

Automat de mancare

*Grigore Lucian-Florin
314CD
Facultatea de Automatica si Calculatoare
Universitatea Politehnica Bucuresti*

Cuprins

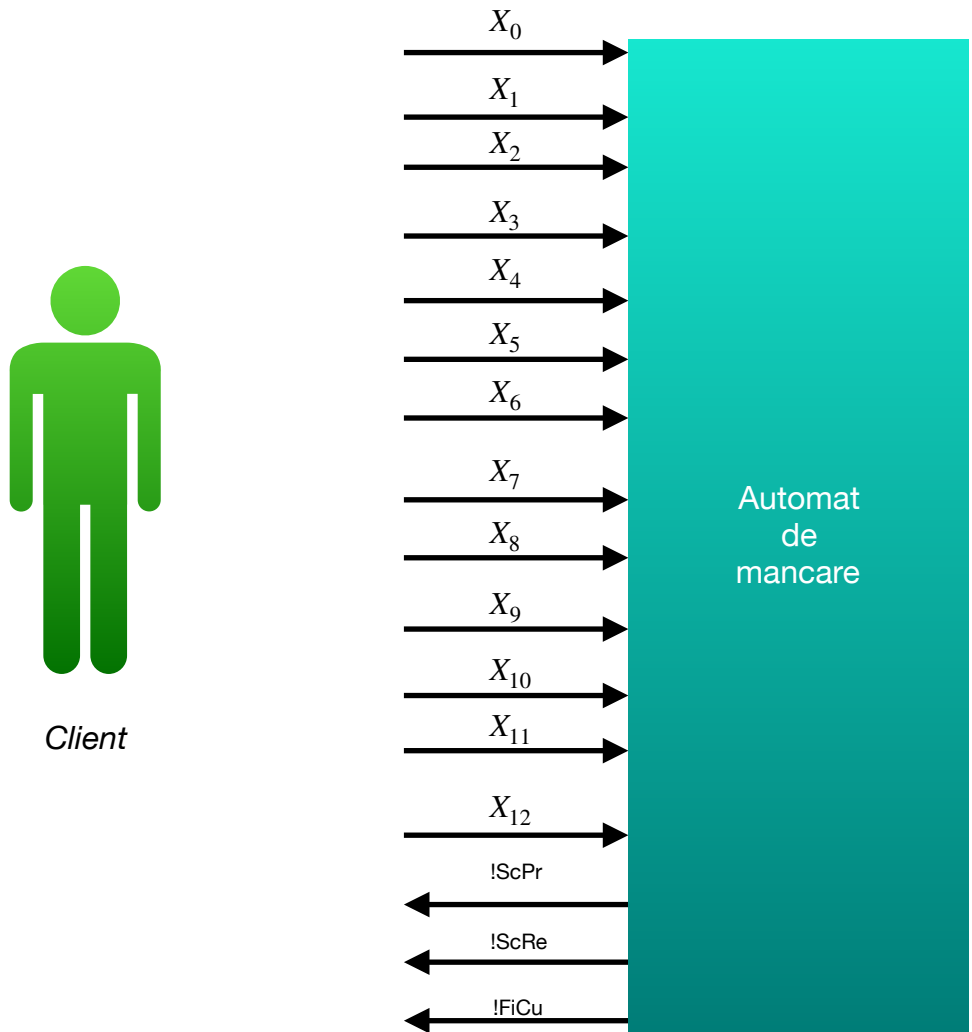
• Tema proiectului.....	3
• Descrierea modului de implementare.....	3
• Explicarea functionalitatii aparatului.....	5
• Organigrama aparatului.....	6
• Spatiul starilor.....	7
• Tabelul tranzitiilor.....	8
• Diagramele de stare urmatoare si ecuatiile rezultate.....	9
• Diagramele Karnaugh si ecuatiile rezultate pentru intrarile CBB-urilor.....	10
• Implementarea circuitului.....	14

Tema proiectului

Tema proiectului este reprezentata de proiectarea unui automat de mancare (asemenea celor care sunt in interiorul facultatii). Cei care doresc sa cumpere ceva pot plati ori cash, ori cu cardul.

