Capitolul 8

Automate cu stări finite

- un automat cu stări finite se defineşte formal prin cvintuplul

$$A = (X, Y, Q, \delta, \lambda),\ \text{unde}$$

$X = \{x_1, x_2, ..., x_n\}$ - mulțimea configurațiilor binare de intrare,
$Y = \{y_1, y_2, ..., y_r\}$ - mulțimea configurațiilor binare de ieşire,
$Q = \{q_1, q_2, ..., q_p\}$ - mulțimea configurațiilor binare de stare,
$\delta : X \times Q \to Q$ - funcția de tranziție a stărilor,
$\lambda : X \times Q \to Y$ - funcția de tranziție a ieşirilor.

- dacă $\lambda : X \times Q \to Y$ automatul este de tip Mealy, iar dacă $\lambda : Q \to Y$ este de tip Moore

- spațiul timpului este discret şi este format din mulțimea multiplilor de T , unde T este perioada semnalului de ceas care produce o modificare a stării circuitului.

- automatele sunt imediate (ieşirile se modifică imediat ce se modifică intrările sau stările) sau cu întârziere (ieşirile se modifică cu o întârziere de o perioadă de ceas față de modificarea intrărilor sau a stărilor)

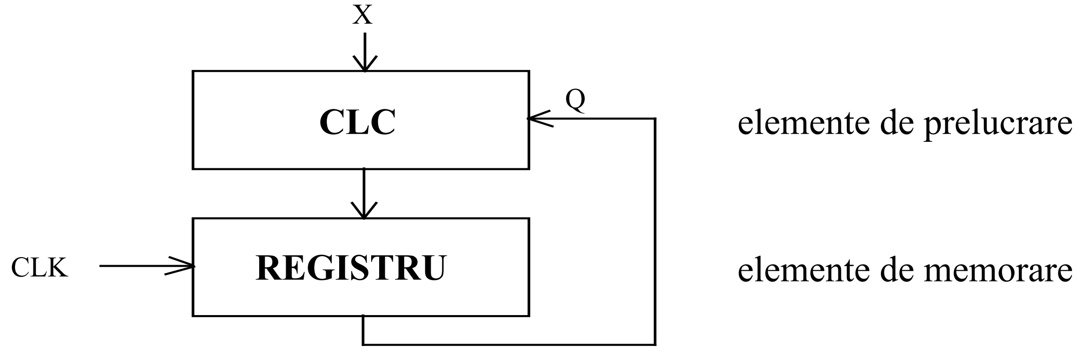


Fig. 8.1 *Separarea funcțională a unui automat finit*

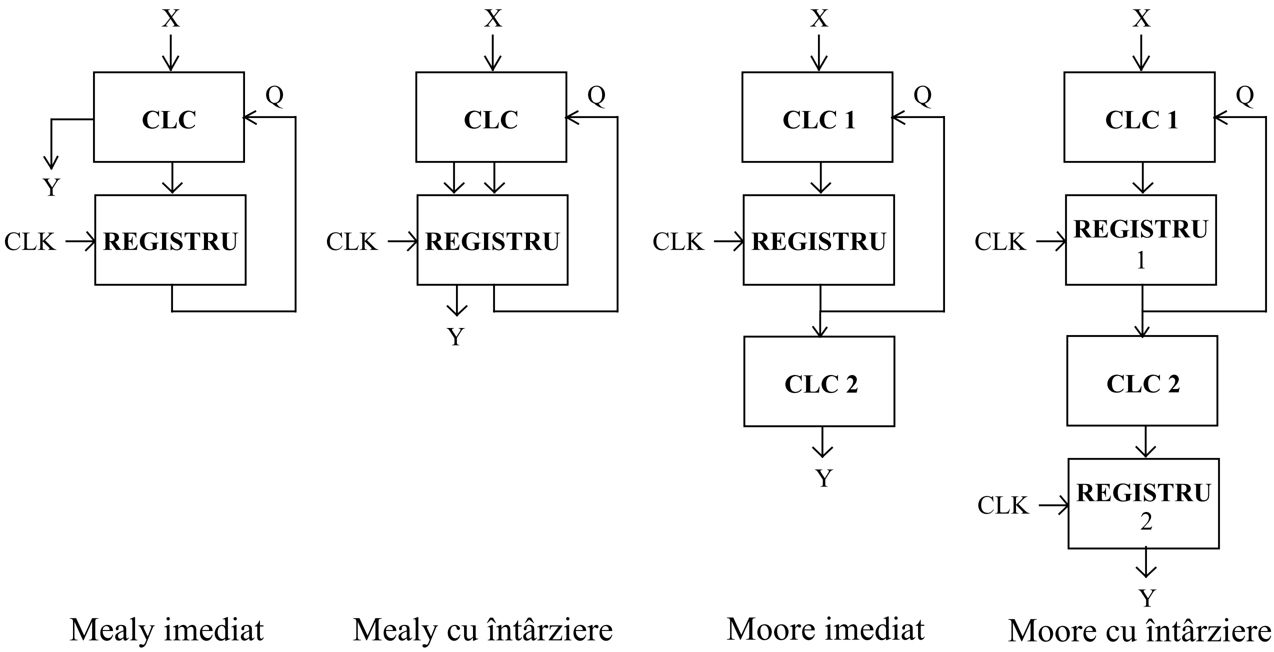


Fig. 8.2 *Structuri fundamentale de automate finite*

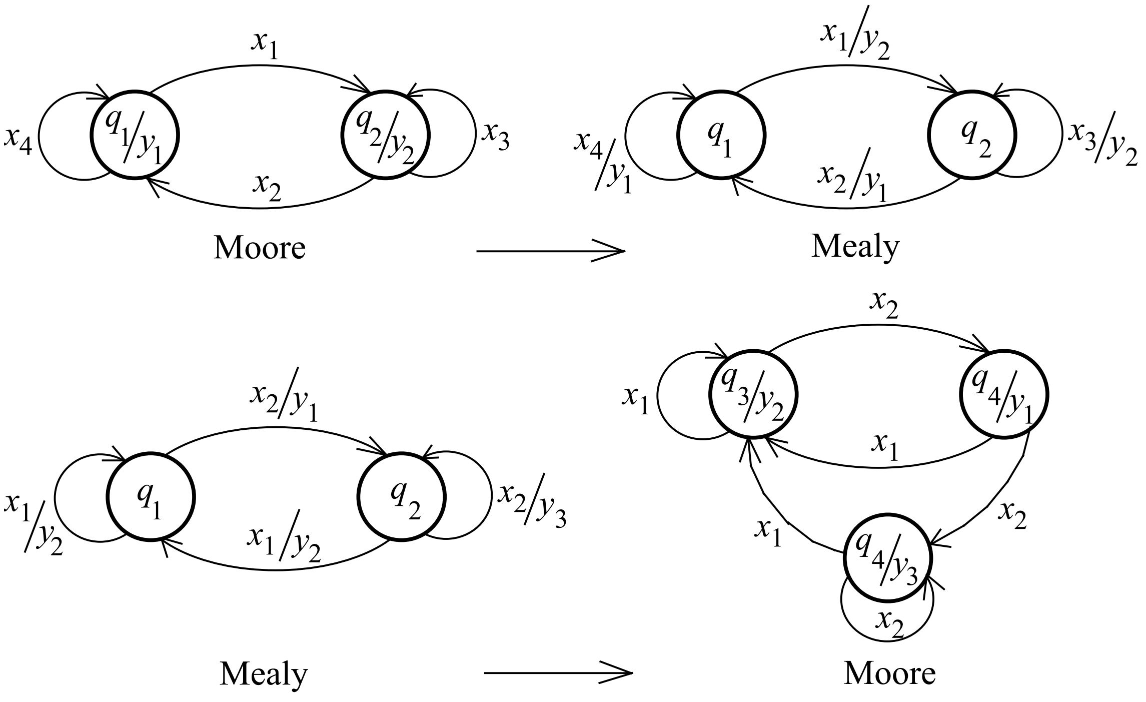


Fig. 8.3 *Transformarea automatelor finite*

- problema fundamentală care se pune la sinteza automatelor finite este cea a <u>optimizării</u>; se poate acționa numai asupra mulțimii stărilor, deci efecte importante de minimizare globală se pot obține prin <u>reducerea stărilor</u> şi <u>codificarea</u> optimă a lor

