

corneliu hutaru

**circuite
logice
si
comenzi
secventiale**

junmed

Corneliu Huțanu
CIRCUITE LOGICE și COMENZI SECVENȚIALE

CORNELIU HUȚANU

CIRCUITE LOGICE
ȘI
COMENZI SECVENȚIALE

IAȘI • JUNIMEA • 1983

Coperta : MIHAI BUJDEI

CUPRINS

Partea I. BAZELE TEORETICE ALE CIRCUITELOR LOGICE

	Pag
1. Algebre booleene	7
1.1. Axiomele și proprietățile algebrelor booleene	7
2. Funcții booleene	11
2.1. Operații cu funcții booleene	13
2.2. Funcții booleene elementare	14
2.2.1. Superpoziția și decompoziția FB	18
2.3. Reprezentări ale funcțiilor booleene	19
2.3.1. Reprezentarea FB prin tabel de adevăr	20
2.3.2. Reprezentarea FB prin diagrame Karnaugh	20
2.3.3. Reprezentarea analitică a FB	22
2.3.4. Reprezentarea FB prin simbol de marcare	26
2.3.5. Reprezentarea FB prin scheme logice (logigrame)	30
2.3.6. Reprezentarea FB prin diagrame în timp	31
2.4. Sisteme complete de funcții	32
2.4.1. Implementarea FCD și FCC cu funcții Pierce și Sheffer	34
2.4.2. Alte sisteme complete de funcții	37
2.5. Clase de funcții booleene	38
2.5.1. Funcții degenerare și nedegenerate	38
2.5.2. Funcții duale și autoduale	39
2.5.3. Funcții cu prag	40
2.5.4. Funcții simetrice	41
2.5.5. Funcții monotone	42
2.5.6. Funcții liniare	42
3. Minimizarea funcțiilor booleene	43
3.1. Minimizarea FB folosind diagramele Karnaugh	50
3.2. Minimizarea FB prin metoda simbolică	55
3.3. Minimizarea FB date sub forma normală	60
3.4. Reprezentarea absolut minimală a FB	62

Partea II. CIRCUITE LOGICE

4. Circuite logice cu contacte de rele	65
5. Circuite logice cu dispozitive semiconductoare	67
5.1. Circuite logice cu diode semiconductoare	67
5.1.1. Circuit elementar cu diodă semiconductoare	68
5.1.2. Circuite logice cu diode, cu mai multe intrări	73
5.1.3. Circuite logice cu diode cu mai multe etaje	78
5.1.4. Proiectarea circuitelor logice cu diode	79
5.2. Circuite logice cu tranzistoare bipolare	85
5.2.1. Circuite logice RTL	87
5.2.1.1. Proiectarea circuitelor RTL	100
5.2.2. Circuite logice cu tranzistoare și diode (DTL)	103
5.2.2.1. Proiectarea circuitelor DTL	106
5.2.3. Circuite logice cu tranzistoare cu cuplaj direct	111
5.3. Circuite logice integrate cu tranzistoare bipolare	115
5.4. Circuite logice cu tranzistoare unipolare	119
5.4.1. Circuite logice statice cu TEC-MOS	123
5.4.2. Circuite logice dinamice cu TEC-MOS	125
5.4.3. Circuite logice cu TEC-MOS complementare	127
5.5. Circuite logice cu tranzistoare cu performanțe deosebite	129
5.6. Circuite logice cu dispozitive semiconductoare cu caracteristică non-monotonă	135
5.6.1. Circuite logice cu diode tunel	135
5.6.2. Circuite logice cu dispozitive semiconductoare cu caracteristică non-monotonă de tip S	142
6. Circuite logice cu dispozitive optoelectronice	143
6.1. Optronul de comutație	143
6.2. Realizarea circuitelor logice cu oprtoni	144
7. Circuite logice cu miezuri magnetice	145
7.1. Circuite logice cu miezuri și diode	148
7.1.1. Circuite logice cu scheme paralele	148
7.1.2. Circuite logice cu scheme serie	151
7.2. Circuite logice cu miezuri și tranzistoare	153

Partea III. COMENZI SECVENȚIALE

8. Noțiuni de teoria automatelor finite	157
8.1. Modele de automate	158
8.2. Reprezentări ale automatelor finite	161
9. Circuite logice combinaționale	164
9.1. Analiza circuitelor combinaționale	165
9.2. Sinteza circuitelor combinaționale	173
9.3. Hazardul în circuitele combinaționale	182
10. Circuite logice secvențiale de comandă	186
10.1. Analiza circuitelor secvențiale	195
10.2. Sinteza circuitelor secvențiale	220
10.2.1. Sinteza circuitelor secvențiale asincrone	222
10.2.1.2 Hazardul în circuitele secvențiale asincrone cu reacții directe	229
10.2.2. Sinteza circuitelor secvențiale sincrone	233
Bibliografie	238

Partea I

BAZELE TEORETICE ALE CIRCUITELOR LOGICE

1. Algebre booleene

Definirea riguroasă a problemelor tehnice privind circuitele logice și comenzile secvențiale se poate face folosind principiile logicii matematice, în particular principiile calculului propozițiilor.

Spre mijlocul secolului al XIX-lea, matematicianul și logicianul englez George Boole (1815—1864) a propus o interpretare a logicii propozițiilor bivalente, fundamentând algebra propozițiilor cu două valori care adesea este denumită algebra logicii sau algebra booleană. Apariția elementelor, circuitelor și sistemelor care în funcționare pot avea două stări distincte a condus la aplicarea în tehnică a perceptelor logicii bivalente. Printre lucrările de pionierat în acest domeniu se înscriu cele aparținând lui C. E. Shannon, V. I. Sestakov, M. Hanzawa, și nu în cele din urmă ale savantului român Gr. C. Moisil, care a avut o contribuție de prim rang și în aplicarea în tehnică a cercetărilor sale în țara noastră.

Algebrele booleene și în special algebra booleană cu două valori constituie fundamentul teoretic al circuitelor logice și comenzilor secvențiale.

1.1. Axiomele și proprietățile algebrelor booleene

Considerând cunoscute noțiunile elementare de teoria mulțimilor, în continuare se vor defini axiomele și proprietățile algebrelor booleene pornind de la structura algebrică, mai generală, de *lattice*.

Fie M o mulțime nevidă, împreună cu două operații binare pe M , denumite *reuniune* și *intersecție* și notate cu \cup și \cap .

Prin definiție, tripletul

$$L = (M, \cup, \cap)$$

este o *latice* și se bucură de următoarele proprietăți :

$$(1.1) \quad m_1 \cup m_2 = m_2 \cup m_1, \quad m_1 \cap m_2 = m_2 \cap m_1, \quad \forall m_1, m_2 \in M \text{ (comutativitatea),}$$

$$(1.2) \quad m_1 \cup (m_2 \cup m_3) = (m_1 \cup m_2) \cup m_3, \quad m_1 \cap (m_2 \cap m_3) = (m_1 \cap m_2) \cap m_3, \\ \forall m_1, m_2, m_3 \in M \text{ (asociativitatea),}$$

$$(1.3) \quad m_1 \cup (m_2 \cap m_1) = m_1, \quad m_1 \cap (m_2 \cup m_1) = m_1, \quad \forall m_1, m_2 \in M \text{ (absorbția).}$$

Proprietățile (1.1)...(1.3) constituie axiome pentru latici. Se poate observa că în acest sistem de axiome se pot schimba între ele simbolurile \cup și \cap . Evident, acest lucru se poate face în orice afirmație care decurge din sistemul de axiome, proprietate cunoscută sub denumirea de *principiul dualității pentru latici*.

Avînd în vedere proprietățile (1.1) și (1.2), operațiile reuniune și intersecție se pot extinde la orice număr arbitrar, dar finit, de termeni indiferent de ordinea termenilor sau factorilor :

$$\bigcup_{j=1}^n m_j = m_1 \cup m_2 \cup \dots \cup m_n, \quad \bigcap_{i=1}^n m_j = m_1 \cap m_2 \cap \dots \cap m_n$$

Plecînd de la axiomele definite mai sus se poate demonstra și următoarea proprietate :

$$(1.4) \quad m \cup m = m, \quad m \cap m = m, \quad \forall m \in M \text{ (idempotența)}$$

În adevăr, dacă se folosește proprietatea de absorbție se obține :

$$m \cap (m \cup m) = m, \text{ de unde } m \cup [m \cap (m \cup m)] = m \cup m,$$

dar, în același timp $m \cup [m \cap (m \cup m)] = m$. Rezultă $m \cup m = m$. Similar se demonstrează și $m \cap m = m$.

