



Università degli studi di Parma

Dipartimento di Ingegneria dell'Informazione

Sintesi Sequenziale Sincrona

Sintesi di Reti Sequenziali a Livello Comportamentale

Sintesi comportamentale

Architettura generale

Diagramma degli stati

Tabella degli stati

Tabella delle Transizioni: Codifica dello Stato

Tabella delle Eccitazioni: Scelta degli elementi di memoria

Trasformazioni tra FF



■ Progetto di reti combinatorie

- ▶ I metodi sono noti e ben assestati
- ▶ Si ricercano nuove soluzioni che aumentino l'efficienza
 - Computazionale degli strumenti automatici
 - Nell'uso delle risorse (es. BDD)
 - Metodologica (es. Signature Cube - '92)

■ Progetto di reti sequenziali

- ▶ Le metodologie di ottimizzazione di circuiti sequenziali sono in costante evoluzione
 - Esistono buoni metodi ma non di uso generale
- ▶ Lo sviluppo di algoritmi e tool efficienti necessita di ulteriori sforzi



- Il modello di un circuito sincrono può essere
- Comportamentale
 - ▶ Le transizioni tra gli stati sono descritte in termini di tabelle o diagrammi
 - ▶ Le informazioni sugli stati sono esplicite
 - ▶ Le informazioni sull'area e sui ritardi sono implicite
- Strutturale
 - ▶ Il modello del circuito è una netlist ovvero un insieme di componenti, registri e logica combinatoria, opportunamente interconnessi
 - ▶ Le informazioni sugli stati sono implicite
 - ▶ Le informazioni sull'area e ritardi sono esplicite



- Il valore delle uscite all'istante t dipende dalla successione degli ingressi che precedono l'istante t
 - ▶ Implica il concetto di stato
- Una FSM è definita dalla quintupla $(I, U, S, \delta, \lambda)$
 - ▶ I - Alfabeto di Ingresso
 - E' costituito dall'insieme finito dei simboli di ingresso
 - Con n linee di ingresso si hanno 2^n simboli
 - ▶ U - Alfabeto d'Uscita
 - E' costituito dall'insieme finito dei simboli d'uscita
 - Con m linee d'uscita si hanno 2^m simboli
 - ▶ S - Insieme degli Stati
 - Insieme finito e non vuoto degli stati
 - ▶ δ - Funzione stato prossimo
 - ▶ λ - Funzione d'uscita



■ Funzione stato prossimo δ

- ▶ Ad ogni stato presente e per ogni simbolo di ingresso la funzione δ associa uno stato futuro

$$\delta : S \times I \rightarrow S$$

- ▶ Ad ogni coppia $\{\text{stato}, \text{simbolo di ingresso}\}$ è associato, se specificato, uno ed uno solo stato futuro

■ Funzione d'uscita λ

- ▶ Genera il simbolo d'uscita
- ▶ Macchine di Mealy (l'uscita dipende da stato e ingresso)

