

ARCHITETTURA DEGLI ELABORATORI

A.A. 2020-2021

Università di Napoli Federico II
Corso di Laurea in Informatica

Docenti

Proff.

Luigi Sauro gruppo 1 (A-G)

Silvia Rossi gruppo 2 (H-Z)



Logiche sequenziali sincrone

- I circuiti asincroni presentano delle criticità a volte difficilmente analizzabili
 - Dipendono dalla struttura fisica dei componenti
- Per questo si cerca di evitare di retroazionare l'output in maniera diretta e si interpone un registro nel ciclo di retroazione
- *Nell'ipotesi che il clock sia più lento del ritardo accumulato sul cammino, il registro consente al sistema di essere sincronizzato col clock: circuito *sincrono**

Logiche sequenziali sincrone

- In generale un circuito sequenziale sincrono ha un insieme finito di stati $\{S_0, \dots, S_{k-1}\}$
- Logica combinatoria:

$$\text{out} = f(\text{in})$$

- Logica sequenziale sincrona:

$$\text{out} = f(\text{in}, s_c)$$

$$s_n = g(\text{in}, s_c)$$

Design di logiche sequenziali sincrone

- Inserire registri nei cammini ciclici
- I registri determinano lo **stato** S_0, \dots, S_{k-1} del sistema
- I cambiamenti di stato sono determinati dalle transizioni del clock: il sistema è sincronizzato con il clock
- *Regole* di composizione:
 - Ogni componente è un registro o un circuito combinatorio
 - Almeno un componente è un registro
 - Tutti i registri sono sincronizzati con un unico clock
 - Ogni ciclo contiene almeno un registro
- Due tipici circuiti sequenziali sincroni
 - Finite State Machines (FSMs)
 - Pipelines

Finite State Machines

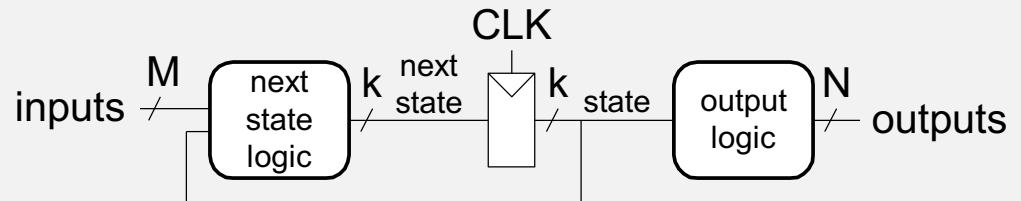
- s_n dipende sia dall'input che da s_c

$$s_n = g(in, s_c)$$

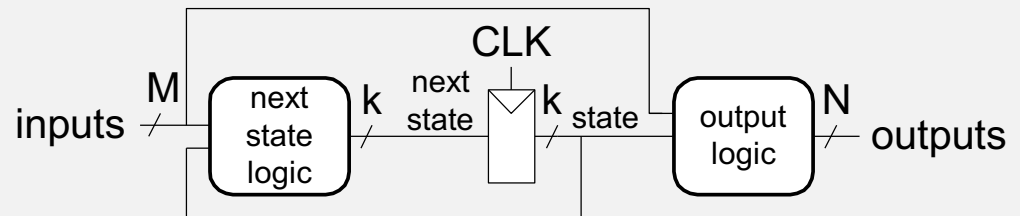
- 2 tipi di FSM a seconda della logica di output:

- Moore FSM: $out = f(s_c)$
- Mealy FSM: $out = f(in, s_c)$

Moore FSM



Mealy FSM



Encoding degli stati

- Encoding binario:
 - i.e., per 4 stati, 00, 01, 10, 11
- Encoding *one-hot*
 - Un bit per stato
 - Solo un bit HIGH alla volta
 - i.e., per 4 stati, 0001, 0010, 0100, 1000
 - Richiede più flip-flops
 - Spesso la logica combinatoria associata è più semplice

Moore vs Mealy FSM

Alyssa P. Hacker has a snail that crawls down a paper tape with 1's and 0's on it. The snail smiles whenever the last two digits it has crawled over are 01. Design Moore and Mealy FSMs of the snail's brain.



