



# Macchine a Stati finiti

Prof. Alberto Borghese  
Dipartimento di Scienze dell'Informazione  
[borgnese@dsi.unimi.it](mailto:borgnese@dsi.unimi.it)

Università degli Studi di Milano

Riferimento al Patterson: Sezione C.10

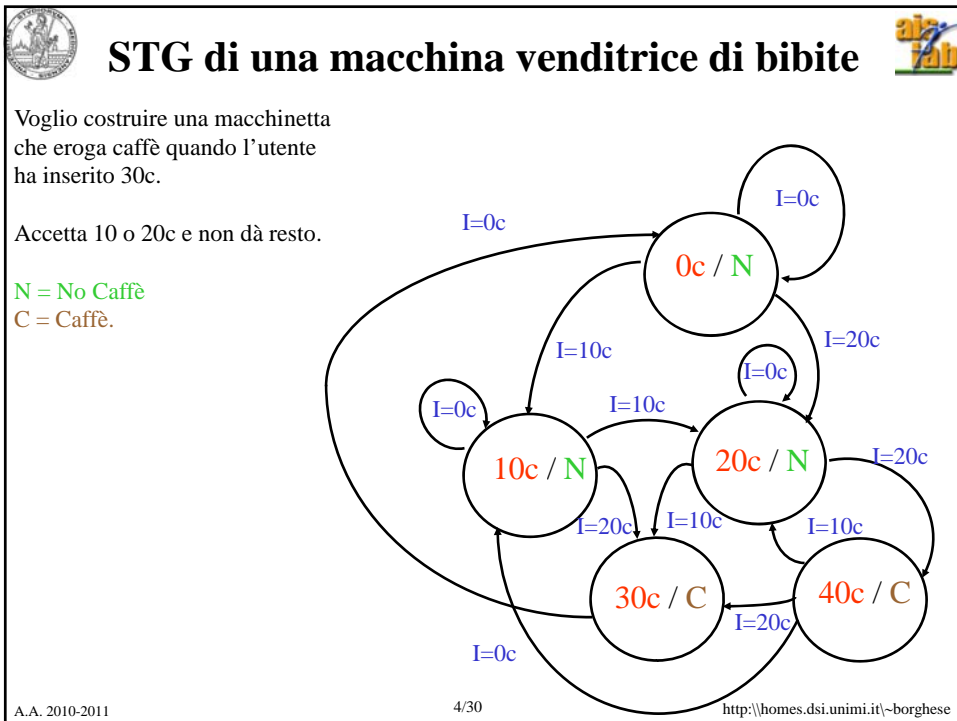
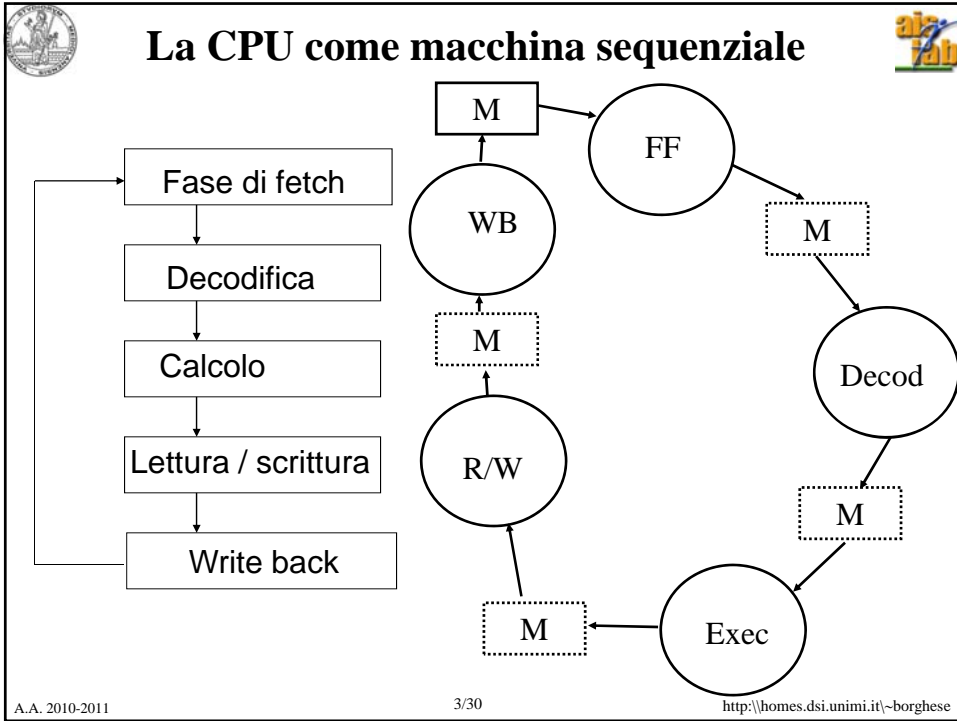