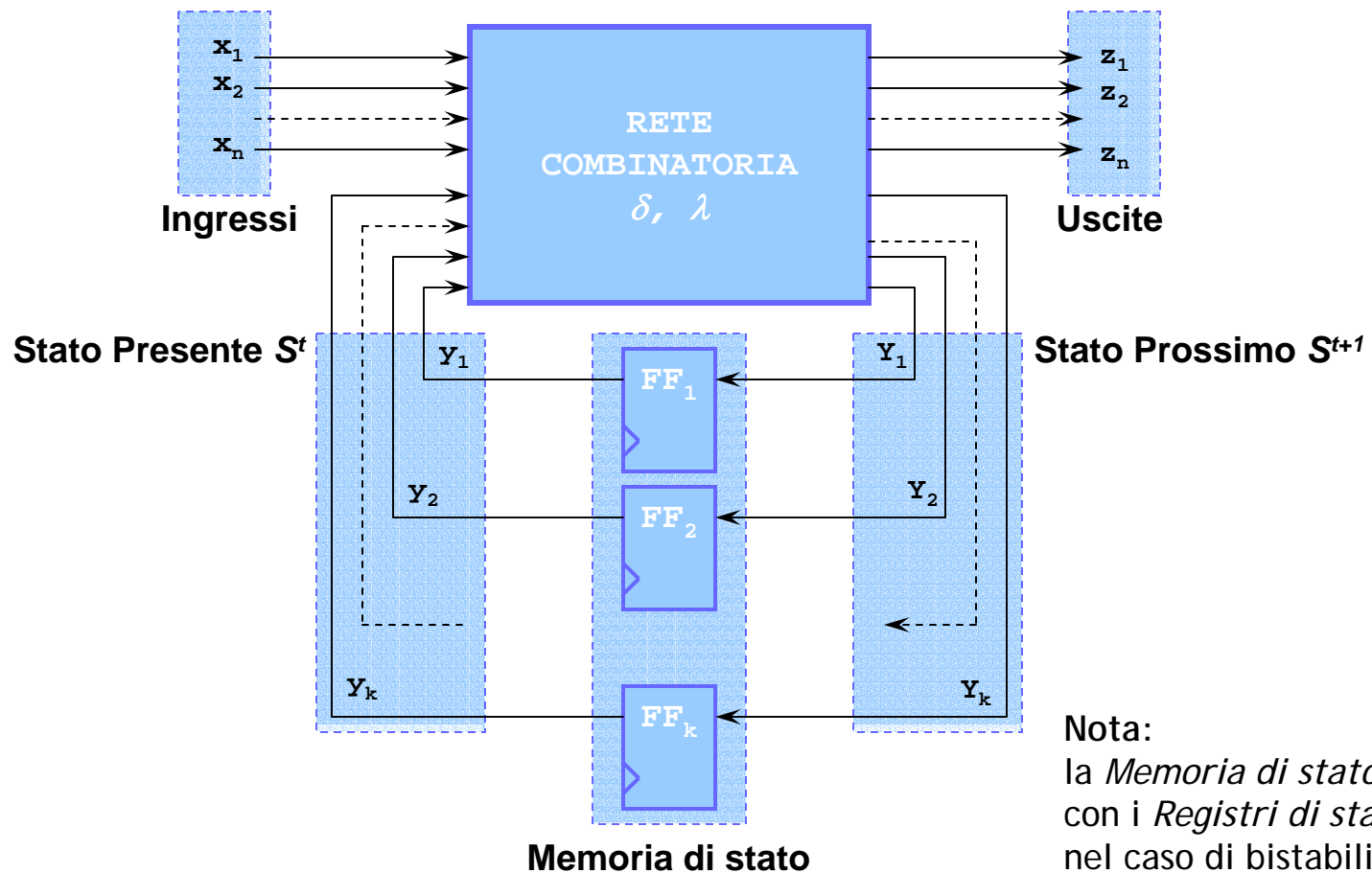
$$\lambda : S \times I \rightarrow U$$

- ▶ Macchine di Moore (l'uscita dipende solamente dallo stato)

$$\lambda : S \rightarrow U$$



■ Struttura generale di una macchina sequenziale:





- Il problema della sintesi comportamentale di una rete sequenziale consiste nella
 - ▶ Identificazione delle le funzioni δ e λ
 - ▶ Sintesi della rete combinatoria che le realizza
- Gli elementi di memoria sono costituiti da Flip-Flop
 - ▶ I flip-flop di tipo D sono quelli usati più comunemente
- La funzione di stato prossimo
 - ▶ Dipende dal tipo di bistabili utilizzati per la realizzazione della memoria di stato
- La funzione di uscita
 - ▶ Non dipende dal tipo di bistabili utilizzati per la realizzazione della memoria di stato



- Una Macchina a Stati Finiti può essere descritta mediante la Tabella degli stati
 - ▶ Indici di colonna sono i simboli di ingresso $i_\alpha \in I$
 - ▶ Indici di riga sono i simboli di stato $s_j \in S$ ed indicano lo stato presente
- Elementi sono
 - ▶ Macchine di Mealy: La coppia $\{u_\beta, s_j\}$
 - $u_\beta = \lambda(i_{\alpha'}, s_j)$ è il simbolo di uscita
 - $s_j = \delta(i_{\alpha'}, s_j)$ è il simbolo stato prossimo
 - ▶ Macchine di Moore: Il simbolo stato prossimo s_j
 - $s_j = \delta(i_{\alpha'}, s_j)$ è il simbolo stato prossimo
 - Nelle macchine di Moore i simboli d'uscita sono associati allo stato presente

Tabella degli stati



Macchine di Mealy

	i_1	i_2	\dots
s_1^t	s_j^{t+1} / u_j	s_k^{t+1} / u_k	\dots
s_2^t	s_m^{t+1} / u_m	s_1^{t+1} / u_1	\dots
\dots	\dots	\dots	\dots

Macchine di Moore

	i_1	i_2	\dots	
s_1^t	s_j^{t+1}	s_k^{t+1}	\dots	u_1
s_2^t	s_m^{t+1}	s_1^{t+1}	\dots	u_2
\dots	\dots	\dots	\dots	\dots



- Spesso, la stesura della Tabella degli Stati è preceduta da una rappresentazione grafica ad essa equivalente, denominata Diagramma degli Stati
- Il Diagramma degli stati è un grafo orientato $G(V, E, L)$
 - ▶ V - Insieme dei nodi
 - Ogni nodo rappresenta uno stato
 - Ad ogni nodo è associato un simbolo d'uscita (Moore)
 - ▶ E - Insieme degli archi
 - Ogni arco rappresenta le transizioni di stato
 - ▶ L - Insieme degli:
 - Ingressi e Uscite (Mealy)
 - Ingressi (Moore)

Macchina di Mealy: Esempio



- Questo esempio mostra l'equivalenza delle due rappresentazioni nel caso di una macchina di Mealy

