21 Aprile 2023

# ARCHITETTURE DI CALCOLO LEZIONE 11

## Esercizi su automi di Mealy e Moore

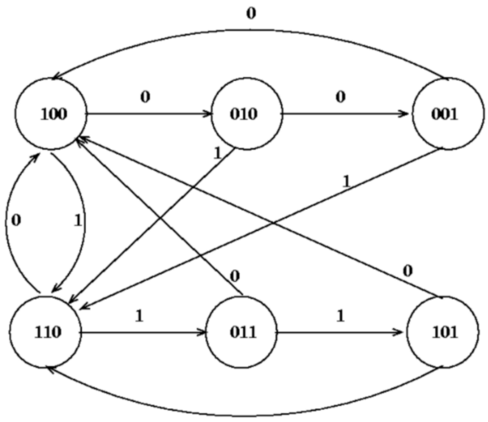
### Premessa

1. **L'esercizio 1 è spiegato nella sbobina precedente**
2. **Non ci saranno esercizi su automi di Mealy nello scritto ma l'argomento potrà ugualmente**
   **essere chiesto all'orale**

## <u>Esercizio 2</u>

- Progettare una rete sequenziale che comanda l'accensione e lo spegnimento di tre lampadine in sequenza
- L'output del circuito sono tre bit che per comodità chiamiamo: S,C,D. Quando questi sono affermati, le lampadine corrispondenti sono accese
- Il ritmo del circuito è determinato dal periodo di clock
- La rete riceve un segnale di ingresso I tale che: — se I=0:
  le lampadine devono accendersi in sequenza, una alla volta, partendo (la prima volta) da S.
  es. 100→010→0001→100→...
  — se I=1:
  le lampadine devono accendersi in sequenza, due alla volta, partendo (la prima volta) da S e C
  es. 110→011→101→110→...
- Determinare: Macchina a stati di Moore + Tabelle + Equazioni minime

### Macchina a stati di Moore




Si è stabilito che gli stadi coincidono con gli output.
La macchina ha 6 stadi, corrispondenti alle possibili combinazioni dell'output (meno le configurazioni 000 e 111 poiché non menzionate dalla traccia).

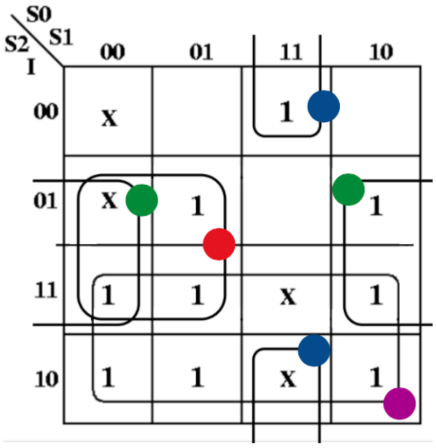
## Tabella di verità “NextState” e mappe di Karnaugh

| S0 | S1 | S2 | I | S0* | S1* | S2* |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | X | X | X |
| 0 | 0 | 1 | 0 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 1 | 1 | 0 | X | X | X |
| 0 | 0 | 0 | 1 | X | X | X |
| 0 | 0 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 1 | 1 | 1 | X | X | X |

S0, S1, e S2 rappresentano i 3 bit degli stati mentre I è l’input.
Le configurazioni 000 e 111 sono in rosso perché, come detto prima, non sono menzionate dalla traccia; vanno ugualmente riportate in quanto possono permettere di creare raggruppamenti più estesi nella mappa di Karnaugh.

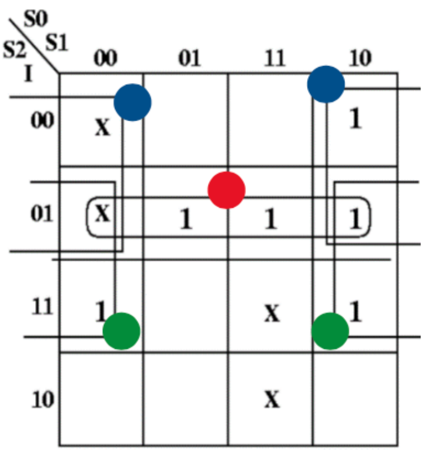
Di seguito l’espressione minima ottenuta dalla mappa di Karnaugh di S0*.

