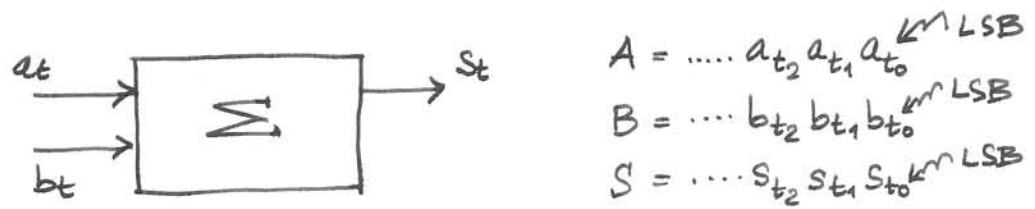


Ho introdotto le reti sequenziali con il SOMMATORE SERIALE :



Il suo comportamento NON è ottenibile con una funzione combinatoria :

| t     | $a_t$ | $b_t$ | $S_t$ |
|-------|-------|-------|-------|
| $t_0$ | 0     | 0     | 0     |
| $t_1$ | 0     | 1     | 1     |
| $t_2$ | 1     | 1     | 0     |
| $t_3$ | 0     | 1     | 0     |
| $t_4$ | ...   | ...   | ...   |

Annotations:

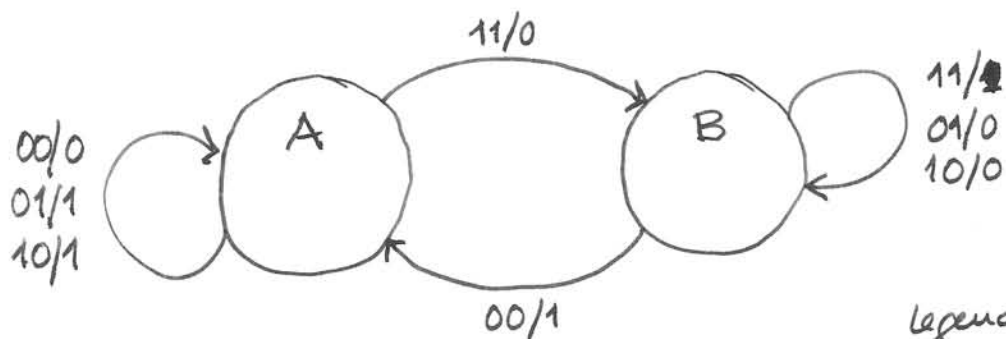
- Arrows pointing to the input pairs (0,1) at  $t_1$  and  $t_3$  with the text: "stesso input (01), uscita diversa"

Le reti combinatorie hanno un output indipendente dal tempo, sempre uguale dato un certo ingresso. E' come una persona che rimane sempre dello stesso umore. Le ROM hanno un contenuto immutabile: se leggiamo a un certo indirizzo ci troviamo sempre la stessa parola. In questo esempio invece la macchina ha un comportamento che dipende dal tempo. Si dice che in un certo istante essa si trova in un dato stato, che può cambiare nel tempo. E' come le persone reali, che cambiano umore nell'arco della giornata.

Nel nostro caso potremmo avere solo due diversi "umori", o meglio stati. La macchina si trova nello stato A se non c'è stato un evento al passo precedente, e B se invece c'è stato un evento:



(ASF)  
Una macchina sequenziale, o automa a stati finiti, può trovarsi in generale in uno di  $N$  stati. Da un istante all'altro può cambiare stato, sulla base degli ingressi esterni e dello stato in cui si trova. Nel ns caso potremmo costruire il DIAGRAMMA DEGLI STATI, che è un grafo che ha come nodi gli stati e come archi le transizioni di stato. Le transizioni da uno stato sono controllate dagli ingressi esterni:



legenda:  
 $a_t b_t / s_t$

Nota: il numero di archi uscenti da ciascuno stato è  $2^m$ , con  $m = \#$  ingressi esterni.