Diagramma degli stati

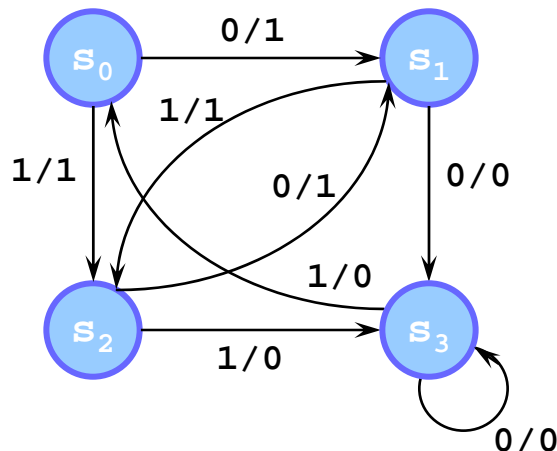


Tabella degli stati

		Ingresso	
		0	1
Stato presente	s ₀	s ₁ /1	s ₂ /1
	s ₁	s ₃ /0	s ₂ /1
	s ₂	s ₁ /1	s ₃ /0
	s ₃	s ₃ /1	s ₀ /0

Stato futuro/ uscita

Macchina di Moore: Esempio



- Questo esempio mostra l'equivalenza delle due rappresentazioni nel caso di una macchina di Moore

Diagramma degli stati

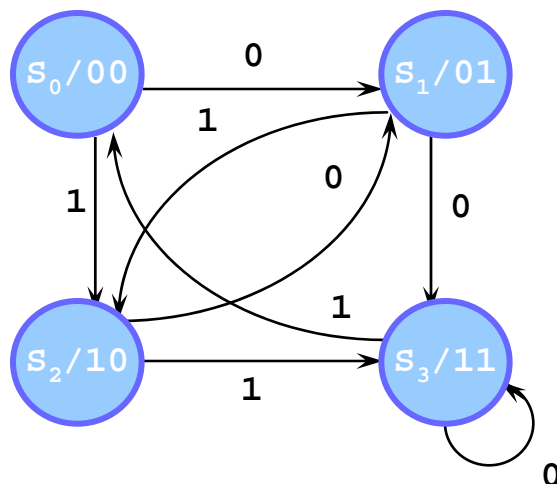


Tabella degli stati

		Ingresso		
		0	1	
Stato presente	s₀	s ₁	s ₂	00
	s₁	s ₃	s ₂	01
	s₂	s ₁	s ₃	10
	s₃	s ₃	s ₀	11
Stato futuro				Uscita



- La sintesi si svolge nei seguenti passi
 - ▶ Realizzazione del diagramma degli stati a partire dalle specifiche informali del problema
 - ▶ Costruzione della tabella degli stati
 - ▶ Riduzione del numero degli stati: ottimizzazione
 - Non considerata in questo contesto
 - ▶ Costruzione della tabella delle transizioni
 - Assegnamento degli stati: Codice & codifica
 - ▶ Costruzione della tabella delle eccitazioni
 - Scelta degli elementi di memoria (tipo)
 - ▶ Sintesi sia della rete combinatoria che realizza la funzione stato prossimo sia della rete combinatoria che realizza la funzione d'uscita

Sintesi: Passi 1 e 2 - Esempio



- Consideriamo una semplice macchina a stati finiti (due stati) come strumento per esporre il processo di sintesi
 - Specifica: realizzare una FSM con due ingressi ed una uscita che abbia il seguente comportamento
 - Configurazione di ingressi 00: l'uscita non cambia valore
 - Configurazione di ingressi 01: l'uscita assume valore 0
 - Configurazione di ingressi 10: l'uscita assume valore 1
 - Configurazione di ingressi 11: l'uscita cambia valore dopo ogni fronte di salita del clock

Diagramma degli stati

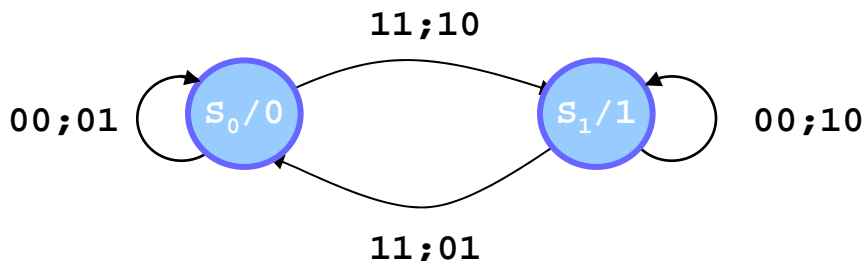


Tabella degli stati

	00	01	11	10	
S ₀	S ₀	S ₀	S ₁	S ₁	0
S ₁	S ₁	S ₀	S ₀	S ₁	1

Sintesi: Codifica degli stati (passo 4)



- Il processo di codifica degli stati ha l'obiettivo di identificare per ogni rappresentazione simbolica dello stato una corrispondente rappresentazione binaria
- La Tabella degli stati è trasformata in Tabella delle Transizioni
 - ▶ Ad un'unica tabella degli stati corrispondono più tabelle delle transizioni
- Due problemi simultanei
 - ▶ Scelta del codice
 - Minima: Numero di elementi di memoria = $\lceil \log_2 |S| \rceil$
 - One-Hot: Numero di elementi di memoria = $|S|$
 - ...
 - ▶ Identificazione della codifica di ogni stato

Sintesi: Codifica degli stati (passo 4)