8.1. Reducerea numărului de stări

- două stări Q_i şi Q_j ale unui automat $A = (X, Y, Q, \delta, \lambda)$ sunt <u>echivalente</u> dacă $\lambda(Q_i, x^k) = \lambda(Q_j, x^k)$ şi $\delta(Q_i, x^k) = \delta(Q_j, x^k)$ pentru orice secvenţă finită posibilă de intrare x^k.
- <u>metoda tabelului implicaţiilor</u> reduce numărul stărilor automatelor complet specificate prin partiţionarea mulţimii stărilor în clase de echivalenţă:
- tabelul conţine un număr de compartimente (egal cu numărul tuturor perechilor posibile de stări) aranjate într-un tabel triunghiular; în fiecare compartiment se trece:
  - ♦ simbolul $\times$ dacă stările din perechea respectivă sunt evident neechivalente
  - ♦ simbolul ✓ dacă stările din perechea respectivă sunt evident echivalente
  - ♦ implicaţiile privind echivalenţa succesorilor, dacă stările din perechea respectivă sunt 1-echivalente (adică $\lambda(Q_i, x^k) = \lambda(Q_j, x^k)$).

- **<u>exemplu</u>:**

starea prezentă	starea următoare/ieşire			
	x_1	x_2	x_3	x_1
Q_1	$Q_1/0$	$Q_3/0$	$Q_4/0$	$Q_6/1$
Q_2	$Q_4/1$	$Q_2/0$	$Q_5/0$	$Q_7/1$
Q_3	$Q_2/0$	$Q_5/1$	$Q_3/0$	$Q_4/1$
Q_4	$Q_2/1$	$Q_4/0$	$Q_5/0$	$Q_8/1$
Q_5	$Q_5/1$	$Q_2/0$	$Q_4/0$	$Q_7/1$
Q_6	$Q_4/0$	$Q_5/1$	$Q_6/0$	$Q_6/1$
Q_7	$Q_1/0$	$Q_6/1$	$Q_7/1$	$Q_4/0$
Q_8	$Q_1/0$	$Q_6/1$	$Q_7/1$	$Q_4/0$

Fig. 8.4 *Automat finit complet specificat*

	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6	Q_7
Q_2	×						
Q_3	×	×					
Q_4	×	$Q_7=Q_8$	×				
Q_5	×	$Q_4=Q_5$	×	$Q_2=Q_5$ $Q_2=Q_4$ $Q_7=Q_8$			
Q_6	×	×	$Q_2=Q_4$ $Q_4=Q_6$	×	×		
Q_7	×	×	×	×	×	×	
Q_8	×	×	×	×	×	×	√

Fig. 8.5 *Tabelul triunghiular al implicațiilor*

- stările Q_4 şi Q_6 sunt neechivalente, deci şi Q_3 şi Q_6 sunt cert neechivalente
- pentru determinarea claselor de echivalență se grupează perechile de stări echivalente în baza <u>proprietății de tranzitivitate a relației de echivalență</u>
- $\mathrm{P}=\{\{Q_1\},\{Q_2,Q_4,Q_5\},\{Q_3\},\{Q_6\},\{Q_7,Q_8\}\}$ este partiția în clase de echivalență pentru automatul definit în figura 8.4. Automatul redus minim are numai 5 stări.
- două stări Q_i şi Q_j ale unui automat incomplet specificat $A=(X,Y,Q,\delta,\lambda)$ sunt <u>compatibile</u> dacă $\lambda(Q_i,x^k)=\lambda(Q_j,x^k)$ şi $\delta(Q_i,x^k)\approx\delta(Q_j,x^k)$ pentru orice secvență finită posibilă de intrare x^k.
- determinarea claselor de compatibilități se face tot cu tabelul implicațiilor, dar aici lipseşte proprietatea de tranzitivitate (pentru ca o stare să fie compatibilă cu stările unei clase de compatibilități, ea trebuie să fie compatibilă cu fiecare din stările clasei respective, în parte)

- **exemplu**:

starea prezentă	starea următoare/ieşire x_1	x_2	x_3	x_1
Q_1	$Q_3/0$	-/-	$Q_4/1$	$Q_2/-$
Q_2	-/-	$Q_2/1$	$Q_6/-$	$Q_3/1$
Q_3	$Q_2/0$	$Q_6/1$	-/-	-/-
Q_4	-/-	$Q_3/1$	$Q_2/0$	$Q_3/-$
Q_5	$Q_1/0$	$Q_6/0$	$Q_4/-$	-/0
Q_6	$Q_3/1$	$Q_2/-$	- /0	$Q_3/1$
Q_7	$Q_2/0$	$Q_4/-$	- /1	$Q_5/0$

