

Capitolo 5

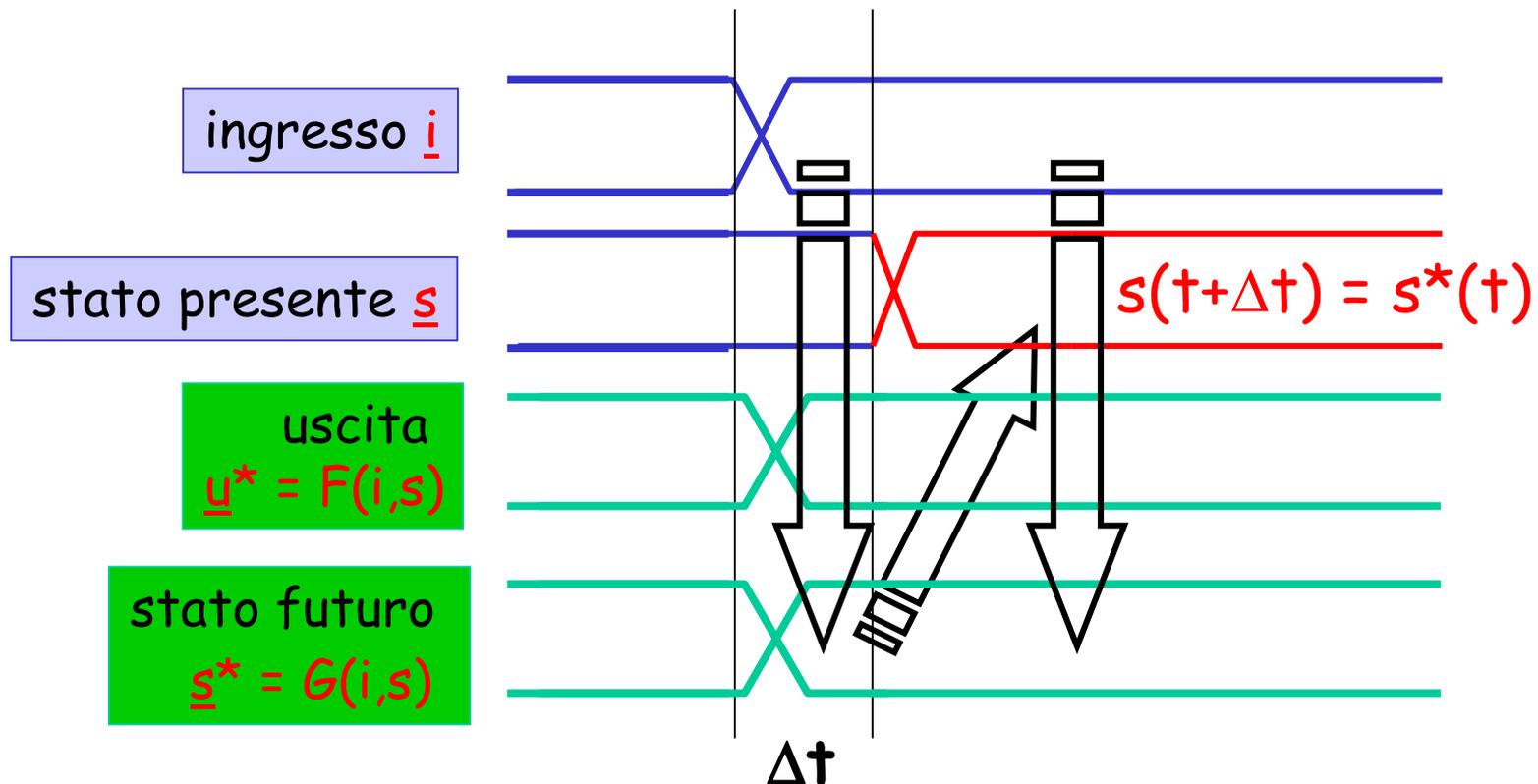
Reti Sequenziali Asincrone

Reti Logiche T

5.1 Struttura e comportamento

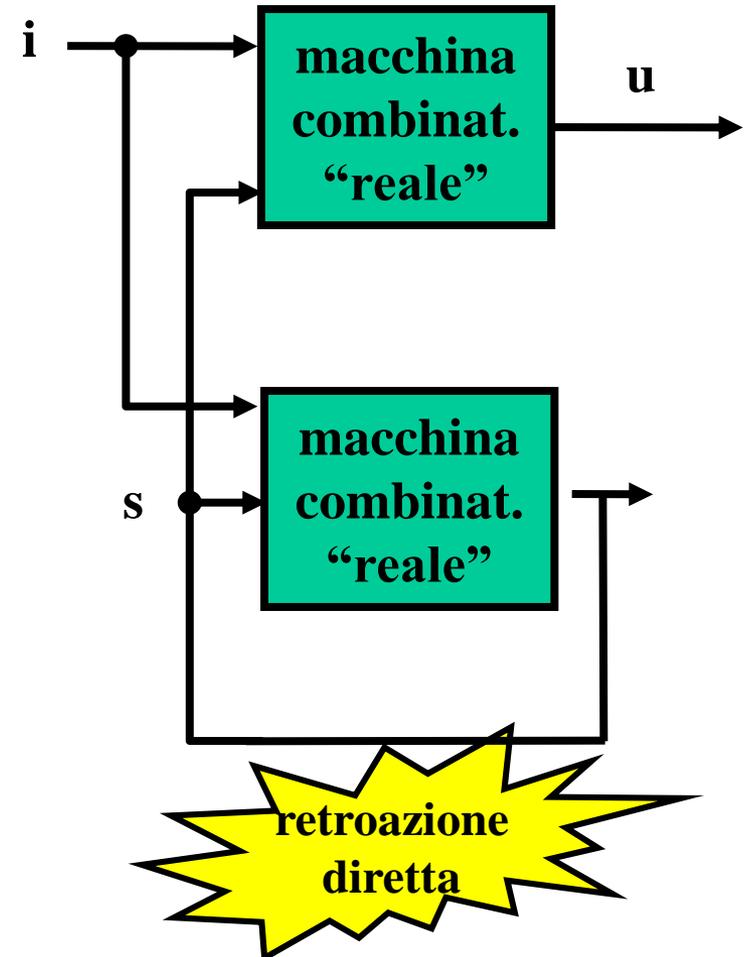
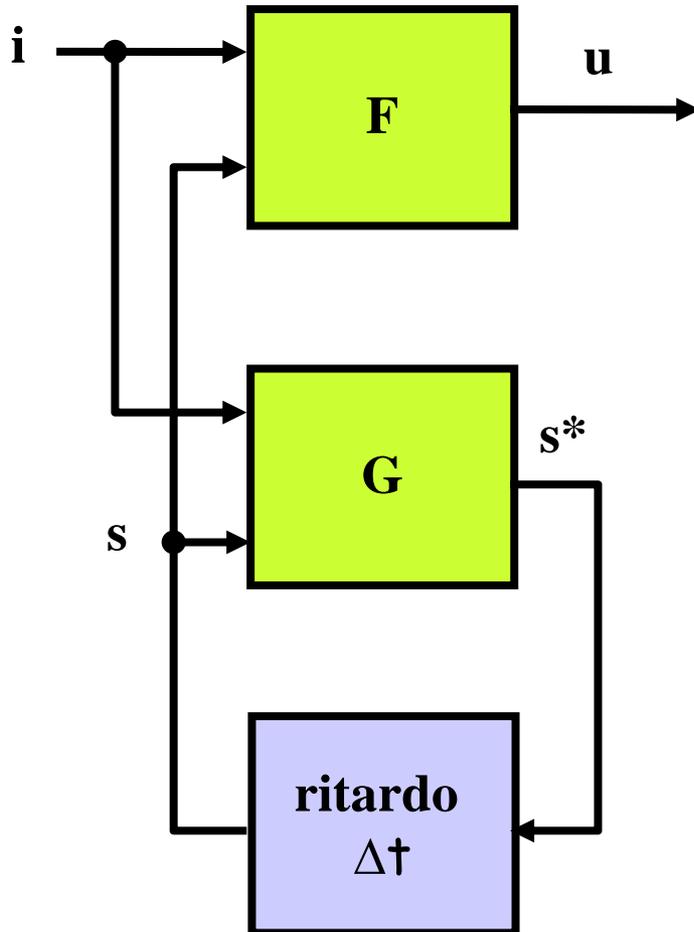
Reti sequenziali asincrone (comportamento)

- **Elaborazione asincrona:** ogni nuovo ingresso determina:
 - una condizione di stabilità dello stato interno
 - non più di una modifica del simbolo d'uscita
- Un simbolo d'uscita si può modificare solo in presenza di una modifica dell'ingresso (*inseguimento dell'ingresso*)
- Per un tempo Δt lo stato futuro è diverso dallo stato presente, dopodichè c'è l'aggiornamento dello stato presente ed il nuovo stato diventa **stabile**



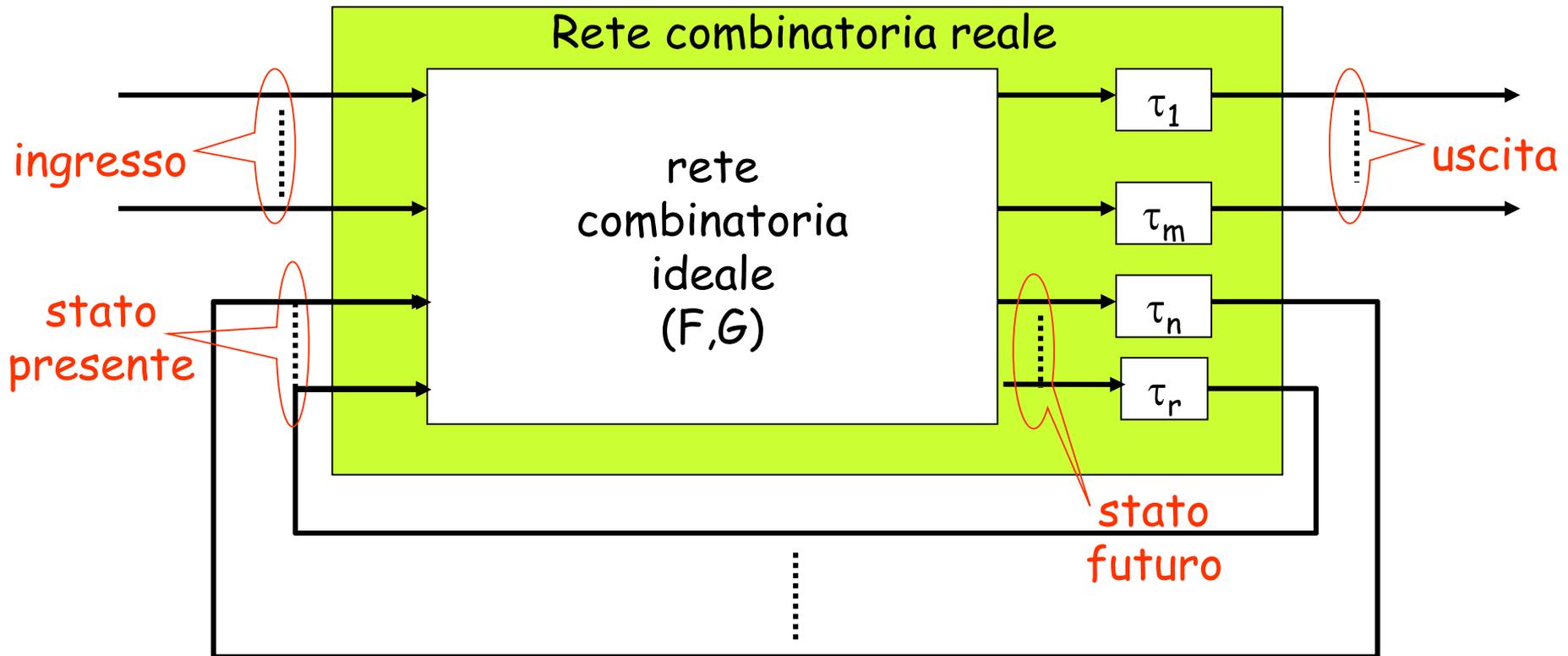
Reti sequenziali asincrone (struttura)

- Il ritardo Δt è dunque necessario per il funzionamento di una RSA
- In condizioni reali, il ritardo in retroazione della rete asincrona è realizzato dalla presenza di ritardi sulle uscite della rete combinatoria che realizza la funzione G
- È dunque possibile collegare in **retroazione diretta** i segnali di stato



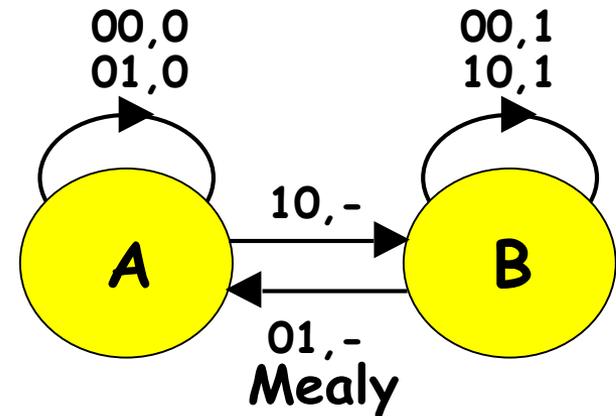
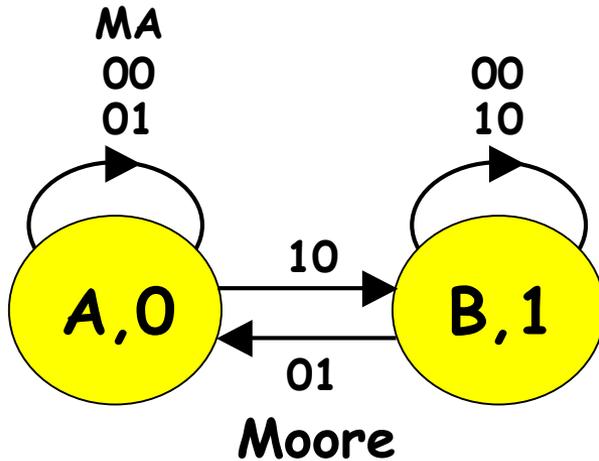
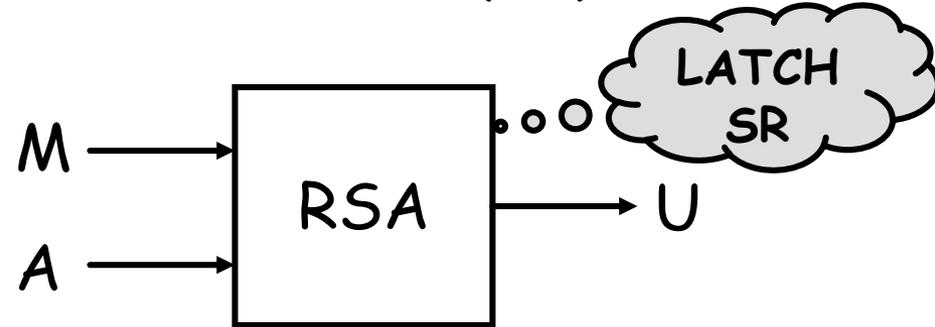
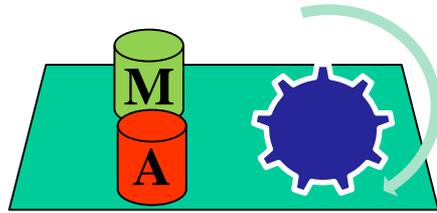
Reti sequenziali asincrone (struttura)

- n segnali in ingresso: 2^n possibili **configurazioni degli ingressi**
- k segnali in retroazione: 2^k **stati interni**
- Il **ritardo intrinseco** della rete combinatoria agisce da "memoria temporanea":
 - lo stato presente non cambia durante il calcolo dello stato futuro.
 - il piccolo valore dei ritardi consente di aggiornare lo stato ad ingresso costante
- È dunque bene che l'ingresso sia modificato solo in condizioni di **stabilità** dello stato interno -> **modifica di un segnale d'ingresso alla volta**



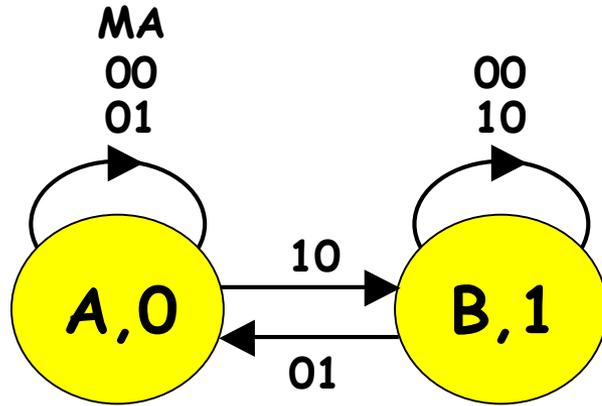
Esempio: Motore con marcia/arresto

- Una rete sequenziale asincrona è formata da due ingressi collegati a due tasti (M, marcia, e A, arresto) che comandano l'attivazione ($U=1$) e l'arresto ($U=0$) di un motore elettrico

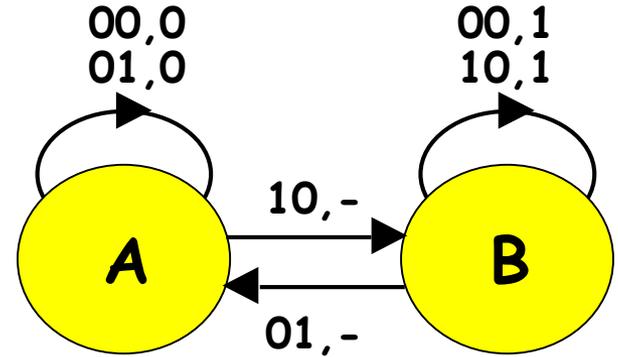


- L'intera sequenza d'ingresso alla rete precedente allo stato attuale viene sintetizzata in **due soli stati**
- Nel modello di Mealy, uso le condizioni di indifferenza sulle **uscite**: non vincolo la rete durante le transizioni di stato in cui l'uscita cambia di valore
- La rete **memorizza** l'ultimo pulsante premuto per identificare l'uscita corrente: tale rete può essere dunque interpretata come *elemento di memoria*

Dal grafo degli stati alla tabella di flusso



Moore



Mealy

		MA				U
s.p.		00	01	11	10	
A	(A)	(A)	-	(B)	0	
B	(B)	(A)	-	(B)	1	

s.f.

		MA				U
s.p.		00	01	11	10	
A	(A,0)	(A,0)	-, -	(B, -)	0	
B	(B,1)	(A, -)	-, -	(B,1)		

s.f.,U

• La pressione contemporanea di entrambi i tasti non rientra nelle normali condizioni di lavoro: nella TdF, con ingresso MA=11, metto delle indifferenze su stato futuro (e uscite nel caso di Mealy)

Automati equivalenti

La descrizione con un automa di un comportamento sequenziale **non è unica**

- **Automati equivalenti:** automi che descrivono lo stesso comportamento con **differente numero di stati interni**
- Ciò comporta, in generale, di avere grafi degli stati differenti per il medesimo comportamento
- **Automa minimo:** automa equivalente avente il **minor numero** di stati interni
- **Come individuare l'automa minimo?**

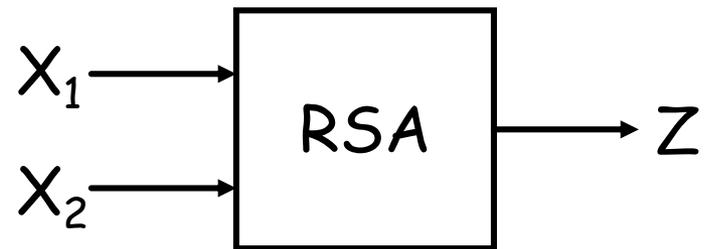
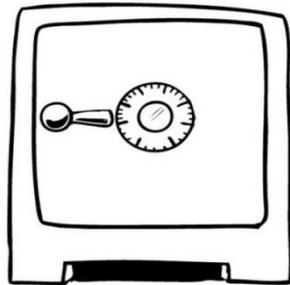
Grafo primitivo

Grafo degli stati primitivo - Grafo in cui ogni stato è stabile per una ed una sola configurazione d'ingresso.

Per individuare le esigenze di stati interni poste dalla specifica di comportamento è spesso utile iniziare il progetto con un **grafo primitivo**.

- Di norma il grafo primitivo **non ha il minimo numero possibile di stati interni**: non corrisponde dunque sempre all'automa minimo
- Dalla **tabella di flusso primitiva** (una sola stabilità per riga), si possono individuare coppie di stati **compatibili**:
 - due stati sono compatibili se e solo se
 1. le uscite sono uguali per tutti i simboli d'ingresso
 2. gli stati futuri sono identici o tra loro compatibili per tutti i simboli d'ingresso
- .. fino a ottenere un grafo **non riducibile** (ovvero, avente stati interni tra loro tutti incompatibili)

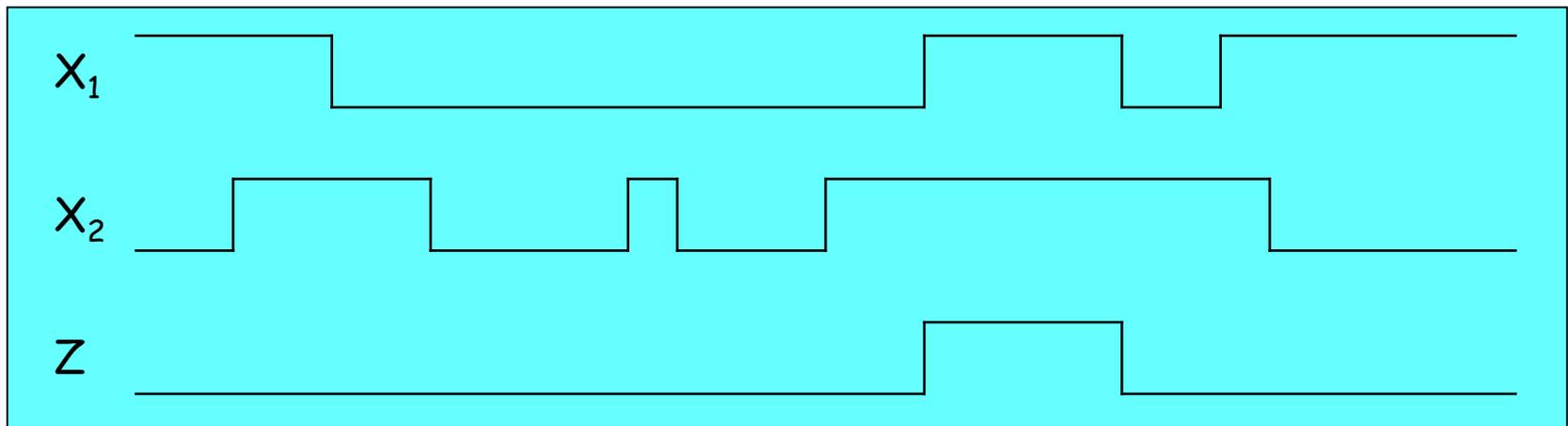
Riconoscitore di sequenze



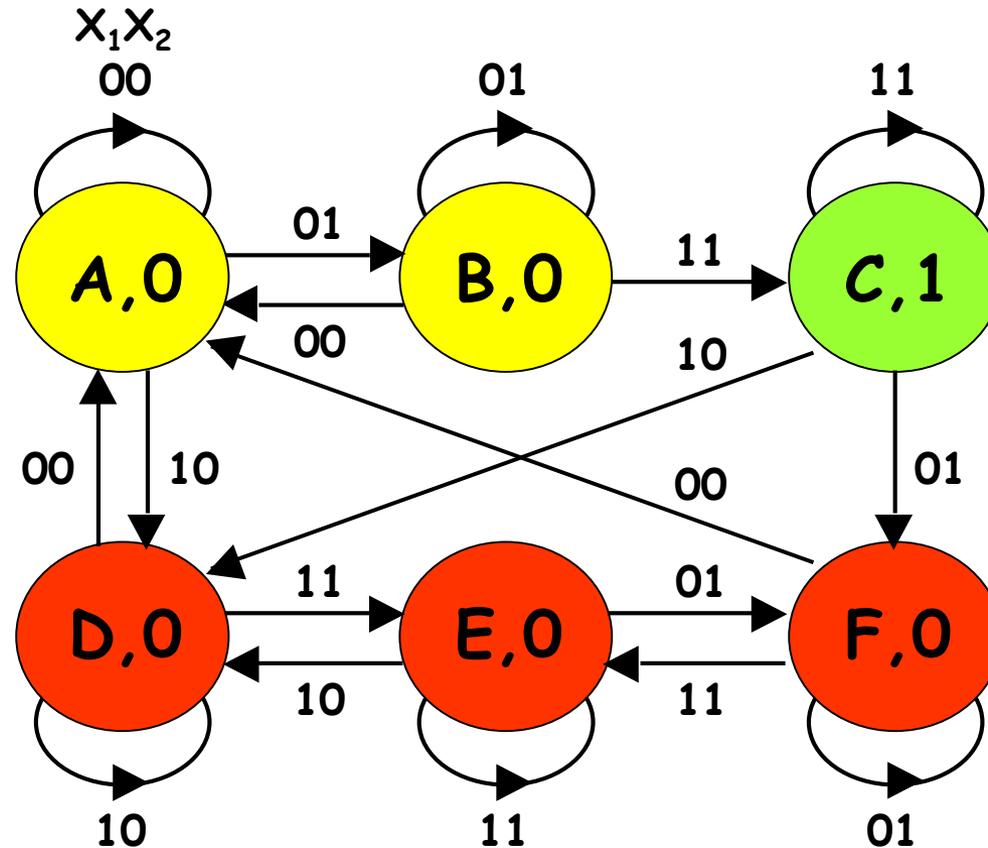
- Rete a 2 ingressi (X_1 , X_2) e 1 uscita (Z)
- Comportamento:
 - $Z = 1$ in corrispondenza dell'ultimo simbolo della sequenza di ingresso

$$X_1 X_2 = 00-01-11.$$

- (I segnali di ingresso non cambiano mai contemporaneamente.)



