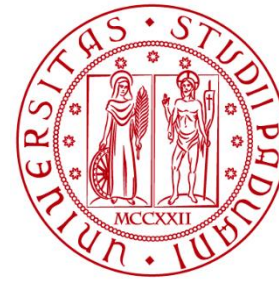




DEI  
DIPARTIMENTO DI  
INGEGNERIA DELL'INFORMAZIONE



UNIVERSITÀ  
DEGLI STUDI  
DI PADOVA

# Sistemi Digitali

## Analisi dei Circuiti Sequenziali

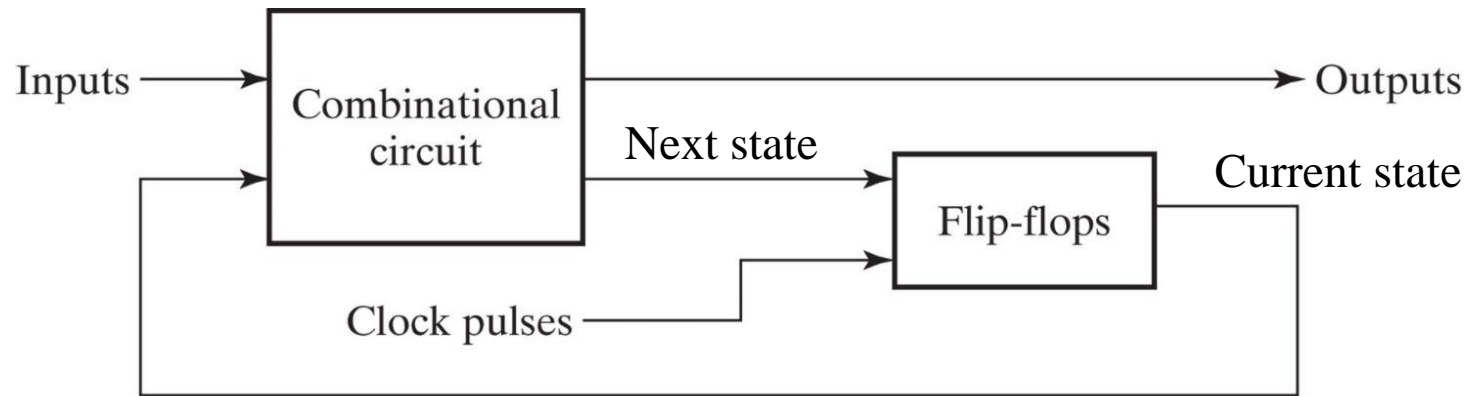
Marta Bagatin, [marta.bagatin@unipd.it](mailto:marta.bagatin@unipd.it)

Corso di Laurea in Ingegneria dell'Informazione  
Anno accademico 2022-2023

# Scopo della lezione

- **Analizzare un circuito sequenziale** tramite lo studio di
  - Equazioni degli ingressi dei flip-flop e delle uscite del circuito
  - Tabella di aggiornamento degli stati
  - Diagramma degli stati
  - Simulazione/ diagramma temporale
- Capire la differenza tra il **modello di Mealy** e il **modello di Moore** per i circuiti sequenziali

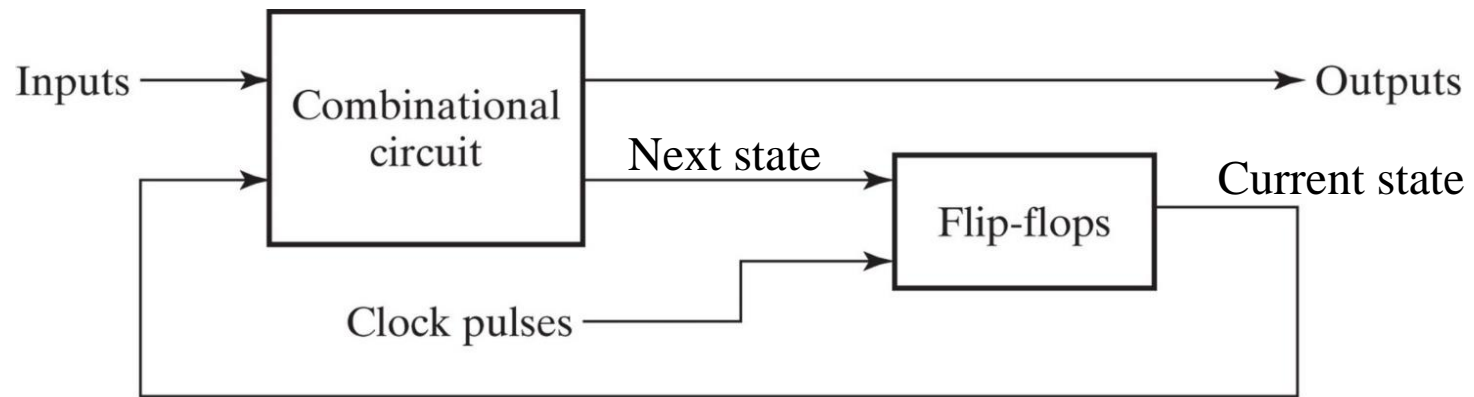
# Circuiti sequenziali sincroni: riepilogo



- In un circuito sequenziale le uscite dipendono non solo dagli ingressi, ma anche dal suo stato, che riflette la storia passata
- Ogni sistema sequenziale contiene un percorso di retroazione
- In un circuito sequenziale le uscite e lo stato futuro sono calcolati da una rete combinatoria, a partire da ingressi e stato corrente
- Un circuito sequenziale sincrono (o macchina a stati finiti) è **temporizzato da un clock**, le **variabili di stato sono immagazzinate in flip-flop**, il cui stato viene **aggiornato solo in corrispondenza a determinati eventi nel clock** (per es. i fronti di salita)

