



Architettura degli Elaboratori I

Corso di Laurea Triennale in Informatica

Università degli Studi di Milano

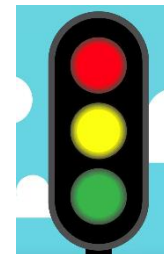
Dipartimento di Informatica "Giovanni Degli Antoni"

Edizione 2 (Cognomi H-Z), A.A. 2022-2023, Nicola.Basilico@unimi.it

Macchine a stati finiti (FSM)

Macchine a stati finiti (FSM)

- Macchine a Stati Finiti (Finite State Machines, Finite State Automata): è un modello matematico dell'elaborazione
- Si assume che l'elaborazione coincida con le operazioni fatte da una **macchina** che:
 - Riceve delle informazioni in **input** (i diversi possibili input sono in numero **finito**)
 - Ha uno **stato** interno (i diversi stati possibili sono in numero **finito**)
 - Emette delle informazioni in **uscita** (le diverse possibili uscite sono in numero **finito**)
- La macchina procede per passi in modo iterativo, ad ogni passo:
 - Calcola le uscite come funzione di input e stato corrente
 - Calcola un nuovo stato come funzione di input e stato corrente
 - Assume il nuovo stato appena calcolato come stato corrente
- Questo formalismo può essere usato per modellare sistemi in cui si svolgono elaborazioni che richiedono poca memoria, dove il concetto di stato raccoglie tutto quello che serve per portare avanti l'elaborazione



Cosa ci ricorda?

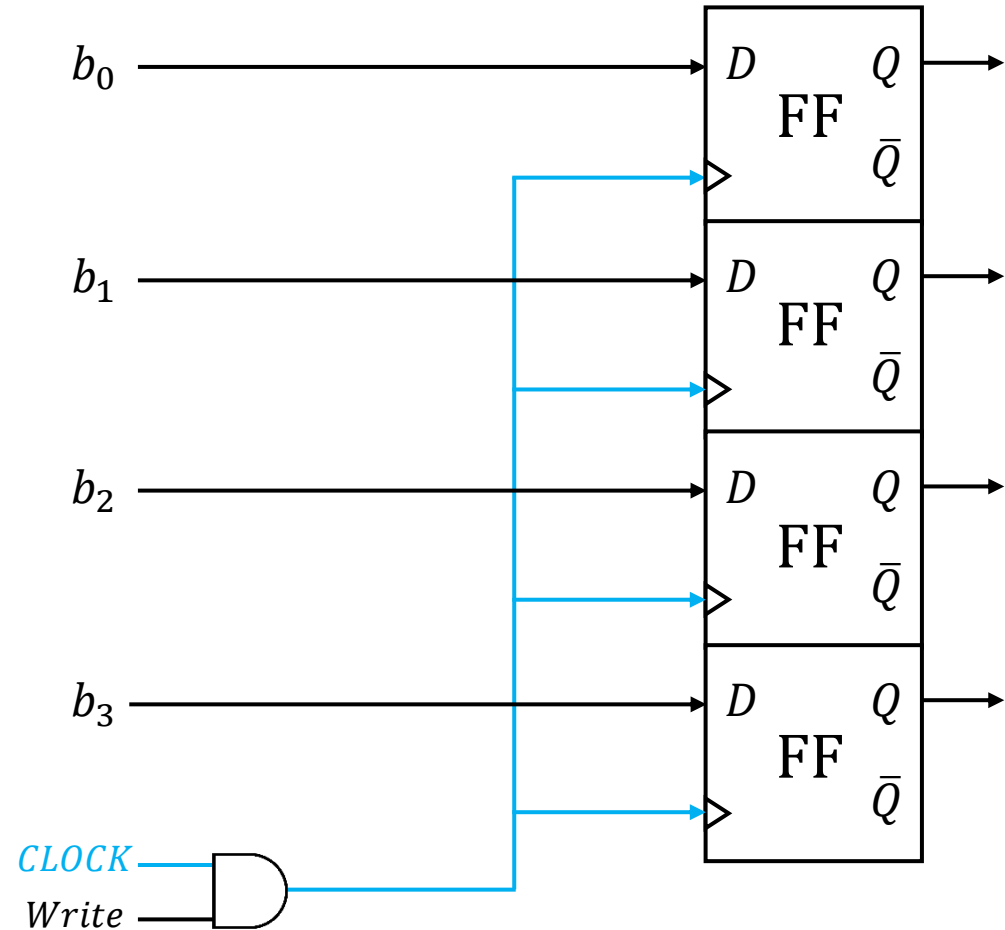
Macchine a stati finiti (FSM)