10011000101101

goal=1

dist_prox_goal ≥ 2

10011000101101

goal=0

dist_prox_goal ≥ 2

10011000101101

goal=0

dist_prox_goal ≥ 1

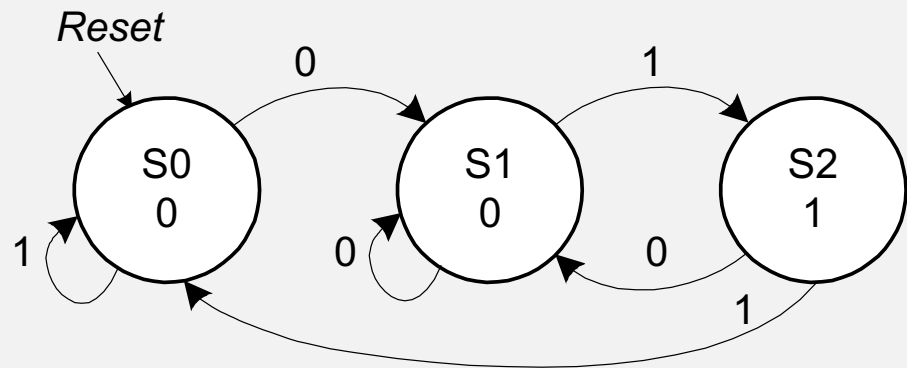
10011000101101

Moore vs Mealy FSM

Alyssa P. Hacker has a snail that crawls down a paper tape with 1's and 0's on it. The snail smiles whenever the last two digits it has crawled over are 01. Design Moore and Mealy FSMs of the snail's brain.



Moore FSM



Moore vs Mealy FSM

Alyssa P. Hacker has a snail that crawls down a paper tape with 1's and 0's on it. The snail smiles whenever the last two digits it has crawled over are 01. Design Moore and Mealy FSMs of the snail's brain.



10011000101101

0→

goal=0; dist_prox_goal ≥ 1

1→goal=0; dist_prox_goal ≥ 2

10011000101101

0→goal=0; dist_prox_goal ≥ 1

1→ goal=1; dist_prox_goal ≥ 2

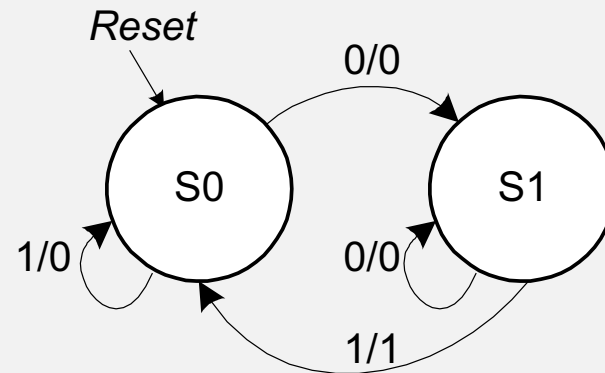
10011000101101

Moore vs Mealy FSM

Alyssa P. Hacker has a snail that crawls down a paper tape with 1's and 0's on it. The snail smiles whenever the last two digits it has crawled over are 01. Design Moore and Mealy FSMs of the snail's brain.



Mealy FSM



Moore FSM State Transition Table

Current State		Inputs	Next State	
S_1	S_0		S'_1	S'_0
0	0	0		
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		

State	Encoding
S0	00
S1	01
S2	10



Tabella transizione Moore FSM

Current State		Inputs	Next State	
S_1	S_0		S'_1	S'_0
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0

State	Encoding
S0	00
S1	01
S2	10

$$S'_1 = S_0 A$$

$$S'_0 = \overline{A}$$

Tabella transizione Moore FSM

Current State		Inputs	Next State	
S_1	S_0		S'_1	S'_0
0	0	0	0	1
0	0	1	0	0
0	1	0	0	1
0	1	1	1	0
1	0	0	0	1
1	0	1	0	0

State	Encoding
S0	00
S1	01
S2	10

$$S'_1 = S_0 A$$

$$S'_0 = \overline{A}$$

Manca S_1 negato, perché?