O *latice* se poate defini [5] și ca o mulțime parțial ordonată $L = (M, \leq)$, care are o cea mai mică margine superioară (c.m.m.m.s) — s și o cea mai mare margine inferioară (c.m.M.m.i.) — p pentru fiecare pereche de elemente. Legătura între cele două definiții se poate face notînd : $s = m_1 \cup m_2$, $p = m_1 \cap m_2$.

Prin definiție, o *latice finită (mărginită)* are un element care este c.m.M.m.i., numit ultim element al laticii, notat prin 0, astfel încît :

$$(1.5) \quad m \cup 0 = m, \quad m \cap 0 = 0, \quad \forall m \in M$$

și un element care este c.m.m.m.s, numit prim element al laticii, notat prin 1, astfel încît :

$$(1.6) \quad m \cup 1 = 1, \quad m \cap 1 = m, \quad \forall m \in M.$$

Fie $L = (M, \cup, \cap, 0, 1)$ o latice finită și $m \in M$. Un *element complementar* sau pe scurt un *complement* al elementului m este elementul \bar{m} (non m), astfel încît :

$$(1.7) \quad \begin{aligned} m \cup \bar{m} &= 1 \text{ (principiul terțului exclus),} \\ m \cap \bar{m} &= 0 \text{ (principiul contradicției).} \end{aligned}$$

Trebuie menționat că nu orice element dintr-o latice finită are un complement. Astfel, în laticea finită $L = (\{0, m, 1\}, \cup, \cap, 0, 1)$, elementul m nu are complement. De asemenea, complementul unui element al unei latices, dacă acesta există, nu este în mod necesar unic. În schimb elementele 0 și 1 au fiecare un complement unic, respectiv 1 și 0 : $\bar{0} = 1, \bar{1} = 0$.

Dacă într-o latice finită orice element m are un complement unic \bar{m} , această latice se numește *complementară*.

Prin definiție o latice L este *distributivă* dacă și numai dacă

$$(1.8a) \quad (m_1 \cup m_2) \cap m_3 = (m_1 \cap m_3) \cup (m_2 \cap m_3),$$

$$(1.8b) \quad (m_1 \cap m_2) \cup m_3 = (m_1 \cup m_3) \cap (m_2 \cup m_3), \quad \forall m_1, m_2, m_3 \in M.$$

Proprietatea (1.8a) poartă denumirea de distributivitatea reuniunii în raport cu intersecția iar (1.8b) distributivitatea intersecției în raport cu reuniunea.

Definiție. O *algebră booleană* este o latice distributivă și complementară. Din definiție rezultă că o algebră booleană este un 4-uplu

$$B = (M, \cup, \cap, \bar{\cdot}),$$

în care „ $\bar{\cdot}$ ” este operația unară de complementare.

Într-o algebră Boole se mai pot demonstra și următoarele proprietăți care au o deosebită importanță pentru studiul circuitelor de comutație :

$$(1.9) \quad \bar{\bar{m}} = m, \quad \forall m \in M \text{ (principiul involuției).}$$

Demonstrație. Se caută complementul lui \bar{m} . Conform cu (1.7) se poate scrie $\bar{m} \cup m = 1$, $\bar{m} \cap m = 0$. Atunci, conform definiției algebrelor Boole complementul lui \bar{m} este m . Deci $(\bar{\bar{m}}) = m$.

De asemenea, într-o algebră Boole sînt adevărate relațiile lui De Morgan :

$$(1.10a) \quad \overline{m_1 \cup m_2} = \bar{m}_1 \cap \bar{m}_2,$$

$$(1.10b) \quad \overline{m_1 \cap m_2} = \bar{m}_1 \cup \bar{m}_2, \quad \forall m_1, m_2 \in M.$$

Demonstrație. Se va demonstra că :

$$(m_1 \cup m_2) \cup (\bar{m}_1 \cap \bar{m}_2) = 1, \quad (m_1 \cup m_2) \cap (\bar{m}_1 \cap \bar{m}_2) = 0,$$

de unde, avînd în vedere unicitatea complementului într-o algebră Boole, se obține prima relație a lui De Morgan. Într-adevăr :

$$\begin{aligned} (m_1 \cup m_2) \cup (\bar{m}_1 \cap \bar{m}_2) &= [(m_1 \cup m_2) \cup \bar{m}_1] \cap [(m_1 \cup m_2) \cup \bar{m}_2] = \\ &= (m_2 \cup 1) \cap (m_1 \cup 1) = 1. \end{aligned}$$

$$\begin{aligned} (m_1 \cup m_2) \cap (\bar{m}_1 \cap \bar{m}_2) &= [m_1 \cap (\bar{m}_1 \cap \bar{m}_2)] \cup [m_2 \cap (\bar{m}_1 \cap \bar{m}_2)] = \\ &= (0 \cap m_2) \cup (0 \cap m_1) = 0. \end{aligned}$$

Similar, demonstrînd,

$$(m_1 \cap m_2) \cup (\bar{m}_1 \cup \bar{m}_2) = 1, \quad (m_1 \cap m_2) \cap (\bar{m}_1 \cup \bar{m}_2) = 0$$

se dovedește veridicitatea relației (1.10b).

Dacă într-o algebră Boole mulțimea M are numai elementele 0 și 1, se obține algebra Boole cu două elemente :

$$B_2 = (\{0, 1\}, \cup, \cap, \bar{})$$

în care operațiile sînt date în tabelele următoare :

\cup		0	1
0		0	1
1		1	1

\cap		0	1
0		0	0
1		0	1

$\bar{}$		0	1
		1	0

Din cele prezentate mai sus rezultă că în algebra Boole calculul este definit prin relațiile (1.1) ... (1.10). Pentru cele trei operații, în afara denumirilor menționate se mai folosesc și următoarele :

— operația SAU, disjuncția sau suma logică pentru reuniune, fiind notată și cu simbolurile „V” sau „+”. Astfel, următoarele notații sînt echivalente: $m_1 \cup m_2 = m_1 \vee m_2 = m_1 + m_2$. Pentru a nu da naștere la confuzii cu adunarea din algebra obișnuită cu care nu este identică se va utiliza în continuarea simbolul „ \cup ”.

— operația SI, conjuncția sau produs logic pentru intersecție, fiind notată și cu simbolurile „&” sau „.”. Ultimul simbol este cel mai folosit, în scriere omițîndu-se. Astfel, următoarele notații sînt echivalente: $m_1 \cap m_2 = m_1 \& m_2 = m_1 \cdot m_2 = m_1 m_2$,

— operația NU sau negația logică pentru complementare, notațiile următoare fiind echivalente: $m = m^c = m'$

Algebra booleană cu două elemente are aplicație directă în teoria circuitelor logice. În acest caz, între valorile mulțimii $\{0,1\}$ și cele două stări ale elementelor funcționînd în regim de comutație se stabilește o corespondență biunivocă. Astfel, o variabilă asociată unui element de comutare poate lua numai două valori, 0 sau 1, definind o variabilă bivalentă booleană sau, pe scurt, o *variabilă booleană*. Rezultă că pentru circuitele materializate cu elemente de comutație modelul matematic îl constituie funcțiile de variabile binare. Deoarece circuitele realizate cu elemente binare nu pot avea decît două stări distincte, funcțiile care descriu aceste circuite vor lua numai două valori. Aceste funcții bivalente de variabile binare se numesc *funcții booleene* sau *funcții logice* și au o deosebită importanță pentru studiul circuitelor logice și al comenzilor secvențiale.

2. Funcții booleene

Se consideră vectorul $X = (x_1, x_2, \dots, x_n)$ a cărui coordonate (x_1, x_2, \dots, x_n) pot lua valorile 0 sau 1. În acest caz rezultă că pot exista 2^n vectori X . Se notează mulțimea acestor vectori cu B_2^n . De asemenea, fiecărui vector din B_2^n i se poate atribui valorile 0 sau 1.

Definiție. Se numește *funcție booleană (FB)* funcția $f(X) = f(x_1, x_2, \dots, x_n)$, $X = (x_1, x_2, \dots, x_n)$ care aplică mulțimea B_2^n în mulțimea $\{0, 1\}$.