Descrierea modului de implementare

Schema-bloc



Se fac verificari pentru anumite situatii precum:

- Un client selecteaza codul unui produs care nu este valid (nu se afla printre codurile folosite in aparat, sau produsul respectiv nu se mai gaseste in aparat)
- In cazul platii cash, banii introdusi pot sau nu sa cumpere produsul ales de client. Daca nu sunt suficienti, produsul nu este eliberat.
- In cazul platii cu cardul, se verifica daca plata a avut loc cu succes inainte de eliberarea produsului
- In ambele cazuri, este oferita optiunea de a continua cumparaturile dupa eliberarea unui sau mai multor produse
- In cazul platii cash, se poate elibera restul ramas, daca acesta exista la final

Automatul trece prin 13 stari:

S0 - 0000 - stare de repaus.

S1 - 0001 - pregatire mod de plata cash, afisarea unor instructiuni pe care trebuie sa le urmeaza clientul si asteptarea confirmarii pentru continuare.

S2 - 1000 - pregatire mod de plata cu cardul, afisarea unor instructiuni pe care trebuie sa le urmeze clientul si asteptarea confirmarii pentru continuare.

S3 - 1001 - asteptarea introducerii banilor.

S4 - 1100 - alegere produs (card).

S5 - 1101 - alegere produs (cash).

S6 - 1110 - confirmarea produsului ales (card) si pregatirea platii cu cardul.

S7 - 1111 - confirmarea produsului ales (cash).

S8 - 0110 - afisare mesaj de confirmare a produsului scos si intrebarea clientului daca doreste continuarea cumparaturilor (card).

S9 - 0011 - afisare mesaj de confirmare a produsului scos si intrebarea clientului daca doreste continuarea cumparaturilor (cash).

S10 - 0111 - finalizarea cumparaturilor (cash) si urmeaza verificarea daca exista vreun rest care poate fi scos de automat.

S11 - 0101 - se ajunge in starea asta doar daca exista un rest care trebuie scos. Se afiseaza un mesaj cu restul de bani scosi.

S12 - 0100 - se ajunge aici dupa terminarea cumparaturilor pentru ambele cazuri (cash sau card) si urmeaza revenirea la starea initiala (repaus).

Sunt folosite 13 variabile de intrare:

X0 - alegerea modului de plata (cash sau card) (cash - 0 , card - 1)

X1 - confirmarea pentru continuare cu modul de plata ales (cash) (continuare - 1, revenire la ecranul principal - 0)

X2 - confirmarea pentru continuare cu modul de plata ales (card) (continuare - 1, revenire la ecranul principal - 0)

X3 - confirmare pentru continuarea introducerii bani (1 - se continua introducerea banilor, 0 - se trece la pasul urmator, adica alegerea produsului dorit)

X4 - verifica daca codul produsului ales este valid (valid - 1, invalid - 0) (card)

X5 - verifica daca codul produsului ales este valid (valid - 1, invalid - 0) (cash)

X6 - verifica daca plata cu cardul a avut loc cu succes (1 - da, 0 - nu)

X7 - verifica daca banii introdusi sunt suficienti pentru produsul ales (1 - ajung, 0 - nu ajung)

X8 - intreaba clientul daca doreste continuarea cumparaturilor (1 - da, 0 - nu) (card)

X9 - intreaba clientul daca doreste continuarea cumparaturilor (1 - da, 0 - nu) (cash)

X10 - vede daca este vreun rest care poate fi inapoiat clientului (1 - exista, 0 - nu exista)

X11 - realizeaza trecerea catre ultima stare, dupa finalizarea cumparaturilor

X12 - realizeaza trecerea catre starea de repaus (initiala) a aparatului

Exista 3 iesiri:

FiCu - Finalizare Cumparaturi

ScPr - Scoatere Produs

ScRe - Scoatere Rest

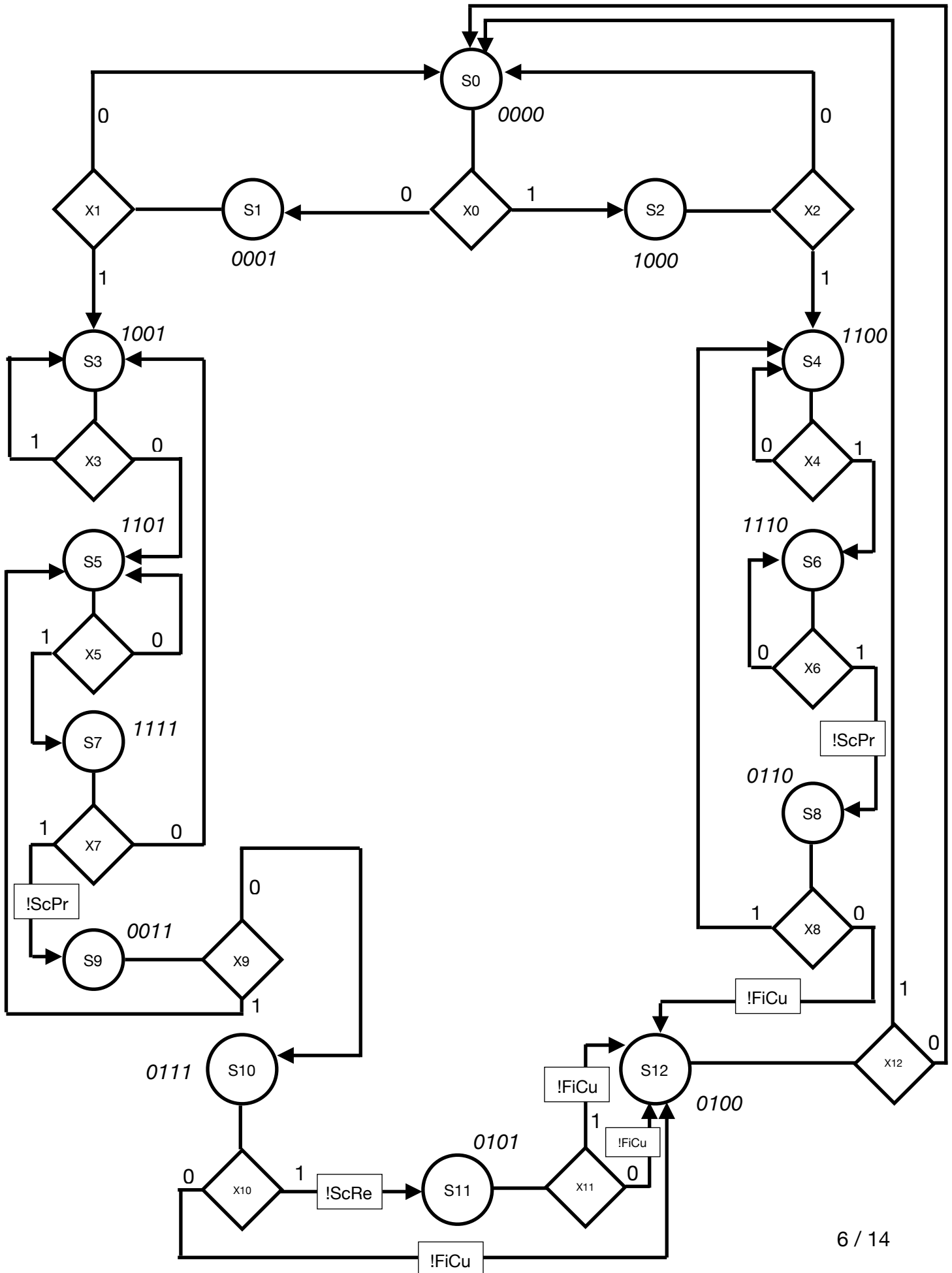
Explicarea functionalitatii aparatului

Starea initiala a aparatului este una de repaus. La inceput, clientul are optiunea de a alege dintre cele doua modalitati de plata: cash sau card. In ambele cazuri, este oferita mai apoi optiunea de a continua cumparaturile (daca metoda de plata este cea dorita de utilizator). Daca nu se doreste continuarea, aparatul se intoarce in starea initiala. In caz contrar, voi explica separat starile prin care trece aparatul pentru fiecare din aceste doua cazuri.

Cash: Aparatul ajunge in starea in care asteapta introducerea banilor. Se ofera clientului optiunea de a continua introducerea banilor, caz in care aparatul ramane in starea aceasta. Dupa terminarea introducerii banilor, aparatul ajunge in starea in care asteapta alegerea produsului de catre client, dupa care testeaza daca utilizatorul a ales un cod de produs valid. Daca nu este un cod valid, aparatul se intoarce in starea asta. Daca codul produsului este bun, aparatul trece in starea urmatoare in care afiseaza utilizatorului produsul pe care l-a ales si testeaza daca banii introdusi sunt suficienti pentru acesta. Daca nu, se revine la starea in care se introduc banii. Daca banii sunt suficienti, produsul este scos, si se ajunge intr-o stare in care se afiseaza un mesaj de confirmare a produsului scos. Se testeaza daca clientul doreste continuarea cumparaturilor. Daca da, atunci aparatul se intoarce in starea in care clientul trebuie sa aleaga un produs. Daca nu, se ajunge in starea care confirma finalizarea cumparaturilor pentru ramura "cash". Urmeaza sa se testeze daca exista un rest de bani ramas in aparat. Daca da, se scoate restul. Atat din starea de finalizare a cumparaturilor, cat si din cea care urmeaza scoaterii restului se ajunge in ultima stare a aparatului. Se finalizeaza cumparaturile si se revine in starea initiala a automatului.