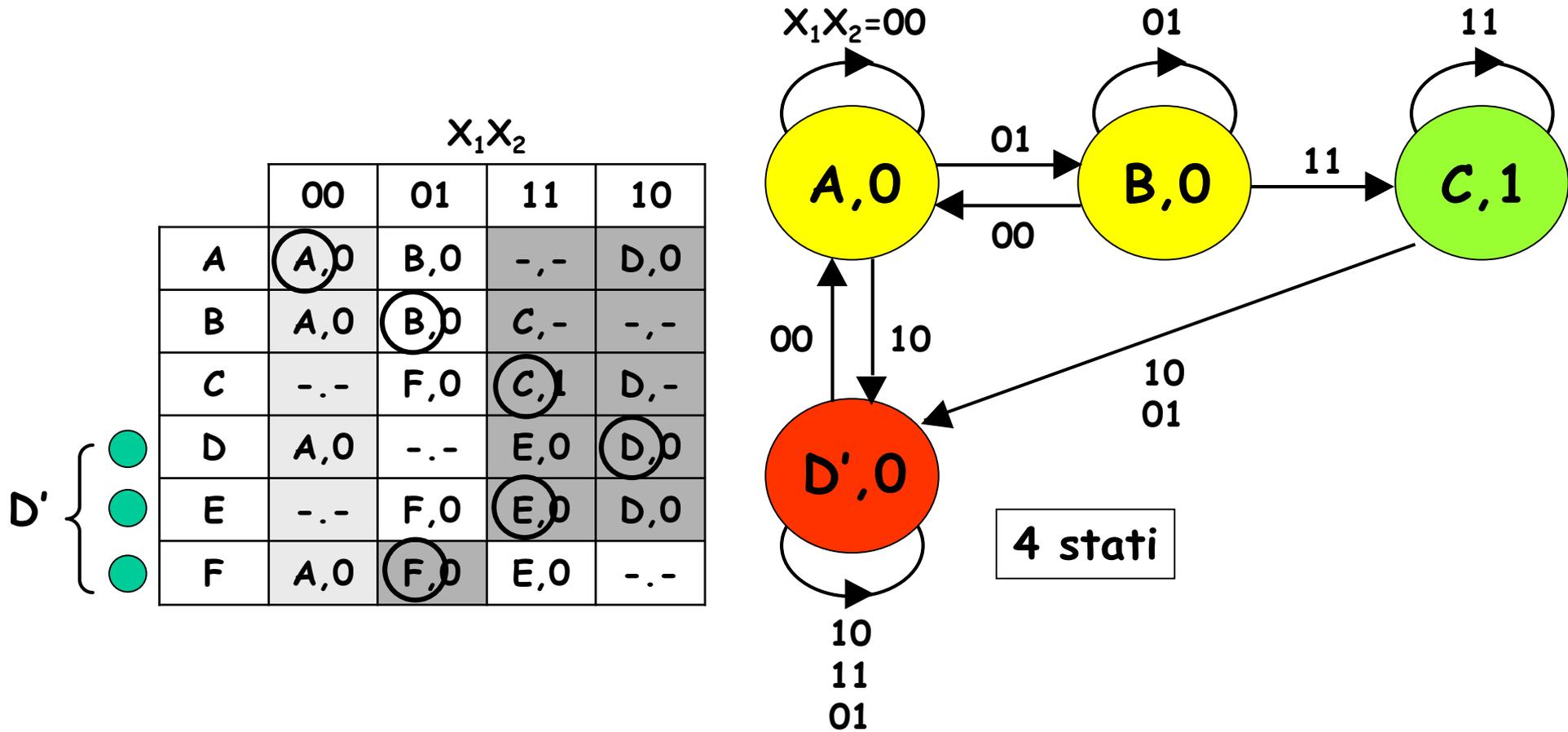
Grafo degli stati primitivo (modello di Moore)



6 stati

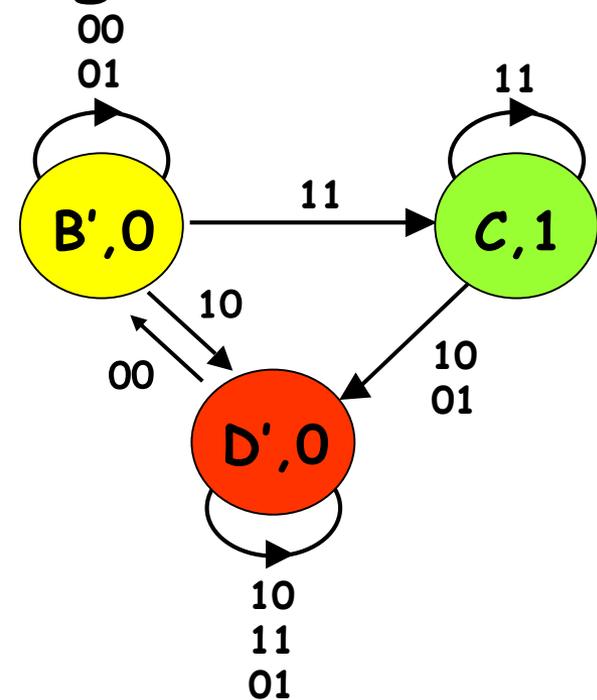
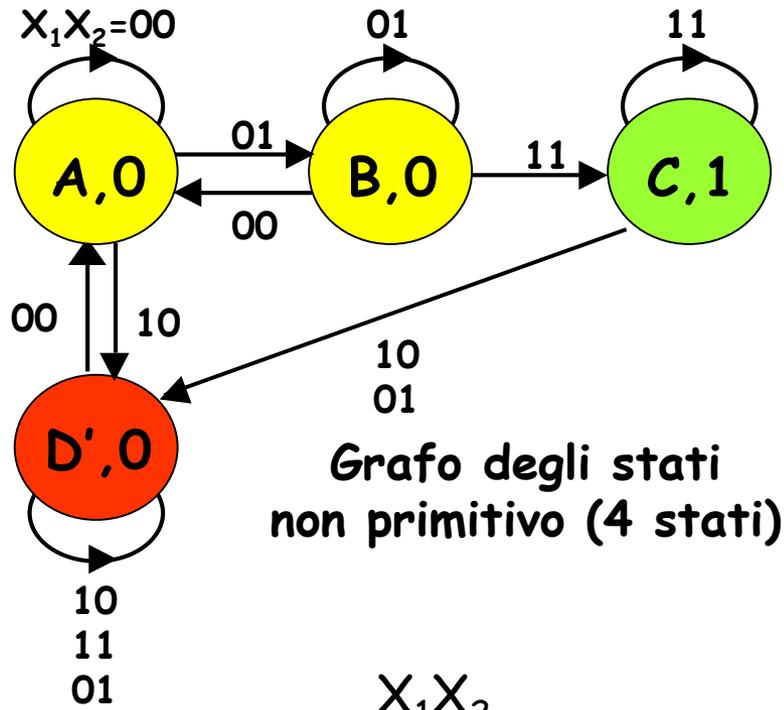
- Ogni stato memorizza una particolare **condizione di lavoro** della rete:
 - A: riconosciuto il primo simbolo di una potenziale sequenza corretta
 - B: riconosciuto il secondo simbolo di una potenziale sequenza corretta
 - C: riconosciuta la sequenza corretta (unico caso con $Z=1$)
 - D,E,F: nessun simbolo riconosciuto
- Ogni stato è stabile per un'unica **configurazione d'ingresso** (grafo primitivo)
- Ogni stato ha una transizione per configurazioni di ingresso **adiacenti** a quella stabile (può cambiare un solo segnale d'ingresso alla volta)

Grafo degli stati non primitivo



- Gli stati D,E,F sono tra loro compatibili a due a due e possono essere raggruppati nel medesimo stato, riducendo il numero di stati totali
- **Grafo non più primitivo:** D' ora è stabile per più di una configurazione d'ingressi
- A partire da una stessa descrizione, esistono **diversi automi** in grado di realizzarla (**automi equivalenti**)

Una riduzione ulteriore degli stati



X_1X_2

	00	01	11	10
A	A,0	B,0	-, -	D',0
B	A,0	B,0	C,-	-, -
C	-, -	D',-	C,1	D',-
D'	A,0	D',0	D',0	D',0

s.f., Z

Tabella di flusso (modello di Mealy)

$\{A,B\} \equiv$

X_1X_2

	00	01	11	10
B'	B',0	B',0	C,-	D',0
C	-, -	D',-	C,1	D',-
D'	B',0	D',0	D',0	D',0

s.f., Z

Tabella di flusso minima (3 stati)

Metodi di minimizzazione degli stati

- Il grafo a tre stati ottenuto nell'esempio precedente è **l'automa minimo**? O è possibile effettuare una ulteriore riduzione?
- Esistono metodi formali di «minimizzazione degli stati», che andremo ora ad analizzare, che aiutano il progettista nell'operazione di riduzione del numero degli stati ai fini di ottenere l'automa minimo

5.2 Minimizzazione degli stati

Minimizzazione degli stati di reti sequenziali asincrone (RSA) / sincrone (RSS)

Problema:

Data una tabella di flusso (TdF) contraddistinta da un numero arbitrario N di stati s_1, s_2, \dots, s_N , individuare la (o una) TdF ad essa equivalente contraddistinta dal minimo numero possibile n di stati s'_1, s'_2, \dots, s'_n (TdF minima).

1° caso (tipicamente RSS):

TdF originaria completamente specificata

Algoritmo di riduzione basato sulla relazione di indistinguibilità tra stati

TdF minima **unica**

2° caso (tipicamente RSA):

TdF originaria non completamente specificata

Algoritmo di riduzione basato sulla relazione di compatibilità tra stati

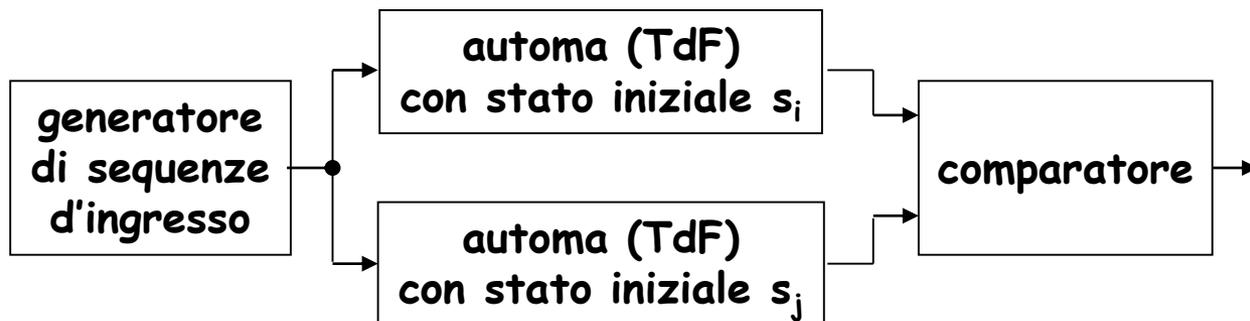
TdF minima **in generale non unica**

1° caso (TdF completamente specificata → relazione di indistinguibilità)

Due stati s_i, s_j sono indistinguibili ($s_i \equiv s_j$) se

1ª definizione

assunti come stati iniziali, danno luogo a sequenze di configurazioni d'uscita identiche per qualunque sequenza di configurazioni d'ingresso



2ª definizione

per qualunque configurazione di ingresso γ

1) le corrispondenti configurazioni di uscita coincidono

$$F(s_i, \gamma) = F(s_j, \gamma)$$

2) gli stati futuri corrispondenti sono indistinguibili

$$G(s_i, \gamma) \equiv G(s_j, \gamma)$$

La relazione di indistinguibilità è una **relazione di equivalenza**, valendo per essa le proprietà

a) riflessiva: $s_i \equiv s_i$

b) simmetrica: $s_i \equiv s_j \rightarrow s_j \equiv s_i$

c) transitiva: $s_i \equiv s_j, s_j \equiv s_k \rightarrow s_i \equiv s_k$

La relazione di indistinguibilità induce sull'insieme degli stati di TdF una partizione in sottoinsiemi disgiunti (**classi di indistinguibilità**).

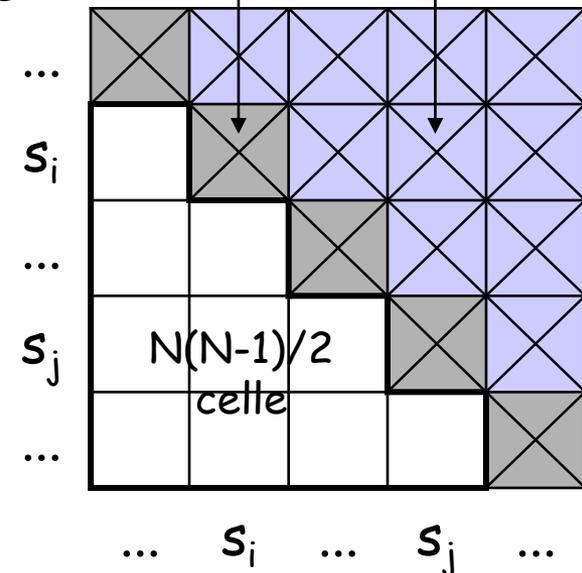
La TdF minima ha tante stati (righe) quante sono le classi di indistinguibilità.

Algoritmo di riduzione:

Individuazione delle coppie di stati indistinguibili
(2^a definizione)

Individuazione delle classi di indistinguibilità

Costruzione della TdF minima



2° caso (TdF non completamente specificata → relazione di compatibilità)

Due stati s_i, s_j sono **compatibili** ($s_i \sim s_j$) se

1ª definizione

assunti come stati iniziali, danno luogo a sequenze di configurazioni d'uscita identiche, a meno di condizioni di indifferenza, per qualunque sequenza di configurazioni d'ingresso

2ª definizione

per qualunque configurazione di ingresso γ

1) le corrispondenti configurazioni di uscita, se entrambe specificate, coincidono

$$F(s_i, \gamma) = F(s_j, \gamma)$$

2) i corrispondenti stati futuri, se entrambi specificati, sono compatibili

$$G(s_i, \gamma) \sim G(s_j, \gamma)$$

La relazione di compatibilità non è una relazione di equivalenza, poiché per essa **non vale la proprietà transitiva**

a) riflessiva: $s_i \sim s_i$

b) simmetrica: $s_i \sim s_j \rightarrow s_j \sim s_i$

c) transitiva: $s_i \sim s_j, s_j \sim s_k \rightarrow /s_i \sim s_k$

Le classi massime di compatibilità (CMC, sottoinsiemi di stati di TdF a due a due compatibili) **non sono necessariamente disgiunte.**

Indistinguibilità vs. Compatibilità

- Secondo la definizione fornita nelle slide precedenti, ai fini della riduzione degli stati in un automa *non completamente specificato* (ovvero che presenta condizioni di indifferenza su uscite o stati futuri), due stati si dicono **compatibili** se, per gli stessi ingressi, hanno le stesse uscite e portano agli stessi stati o a stati fra loro compatibili **laddove specificato** (ovvero, a meno di condizioni di indifferenza).
- La compatibilità tra stati diventa **equivalenza** nel caso di automi completamente specificati (*stati indistinguibili*)
- La compatibilità è una relazione meno forte di quella di indistinguibilità: non vale la **proprietà transitiva**

	0	1
A	C,1	A,1
B	C,1	A,1
C	C,1	A,1

Se (A,B) e (B,C) sono
tra loro equivalenti
allora
A,C sono equivalenti

Automa
completamente specificato

	0	1
A	C,1	A,-
B	C,-	A,1
C	C,0	A,-

A,B compatibili
B,C compatibili

MA
A,C incompatibili

Automa non
completamente specificato

Algoritmo di riduzione - TdF non completamente specificata :

Individuazione delle coppie di stati compatibili (2^a definizione)

Individuazione delle **classi massime di compatibilità (CMC)**

Selezione dell'insieme S di minima cardinalità di classi di compatibilità, non necessariamente massime, tali da soddisfare le condizioni di

- **copertura:** l'unione delle classi in S comprende tutti gli stati di TdF
- **chiusura:** per ogni ingresso, gli stati futuri corrispondenti agli stati di ogni classe in S appartengono tutti ad una classe in S

Il numero n di classi di compatibilità in S è compreso nel range $(n_{\min}, n_{\max}]$, essendo n_{\min} la cardinalità della più ampia classe massima di incompatibilità e n_{\max} il numero delle classi massime di compatibilità

Costruzione della tabella di flusso minima avente n stati (righe)

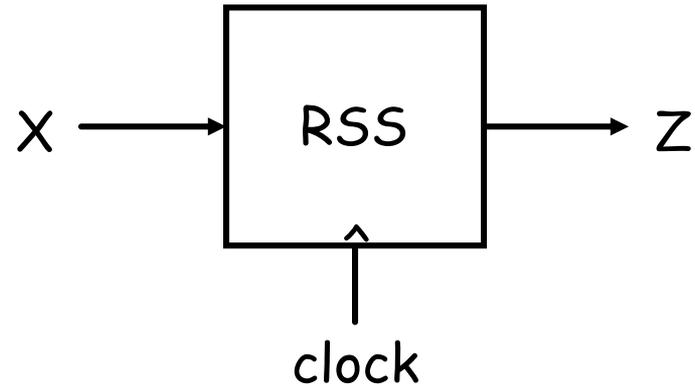
**N.B. - TdF non completamente specificate
a causa di vincoli su sequenze d'ingresso di lunghezza due:**

l'insieme di minima cardinalità di classi massime di compatibilità che soddisfa la condizione di copertura, soddisfa anche la condizione di chiusura (**Teorema di Unger**).

Il teorema di Unger assume particolare rilievo nel caso di **RSA**, dal momento che le indifferenze presenti nelle TdF di tali reti derivano tipicamente dall'ipotesi che i segnali di ingresso possono cambiare di valore uno solo alla volta (ovvero da sequenze di ingresso di lunghezza due inammissibili).

NOTA:

- L'automa minimo non necessariamente è costituito da tutte CMC
- Se si includono classi non massime, o se Unger non è applicabile (es., RSS non completamente specificata) si deve attentamente verificare copertura e chiusura



Comportamento:

in ogni intervallo di clock l'uscita Z deve assumere il valore logico 1 soltanto se gli ultimi quattro valori di X costituiscono una palindroma.

X	0	1	1	0	1	0	0	1	1	1	1	1	...
Z				1	0	0	0	1	0	0	1	1	...

Diagramma degli stati (modello di Mealy)

Occorre prevedere 8 distinti stati, ciascuno dei quali identifica una ben precisa configurazione dei valori assunti da X nei tre precedenti intervalli di clock.

Denotando gli stati direttamente con tali configurazioni binarie ed assumendo che in esse i bit siano ordinati in modo tale che l'ultimo a destra si riferisca all'ultimo valore assunto da X, è immediato costruire il diagramma degli stati:

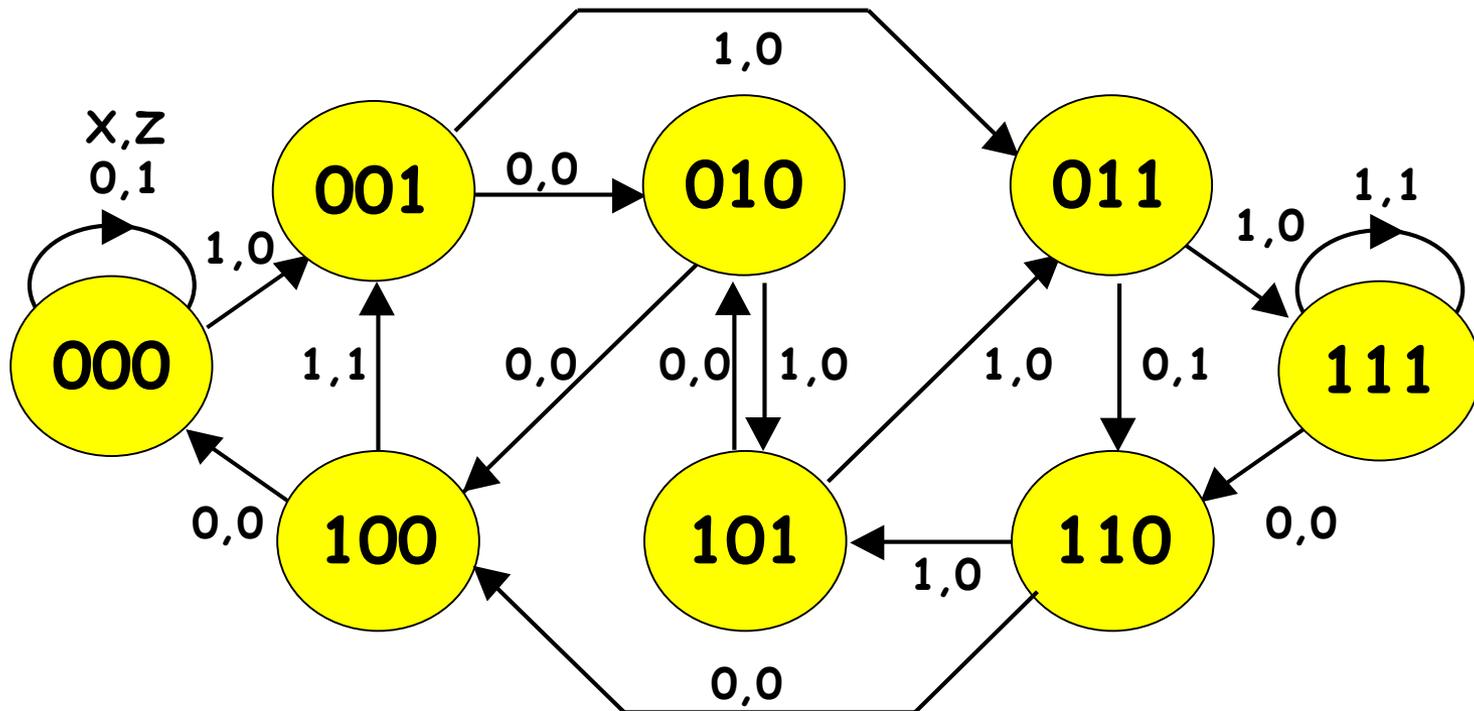


Tabella di flusso

X

	0	1
0	0,1	1,0
1	2,0	3,0
2	4,0	5,0
3	6,1	7,0
4	0,0	1,1
5	2,0	3,0
6	4,0	5,0
7	6,0	7,1

NOTA: RSS completamente specificata -> ricerca di relazioni di indistinguibilità

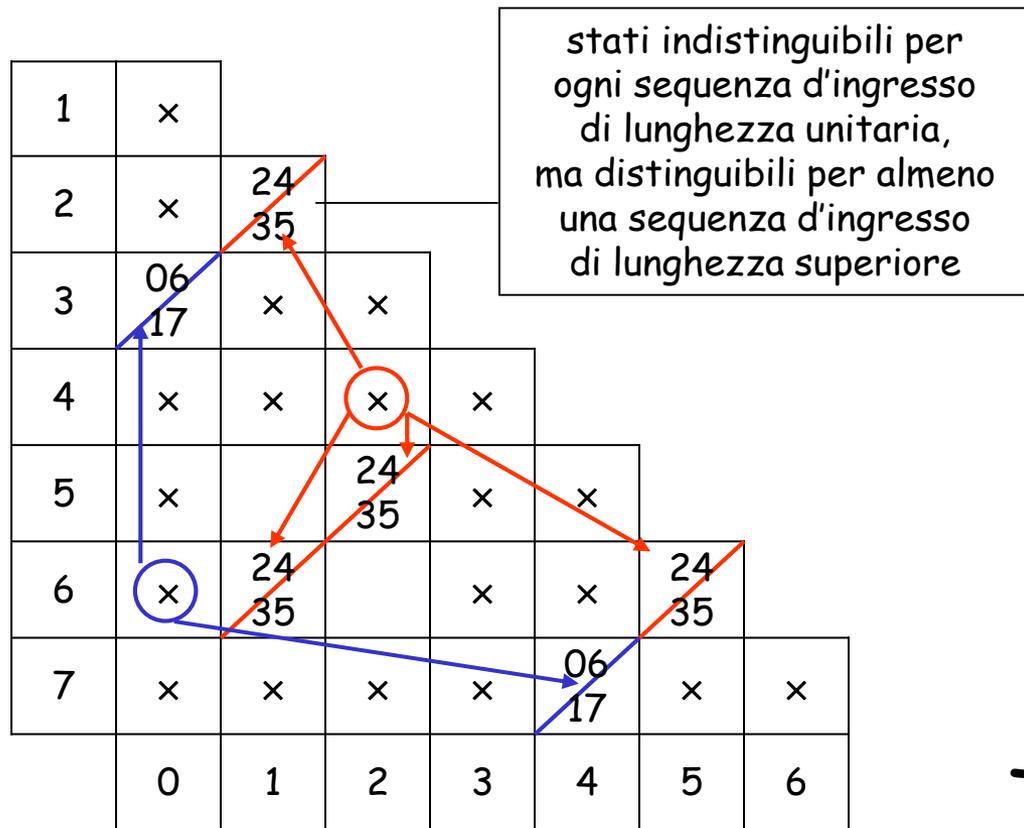
Tabella triangolare (caso complet. specificato)

1	x						
2	x	24 35					
3	06 17	x	x				
4	x	x	x	x			
5	x		24 35	x	x		
6	x	24 35		x	x	24 35	
7	x	x	x	x	06 17	x	x
	0	1	2	3	4	5	6

stati distinguibili per almeno una configurazione d'ingresso

stati indistinguibili per ogni configurazione d'ingresso, ma condizionatamente indistinguibili per sequenze d'ingresso di lunghezza non unitaria

stati indistinguibili per qualunque sequenza d'ingresso



X

	0	1
0	0,1	1,0
1	2,0	3,0
2	4,0	5,0
3	6,1	7,0
4	0,0	1,1
5	2,0	3,0
6	4,0	5,0
7	6,0	7,1

Classi di indistinguibilità:

{2,6}, {1,5}

{0}, {3}, {4}, {7}

N=8 → **n=6**

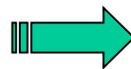
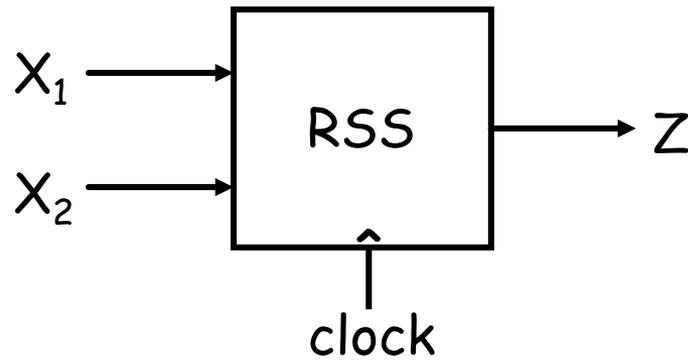


Tabella di flusso minima

X

	0	1	
0	0,1	1,0	
{1,5}	1	2,0	3,0
{2,6}	2	4,0	1,0
3	2,1	7,0	
4	0,0	1,1	
7	2,0	7,1	





Comportamento (tabella di flusso)

		X_1X_2			
		00	01	11	10
s.p.	1	2,0	-, -	3, -	2,0
	2	3,0	5,1	2,0	-, -
	3	3,0	4,1	-, -	5,0
	4	-, -	1,1	2, -	-, -
	5	-, -	-, -	1,1	-, -
		s.f., Z			

Tabella triangolare

2	23			
3	23 25	45		
4	23	15	14	
5	13	×		12
	1	2	3	4

Classi massime di **compatibilità** / **incompatibilità**

{1,2} {1,4} {2,3} {3,4,5} / {1,3} {1,5} {2,4} {2,5}

X_1X_2

	00	01	11	10
1	2,0	-, -	3, -	2,0
2	3,0	5,1	2,0	-, -
3	3,0	4,1	-, -	5,0
4	-, -	1,1	2, -	-, -
5	-, -	-, -	1,1	-, -

s.p.

s.f., Z

 $a \equiv \{1,2\}$ $b \equiv \{3,4,5\}$ $c \equiv \{1,4\}$ $d \equiv \{2,3\}$ 

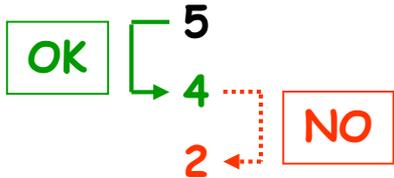
Tabella di flusso ridotta

 X_1X_2

	00	01	11	10
a	d,0	b,1	d,0	a/d,0
b	b/d,0	c,1	a,1	b,0
c	a/d,0	a/c,1	d, -	a/d,0
d	b/d,0	b,1	a/d,0	b,0

s.p.

s.f., Z

condizione di chiusura  $a \equiv \{1,2\}$ $b \equiv \{3,4,5\}$

condizione di copertura

 X_1X_2

	00	01	11	10
a	?,0	b,1	?,0	a,0
b	b,0	?,1	a,1	b,0

s.p.

s.f., Z

condizione di chiusura 

NOTA: Essendo una RSS non completamente specificata, non posso applicare Unger
 -> devo esplicitamente verificare **copertura** e **chiusura**

X_1X_2

	00	01	11	10
1	2,0	-, -	3, -	2,0
2	3,0	5,1	2,0	-, -
3	3,0	4,1	-, -	5,0
4	-, -	1,1	2, -	-, -
5	-, -	-, -	1,1	-, -

s.f., Z

 $a \equiv \{1,2\}$ $b \equiv \{3,4,5\}$ $c \equiv \{1,4\}$ $d \equiv \{2,3\}$ 

s.p.

	00	01	11	10
a	?,0	b,1	?,0	a,0
b	b,0	c,1	a,1	b,0
c	a,0	a/c,1	?, -	a,0

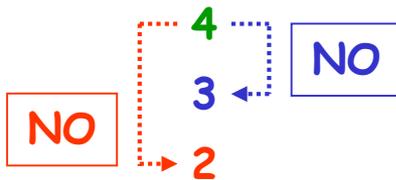
s.f., Z
 X_1X_2

	00	01	11	10
a	d,0	b,1	d,0	a/d,0
b	b/d,0	?,1	a,1	b,0
d	b/d,0	b,1	a/d,0	b,0

s.f., Z
 X_1X_2

	00	01	11	10
b	b/d,0	c,1	?,0	b,0
c	d,0	c,1	d, -	d,0
d	b/d,0	b,1	d,0	b,0

s.f., Z



s.p.



s.p.

Queste 3 combinazioni di CMC soddisfano la copertura, ma non la chiusura

Utilizzando le possibili combinazioni di CMC, non riesco a ottenere automi che soddisfano copertura e chiusura con meno di 4 stati

X_1X_2

	00	01	11	10
1	2,0	-, -	3, -	2,0
2	3,0	5,1	2,0	-, -
3	3,0	4,1	-, -	5,0
4	-, -	1,1	2, -	-, -
5	-, -	-, -	1,1	-, -

s.f., Z

 $a \equiv \{1,2\}$ $d \equiv \{2,3\}$ $e \equiv \{4,5\}$ 

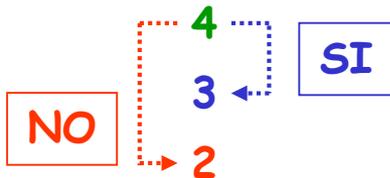
Tabella di flusso minima

 X_1X_2

	00	01	11	10
a	d,0	e,1	d,0	a/d,0
d	d,0	e,1	a/d,0	e,0
e	-, -	a,1	a,1	-, -

s.p.

s.f., Z



condizione di chiusura



Soddisfatta sia copertura, sia chiusura

L'automa minimo si ottiene in questo caso includendo classi di compatibilità che non sono CMC

Tabella triangolare per reti non c. specificate

- Ogni elemento indica:
 - incompatibilità:** la coppia ha uscite diverse (ove determinate) per la stessa configurazione di ingresso
 - compatibilità:** la coppia ha medesime uscite e medesimi stati futuri per ogni configurazione di ingresso
 - compatibilità condizionata:** stesse uscite per ogni ingresso ma diversi stati futuri, si indicano le coppie a cui si rimanda la verifica di compatibilità.
- Dopo aver riempito la tabella, si risolvono iterativamente le compatibilità condizionate: l'analisi termina quando non sono più possibili eliminazioni. Le coppie rimaste sono tra loro compatibili.