## Sommario

### Macchine a stati finiti

Esempio: sintesi di un controllore per venditore di bibite.

Esempio: sintesi di un controllore di un semaforo.





## Macchina a Stati Finiti (di Moore)



La Macchina di Moore è definita, in teoria degli automi, dalla sestupla:

$$\langle X, I, Y, f(\cdot), g(\cdot), X_{ini} \rangle$$

**X**: insieme degli stati (in numero finito).

**I**: insieme di ingresso: tutti i simboli che si possono presentare in ingresso.

**Y**: insieme di uscita: tutti i simboli che si possono generare in uscita.

**f(.)**: funzione stato prossimo:  $X' = f(X, I)$ . Definisce l'evoluzione della macchina nel tempo. L'evoluzione è deterministica.

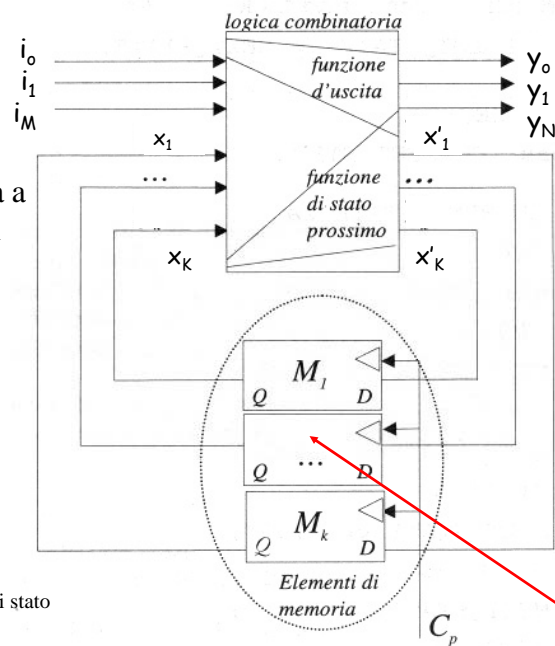
**g(.)**: funzione di uscita:  $Y = g(X)$  nelle macchine di Moore.

**Stato iniziale**:  $X_{ini}$ . Per il buon funzionamento della macchina è previsto uno stato iniziale, al quale la macchina può essere portata mediante un comando di reset.



Macchina a stati finiti binaria

M ingressi  
K variabili di stato  
N uscite



Macchina di Huffman

Stato



## Descrizione di una macchina di Moore



**STG: State Transition Graph** (Diagramma degli stati o Grafo delle transizioni). Ad ogni nodo è associato uno stato. Un arco orientato da uno stato  $x_i$  ad uno stato  $x_j$ , contrassegnato da un simbolo (di ingresso)  $\alpha$ , rappresenta una transizione (passaggio di stato) che si verifica quando la macchina, essendo nello stato  $x_i$ , riceve come ingresso il simbolo  $\alpha$ .

**STT: State Transition Table** (Tabella degli Stati). Per ogni coppia, (Stato presente – Ingresso), si definisce l'Uscita e lo Stato Prossimo. La forma è tabellare e ricorda le tabelle della verità da cui è derivata.



## Sommario



Macchine a stati finiti

Esempio: sintesi di un controllore per venditore di bibite.

Esempio: sintesi di un controllore di un semaforo.



## STG di una macchina venditrice di bibite (Semplificata)



Voglio costruire una macchinetta che eroga caffè quando l'utente ha inserito 30c.  
Accetta solamente monete da 10c.