Card: Aparatul ajunge in starea in care se asteapta alegerea unui produs. Daca codul produsului nu este valid, se ramane in starea asta. Daca este, se confirma produsul ales si se asteapta plata cu cardul. Se verifica daca plata a avut loc cu succes. Daca nu, aparatul ramane in starea asta. Daca a avut loc cu succes, produsul este scos, iar starea in care aparatul ajunge confirma produsul scos. Se ofera optiunea de a continua cumparaturile. Daca se doreste acest lucru, aparatul revine in starea in care clientul trebuie sa aleaga un produs. Daca nu, se finalizeaza cumparaturile si se ajunge in ultima stare (aceeasi cu cea in care se ajunge si din ramura "cash").

Organigrama



Spatiul starilor

$Q_1 Q_0$		$Q_3 Q_2$			
		00	01	11	10
$Q_3 Q_2$	00	REPAUS	INITCASH	PRODCHECKOUT	-
	01	ENDSHOP	CHANGE	ENDSHOPCASH	PRODCONF-CARD
	11	PRODSELCARD	PRODSELCASH	CONFPROD	INITPAYCARD
	10	INITCARD	INITPAYCASH	-	-

Tabelul tranzitiilor

Q_3^n	Q_2^n	Q_1^n	Q_0^n	Q_3^{n+1}	Q_2^{n+1}	Q_1^{n+1}	Q_0^{n+1}	NUME
0	0	0	0	X0	0	0	!X0	REPAUS
0	0	0	1	X1	0	0	!X1	INITCASH
0	0	1	0	-	-	-	-	-
0	0	1	1	X9	1	!X9	1	PRODCHECKOUT
0	1	0	0	0	0	0	0	ENDSHOP
0	1	0	1	0	1	0	0	CHANGE
0	1	1	0	X8	1	0	0	PRODCONF CARD
0	1	1	1	0	1	0	X10	ENDSHOPCASH
1	0	0	0	X2	X2	0	0	INITCARD
1	0	0	1	1	!X3	0	1	INITPAYCASH
1	0	1	0	-	-	-	-	-
1	0	1	1	-	-	-	-	-
1	1	0	0	1	1	X4	0	PRODSELCARD
1	1	0	1	1	1	X5	1	PRODSELCASH
1	1	1	0	!X6	1	1	0	INITPAYCARD
1	1	1	1	!X7	0	X7	1	CONFPROD

Diagramele de stare urmatoare si ecuatiile rezultate

$Q_3^n Q_2^n \backslash Q_1^n Q_0^n$	00	01	11	10
00	X0	X1	X9	-
01	0	0	0	X8
11	1	1	!X7	!X6
10	X2	1	-	-

Q_3^{n+1}

$$Q_3^{n+1} = (X_0)(!Q_3^n)(!Q_2^n)(!Q_0^n) + (X_2)(Q_3^n)(!Q_1^n) + (X_1)(!Q_2^n)(!Q_1^n)(Q_0^n) + (X_9)(!Q_2^n)(Q_1^n)(Q_0^n) + (!X_7)(Q_3^n)(Q_1^n)(Q_0^n) + (!X_6)(Q_3^n)(Q_1^n)(!Q_0^n) + (X_8)(!Q_3^n)(Q_1^n)(!Q_0^n) + (Q_3^n)(Q_2^n)(!Q_1^n) + (Q_3^n)(!Q_1^n)(Q_0^n)$$

$Q_3^n Q_2^n \backslash Q_1^n Q_0^n$	00	01	11	10
00	0	0	X9	-
01	0	1	1	1
11	1	1	0	1
10	X2	!X3	-	-