Fig. 8.6 *Automat finit incomplet specificat*

Q_2	$Q_2 \sim Q_3$ $Q_4 \sim Q_6$					
Q_3	$Q_2 \sim Q_3$	$Q_2 \sim Q_6$				
Q_4	$\times$	$Q_2 \sim Q_3$ $Q_2 \sim Q_6$	$Q_3 \sim Q_6$			
Q_5	$Q_1 \sim Q_3$	$\times$	$Q_1 \sim Q_2$	$Q_2 \sim Q_4$ $Q_3 \sim Q_6$		
Q_6	$\times$	$\surd$	$\times$	$Q_2 \sim Q_3$	$\times$	
Q_7	$Q_2 \sim Q_3$ $Q_2 \sim Q_5$	$\times$	$Q_4 \sim Q_6$	$\times$	$Q_1 \sim Q_2$ $Q_4 \sim Q_6$	$\times$
	Q_1	Q_2	Q_3	Q_4	Q_5	Q_6

Fig. 8.7 *Tabelul triunghiular al implicaţiilor*

- stările Q_3 şi Q_6 sunt incompatibile, deci şi stările Q_3 şi Q_4 sunt în mod cert incompatibile
- stările Q_i, Q_j, Q_k fac parte din aceeaşi clasă de compatibilităţi, numai dacă $Q_i \sim Q_j$, $Q_j \sim Q_k$ şi $Q_i \sim Q_k$
- totalitatea <u>claselor de compatibilităţi maxime</u> formează o acoperire a mulţimii stărilor automatului: $\{Q_1, Q_2, Q_3\}$, $\{Q_1, Q_3, Q_5\}$, $\{Q_2, Q_4, Q_6\}$ şi $\{Q_3, Q_5, Q_7\}$

- o clasă de compatibilități C_i a unei acoperiri implică existența altor clase de compatibilități C_j în cadrul acoperirii (stări succesoare care nu aparțin clasei C_i)

Compatibilități maxime (C_i)	(C_1) $\{Q_1, Q_2, Q_3\}$	(C_2) $\{Q_1, Q_3, Q_5\}$	(C_3) $\{Q_2, Q_4, Q_6\}$	(C_4) $\{Q_3, Q_5, Q_7\}$
Mulțimi implicate (I_{C_i})	$\{Q_2, Q_6\}$ $\{Q_4, Q_6\}$	$\{Q_1, Q_2, Q_3\}$	$\{Q_2, Q_3\}$	$\{Q_1, Q_2\}$ $\{Q_4, Q_6\}$

Fig. 8.8 *Clase de compatibilități maxime şi implicațiile lor*

- o clasă de compatibilități C_j dintr-o acoperire dată poate fi exclusă de o altă clasă C_i, dacă $C_j \subset C_i$, iar clasa exclusă implică mai multe clase de compatibilități, adică $I_{C_i} \subseteq I_{C_j}$
- clasele de compatibilități care nu sunt excluse de alte clase de compatibilități ale aceluiaşi automat incomplet specificat se numesc clase de compatibilități prime

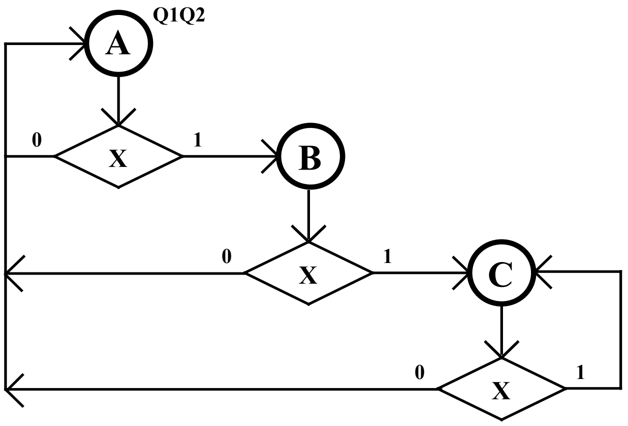
Compatibilități prime (C_j)	(C_5) $\{Q_1, Q_2\}$	(C_6) $\{Q_2, Q_3\}$	(C_7) $\{Q_1, Q_3\}$	(C_8) $\{Q_1, Q_5\}$	(C_9) $\{Q_3, Q_5\}$	(C_{10}) $\{Q_2, Q_6\}$	(C_{11}) $\{Q_3, Q_7\}$
Mulțimi implicate (I_{C_j})	$\{Q_2, Q_3\}$ $\{Q_4, Q_6\}$	$\{Q_2, Q_6\}$	$\{Q_2, Q_3\}$	$\{Q_1, Q_3\}$	$\{Q_1, Q_2\}$	Φ	$\{Q_4, Q_6\}$