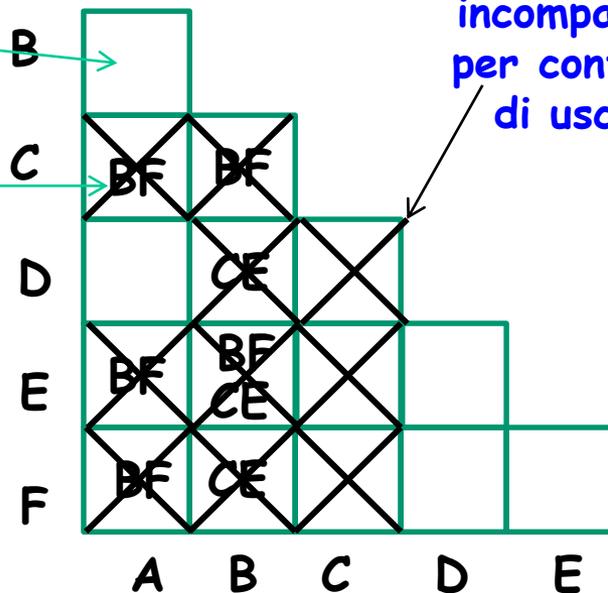
	X_1X_2			
	00	01	11	10
A	A,0	B,0	-, -	D,0
B	A,0	B,0	C,-	-, -
C	-, -	F,0	C,1	D,-
D	A,0	-, -	E,0	D,0
E	-, -	F,0	E,0	D,0
F	A,0	F,0	E,0	-, -

Riconoscitore di sequenza
(dal grafo primitivo)

A e B
compatibili
senza
condizioni

A e C
compatibili
se lo sono
B e F

C e D
incompatibili
per conflitto
di uscite



Riconoscitore di sequenze - reloaded

Grafo degli stati (modello di Moore)

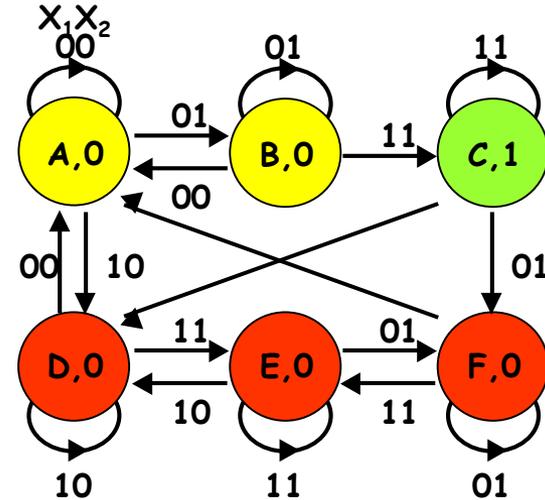


Tabella di flusso (modello di Mealy)

X_1X_2

	00	01	11	10
A	A,0	B,0	-, -	D,0
B	A,0	B,0	C, -	-, -
C	-, -	F, -	C,1	D, -
D	A,0	-, -	E,0	D,0
E	-, -	F,0	E,0	D,0
F	A,0	F,0	E,0	-, -

s.p.

s.f., Z

tabella triangolare

B					
C	BF	BF			
D		CE	CE		
E	BF	BF CE	BF CE		
F	BF	CE	CE		
	A	B	C	D	E

Classi massime di compatibilità: {A,B} {A,D} {C} {D,E,F} 6 → 4

Teorema di Unger: {A,B} ~~{A,D}~~ {C} {D,E,F}

condizione di copertura

4 → 3

condizione di chiusura



Tabella di flusso minima

X_1X_2

s.p.

	00	01	11	10
A	A,0	B,0	-, -	D,0
B	A,0	B,0	C, -	-, -
C	-, -	F, -	C,1	D, -
D	A,0	-, -	E,0	D,0
E	-, -	F,0	E,0	D,0
F	A,0	F,0	E,0	-, -

s.f., Z

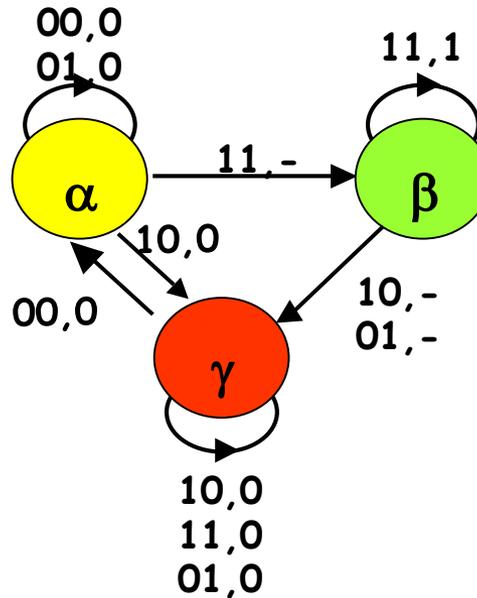
$\alpha \equiv \{A,B\}$ $\beta \equiv \{C\}$ $\gamma \equiv \{D,E,F\}$

X_1X_2

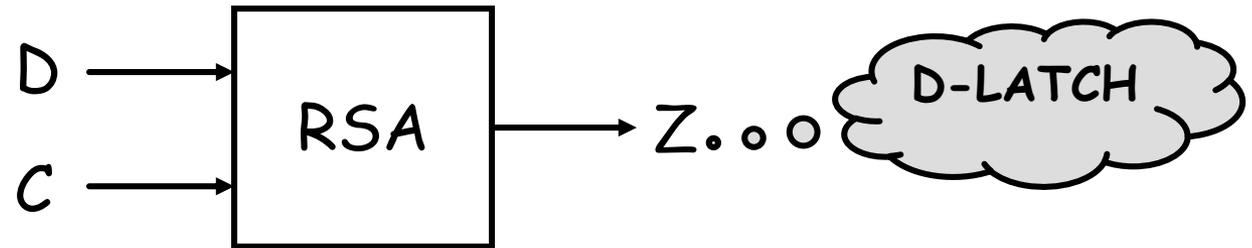
s.p.

	00	01	11	10
α	$\alpha,0$	$\alpha,0$	$\beta, -$	$\gamma,0$
β	-, -	$\gamma, -$	$\beta,1$	$\gamma, -$
γ	$\alpha,0$	$\gamma,0$	$\gamma,0$	$\gamma,0$

s.f., Z



Campionatore di segnale «level-triggered»



Comportamento:

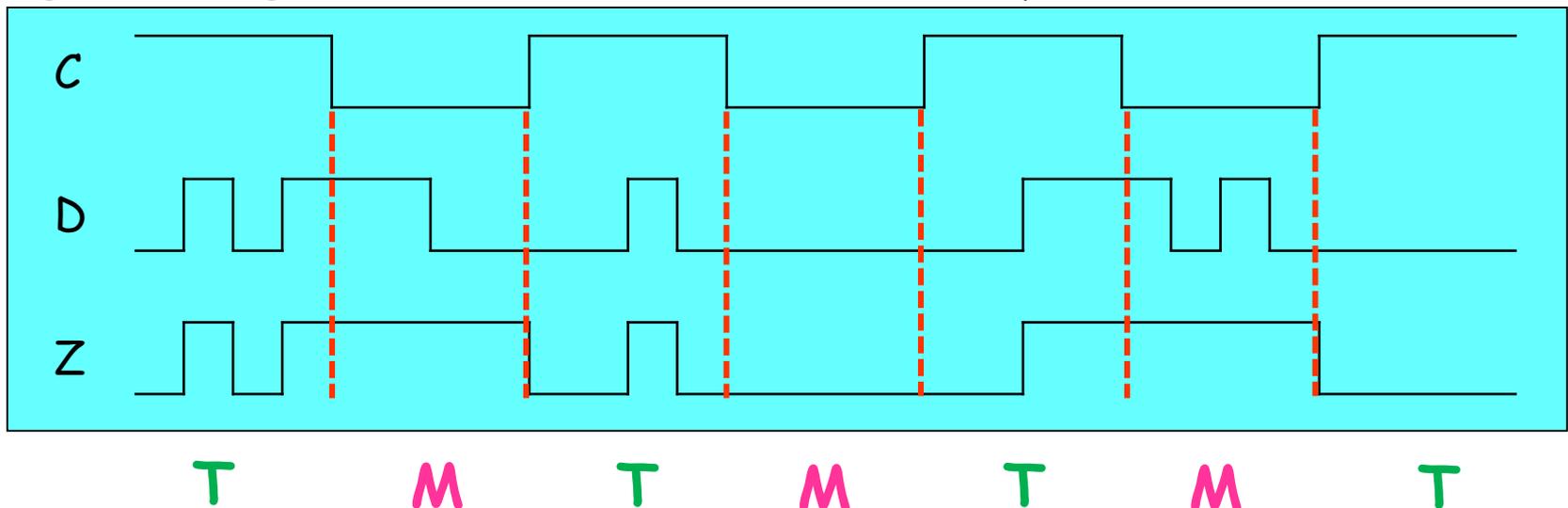
C svolge il ruolo di segnale di campionamento, *D* di segnale campionato.

Il campionamento ha luogo allorché $C=1$ ($Z=D$, uscita trasparente).

Con $C=0$: la rete memorizza l'ultimo valore assunto da *D* quando $C=1$ (memorizzazione)

Z riflette l'ultimo valore campionato

(i segnali di ingresso non cambiano mai contemporaneamente).



Grafo degli stati primitivo (modello di Moore)

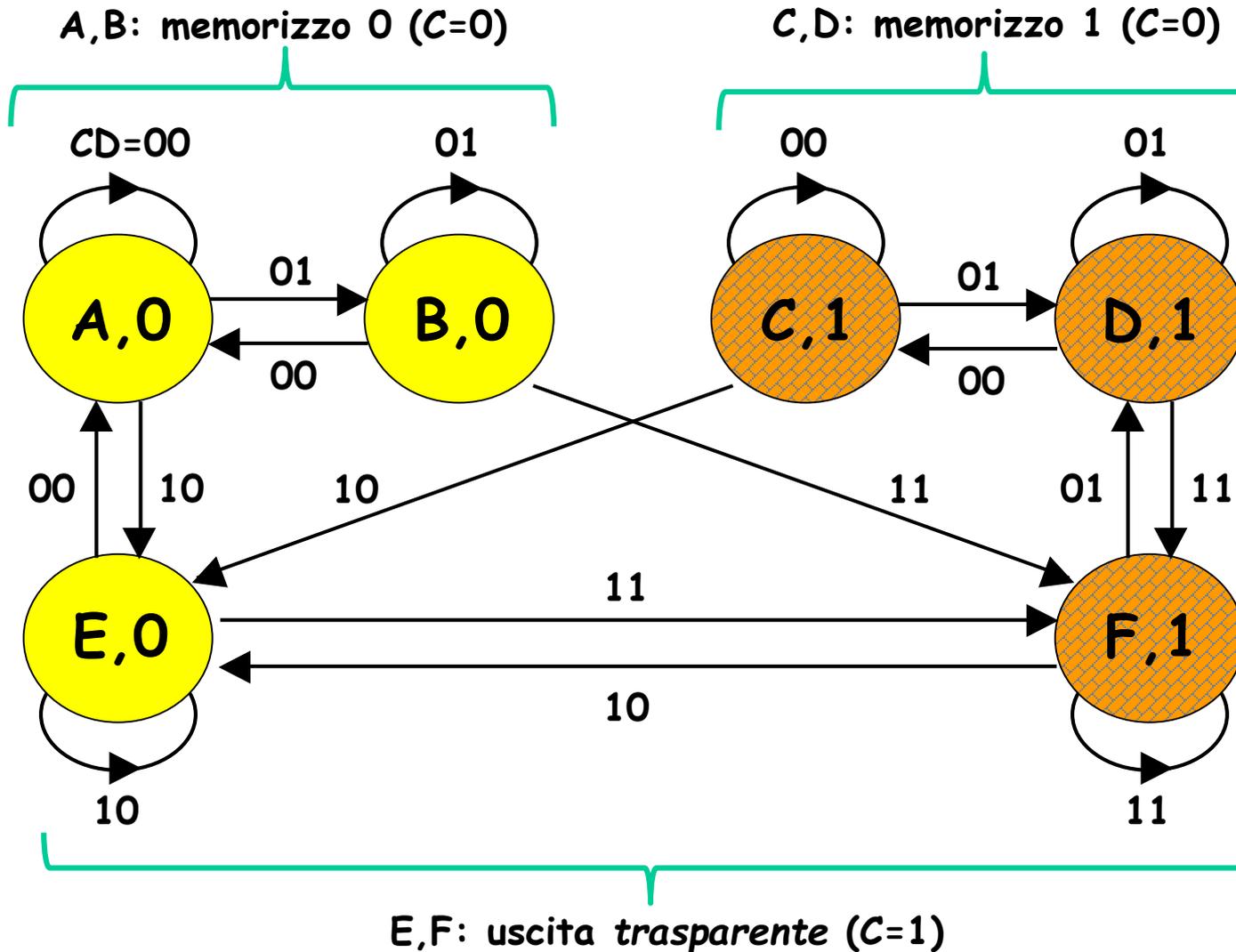


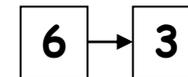
Tabella di flusso (modello di Mealy) e tabella triangolare

	CD			
	00	01	11	10
A	A,0	B,0	-, -	E,0
B	A,0	B,0	F,-	-, -
C	C,1	D,1	-, -	E,-
D	C,1	D,1	F,1	-, -
E	A,0	-, -	F,-	E,0
F	-, -	D,1	F,1	E,-

B					
C	x	x			
D	x	x			
E			x	x	
F	x	x			
	A	B	C	D	E

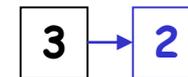
Classi massime di compatibilità

{A, B, E} {C, D, F} {E, F}



Teorema di Unger:

{A, B, E} {C, D, F} ~~{E, F}~~



condizione di copertura

condizione di chiusura



Tabella di flusso minima

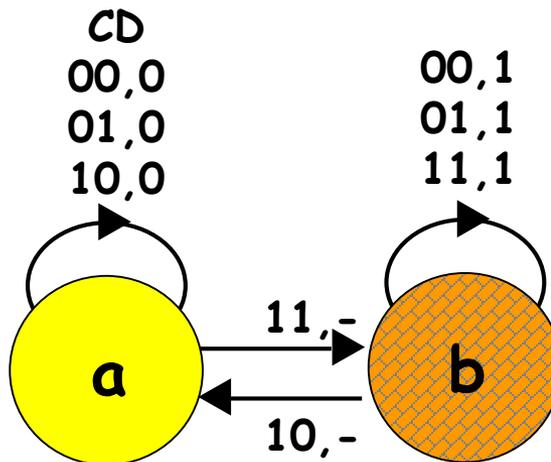
	CD			
	00	01	11	10
A	A,0	B,0	-, -	E,0
B	A,0	B,0	F,-	-, -
C	C,1	D,1	-, -	E,-
D	C,1	D,1	F,1	-, -
E	A,0	-, -	F,-	E,0
F	-, -	D,1	F,1	E,-



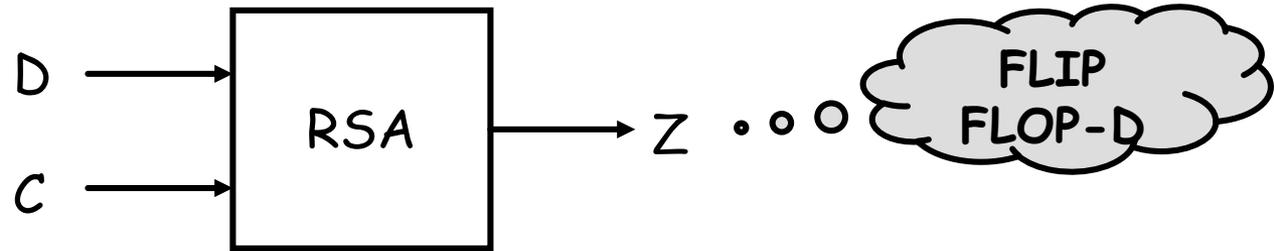
	CD			
	00	01	11	10
a	a,0	a,0	b,-	a,0
b	b,1	b,1	b,1	a,-

$a \equiv \{A, B, E\}$

$b \equiv \{C, D, F\}$

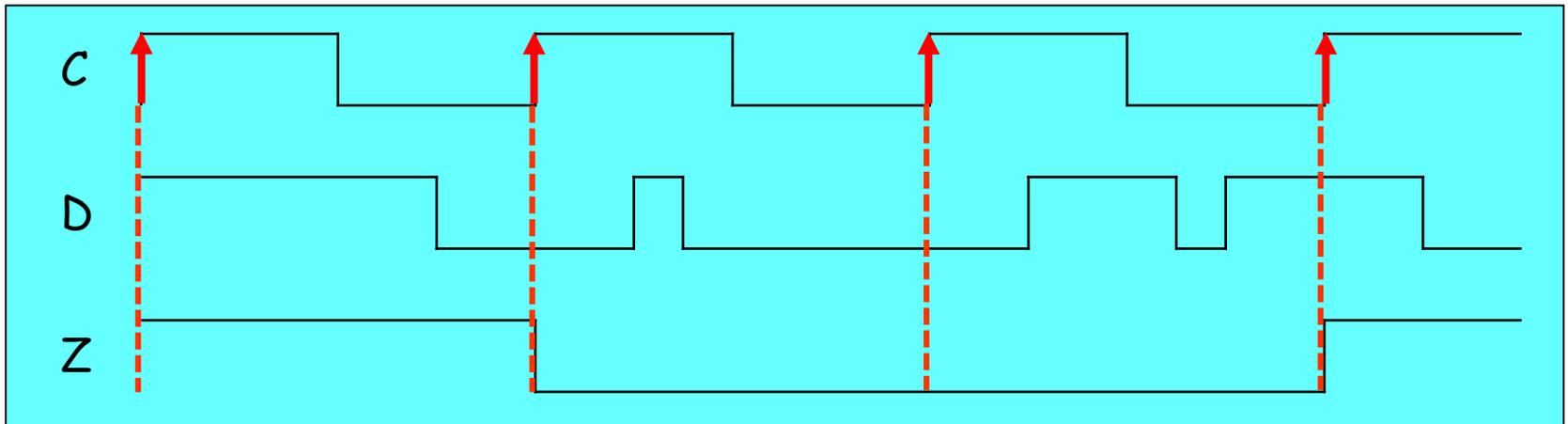


Campionatore di segnale «edge-triggered»

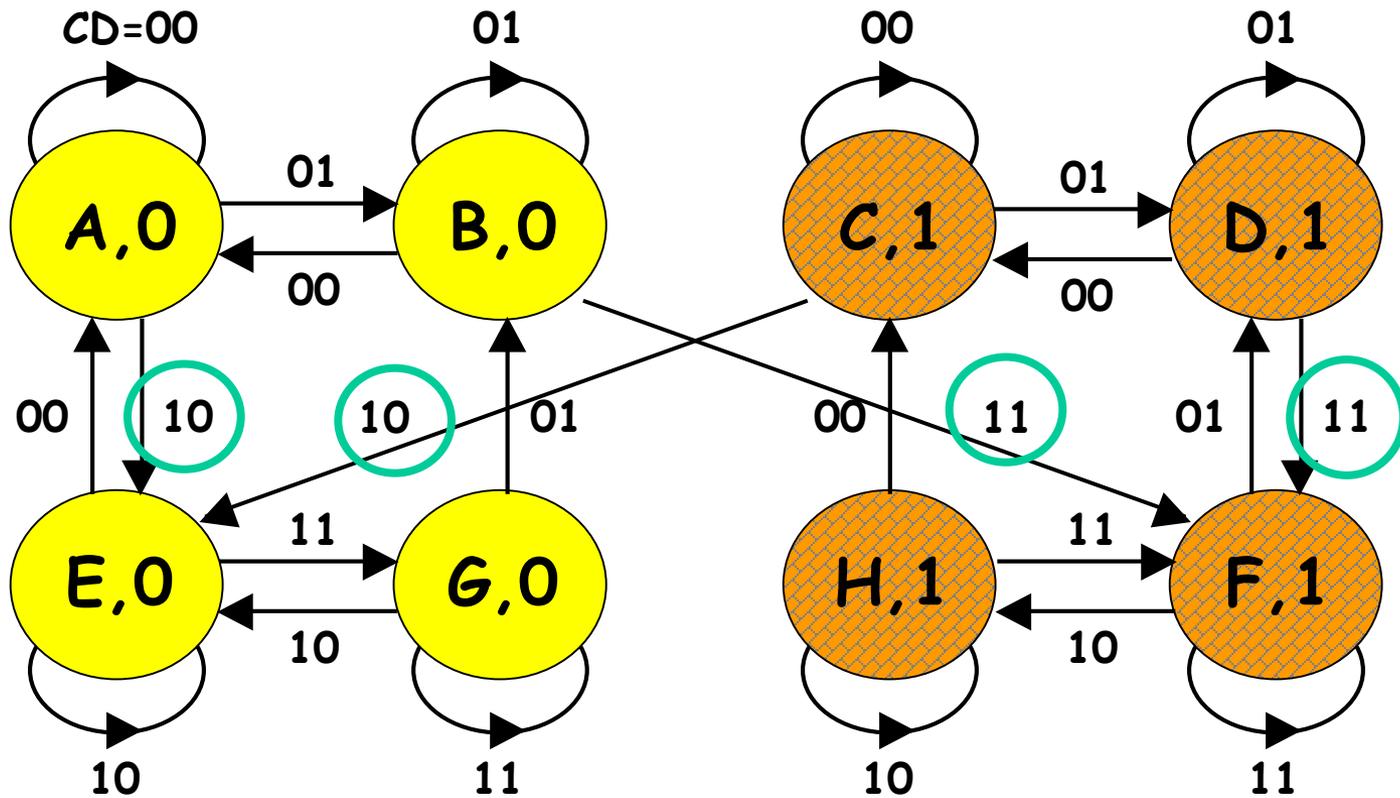


Comportamento:

C svolge il ruolo di segnale di campionamento, *D* di segnale campionato. Il campionamento ha luogo allorché *C* transita dal valore logico 0 al valore logico 1 (campionamento su fronti di salita, *edge-triggered*). *Z* riflette l'ultimo valore campionato (i segnali di ingresso non cambiano mai contemporaneamente).



Grafo degli stati primitivo (modello di Moore)



 Fronte di salita di C, campionamento

Due stati in più (G,H) rispetto al campionatore edge-triggered

Tabella di flusso (modello di Mealy) e tabella triangolare

CD

	00	01	11	10
A	A,0	B,0	-, -	E,0
B	A,0	B,0	F,-	-, -
C	C,1	D,1	-, -	E,-
D	C,1	D,1	F,1	-, -
E	A,0	-, -	G,0	E,0
F	-, -	D,1	F,1	H,1
G	-, -	B,0	G,0	E,0
H	C,1	-, -	F,1	H,1

B							
C	x	x					
D	x	x					
E		FG	x	x			
F	x	x	EH		x		
G		FG	x	x		x	
H	x	x	EH		x		x
	A	B	C	D	E	F	G

Classi di compatibilità

[A,B], [A,E], [A,G]

[C,D]

[D,H], [D,F]

[E,G]

[F,H]

Classi massime di compatibilità

[A,E,G], [A,B], [C,D], [D,F,H]

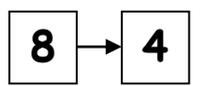
Classi di compatibilità che soddisfano copertura e chiusura

[A,E,G], [A,B], [C,D], [D,F,H]

[E,G], [A,B], [C,D], [F,H]

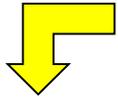
[A,E,G], B, C, [D,F,H]

(Nota: non devono necessariamente essere massime!)



Campionatore edge-triggered

CD



$\alpha \equiv \{A, E, G\}$

$\beta \equiv \{B\}$

$\gamma \equiv \{C\}$

$\delta \equiv \{D, F, H\}$



CD

	00	01	11	10
α	$\alpha, 0$	$\beta, 0$	$\alpha, 0$	$\alpha, 0$
β	$\alpha, 0$	$\beta, 0$	$\delta, -$	$-, -$
γ	$\gamma, 1$	$\delta, 1$	$-, -$	$\alpha, -$
δ	$\gamma, 1$	$\delta, 1$	$\delta, 1$	$\delta, 1$

Flip-Flop D Master-Slave

	00	01	11	10
A	A, 0	B, 0	$-, -$	E, 0
B	A, 0	B, 0	F, -	$-, -$
C	C, 1	D, 1	$-, -$	E, -
D	C, 1	D, 1	F, 1	$-, -$
E	A, 0	$-, -$	G, 0	E, 0
F	$-, -$	D, 1	F, 1	H, 1
G	$-, -$	B, 0	G, 0	E, 0
H	C, 1	$-, -$	F, 1	H, 1

$a \equiv \{A, E, G\}$

$b \equiv \{A, B\}$

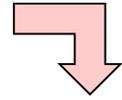
$c \equiv \{C, D\}$

$d \equiv \{D, F, H\}$



CD

	00	01	11	10
a	a/b, 0	b, 0	a, 0	a, 0
b	a/b, 0	b, 0	d, -	a, 0
c	c, 1	c/d, 1	d, 1	a, -
d	c, 1	c/d, 1	d, 1	d, 1

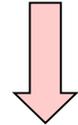


$1 \equiv \{A, B\}$

$2 \equiv \{E, G\}$

$3 \equiv \{C, D\}$

$4 \equiv \{F, H\}$

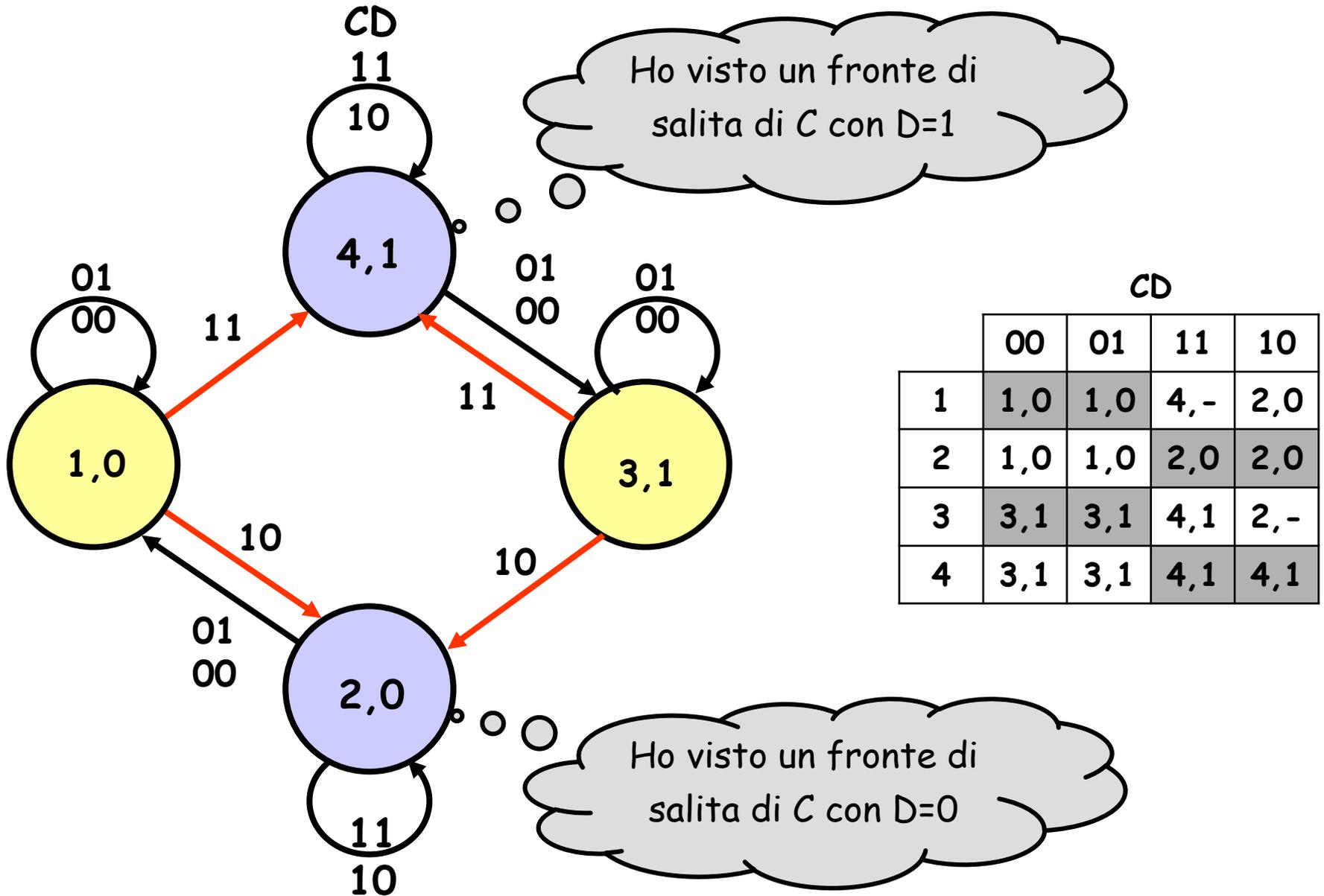


CD

	00	01	11	10
1	1, 0	1, 0	4, -	2, 0
2	1, 0	1, 0	2, 0	2, 0
3	3, 1	3, 1	4, 1	2, -
4	3, 1	3, 1	4, 1	4, 1

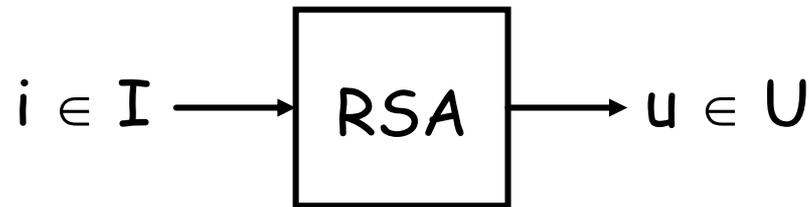
Flip-Flop D Edge-Triggered

Grafo degli stati non primitivo (modello di Moore)



Mealy vs. Moore

L'automa di Mealy coinvolge in generale un numero di stati inferiore rispetto all'automa di Moore:
 grafo degli stati → modello di Moore
 tabella di flusso → modello di Mealy



$I = \{a, b, c, d\}$

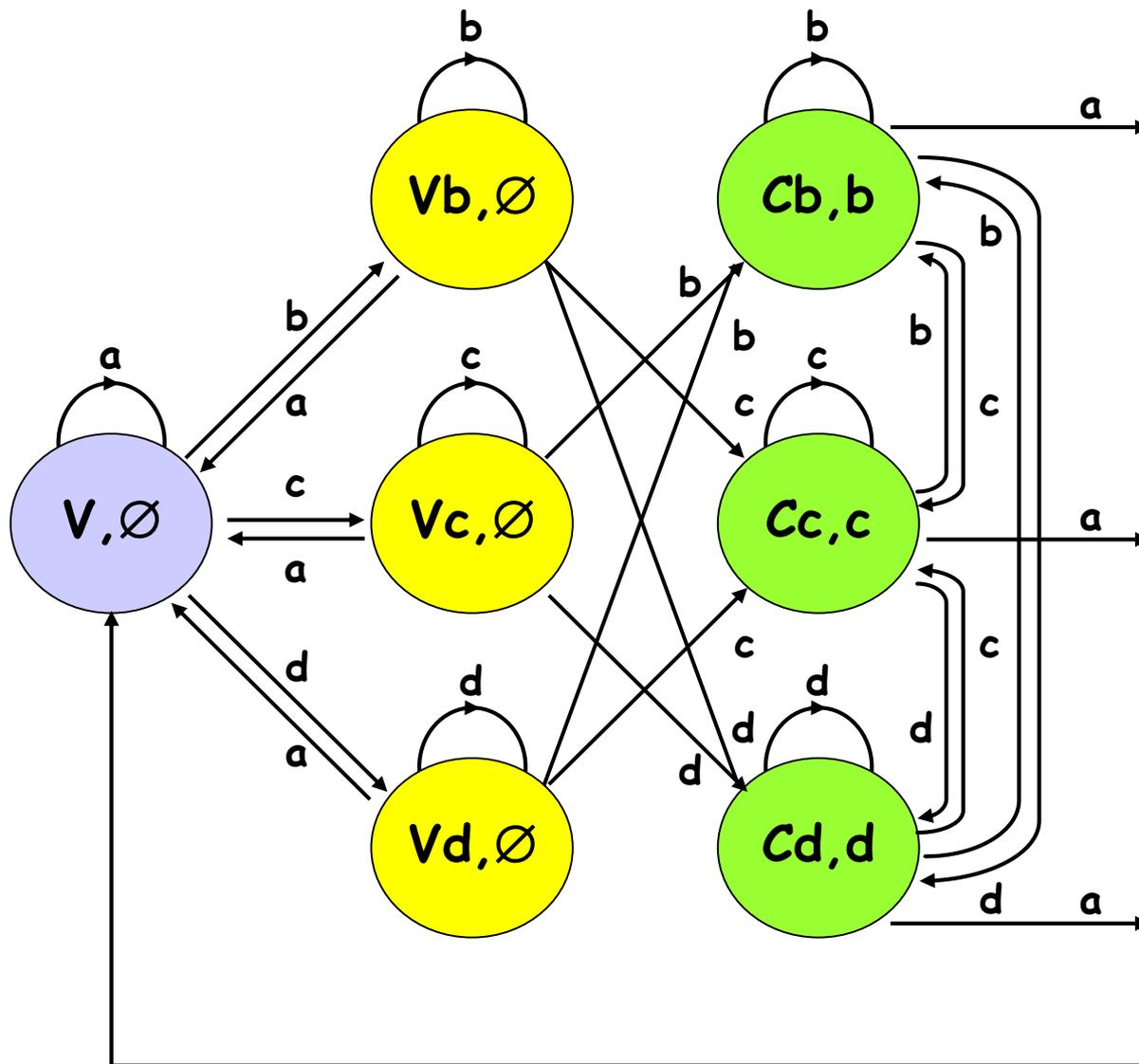
$U = \{\emptyset, b, c, d\}$

Comportamento:

il simbolo di uscita "u" deve riprodurre il simbolo "i" contestualmente presente in ingresso solo nel caso in cui "i" sia una consonante preceduta da un'altra consonante; in caso contrario, $u = \emptyset$.