$I = \{0c, 10c\}$

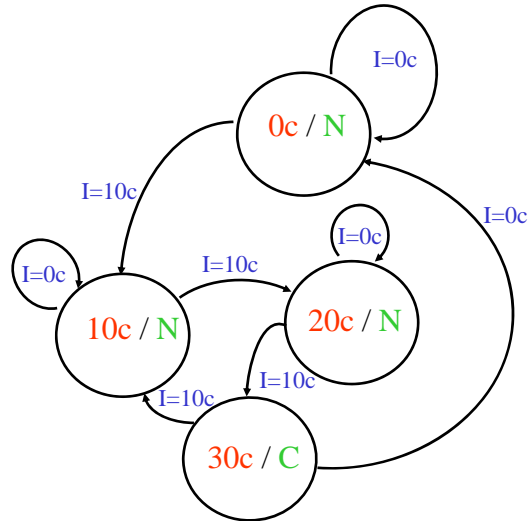
$Y = \{\underline{N}ulla, \underline{C}affè\}$

$X = \text{"Monete accumulate"}$   
 $= \{0, 10c, 20c, 30c\}$

$X' = f(X, I)$

$Y = g(X)$

$X_0 = 0c$

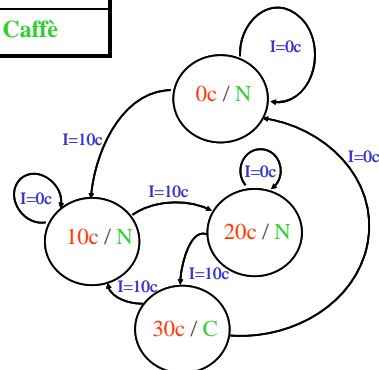


## STT della vendor machine



X \ I	I		Y
	0c	10c	
0c	0c	10c	Nulla
10c	10c	20c	Nulla
20c	20c	30c	Nulla
30c	0c	10c	Caffè

Il controllore controlla ogni 0.5s l'ingresso e ogni 0.5s aggiorna lo stato e l'uscita.





## Dal linguistico al numerico

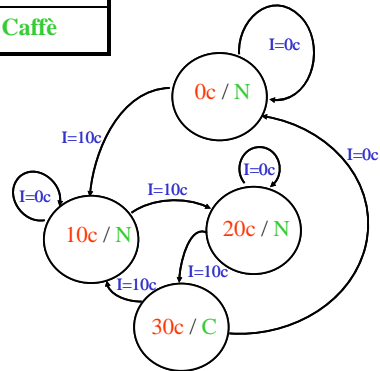


X \ I	I		Y
	0c	10c	
0c	0c	10c	Nulla
10c	10c	20c	Nulla
20c	20c	30c	Nulla
30c	0c	10c	Caffè

Occorre codificare ingresso, stato ed uscita.

=

Assegnare un codice numerico ad ogni possibile stato, ingresso, uscita.



## Codifica della STT della vendor machine



X \ I	I		Y
	0c (0)	10c (1)	
0c (00)	0c (00)	10c (01)	Nulla (0)
10c (01)	10c (01)	20c (10)	Nulla (0)
20c (10)	20c (10)	30c (11)	Nulla (0)
30c (11)	0c (00)	10c (01)	Caffè (1)

$I = [0c, 10c] =$

$\{0, 1\}$

$Y = [\text{Nulla}, \text{Caffè}] =$

$\{0, 1\}$

$X = [0, 10c, 20c, 30c] =$

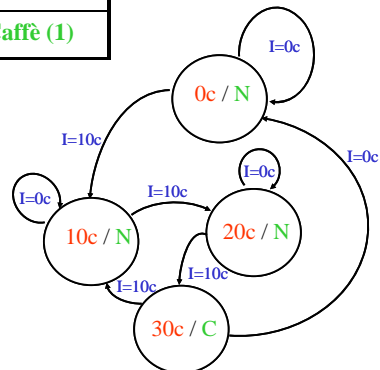
$\{00, 01, 10, 11\}$

$X' = f(X, I)$

da sintetizzare

$Y = g(X)$

da sintetizzare





## Macchina a Stati Finiti (di Moore)



La Macchina di Moore è definita, in teoria degli automi, dalla sestupla:

$$\langle X, I, Y, f(\cdot), g(\cdot), X_{ini} \rangle$$

**X**: insieme degli stati (in numero finito).

**I**: insieme di ingresso: tutti i simboli che si possono presentare in ingresso. In caso di codifica binaria, se abbiamo  $n$  linee in ingresso (variabili binarie), avremo  $2^n$  possibili simboli da leggere in ingresso (configurazioni).