- Come definire circuiti sequenziali dove gli stati possibili possono essere più di 2?
- Abbandoniamo la logica di set/reset su cui si sono basati tutti i circuiti sequenziali elementari che abbiamo introdotto
- Possiamo mettere insieme più bistabili e rappresentare su n bit più di 2 stati, ma come gestire le transizioni?
- Per avere un circuito sequenziale con > 2 stati dobbiamo risolvere 2 problemi fondamentali:
 - **Ci serve più di un bit di memoria:** se il circuito può avere m stati, allora servirà una memoria con almeno $\lceil \log_2 m \rceil$ bit
 - **Ci serve un formalismo più astratto** per descrivere la dinamica del circuito (transizioni di stato e valori delle uscite)

Registri

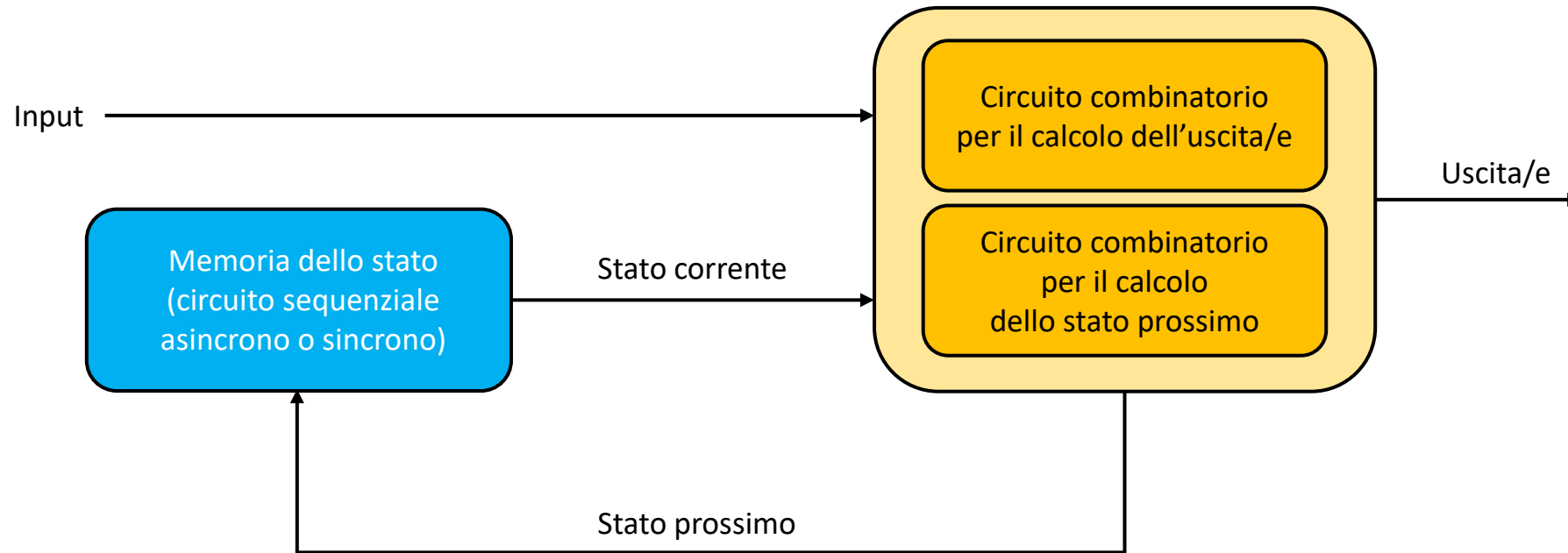
- Come si costruisce una memoria su più bit? Usando n Flip Flop, costruisco un registro da n bit

- Esempio:** registro su 4 bit con segnale di write (opzionale)
- Questo registro può contenere 2^4 diversi stati



Circuiti sequenziali

- Adottiamo questo schema generale per un circuito sequenziale



- **Esempio:** il bistabile SR può essere implementato come un caso particolare di questo schema in cui:
 - Input: s, r
 - Circuito per il calcolo delle uscite: lo stato Q (filo) e lo stato negato \bar{Q} (NOT)
 - Circuito per lo stato prossimo: $s + \bar{r}Q$
 - Memoria: 1 bit

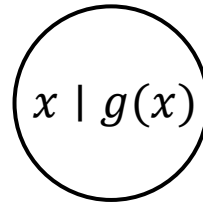
Formalizzazione della FSM

Dal punto di vista matematico, quali elementi definisco una FSM?