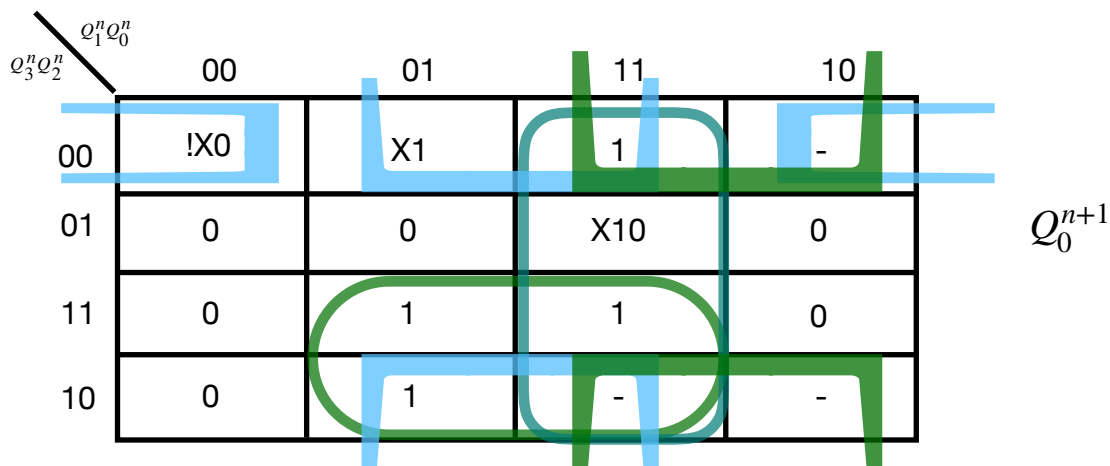
Q_2^{n+1}

$$Q_2^{n+1} = (X_2)(Q_3^n)(!Q_0^n) + (!X_3)(Q_3^n)(!Q_1^n)(Q_0^n) + (X_9)(!Q_3^n)(Q_1^n) + (!Q_3^n)(Q_2^n)(Q_0^n) + (Q_3^n)(Q_2^n)(!Q_1^n) + (Q_1^n)(!Q_0^n)$$

$Q_3^n Q_2^n \backslash Q_1^n Q_0^n$	00	01	11	10
00	0	0	!X9	-
01	0	0	0	0
11	X4	X5	X7	1
10	0	0	-	-

Q_1^{n+1}

$$Q_1^{n+1} = (X_5)(Q_3^n)(Q_2^n)(!Q_1^n)(Q_0^n) + (X_4)(Q_3^n)(Q_2^n)(!Q_0^n) + (X_7)(Q_3^n)(Q_1^n) + (!X_9)(!Q_2^n)(Q_1^n) + (Q_3^n)(Q_1^n)(!Q_0^n)$$

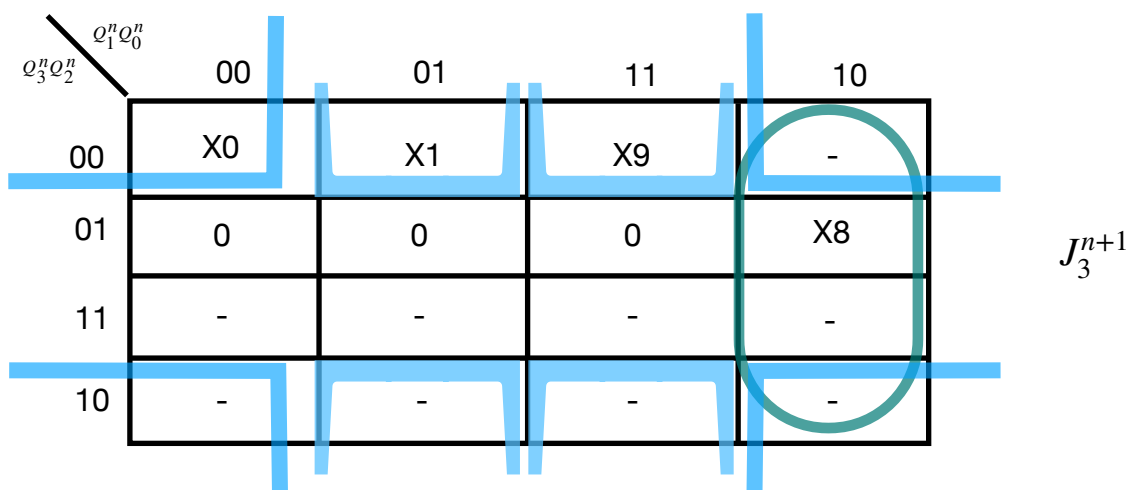


$$Q_0^{n+1} = (!X_0)(!Q_3^n)(!Q_2^n)(!Q_0^n) + (X_1)(!Q_2^n)(Q_0^n) + (X_{10})(Q_1^n)(Q_0^n) + (Q_3^n)(Q_0^n) + (!Q_2^n)(Q_1^n)$$