- Scelto il codice, la codifica degli stati influisce
 - ▶ Sull'area
 - ▶ Sulle prestazioni
- Il problema della identificazione della codifica ottima
 - ▶ È un problema NP-completo
 - ▶ Impone l'uso di euristiche per prevedere l'influenza sul processo di ottimizzazione dell'interazione tra il tipo di elemento di memoria utilizzato e la codifica scelta
- Spesso, scelto il codice, si preferisce non ricorrere ad alcuna specifica strategia di codifica
 - ▶ Il costo rispetto alla affidabilità del risultato ottenuto è ritenuto eccessivo



■ Binario Naturale

- ▶ Il numero di bit è il minimo indispensabile
- ▶ Al primo stato corrisponde la configurazione di bit associata a 0, al secondo stato corrisponde la configurazione di bit associata ad 1...
- ▶ L'ordinamento degli stati è quello determinato in fase di realizzazione della tabella degli stati

■ One-Hot

- ▶ Il numero di bit necessari è pari al numero degli stati
- ▶ In ogni codifica, un solo bit assume valore 1, mentre tutti i bit rimanenti assumono valore 0
 - Le codifiche degli stati sono tutte a distanza di Hamming 2

Sintesi: Codifica degli stati (passo 4)



■ Esempio

- ▶ Codifica dei 4 stati s_0 , s_1 , s_2 ed s_3

Stato	Codifica Binaria naturale	Codifica One-Hot
s_0	00	0001
s_1	01	0010
s_2	10	0100
s_3	11	1000

Sintesi: Codifica degli stati - Esempio



Tabella degli stati

	00	01	11	10	U
s_0	s_0	s_0	s_1	s_1	0
s_1	s_1	s_0	s_0	s_1	1

Tabella delle transizioni: Codifica naturale

$s_0=0, s_1=1$

	00	01	11	10	U
0	0	0	1	1	0
1	1	0	0	1	1

$s_0=1, s_1=0$

	00	01	11	10	U
1	1	1	0	0	0
0	0	1	1	0	1

Tabella delle transizioni: Codifica naturale

$s_0=01, s_1=10$

	00	01	11	10	U
01	01	01	10	10	0
10	10	01	01	10	1

$s_0=10, s_1=01$

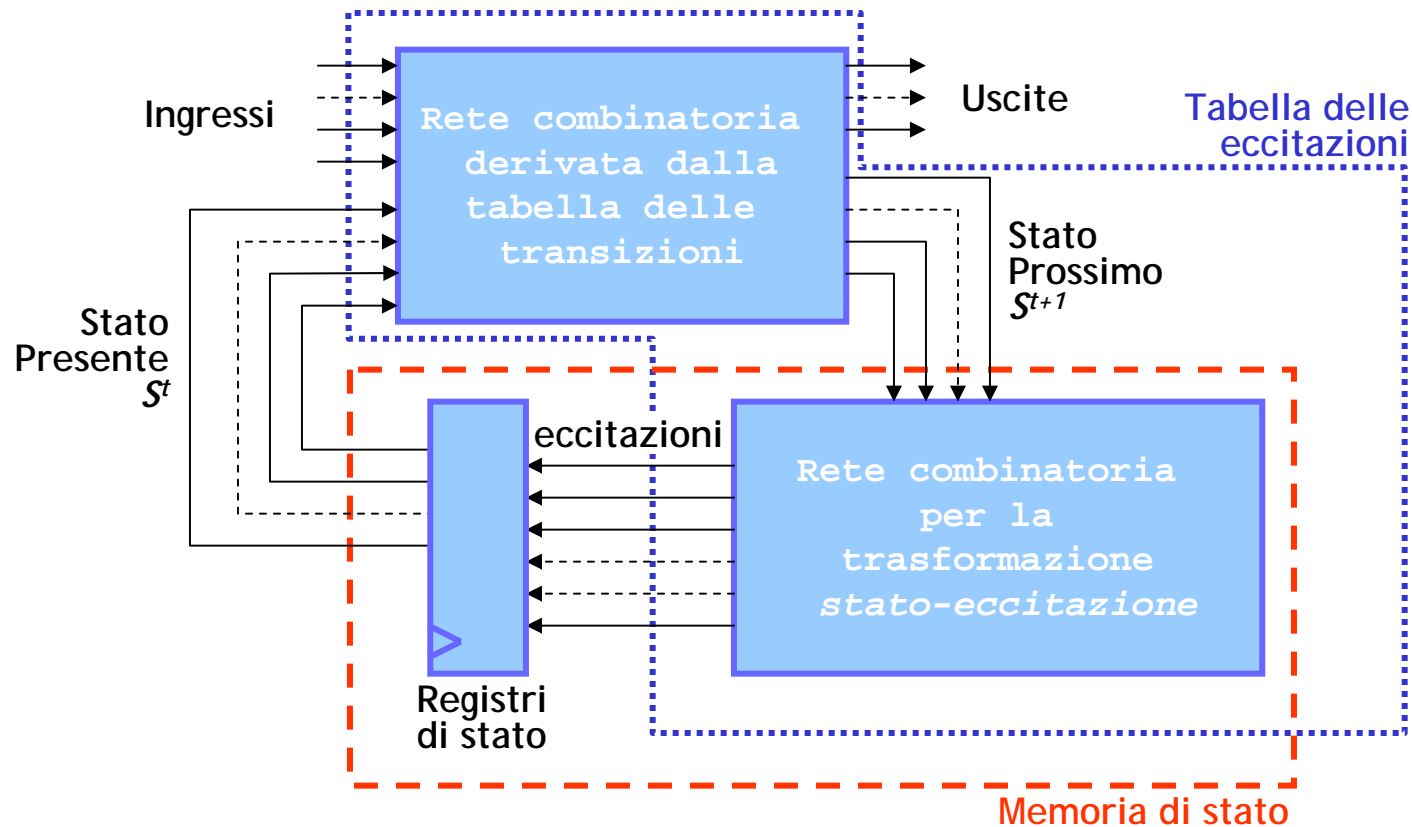
	00	01	11	10	U
10	10	10	01	01	0
01	01	10	10	01	1