# Analisi di un circuito sequenziale

# Analisi di circuiti sequenziali sincroni



- Per analizzare il comportamento di un circuito sequenziale sincrono dobbiamo descrivere l'**evoluzione temporale di uscite e stato futuro, in funzione degli ingressi e dello stato corrente del sistema**. In particolare:
  - Equazioni di ingresso dei flip-flop: calcolano lo stato futuro del sistema
  - Equazioni dell'uscita: calcolano l'uscita del sistema
  - Tabella degli stati: elenca stato futuro e uscita in corrispondenza a tutte le possibili combinazioni di ingressi e stati correnti
  - Diagramma degli stati: visualizza graficamente le informazioni contenute nella tabella degli stati
  - Simulazione (diagramma temporale): verifica il funzionamento del sistema

# Es: Analisi di un circuito sequenziale sincrono

(Fig. 4.13)

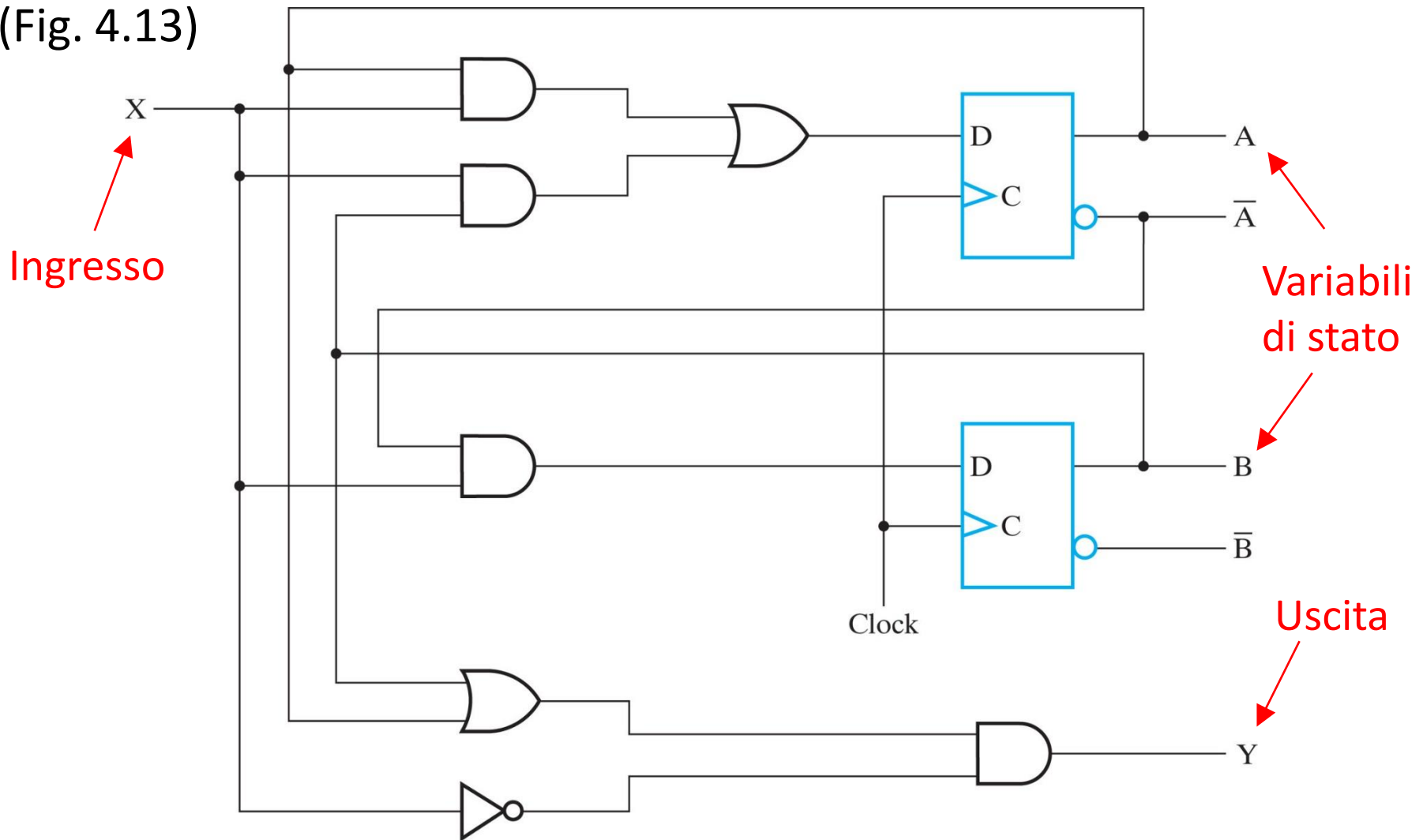
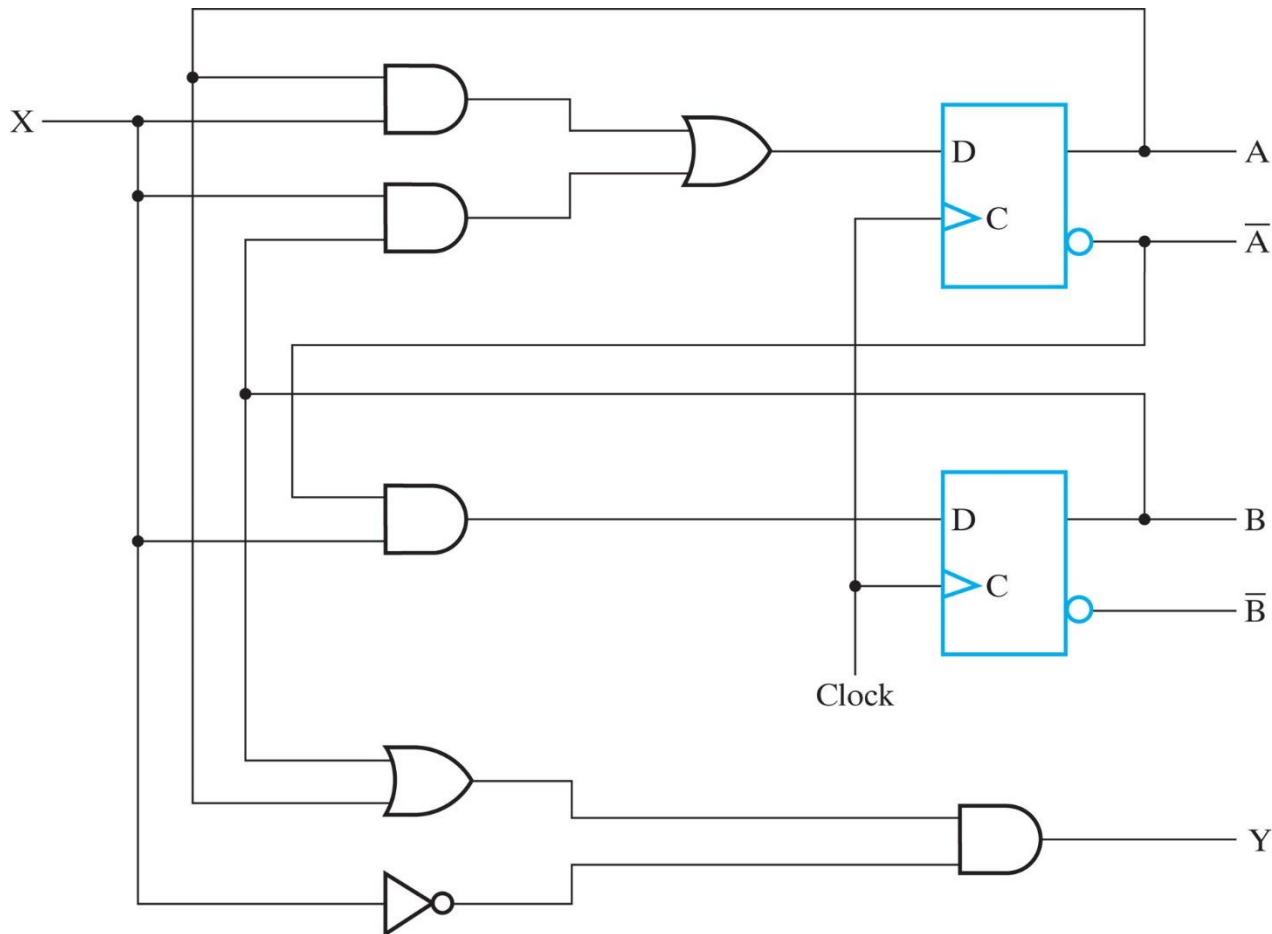