**Y**: insieme di uscita: tutti i simboli che si possono generare in uscita. In caso di codifica binaria, se abbiamo  $m$  linee in uscita (variabili binarie), avremo  $2^m$  possibili simboli in uscita (configurazioni).

**f(.)**: funzione stato prossimo:  $X' = f(X, I)$ . Definisce l'evoluzione della macchina nel tempo. L'evoluzione è deterministica. La funzione è una funzione logica.

**g(.)**: funzione di uscita:  $Y = g(X)$  nelle macchine di Moore. E' una funzione logica.

**Stato iniziale:  $X_{ini}$** . Per il buon funzionamento della macchina è previsto uno stato iniziale, al quale la macchina può essere portata mediante un comando di reset.



## Sintesi della funzione di uscita della FSM della Vendor Machine



X \ I	I		Y
	0c (0)	10c (1)	
0c (00)	0c (00)	10c (01)	Nulla (0)
10c (01)	10c (01)	20c (10)	Nulla (0)
20c (10)	20c (10)	30c (11)	Nulla (0)
30c (11)	0c (00)	10c (01)	Caffè (1)

X è su 2 cifre =>  
2 bit  $X_1$  e  $X_0$

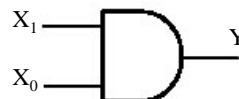
$$I = \{0, 1\}$$

$$Y = \{0, 1\}$$

$$X = \{00, 01, 10, 11\}$$

$$Y = g(X) = g(x_1, x_0)$$

$$X' = F(X, I) = ?$$



$x_1$	$x_0$	Y
0	0	0
0	1	0
1	0	0
1	1	1



## Sintesi della funzione stato prossimo della FSM della Vendor Machine



$x_1 x_0$	I		Y
	0	1	
0 0	00	01	0
0 1	01	10	0
1 0	10	11	0
1 1	00	01	1

X è su 2 cifre =>  
2 bit  $x_1$  e  $x_0$

$x_1 x_0 I$	$x'_1 x'_0$
0 0 0	0 0
0 0 1	0 1
0 1 0	0 1
0 1 1	1 0
1 0 0	1 0
1 0 1	1 1
1 1 0	0 0
1 1 1	0 1

$I = \{0, 1\}$

$Y = \{0, 1\}$

$X = \{00, 01, 10, 11\}$

$$X' = F(X, I) \Rightarrow x'_0 = I(x_1 + \bar{x}_0) + \bar{I}x_1\bar{x}_0 = I(x_1 + x_0) + \bar{I}x_1x_0$$

$$x'_1 = x_1\bar{x}_0 + \bar{x}_1x_0I$$

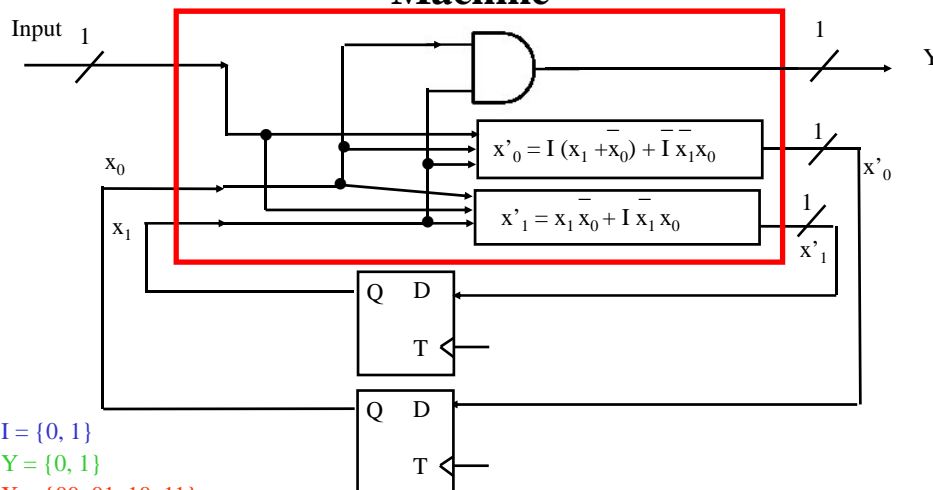
$$Y = g(X) = x_1 x_0$$

I 2 bit di stato prossimo vengono sintetizzati separatamente. Sono entrambi funzione dei 2 bit di stato all'istante attuale e del bit di ingresso

