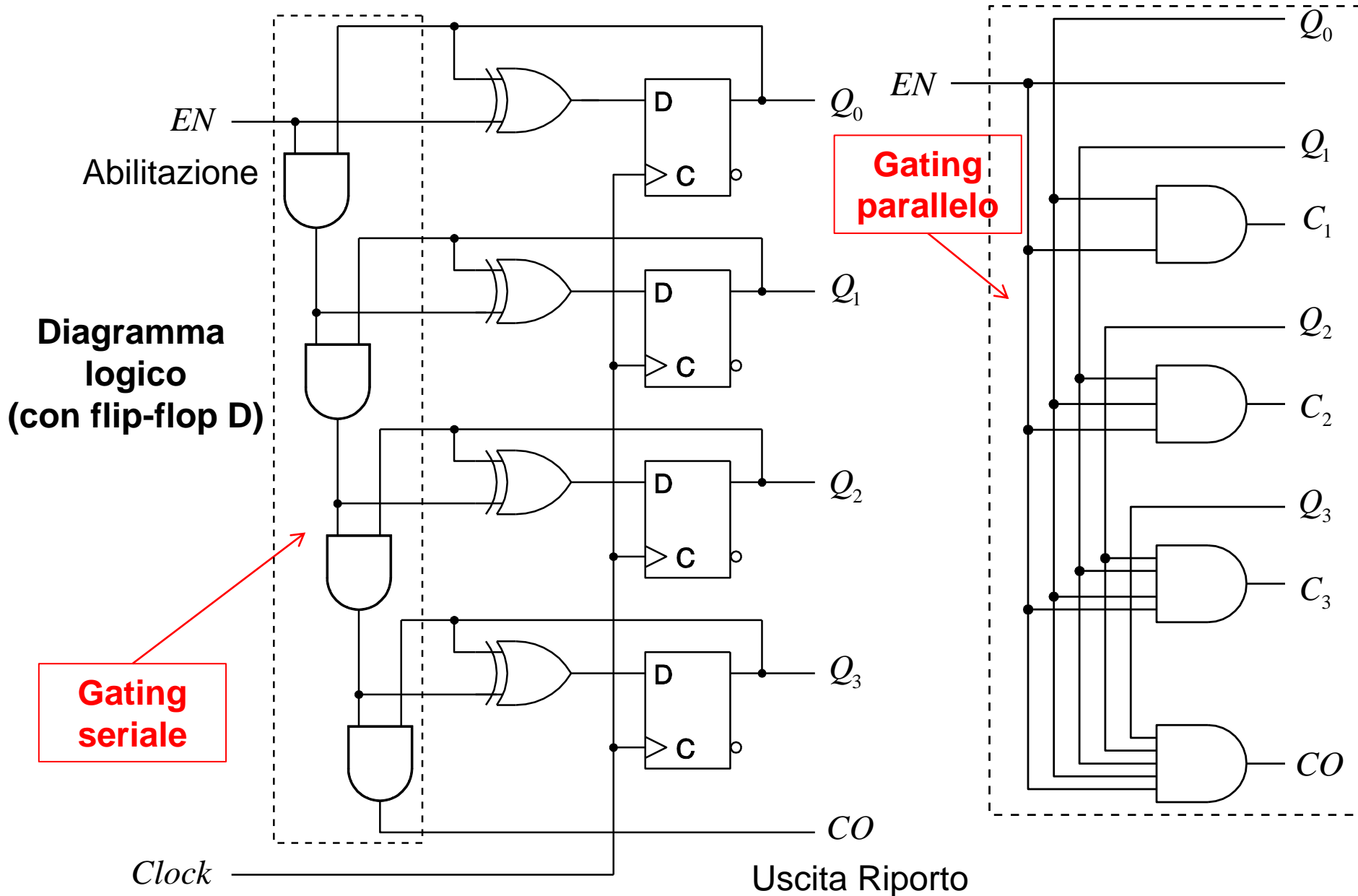
$$D_{Q_i} = Q_i \oplus (Q_0 \cdot Q_1 \cdot \dots \cdot Q_{i-1} \cdot EN) \quad i = 1, \dots, n$$