Fie K o submulțime a lui B_2^n și \bar{K} complementara lui K față de B_2^n : $K, \bar{K} \subset B_2^n$, $K \cup \bar{K} = B_2^n$, $K \cap \bar{K} = \emptyset$. Atunci o FB de n argumente $f(X) = f(x_1, x_2, \dots, x_n)$, $X = (x_1, x_2, x_3, \dots, x_n)$ se poate defini și astfel:

$$X \in K \rightarrow f(X) = 1, X \in \bar{K} \rightarrow f(X) = 0.$$

Deci, unei funcții booleene i se asociază un vector $V_f = (f(X))$ cu 2^n componente egale cu 0 sau 1, fiecare componentă fiind asociată unui vector X dat.

Consecință. Deoarece există 2^{2^n} vectori bivalenți cu 2^n componente, rezultă că numărul FB distincte de n argumente este finit și egal cu 2^{2^n} .

Să notăm valorile fixe ale coordonatelor unui vector din B_2^n prin $(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$. Aceste valori pot fi privite ca o combinație de valori ale argumentelor unei FB . Deoarece numărul acestor combinații este finit și egal cu 2^n atunci orice FB poate fi complet definită printr-un tabel finit cu 2^n rânduri. În acest tabel în partea stângă se trec combinațiile valorilor argumentelor iar în partea dreaptă valorile corespunzătoare, 0 sau 1, ale funcției :

x_1	x_2	\dots	x_{n-1}	x_n	$f(x_1, x_2, \dots, x_n)$
0	0	\dots	0	0	α_1
0	0	\dots	0	1	α_2
0	0	\dots	1	0	α_3
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
1	1	\dots	1	1	α_{2^n}

cu $\alpha_i \in \{0, 1\}$.

Există situații când pentru unele combinații ale valorilor argumentelor o FB să nu aibă valoarea determinată. Astfel de funcții nedeterminate pentru una sau mai multe combinații ale valorilor argumentelor, se numesc *FB incomplet definite*. În mod obișnuit în tabelele de definiție ale funcțiilor valorilor nedeterminate sînt indicate cu asterisc (*) sau \emptyset .

Exemplu : Fie $f(x_1, x_2, x_3)$ dată prin tabelul următor :

x_1	x_2	x_3	$f(x_1, x_2, x_3)$	x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0	1	0	0	0
0	0	1	1	1	0	1	*
0	1	0	*	1	1	0	*
0	1	1	1	1	1	1	1

Funcția este nedeterminată pentru combinațiile $(0, 1, 0)$, $(1, 0, 1)$ și $(1, 1, 0)$ ale valorilor argumentelor, ele putând fi aplicate arbitrar în $\{0, 1\}$. Atribuind funcției valorile 0 sau 1 pentru combinațiile respective ale valorilor argumentelor se pot obține 8 funcții complet definite.

În general, dacă o FB nu este definită pentru r combinații ale valorilor argumentelor atunci prin definire arbitrară se pot obține 2^r funcții noi complet definite. Funcțiile incomplet definite se întâlnesc frecvent în practica comenzilor secvențiale, evidențierea situațiilor de nedefinire și atribuire voită a valorilor 0 sau 1 fiind foarte importantă pentru simplificarea lor.

2.1. Operații cu funcții booleene

Operațiile cu FB se definesc pe domeniul valorilor funcțiilor. Se consideră două funcții $f_1(x_1, x_2, \dots, x_n)$ și $f_2(x_1, x_2, \dots, x_n)$. Se spune că aceste funcții sînt *identice* dacă iau valori identice pentru toate combinațiile posibile ale valorilor argumentelor. În mod obișnuit identitatea a două FB se scrie astfel :

$$(2.1) \quad f_1(x_1, x_2, \dots, x_n) = f_2(x_1, x_2, \dots, x_n).$$

Dacă pentru cel puțin o singură combinație a valorilor argumentelor (un n -uplu) cele două funcții nu au aceeași valoare atunci :

$$f_1(x_1, x_2, \dots, x_n) \neq f_2(x_1, x_2, \dots, x_n).$$

Fie f_1 , f_2 și f_3 funcții booleene de n argumente. Operațiile \cup , \cap și $-$ cu funcții se definesc în modul următor :

a) reuniunea funcțiilor

(2.2) $f_1(x_1, x_2, \dots, x_n) \cup f_2(x_1, x_2, \dots, x_n) = f_3(x_1, x_2, \dots, x_n)$, dacă și numai dacă valorile funcțiilor se combină corespunzător tabelului operației \cup pentru fiecare combinație a valorilor argumentelor ;

b) intersecția

(2.3) $f_1(x_1, x_2, \dots, x_n) \cap f_2(x_1, x_2, \dots, x_n) = f_3(x_1, x_2, \dots, x_n)$, dacă și numai dacă valorile funcțiilor se combină conform tabelului operației \cap pentru fiecare n -uplu al argumentelor ;

c) negarea

(2.4) $\bar{f}_1(x_1, x_2, \dots, x_n) = f_2(x_1, x_2, \dots, x_n)$, dacă și numai dacă valorile funcțiilor respectă tabelul de definiție al operației de complementare.

Din cele de mai sus rezultă că pentru operarea cu FB se consideră succesiv valorile funcțiilor corespunzător celor 2^n combinații ale argumentelor. Funcțiile f_1 și f_2 aplică fiecare din cele 2^n n -uple în mulțimea $\{0, 1\}$. Se obțin astfel 2^n perechi de valori ale funcțiilor. Operațiile binare SAU și SI asupra celor două funcții aplică cele 2^n perechi în mulțimea $\{0, 1\}$. Operația unară de complementare aplică cele 2^n n -uple ale unei funcții în $\{0, 1\}$. Pentru operarea cu FB este avantajoasă folosirea tabelelor.

Exemplu. Se dau tabelar următoarele funcții de două argumente

x_1	x_2	$f_1(x_1, x_2)$	$f_2(x_1, x_2)$	$f_1 \cup f_2$	$f_1 \cap f_2$	\bar{f}_1	\bar{f}_2
0	0	0	1	1	0	1	0
0	1	0	0	0	0	1	1
1	0	1	1	1	1	0	0
1	1	1	0	1	0	0	1

Funcția f_1 aplică pe $(0, 0)$ și $(0, 1)$ în 0 și pe $(1, 0)$ și $(1, 1)$ în 1, iar f_2 aplică pe $(0, 1)$ și $(1, 1)$ în 0 și pe $(0, 0)$ și $(1, 0)$ în 1. Operația de reuniune asupra celor două funcții aplică perechile valorilor funcțiilor $(0, 0)$ în 0 iar $(0, 1)$, $(1, 1)$ și $(1, 0)$ în 1. De asemenea, $f_1 \cap f_2$ aplică perechile $(0, 1)$ și $(0, 0)$ în 0 și $(1, 1)$ și $(1, 0)$ în 1. Operația de complementare a celor două funcții este evidentă.

2.2. Funcții booleene elementare

În acest paragraf se vor defini funcțiile booleene fundamentale, cu ajutorul cărora se pot construi funcții mai complexe. Aceste FB denumite și funcții elementare au o deosebită importanță practică pentru realizarea circuitelor logice modulare. Obișnuit, funcțiile elementare se definesc pe mulțimea funcțiilor de două argumente. Cele $2^{2^2} = 16$ funcții booleene de două argumente sînt prezentate în tabelul următor:

x_1	x_2	f_0	f_1	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}	f_{11}	f_{12}	f_{13}	f_{14}	f_{15}
0	0	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
0	1	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	0	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Din examinarea tabelului rezultă și se definesc următoarele funcții elementare :

— funcțiile f_0 și f_{15} sînt funcții constante, nedepinzînd de argumentele x_1 și x_2 . Sînt denumite și funcțiile logice *constanta 0* și respectiv *constanta 1* :

$$f_0(x_1, x_2) = 0, f_{15}(x_1, x_2) = 1$$

— funcțiile f_{10} și f_{12} corespund valorilor argumentelor :

$$f_{10}(x_1, x_2) = x_2, f_{12}(x_1, x_2) = x_1$$

și se mai numesc funcții *identitate* :

— funcțiile f_3 și f_5 corespund funcțiilor f_{10} și f_{12} negate :

$$f_3(x_1, x_2) = \bar{f}_{12}(x_1, x_2) = \bar{x}_1, f_5(x_1, x_2) = \bar{f}_{10}(x_1, x_2) = \bar{x}_2$$

și se numesc *funcțiile negație*.