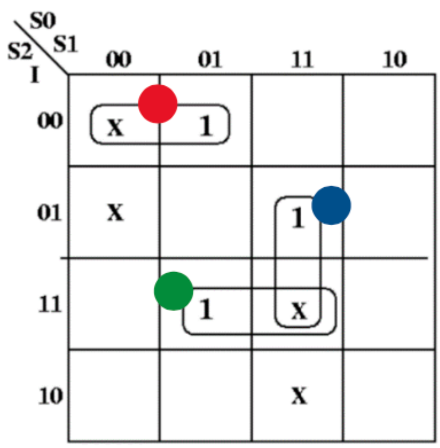
$$S0^* = (S0)(S1)(\underline{I}) + (S2) + (\underline{S0})(I) + (\underline{S1})(I)$$



$$S1^* = (\underline{S2})I + (\underline{S1})(\underline{S2}) + (\underline{S1})(I)$$

$$S2^* = (\underline{S0})(\underline{S2})(\underline{I}) + (S0)(S1)(I) + (S1)(S2)(I)$$





## Tabella di verità “Output”

| S0 | S1 | S2 | S | C | D |
|---|---|---|---|---|---|
| 0 | 0 | 0 | X | X | X |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 1 | 0 | 0 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 1 | 1 | 0 | 1 | 1 | 0 |
| 1 | 1 | 1 | X | X | X |

Come si può notare gli stati e l’output coincidono, poiché lo abbiamo imposto noi all’inizio dell’esercizio.Di questa tabella non vengono realizzate le tre mappe di Karnaugh in quanto si è preferito applicare delle semplici regole algebriche.
Esempio:
S = S0S1S2 + S0S1S2 + S0S1S2
S = S0 (S1S2 + S1S2 + S1S2) le variabili tra parentesi, qualsiasi valore assumano, daranno sempre 1 (ad eccezione del caso S1 = 1, S2 =1, non accettabile), per cui:
S0 x 1 = S0.

Ripetendo il processo, alla fine si ottiene:
S = S0
C = S1
D = S2

# Esercizio 3

Progettare una rete sequenziale di Moore, che:

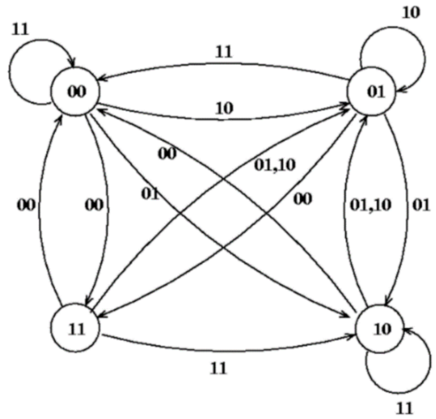
- riceva in ingresso due segnali $I_1$ e $I_2$

- restituisca in uscita due segnali $O_1$ e $O_2$, tali che:
  - se l'uscita precedente era ($O_1$ $O_2$) = (0 _)→$O_1O_2$ dovranno essere il complemento di $I_1I_2$ (es. stato:00; input:0, 1; output:10)
  - altrimenti→$O_1O_2$ saranno la somma di $I_1$ + $I_2$ (es. stato:10; input:1, 1; output:10)
- $O_1$=$O_2$=0

Si richiede di:

1. **Disegnare la macchina a stati finiti**
2. **Scrivere la tabella di verità**
3. **Trovare le forme SP minime**
4. **Disegnare il circuito**

## Macchina a stati finiti di Moore



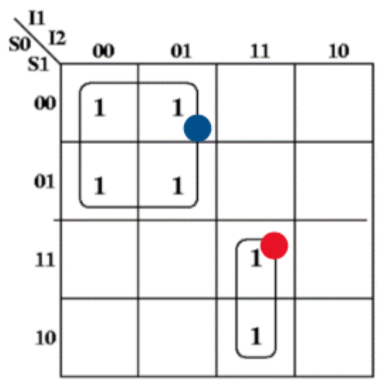

In questo caso output e stato coincidono.

## Tabella di verità "NextState" e mappe di Karnaugh

| S0 | S1 | I1 | I2 | S0* | S1* |
|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 0 |
| 0 | 0 | 1 | 0 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 |
| 0 | 1 | 0 | 0 | 1 | 1 |
| 0 | 1 | 0 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 0 | 1 |
| 0 | 1 | 1 | 1 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 0 | 0 |
| 1 | 1 | 0 | 1 | 0 | 1 |
| 1 | 1 | 1 | 0 | 0 | 1 |
| 1 | 1 | 1 | 1 | 1 | 0 |

Mappa di Karnaugh di $S_0^*$

$S_0^* = S_0 I_1 + I_1 I_2 S_0$




Mappa di Karnaugh di $S_1^*$

$S_1^* = S_0 I_2 + I_1 I_2 + S_0 I_1 I_2$

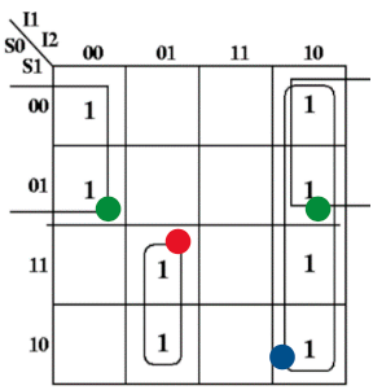
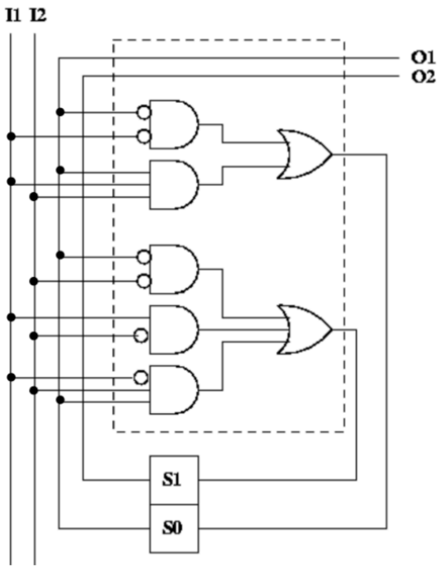


**Tabella di verità "Output"**

| S0 | S1 | O1 | O2 |
|---|---|---|---|
| 0 | 0 | 0 | 0 |
| 0 | 1 | 0 | 1 |
| 1 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 |

Output e stato coincidono, quindi:
$O_1=S_0S_1 + S_0S_1 = S_0(S_1 + S_1)=S_0$
$O_2=S_0S_1+S_0S_1=S_1(S_0 + S_0)=S_1$

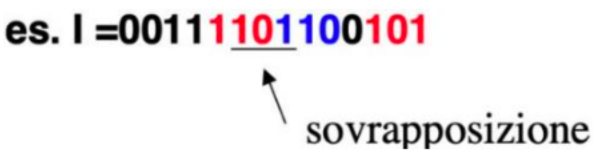
**Circuito finale**




Dalle espressioni scritte in precedenza otteniamo il seguente circuito.

# Esercizio 4

Progettare una rete sequenziale che:
- riceva in input un segnale I
- rilevi la presenza delle sequenze 101 e 110, anche sovrapposte.

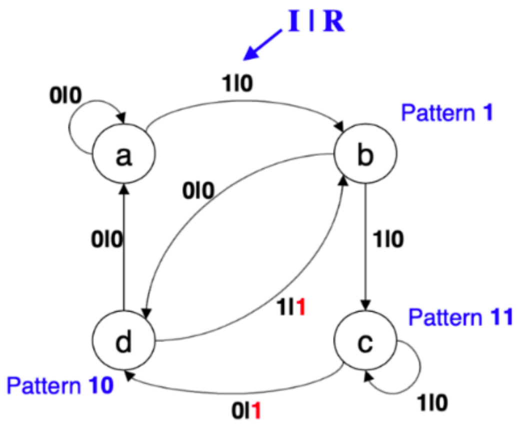



La rete ha un solo segnale di output R, tale che:
- R = 1 se una delle due sequenze è stata rilevata • R = O altrimenti

Si richiede di:

1. **Disegnare la macchina a stati finiti**
2. **Scrivere la tabella di verità**
3. **Trovare le forme SP minime**
4. **Disegnare il circuito**

## Macchina a stati finiti di Mealy




Le principali differenze con gli automi di Moore sono:

•Gli output non sono più posti negli stati (cerchi)

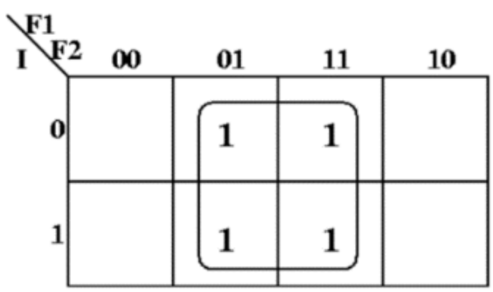
•Gli stati sono contrassegnati con un'etichetta (in questo caso a, b, c, d)

•Sugli archi troviamo input/output

## Tabella di verità "NextState" e mappe di Karnaugh

| | $F_1$ | $F_2$ | I | R | $F_1$* | $F_2$* |
|---|---|---|---|---|---|---|
| a | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0 | 0 | 1 | 0 | 0 | 1 |
| b | 0 | 1 | 0 | 0 | 1 | 0 |
| | 0 | 1 | 1 | 0 | 1 | 1 |
| d | 1 | 0 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 1 | 1 | 0 | 1 |
| c | 1 | 1 | 0 | 1 | 1 | 0 |
| | 1 | 1 | 1 | 0 | 1 | 1 |

F1* = F2




F2* = I

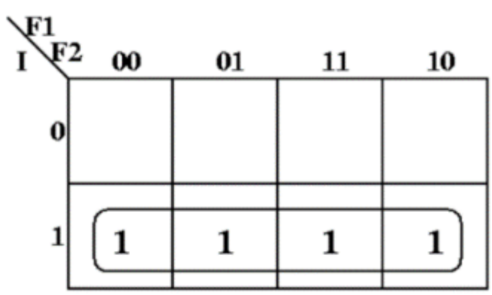


**Tabella di verità "Output" e mappa di Karnaugh**

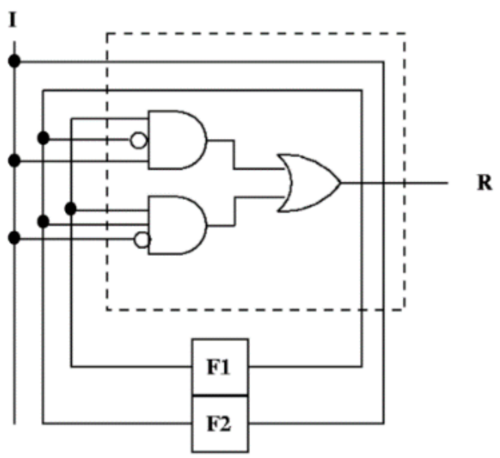
| | $F_1$ | $F_2$ | $I$ | $R$ | $F_1^*$ | $F_2^*$ |
|---|---|---|---|---|---|---|
| a | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0 | 0 | 1 | 0 | 0 | 1 |
| b | 0 | 1 | 0 | 0 | 1 | 0 |
| | 0 | 1 | 1 | 0 | 1 | 1 |
| d | 1 | 0 | 0 | 0 | 0 | 0 |
| | 1 | 0 | 1 | 1 | 0 | 1 |
| c | 1 | 1 | 0 | 1 | 1 | 0 |
| | 1 | 1 | 1 | 0 | 1 | 1 |

L'output è influenzato non solo dall'input ma anche dallo stato.

$$R = F_1F_2I + F_1F_2I$$

| I \ F1 F2 | 00 | 01 | 11 | 10 |
|---|---|---|---|---|
| 0 | | | 1 | |
| 1 | | | | 1 |

**Circuito finale**

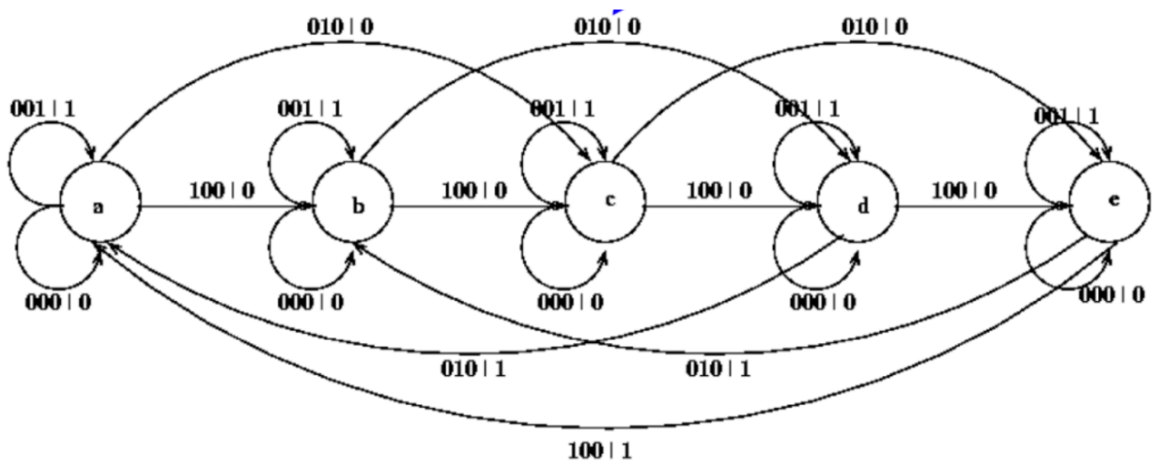


# <u>Esercizio 5</u>

Disegnare una macchina a stati finiti di Mealy per il controllo di un distributore automatico di bibite.

- Il costo di una bibita è di 50 centesimi
- Il distributore accetta monete da 10, 20, 50 centesimi.
- I segnali di ingresso I10, I20 e I50 vengono settati in corrispondenza della moneta
  introdotta. Può essere introdotta una sola moneta alla volta.
- L'uscita O vale
  - 1 se la cifra totale introdotta è >= 50
  - 0 altrimenti.
- Quando 0=1→ la cifra introitata viene ridotta di 50 centesimi e la bibita viene restituita
- Fare in modo che l'eventuale resto possa essere utilizzato dal cliente successivo.
  Scrivere le tabelle di verità relative alla macchina a stati finiti progettata.

## Macchina a stati finiti di Mealy




Se al clock non vi è alcun inserimento di monete, si può interpretare questa situazione come un input pari a “000”.

## Tabella di verità parziale

In questo caso sia le tavole di verità che le mappe di Karnaugh risultano “complicate” poiché:

| | S0 | S1 | S2 | I10 | I20 | I50 | O | S1* | S2* | S3* |
|---|---|---|---|---|---|---|---|---|---|---|
| a | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| | 0 | 0 | 0 | 0 | 1 | 1 | X | X | X | X |
| | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 |
| | 0 | 0 | 0 | 1 | 0 | 1 | X | X | X | X |
| | 0 | 0 | 0 | 1 | 1 | 0 | X | X | X | X |
| | 0 | 0 | 0 | 1 | 1 | 1 | X | X | X | X |
| b | 0 | 0 | 1 | 0 | 0 | 0 | | | | |
| | etc... | | | | | | | | | |

•nelle tavole per ogni stato ci sono otto combinazioni di possibili input;
•per le mappe, sarebbe necessario costruirne quattro (visto che ci sono sei variabili) per ogni stato successivo.