Contatori Binari Sincroni : Progettazione con flip flop D

**Diagramma logico
(con flip-flop D)**

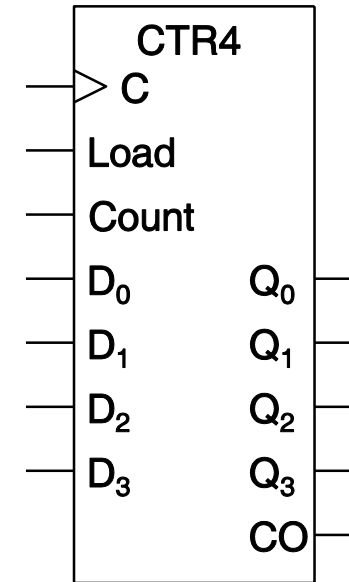
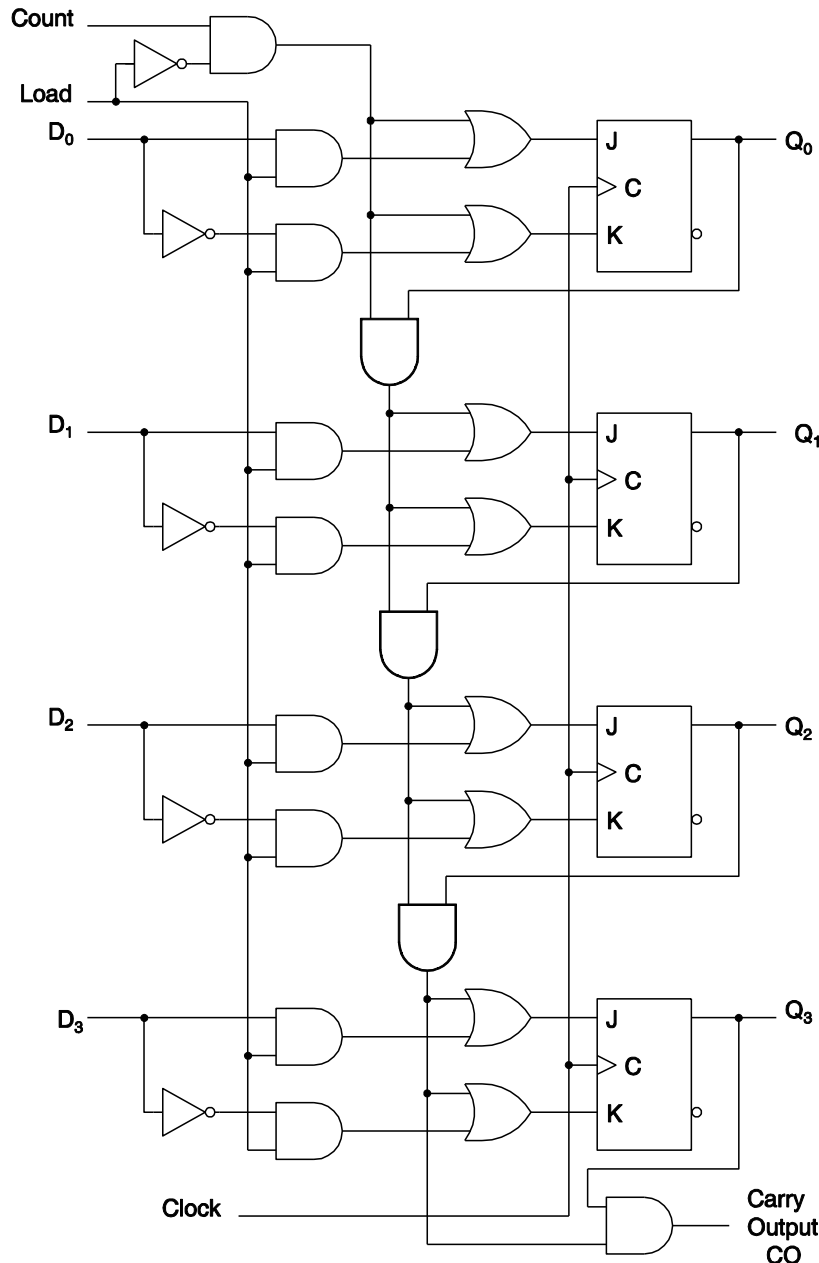


Contatori Binari Sincroni : Gating Seriale e Parallelo



Contatori Binari a Caricamento Parallelo

Registri di Caricamento e Incremento



Ingressi di controllo		Funzionamento
Load	Count	
0	0	Nessun cambiamento
0	1	Conteggio
1	d	Caricamento parallelo dei dati

Contatore BCD

Contatore BCD sincrono (senza ingresso di caricamento)