Diagramma logico: composto da flip-flop e da un insieme di porte logiche

# Equazioni d'ingresso dei flip-flop

- Le equazioni d'ingresso dei flip-flop descrivono la **logica combinatoria che genera i segnali d'ingresso per i flip-flop** (stato futuro, che al ciclo di clock successivo diventerà lo stato presente)



Equazioni d'ingresso dei flip-flop:

$$D_A = AX + BX$$

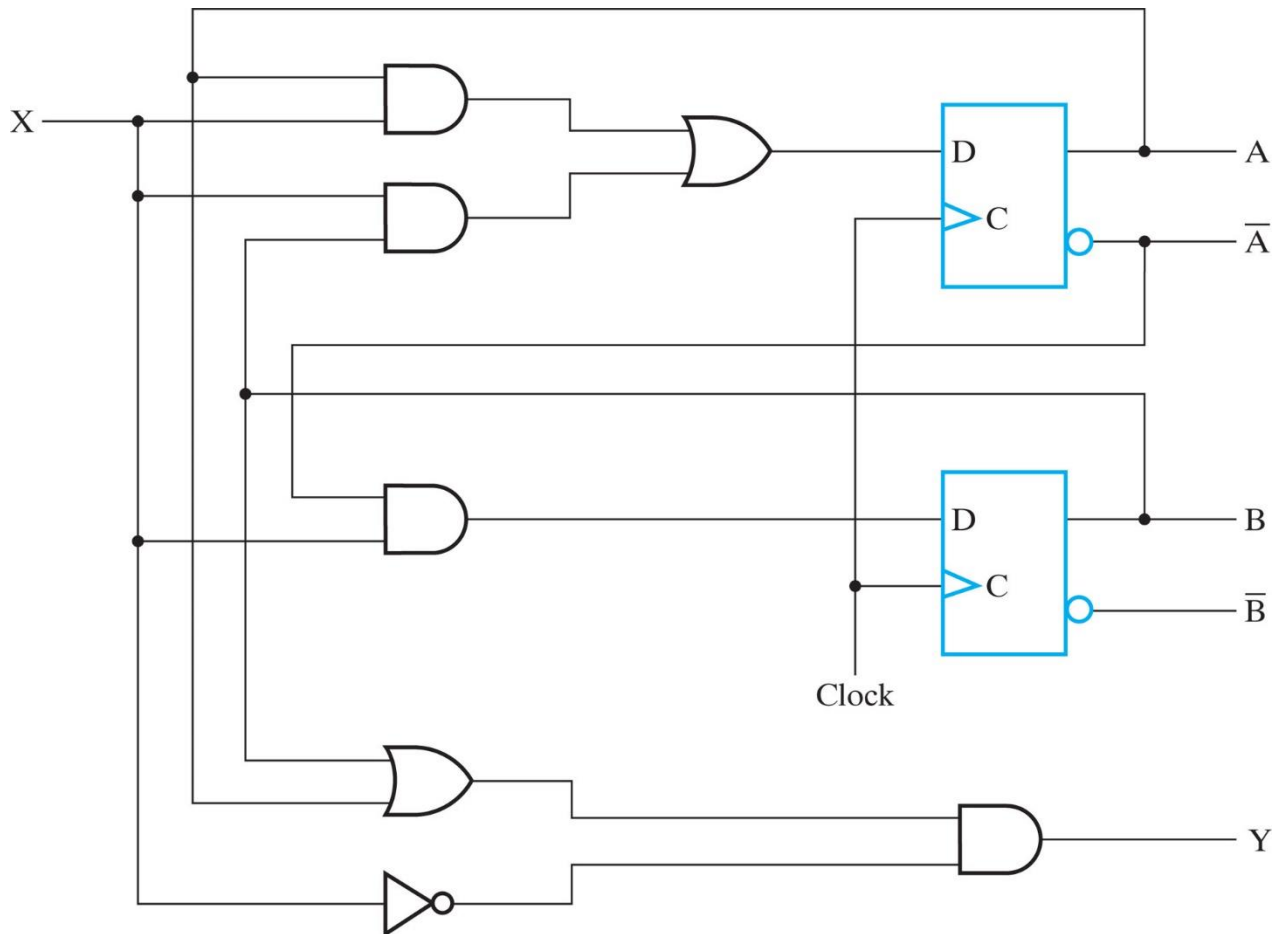
$$D_B = \bar{A}X$$



D indica il tipo di flip-flop, il pedice indica il nome della variabile di stato

# Equazione di uscita del circuito

- L'equazione di uscita del circuito descrive **la logica combinatoria che genera il segnale di uscita del circuito sequenziale**