<http://homes.dsi.unimi.it/~borghese>



## Sintesi del circuito della FSM della Vendor Machine



$I = \{0, 1\}$

$Y = \{0, 1\}$

$X = \{00, 01, 10, 11\}$

$$X' = F(X, I) \Rightarrow x'_0 = I(x_1 + \bar{x}_0) + \bar{I}x_1x_0$$

$$x'_1 = x_1\bar{x}_0 + \bar{x}_1x_0I$$

$$Y = g(X) = x_1 x_0$$

A.A. 2010-2011

16/30

<http://homes.dsi.unimi.it/~borghese>





## Una vendor machine più completa.



Monete diverse dai 10c.

Scelta di bevande diverse.

Bevande diversi con costi diversi.

Periodo di refrattarietà nella quale non si possono inserire monete (periodo di preparazione del caffè).

.....



## Sommario



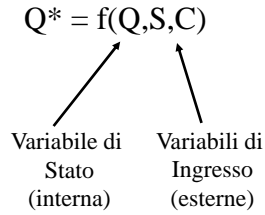
Macchine a stati finiti.

Esempio: sintesi di un controllore per venditore di bibite.

**Esempio: sintesi di un controllore di un semaforo.**



## Latch di tipo SC come macchina a stati finiti



Q	SC = 00	SC = 01	SC = 10	SC = 11
0	0	0	1	X
1	1	0	1	X

No change (Q\* = Q)    Clear Reset    Set

Q è l'uscita del latch: stato presente.

Q\* è il valore dell'uscita al tempo successivo: stato prossimo.

L'uscita del latch coincide con il suo stato (interno).

NB non ha senso implementare il latch come macchina di Huffman.



## Controllore di un semaforo



2 strade: nord-sud, NS, ed est-ovest, EO, che devono essere controllate da un semaforo. Il sistema di controllo dà via libera alternativamente alla direttrice NS o EO.

Il sistema di controllo pilota un semaforo il quale accenderà alternativamente il verde sulla direttrice NS o sulla direttrice EO (quando il semaforo non è verde, per semplicità supponiamo che sia rosso).

Il semaforo può commutare ogni 30 secondi (clock con frequenza = ?).

Supponiamo che esista una video-camera in grado di "leggere", per ogni direttrice, se esiste almeno un'auto in attesa, oppure un'auto che si accinga ad attraversare (le due condizioni sono trattate allo stesso modo).

Il semaforo deve cambiare colore (da rosso a verde quando esiste un'auto in attesa sulla sua direttrice).

Se ci sono due auto in attesa sulle due direttrici, viene posto a verde il semaforo rosso e a rosso il semaforo verde (viene dato via libera all'altra direttrice).

Supponiamo che all'accensione venga dato via libera alla direttrice NS.



## Stato, Input, Output del semaforo



**Ingresso:** {Nulla, Auto<sub>NS</sub>, Auto<sub>EO</sub>, Auto<sub>Both</sub>}

**Uscita:** {Luce<sub>NS</sub>, Luce<sub>EO</sub>}

**Stato:** ?

**f(X,I) = ?**

**G(X) = ?**



## STG del semaforo



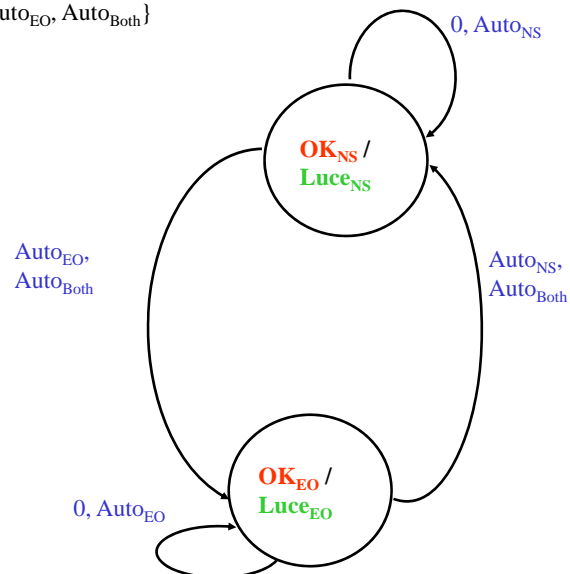
**Ingresso:** {Nulla, Auto<sub>NS</sub>, Auto<sub>EO</sub>, Auto<sub>Both</sub>}