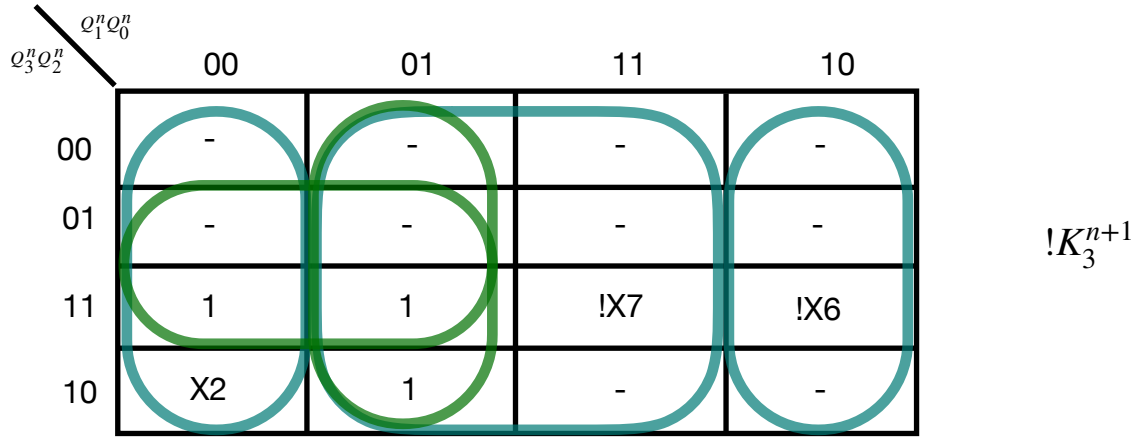
Diagramele Karnaugh si ecuatiile rezultate pentru intrarile CBB-urilor

Se vor face urmatoarele implementari:

- Q3 folosind CBB JK si porti
- Q2 folosind CBB JK, J fiind implementat cu MUX 4:1, iar K cu MUX 2:1
- Q1 folosind CBB D si MUX 8:1
- Q0 folosind CBB D si porti

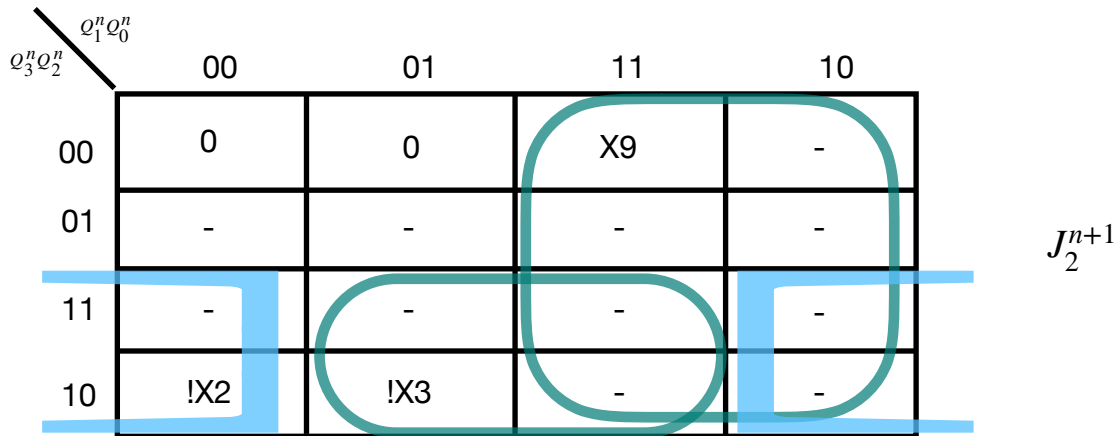


$$J_3^{n+1} = (X_8)(Q_1^n)(!Q_0^n) + (X_1)(!Q_2^n)(!Q_1^n)(Q_0^n) + (X_0)(!Q_2^n)(!Q_0^n) + (X_9)(!Q_2^n)(Q_1^n)(Q_0^n)$$