Equazione di uscita del circuito:

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



# Tabella degli stati

Tempo t			Tempo t +1		Tempo t
<u>Present State</u>		<u>Input</u>	<u>Next State</u>		<u>Output</u>
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

- Descrive le **transizioni degli stati**: per ogni combinazione di stato presente e ingresso elenca lo stato futuro (stato al tempo t + 1, cioè l'ingresso dei flip-flop che diventerà la loro uscita al successivo ciclo di clock)
- Descrive anche le uscite, in funzione dello stato presente e dell'ingresso (tempo t, cioè all'attuale ciclo di clock)

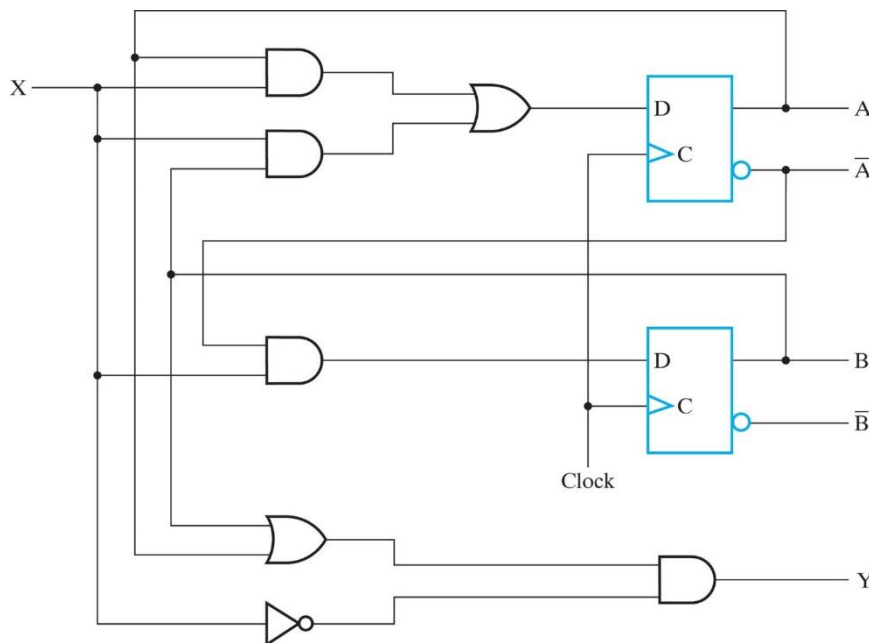
# Tabella degli stati

Tempo t			Tempo t +1		Tempo t
<u>Present State</u>		<u>Input</u>	<u>Next State</u>		<u>Output</u>
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

- Per ogni valore degli ingressi, sono elencate tutte le possibili combinazioni degli stati: in un circuito con  $m$  flip-flop (cioè con  $m$  variabili di stato) e  $n$  ingressi, la tabella degli stati ha  $2^{m+n}$  righe e un numero di colonne per lo stato pari a  $m$

# Tabella degli stati: derivazione

Present State		Input	Next State		Output
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



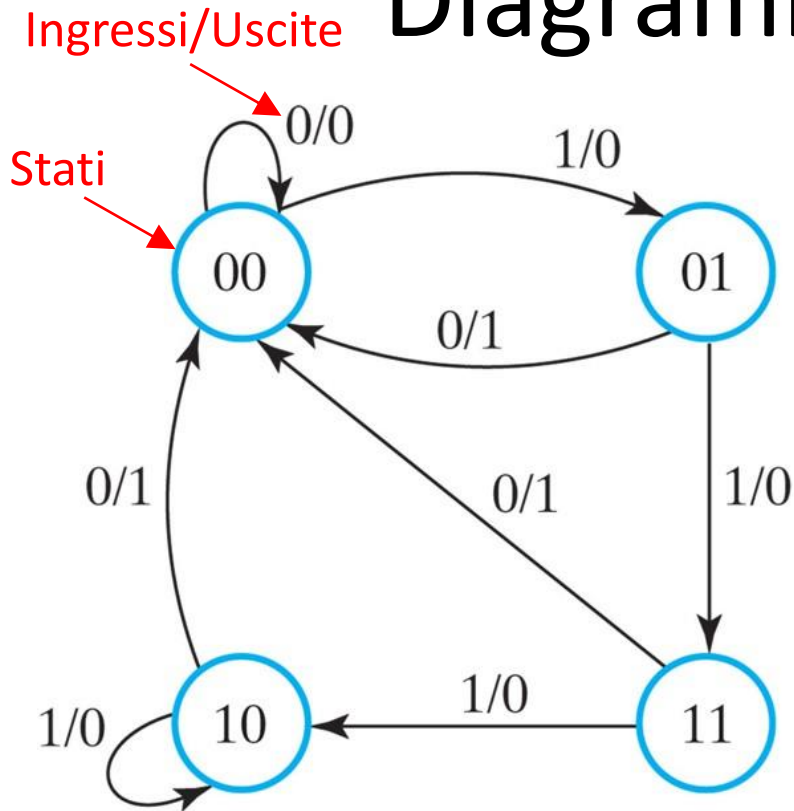
- Costruire una tabella con tutte le combinazioni di stati presenti e input
- Per ciascuna riga, determinare gli stati futuri in base al diagramma logico o alle equazioni d'ingresso dei flip-flop [es. in un D-FF:  $A(t+1) = D_A(t)$ ]
- Per ciascuna riga, determinare le uscite in base al diagramma logico o all'equazione dell'uscita del sistema

# Tabella bidimensionale degli stati

Present State		Next State				Output	
		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

- E' una tabella alternativa alla tabella degli stati per descrivere le transizioni: a ciascuno stato viene associata un'unica riga per elencare lo stato futuro e l'uscita, in colonne diverse a seconda del valore dell'ingresso

# Diagramma degli stati

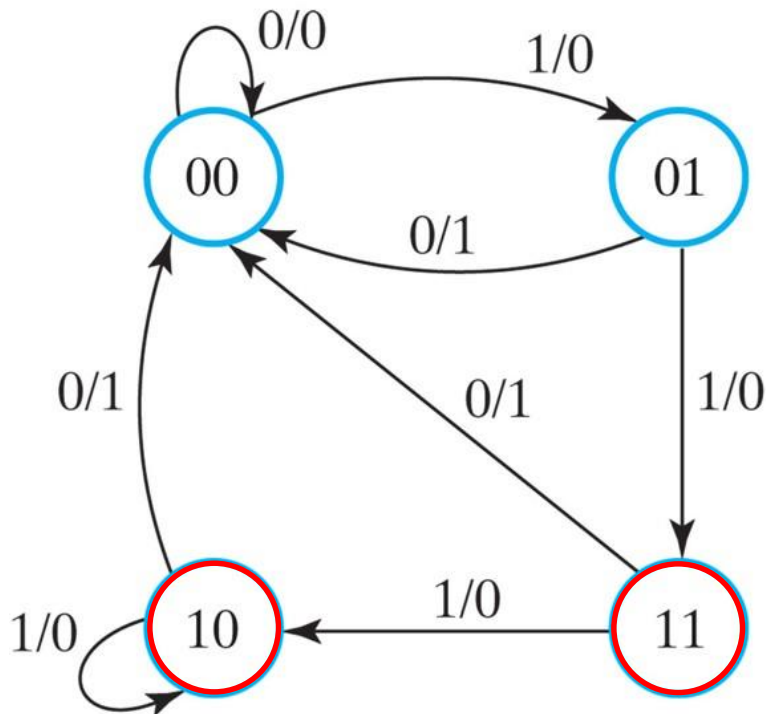


- Gli **stati** del sistema sono rappresentati da cerchi
- Le **transizioni tra stati** sono rappresentate da frecce che partono da uno stato e arrivano ad un altro stato. Ingressi e uscite corrispondenti ad ogni transizione sono indicati vicino alla freccia e separati da una barra

- Il diagramma degli stati contiene le stesse informazioni contenute nella **tabella degli stati**. Le informazioni sono le stesse, ma visualizzate **in forma grafica**
- Osservando il diagramma degli stati, il funzionamento del circuito può essere interpretato in modo più intuitivo

# Stati equivalenti

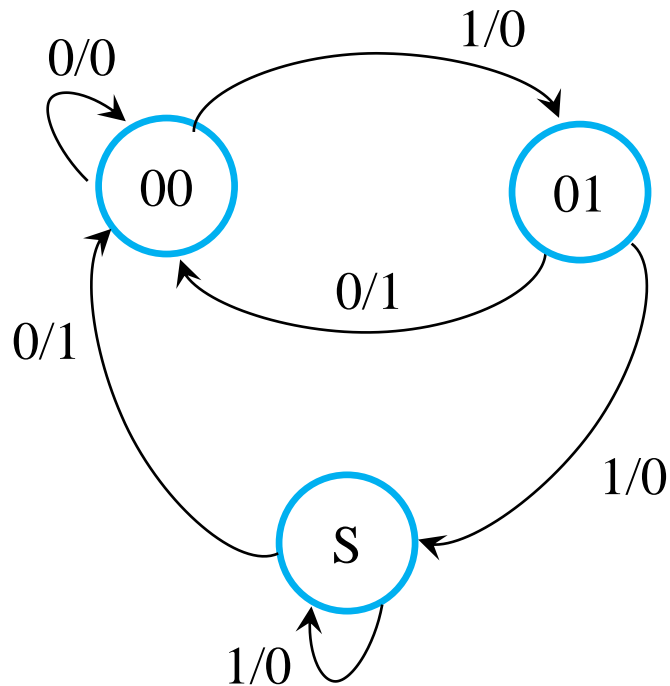
- Due stati sono tra loro equivalenti se, a partire da quegli stati, l'applicazione di ogni possibile ingresso produce le stesse uscite e porta a transizioni nello stesso (o equivalente) stato futuro



- Gli stati '11' e '10' sono equivalenti
  - Applicando l'input '1', da entrambi gli stati viene prodotta l'uscita '0' e si transita nello stato '10'
  - Applicando l'input '0', da entrambi gli stati viene prodotta l'uscita '1' e si transita nello stato '00'

# Stati equivalenti

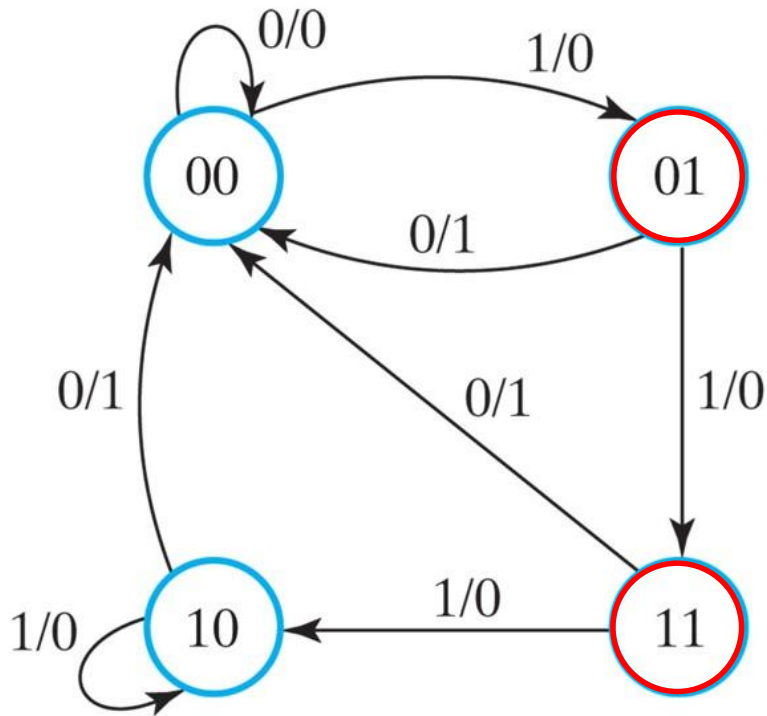
- Due stati sono tra loro equivalenti se, a partire da quegli stati, l'applicazione di ogni possibile ingresso produce le stesse uscite e porta a transizioni nello stesso (o equivalente) stato futuro



- Quindi possiamo realizzare un **circuito equivalente**, combinando i due stati equivalenti in un unico stato S
- Nota: l'equivalenza si poteva anche vedere dalla tabella degli stati

# Stati equivalenti

- Due stati sono tra loro equivalenti se, a partire da quegli stati, l'applicazione di ogni possibile ingresso produce le stesse uscite e porta a transizioni nello stesso (o equivalente) stato futuro

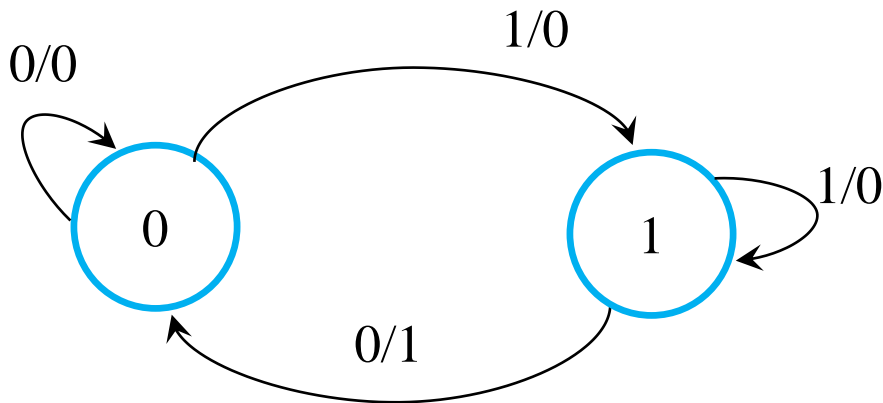


- Ancora, tornando al circuito iniziale:
- Gli stati '01' e '11' sono equivalenti
  - Applicando l'input '1', si transita rispettivamente nello stato '11' e '10' (che abbiamo visto essere equivalenti) e viene prodotta l'uscita '0'
  - Applicando l'input '0', da entrambi gli stati si transita nello stato '00' e viene prodotta l'uscita '1'



# Stati equivalenti

- Due stati sono tra loro equivalenti se, a partire da quegli stati, l'applicazione di ogni possibile ingresso produce le stesse uscite e porta a transizioni nello stesso (o equivalente) stato futuro



- Quindi possiamo realizzare un circuito equivalente, che comprende 2 stati (= 1 flip-flop), invece di 4 stati (=2 flip-flop)

- L'identificazione di stati equivalenti può essere **utile per ridurre il numero degli stati di un circuito** (quindi il numero di flip-flop)
- Tuttavia, non è detto che questa semplificazione porti ad un costo minore (il costo del circuito sequenziale dipende anche dalla parte combinatoria, oltre che dai flip-flop)

# Modello di Mealy e di Moore

# Modello di Mealy e di Moore

- In un circuito sequenziale di tipo **Mealy** le **uscite dipendono sia dagli ingressi che dallo stato corrente** del sistema
  - L'esempio appena visto è di tipo Mealy
- In un circuito sequenziale di tipo **Moore** le **uscite dipendono SOLO dallo stato corrente** del sistema
  - Le uscite sono indicate all'interno di ciascuno stato, invece che vicino a ciascuna transizione
- Un circuito di Mealy in generale può
  - avere un numero minore di stati
  - contenere meno logica ed essere più veloce (reagisce più velocemente ad un cambio degli ingressi)

# Es: Circuito di tipo Moore

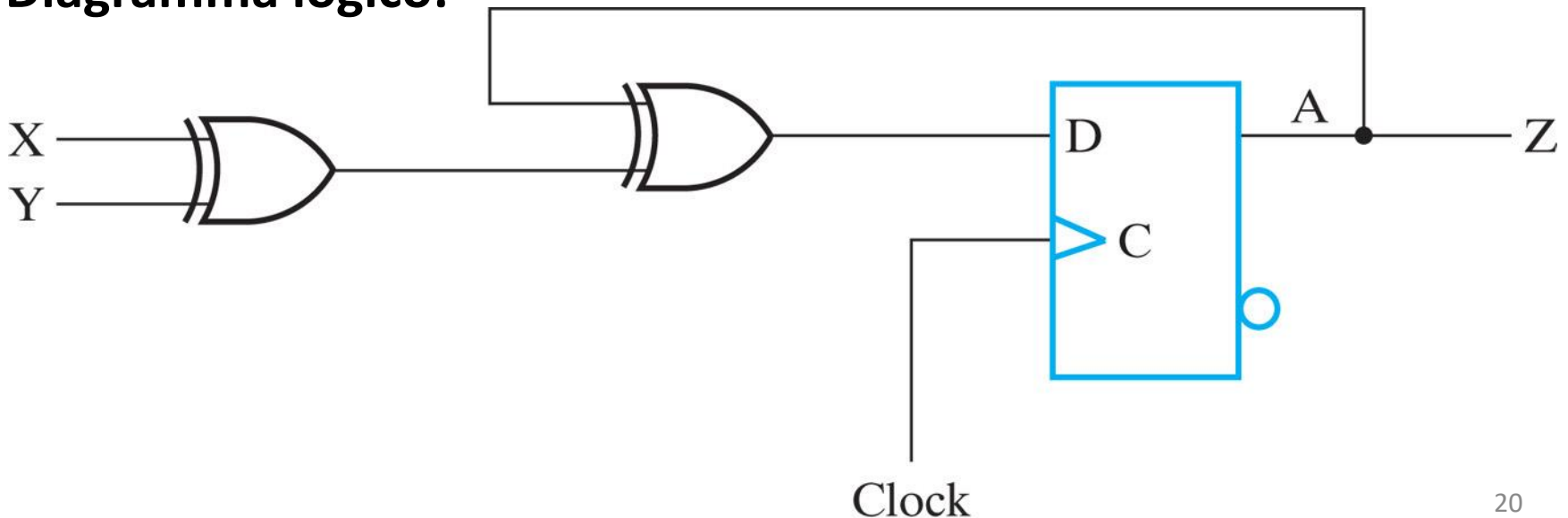
- Trovare il diagramma logico, la tabella degli stati e il diagramma degli stati relativi al circuito specificato dall'equazione di ingresso:

$$D_A = A \oplus X \oplus Y$$

e dall'equazione di uscita:

$$Z = A$$

**Diagramma logico:**



# Es: Circuito di tipo Moore

- Trovare il diagramma logico, la tabella degli stati e il diagramma degli stati relativi al circuito specificato dall'equazione di ingresso:

$$D_A = A \oplus X \oplus Y$$

e dall'equazione di uscita:

$$Z = A$$

**Tabella degli stati:**

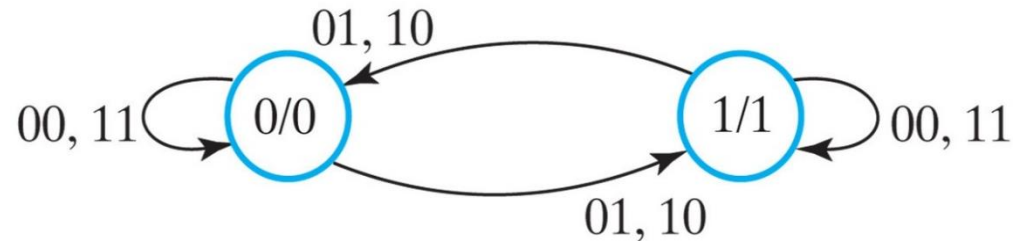
Present State	Inputs		Next State	Output
A	X	Y	A	Z
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	0
1	0	0	1	1
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

# Es: Circuito di tipo Moore

**Tabella degli stati:**

Present State	Inputs		Next State	Output
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

**Diagramma degli stati:**



- L'uscita è indicata all'interno di ciascuno stato (non dipende dall'ingresso) anziché vicino alle frecce delle transizioni
- Ogni transizione è rappresentata da una freccia che parte da uno stato e arriva ad un altro stato, in corrispondenza ad ogni ingresso

# Simulazione di un circuito sequenziale

# Simulazione di un circuito sequenziale

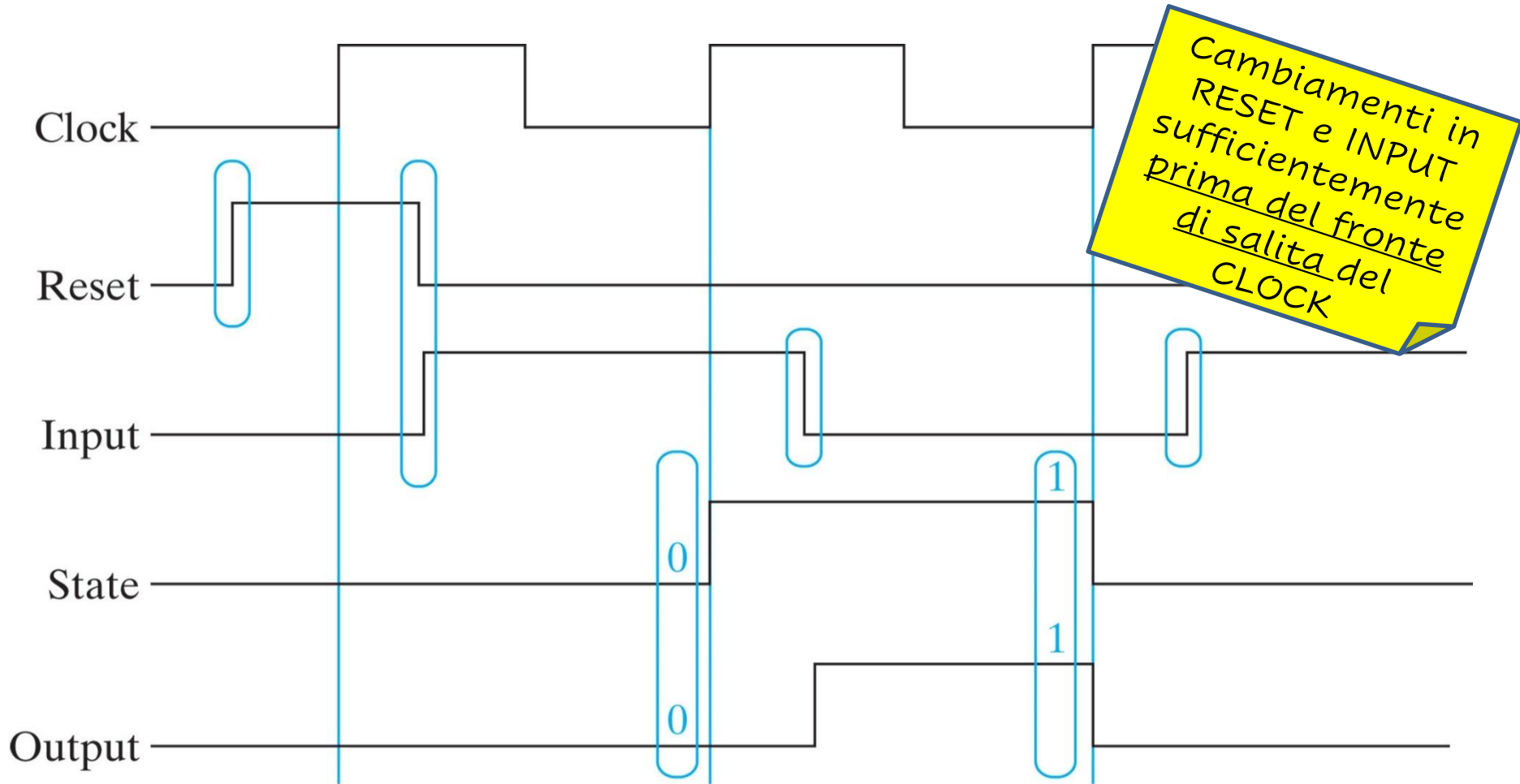
- Emergono nuove questioni rispetto ai circuiti combinatori
  - i. Il comportamento del circuito dipende dall'**ordine in cui sono applicati gli ingressi**, anche in relazione alle variazioni del clock
  - ii. Bisogna partire da uno scenario in cui il circuito si trova in uno stato noto, perciò è necessario inizializzare il circuito (fornire un **segnale di reset**) all'inizio della simulazione
  - iii. Per verificare il comportamento del circuito dobbiamo **applicare determinate sequenze di ingresso**, temporizzate correttamente rispetto al segnale di clock e **osservare le uscite a istanti opportuni rispetto al clock**
- La simulazione di un circuito sequenziale può essere di tipo
  - **funzionale** (gli elementi del circuito sono ideali, con ritardo nullo o molto piccolo): per la verifica della corretta funzionalità del circuito
  - **temporale** (gli elementi del circuito hanno tempi di ritardo realistici): per la verifica della corretta temporizzazione del circuito