Remarcă. Funcțiile definite mai sus depind numai de unul din argumente sau de nici unul. Se spune că aceste funcții elementare sînt *funcții degenerate* de două argumente.

— funcția f_8 corespunde produsului logic al funcțiilor f_{10} și f_{12} și se numește *funcția conjuncție* sau *funcția SI* :

$$f_8(x_1, x_2) = f_{10}(x_1, x_2) \cdot f_{12}(x_1, x_2) = x_1 x_2$$

— funcția f_{14} corespunde sumei logice a funcțiilor f_{10} și f_{12} , numindu-se *funcția disjuncție* sau *funcția SAU* :

$$f_{14}(x_1, x_2) = f_{10}(x_1, x_2) \cup f_{12}(x_1, x_2) = x_1 \cup x_2$$

— funcția f_1 poartă denumirea de *funcția lui Pierce* sau *funcția lui Webb* și se notează în modul următor :

$$f_1(x_1, x_2) = x_1 \downarrow x_2$$

Examinînd valorile acestei funcții se observă că ea este negarea funcției disjuncție :

$$f_1(x_1, x_2) = \bar{f}_{14}(x_1, x_2) = \overline{x_1 \cup x_2} = \bar{x}_1 \cdot \bar{x}_2$$

Avînd în vedere cele de mai sus se mai poate scrie :

$$f_1(x_1, x_2) = f_3(x_1, x_2) \cdot f_5(x_1, x_2) = f_{10}(x_1, x_2) \downarrow f_{12}(x_1, x_2)$$

Din relația precedentă se observă că simbolul „ \downarrow ” are rol de operator, definind situația de „nici x_1 și nici x_2 ”. Din acest motiv funcția f_1 se mai numește și *funcția NICI*. În literatură este întîlnită și sub denumirea de *funcția NOR*, provenită din limba engleză (NOT OR=NOR).

— funcția f_7 se numește *funcția lui Sheffer* și se simbolizează astfel :

$$f_7(x_1, x_2) = x_1 \uparrow x_2 = x_1/x_2$$

Examinînd în tabel valorile acestei funcții, se poate observa că ea reprezintă negarea funcției conjuncție de unde-i provine și denumirea de *funcție SI—NU* sau *funcție NAND* (în l. engleză NOT AND=NAND) :

$$f_7(x_1, x_2) = \bar{f}_8(x_1, x_2) = \overline{x_1 x_2} = \bar{x}_1 \cup \bar{x}_2 = f_{10}(x_1, x_2) \uparrow f_{12}(x_1, x_2)$$

Corespunzător relațiilor de mai sus se mai poate spune că funcția lui Sheffer definește situația de „numai x_1 sau numai x_2 ”, fapt ce a condus și la denumirea de *funcție NUMAI*.

Funcțiile NICI și NUMAI au o deosebită importanță atît pentru teoria funcțiilor booleene cît și pentru aplicarea practică a acestora în realizarea circuitelor logice modulare.

— Funcția f_9 are de asemenea importanță pentru teoria *FB*, fiind denumită funcția echivalență :

$$f_9(x_1, x_2) = x_1 \sim x_2,$$

desemnînd „echivalența între x_1 și x_2 ”. Se poate demonstra că această funcție poate fi descrisă prin funcțiile conjuncție, disjuncție și negație:

$$f_9(x_1, x_2) = x_1 \sim x_2 = (\bar{x}_1 \cup x_2) (x_1 \cup \bar{x}_2)$$

În adevăr, folosind tabelele, pentru cele două părți ale identității se obține :

x_1	x_2	$x_1 \sim x_2$	\bar{x}_1	\bar{x}_2	$\bar{x}_1 \cup x_2$	$x_1 \cup \bar{x}_2$	$(\bar{x}_1 \cup x_2)(x_1 \cup \bar{x}_2)$
0	0	1	1	1	1	1	1
0	1	0	1	0	1	0	0
1	0	0	0	1	0	1	0
1	1	1	0	0	1	1	1

Întrucît pentru toate combinațiile posibile ale valorilor argumentelor cele două părți ale identității au aceleași valori demonstrația este făcută.

— funcția f_6 se numește *funcția suma modulo 2* sau *funcția SAU—EXCLUSIV* și se notează astfel:

$$f_6(x_1, x_2) = x_1 \oplus x_2$$

Comparînd în tabelul de definiție valorile acestei funcții cu ale funcției f_9 , rezultă:

$$f_6(x_1, x_2) = \bar{f}_9(x_1, x_2) = \overline{x_2 \sim x_2} = \overline{(\bar{x}_1 \cup x_2)(x_1 \cup \bar{x}_2)} = x_1 x_2 \cup x_1 x_2$$

— funcțiile f_{11} și f_{13} sînt denumite *funcțiile implicație*:

$$f_{11}(x_1, x_2) = x_1 \rightarrow x_2, \text{ implicația lui } x_1 \text{ în } x_2.$$

$$f_{13}(x_1, x_2) = x_1 \leftarrow x_2, \text{ implicația lui } x_2 \text{ în } x_1 \text{ sau implicația inversă.}$$

Folosind reprezentarea tabelară se poate arăta ușor că:

$$f_{11}(x_1, x_2) = x_1 \rightarrow x_2 = \bar{x}_1 \cup x_2, \quad f_{13}(x_1, x_2) = x_1 \leftarrow x_2 = x_1 \cup \bar{x}_2$$

— funcțiile f_2 și f_4 sînt denumite *funcțiile interdicție* sau *funcțiile inhibare*:

$$f_2(x_1, x_2) = x_1 \neq x_2 = \bar{x}_1 x_2, \quad x_1 \text{ inhibă } x_2$$

$$f_4(x_1, x_2) = x_2 \neq x_1 = x_1 \bar{x}_2, \quad x_2 \text{ inhibă } x_1 \text{ sau inhibare inversă.}$$

Din definirea funcțiilor booleene elementare de două argumente se poate desprinde concluzia că se pot genera *FB* noi prin:

- 1) permutarea argumentelor,
- 2) introducerea funcțiilor în locul argumentelor noilor funcții.

2.2.1. Superpoziția și decompoziția FB

Se consideră mulțimea variabilelor boolcene $\mathbf{X} = \{x_1, x_2, \dots, \dots, x_n\}$ cu $n > 2$. Această mulțime poate fi divizată în două submulțimi, spre exemplu: $\mathbf{X}_1 = \{x_1, x_2, \dots, x_r\}$, unde $n > r$, și $\mathbf{X}_2 = \{x_{r+1}, x_{r+2}, \dots, x_n\}$. Cu cele două submulțimi se pot forma două FB oarecare $f_1(\mathbf{X}_1)$ și $f_2(\mathbf{X}_2)$. Dacă cu aceste funcții se construiește o nouă funcție

$$f(\mathbf{X}) = f_3 \{f_1(\mathbf{X}_1), f_2(\mathbf{X}_2)\}.$$

se spune că aceasta s-a obținut prin *superpoziția* sau *compunerea* funcțiilor f_1 și f_2 .

Dacă submulțimile \mathbf{X}_1 și \mathbf{X}_2 sînt disjuncte superpoziția se numește *disjunctă* iar dacă cele două submulțimi nu sînt disjuncte, adică $\mathbf{X}_1, \mathbf{X}_2 \neq 0$, atunci superpoziția se va numi *nedisjunctă*.

Exemplu. Funcțiile lui Pierce și Sheffer sînt superpoziții ale funcțiilor negație, disjuncție și conjuncție.

În practică, o mare importanță o are problema inversă, și anume, funcția $f(\mathbf{X})$ să fie realizată (implementată) cu un anumit set de funcții elementare (functori). Aceasta constituie problema decompoziției funcțiilor booleene.

Dacă o FB de n argumente poate fi reprezentată sub forma:

$$(2.5) \quad f(\mathbf{X}) = \varphi_m \{ \varphi_{m-1} [\mathbf{X}^{n-1}], \dots, \varphi_1 [x^1], \mathbf{X}^0 \},$$

unde funcțiile φ_i ($i=0, 1, \dots, m-1$) depind de un număr mai mic de argumente ($\mathbf{X}^i \subset \mathbf{X}$), se obține *decompoziția funcțională* a funcției considerate.