Sintesi: Scelta del Bistabile (passo 5)



- La **tabella delle transizioni** descrive la relazione tra i bit di stato presente e quelli di stato futuro
 - ▶ La configurazione in bit dello stato presente è in diretta corrispondenza con le uscite degli elementi di memoria
 - ▶ La configurazione in bit dello stato futuro precisa quello che si desidera ottenere
- Cambiando il tipo di bistabile variano i segnali che devono essere generati per realizzare la transizione stato presente - stato prossimo desiderata
 - ▶ I segnali di ingresso di un bistabile sono detti eccitazioni
- La **Tabella delle Eccitazioni** di un bistabile rappresenta il mezzo di collegamento tra la tabella delle transizioni e la tabella delle eccitazioni

Sintesi: Scelta del Bistabile (passo 5)



Richiamo: Transizioni ed Eccitazioni



Tabelle delle Transizioni

C	S	R	Q*
0	-	-	Q
1	0	0	Q
1	0	1	0
1	1	0	1
1	1	1	-

C	J	K	Q*
0	-	-	Q
1	0	0	Q
1	0	1	0
1	1	0	1
1	1	1	Q'

C	D	Q*
0	-	Q
1	0	0
1	1	1

C	T	Q*
0	-	Q
1	0	Q
1	1	Q'

Tabelle delle Eccitazioni

Q	Q*	C	S	R
0	0	0	-	-
1	1	0	-	-
0	0	1	0	-
0	1	1	1	0
1	0	1	0	1
1	1	1	-	0

Q	Q*	C	J	K
0	0	0	-	-
1	1	0	-	-
0	0	1	0	-
0	1	1	1	-
1	0	1	-	1
1	1	1	-	0

Q	Q*	C	D
0	0	0	-
1	1	0	-
0	0	1	0
0	1	1	1
1	0	1	0
1	1	1	1

Q	Q*	C	T
0	0	0	-
1	1	0	-
0	0	1	0
0	1	1	1
1	0	1	1
1	1	1	0

Sintesi: Scelta del Bistabile: **SR**



Tabella delle transizioni

Codifica naturale: $s_0=0$, $s_1=1$

	00	01	11	10	U
0	0	0	1	1	0
1	1	0	0	1	1

Tabella delle Eccitazioni

Bistabile utilizzato: **SR**

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

Tabella delle Eccitazioni
del bistabile **SR**

Q	Q*	C	S	R
0	0	0	-	-
1	1	0	-	-
0	0	1	0	-
0	1	1	1	0
1	0	1	0	1
1	1	1	-	0

Sintesi: Funzioni di stato prossimo



Tabelle delle Eccitazioni

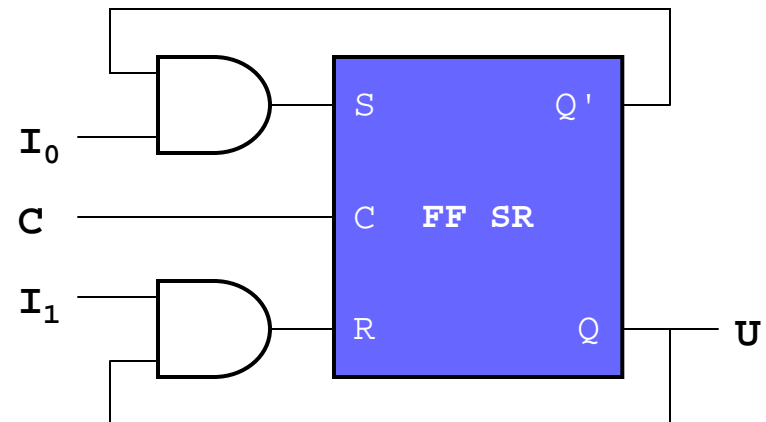
		$I_0 I_1$				
		00	01	11	10	U
Q	0	0-	0-	10	10	0
	1	-0	01	01	-0	1