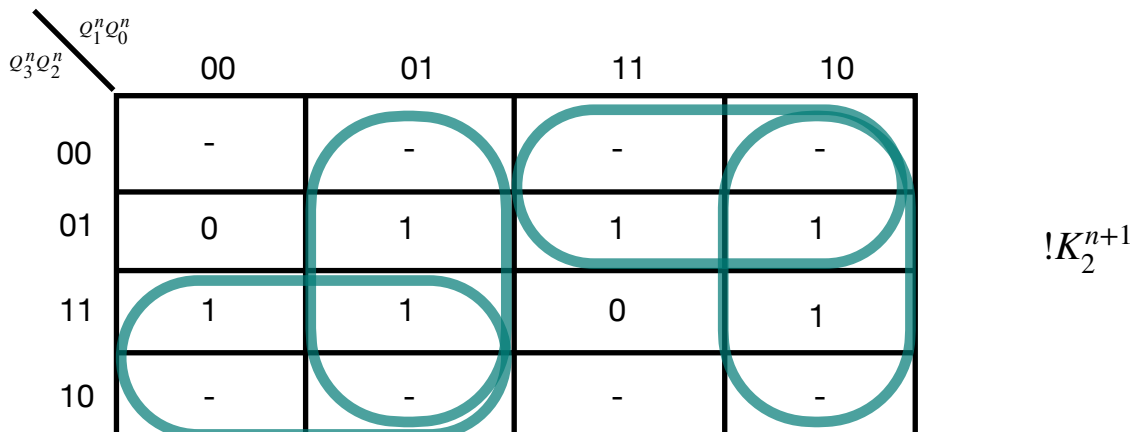
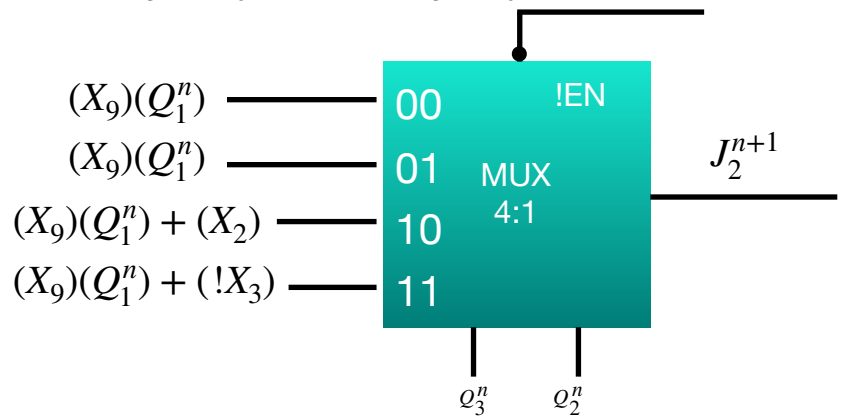


$$!K_3^{n+1} = (!X_7)(Q_0^n) + (X_2)(!Q_1^n)(!Q_0^n) + (!X_6)(Q_1^n)(!Q_0^n) + (!Q_1^n)(Q_0^n) + (Q_2^n)(!Q_1^n)$$

$$K_3^{n+1} = (X_7 + !Q_0^n)(!X_2 + Q_1^n + Q_0^n)(X_6 + !Q_1^n + Q_0^n)(Q_1^n + !Q_0^n)(!Q_2^n + Q_1^n)$$

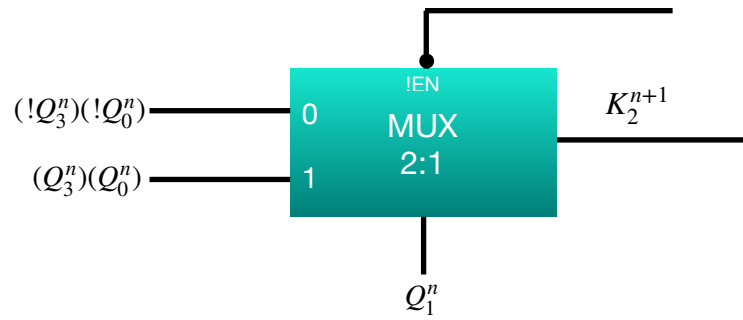


$$J_2^{n+1} = (X_9)(Q_1^n) + (X_2)(Q_3^n)(!Q_0^n) + (!X_3)(Q_3^n)(Q_0^n)$$



$$!K_2^{n+1} = (Q_3^n)(!Q_1^n) + (!Q_1^n)(Q_0^n) + (!Q_3^n)(Q_1^n) + (Q_1^n)(!Q_0^n)$$