Cînd mulțimea \mathbf{X}^i a argumentelor de care depinde φ_i și mulțimea \mathbf{X}^0 nu au elemente comune, atunci decompoziția se numește *disjunctă* sau *neintersectabilă*. Dacă există un singur argument x , care să aparțină mai multora din mulțimile \mathbf{X}^i , decompoziția se numește *nedisjunctivă* sau *intersectabilă*.

Dacă funcția $f(\mathbf{X})$ se poate reprezenta sub forma

$$(2.6) \quad f(\mathbf{X}) = \varphi_2 [\varphi_1(\mathbf{X}^1), \mathbf{X}^0],$$

atunci decompoziția funcțională se numește *simplă*.

Un caz particular de decompoziție funcțională îl constituie *formulele de dezvoltare ale lui Shannon*. Conform acestora, pentru o funcție de o singură variabilă $y = f(x)$, sînt valabile următoarele dezvoltări:

$$y = \bar{x}f(0) \cup xf(1), \text{ sau } y = [\bar{x} \cup f(1)] [x \cup f(0)].$$

în care $f(0)$ este valoarea funcției obținută prin substituirea lui $x=0$, iar $f(1)$ corespunde valorii $x=1$.

Avînd în vedere cele de mai sus, o FB de n argumente poate fi descompusă după argumentul x_i :

$$f(X) = \bar{x}_i f(x_1, x_2, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \cup x_i f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n),$$

care mai poate fi scrisă sub forma

$$f(X) = \varphi_3 \{ \varphi_2(X^2), \varphi_1(X^1), X^0 \},$$

unde

$$\varphi_1(X^1) = f(x_1, x_2, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n),$$

$$\varphi_2(X^2) = f(x_1, x_2, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n),$$

$$\varphi_3(X) = \bar{x}_i \varphi_1(X^1) \cup x_i \varphi_2(X^2),$$

$$X^1 = \{x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_n\},$$

$$X^2 = \{x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_n\},$$

$$X^0 = \{x_i\}.$$

Notînd X^1 și X^2 prin X^a și X^0 prin X^b , cardinalul acestor mulțimi va fi $|X^a| = n-1$ respectiv $|X^b| = 1$. Generalizînd decompoziția pentru cazul

$$2 \leq |X^a| = n - |X^b| \leq n-1, \quad X^a \cap X^b = \emptyset, \quad X^a \cup X^b = X,$$

conform cu (2.6) se obține o decompoziție simplă disjunctă, diferită de decompoziția banală corespunzătoare dezvoltării după formulele lui Shannon.

Shannon a fost cel care a pus bazele decompoziției FB, extinse apoi de Ashenurst și Curtis. Decompoziția FB este foarte actuală, avînd în vedere dezvoltarea puternică a circuitelor logice modulare integrate.

2.3. Reprezentări ale funcțiilor booleene

Studiul FB se face în multe cazuri pe reprezentările acestora. Există o mare diversitate de reprezentări ale FB care pot fi grupate în reprezentări grafice (geometrice) și analitice. Reprezentările din prima categorie sînt intuitive și se folosesc pentru studiul FB cu un număr redus de argumente. Din această categorie fac parte reprezentările prin tabel de adevăr, diagrame Euler, Venn, Veitch sau Karnaugh, prin grafuri sau pe hipercub. A

doua categorii asigură o reprezentare prin expresii algebrice sau sub formă de coduri. Reprezentările din această categorie permit studiul FB cu un număr arbitrar de argumente, cu posibilitatea utilizării mijloacelor numerice de calcul. În continuare se va insista asupra celor mai folosite reprezentări ale FB în scopuri tehnice.

2.3.1. Reprezentarea FB prin tabel de adevăr

Acest mod de reprezentare corespunde reprezentării tabelare a FB și de care s-a uzitat pînă aici. Tabelele denumite *de adevăr* sau *combinatoriale* conțin în partea stîngă un număr de linii egal cu numărul combinațiilor posibile ale valorilor argumentelor, iar în partea dreaptă valorile funcției pentru fiecare combinație de valori ale argumentelor. Tabelul de adevăr este cea mai completă reprezentare a unei FB deoarece pentru fiecare combinație posibilă a valorilor argumentelor se indică valoarea funcției.

2.3.2. Reprezentarea FB prin diagrame Karnaugh

Diagrama Karnaugh este tot o reprezentare tabelară dar în raport cu tabelul de adevăr este mai compactă datorită dispunerii bidirecționale a valorilor argumentelor.

În cazul general, al unei FB de n argumente, diagrama Karnaugh conține 2^p linii și 2^q coloane, astfel ca $p+q=n$. Dacă n este par, în mod obișnuit $p=q$, iar dacă n este impar, $q=p+1$ (sau $p=q+1$). Rezultă o diagramă cu $2^p \cdot 2^q = 2^n$ cîmpuri în care se trec valorile funcției pentru combinațiile corespunzătoare ale valorilor argumentelor. Valorile argumentelor se indică la capetele liniilor și coloanelor diagramei.

Veitch a fost cel care a introdus acest mod de reprezentare a FB . În reprezentarea propusă de acesta combinațiile valorilor argumentelor pe linii și coloane se dispun conform codului binar natural. Această dispunere conduce la dificultăți în folosirea diagramei pentru simplificarea FB (pentru care a și fost concepută). Puțin mai târziu Karnaugh propune construirea acestor diagrame folosind codul Gray (binar reflectat) care fiind un cod continuu și ciclic asigură *adiacenta* între cîmpurile diagramei. Din motivele arătate această reprezentare mai este cunoscută și ca diagrama Veitch-Karnaugh.

Notă. Într-un sistem de numerație de bază B , două cifre sînt adiacente dacă diferă cu o unitate modulo B . În sistemul de numerație binar două cifre vor fi adiacente dacă diferă cu cifra 1.

Exemplu. Se consideră FB de 4 argumente dată prin tabel de adevăr. Să se reprezinte prin diagramă Karnaugh.

x_1	x_2	x_3	x_4	$f(x_1, x_2, x_3, x_4)$	x_1	x_2	x_3	x_4	$f(x_1, x_2, x_3, x_4)$
0	0	0	0	0	1	0	0	0	1
0	0	0	1	1	1	0	0	1	0
0	0	1	0	1	1	0	1	0	1
0	0	1	1	0	1	0	1	1	1
0	1	0	0	0	1	1	0	0	1
0	1	0	1	1	1	1	0	1	0
0	1	1	0	0	1	1	1	0	0
0	1	1	1	0	1	1	1	1	1

Diagrama Karnaugh va conține 2^4 cîmpuri realizate cu $2^2=2^2$ linii și $2^2=2^2$ coloane (fig. 2.1). Combinațiile valorilor argumentelor x_1 și x_2 sînt dispuse în partea superioară a diagramei iar ale argumentelor x_3 și x_4 vertical, în partea stîngă. La intersecția unei coloane și a unei linii este cîmpul diagramei în care se trece valoarea funcției corespunzător combinației valorilor argumentelor. Astfel, pentru $x_1=0$, $x_2=1$ și $x_3=1$, $x_4=0$ la intersecția coloanei a doua (01) cu rîndul al 4-lea (10) se indică valoarea funcției $f(x_1, x_2, x_3, x_4)=0$, ș.a.

Dacă pentru un număr mic de argumente combinațiile valorilor argumentelor conform codului Gray sînt ușor de stabilit, pentru un număr mai mare de $5(n > 5)$ devine dificil și se poate

$x_1 \backslash x_2$	x_3	x_4	00	01	11	10
00	0	0	0	0	1	1
01	1	1	1	0	0	0
11	0	0	0	0	1	1
10	1	0	0	0	0	1

Fig. 2.1

greși construcția diagramei. Folosind însă o reprezentare grafică a codului Gray realizarea diagramelor Karnaugh devine simplă și sigură. În această reprezentare biților 1 li se atașează un segment de dreaptă cu lungime corespunzătoare coloanei sau liniei pentru care argumentul are valoarea 1. Folosind imaginea grafică a codului Gray [31], în fig. 2.2 sînt prezentate diagramele Karnaugh pentru funcții de 3, 4 și 5 argumente. Într-un mod similar se pot construi diagramele Karnaugh și pentru FB cu $n > 5$.