Mappe di Karnaugh

		$I_0 I_1$				
		00	01	11	10	
Q	0	0	0	1	1	$S = Q' I_0$
	1	-	0	0	-	

		$I_0 I_1$				
		00	01	11	10	
Q	0	-	-	0	0	$R = Q I_1$
	1	0	1	1	0	

Rete logica



Sintesi: Scelta del Bistabile: JK



Tabella delle transizioni

Codifica naturale: $s_0=0$, $s_1=1$

	00	01	11	10	U
0	0	0	1	1	0
1	1	0	0	1	1

Tabella delle Eccitazioni

Bistabile utilizzato: JK

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

Tabella delle Eccitazioni
del bistabile JK

Q	Q*	C	J	K
0	0	0	-	-
1	1	0	-	-
0	0	1	0	-
0	1	1	1	-
1	0	1	-	1
1	1	1	-	0

Sintesi: Funzioni di stato prossimo



Tabelle delle Eccitazioni

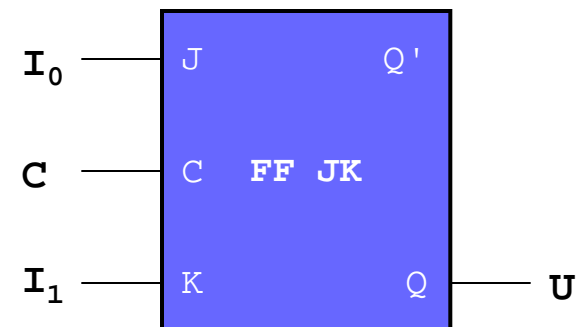
		$I_0 I_1$				
		00	01	11	10	U
Q	0	0-	0-	1-	1-	0
	1	-0	-1	-1	-0	1

Mappe di Karnaugh

		$I_0 I_1$				
		00	01	11	10	
Q	0	0	0	1	1	$J = I_0$
	1	-	-	-	-	

		$I_0 I_1$				
		00	01	11	10	
Q	0	-	-	-	-	$K = I_1$
	1	0	1	1	0	

Rete logica



Sintesi: Scelta del Bistabile: D



Tabella delle transizioni

Codifica naturale: $s_0=0$, $s_1=1$

	00	01	11	10	U
0	0	0	1	1	0
1	1	0	0	1	1

Tabella delle Eccitazioni

Bistabile utilizzato: D

	00	01	11	10	U
0	0	0	1	1	0
1	1	0	0	1	1

Tabella delle Eccitazioni
del bistabile D

Q	Q*	C	D
0	0	0	-
1	1	0	-
0	0	1	0
0	1	1	1
1	0	1	0
1	1	1	1

Sintesi: Funzioni di stato prossimo



Tabelle delle Eccitazioni

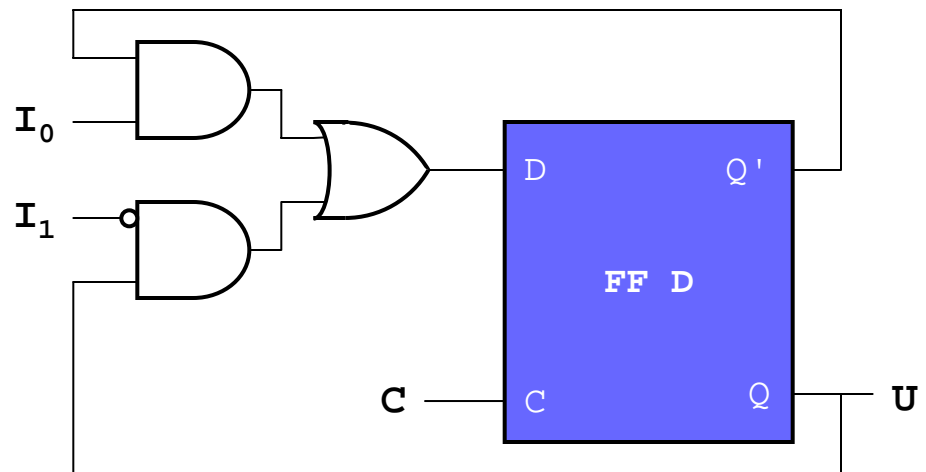
		$I_0 I_1$				U
		00	01	11	10	
Q	0	0	0	1	1	0
	1	1	0	0	1	1