Moore FSM Output Table

Current State		Output
s_1	s_0	Y
0	0	
0	1	
1	0	

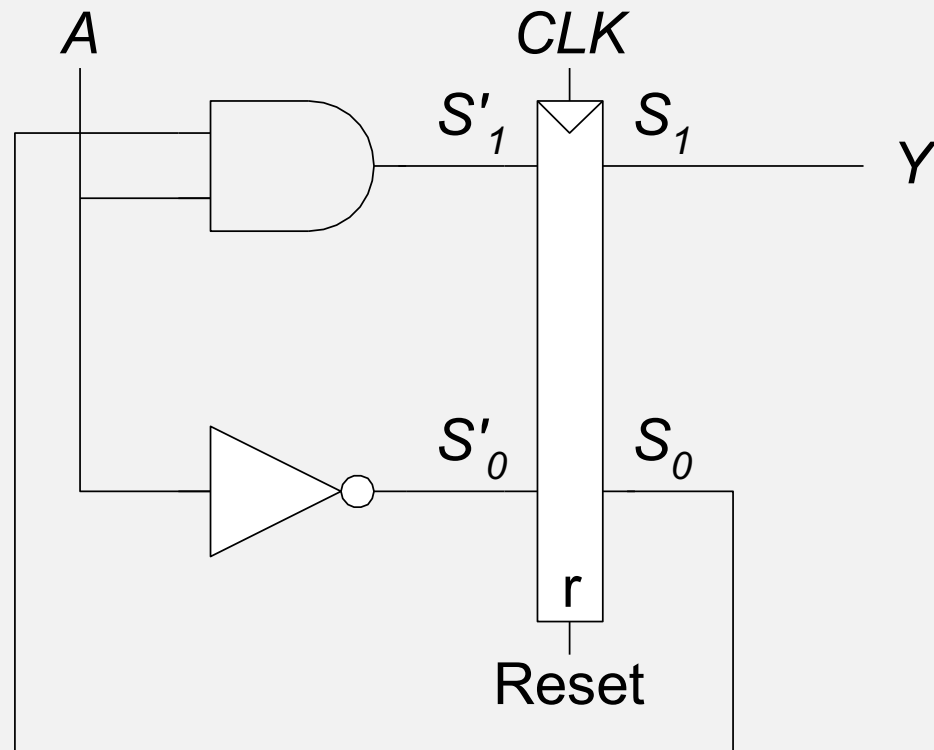


Tabella output Moore FSM

Current State		Output
S_1	S_0	Y
0	0	0
0	1	0
1	0	1

$$Y = S_1$$

Schema Moore FSM



Mealy FSM State Transition & Output Table

Current State	Input	Next State	Output
S_0	A	S'_0	Y
0	0		
0	1		
1	0		
1	1		

State	Encoding
S0	00
S1	01



Tabella transizione/output Mealy FSM

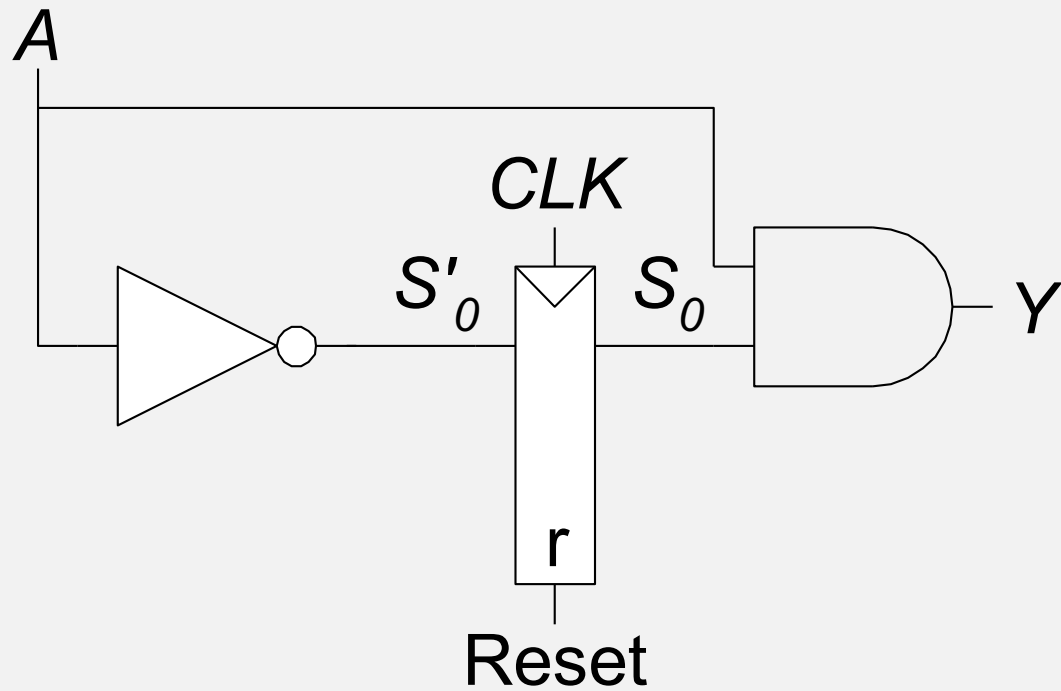
Current State	Input	Next State	Output
S	A	S'	Y
0	0	1	0
0	1	0	0
1	0	1	0
1	1	0	1