Il diagramma degli stati può essere trasformato in una tabella, detta DELLE TRANSIZIONI:

| $a_t$ | $b_t$ | SP    | SF    | $S_t$ |
|-------|-------|-------|-------|-------|
| 0     | 0     | $A^1$ | $A^1$ | 0     |
| 0     | 0     | $B^0$ | $A^1$ | 1     |
| 0     | 1     | $A^1$ | $A^1$ | 1     |
| 0     | 1     | $B^0$ | $B^0$ | 0     |
| 1     | 0     | $A^1$ | $A^1$ | 1     |
| 1     | 0     | $B^0$ | $B^0$ | 0     |
| 1     | 1     | $A^1$ | $B^0$ | 0     |
| 1     | 1     | $B^0$ | $B^0$ | 1     |

Per proseguire nel progetto, gli stati della macchina vanno codificati in binario. Qui il # di stati è 2, per cui basta 1 BIT DI STATO. Possiamo fare due possibili associazioni:

a)  $\begin{cases} A \rightarrow 0 \\ B \rightarrow 1 \end{cases}$

b)  $\begin{cases} A \rightarrow 1 \\ B \rightarrow 0 \end{cases}$

| $a_t$ | $b_t$ | $y$ | $y'$ | $S_t$ |
|-------|-------|-----|------|-------|
| 0     | 0     | 0   | 0    | 0     |
| 0     | 0     | 1   | 0    | 1     |
| 0     | 1     | 0   | 0    | 1     |
| 0     | 1     | 1   | 1    | 0     |
| 1     | 0     | 0   | 0    | 1     |
| 1     | 0     | 1   | 1    | 0     |
| 1     | 1     | 0   | 1    | 0     |
| 1     | 1     | 1   | 1    | 1     |

questo è il FULL ADDER:

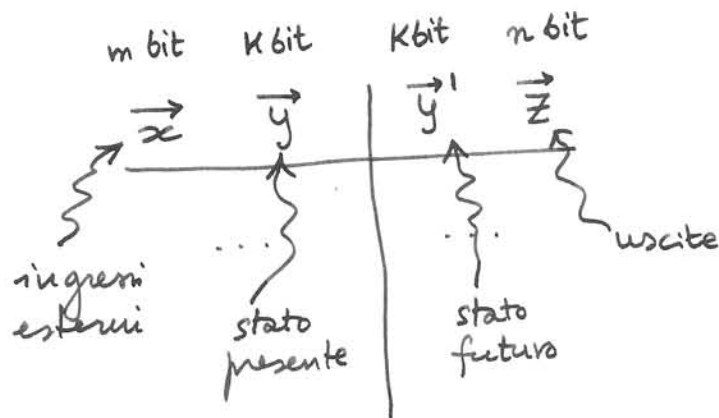
$$\begin{cases} y = c_{t-1} \\ y' = c_t \end{cases}$$

| $a_t$ | $b_t$ | $y$ | $y'$ | $S_t$ |
|-------|-------|-----|------|-------|
| 0     | 0     | 0   | 1    | 1     |
| 0     | 0     | 1   | 1    | 0     |
| 0     | 1     | 0   | 0    | 0     |
| 0     | 1     | 1   | 1    | 1     |
| 1     | 0     | 0   | 0    | 0     |
| 1     | 0     | 1   | 1    | 1     |
| 1     | 1     | 0   | 0    | 1     |
| 1     | 1     | 1   | 0    | 0     |

questo è diverso dal FULL ADDER:

$$\begin{cases} y = \bar{c}_{t-1} \\ y' = \bar{c}_t \end{cases}$$

Più in generale, la tabella delle transizioni di stato e uscita ha la seguente forma