Mappe di Karnaugh

		$I_0 I_1$			
		00	01	11	10
Q	0	0	0	1	1
	1	1	0	0	1

$$D = Q' I_0 + Q I_1'$$

Rete logica



Sintesi: Scelta del Bistabile: **T**



Tabella delle transizioni

Codifica naturale: $s_0=0$, $s_1=1$

	00	01	11	10	U
0	0	0	1	1	0
1	1	0	0	1	1

Tabella delle Eccitazioni

Bistabile utilizzato: **T**

	00	01	11	10	U
0	0	0	1	1	0
1	0	1	1	0	1

Tabella delle Eccitazioni
del bistabile **T**

Q	Q*	C	T
0	0	0	-
1	1	0	-
0	0	1	0
0	1	1	1
1	0	1	1
1	1	1	0

Sintesi: Funzioni di stato prossimo



Tabelle delle Eccitazioni

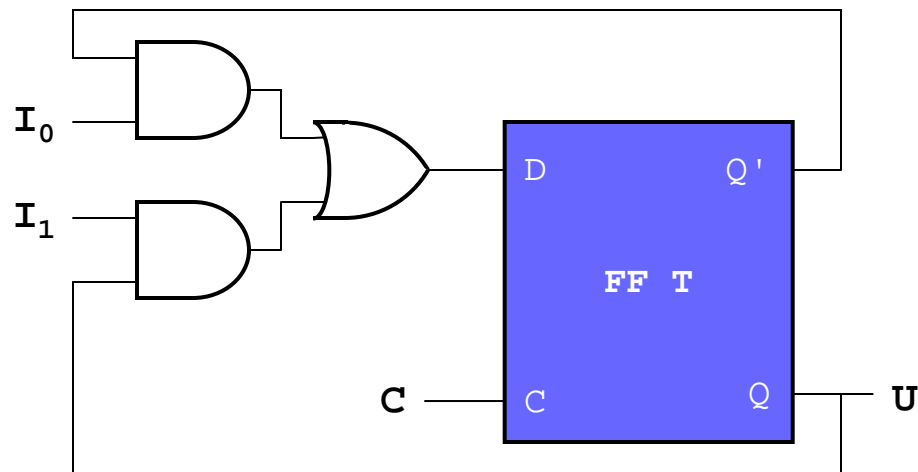
		$I_0 I_1$				
		00	01	11	10	U
Q	0	0	0	1	1	0
	1	0	1	1	0	1

Mappe di Karnaugh

		$I_0 I_1$			
		00	01	11	10
Q	0	0	0	1	1
	1	0	1	1	0

$$D = Q' I_0 + Q I_1$$

Rete logica



Sintesi: Scelta del Bistabile: JK



Tabella delle transizioni

Codifica One-Hot: $s_0=01$, $s_1=10$

	00	01	11	10	U
01	01	01	10	10	0
10	10	01	01	10	1

Tabella delle Eccitazioni

Bistabile utilizzato: JK

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

Tabella delle Eccitazioni
del bistabile JK

Q	Q*	C	J	K
0	0	0	-	-
1	1	0	-	-
0	0	1	0	-
0	1	1	1	-
1	0	1	-	1
1	1	1	-	0

Sintesi: Funzioni di stato prossimo



Tabelle delle Eccitazioni

		$I_0 I_1$				U
		00	01	11	10	
$Q_0 Q_1$	01	0- -0	0- -0	1- -1	1- -1	0
	10	-0 0-	-1 1-	-1 1-	-0 0-	1

Mappe di Karnaugh

		$I_0 I_1$				
		00	01	11	10	
Q_0	0	0	0	1	1	$J_0 = I_0$
	1	-	-	-	-	

		$I_0 I_1$				
		00	01	11	10	
Q_1	0	-	-	-	-	$J_1 = I_1$
	1	0	1	1	0	

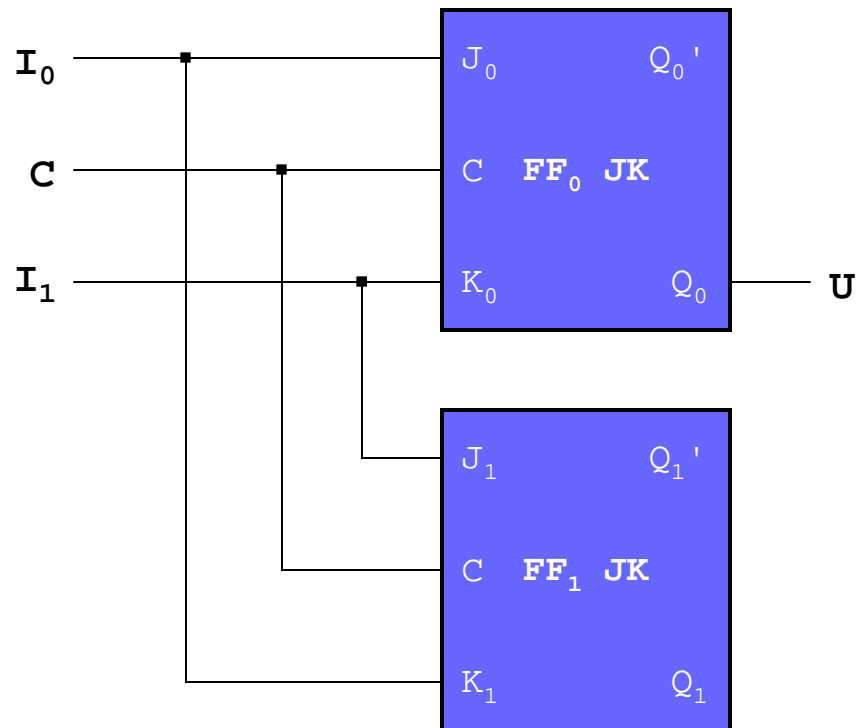
		$I_0 I_1$				
		00	01	11	10	
Q_0	0	-	-	-	-	$K_0 = I_1$
	1	0	1	1	0	