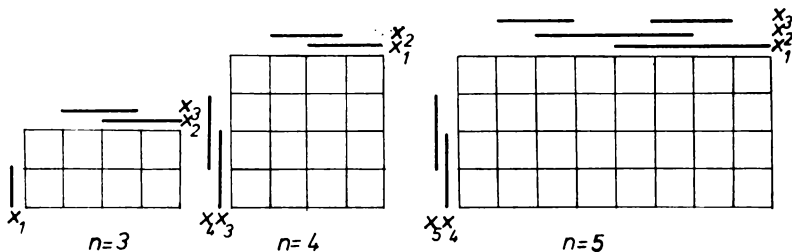


Fig. 2.2

2.3.3. Reprezentarea analitică a FB

Se consideră combinația valorilor argumentelor unei FB de n argumente $(\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n)$. Numărul „ i ” atașat n -uplului și definit prin relația :

$$(2.7) \quad i = \tilde{x}_1 2^{n-1} + \tilde{x}_2 2^{n-2} + \dots + \tilde{x}_n 2^0,$$

se numește *numărul combinației*.

De asemenea, se consideră funcția $P_i(x_1, x_2, \dots, x_n)$ care se definește în modul următor :

$$(2.8) \quad P_i = \begin{cases} 1, & \text{dacă numărul combinației este „}i\text{”} \\ 0 & \text{în caz contrar.} \end{cases}$$

Această funcție se denumește *constituent al unității* sau *funcția caracteristică a unității*. Presupunînd că se cunoaște expresia analitică a funcției P_i , a cărei determinare explicită se va face mai târziu, atunci se poate demonstra următoarea :

Teoremă. Orice FB dată prin tabel de adevăr poate fi scrisă sub următoarea formă analitică

$$(2.9) \quad f(x_1, x_2, \dots, x_n) = P_{i_1} \cup P_{i_2} \cup \dots \cup P_{i_k} = \bigcup_{i_j \in M_1} P_{i_j},$$

unde M_1 este mulțimea numerelor combinațiilor valorilor argumentelor pentru care funcția ia valoarea 1.

Demonstrație. Se consideră oricare dintre valorile argumentelor funcției considerate. Sînt posibile două cazuri distincte : funcția să ia valoarea 1 sau valoarea 0. Dacă funcția este egală cu unitatea, atunci în partea dreaptă a relației (2.9) se află funcția P_{i_j} al cărui i_j corespunde numărului combinației considerate. În acest caz, conform cu (2.8), pentru combinația respectivă a valorilor argumentelor P_{i_j} va fi egală cu 1. În virtutea proprietății (1.6) toată partea dreaptă a relației (2.9) va fi egală cu unitatea. Dacă însă pentru combinația considerată funcția este egală cu 0, atunci după cum decurge din formularea teoremei printre funcțiile P_{i_j} din relația (2.9) nu va fi nici una pentru care indicele să coincidă cu numărul combinației. În acest caz, conform cu (2.8), toți membrii disjuncției din partea dreaptă a relației (2.9) vor fi egali cu 0, deci partea dreaptă va fi egală cu 0. Întrucît s-a demonstrat că ambele părți ale relației (2.9) sînt identice pentru o combinație oarecare a valorilor argumentelor rezultă că este adevărată pentru orice altă combinație. Teorema este demonstrată.

Reprezentarea unei FB se poate face și sub altă formă analitică dacă se consideră funcția $S_i(x_1, x_2, \dots, x_n)$ definită în modu următor :

$$(2.10) \quad S_i = \begin{cases} 0, & \text{dacă numărul combinației este „i”,} \\ 1, & \text{în celelalte cazuri} \end{cases}$$

Funcția S_i va fi denumită *funcția caracteristică a lui zero sau constituentul lui zero*. Din relațiile de definiție (2.8) și (2.10) rezultă că $P_i(x_1, x_2, \dots, x_n) = \bar{S}_i(x_1, x_2, \dots, x_n)$!

Într-un mod similar demonstrării relației (2.9) se poate arăta că orice FB de n argumente poate fi reprezentată analitic și sub forma :

$$(2.11) \quad f(x_1, x_2, \dots, x_n) = S_{i_1} \cap S_{i_2} \cap \dots \cap S_{i_k} = \bigcap_{i_j \in M_0} S_{i_j}.$$

Reprezentarea FB sub forma (2.9) se numește *reprezentare disjunctivă*, iar sub forma (2.11) *reprezentare conjunctivă*.

Pentru a stabili expresiile funcțiilor P_i și S_i se introduce următoarea notație pentru o variabilă booleană :

$$(2.12) \quad x^a = \begin{cases} x, & \text{dacă } a=1 \\ \bar{x}, & \text{dacă } a=0. \end{cases}$$

Tab. 2.1

x_i	a_i	$x_i^{a_i}$
0	0	$x_i^0 = \bar{x}_i = \bar{0} = 1$
0	1	$x_i^1 = x_i = 0$
1	0	$x_i^0 = \bar{x}_i = \bar{1} = 0$
1	1	$x_i^1 = x_i = 1$

Se consideră expresia booleană :

$$(2.13) \quad x_1^{a_1} x_2^{a_2} \dots x_n^{a_n}.$$

În această conjuncție de n argumente orice termen $x_i^{a_i}$ este 1 dacă și numai dacă $x_i = a_i$. În adevăr, considerînd toate combinațiile posibile pentru x_i și a_i și ținînd cont de (2.12) se obțin rezultatele concentrate în tabelul 2.1. Conform celor demonstrate rezultă că funcția (2.13) este 1 numai dacă $x_1 = a_1, x_2 = a_2, \dots, x_n = a_n$, fiind 0 pentru toate celelalte cazuri. Avînd în vedere definiția (2.8) a funcției P_i rezultă :

$$P_i(x_1, x_2, \dots, x_n) = x_1^{a_1} x_2^{a_2} \dots x_n^{a_n},$$

cu condiția ca $i = a_1 2^{n-1} + a_2 2^{n-2} + \dots + a_n 2^0$.

În aceste condiții, orice FB poate fi descrisă printr-o expresie analitică de forma :

$$(2.14) \quad f(x_1, x_2, \dots, x_n) = \bigcup_{\mathbf{1}} (x_1^{a_1} x_2^{a_2} \dots x_n^{a_n}),$$

unde prin $\bigcup_{\mathbf{1}}$ s-a notat faptul că se consideră disjuncția termenilor conjunctivi (2.13) pentru care funcția f ia valoarea 1.

Reprezentarea FB sub forma (2.14) se numește *forma canonică disjunctivă (FCD)* a funcției, iar termenii (2.13) *termeni canonici conjunctivi (TCC)* sau *termeni minimali (mintermi)*.

Teorema demonstrată anterior și formula (2.14) permit stabilirea algoritmului trecerii de la tabelul de adevăr al unei FB la FCD.

Algoritm 2.1. Forma canonică disjunctivă a unei FB dată prin tabel de adevăr (sau diagramă Karnaugh) se obține astfel :

1. Din tabelul de adevăr (diagrama Karnaugh) se consideră toate n -uplele pe care funcția le aplică în 1.

2. Se scriu termenii canonici conjunctivi care corespund acestor n -uple. În acești termeni argumentul x_i intră ca atare sau negat după cum în n -uplu, considerat are valoarea 1 sau respectiv 0.

3. Termenii canonici conjunctivi obținuți se reunesc cu operația disjuncție.

Exemplu. Se consideră funcția $f(x_1, x_2, x_3)$ dată prin tabelul de adevăr :

x_1	x_2	x_3	$f(x_1, x_2, x_3)$	x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0	1	0	0	1
0	0	1	1	1	0	1	1
0	1	0	1	1	1	0	0
0	1	1	0	1	1	1	1

Se consideră combinațiile [valorilor argumentelor pentru care funcția are valoarea 1. Se scriu TCC corespunzători :

$$\bar{x}_1\bar{x}_2x_3, \bar{x}_1x_2\bar{x}_3, x_1\bar{x}_2\bar{x}_3, x_1\bar{x}_2x_3, x_1x_2x_3.$$

Reunind TCC prin operația disjuncție se obține FCD :

$$f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2x_3 \cup \bar{x}_1x_2\bar{x}_3 \cup x_1\bar{x}_2\bar{x}_3 \cup x_1\bar{x}_2x_3 \cup x_1x_2x_3.$$

Corespunzător relației (2.11) și definiției funcțiilor caracteristice, se poate scrie :

$$f(x_1, x_2, \dots, x_n) = \bigcap_0 S_{t_j} = \bigcap_0 \bar{P}_{t_j} = \bigcap_0 \overline{x_1^{a_1} x_2^{a_2} \dots x_n^{a_n}}$$

Aplicînd relațiile De Morgan, rezultă :

$$(2.15) \quad f(x_1, x_2, \dots, x_n) = \bigcap_0 (\overline{x_1^{a_1}} \cup \overline{x_2^{a_2}} \cup \dots \cup \overline{x_n^{a_n}}),$$

unde prin \bigcap_0 s-a notat faptul că se consideră numai n -uplele pentru care funcția ia valoarea 0.