Fig. 8.9 *Clase de compatibilități prime şi implicațiile lor*

- prin înlăturarea unei stări din clasa C_1 se formează clasele C_5, C_6 şi C_7; clasa C_2 formează clasele noi C_8 şi C_9; clasa C_3 formează numai clasa C_{10} (clasele $\{Q_2, Q_4\}$ şi $\{Q_4, Q_6\}$ sunt excluse de C_3), iar clasa C_4 formează numai clasa C_{11}
- acoperirea minimă are numai 3 elemente: clasa C_3 pentru că este singura care îl conține pe Q_4, C_1 care este implicată de C_3 şi C_4 care implică clasele C_1 şi C_3

8.2. Codificarea stărilor

- problema fundamentală în sinteza automatelor finite este codificarea <u>optimă</u> a stărilor



dacă A = 01 , B = 10 şi C = 11:

$Q_1^+ = x$ şi $Q_2^+ = \overline{x} + Q_1$

dacă A = 01 , B = 00 şi C = 11:

$Q_1^+ = x \cdot (\overline{Q}_0 + Q_1)$

$Q_2^+ = \overline{x} + Q_1 + \overline{Q}_2$

Fig. 8.10 *Două codificări posibile pentru un automat finit*

- o codificare aleatoare a stărilor poate oferi o soluție corectă (numai dacă x - sincronă), dar complexitatea CLC-ului ar putea fi mult prea mare
- o codificare binară minimală a celor m stări ale automatului redus minim se poate face folosind coduri cu r biți, conform relației $2^{r-1} < m \le 2^r$
- numărul de codificări distincte posibile este: $N_C = \dfrac{2^r!}{\left(2^r - m\right)!}$

starea prezentă	starea următoare / ieşire			
	$x_1x_2 = 00$	$x_1x_2 = 01$	$x_1x_2 = 11$	$x_1x_2 = 10$
S_1	$S_1 / 0$	$S_2 / 0$	$S_3 / 1$	$S_2 / 1$
S_2	$S_2 / 0$	$S_3 / 1$	$S_1 / 0$	$S_3 / 1$
S_3	$S_3 / 0$	$S_1 / 1$	$S_1 / 1$	$S_1 / 1$

Fig. 8.11 *Exemplu de tabel al tranzițiilor şi ieşirilor*

- se codifică cele 3 stări S_i ale automatului cu 2 cifre binare $a_i b_i$, unde $a_i, b_i \in \{0,1\}$, iar $i = 1,2,3$

starea prezentă $(Q_1 Q_2)$	starea următoare / ieşire $(Q_1^+ Q_2^+ / y)$			
	$x_1 x_2 = 00$	$x_1 x_2 = 01$	$x_1 x_2 = 11$	$x_1 x_2 = 10$
$a_1 b_1$	$a_1 b_1 / 0$	$a_2 b_2 / 0$	$a_3 b_3 / 1$	$a_2 b_2 / 1$
$a_2 b_2$	$a_2 b_2 / 0$	$a_3 b_3 / 1$	$a_1 b_1 / 0$	$a_3 b_3 / 1$
$a_3 b_3$	$a_3 b_3 / 0$	$a_1 b_1 / 1$	$a_1 b_1 / 1$	$a_1 b_1 / 1$

Fig. 8.12 *Tabelul binar al tranziţiilor şi ieşirilor*

- folosim notaţia simbolică : $Q_i^k = \begin{cases} \overline{Q_i} & \text{pentru } k = 0 \\ Q_i & \text{pentru } k = 1 \end{cases}$

- expresiile stărilor următoare ale variabilelor binare de stare:

$$Q_1^+ = a_1\left[Q_1^{a_1} Q_2^{b_1} \overline{x}_1 \overline{x}_2 + Q_1^{a_3} Q_2^{b_3} \overline{x}_1 x_2 + \left(Q_1^{a_2} Q_2^{b_2} + Q_1^{a_3} Q_2^{b_3}\right) x_1 x_2 + Q_1^{a_3} Q_2^{b_3} x_1 \overline{x}_2\right] +$$
$$a_2\left[Q_1^{a_2} Q_2^{b_2} \overline{x}_1 \overline{x}_2 + Q_1^{a_1} Q_2^{b_1} \overline{x}_1 x_2 + Q_1^{a_1} Q_2^{b_1} x_1 \overline{x}_2\right] + a_3\left[Q_1^{a_3} Q_2^{b_3} \overline{x}_1 \overline{x}_2 + Q_1^{a_2} Q_2^{b_2} \overline{x}_1 x_2 +\right.$$
$$\left. + Q_1^{a_1} Q_2^{b_1} x_1 x_2 + Q_1^{a_2} Q_2^{b_2} x_1 \overline{x}_2\right] = a_1 E_1 + a_2 E_2 + a_3 E_3$$
$$Q_2^+ = b_1 E_1 + b_2 E_2 + b_3 E_3$$

<u>**Regula 1**</u>**:** Stările prezente care au aceeaşi stare următoare, pentru aceleaşi intrări, primesc coduri adiacente.

- expresiile stărilor următoare se pot scrie sub forma:

$$Q_1^+ = Q_1^{a_1} Q_2^{b_1}\left(a_1 \overline{x}_1 \overline{x}_2 + a_2 \overline{x}_1 x_2 + a_2 x_1 \overline{x}_2 + a_3 x_1 x_2\right) + Q_1^{a_2} Q_2^{b_2}\left(a_1 x_1 x_2 + a_2 \overline{x}_1 \overline{x}_2 +\right.$$
$$\left. a_3 \overline{x}_1 x_2 + a_3 x_1 \overline{x}_2\right) + Q_1^{a_3} Q_2^{b_3}\left(a_1 \overline{x}_1 x_2 + a_1 x_1 x_2 + a_1 x_1 \overline{x}_2 + a_3 \overline{x}_1 \overline{x}_2\right)$$

<u>**Regula 2**</u>**:** Stările următoare unei stări prezente, care se află pe coloane cu intrări adiacente, primesc coduri adiacente.

- ieşirea y se scrie sub forma:

$$y = \overline{x}_1 x_2\left(Q_1^{a_2} Q_2^{b_2} + Q_1^{a_3} Q_2^{b_3}\right) + x_1 x_2\left(Q_1^{a_1} Q_2^{b_1} + Q_1^{a_3} Q_2^{b_3}\right) +$$
$$+ x_1 \overline{x}_2\left(Q_1^{a_1} Q_2^{b_1} + Q_1^{a_2} Q_2^{b_2} + Q_1^{a_3} Q_2^{b_3}\right)$$

Regula 3: Stările prezente, care au în corespondență ieşiri identice pentru aceleaşi intrări, primesc coduri adiacente.

- pentru simplificarea expresiilor stărilor următoare se recomandă ca stările care apar mai des ca stare următoare să primească coduri ce conțin mai multe zerouri

Regula 4: Codurile cu mai multe zerouri se atribuie stărilor care apar o singură dată pe mai multe coloane.

- regula 1 recomandă coduri adiacente pentru stările S_2 şi S_3, regula 2 recomandă coduri adiacente pentru S_2 şi S_3, S_1 şi S_3, S_1 şi S_2, regula 3 recomandă coduri adiacente pentru S_1 şi S_3, S_2 şi S_3, iar regula 4 recomandă codul 00 pentru starea S_3
- rezultă codurile $S_1 = 01$, $S_2 = 10$ şi $S_3 = 00$, dar condiția de adiacență pentru S_1 şi S_2 nu s-a putut respecta

starea prezentă $(Q_1 Q_2)$	starea următoare / ieşire $(Q_1^+ Q_2^+ / y)$			
	$x_1 x_2 = 00$	$x_1 x_2 = 01$	$x_1 x_2 = 11$	$x_1 x_2 = 10$
00	00 / 0	01 / 1	01 / 1	01 / 1
01	01 / 0	10 / 0	00 / 1	10 / 1
11	- / -	- / -	- / -	- / -
10	10 / 0	00 / 1	01 / 0	01 / 1

Fig. 8.13 *Tabelul tranzițiilor rezultat în urma codificării propuse*

- expresiile obținute cu aceste coduri sunt:

$$Q_1^+ = Q_1\overline{x}_1\overline{x}_2 + Q_2\overline{x}_1 x_2 + Q_2 x_1\overline{x}_2$$
$$Q_2^+ = Q_2\overline{x}_1\overline{x}_2 + \overline{Q}_1\overline{Q}_2\overline{x}_1 x_2 + \overline{Q}_1\overline{Q}_2 x_1\overline{x}_2 + \overline{Q}_2 x_1 x_2$$
$$y = \overline{Q}_2\overline{x}_1 x_2 + \overline{Q}_1 x_1 x_2 + x_1\overline{x}_2$$

Regula 5: Stările circuitului între care există tranziții primesc coduri adiacente.