		$I_0 I_1$				
		00	01	11	10	
Q_1	0	0	0	1	1	$K_1 = I_0$
	1	-	-	-	-	

Sintesi: Funzioni di stato prossimo



Rete logica





- Facendo riferimento al caso a codice a minor numero di bit, l'esempio svolto è relativo alla trasformazione di un elemento di memoria in un altro
 - ▶ Ottenere un **JK** utilizzando **SR**, **JK**, **D** o **T**
- Le trasformazioni sono molto utili quando è richiesto uno specifico elemento di memoria ma non è disponibile
- Un modo differente per raggiungere lo stesso obiettivo è quello di utilizzare le equazioni caratteristiche

$$\text{SR:} \quad Q^* = C'Q + C(S + R'Q)$$

$$\text{JK:} \quad Q^* = C'Q + C(JQ' + K'Q)$$

$$\text{D:} \quad Q^* = C'Q + CD$$

$$\text{T:} \quad Q^* = C'Q + C(TQ' + T'Q)$$



■ Realizzazione di un flip-flop T mediante un flip-flop SR

▶ Flip-flop SR: $Q^* = C'Q + C(S + R'Q)$

▶ Flip-flop T: $Q^* = C'Q + C(TQ' + T'Q)$

■ Deve essere: $S + R'Q = TQ' + T'Q$

▶ Se $Q = 0$

$$\bullet S + R'(0) = T(1) + T'(0) \rightarrow S = T, R = -$$

▶ Se $Q = 1$ e $T = 0$

$$\bullet S + R'(1) = (0)(0) + (1)(1) \rightarrow S + R' = 1 \rightarrow S = 1, R = -$$

▶ Se $Q = 1$ e $T = 1$

$$\bullet S + R'(1) = (1)(0) + (0)(1) \rightarrow S + R' = 0 \rightarrow S = 0, R = 1$$

■ Quindi:

$$\bullet S = Q'(T) + QT'(1) + QT(0) = Q'T + QT'$$

$$\bullet R = QT(1) = QT$$



- Realizzazione di un flip-flop **SR** mediante un flip-flop **T**
 - ▶ Flip-flop **T**: $Q^* = C'Q + C(TQ' + T'Q)$
 - ▶ Flip-flop **SR**: $Q^* = C'Q + C(S + R'Q)$
- Deve essere: $TQ' + T'Q = S + R'Q$
 - ▶ Se $Q = 0$
 - $T(1) + T'(0) = S + R'(0) \rightarrow T = S$
 - ▶ Se $Q = 1$
 - $T(0) + T'(1) = S + R'(1) \rightarrow T' = S + R' \rightarrow T = S'R$
- Quindi:
 - ▶ $T = Q'(S) + Q(S'R) = Q'S + QS'R$



■ Sintesi mediante flip-flop di tipo SR

Tabella degli Stati

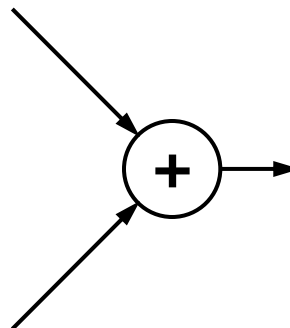
	00	01	11	10	Z
S0	S0	S0	S2	S1	1
S1	S1	S1	S0	S1	0
S2	S2	S3	S0	S2	1
S3	S3	S3	S2	S3	0

Codifica

S0	00
S1	01
S2	11
S3	10

Tabella delle Transizioni

	00	01	11	10	Z
00	00	00	11	01	1
01	01	01	00	01	0
11	11	10	00	11	1
10	10	10	11	10	0



Sintesi: Esempio



Tabella delle Transizioni

	00	01	11	10
00	00	00	11	01
01	01	01	00	01
11	11	10	00	11
10	10	10	11	10

Tabella delle Eccitazioni del bistabile di tipo SR

Q	Q'	SR
0	0	0-
0	1	10
1	0	01
1	1	-0

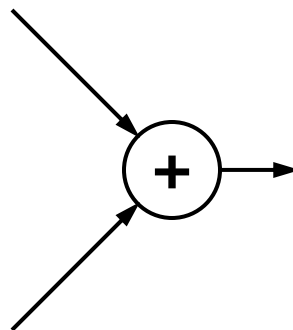


Tabella delle Eccitazioni

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

Sintesi: Esempio



Mappe di Karnaugh

$I_1 I_0$					
		00	01	11	10
$Q_1 Q_0$	00	0	0	1	1
	01	-	-	0	-
	11	-	0	0	-
	10	0	0	1	0

$$S_0 = Q_1' Q_0' I_1 + I_1 I_0 Q_0'$$

$$= Q_1' Q_0' I_1 + S_1$$

$I_1 I_0$					
		00	01	11	10
$Q_1 Q_0$	00	-	-	0	0
	01	0	1	-	0
	11	0	1	1	0
	10	-	-	0	-

$$R_0 = Q_0 I_0$$

$I_1 I_0$					
		00	01	11	10
$Q_1 Q_0$	00	0	0	1	0
	01	0	0	0	0
	11	-	-	0	-
	10	-	-	-	-

$$S_1 = I_1 I_0 Q_0'$$

$I_1 I_0$					
		00	01	11	10
$Q_1 Q_0$	00	-	-	0	-
	01	-	-	-	-
	11	0	0	1	0
	10	0	0	0	0

$$R_1 = I_1 I_0 Q_0$$

$$= I_1 R_0$$