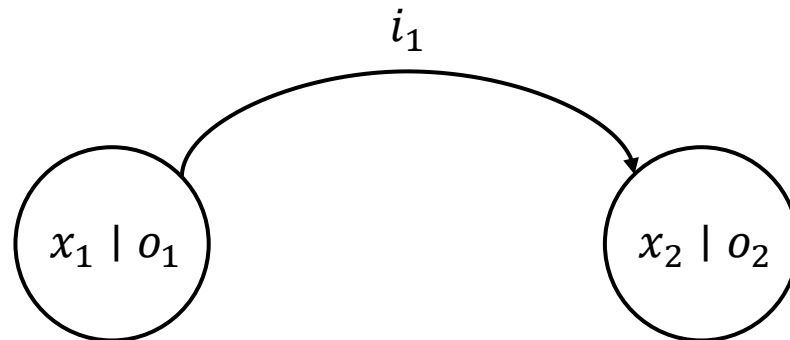
- $X = \{x_1, x_2, \dots, x_m\}$ insieme finito degli stati
- $x^i \in X$ stato iniziale della macchina
- $I = \{i_1, i_2, \dots, i_w\}$ insieme degli input che la macchina può ricevere
- $O = \{o_1, o_2, \dots, o_k\}$ insieme degli output che la macchina può generare
- $T: X \times I \rightarrow X$ funzione di transizione o di stato prossimo, dato lo stato corrente x e un input i , $T(x, i)$ rappresenta il prossimo stato della macchina
- $g()$ funzione di uscita:
 - $g: X \rightarrow O$ **macchina di Moore**: dato lo stato corrente x , $g(x)$ rappresenta l'uscita
 - $g: X \times I \rightarrow O$ macchina di Mealy: dato lo stato corrente x e l'input i , $g(x, i)$ rappresenta l'uscita

Grafo delle transizioni di stato

- Esiste un modo grafico più conveniente con cui rappresentare la FSM: **grafo orientato**
- Ogni nodo del grafo rappresenta uno stato x della FSM e l'uscita associata $g(x)$



- Ogni arco del grafo rappresenta una transizione tra due stati della macchina



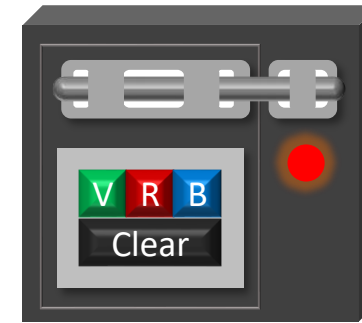
Dove:

- $g(x_1) = o_1, g(x_2) = o_2$
- $T(x_1, i_1) = x_2$

Esempio

Cassaforte con serratura digitale

- 4 pulsanti: V,R,B, Clear
- Led rosso: serratura chiusa
- Led verde: serratura aperta
- La sequenza V,R,B apre la cassaforte
- Il tasto clear
 - Se la cassaforte è chiusa resetta la pulsantiera
 - Se la cassaforte è aperta è l'unico modo per chiuderla (la pressione di altri tasti mentre la cassaforte è aperta non comporta nulla)



Problema: come formalizzare la FSM che implementa la logica di questa cassaforte?