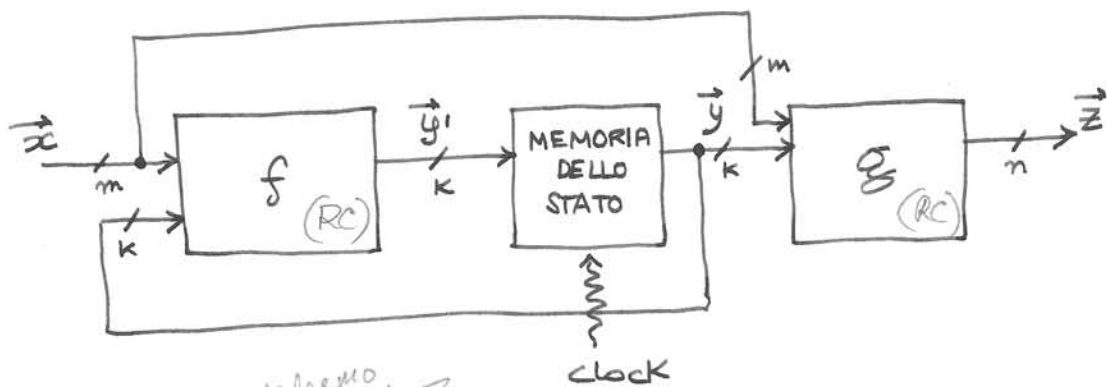
con

$$\begin{cases} \vec{y}' = f(\vec{x}, \vec{y}) \\ \vec{z} = g(\vec{x}, \vec{y}) \end{cases}$$

TRANSIZIONI DI STATO  
USCITA

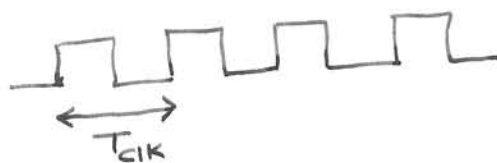
$\vec{z}$  equazioni caratteristiche della macchina

Una macchina sequenziale si può anche rappresentare con uno schema a blocchi orientato alla realizzazione:

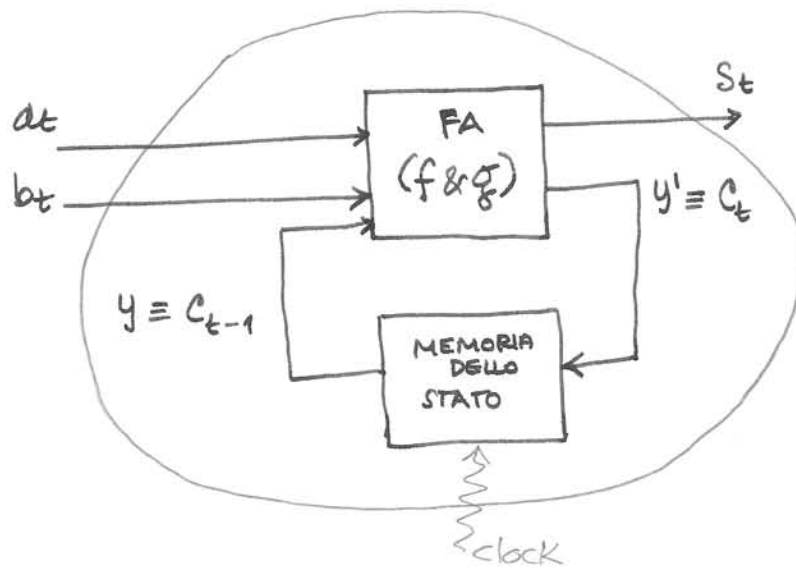


che vedremo presto!

Il blocco di MEMORIA conserva il valore dello stato per un quanto di tempo, trascorso il quale lo stato presente si trasforma in quello futuro, o meglio lo stato futuro (in INGRESSO ALLA MEMORIA) diventa il nuovo stato presente. Il blocco di memoria si comporta dunque come elemento ritardatore, tipicamente controllato da un orologio (clock) esterno. I quanti di tempo sono in tali casi periodi di clock:

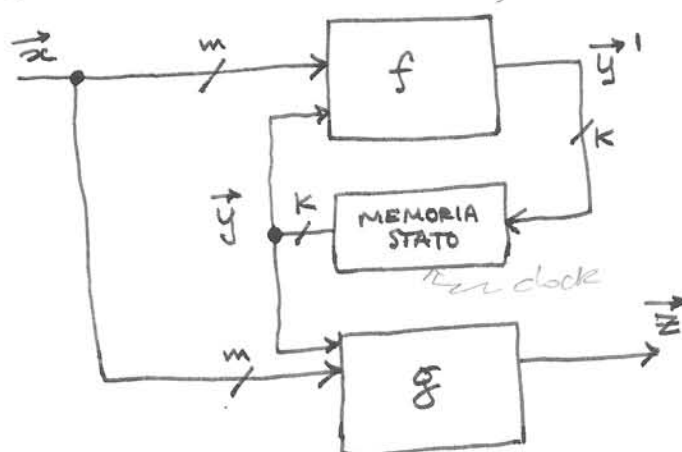


Nel caso del sommatore seriale, possiamo accorpare i blocchi  $f$  e  $g$  che, insieme, costituiscono un full adder (se la codifica è  $A \rightarrow 0$  e  $B \rightarrow 1$ ):



Notiamo che questo sommatore, a differenza di quello parallelo (e combinatorio), riesce a calcolare la somma di parole di lunghezza qualsiasi. Il tempo di attesa del risultato varia però con la dimensione delle parole di ingresso.

Una versione alternativa dello schema a blocchi generale è la seguente (vedi anche schema sopra, dove  $f$  e  $g$  sono accorpate)



Per riassumere, gli ASF possono essere descritte  
(in inglese FSM, "finite state machines")

in quattro modi diversi:

- ① diagramma degli stati
- ② tabelle di transizione
- ③ equazioni caratteristiche
- ④ schema a blocchi

per lez. 20/10/14

La descrizione appena vista è quella di MEALY,  
che è caratterizzata dal fatto che

L'USCITA DIPENDE DALL'INGRESSO ( $\vec{x}$ )  
(OLTRE CHE DALLO STATO PRESENTE ( $\vec{y}$ ))

Secondo la descrizione di MOORE, invece,

L'USCITA DIPENDE DAL SOLO STATO PRESENTE ( $\vec{y}$ )

È quest'ultima descrizione che useremo più spesso.

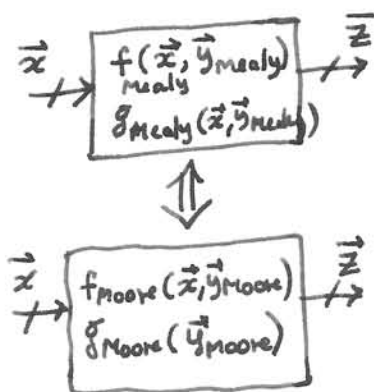
Le equazioni caratteristiche di una macchina di Moore  
sono quindi

$$\begin{cases} \vec{y}' = f(\vec{x}, \vec{y}) \\ \vec{z} = g(\vec{y}) \end{cases}$$

e la tabella delle transizioni si spezza in due:

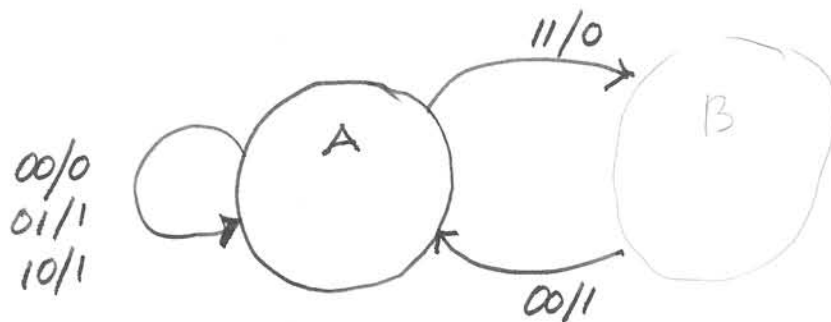
| $\vec{x}$ | $\vec{y}$ | $\vec{y}'$ | $\vec{y}$ | $\vec{z}$ |
|-----------|-----------|------------|-----------|-----------|
| ...       |           | ...        | ...       | ...       |