$$K_2^{n+1} = (!Q_3^n + Q_1^n)(Q_1^n + !Q_0^n)(Q_3^n + !Q_1^n)(!Q_1^n + Q_0^n)$$



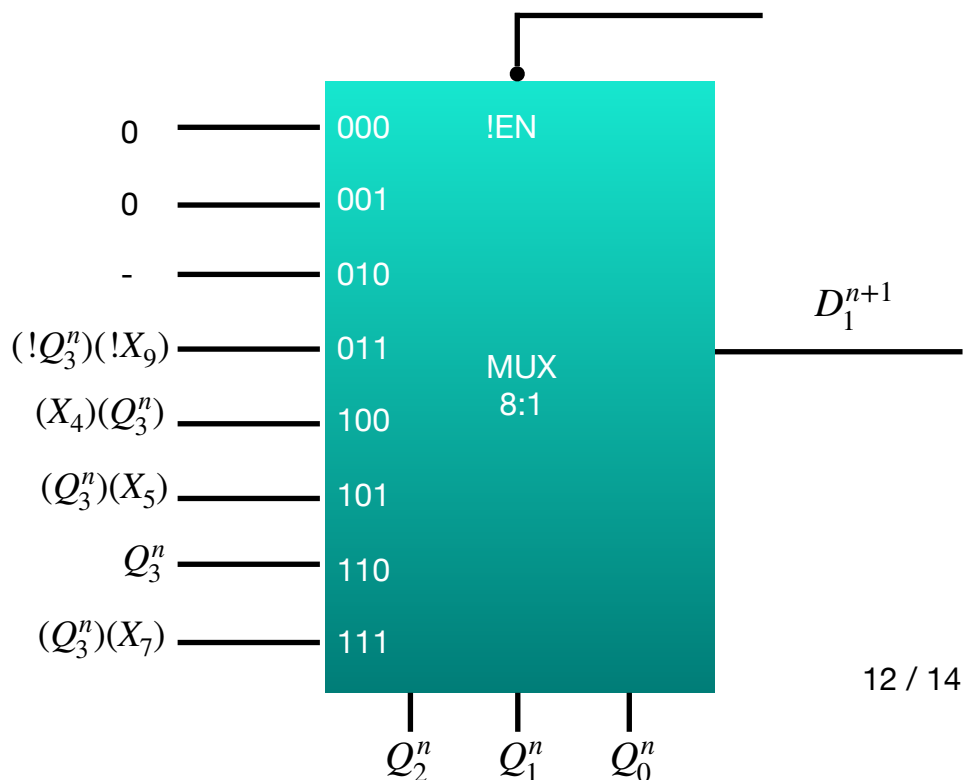
$Q_3^n Q_2^n$ \ $Q_1^n Q_0^n$	00	01	11	10
00	0	0	$!X_9$	-
01	0	0	0	0
11	X4	X5	X7	1
10	0	0	-	-

D_1^{n+1}

$$D_1^{n+1} = (Q_3^n)(Q_1^n)(!Q_0^n) + (!X_9)(!Q_3^n)(!Q_2^n)(Q_1^n) + (X_5)(Q_3^n)(Q_2^n)(!Q_1^n)(Q_0^n) + (X_4)(Q_3^n)(Q_2^n)(!Q_1^n)(!Q_0^n) + (X_7)(Q_3^n)(Q_2^n)(Q_1^n)(Q_0^n)$$

Q_2^n \ $Q_1^n Q_0^n$	00	01	11	10
0	0	0	$(!X_9)(!Q_3^n)$	-
1	$(X_4)(Q_3^n)$	$(X_5)(Q_3^n)$	$(X_7)(Q_3^n)$	Q_3^n

(am scris ecuatia dupa trecerea lui Q_3 in coloana rezultat)



$Q_3^n Q_2^n \backslash Q_1^n Q_0^n$		00	01	11	10
00	!X0	X1	1	-	
01	0	0	X10	0	
11	0	1	1	0	
10	0	1	-	-	

D_0^{n+1}

$$D_0^{n+1} = (X_{10})(Q_1^n)(Q_0^n) + (!X_0)(!Q_3^n)(!Q_2^n)(!Q_0^n) + (X_1)(!Q_2^n)(Q_0^n) + (Q_3^n)(Q_0^n) + (!Q_2^n)(Q_1^n)$$

Implementarea circuitului