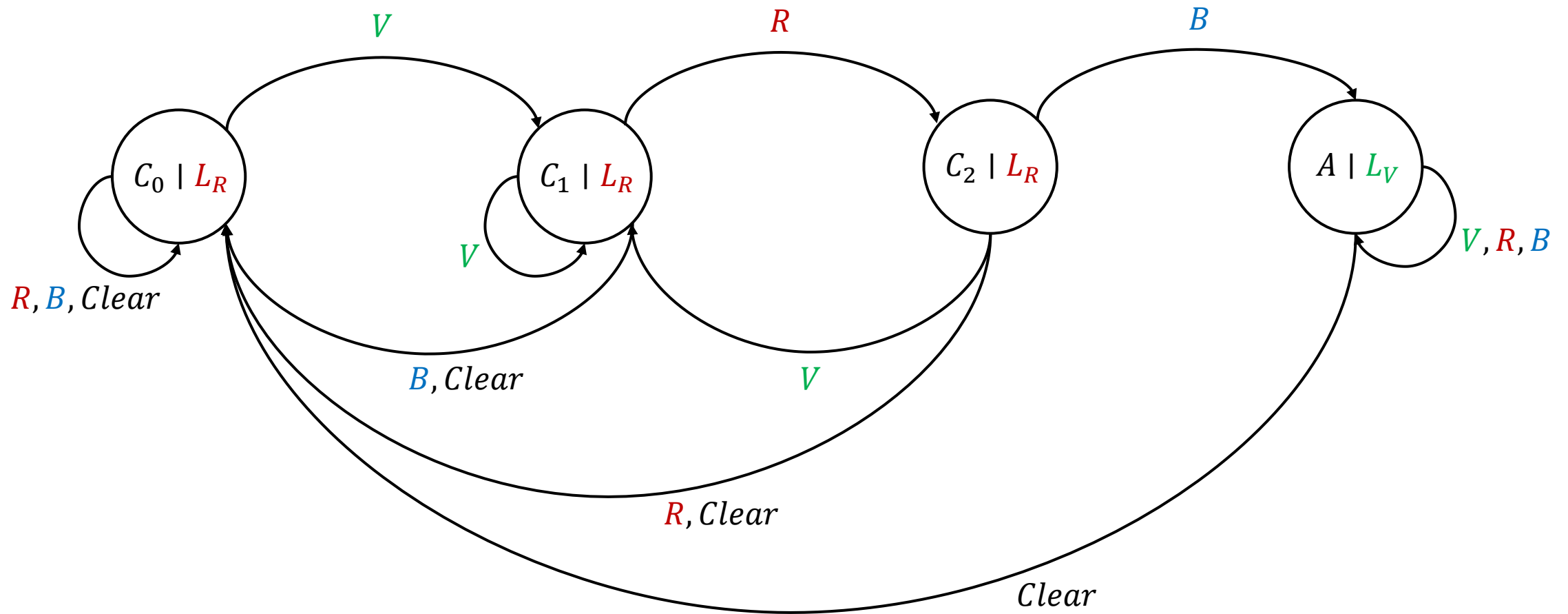
Esempio

Definiamo il modello matematico che serve per descrivere questa FSM

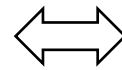
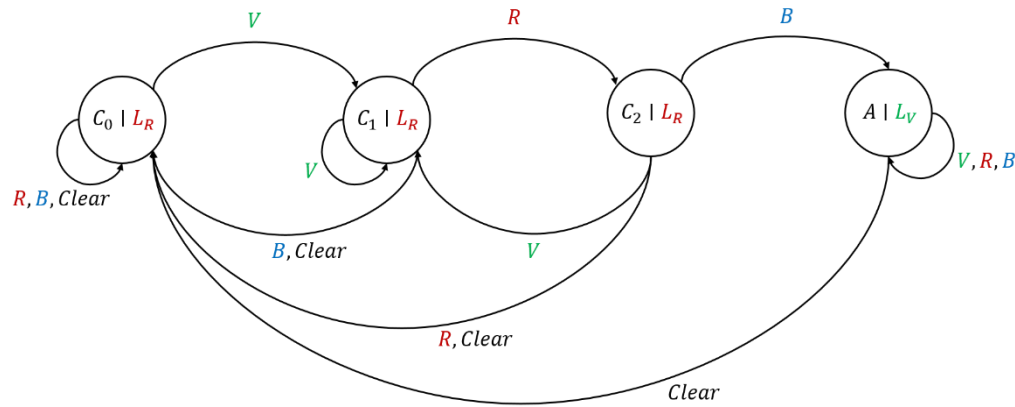
- Insieme degli stati $X = \{C_0, C_1, C_2, A\}$
 - Stato C_i serratura chiusa e ho riconosciuto i primi i simboli della combinazione
 - Stato A , serratura aperta
- Stato iniziale della macchina $C_0 \in X$
- Insieme degli input $I = \{V, R, B, Clear\}$
- Insieme degli output $O = \{L_R, L_V\}$
 - L_R led rosso
 - L_V verde
- Funzione di uscita $g(C_i) = L_R \ \forall i \in \{0,1,2\}, g(A) = L_V$

Esempio (grafo delle transizioni)

- Funzione di transizione



Esempio (tabella delle transizioni)



	V	R	B	Clear	Uscita
C ₀	C ₁	C ₀	C ₀	C ₀	L _R
C ₁	C ₁	C ₂	C ₀	C ₀	L _R
C ₂	C ₁	C ₀	A	C ₀	L _R
A	A	A	A	C ₀	L _V