## NOTA BENE



Anche se abbiamo usato gli stessi simboli per lo stato ( $\vec{y}$ ) e le funzioni  $f$  e  $g$ , è chiaro che una stessa macchina realizzata secondo la descrizione di Mealy ha un numero di stati <sup>(diverso)</sup> (e funzioni <sup>(diverse)</sup> caratteristiche) dall'equivalente realizzazione secondo Moore. Addirittura, il # di bit di stato può cambiare. In particolare, gli stati di Moore sono in numero sempre superiori a quelli di Mealy. Vediamo perché...

Osserviamo ad es. lo stato (di Mealy) A per il ns. sommatore seriale:



Vediamo che possiamo arrivare ad A in quattro modi diversi, in tre dei quali l'uscita prodotta è 1, mentre in un caso l'uscita prodotta è 0. Nella rappresentazione secondo Moore ciò non sarebbe possibile, dovendosi avere che l'uscita è funzione dello stato. Quindi nel nuovo stato presente A, abbiamo tre casi in cui l'uscita è 1 e uno in cui l'uscita è 0, e non c'è alcuna funzione <sup>(che ad un ingresso costante)</sup> <sup>(quello stato)</sup> <sup>ancora</sup> combinatoria <sup>due valori diversi!</sup> <sup>Viqui</sup>

Potremmo vedere la cosa anche in quest'altro modo:  
 nello stato A, la macchina <sup>d' Mealy</sup> produce in uscita

0 se gli ingressi sono 00 • 11  
 1 se gli ingressi sono 01 • 10

(SF:A) (SF:B)  
 (SF:A) (SF:A)

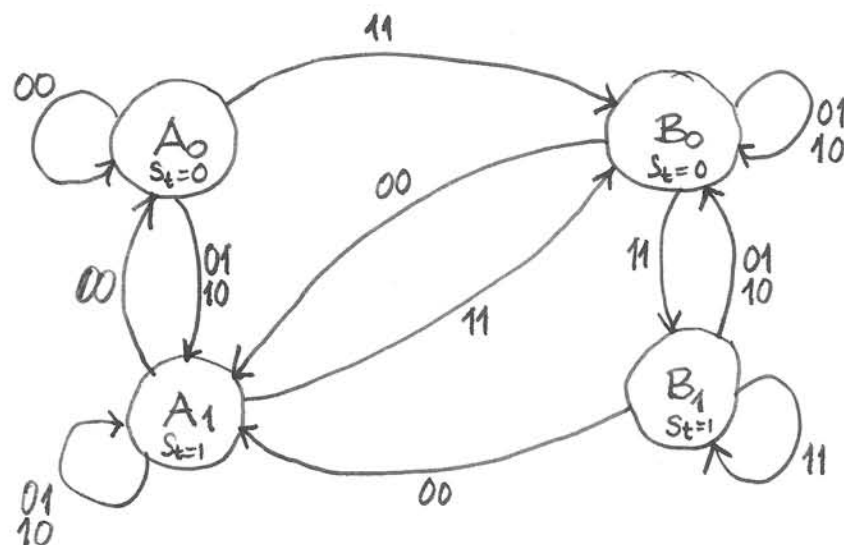
Invece, la macchina di Moore, può produrre solo un valore di uscita (0 oppure 1) se si trova in un dato stato. Quindi DOBBIAMO SDOPPIARE LO STATO A negli stati  $A_0$  e  $A_1$ , con le seguenti proprietà:

$A_0 \rightarrow$  no riporto, uscita prodotta 0  
 $A_1 \rightarrow$  no riporto, uscita prodotta 1

Analogamente:

$B_0 \rightarrow$  riporto, uscita prodotta 0  
 $B_1 \rightarrow$  " , " " 1

Vediamo l'automa trasformato Mealy  $\rightarrow$  Moore:



## Mealy

| $a_t$ | $b_t$ | SP | SF | $S_t$ |
|-------|-------|----|----|-------|
| 0     | 0     | A  | A  | 0     |
| 0     | 0     | B  | A  | 1     |
| 0     | 1     | A  | A  | 1     |
| 0     | 1     | B  | B  | 0     |
| 1     | 0     | A  | A  | 1     |
| 1     | 0     | B  | B  | 0     |
| 1     | 1     | A  | B  | 0     |
| 1     | 1     | B  | B  | 1     |

## Moore

stato di Mealy

| $a_t$ | $b_t$ | SP <sup>(t)</sup> | SF <sup>(t)</sup> |
|-------|-------|-------------------|-------------------|
| 0     | 0     | A <sub>0</sub>    | A <sub>0</sub>    |
| 0     | 0     | A <sub>1</sub>    | A <sub>0</sub>    |
| 0     | 1     | B <sub>0</sub>    | A <sub>1</sub>    |
| 0     | 1     | B <sub>1</sub>    | A <sub>1</sub>    |
| 1     | 0     | A <sub>0</sub>    | A <sub>1</sub>    |
| 1     | 0     | A <sub>1</sub>    | A <sub>1</sub>    |
| 1     | 1     | B <sub>0</sub>    | B <sub>0</sub>    |
| 1     | 1     | B <sub>1</sub>    | B <sub>0</sub>    |
| 1     | 0     | A <sub>0</sub>    | A <sub>1</sub>    |
| 1     | 0     | A <sub>1</sub>    | A <sub>1</sub>    |
| 1     | 1     | B <sub>0</sub>    | B <sub>0</sub>    |
| 1     | 1     | B <sub>1</sub>    | B <sub>0</sub>    |
| 1     | 1     | A <sub>0</sub>    | B <sub>0</sub>    |
| 1     | 1     | A <sub>1</sub>    | B <sub>0</sub>    |
| 1     | 1     | B <sub>0</sub>    | B <sub>1</sub>    |
| 1     | 1     | B <sub>1</sub>    | B <sub>1</sub>    |

| SP             | $S_t$ |
|----------------|-------|
| A <sub>0</sub> | 0     |
| A <sub>1</sub> | 1     |
| B <sub>0</sub> | 0     |
| B <sub>1</sub> | 1     |

Notiamo come nel caso Moore  
il # di stati sia raddoppiato, per il nostro esempio.  
Cio' richiede DUE BIT DI STATO, anzich' 1.

Con la codifica

|                     |
|---------------------|
| A <sub>0</sub> → 00 |
| A <sub>1</sub> → 01 |
| B <sub>0</sub> → 10 |
| B <sub>1</sub> → 11 |

↑↑  
y<sub>1</sub> y<sub>0</sub>

La cifra LSB codifica l'uscita,  
mentre il MSB codifica lo stato  
di Mealy. (y<sub>0</sub>)

Perche' per ogni stato di Mealy  
(cio' per ogni stato di Moore) c'e' un  
solo stato di Moore, la funz.  
di transizione di stato viene  
semplificata:

| $a_t$ | $b_t$ | y <sub>1</sub> | y <sub>0</sub> |
|-------|-------|----------------|----------------|
| 0     | 0     | 0              | 0              |
| 0     | 0     | 0              | 1              |
| 0     | 1     | 0              | 1              |
| 0     | 1     | 1              | 0              |
| 1     | 0     | 0              | 1              |
| 1     | 0     | 1              | 0              |
| 1     | 1     | 1              | 0              |
| 1     | 1     | 1              | 1              |

puento non e'  
che il FA!

| y <sub>1</sub> y <sub>0</sub> | $S_t$ |
|-------------------------------|-------|
| 00                            | 0     |
| 01                            | 1     |
| 10                            | 0     |
| 11                            | 1     |

⇒  $S_t = y_0$

La macchina in versione Moore è dunque:

