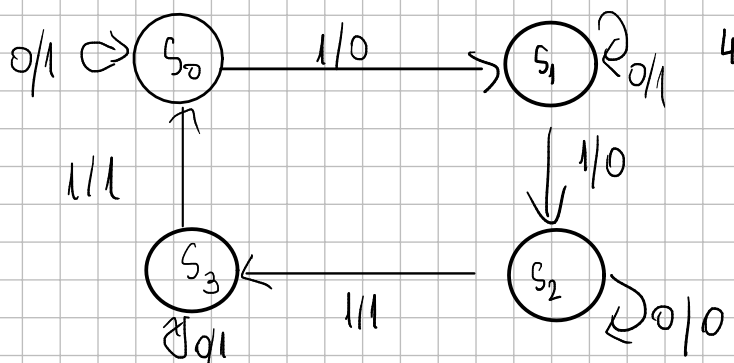
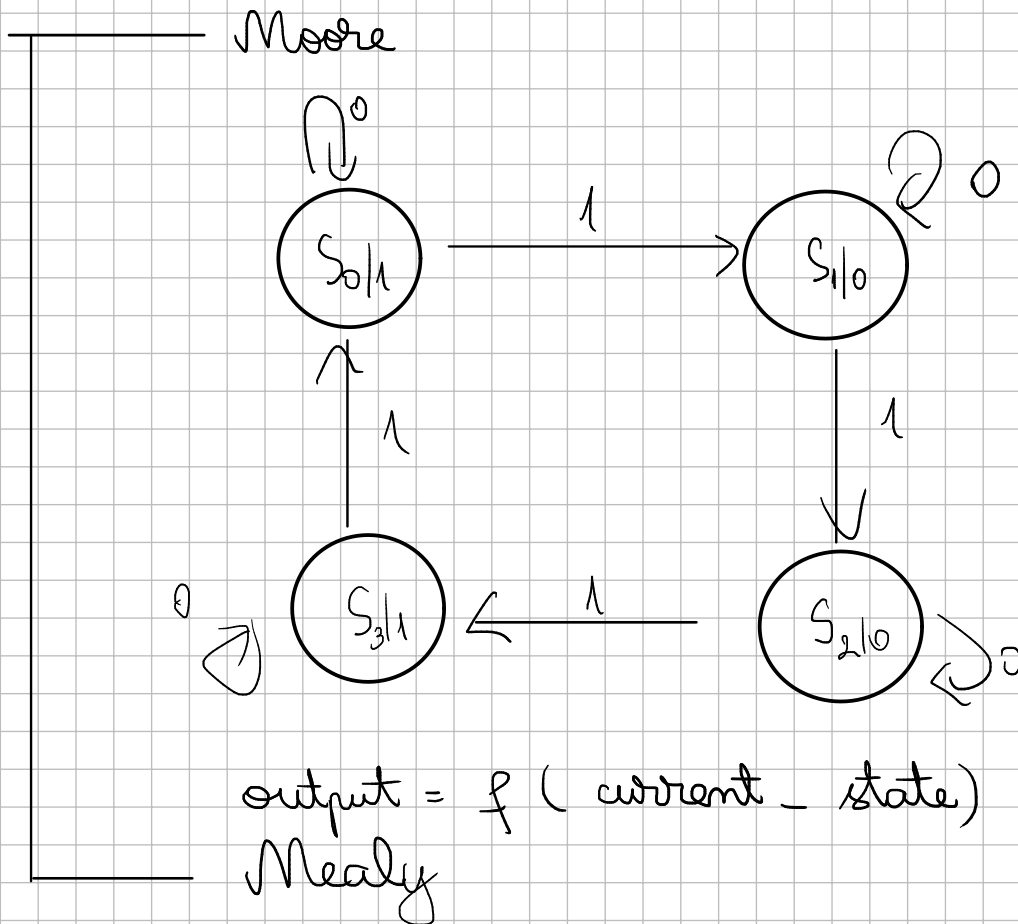


1p. identificare tip FSM



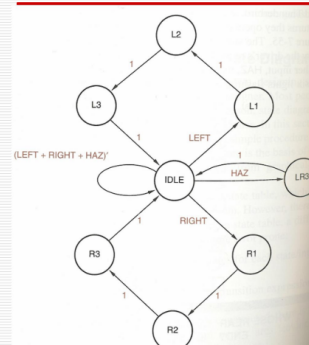
$$\text{output} = f(\text{current_state}, \text{input})$$

0.5 p este sau nu ambiguu

- Probleme care pot apărea:
1. Diagrama de stare ambiguă! Pot apărea situații în care să existe perechi (stare, intrări) pentru care să nu fie specificată starea următoare.
 2. Nu este garantat faptul că condițiile pentru tranziție asociate arcelor care pleacă dintr-o stare acoperă toate combinațiile de intrare exact o dată.

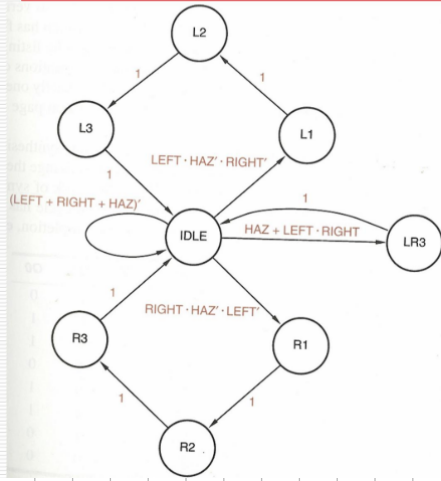
☐ Ce se întâmplă dacă LEFT și HAZ sunt asertuite simultan?

Ex.: semnalizare Ford diagrama Moore



Output Table						
State	LC	LB	LA	RA	RB	RC
IDLE	0	0	0	0	0	0
L1	0	0	1	0	0	0
L2	0	1	1	0	0	0
L3	1	1	1	0	0	0
R1	0	0	0	1	0	0
R2	0	0	0	1	1	0
R3	0	0	0	1	1	1
LR3	1	1	1	1	1	1

Probleme?



- ❑ Solutie: Ii dam la intrarea HAZ prioritate, iar cazul LEFT si RIGHT asertuite simultan il tratam ca si hazard.
- ❑ Noua diagrama nu e ambigua: conditiile de pe arce se exclud reciproc si surprind toate combinatiile de intrare!

1p. identificare nr. stari, nr. intrari, nr. iesiri

a) 4 stări, 1 intrare, 1 ieșire

b) — || —

1 p tabelul cu codificarea stărilor, stabilire nr biti pt codificare stare

a)

state	codificatore
S0	00
S1	01
S2	10
S3	11

2 bits

1p tabelul de tranzitii la starea urmatoare

current_state		input	next_state	
s_0 {	0 0	0	0	0
	0 0	1	0	1
s_1 {	0 1	0	0	1
	0 1	1	1	0
s_2 {	1 0	0	1	0
	1 0	1	1	1
s_3 {	1 1	0	1	1
	1 1	1	0	0

1p. tabel iesiri

a)

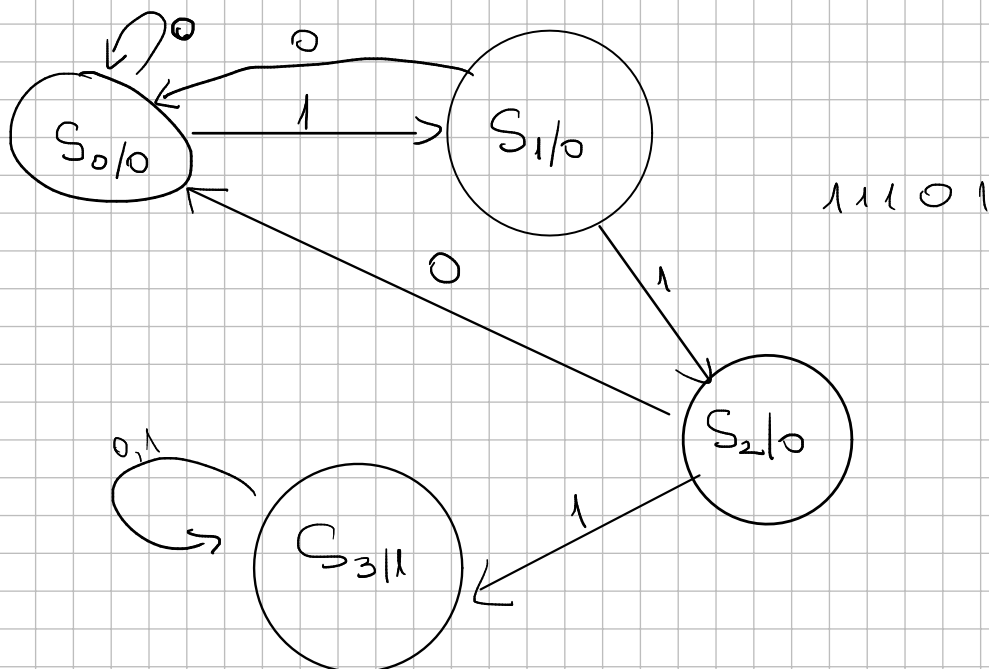
stare	iesire (1 bit)
0 0	1
0 1	0
1 0	0
1 1	1

b)

stare	intrare	iesire
0 0	0	1
0 0	1	0
0 1	0	1
0 1	1	0
1 0	0	0
1 0	1	1
1 1	0	1
1 1	1	1

0.5 p dezambiguizare

1. Determinați graful de tranziții al unui automat care semnalează pe ieșire existența pe intrare a 3 sau mai mulți biți secvențiali și consecutivi egali cu 1.

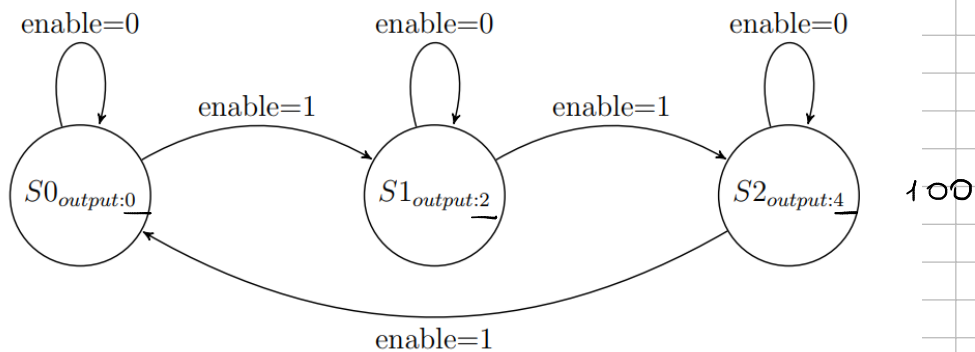


4 stări, 1 intrare, 1 ieșire

stare	Q ₁ Q ₀ codificare	
S ₀	0	0
S ₁	0	1
S ₂	1	0
S ₃	1	1

current Q ₁ Q ₀	input	next Q ₁ ' Q ₀ '
0 0	0	0 0
0 0	1	0 1
0 1	0	0 0
0 1	1	1 0
1 0	0	0 0
1 0	1	1 1
1 1	X	1 1

stare Q ₁ Q ₀	iesire
0 0	0
0 1	0
1 0	0
1 1	1



3 stări, 1 intrare, 3 ieșiri

stare	codificare	C - S	input	m - b
S ₀	00	00	0	00
		00	1	01
S ₁	01	01	0	01
		01	1	10
S ₂	10	10	0	10
		10	1	00

stare	S ₂ , S ₁ , S ₀ iesire
00	000
01	010
10	100