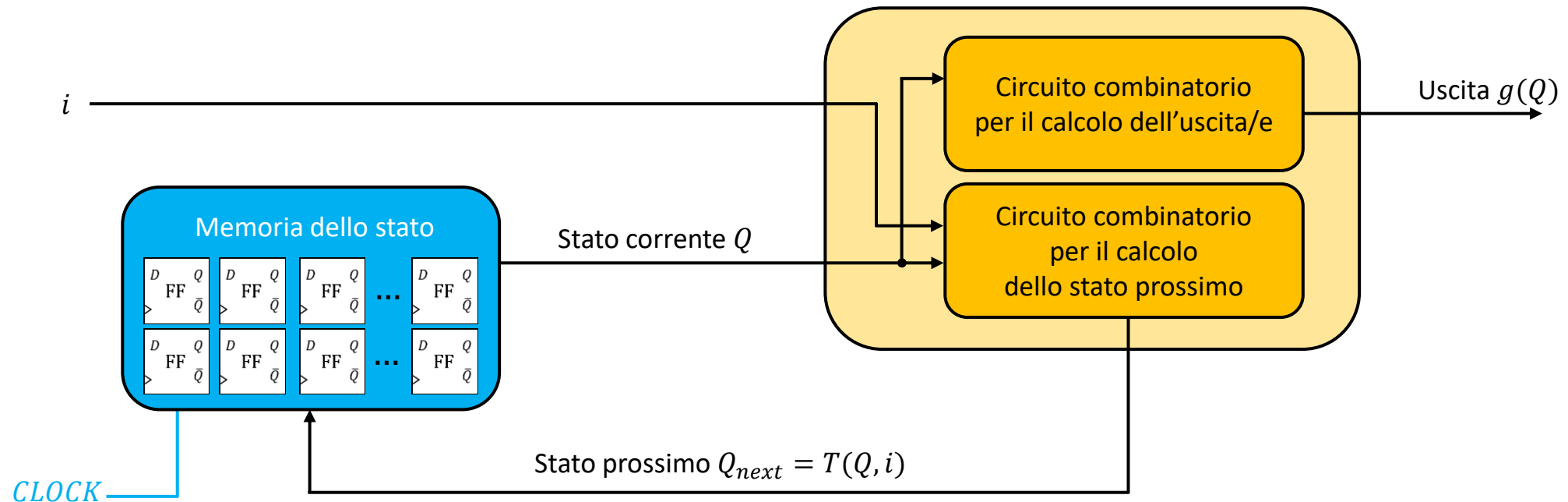
- La tabella delle transizioni riporta in forma tabellare la stessa specifica descritta dal grafo
- Per ogni stato corrente Q (sulla riga) e input i (su una colonna, esclusa l'ultima) viene riportato lo stato successivo $Q_{next} = T(Q, i)$
- L'ultima colonna riporta, per ogni stato Q , il valore dell'uscita $g(Q)$ (ricorda: nella macchina di Moore l'uscita dipende solo dallo stato corrente)

Sintesi di una FSM

- Problema: sintetizzare un circuito che implementa una macchina a stati finiti la cui specifica è descritta dal formalismo precedente
- In sostanza, una volta che abbiamo definito la specifica della macchina, **come è fatto** e **come si costruisce** il circuito che la implementa?
- **Cosa vuol dire che il circuito sequenziale implementa la specifica della FSM?**
 - Ha una memoria interna in cui mantiene un valore che rappresenta lo stato corrente. Se la FSM può avere m stati allora la memoria deve avere almeno $\lceil \log_2 m \rceil$ bit (useremo un Flip Flop per ogni bit)
 - Se in memoria abbiamo il valore che rappresenta lo stato Q e il circuito riceve un segnale che rappresenta l'input i , la memoria viene sovrascritta con il valore che rappresenta lo stato $T(Q, i)$ e l'uscita viene posta a $g(Q)$

Architettura del circuito di una FSM

- Cominciamo dalla prima domanda: **come è fatto il circuito che implementa la FSM?**
- Useremo lo schema introdotto in precedenza, che ora possiamo descrivere in modo più specifico:



- Cosa ci serve per costruire un circuito di questo tipo?
 1. Sapere da quanti Flip Flop è costituita la memoria dello stato
 2. Sapere come è fatto il circuito per il calcolo dell'uscita
 3. Sapere come è fatto il circuito per il calcolo dello stato prossimo