Din relația (2.15) rezultă :

$$S_i(x_1, x_2, \dots, x_n) = \overline{x_1^{a_1}} \cup \overline{x_2^{a_2}} \cup \dots \cup \overline{x_n^{a_n}},$$

cu condiția ca $i = a_1 2^{n-1} + a_2 2^{n-2} + \dots + a_n 2^0$.

Rezultatele obținute în tabelul 2.1 confirmă condiția impusă și totodată permit dovedirea identității $\overline{x_i^{a_i}} = x_i^{\overline{a_i}}$. În acest caz relația (2.15) se mai poate scrie :

$$(2.16) \quad f(x_1, x_2, x_3, \dots, x_n) = \bigcap_0 (x_1^{\overline{a_1}} \cup x_2^{\overline{a_2}} \cup \dots \cup x_n^{\overline{a_n}})$$

Relațiile (2.15) și (2.16) sînt cunoscute sub denumirea de *formă canonică conjunctivă* (FCC), fiind *duala* formei canonice disjunctive. Termenii disjunctivi ($x_1^{\overline{a_1}} \cup x_2^{\overline{a_2}} \cup \dots \cup x_n^{\overline{a_n}}$) sînt denumiți *termen canonici disjunctivi* (TCD) sau *termeni maximali* (*max-termeni*). Și în acest caz relațiile (2.15) și (2.16) permit stabilirea algoritmului realizării FCC dacă se cunoaște tabelul funcției.

Algoritmul 2.2. Forma canonică conjunctivă a unei FB dată prin tabel de adevăr se obține în modul următor :

1. Din tabelul de adevăr al funcției se consideră toate *n*-uplele pe care funcția le aplică în 0.

2. Se scriu termenii canonici disjunctivi care corespund acestor *n*-uple. În expresia TCD argumentul x_i intră ca atare sau negat după cum în combinația considerată are valoarea 0 sau 1.

3. Termenii canonici disjunctivi obținuți la pasul 2 se reunesc prin semnul conjuncției.

Exemplu. Să stabilească FCC pentru funcția dată prin tabel în exemplul precedent. Conform algoritmului rezultă următorii TCD : $x_1 \cup x_2 \cup x_3, x_1 \cup \overline{x_2} \cup \overline{x_3}, \overline{x_1} \cup \overline{x_2} \cup x_3$. Forma canonică conjunctivă rezultă imediat : $f(x_1, x_2, x_3) = (x_1 \cup x_2 \cup x_3) (x_1 \cup \overline{x_2} \cup \overline{x_3}) (\overline{x_1} \cup \overline{x_2} \cup x_3)$.

Cele două forme canonice, disjunctivă și conjunctivă, sînt unice pentru o funcție booleană complet definită. Alegerea unei forme sau a celeilalte depinde de criteriul care stă la baza dezvoltării funcției în formă analitică. Dacă acest criteriu este cel al economicității atunci alegerea FCD sau FCC depinde de forma tabelului funcției respective. Astfel, dacă majoritatea valorilor funcției sînt zero este de preferat FCD ; în caz contrar o economie mai mare o asigură FCC.

2.3.4. Reprezentarea FB prin simbol de marcare

Simbolul de marcare este o reprezentare numerică a FB și derivă din reprezentarea prin tabel de adevăr. În tabelul de adevăr fiecărei combinații a valorilor argumentelor corespunde o valoare, 0 sau 1, pentru funcție. Se poate spune că fiecare

combinație definește „starea“ funcției. În cadrul reprezentării FB prin simbol de marcare se definește *numărul de stare* care coincide ca valoare cu numărul combinației :

$$n_s = \tilde{x}_1 2^{n-1} + \tilde{x}_2 2^{n-2} + \dots + \tilde{x}_n 2^0.$$

Pentru o ordine dată a argumentelor, șirul valorilor binare ale unei funcții în ordine crescătoare a numerelor de stare definește complet funcția respectivă. Numărul binar rezultat din înșiruirea biților 0 și 1 ai funcției se numește *număr de ordine* și se notează cu N . Astfel, indicarea ordinii argumentelor și a numărului de ordine constituie „marca“ unei FB .

Simbolul de marcare (M) constituie reprezentarea simbolică a modului de definire a unei FB mai sus menționat. Pentru o FB de n argumente pentru care s-a impus ordinea $x_1 x_2 \dots x_n$, reprezentarea prin simbol de marcare se definește sub forma :

$$(2.17) \quad f(x_1, x_2, \dots, x_n) = M_{(N)}^{x_1 x_2 \dots x_n}$$

Exemplu. Să se determine simbolul de marcare pentru funcția dată prin tabelul următor :

x_1	x_2	x_3	$f(x_1, x_2, x_3)$	x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0	1	0	0	1
0	0	1	1	1	0	1	1
0	1	0	0	1	1	0	0
0	1	1	1	1	1	1	1

Conform cu (2.17) simbolul de marcare se scrie :

$$f(x_1, x_2, x_3) = M_{(01011101)}^{x_1 x_2 x_3}$$

Dacă numărul de ordine se scrie în codul octal se obține o reprezentare mai compactă a simbolului de marcare. Astfel, pentru exemplul considerat rezultă :

$$f(x_1, x_2, x_3) = M_{(01011101)_8}^{x_1 x_2 x_3} = M_{(135)_8}^{x_1 x_2 x_3}$$

Deși simbolul de marcare definit prin (2.17) asigură o reprezentare avantajoasă, această formă nu este aptă pentru a se

opera cu ea. Pentru a se obține o *formă operativă* a simbolului de marcarea se înlocuiește numărul de ordine cu șirul crescător al numerelor de stare în cod octal.

Dacă în forma operativă se consideră numai numerele de stare pe care funcția le aplică în 1, simbolul de marcare are conținutul aceleiași informații ca și *FCD*. Rezultă următoarea relație de echivalență :

$$(2.18) \quad f(x_1, x_2, \dots, x_n) = \bigcup_1 x_1^{a_1} x_2^{a_2} \dots x_n^{a_n} = D_{(n_1, n_2, \dots, n_k)}^{x_1 x_2 \dots x_n},$$

unde (n_1, n_2, \dots, n_k) sînt numerele de stare în octal pentru care funcția are valoarea 1. Definierea simbolului de marcarea conform relației (2.18) poartă denumirea de *simbol D*, evidențiind echivalența cu *FCD*.

Într-un mod analog, dacă se consideră numai numerele de stare pentru care funcția ia valoarea 0, forma canonică conjunctivă poate fi exprimată prin simbolul de marcarea *C* :

$$(2.19) \quad f(x_1, x_2, \dots, x_n) = \bigcap_0 (x_1^{a_1} \cup x_2^{a_2} \cup \dots \cup x_n^{a_n}) = C_{(\bar{n}_1, \bar{n}_2, \dots, \bar{n}_n)}^{x_1 x_2 \dots x_n},$$

unde $(\bar{n}_1, \bar{n}_2, \dots, \bar{n}_n)$ sînt numerele de stare în cod octal pentru care funcția are valoarea 0, complementate.

Deoarece pentru o *FB* complet definită mulțimile numerelor de stare (n_1, n_2, \dots, n_k) și $(\bar{n}_1, \bar{n}_2, \dots, \bar{n}_n)$ sînt disjuncte, iar relațiile (2.18) și (2.19) sînt forme diferite ale aceleiași funcții, se poate scrie :

$$(2.20) \quad D_{(n_1, n_2, \dots, n_k)}^{x_1 x_2 \dots x_n} = \bar{C}_{(\bar{n}_1, \bar{n}_2, \dots, \bar{n}_n)}^{x_1 x_2 \dots x_n}$$

Din definiția numărului de stare și expresiile simbolurilor *D* și *C* rezultă că în (2.18) numerele de stare corespund mintermilor din *FCD*, iar în (2.19) numerele de stare corespund maxtermilor din *FCC*. Astfel, expresiile (2.18) și (2.19) permit folosirea algoritmilor 2.1 și 2.2 și pentru trecerea de la tabel de adevăr sau de la formele canonice la reprezentarea prin simbol de marcarea.