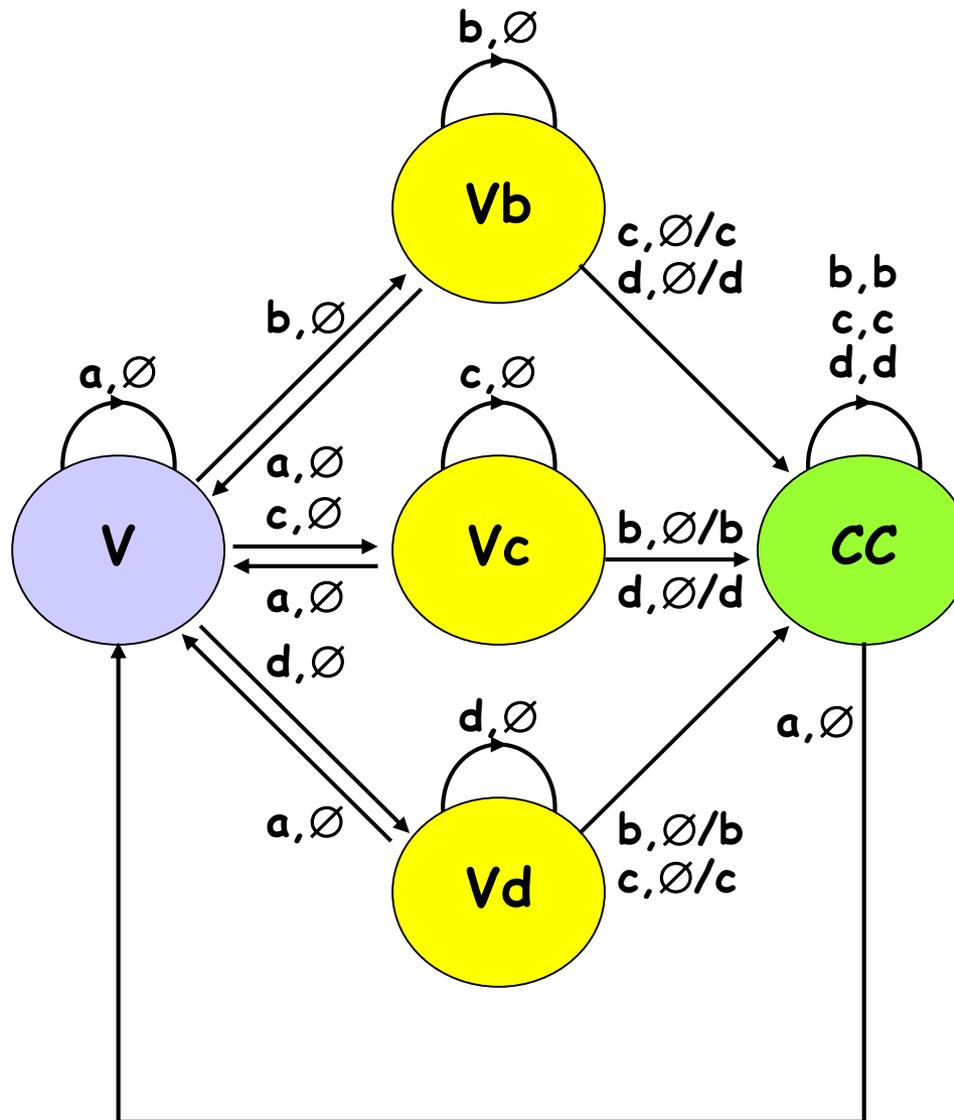
a	b	c	d	c	b	d	b	c	a
	\emptyset	c	d	c	b	d	b	c	\emptyset

Mealy vs. Moore



Modello di Moore: 7 stati (33/43 se $I \equiv$ alfabeto italiano/inglese)

Mealy vs. Moore



Modello di Mealy: 5 stati (18/23 se $I \equiv$ alfabeto italiano/inglese)

5.3 Sintesi e Analisi

Il procedimento di sintesi

Il procedimento di sintesi di una rete sequenziale asincrona è formato da 5 passi e consente di dedurre lo schema logico dal comportamento:

- 1: individuazione del grafo degli stati,
- 2: definizione della tabella di flusso,
- 3: codifica degli stati e definizione della tabella delle transizioni,
- 4: sintesi della parte combinatoria,
- 5: schema logico.

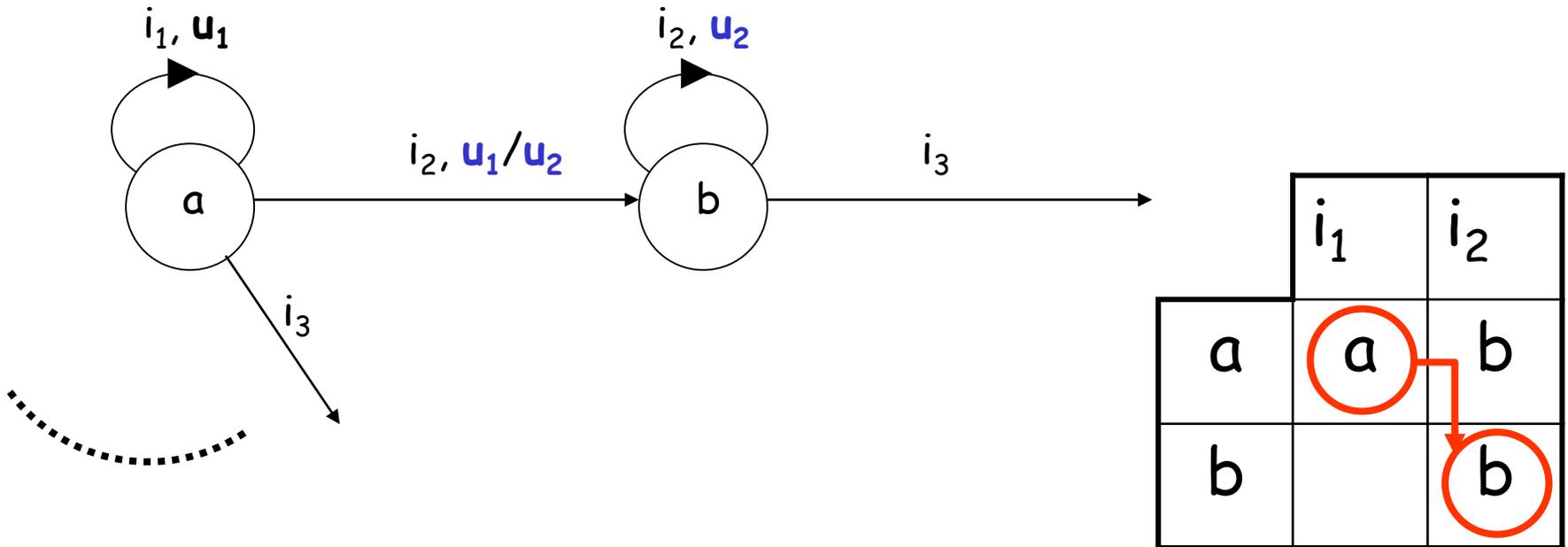
RSA: regole di corretto impiego

Obiettivo: ottenere un funzionamento corretto (passaggio da una condizione di stabilità all'altra) per qualsiasi valore dei ritardi presenti sugli anelli di retroazione.

- A tale fine, è necessario che siano verificate una serie di condizioni (che andiamo ora a analizzare) in merito a:
 1. Durata degli ingressi
 2. Codifica degli ingressi
 3. Codifica degli stati
 4. Alee statiche

1. Durata degli ingressi

L'ingresso può essere modificato solo dopo che la rete ha raggiunto la nuova stabilità

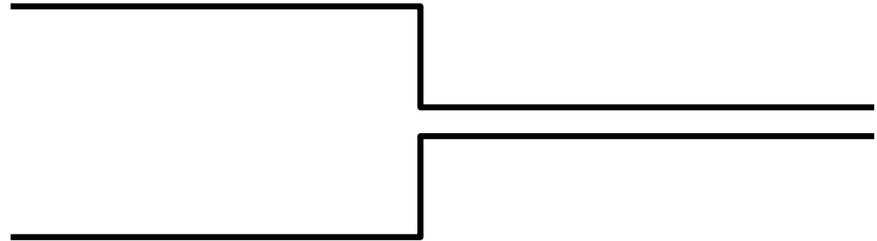


Il passaggio da una ad un'altra condizione di stabilità è detto funzionamento in modo fondamentale

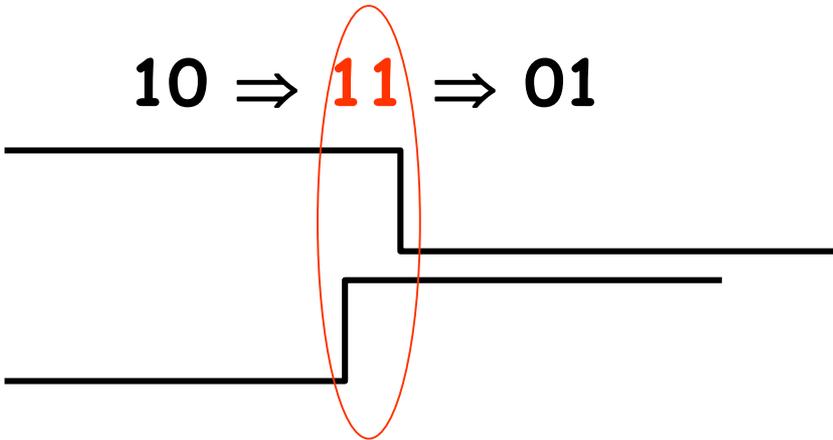
2. Codifica degli ingressi

- Se simboli in ingresso consecutivi differiscono per più di un bit, questo genererà configurazioni di ingresso «spurie»
- Dunque, la codifica dei simboli d'ingresso non può essere arbitraria: configurazioni consecutive devono essere **adiacenti**

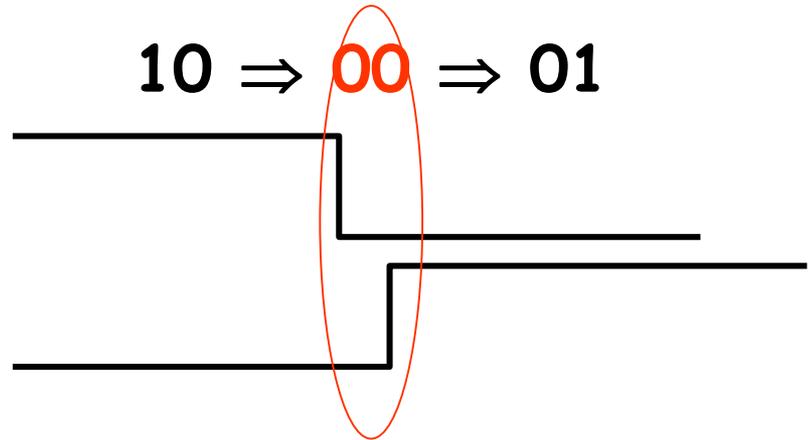
*Esempio di situazione
d'ingresso pericolosa:
10 ⇒ 01*



10 ⇒ **11** ⇒ 01



10 ⇒ **00** ⇒ 01



La codifica dei simboli d'ingresso deve garantire che simboli consecutivi differiscano sempre per il valore di un solo bit

Esempio di possibili malfunzionamenti

- Ipotizziamo di trovarci in A e di ricevere in ingresso la sequenza 00-11-10

	X_1X_2				
	00	01	11	10	Z
A	A	C	B	D	0
B	A	D	B	C	0
C	A	C	C	C	1
D	A	D	D	D	0

X_1	0	1	1
X_2	0	1	0
s.p.	A	B	C
Z	0	0	1

Caso ideale

X_1	0	0	1	1
X_2	0	1	1	0
s.p.	A	C	C	C
Z	0	1	1	1

X_1	0	1	1	1
X_2	0	0	1	0
s.p.	A	D	D	D
Z	0	0	0	0

Casi reali

3. Codifica degli stati

- Una situazione analoga si verifica anche per quanto riguarda la **codifica degli stati interni**:

Per eliminare a priori la presenza di stati interni "spuri", le configurazioni associate ad ogni coppia (stato presente, stato futuro) devono essere adiacenti

- Questo richiede un codice molto ridondante, che utilizza molti più bit del necessario (condizione molto restrittiva)
- Fortunatamente si tratta di un vincolo spesso eccessivo: si può garantire il corretto funzionamento della rete anche in presenza di variazione contemporanea di più di un bit di stato interno («corse non critiche», v. prossima slide)

Corse critiche e corse non critiche

- Segnali in retroazione per cui è stata prevista una modifica **contemporanea** di valore si trovano in una situazione di **corsa**
- nel circuito i cambiamenti si verificheranno a istanti diversi e con un ordine dettato dai ritardi interni.
- Una **corsa** è **critica** se si possono raggiungere stabilità diverse.
- Una **corsa non critica** genera una **transizione multipla**, lecita a patto che
 - L'uscita non presenti andamenti diversi dal comportamento voluto
 - Gli ingressi rimangano stabili fino al raggiungimento dello stato d'arrivo

Stato 11, ingresso I1

11 \Rightarrow 00 \Rightarrow 00

11 \Rightarrow 01 \Rightarrow 00

11 \Rightarrow 10 \Rightarrow 00

corsa non critica

	I1	I2	I3
00	00	01	00
01	00	01	10
11	00	11	11
10	00	11	10

Transizione multipla

Stato 01, ingresso I3

01 \Rightarrow 10 \Rightarrow 10

01 \Rightarrow 00 \Rightarrow 00

01 \Rightarrow 11 \Rightarrow 11

corsa critica

Prevenzione a priori delle corse critiche

- Come si nota nell'esempio precedente, la presenza di corse critiche si ha nei casi in cui la colonna presenti **più di uno stato stabile**
- In caso di colonne con **una sola stabilità**, infatti, se tutti gli stati non stabili riconducono allo stato stabile, si avranno sempre corse non critiche
- Si possono eliminare a priori le situazioni di corse critiche seguendo le seguenti regole:
 1. Nelle colonne con **una sola stabilità** si inserisce il simbolo dello stato stabile al posto di eventuali condizioni d'indifferenza.
 2. Per le sole colonne **con più stabilità** si traccia il **grafo delle adiacenze**: ad ogni stato è associato un nodo e ad ogni coppia stato presente - stato futuro un ramo orientato che connette i due nodi corrispondenti.
 3. Si sovrappone il grafo ad una mappa per il minimo numero di variabili di stato e si verifica se è possibile **assegnare configurazioni adiacenti ad ogni coppia di stati coinvolta in una transizione**.
 4. Se è impossibile soddisfare tutte i vincoli di adiacenza, si cerca di ridurli ricorrendo a **transizioni multiple**.
 5. Se non ci si riesce, si **incrementa il numero delle variabili di stato** e si ritorna a 4.

Una codifica priva di corse

FF-D (edge triggered)

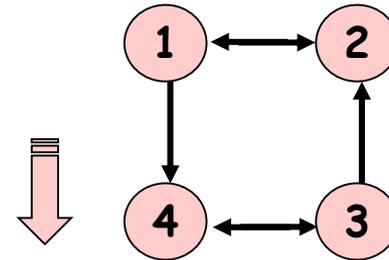
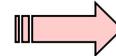
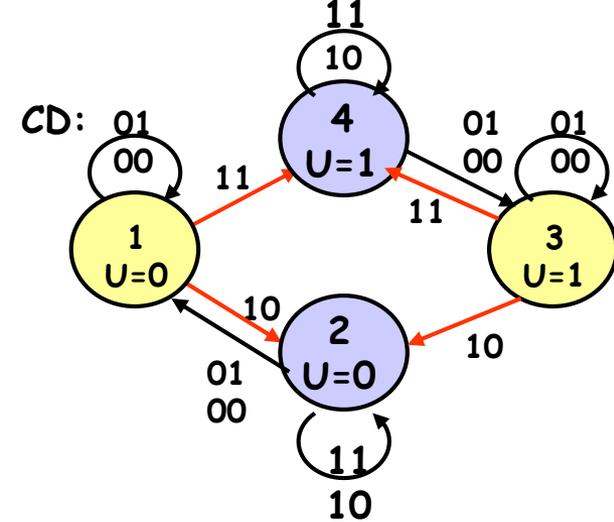
CD

	00	01	11	10
1	1,0	1,0	4,-	2,0
2	1,0	1,0	2,0	2,0
3	3,1	3,1	4,1	2,-
4	3,1	3,1	4,1	4,1

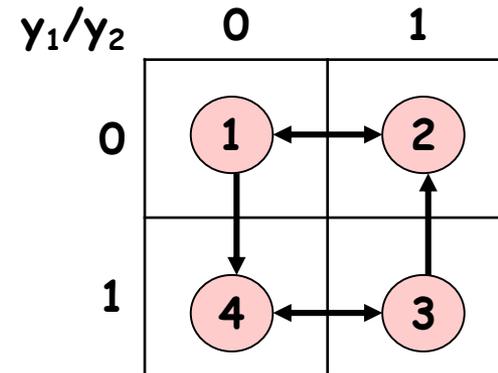
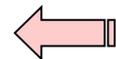
CD

	00	01	11	10
00	00,0	00,0	10,-	01,0
01	00,0	00,0	01,0	01,0
11	11,1	11,1	10,1	01,-
10	11,1	11,1	10,1	10,1

y_1y_2



Grafo delle adiacenze

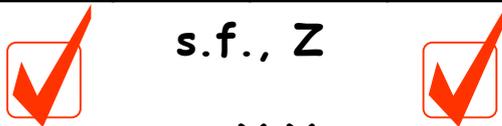


Mappa di codifica

Una codifica con corse non critiche

Riconoscitore della sequenza 00-01-11

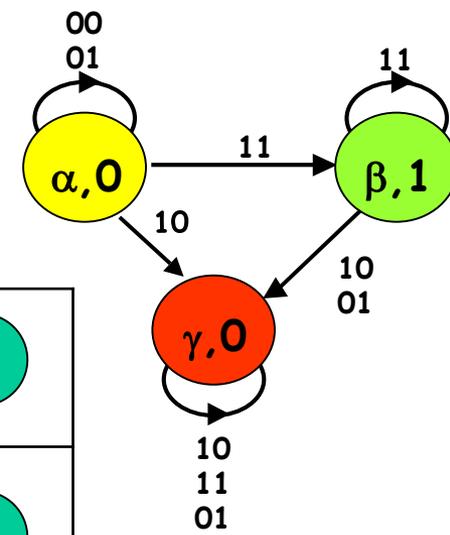
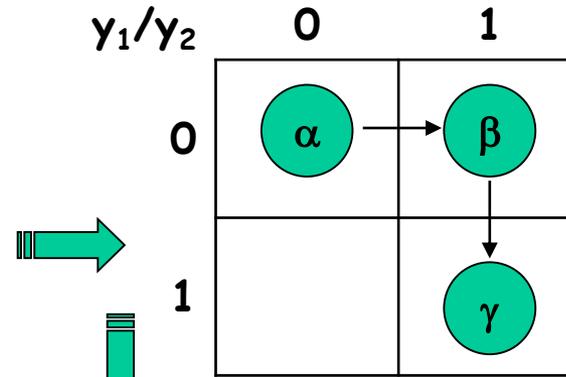
		X_1X_2			
		00	01	11	10
s.p.	α	$\alpha,0$	$\alpha,0$	$\beta,-$	$\gamma,0$
	β	$-, -$	$\gamma,-$	$\beta,1$	$\gamma,-$
	γ	$\alpha,0$	$\gamma,0$	$\gamma,0$	$\gamma,0$



s.f., Z

X_1X_2

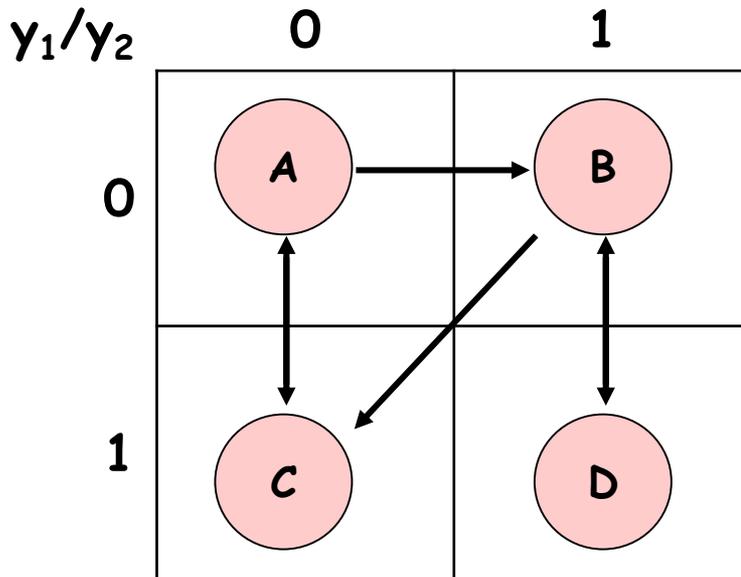
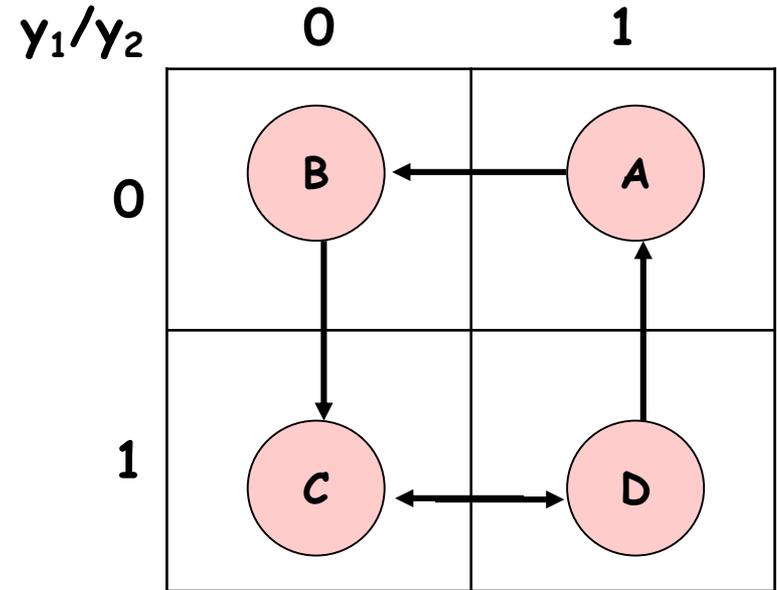
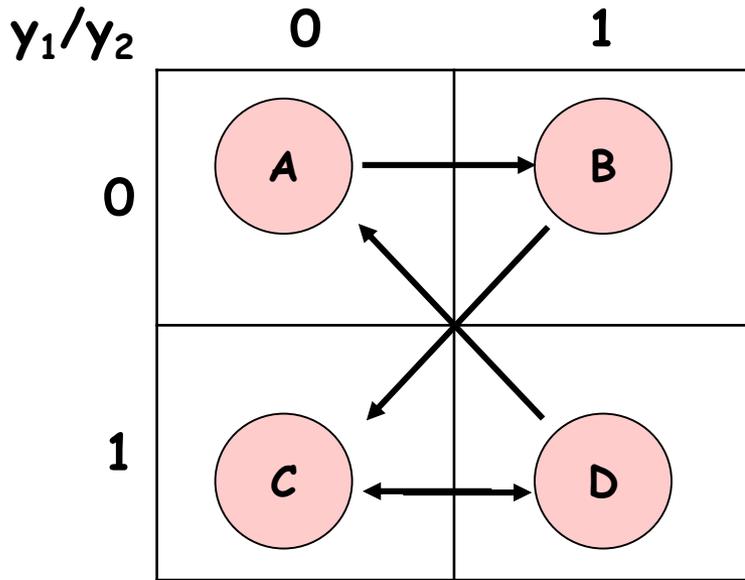
		X_1X_2			
		00	01	11	10
Y_1Y_2	00	00,0	00,0	01,-	11,0
	01	00,0	11,-	01,1	11,0
	11	00,0	11,0	11,0	11,0
	10	00,0	--,-	--,-	11,0



		X_1X_2			
		00	01	11	10
Y_1Y_2	00	00,0	00,0	01,-	11,0
	01	--,-	11,-	01,1	11,-
	11	00,0	11,0	11,0	11,0
	10	--,-	--,-	--,-	--,-

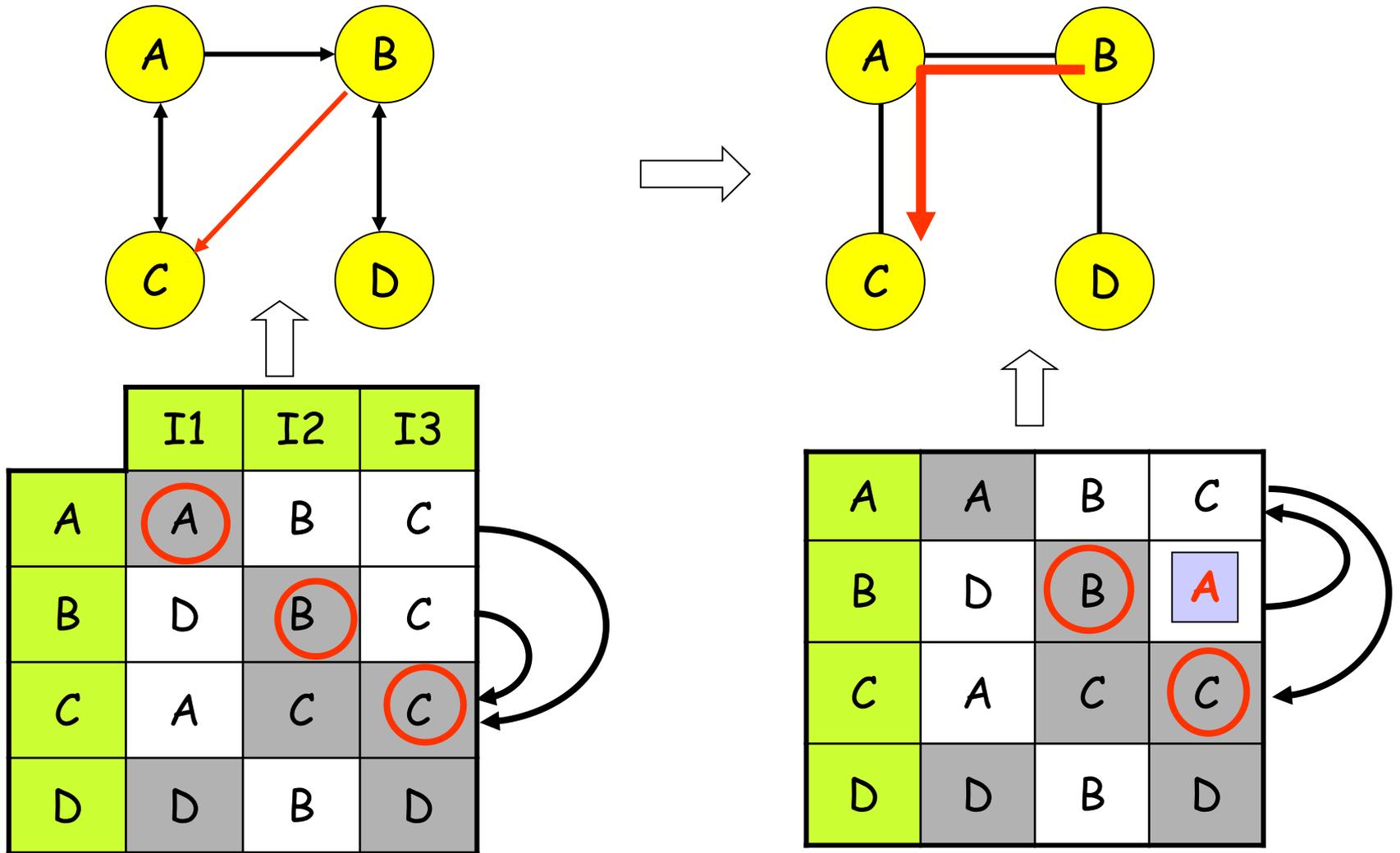
- La presenza di due colonne con una sola stabilità semplifica il grafo delle adiacenze: le corse presenti nella rete sono tutte non critiche (a patto di sostituire in tali colonne le indifferenze con lo stato stabile)

Passo 4: ricerca di configurazioni adiacenti



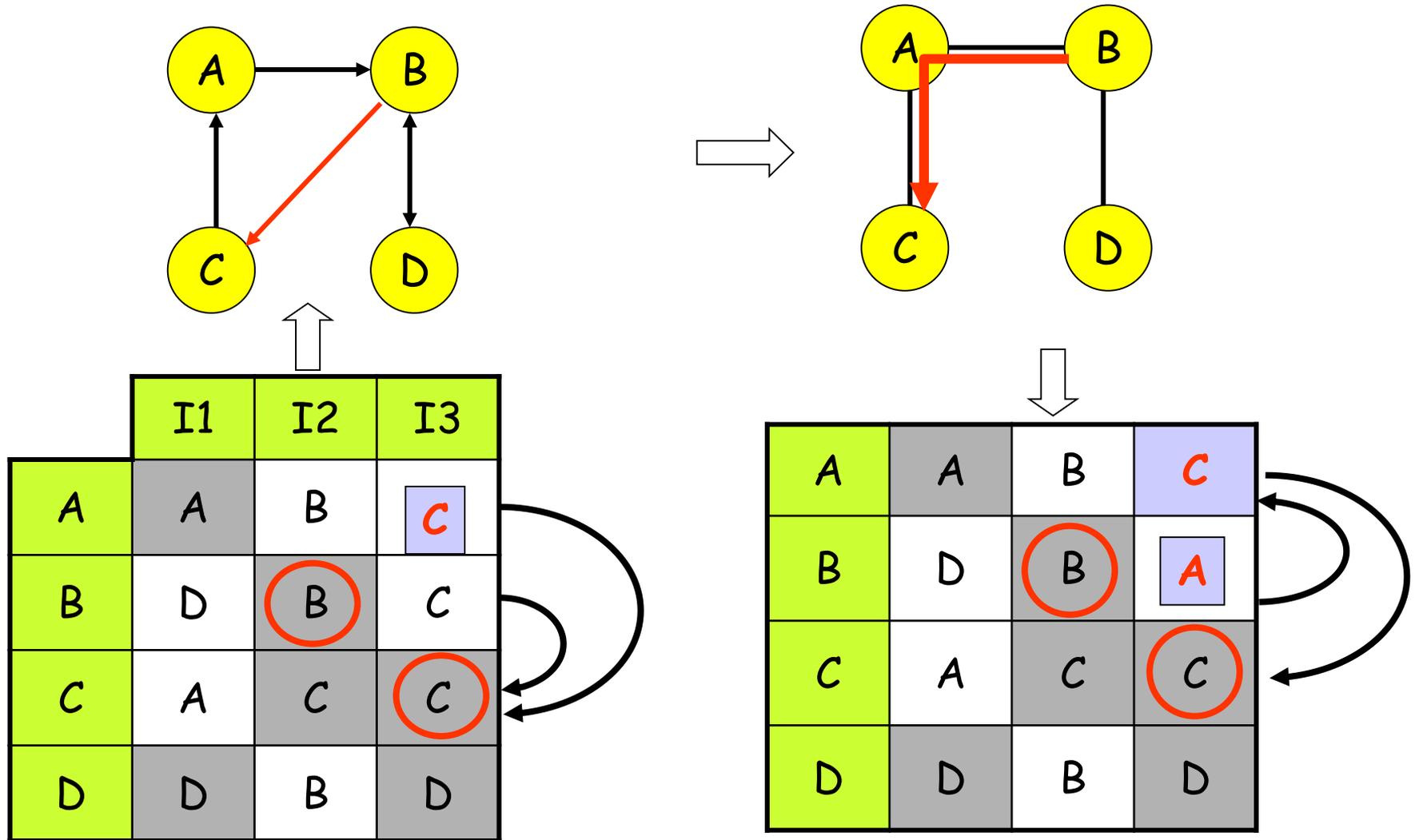
- Tramite il grafo delle adiacenze, si cerca una codifica che permetta solo transizioni tra stati adiacenti
- Non sempre esiste una codifica con queste caratteristiche: in tale caso, occorre procedere con il passo successivo (utilizzo di *transizioni multiple*)

Passo 5: utilizzo di transizioni multiple ...



- In questo caso, non esistono codifiche degli stati con due soli bit che evitino la presenza di corse critiche
- Occorre dunque applicare il passo 4) riduzione dei vincoli di adiacenza introducendo transizioni multiple

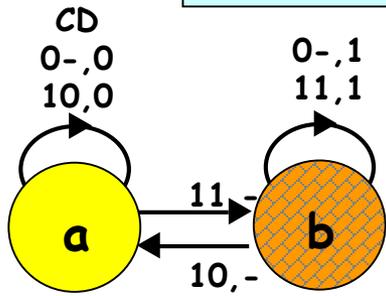
... Transizioni multiple



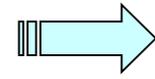
- Il passo 4) può essere applicato anche in presenza di stati con stato futuro indifferente (stato A nell'esempio)
- In questo caso si sfrutta l'impossibilità dell'ingresso I3 per lo stato A per inserire la transizione multipla al posto dell'indifferenza

4. Alee statiche

D-latch



	CD			
	00	01	11	10
a	a,0	a,0	b,-	a,0
b	b,1	b,1	b,1	a,-

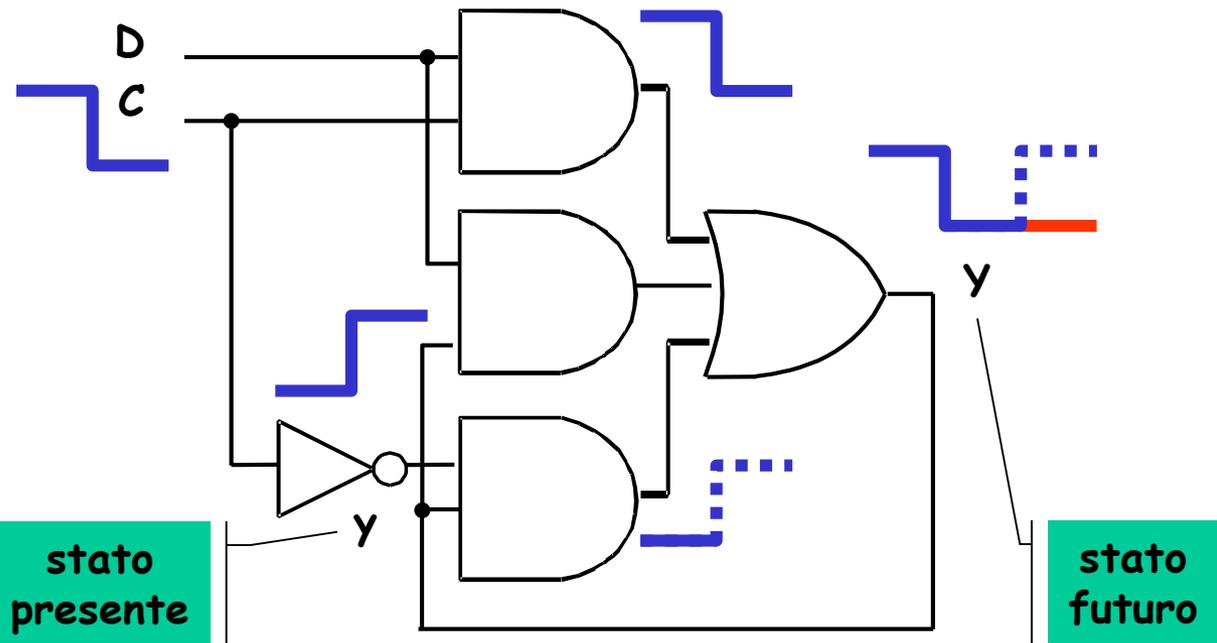


	CD			
	00	01	11	10
y	0,0	0,0	1,-	0,0
b≡1	1,1	1,1	1,1	0,-

Y,Z

$$Z = y$$

$$Y = CD + C' y$$



stato presente

stato futuro

	CD			
	00	01	11	10
a≡0	0	0	1	0
b≡1	1	1	1	0

y

$$Y = CD + C' y + D y$$

- La copertura non ridondante della mappa della variabile di stato genera alee statiche durante il transitorio, che può portare a una transizione non corretta
- Qui, y si abbassa temporaneamente, portando a una stabilità sullo stato a

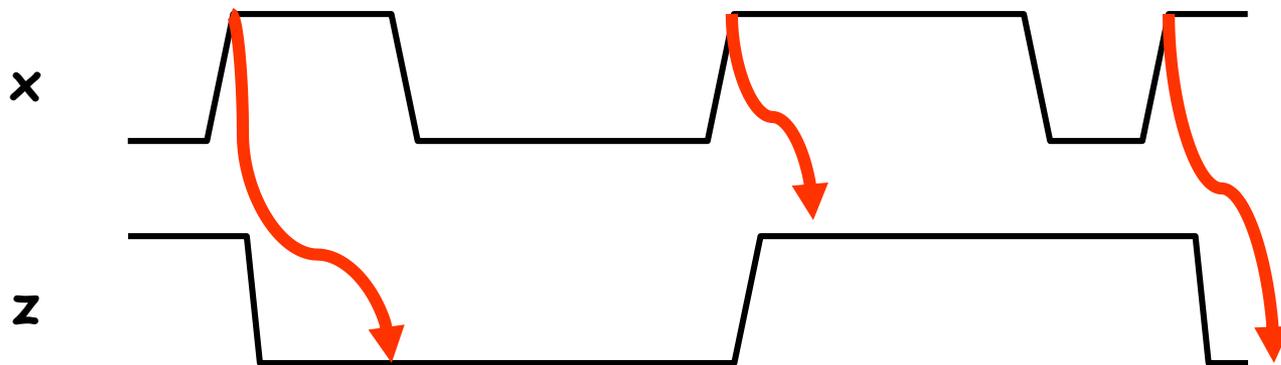
Coperture ridondanti - Per evitare a priori le alee statiche, "ogni coppia di celle adiacenti da coprire deve essere racchiusa in un RR" (anche gli eventuali RR ridondanti devono avere dimensione massima).

Esempio di sintesi: la lampada da tavolo

Comportamento: "z cambia di valore ad ogni fronte di salita di x"



Lampada da tavolo

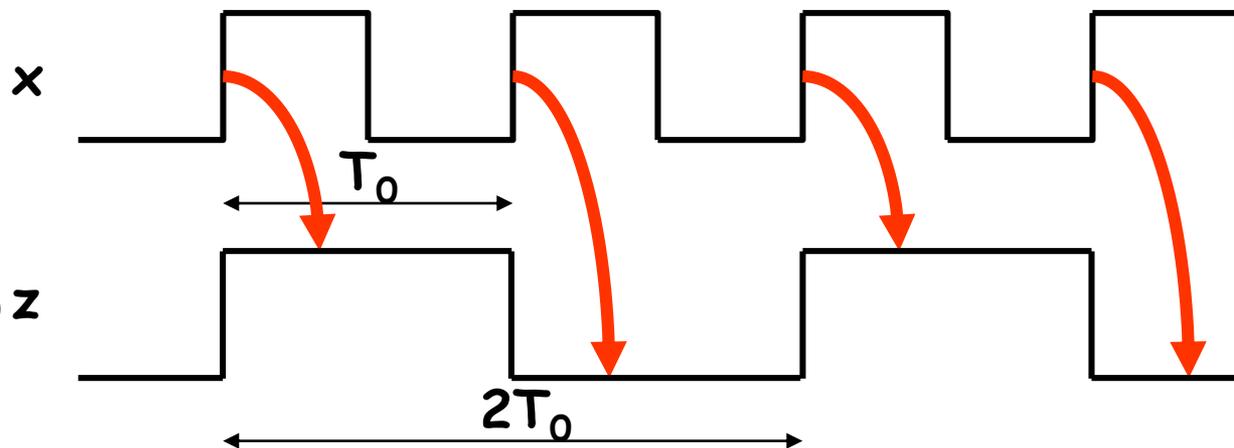


stesso
comportamento
del:

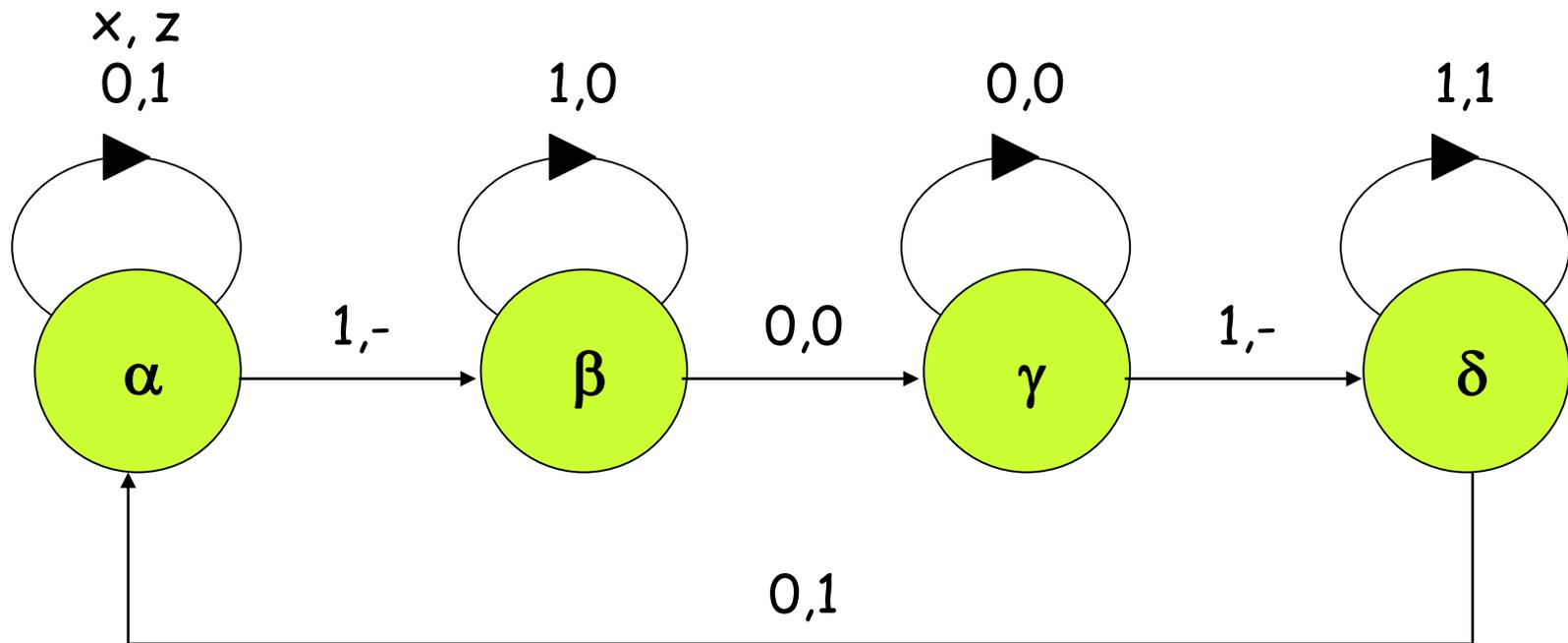
Divisore x2

della frequenza

di un segnale periodico z



1: Grafo degli stati



Notiamo le seguenti due caratteristiche della rete:

1. **Stabilità:** ogni stato è stabile per l'ingresso che lo genera.
2. **Indifferenza sull'uscita:** essendo i transistori brevi, la modifica di uscita può avvenire già durante la transizione oppure, indifferentemente, essere differita al raggiungimento della stabilità. L'utilizzo dell'indifferenza permette più libertà durante il passo successivo di copertura delle mappe delle variabili d'uscita

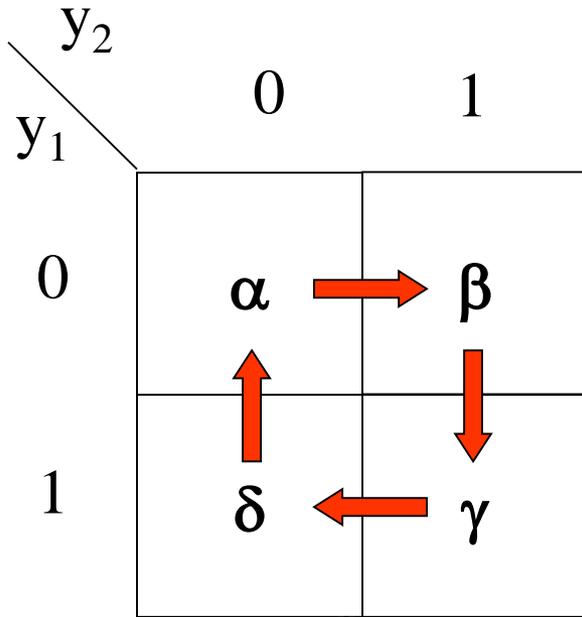
2: Tabella di flusso

stato \ x	0	1
	α	$\alpha, 1$
β	$\gamma, 0$	$\beta, 0$
γ	$\gamma, 0$	$\delta, -$
δ	$\alpha, 1$	$\delta, 1$

CONTROLLI FORMALI

1. In ogni riga ci deve essere almeno una condizione di stabilità.
2. In ogni colonna si deve raggiungere sempre una stabilità.
3. Le situazioni di instabilità devono indicare uno stato futuro stabile nella colonna (assenza di transizioni multiple).

3: Tabella delle transizioni



Grafo delle adiacenze
e mappa di codifica

Codifica degli stati - A stati consecutivi (stato presente e futuro) si devono assegnare configurazioni **adiacenti**.

$y_1, y_2 \backslash x$	0	1
$\alpha: 00$	00,1	01,-
$\beta: 01$	11,0	01,0
$\gamma: 11$	11,0	10,-
$\delta: 10$	00,1	10,1

$Y_1 Y_2, z$

Il grafo delle adiacenze mostra assenza di corse critiche: si può procedere con la codifica degli stati

4: Espressioni

Ipotesi: si desiderano reti minime di tipo SP

	y_1y_2			
x	00	01	11	10
0	0	1	1	0
1	0	0	1	1

	y_1y_2			
x	00	01	11	10
0	0	1	1	0
1	1	1	0	0

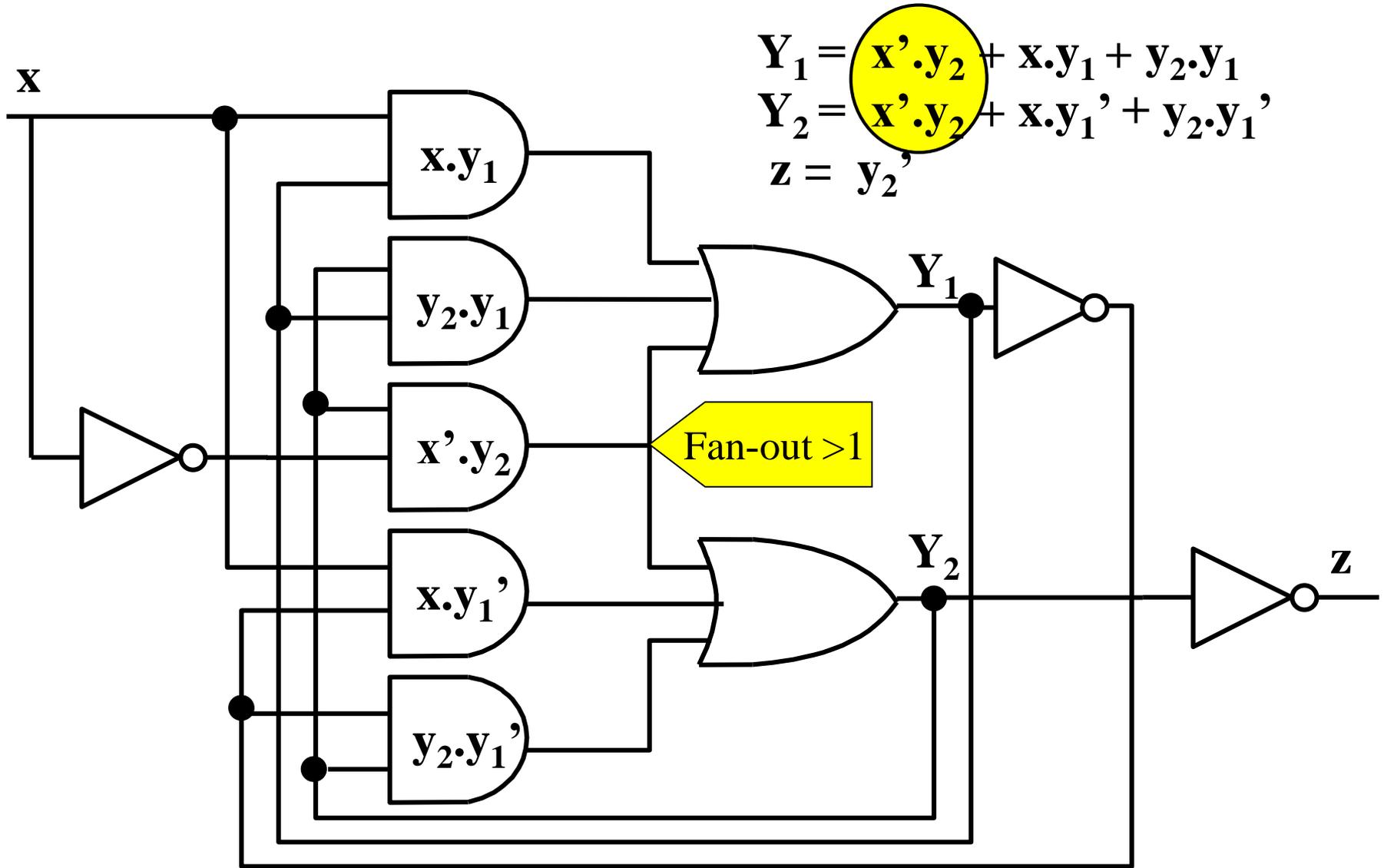
	y_1y_2			
x	00	01	11	10
0	1	0	0	1
1	-	0	-	1

$$Y_1 = x' \cdot y_2 + x \cdot y_1 + y_2 \cdot y_1 \quad Y_2 = x' \cdot y_2 + x \cdot y_1' + y_2 \cdot y_1'$$

$$z = y_2'$$

Coperture ridondanti per evitare a priori il pericolo di alee statiche

5: Schema logico



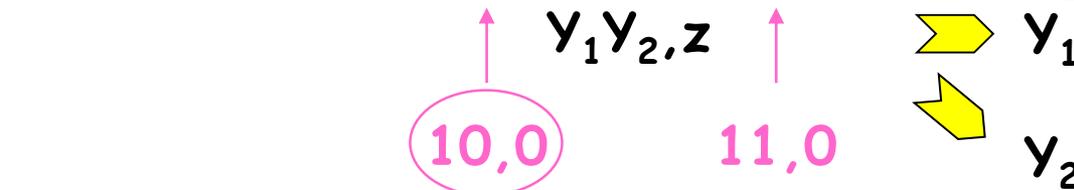
La realizzazione di termini uguali deve essere attribuita ad un unico gate, sfruttando la sua proprietà di avere fan-out maggiore di 1

Riconoscitore di sequenza - parte finale

- Terminiamo la sintesi della rete «riconoscitore di sequenza» vista in precedenza
- Solo 3 stati realizzano il funzionamento della rete (lo stato 10 non ha stabilità)

		x_1x_2			
		00	01	11	10
y_1y_2	00	00,0	00,0	01,-	11,0
	01	00,0	11,-	01,1	11,0
	11	00,0	11,0	11,0	11,0
	10	00,0	--,-	--,-	11,0

		00	01	11	10
00	0	0	0	0	1
01	0	1	0	0	1
11	0	1	1	0	1
10	0	-	-	-	1



**l'uno o l'altro ?
entrambi ! (alea)**

$$y_1 = x_1x_2' + x_1'x_2y_2 + x_1y_1 + x_2y_1$$

$$y_2 = x_1 + x_2y_2 + x_2y_1$$

$$z = x_1x_2y_1'$$

autoinizializzazione

		00	01	11	10
00	0	0	0	1	1
01	0	1	0	1	1
11	0	1	1	0	1
10	0	-	-	-	1

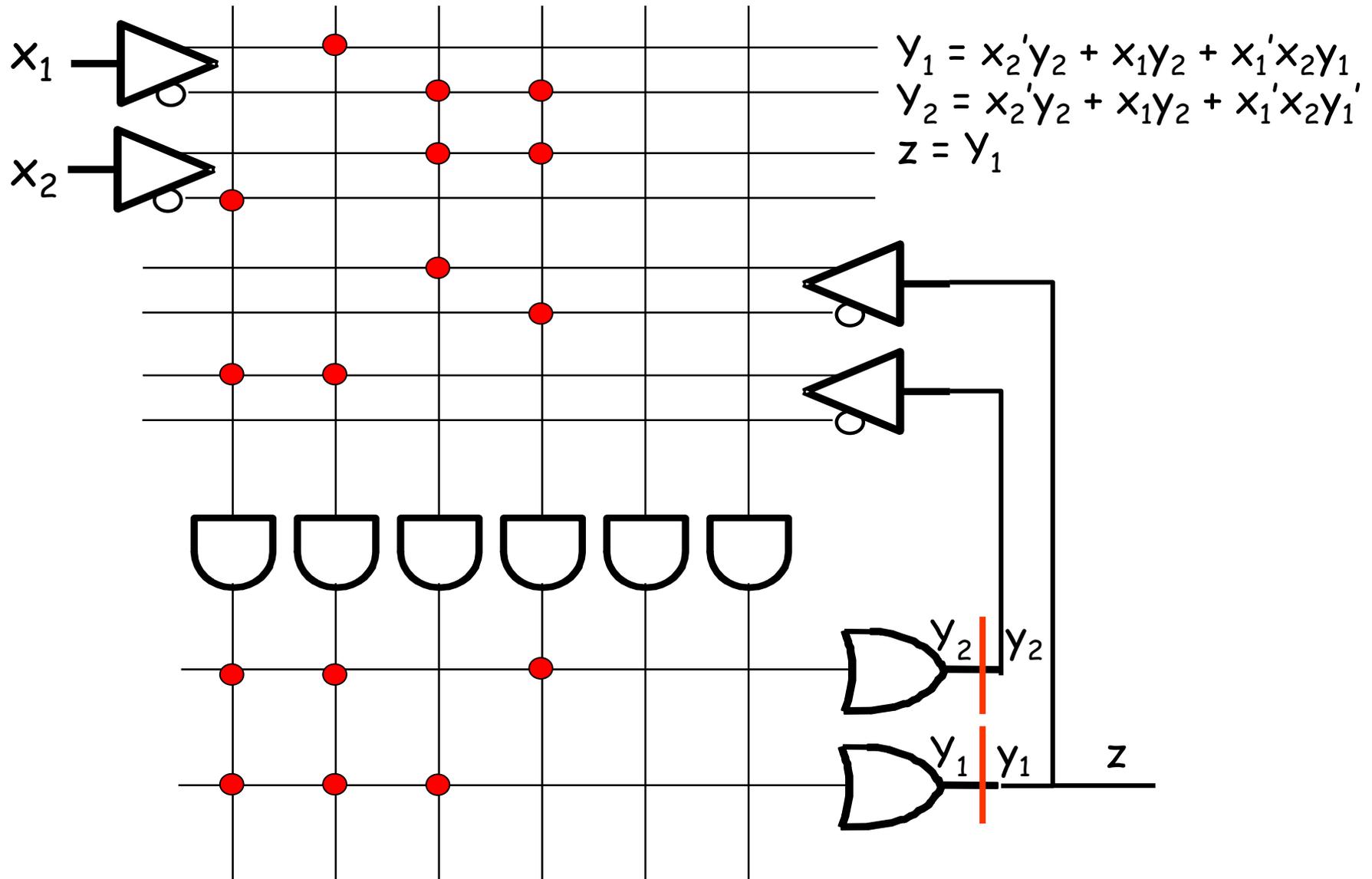
- Con le coperture effettuate, lo stato 10 ha una stabilità con ingresso 01: aggiungiamo una copertura così da eliminarla, al fine di autoinizializzare la rete

Il procedimento di analisi

Il procedimento di analisi di una rete sequenziale asincrona è formato da 5 passi e consente di dedurre il comportamento dallo schema logico:

- 1: individuazione delle variabili di stato,
- 2: analisi della parte combinatoria,
- 3: individuazione della tabella delle transizioni,
- 4: studio delle condizioni di stabilità,
- 5: individuazione della tabella di flusso e del grafo degli stati.

Esempio: PLA con due retroazioni (passi 1&2)



- La rete ha 2 ingressi (x_1, x_2), 2 variabili di stato (y_1, y_2), 1 uscita (z)

Individuazione della TdT (passo 3)

$$Y_1 = x_2' y_2 + x_1 y_2 + x_1' x_2 y_1$$

$$Y_2 = x_2' y_2 + x_1 y_2 + x_1' x_2 y_1'$$

$$z = Y_1$$

	$x_1 x_2$			
$y_1 y_2$	00	01	11	10
00	0	0	0	0
01	1	0	1	1
11	1	1	1	1
10	0	1	0	0

Y_1

	$x_1 x_2$			
$y_1 y_2$	00	01	11	10
00	0	1	0	0
01	1	1	1	1
11	1	0	1	1
10	0	0	0	0

Y_2

- Se ne deduce che nella progettazione del circuito:
 - Si è utilizzato il criterio di costo minimo (n° minimo di RR, ciascuno di dimensioni massime)
 - Non si sono eliminate a priori le alee statiche (mancano i RR ridondanti $y_1 y_2$ in Y_1 , $y_1' y_2$ in Y_2)

Stabilità, TdF e GdS (passi 4 e 5)

Y_1Y_2	x_1x_2			
	00	01	11	10
$00 \equiv A$	A,0	B,0	A,0	A,0
$01 \equiv B$	C,-	B,0	C,-	-, -
$11 \equiv C$	C,1	D,1	C,1	C,1
$10 \equiv D$	A,-	D,1	A,-	-, -

Nel passaggio da TdT a TdF, poniamo delle indifferenze

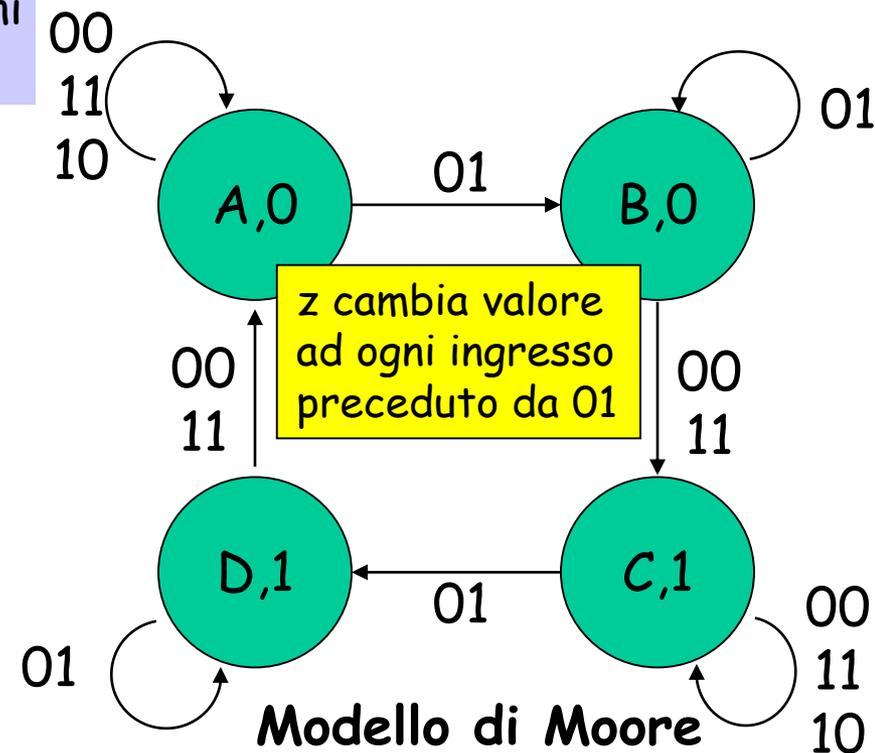
- sulle uscite nelle transizioni in cui lo stato futuro è associato a un cambiamento dell'uscita
- sullo stato futuro, se associato a una configurazione degli ingressi non adiacente a quella di (almeno) una stabilità sulla stessa riga

Ogni colonna ha due stabilità e due transizioni in arrivo: funziona in modo **fondamentale**

Y_1Y_2	x_1x_2			
	00	01	11	10
00	00,0	01,0	00,0	00,0
01	11,1	01,0	11,1	11,1
11	11,1	10,1	11,1	11,1
10	00,0	10,1	00,0	00,0

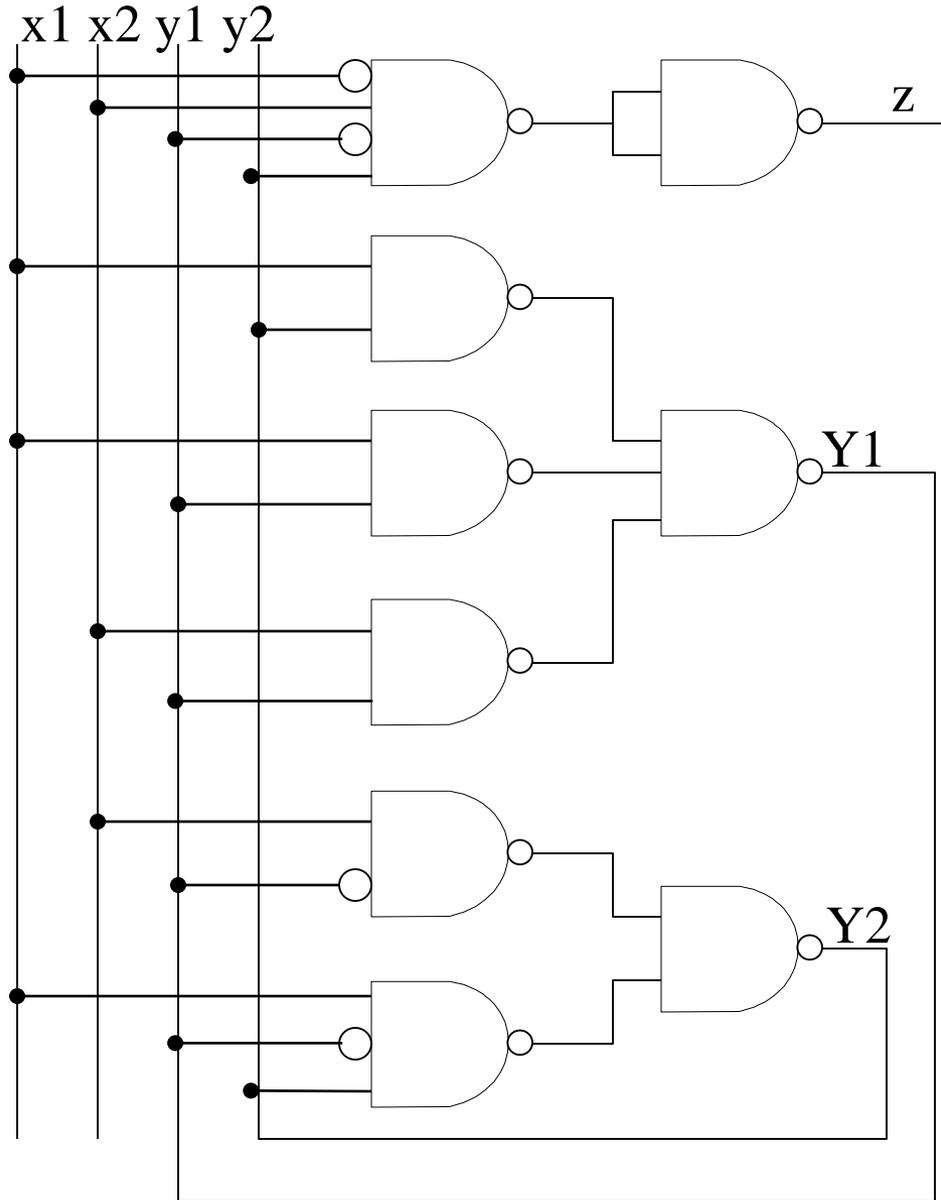
Ogni transizione modifica un solo bit di stato: non ci sono corse critiche

Modello di Mealy



Modello di Moore

Esempio 2: rete a NAND con 2 retroazioni



La rete ha

- 2 ingressi (x_1, x_2)
- 2 variabili di stato (y_1, y_2)
- 1 uscita (z)

$$Y_1 = (x_1 \uparrow y_2) \uparrow (x_1 \uparrow y_1) \uparrow (x_2 \uparrow y_1) \\ = x_1 \cdot y_2 + x_1 \cdot y_1 + x_2 \cdot y_1$$

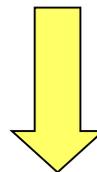
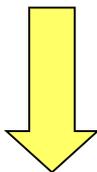
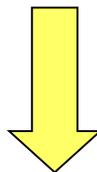
$$Y_2 = (x_2 \uparrow y_1') \uparrow (x_1 \uparrow y_1' \uparrow y_2) \\ = x_2 \cdot y_1' + x_1 \cdot y_1' \cdot y_2$$

$$z = x_1' \cdot x_2 \cdot y_1' \cdot y_2$$

$$Y1 = x1 \cdot y2 + x1 \cdot y1 + x2 \cdot y1$$

$$Y2 = x2 \cdot y1' + x1 \cdot y1' \cdot y2$$

$$z = x1' \cdot x2 \cdot y1' \cdot y2$$



		x_1x_2			
		00	01	11	10
y_1y_2	00	0	0	0	0
	01	0	0	1	1
	11	0	1	1	1
	10	0	1	1	1

		x_1x_2			
		00	01	11	10
y_1y_2	00	0	1	1	0
	01	0	1	1	1
	11	0	0	0	0
	10	0	0	0	0

		x_1x_2			
		00	01	11	10
y_1y_2	00	0	0	0	0
	01	0	1	0	0
	11	0	0	0	0
	10	0	0	0	0

Y1

Y2

z

Corsa non critica
11-10-00
11-01-00



Stato instabile

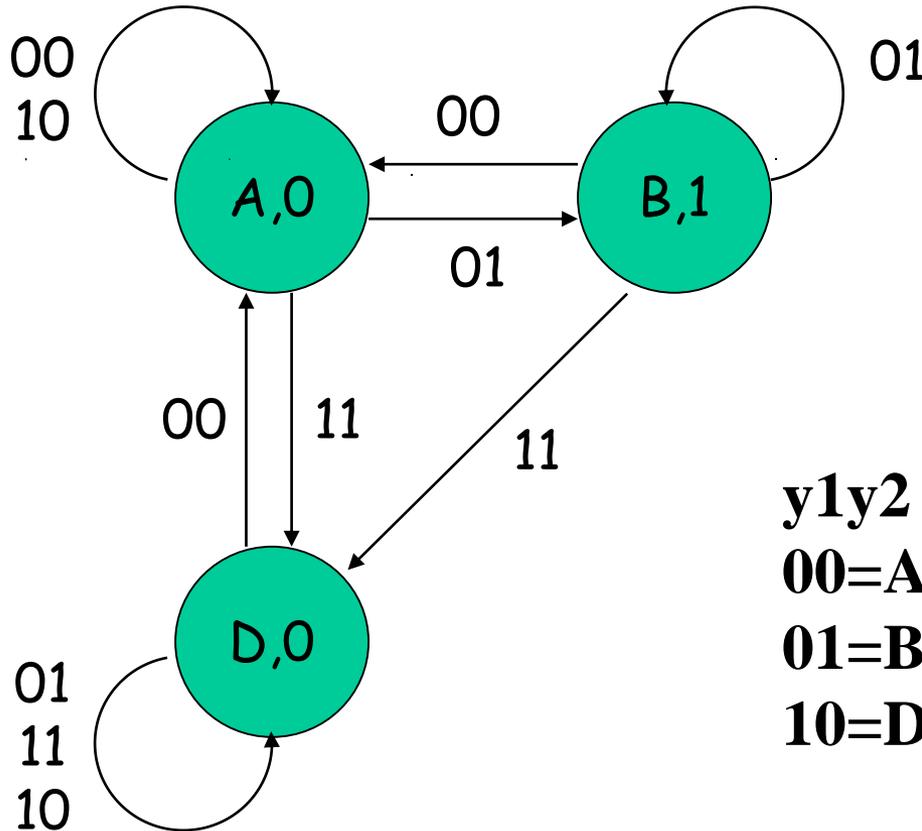
		x_1x_2			
		00	01	11	10
y_1y_2	00	00,0	01,0	01,0	00,0
	01	00,0	01,1	11,0	11,0
	11	00,0	10,0	10,0	10,0
	10	00,0	10,0	10,0	10,0

Y1Y2,z

TdF e GdS

- Lo stato "11" non ha stabilità: non fa parte dell'automa minimo

	x1x2			
y1y2	00	01	11	10
00	00,0	01,0	01,0	00,0
01	00,0	01,1	11,0	11,0
11	00,0	10,0	10,0	10,0
10	00,0	10,0	10,0	10,0



Y1Y2,z

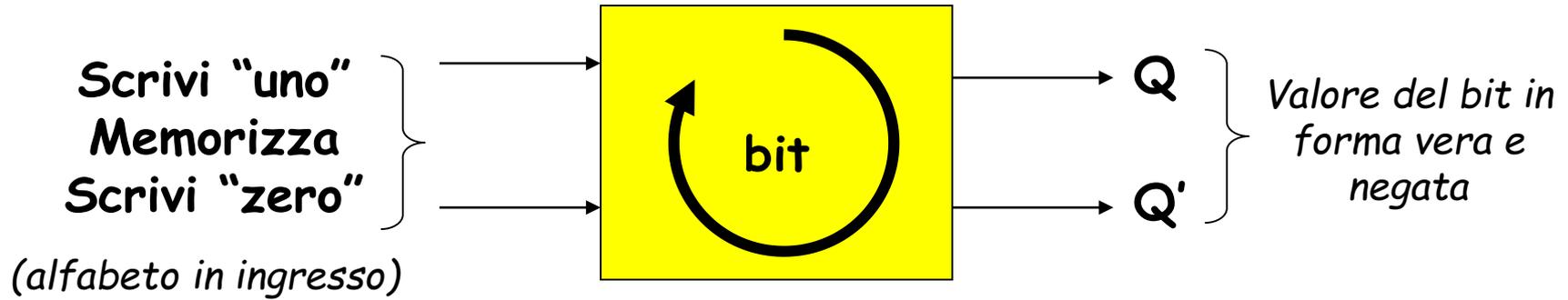
	x1x2			
y1y2	00	01	11	10
00=A	A,0	B,-	D,0	A,0
01=B	A,-	B,1	D,-	-,-
10=D	A,0	D,0	D,0	D,0

Non adiacente all'unica configurazione d'ingresso stabile di B

5.4 Memorie binarie

Memorie binarie

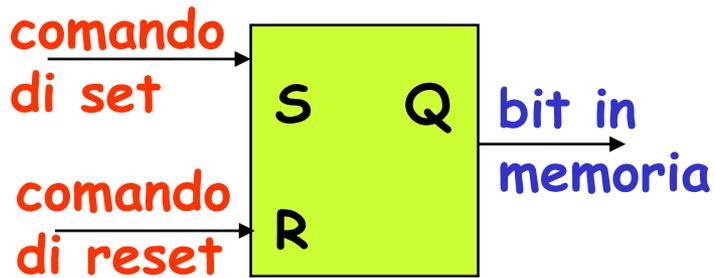
- Scopo: "ricordare" il valore di un bit
- Componenti **primitivi** per le macchine digitali



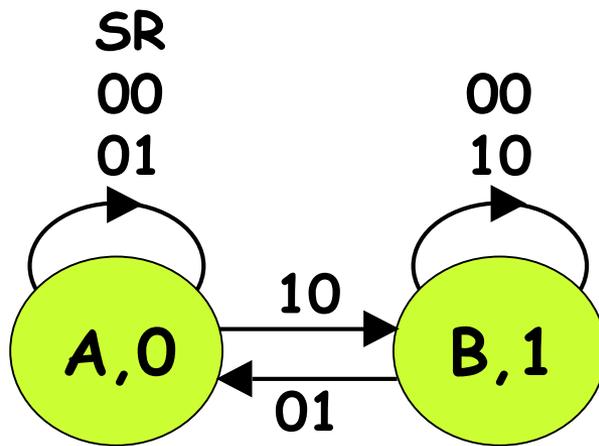
- Tre component principali:
 - Latch SR
 - Latch CD (o D-Latch)
 - Flip-Flop D
- Si differenziano per **cosa** scrivere e **quando** scrivere
- Tali differenze ne influenzano
 - la **complessità strutturale**
 - la **semplicità d'uso**

Latch SR

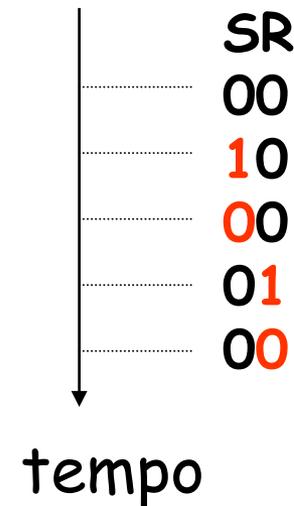
- Memoria binaria con due stati interni e due comandi: Set e Reset
- È il motore marcia/arresto visto in precedenza



S	R	Q
0	0	Q
1	0	1
0	1	0



Grafo degli Stati



TdF, TdT, equazioni caratteristiche

- y : variabile di stato presente, Y : variabile di stato futuro
- Tabella di flusso ha 4 condizioni di stabilità

SR

		00	01	11	10
s.p.	A	$(A,0)$	$(A,0)$	-,-	$B,-$
	B	$(B,1)$	$A,-$	-,-	$(B,1)$

s.f.,Q

SR

		00	01	11	10
Y	$A \equiv 0$	0,0	0,0	-,-	1,-
	$B \equiv 1$	1,1	0,-	-,-	1,1

Y,Q

$$Q = y$$

SR

		00	01	11	10
Y	0	0	0	-	1
	1	1	0	-	1

$$Y = S + R' \cdot y$$

Eq. caratteristica minima SP

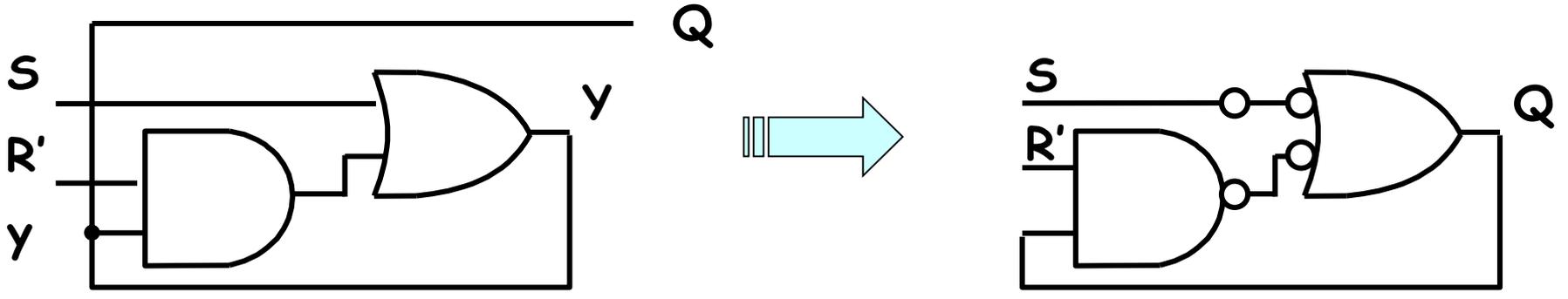
SR

		00	01	11	10
Y	0	0	0	-	1
	1	1	0	-	1

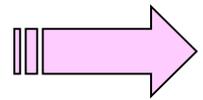
$$Y = R' \cdot (S + y)$$

Eq. caratteristica minima PS

Schema logico (dall'eq. caratteristica SP)

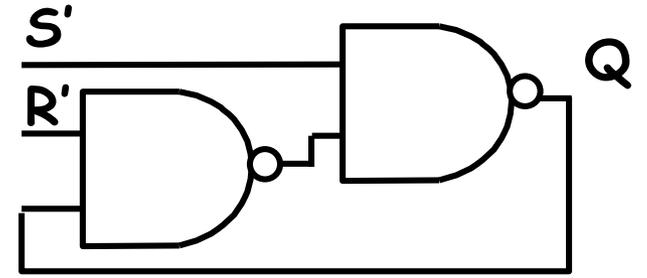
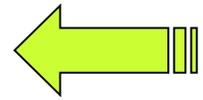
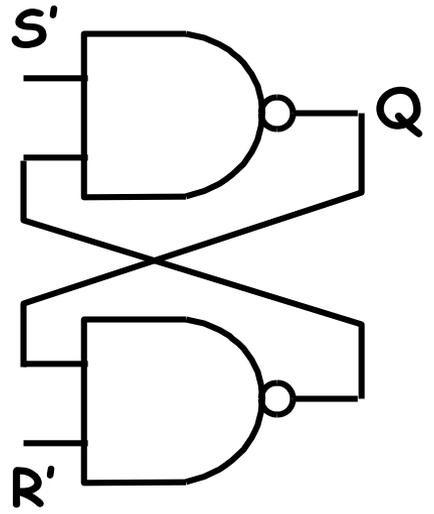
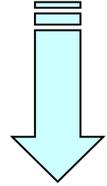
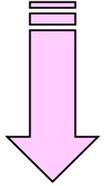


$$y = S + R' \cdot y$$

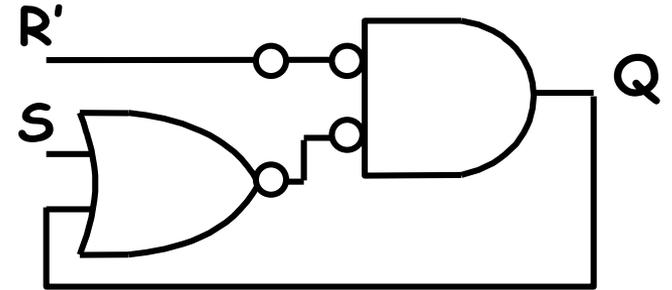
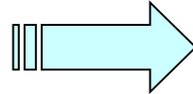
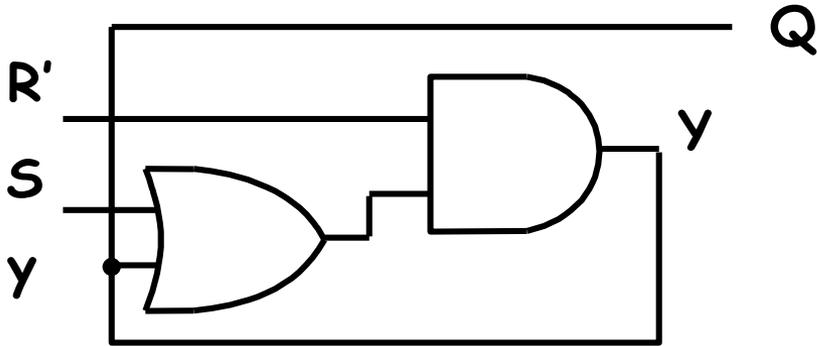


$$= S' \uparrow (R' \uparrow y)$$

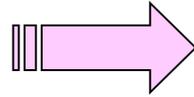
$$Q = y$$



Schema logico (dall'eq. caratteristica PS)

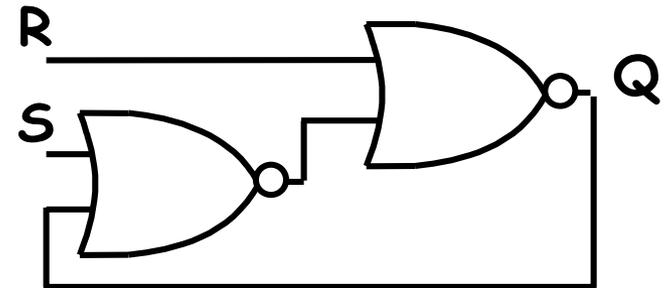
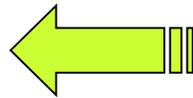
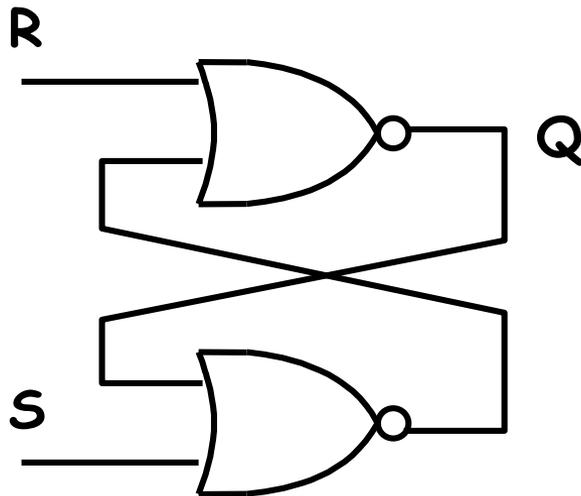
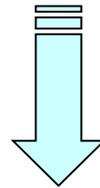
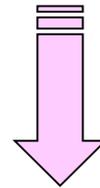


$$Y = R' \cdot (S + y)$$

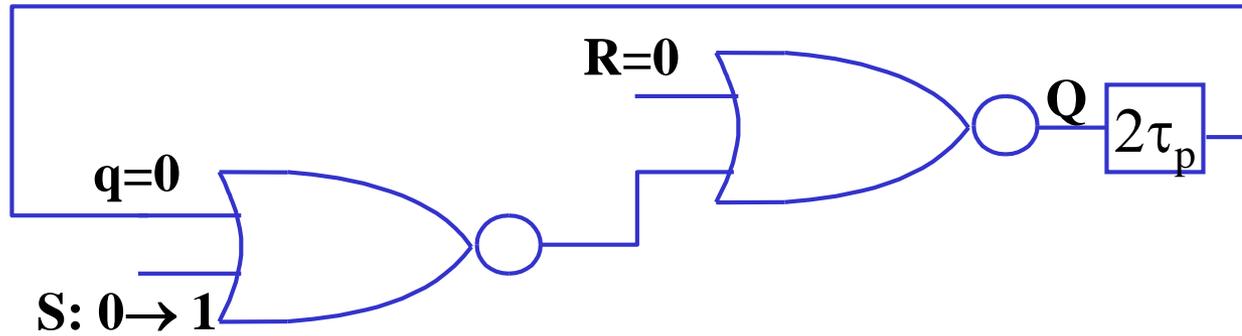


$$= R \downarrow (S \downarrow y)$$

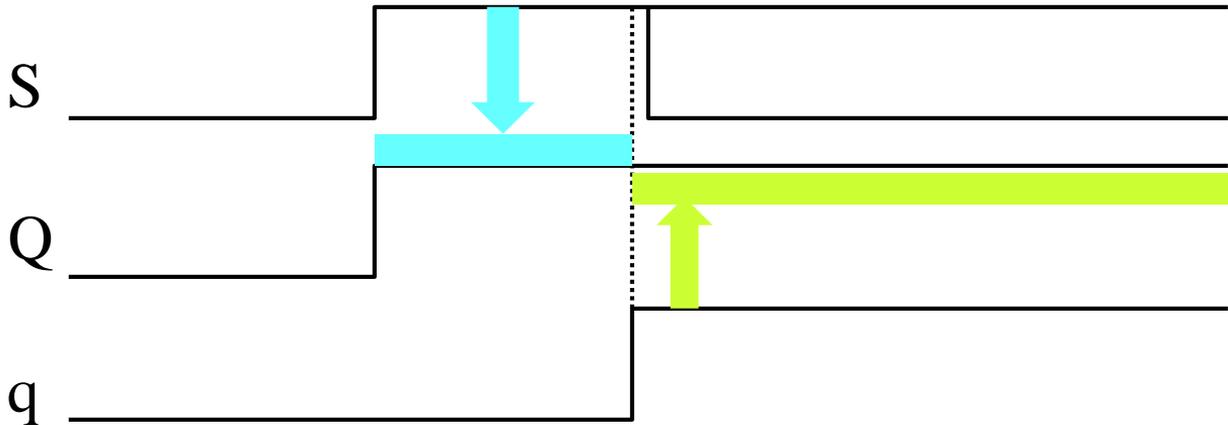
$$Q = y$$



Cosa occorre per scrivere un 1 (o uno 0)?

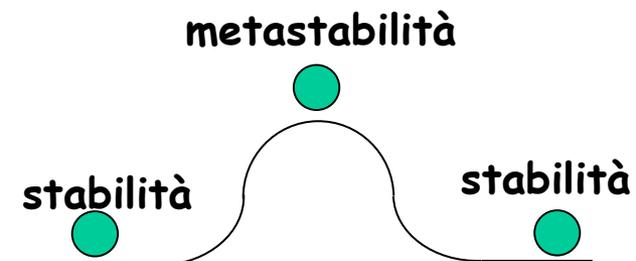


$$\Delta t = 2\tau_p$$



Dopo Δt dal fronte di salita di S , q passa da 0 a 1, condizione che si mantiene anche se S torna a 0

- La durata minima di un comando di set/reset (τ_w) è indicata dal costruttore sul data sheet (es. prossima slide)
- Se l'impulso di set/reset ha durata inferiore a tale valore minimo il latch può andare in **metastabilità**
- Condizione da rispettare è dunque: $\tau_w \geq 2 \tau_p$



SN54279, SN54LS279A, SN74279, SN74LS279A QUADRUPLE S-R LATCHES

SDLS093 – DECEMBER 1983 – REVISED MARCH 1988

recommended operating conditions

	SN54279			SN74279			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
V _{CC} Supply voltage	4.5	5	5.5	4.75	5	5.25	V
V _{IH} High-level input voltage	2			2			V
V _{IL} Low-level input voltage			0.8			0.8	V
I _{OH} High-level output current			-0.8			-0.8	mA
I _{OL} Low-level output current			16			16	mA
t _w Pulse duration, low	20			20			ns
T _A Operating free-air temperature	-55		125	0		70	°C

switching characteristics, V_{CC} = 5 V, T_A = 25°C (see note 3)

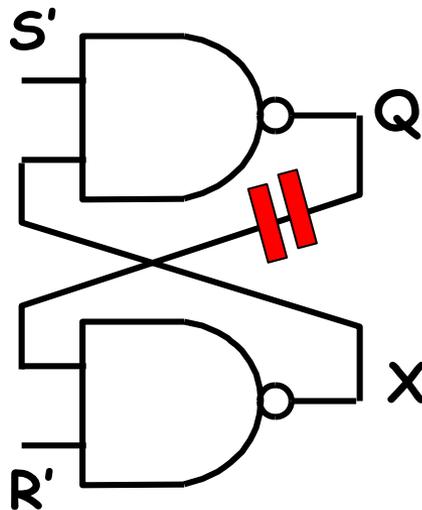
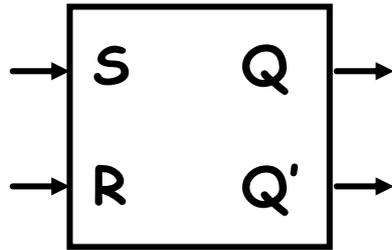
PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	MIN	TYP	MAX	UNIT
t _{PLH}	S	Q	R _L = 400 Ω, C _L = 15 pF		12	22	ns
t _{PHL}					9	15	
t _{PHL}	R	Q				15	27

NOTE 3: Load circuits and voltage waveforms are shown in Section 1.