**Uscita:** {Luce<sub>EO</sub>, Luce<sub>NS</sub>}

**Stato:** {OK<sub>NS</sub>, OK<sub>EO</sub>}

**f(X,I) = ?**

**G(X) = ?**



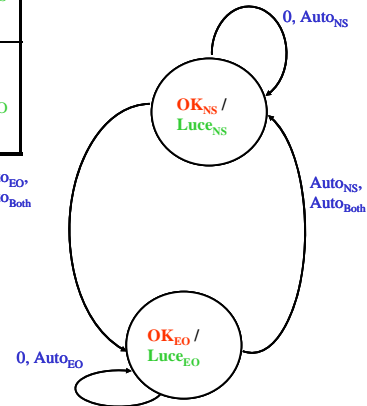


## STT del semaforo



I X	Input				Uscita
	Nulla	Auto <sub>NS</sub>	Auto <sub>EO</sub>	Auto <sub>Both</sub>	
OK <sub>NS</sub>	OK <sub>NS</sub>	OK <sub>NS</sub>	OK <sub>EO</sub>	OK <sub>EO</sub>	Luce <sub>NS</sub>
OK <sub>EO</sub>	OK <sub>EO</sub>	OK <sub>NS</sub>	OK <sub>EO</sub>	OK <sub>NS</sub>	Luce <sub>EO</sub>

Auto<sub>EO</sub>,  
Auto<sub>Both</sub>



## STT del semaforo binaria



I X	Input				Uscita
	Nulla = 00	Auto <sub>NS</sub> = 01	Auto <sub>EO</sub> = 10	Auto <sub>Both</sub> = 11	
OK <sub>NS</sub> = 0	OK <sub>NS</sub> = 0	OK <sub>NS</sub> = 0	OK <sub>EO</sub> = 1	OK <sub>EO</sub> = 1	Luce <sub>NS</sub> = 1
OK <sub>EO</sub> = 1	OK <sub>EO</sub> = 1	OK <sub>NS</sub> = 0	OK <sub>EO</sub> = 1	OK <sub>NS</sub> = 0	Luce <sub>EO</sub> = 0

**Ingresso:** {Nulla, Auto<sub>NS</sub>, Auto<sub>EO</sub>, Auto<sub>Both</sub>} = {00, 01, 10, 11}

**Uscita:** {Luce<sub>EO</sub>, Luce<sub>NS</sub>} = {0, 1}

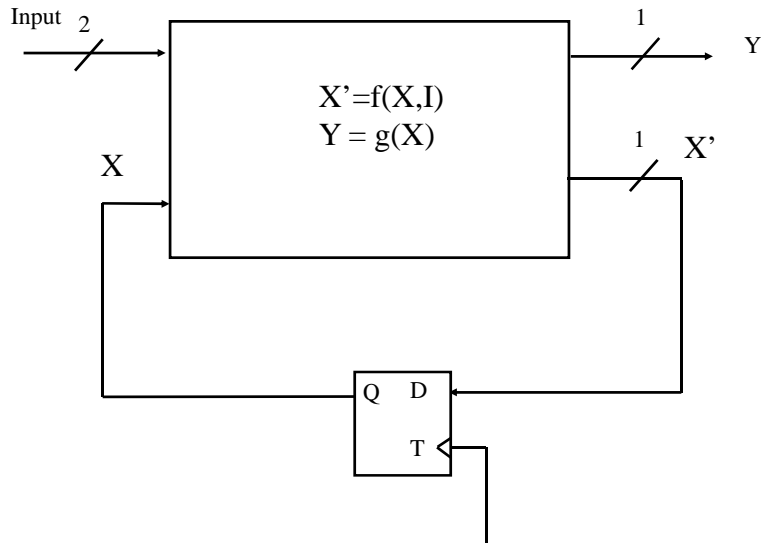
**Stato:** {OK<sub>NS</sub>, OK<sub>EO</sub>} = {0, 1}

**f(X,I)** = ?

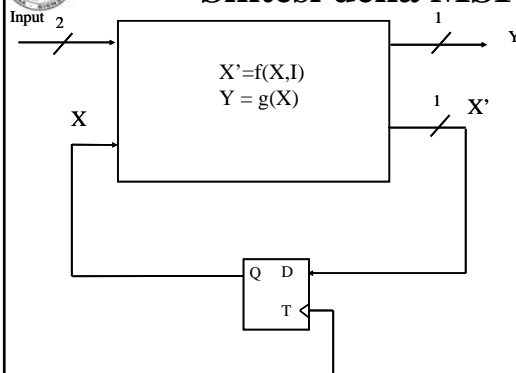
**G(X)** = ?



## Sintesi della MSF del semaforo



## Sintesi della MSF del semaforo



$$X' = \bar{X} i_1 \bar{i}_0 + \bar{X} i_1 i_0 + X \bar{i}_1 \bar{i}_0 + X i_1 \bar{i}_0 = \bar{X} i_1 + X i_0$$

$$Y = \bar{X}$$

I \ X	00	01	10	11	Uscita
0	0	0	1	1	1
1	1	0	1	0	0

**Ingresso:** {00, 01, 10, 11}

**Uscita:** {0, 1}

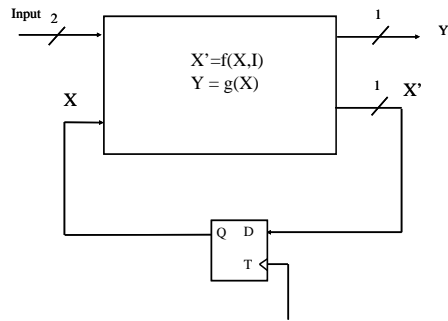
**Stato:** {0, 1}

**f(X, I) = ?**

**G(X) = ?**

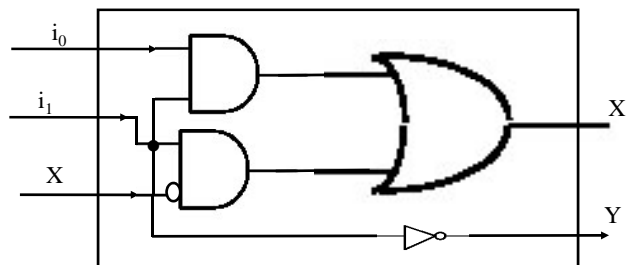


## Sintesi del circuito della MSF del semaforo



$$X' = \bar{X} i_1 \bar{i}_0 + \bar{X} i_1 i_0 + X \bar{i}_1 \bar{i}_0 + X i_1 \bar{i}_0 = \bar{X} i_1 + X \bar{i}_1$$

$$Y = \bar{X}$$



## I passi della progettazione di una MSF



Il committente fornisce le specifiche di funzionamento.

Definizione delle variabili di Input, Stato e Output. Definizione degli insiemi di simboli che possono essere assunti dalle variabili di Input e di Output.

Costruzione dello STG => Definizione dell'insieme di simboli che possono essere assunti dallo stato.

Costruzione della STT => Definizione implicita delle funzioni stato prossimo ed uscita.

Codifica della STT => Definizione del numero di bit per Input, Stato e Output.

STT Codificata => Circuiti combinatori che sintetizzano le funzioni  $f(X, I)$  e  $g(X)$ .



## Esercizi



• Costruire una macchina a stati finiti (di Moore), in grado di individuare all'interno di una parola di 0 e 1 le seguenti configurazioni: 1010 e 1110. Le configurazioni si possono concatenare (e.g. 101010 da' uscita vera, al secondo e terzo 0). Stato iniziale 00.

• Costruire una macchina a stati finiti (di Moore), con due ingressi,  $x_1$  e  $x_2$ , che fornisce 1 quando negli ultimi 3 istanti si è verificata la seguente configurazione:

	$t = -2$	$t = -1$	$t = 0$
$x_1$	0	X	1
$x_2$	x	1	0

Stato iniziale  $x_1 = 0$   $x_2 = 0$ .

• Costruire un venditore di bibite che distribuisce una bibita quando si raggiungono i 35 cents inseriti. Non dà resto.



## Sommario



Macchine a stati finiti.

Esempio: sintesi di un controllore per venditore di bibite.

Esempio: sintesi di un controllore di un semaforo.