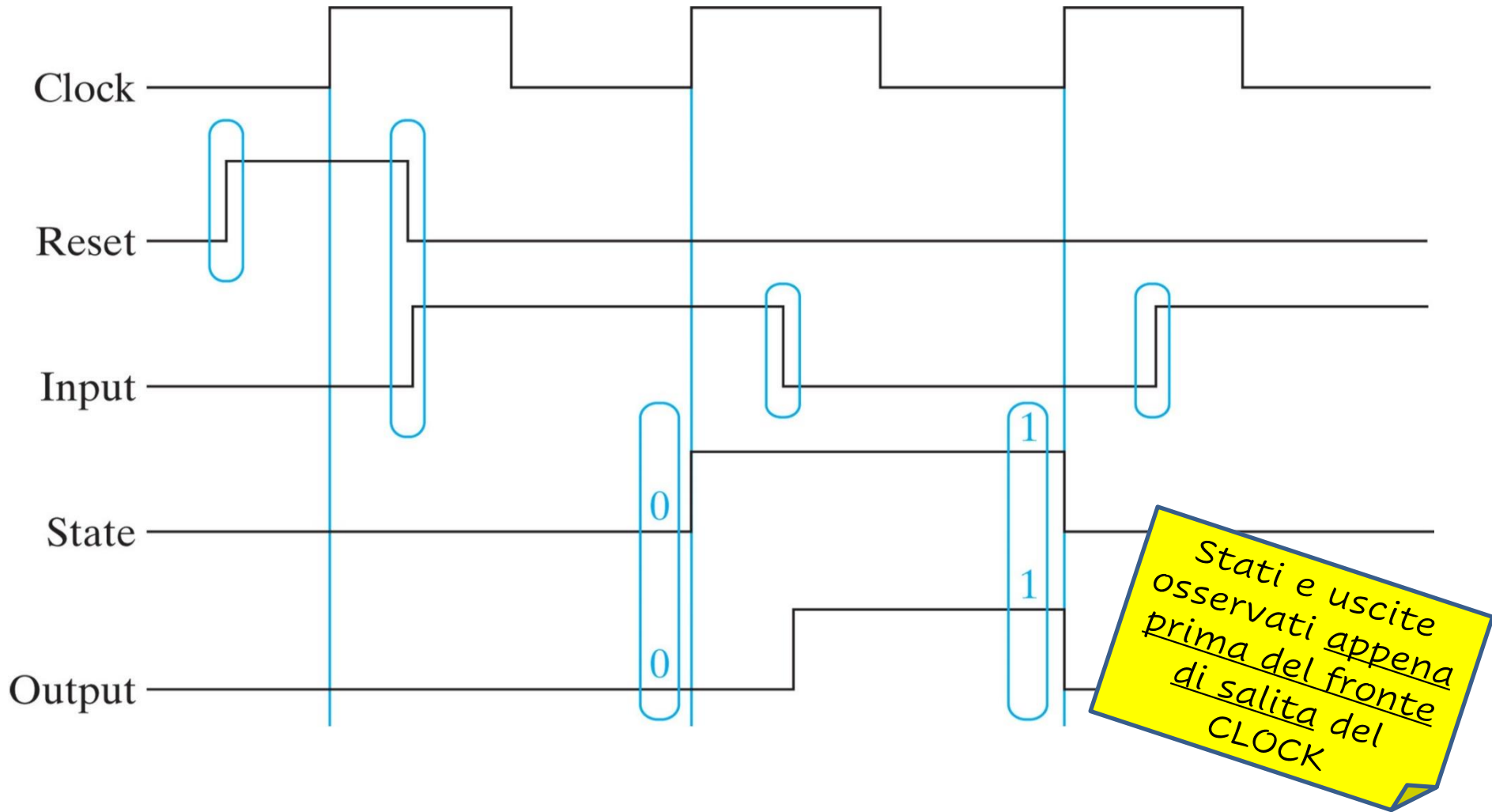
# Simulazione di un circuito sequenziale

- Per una corretta simulazione è importante
  - Se vengono usati piccoli tempi di ritardo (anche in simulazioni funzionali), accertarsi che il **periodo di clock sia sufficientemente lungo** da eccedere il ritardo della logica combinatoria nel caso peggiore
  - **Applicare gli ingressi ad un tempo sufficientemente anteriore rispetto al fronte di salita del clock**, e.g. al fronte di discesa: la logica combinatoria ha tempo di calcolare le uscite e lo stato futuro fino al successivo fronte di salita
  - Esaminare il risultato della simulazione quando le variabili di stato e le uscite hanno raggiunto i loro valori finali, e.g. appena prima del fronte di salita del clock

# Diagramma temporale



# Diagramma temporale



# Disclaimer

Figures from *Logic and Computer Design Fundamentals*,  
Fifth Edition, GE Mano | Kime | Martin

© 2016 Pearson Education, Ltd