Uscite complementari

- Verifichiamo, tramite procedimento di analisi, che lo schema a NAND del latch SR (analogamente quello a NOR, e a differenza dello schema a AND,OR) mette a disposizione il bit Q anche in forma negata senza necessità di impiego di un NOT: ovvero che $X=Q'$



SR

	00	01	11	10
0	0,01	0,01	1,11	1,11
1	1,10	0,01	1,11	1,10

Y, QX

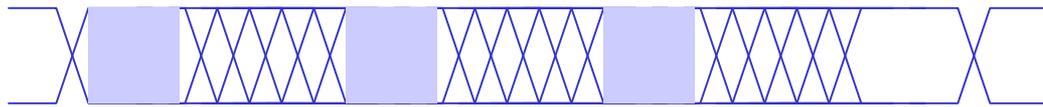
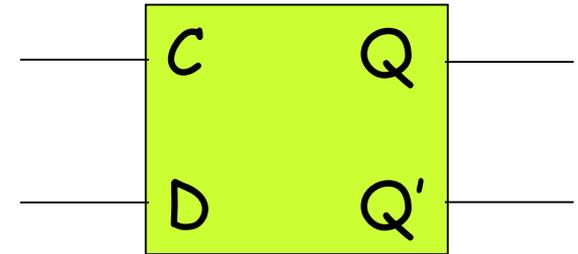
$$Y = S + R' \cdot y \quad Q = y \quad (\text{eq. caratteristica SP})$$

$$X = R' \uparrow y = R + y' = Q'$$

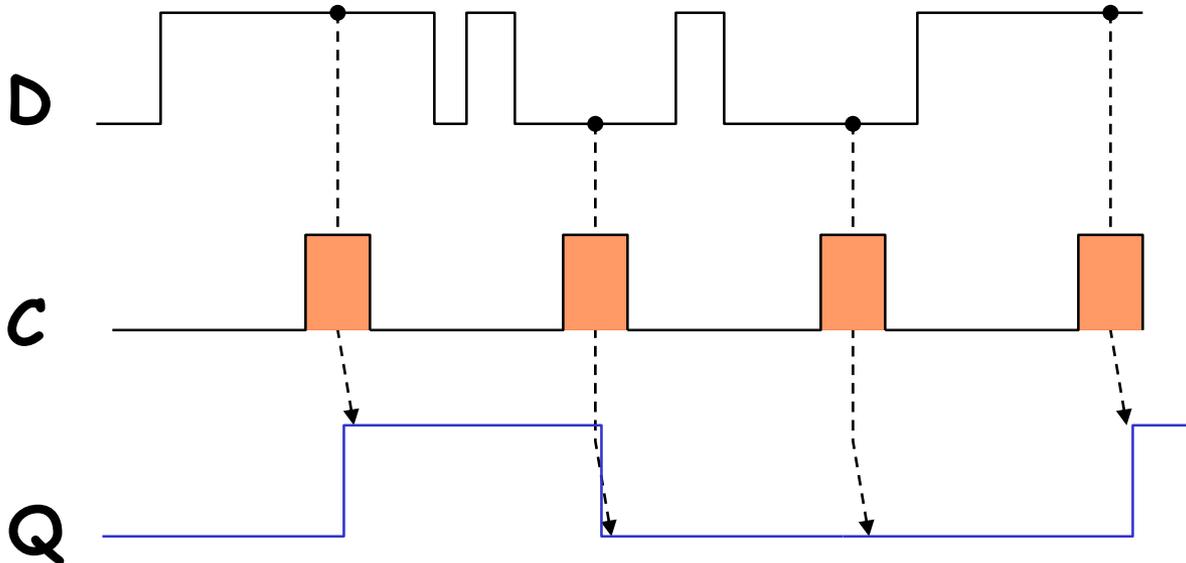
- Dunque $X \approx Q'$ considerando che l'ingresso 11 non è consentito e che con ingresso 10 lo stato $y=0$ non è stabile.

Latch CD (o D-Latch)

- memoria binaria in cui il valore 1 di un segnale **C** individua "quando scrivere", il valore contemporaneo di un segnale **D** "cosa scrivere".
- è il campionatore di segnale «level-triggered»
- la forma d'onda in ingresso (D) è «ripulita» da andamenti indesiderati in transitorio
- componente di memoria all'interno del **registro buffer** e della **memoria RAM**



Segnale con valori significativi solo in certi intervalli

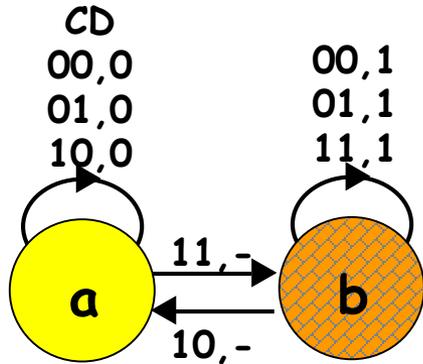


Es: segnale con "glitch"

segnale di campionamento "attivo alto"

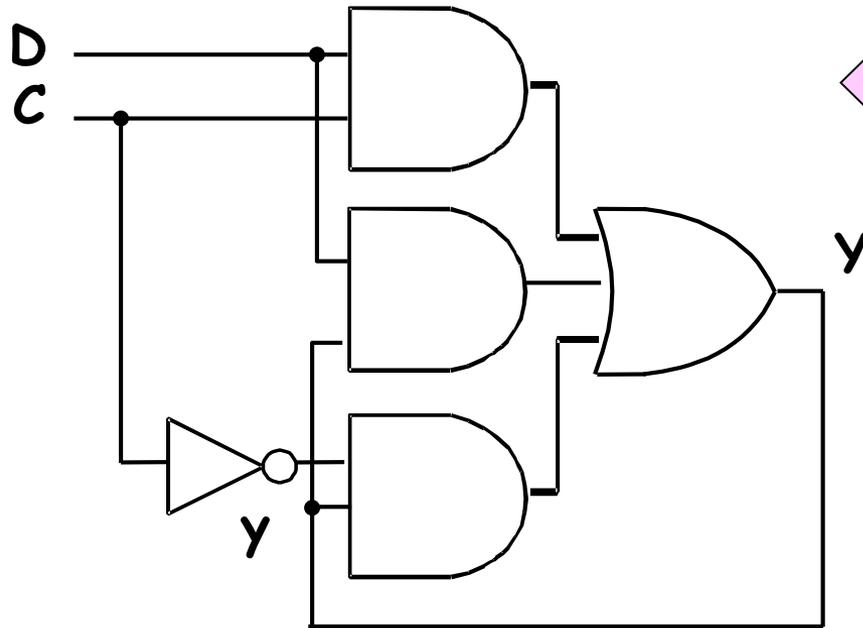
segnale "ricostruito" dal latch

Tabella delle transizioni, espressioni logiche



		CD			
		00	01	11	10
y	0	0,0	0,0	1,-	0,0
	1	1,1	1,1	1,1	0,-
		y,Q			

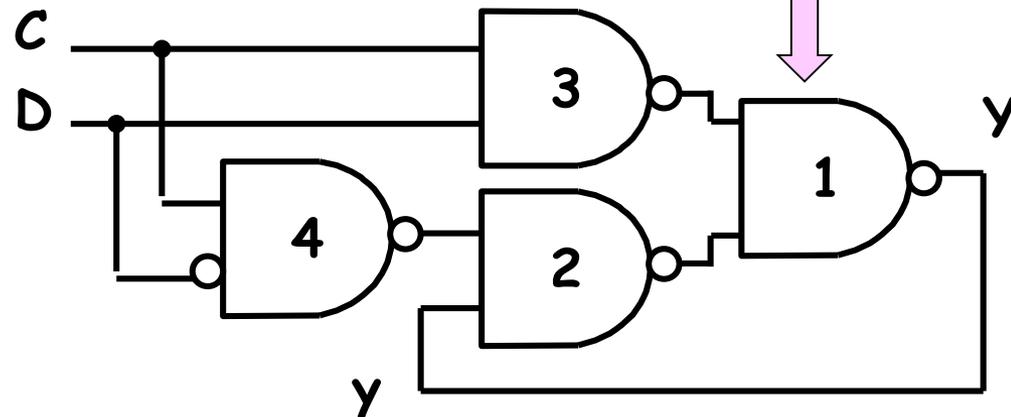
$$Q = y$$



$$y = C D + C' y + D y \quad \text{Eq. caratt. SP}$$

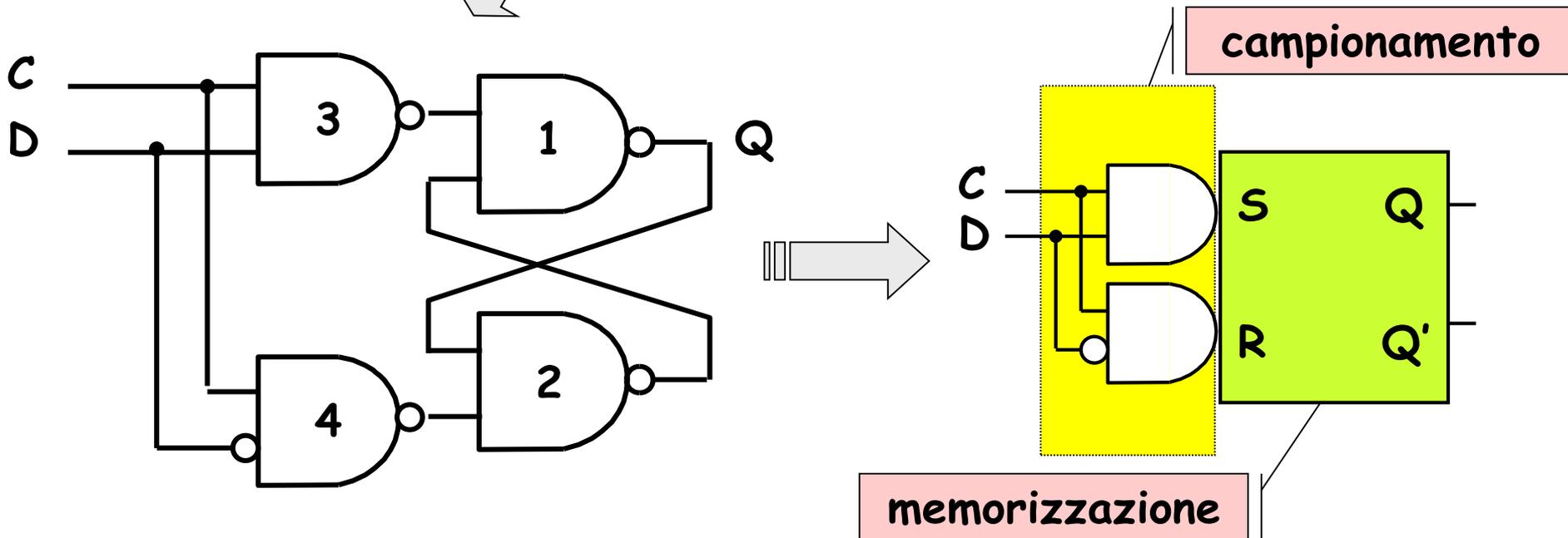
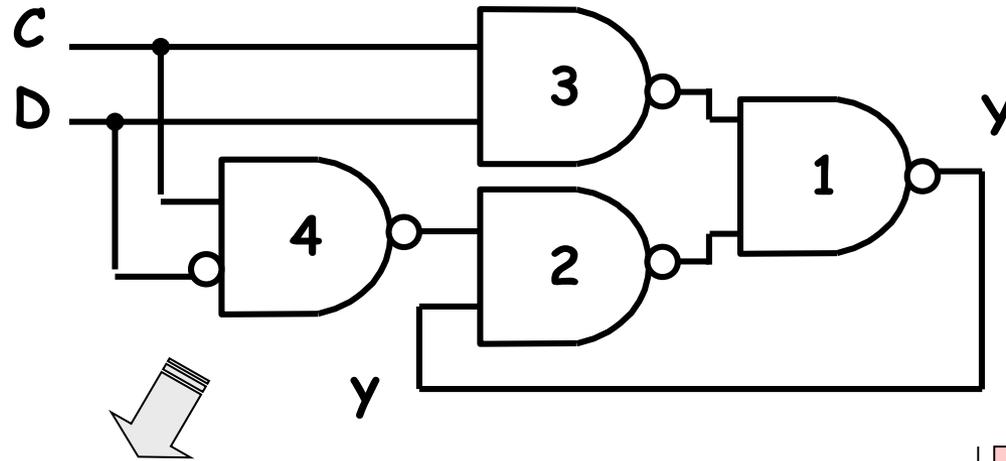
$$= C D + y (C' + D)$$

$$= (C \uparrow D) \uparrow (y \uparrow (C \uparrow D'))$$



- le equazioni caratteristiche (SP e PS) includono, come già visto, un termine ridondante per l'eliminazione dell'alea statica

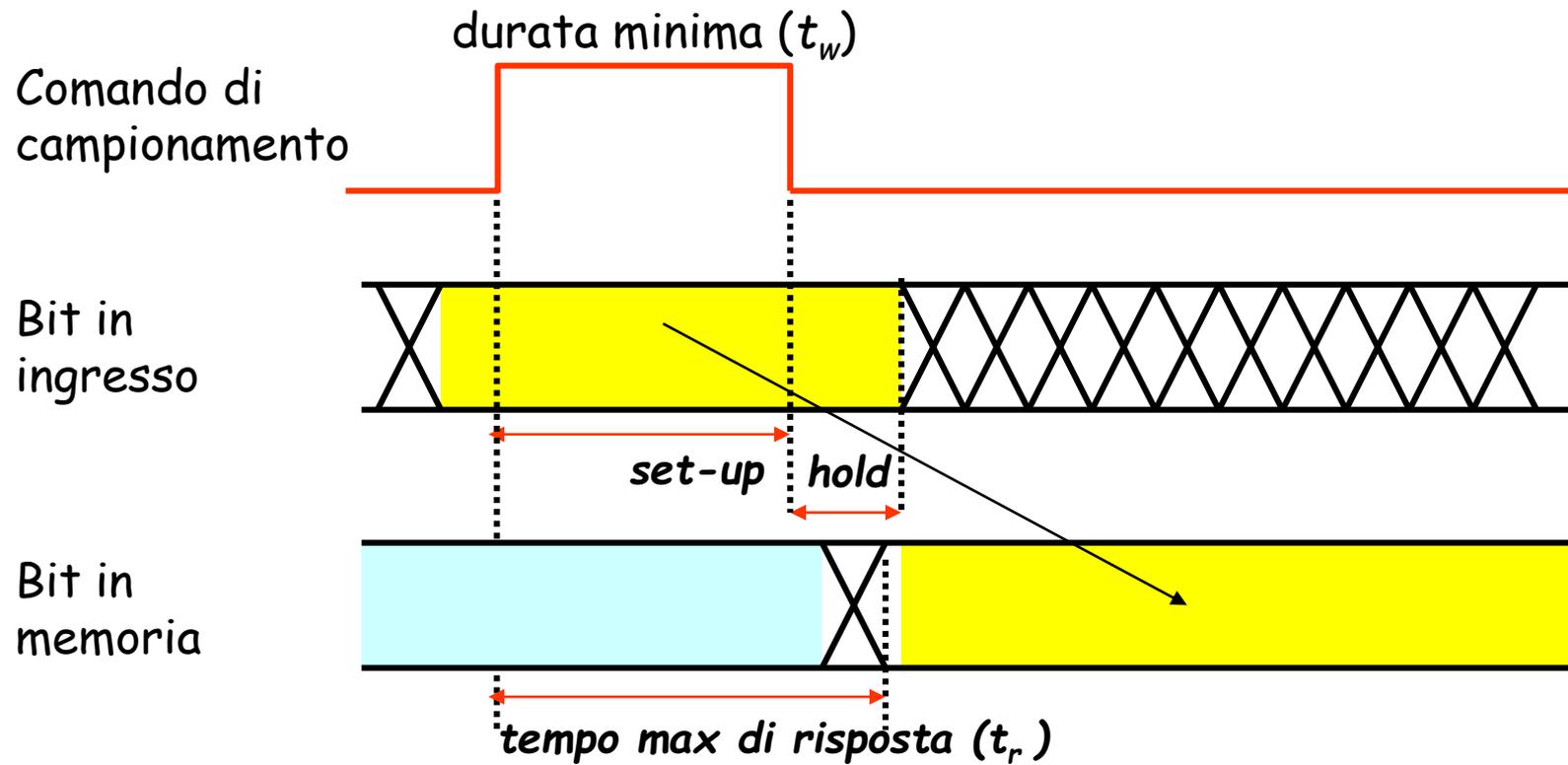
Schema logico



- Per la realizzazione del latch CD si può impiegare un latch SR preceduto da una trascodifica da codice CD a codice SR (tramite due AND)
- Il primo stadio (i due AND) campiona il valore di D quando $C=1$, il secondo stadio (latch SR) lo memorizza mentre $C=0$

I tempi di *set-up*, di *hold* e di *risposta*

- Avendo più gate in cascata, il transitorio del latch CD dura più di quello del latch SR
- Vi sono due tempi da considerare: *set-up* (t_{su} , tempo di propagazione attraverso i gate) e *hold* (t_h , tempo necessario per innescare la retroazione)
- D deve essere costante durante $t_{su} + t_h$
- Il comando di campionamento ($C=1$) deve rispettare: $t_w \geq t_{su}$ (altrimenti: rischio metastabilità)
- Il tempo max di risposta (response time, t_r) viene indicato nel data sheet (specificando se da L a H o da H a L, v. prossima slide)



SN5475, SN5477, SN54LS75, SN54LS77 SN7475, SN74LS75 4-BIT BISTABLE LATCHES

SDLS120 – MARCH 1974 – REVISED MARCH 1988

recommended operating conditions

	SN5475, SN5477			SN7475			UNIT
	MIN	NOM	MAX	MIN	NOM	MAX	
Supply voltage, V_{CC}	4.5	5	5.5	4.75	5	5.25	V
High-level output current, I_{OH}			-400			-400	μ A
Low-level output current, I_{OL}			16			16	mA
Width of enabling pulse, t_w	20			20			ns
Setup time, t_{SU}	20			20			ns
Hold time, t_H	5			5			ns
Operating free-air temperature, T_A	-55		125	0		70	$^{\circ}$ C

switching characteristics, $V_{CC} = 5$ V, $T_A = 25^{\circ}$ C

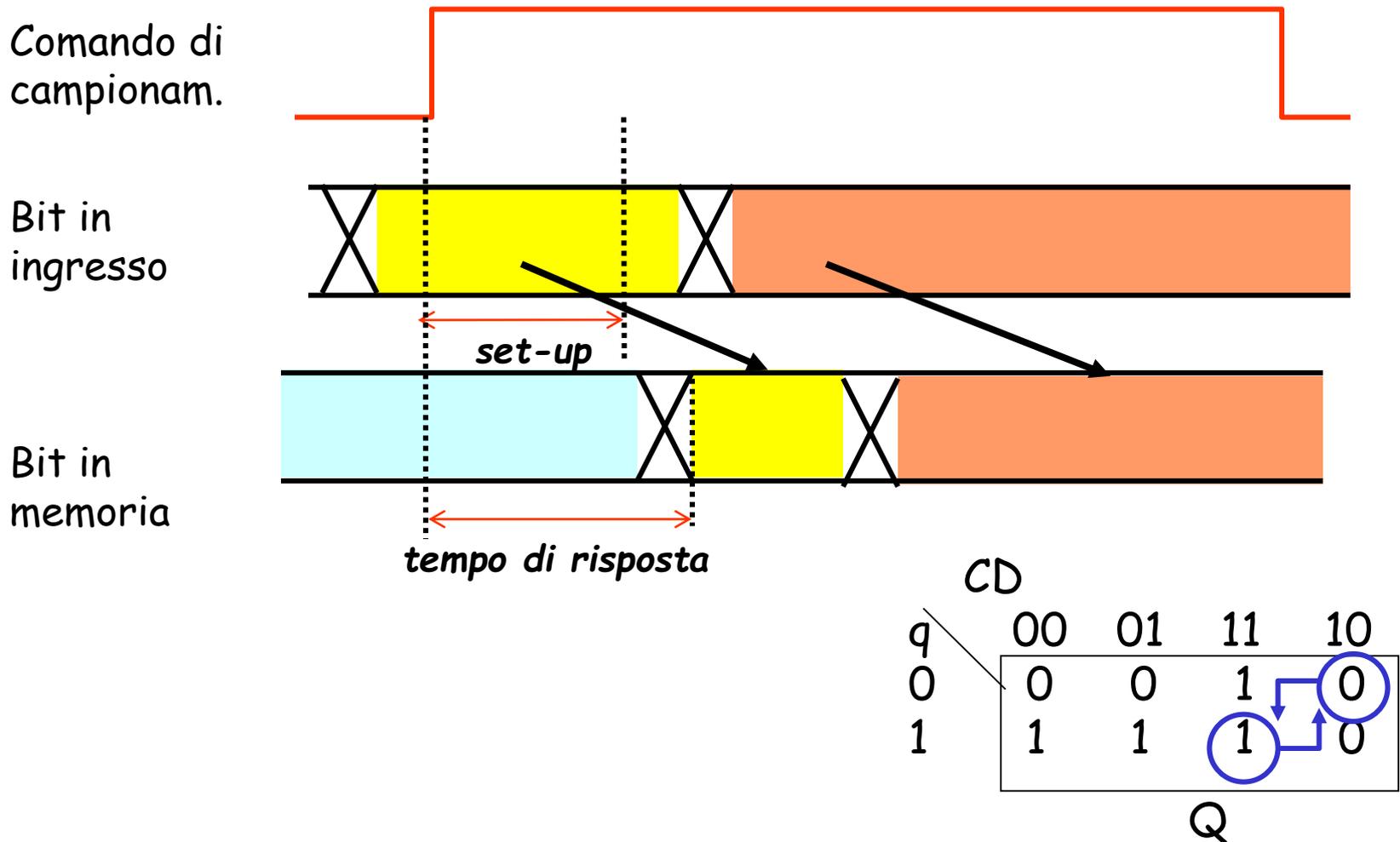
PARAMETER	FROM (INPUT)	TO (OUTPUT)	TEST CONDITIONS	MIN	TYP	MAX	UNIT
t_{PLH}	D	Q	$C_L = 15$ pF, $R_L = 400$ Ω , See Figure 1		16	30	ns
t_{PHL}					14	25	
t_{PLH}^{\dagger}	D	\bar{Q}			24	40	ns
t_{PHL}^{\dagger}					7	15	
t_{PLH}	C	Q			16	30	ns
t_{PHL}					7	15	
t_{PLH}^{\dagger}	C	\bar{Q}			16	30	ns
t_{PHL}^{\dagger}					7	15	

t_{PLH} \equiv propagation delay time, low-to-high-level output

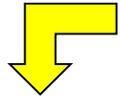
t_{PHL} \equiv propagation delay time, high-to-low-level output

L'uscita "trasparente"

- Se il segnale C resta a 1 per più della durata minima e se durante questo intervallo D modifica il suo valore, allora le stesse modifiche si riscontrano anche su Q (uscita «trasparente»)
- In questa circostanza, la rete commuta tra due stabilità della TdT



I Flip-Flop D

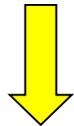


$\alpha \equiv \{A, E, G\}$

$\beta \equiv \{B\}$

$\gamma \equiv \{C\}$

$\delta \equiv \{D, F, H\}$



CD

	00	01	11	10
α	$\alpha, 0$	$\beta, 0$	$\alpha, 0$	$\alpha, 0$
β	$\alpha, 0$	$\beta, 0$	$\delta, -$	$-$
γ	$\gamma, 1$	$\delta, 1$	$-$	$\alpha, -$
δ	$\gamma, 1$	$\delta, 1$	$\delta, 1$	$\delta, 1$

Flip-Flop D Master-Slave

	00	01	11	10
A	A, 0	B, 0	-	E, 0
B	A, 0	B, 0	F, -	-
C	C, 1	D, 1	-	E, -
D	C, 1	D, 1	F, 1	-
E	A, 0	-	G, 0	E, 0
F	-	D, 1	F, 1	H, 1
G	-	B, 0	G, 1	E, 0
H	C, 1	-	F, 1	H, 1

$a \equiv \{A, E, G\}$

$b \equiv \{A, B\}$

$c \equiv \{C, D\}$

$d \equiv \{D, F, H\}$



CD

	00	01	11	10
a	a/b, 0	b, 0	a, 0	a, 0
b	a/b, 0	b, 0	d, -	a, 0
c	c, 1	c/d, 1	d, 1	a, -
d	c, 1	c/d, 1	d, 1	d, 1



$1 \equiv \{A, B\}$

$2 \equiv \{E, G\}$

$3 \equiv \{C, D\}$

$4 \equiv \{F, H\}$



CD

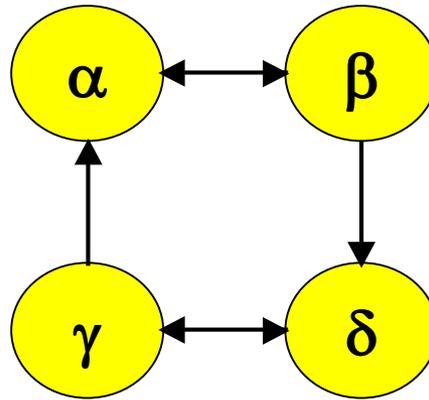
	00	01	11	10
1	1, 0	1, 0	4, -	2, 0
2	1, 0	1, 0	2, 0	2, 0
3	3, 1	3, 1	4, 1	2, -
4	3, 1	3, 1	4, 1	4, 1

Flip-Flop D Edge-Triggered

II Flip-Flop Master-Slave

CD

	00	01	11	10
α	$\alpha, 0$	$\beta, 0$	$\alpha, 0$	$\alpha, 0$
β	$\alpha, 0$	$\beta, 0$	$\delta, -$	$-, -$
γ	$\gamma, 1$	$\delta, 1$	$-, -$	$\alpha, -$
δ	$\gamma, 1$	$\delta, 1$	$\delta, 1$	$\delta, 1$



Y_1

	0	1
0	α	β
1	γ	δ

CD

$Y_1 Y_2$

	00	01	11	10
00	$00, 0$	$01, 0$	$00, 0$	$00, 0$
01	$00, 0$	$01, 0$	$11, -$	$-, -$
11	$10, 1$	$11, 1$	$11, 1$	$11, 1$
10	$10, 1$	$11, 1$	$-, -$	$00, -$

$Q = Y_1$

$Y_1 Y_2, Q$

$Y_1 = C Y_2 + C' Y_1 + Y_1 Y_2$

$= (C \uparrow Y_2) \uparrow (Y_1 \uparrow (C \uparrow Y_2'))$

$Y_2 = C' D + C Y_2 + D Y_2$

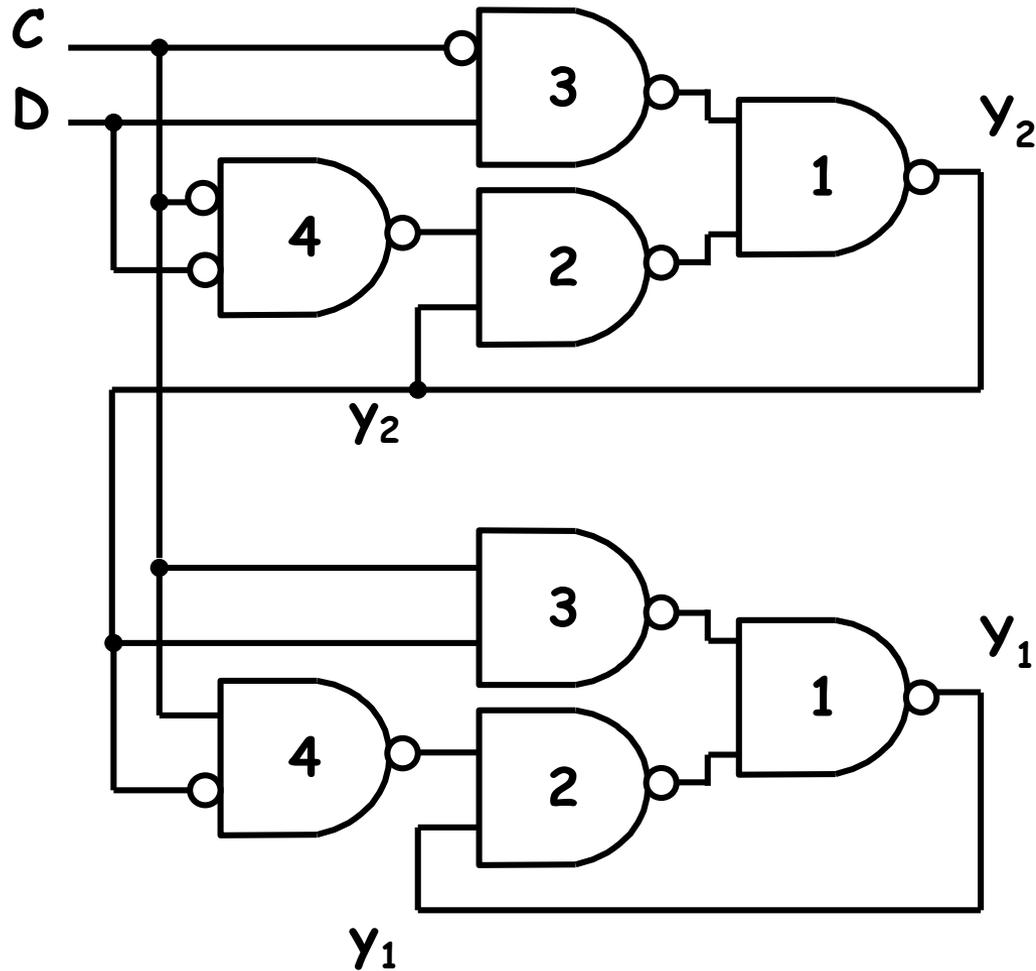
$= (C' \uparrow D) \uparrow (Y_2 \uparrow (C' \uparrow D'))$