State	Encoding
S0	0
S1	1

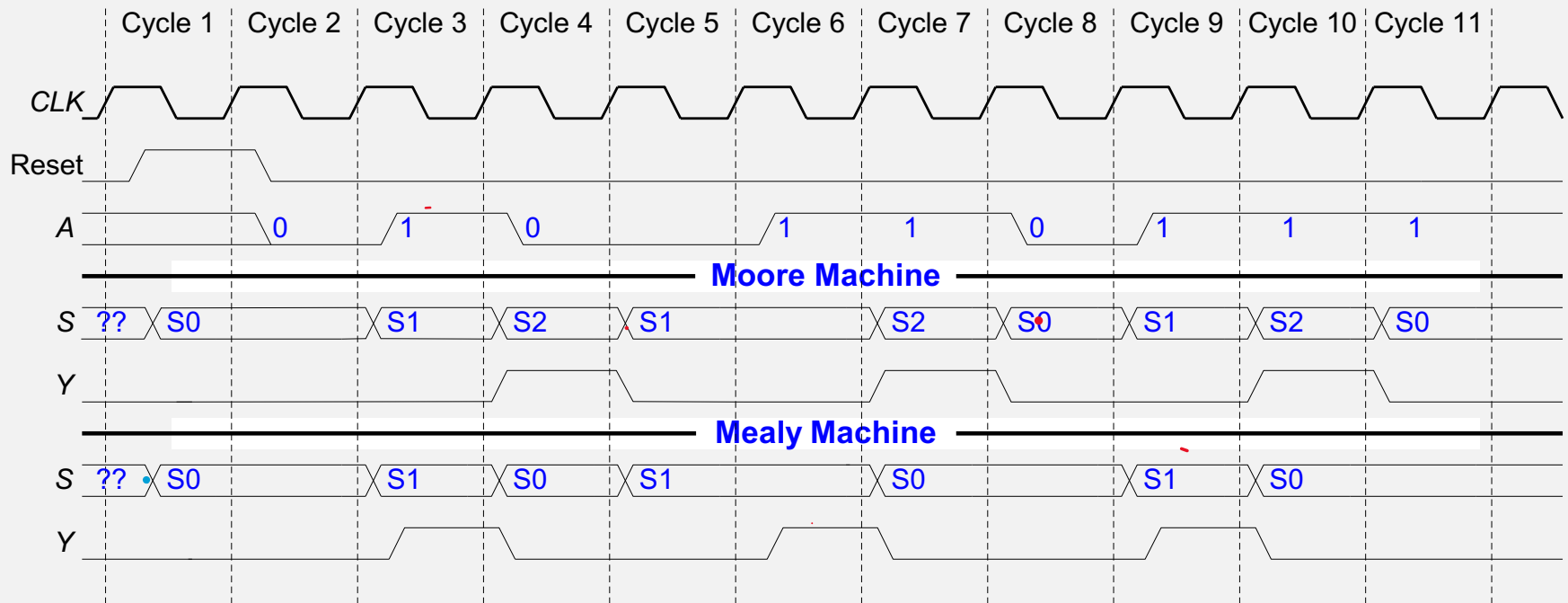
$$S' = \overline{A}$$

$$Y = SA$$

Schema Mealy FSM



Moore & Mealy Timing Diagram

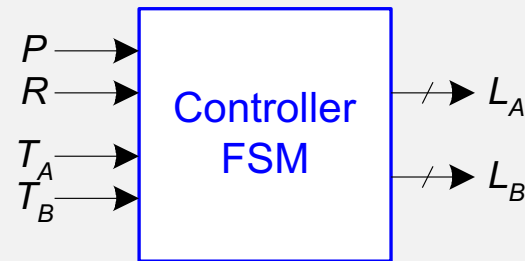


Fattorizzazione di FSM

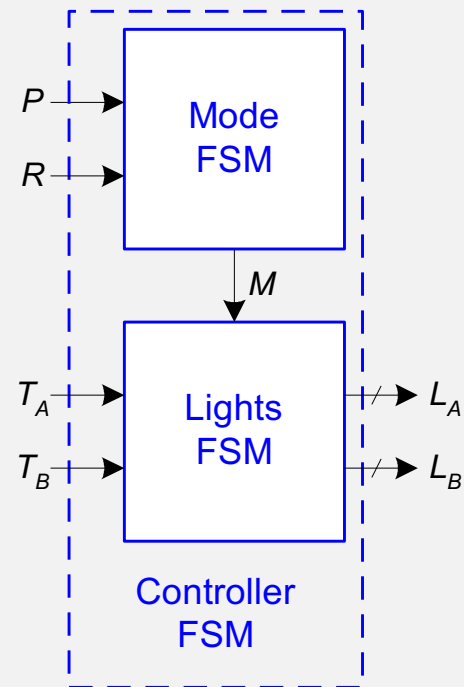
- Fattorizzare consiste nel suddividere una FSM complessa in FSM più piccole che interagiscono fra loro
- Esempio: Considerate di voler modificare il controller di semafori per tener conto di possibili parate
 - Altri due inputs: P, R
 - Se $P = 1$, entra in modalità *Parade* e il semaforo di Bravado Blvd rimane verde
 - Se $R = 1$, lascia la modalità *Parade*

Parade FSM

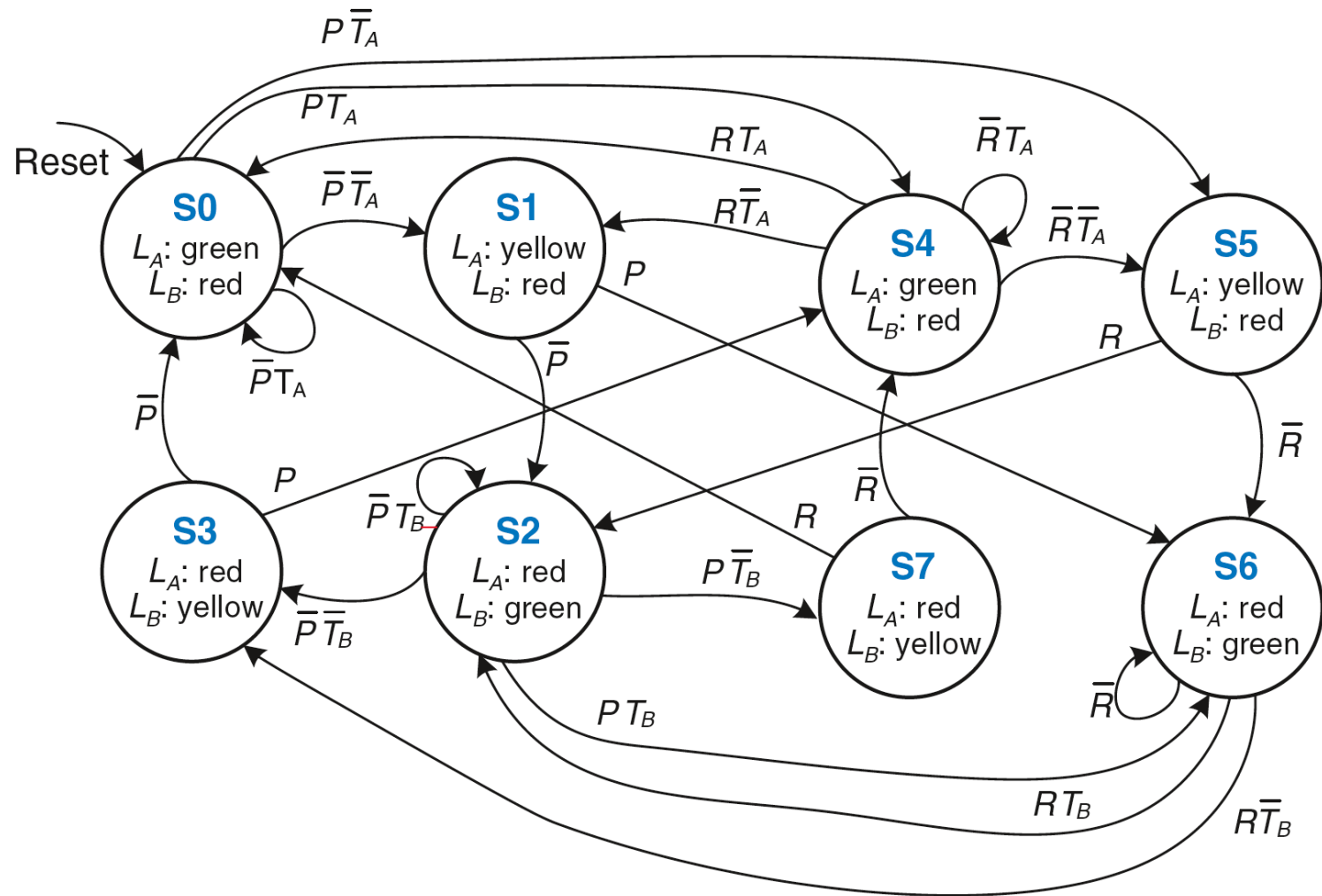
FSM non fattorizzato



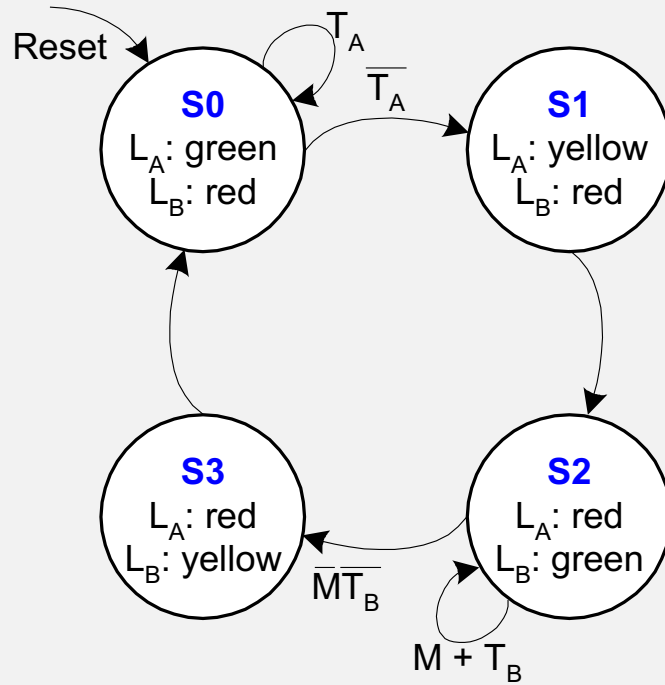
FSM fattorizzato



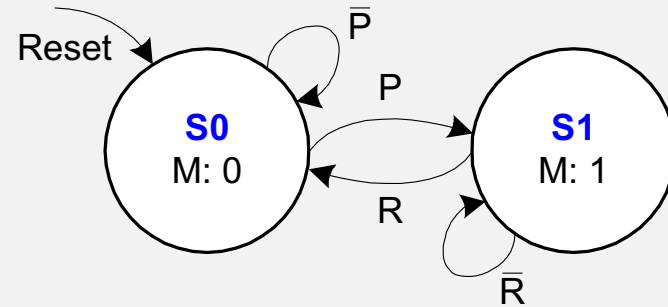
FSM non fattorizzato



FSM fattorizzato



Lights FSM



Mode FSM

Progettare una FSM

1. Identificare gli input e output
2. Abbozzare uno state transition diagram
3. Scrivere la state transition table
4. Selezionare un encoding degli stati
5. Macchina di Moore/Mealy:
 - a. Riscrivere la state transition table con l'encoding degli stati
 - b. Scrivere la output table
6. Scrivere le equazioni booleane relative alla logica di prossimo stato e alla logica di output
6. Minimizzare le equazioni
7. Fare uno schema del circuito

Esempio

- Progettare una Mealy FSM F con due input (A e B) e un output Q.
 - $Q=1$ sse A e B assumono rispettivamente il valore precedente
 - es:

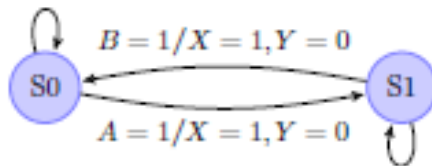
A	0	0	1	0	0	0	1	1	0
B	1	1	1	1	1	0	1	1	1
Q	0	1	0	0	1	0	0	1	0

Esercizi

■ Esercizi 3.23, 3.31

3. Il seguente diagramma di transizione per una macchina di Mealy ha due input A e B e due output X e Y . Indicare le formule SOP **minime** relative alla variabile di stato (S) e alle due variabili di output.

$A = 0/X = 0, Y = 1$



$B = 0/X = 0, Y = 1$

Codifica dello stato:

stato	S
S0	0
S1	1

Formule minime SOP:

- S' : _____
- X : _____
- Y : _____