(ridondanza per risolvere l'alea statica)

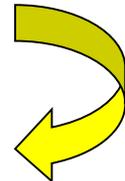
Schema logico ...

$$Y_1 = (C \uparrow y_2) \uparrow (y_1 \uparrow (C \uparrow y_2'))$$

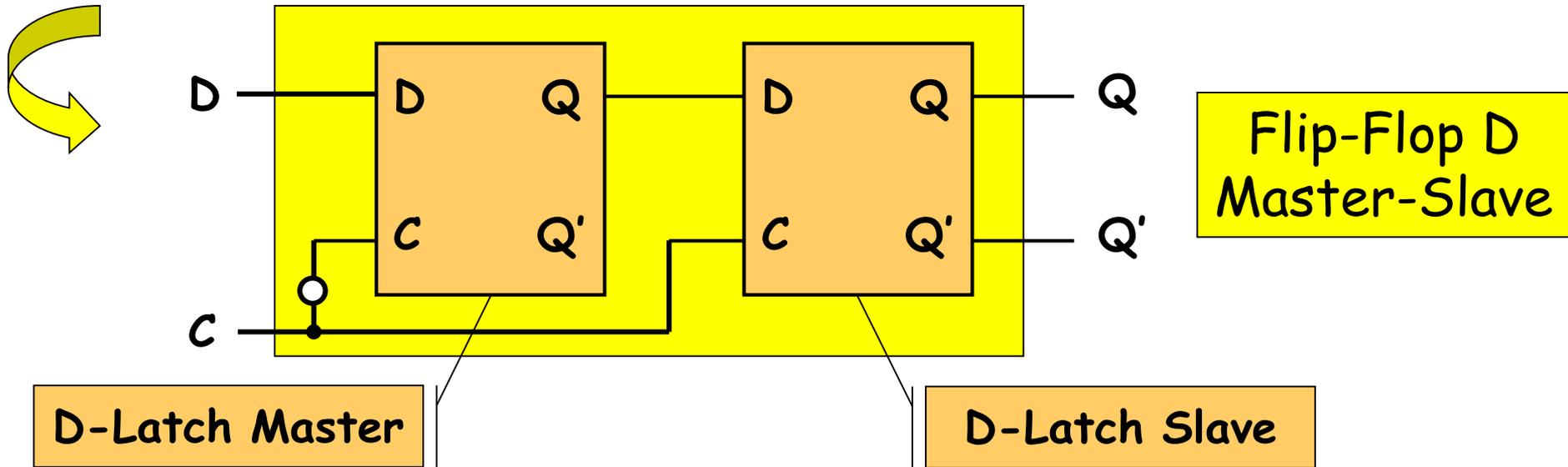
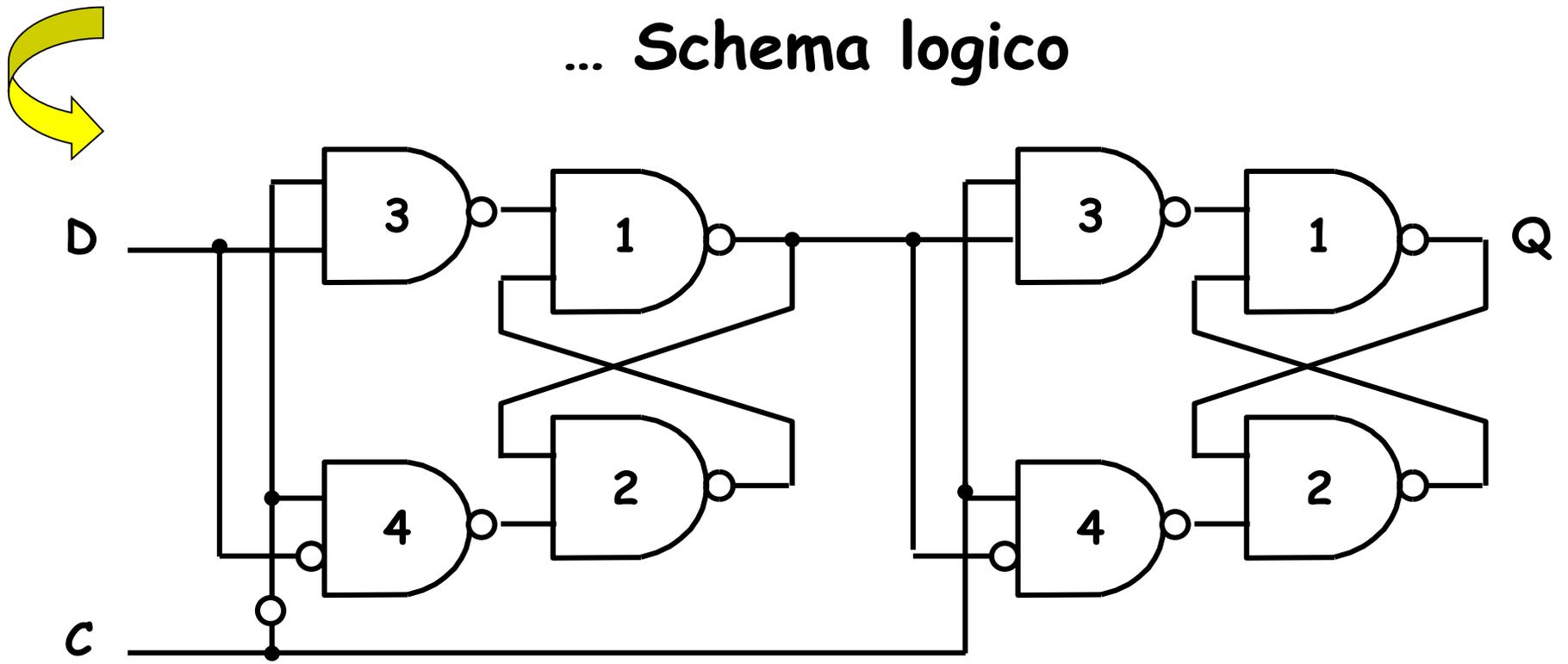
$$Y_2 = (C' \uparrow D) \uparrow (y_2 \uparrow (C' \uparrow D'))$$



$$Q = Y_1$$

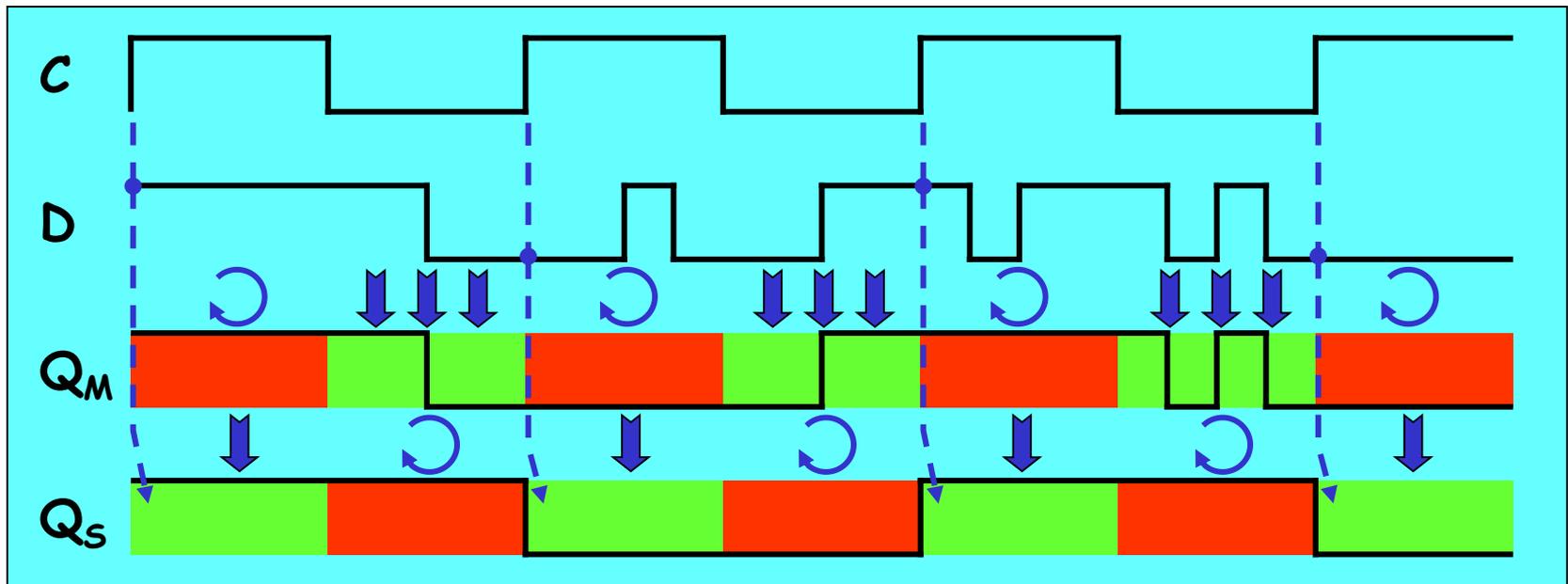
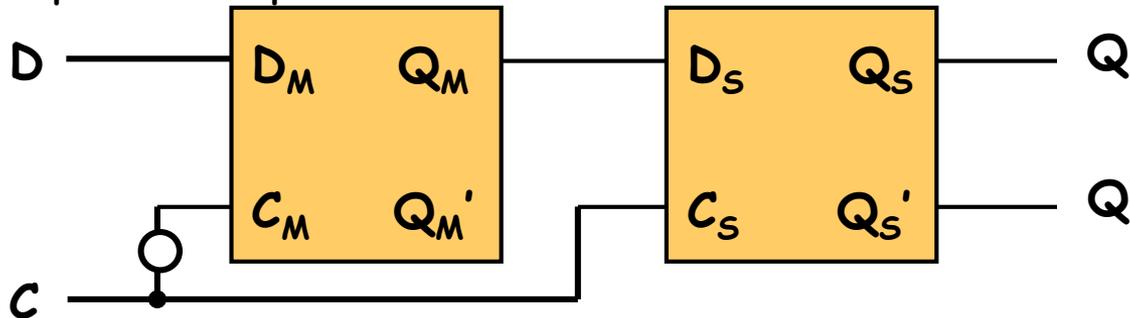


... Schema logico



Il Flip-Flop D Master-Slave ...

- Realizzabile tramite due Latch CD in cascata con C in forma vera e negata
- Con $C=0$, solo Q_M riproduce le variazioni di D (Q_S memorizza il valore precedente)
- Quando C passa da 0 a 1 (fronte di salita), il Master memorizza l'ultimo valore di D e lo consegna per il campionamento allo Slave

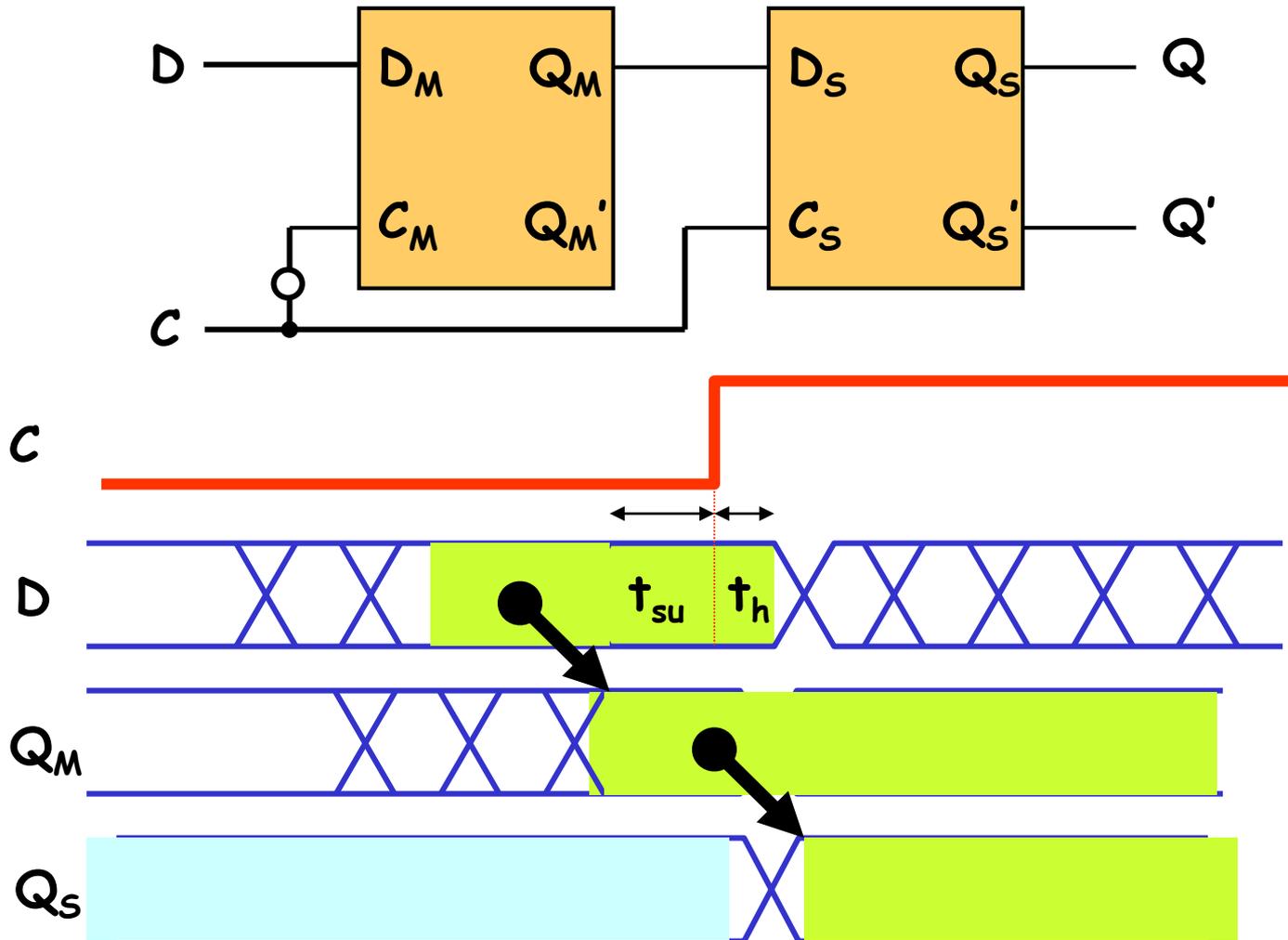


 campionamento

 memorizzazione

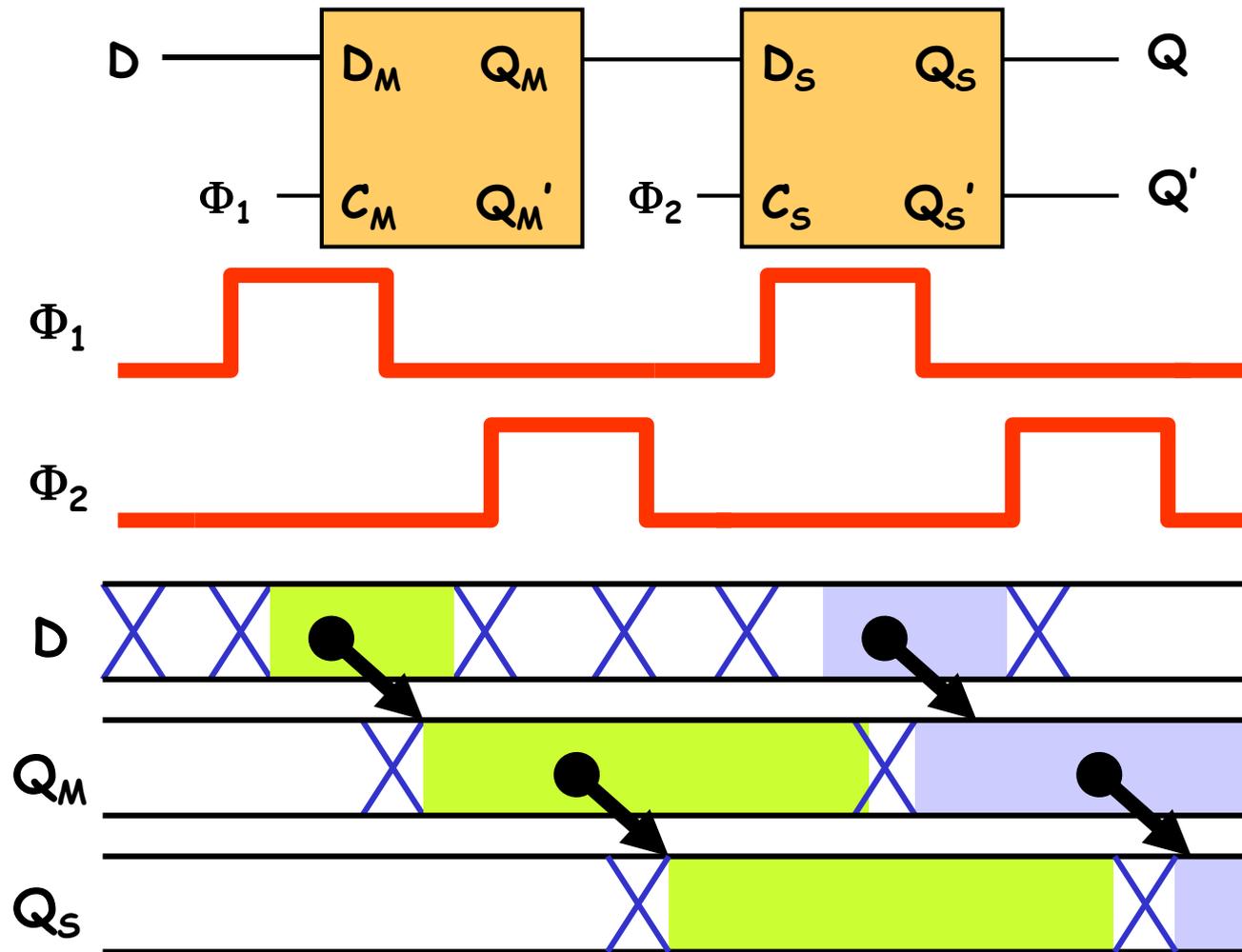
... Il Flip-Flop D Master-Slave

- Al contrario del Latch CD, non presenta più il difetto dell'uscita trasparente: Q si può modificare solo in corrispondenza dei fronti di salita di C



Il clock a due fasi

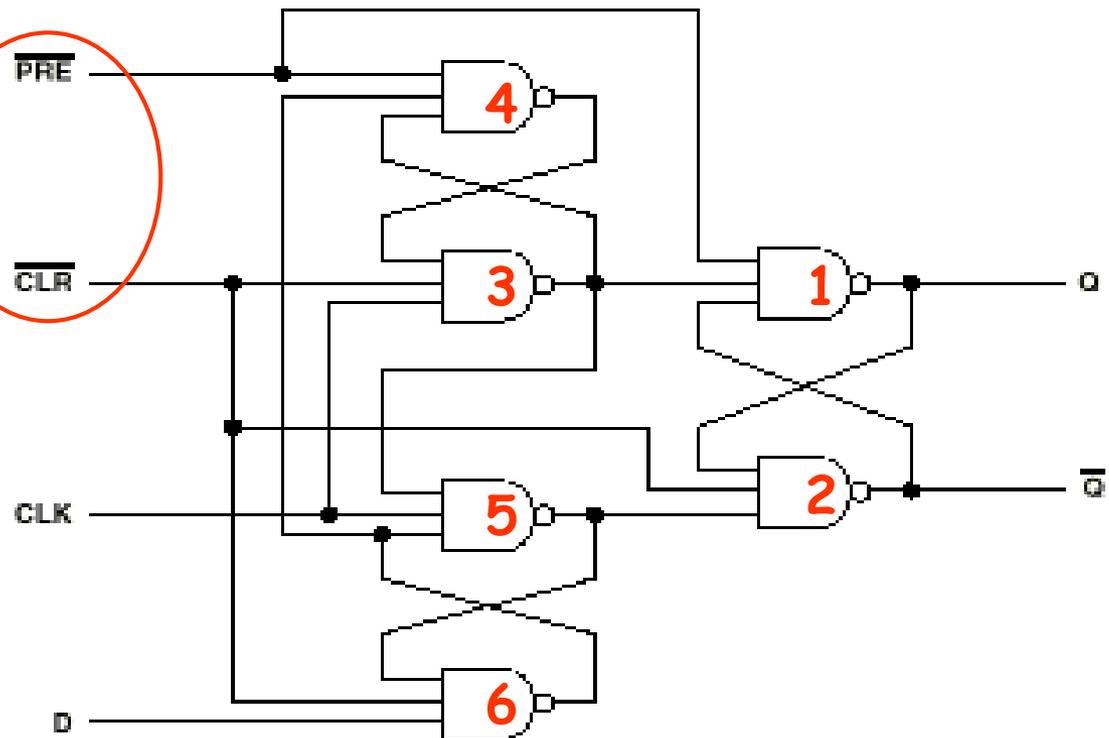
- Il NOT sul segnale C è pericoloso (come ogni *clock-gating*)
- Infatti, esso introduce un ritardo che sfasa i due segnali C dei due latch, che può far sì che si propaghi su Q il valore scorretto di D
- Per evitare questo fenomeno, al suo posto si utilizzano due segnali distinti (Φ_1 , Φ_2) perfettamente sincronizzati tra loro (**clock a due fasi**)



Il flip-flop D "Edge-Triggered"

- Questo Flip-Flop è stato ottenuto tramite CAD (contiene scelte di progetto non intuitive, come l'utilizzo di 3 variabili di stato anziché 2)
- Vantaggi rispetto al Flip-Flop Master-Slave: 6 NAND (anziché 8), no clock-gating, D in ingresso a un solo NAND
- Possiede due ulteriori **comandi asincroni**, prioritari rispetto a C e D: PRE (preset) e CLR (clear)
- Eseguiamo l'analisi partendo dallo schema circuitale

logic diagram (positive logic)



comandi asincroni:

Clear ($CLR' = 0$) $\rightarrow Q = 0$

Preset ($PRE' = 0$) $\rightarrow Q = 1$

Lo schema logico secondo il modello di riferimento

rete combinatoria di uscita

rete combinatoria di aggiornamento dello stato

$$Q = y_1$$

$$y_1 = y_2 \uparrow (y_1 \uparrow y_3)$$

$$y_2 = C \uparrow (y_2 \uparrow (D \uparrow y_3))$$

$$y_3 = C \uparrow y_2 \uparrow (D \uparrow y_3)$$

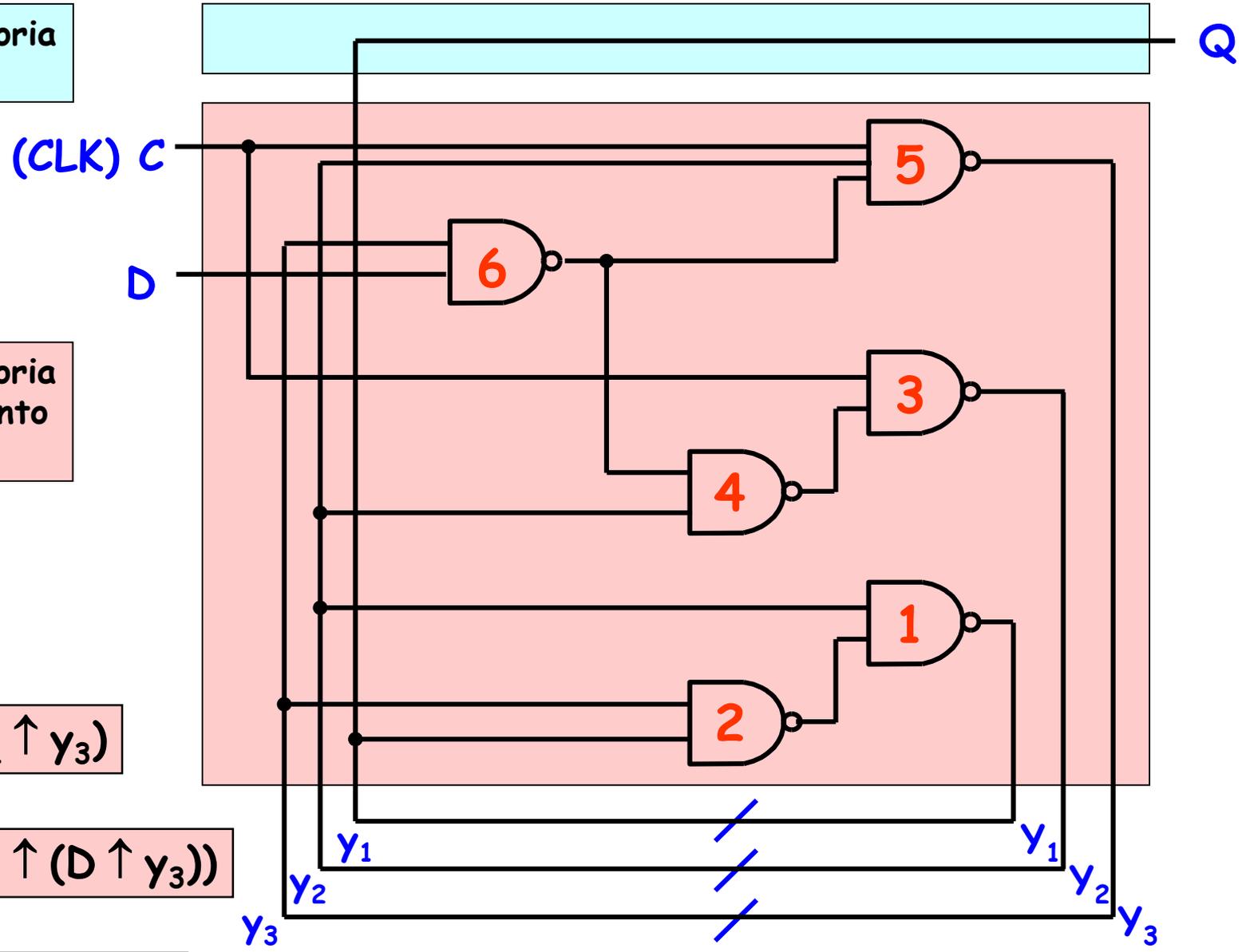


Tabella delle transizioni

$$Q = y_1$$

Espressioni ↑

$$y_1 = y_2 \uparrow (y_1 \uparrow y_3)$$

$$y_2 = C \uparrow (y_2 \uparrow (D \uparrow y_3))$$

$$y_3 = C \uparrow y_2 \uparrow (D \uparrow y_3)$$

Espressioni SP

$$y_1 = y_2' + y_1 y_3$$

$$y_2 = C' + y_2 (D' + y_3')$$

$$y_3 = C' + y_2' + D y_3$$

		CD			
		00	01	11	10
$y_1 y_2 y_3$	000	111,0	111,0	101,0	101,0
	001	111,0	111,0	101,0	101,0
	011	011,0	011,0	001,0	010,0
	010	011,0	011,0	010,0	010,0
	100	111,1	111,1	101,1	101,1
	101	111,1	111,1	101,1	101,1
	111	111,1	111,1	101,1	110,1
	110	011,1	011,1	010,1	010,1

$y_1 y_2 y_3, Q$

stati

transizioni

stabili

dirette

instabili

multiple

Tabella di flusso, grafo degli stati

		CD			
		00	01	11	10
$Y_1Y_2Y_3$	000	111,0	111,0	101,0	101,0
	001	111,0	111,0	101,0	101,0
	011	011,0	011,0	001,0	010,0
	010	011,0	011,0	010,0	010,0
	100	111,1	111,1	101,1	101,1
	101	111,1	111,1	101,1	101,1
	111	111,1	111,1	101,1	110,1
	110	011,1	011,1	010,1	010,1

		CD			
		00	01	11	10
1	1,0	1,0	4,-	2,0	
2	1,0	1,0	2,0	2,0	
3	3,1	3,1	4,1	2,-	
4	3,1	3,1	4,1	4,1	

$Y_1Y_2Y_3$

$Y_1Y_2Y_3, Q$

stati

stabili

instabili

transizioni

dirette

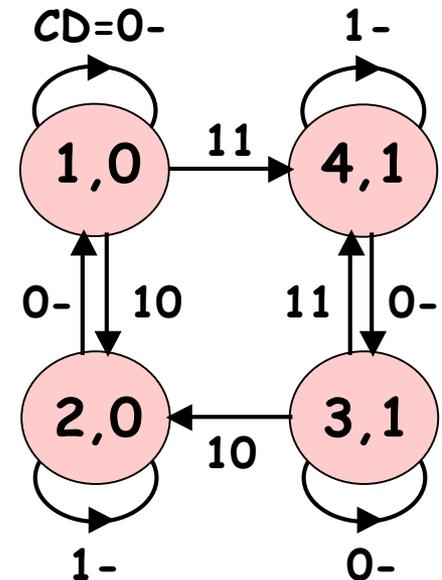
multiple

011 → 1

010 → 2

111 → 3

101 → 4



I tempi di *set-up*, di *hold* e di *risposta*

- D deve rimanere costante sia durante t_{su} , sia durante t_h
- t_r : tempo max di durata del transitorio sulle uscite (Q e Q')
- Transitorio più lungo di quello del latch CD (in quanto il percorso è formato da più gate in cascata)