Exemplu. Să se stabilească în formă operativă simbolurile *D* și *C* pentru funcția dată prin tabel în exemplul precedent.

Numerele de stare pe care funcția le aplică în 1 sînt : $(001)_2 = (1)_8$; $(011)_2 = (3)_8$; $(100)_2 = (4)_8$; $(101)_2 = (5)_8$; $(111)_2 = (7)_8$. Conform cu (2.18) se poate scrie simbolul de marcarea *D* :

$$f(x_1, x_2, x_3) = D_{(1, 3, 4, 5, 7)}^{x_1 x_2 x_3} = \bar{x}_1 \bar{x}_2 x_3 \cup \bar{x}_1 x_2 x_3 \cup x_1 \bar{x}_2 \bar{x}_3 \cup x_1 \bar{x}_2 x_3 \cup x_1 x_2 x_3.$$

Numererele de stare pe care funcția le aplică în 0 sînt : $(000)_2 = (0)_8$; $(010)_2 = (2)_8$; $(110)_2 = (6)_8$. Conform cu (2.19), rezultă :

$$f(x_1, x_2, x_3) = C_{(0, 2, 6)}^{x_1, x_2, x_3} = C_{(7, 5, 1)}^{x_1, x_2, x_3} = (x_1 \cup x_2 \cup x_3) (x_1 \cup \bar{x}_2 \cup x_3) (\bar{x}_1 \cup \bar{x}_2 \cup x_3).$$

În cazul *FB* incomplet definite este necesar ca atît în (2.18) cît și în (2.19) să se indice numerele de stare pe care funcția le aplică în 0 și în 1, precum și stările indiferente. Numerele de stare indiferente se indică cu asterisc. Pentru *FB* incomplet definite relația (2.20) nu mai este adevărată întrucît numerele de stare indiferente se pot aplica diferit în 0 sau 1.

Exemplu. Să se stabilească simbolurile de marcare *D* și *C* pentru funcția incomplet definită dată prin următorul tabel :

x_1	x_2	x_3	x_4	$f(x_1, x_2, x_3, x_4)$	x_1	x_2	x_3	x_4	$f(x_1, x_2, x_3, x_4)$
0	0	0	0	0	1	0	0	0	1
0	0	0	1	*	1	0	0	1	0
0	0	1	0	1	1	0	1	0	0
0	0	1	1	1	1	0	1	1	1
0	1	0	0	0	1	1	0	0	1
0	1	0	1	1	1	1	0	1	*
0	1	1	0	*	1	1	1	0	0
0	1	1	1	*	1	1	1	1	1

Neținînd cont de numerele de stare indiferente, simbolurile *D* și *C* se scriu :

$$f(x_1, x_2, x_3, x_4) = D_{\substack{1: (2, 3, 5, 10, 13, 14, 17) \\ 0: (0, 4, 11, 12, 16)}}$$

$$f(x_1, \dots, x_4) = C_{\substack{0: (0, 4, 11, 12, 16) \\ 1: (2, 3, 5, 10, 13, 14, 17)}}$$

În cele două reprezentări s-au specificat numai numerele de stare aplicate în 1 și respectiv în 0 ; numerele indiferente fiind cele necuprinse. Deoarece numerele de stare indiferente pot fi aplicate atît în 1 cît și în 0, dacă numerele 1* și 7* se aplică în 1 iar numerele 6* și 15* se aplică în 0 rezultă simbolul :

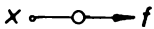





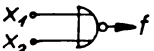

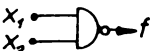


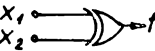

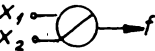
$$f(x_1, x_2, x_3, x_4) = D_{1: (1^* 2, 3, 5, 7^*, 10, 13, 14, 17)} \\ 0: (0, 4, 6^*, 11, 12, 15^*, 16)$$

Într-un mod similar se poate defini și simbolul C .

2.3.5. Reprezentarea FB prin scheme logice (logigrame)

Schema logică (logigrama) este o reprezentare grafică a FB obținută prin adoptarea unor semne convenționale pentru operațiile logice. Logigrama indică în fapt topologia unui circuit logic care materializează o FB . Ca urmare, simbolurile adoptate pentru operațiile logice constituie o reprezentare a circuitelor logice care materializează funcțiile logice elementare. În tabelul 2.2 sînt indicate cele mai utilizate semne grafice pentru principalele funcții elementare de două argumente.

Tab. 2.2

Denumirea funcției	Funcția	Reprezentarea simbolică (logigrama)
Negația	$f = \bar{x}$	 
Conjunția	$f = x_1 x_2$	 
Disjuncția	$f = x_1 \cup x_2$	 
Peirce (Webb)	$f = x_1 \downarrow x_2$	 
Sheffer	$f = x_1 \uparrow x_2$	 
Suma modulo 2	$f = x_1 \oplus x_2$	 
Interdicție	$f = x_1 x_2$	 

Folosind aceste simboluri grafice expresiile algebrice ale FB pot fi reprezentate sub formă de scheme logice.

Exemplu. Să se reprezinte prin logigramă funcția $f(x_1, x_2, x_3) = x_1 x_2 x_3 \cup x_1 \bar{x}_2 x_3 \cup x_1 \bar{x}_2 \bar{x}_3 \cup \bar{x}_1 \bar{x}_2 \bar{x}_3$. Avînd în vedere tabelul 2.2, în fig. 2.3 este prezentată logigrama funcției date. Schema logică

indică și nivelele logice compuse din elemente fizice care operează simultan. De câte ori este posibil, elementele aceleiași nivel logic se reprezintă pe aceeași linie.

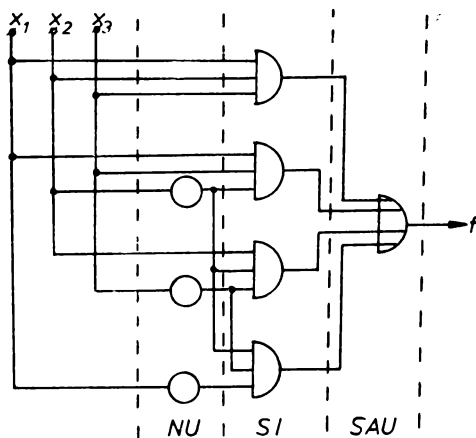


Fig. 2.3

2.3.6. Reprezentarea FB prin diagrame în timp

Diagrama în timp reprezintă grafic o FB prin forma semnalelor corespunzătoare argumentelor și funcției. Cifrele binare 0 și 1 se atașează semnalelor de nivel coborât și respectiv ridicat, astfel ca să existe o diferențiere netă a acestora. Reprezentarea prin diagrame în timp este deosebit de utilă în studiul sistemelor secvențiale în a căror evoluție intervine și timpul. De asemenea, folosind această reprezentare se pot studia fenomenele tranzitorii de comutare și fenomenele de hazard datorate funcționării neideale a elementelor care materializează variabile sau funcții booleene.

Exemplu. Să se reprezinte prin diagramă în timp funcția $f(x_1, x_2) = x_1 \uparrow x_2$, cunoscând evoluția în timp a semnalelor atașate argumentelor (fig. 2.4 a). Avînd în vedere tabelul de definiție al funcției NUMAI în

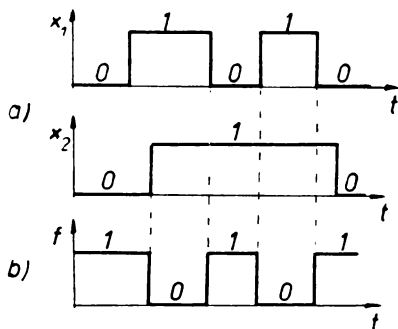


Fig. 2.4.

fig. 2.4b se dă reprezentarea prin diagramă în timp a funcției considerate, pentru evoluția dată a argumentelor.

2.4. Sisteme complete de funcții

În capitolul 2.2 s-a arătat cum se pot forma FB prin superpoziția funcțiilor elementare. Problema formării FB poate fi privită și astfel: să se stabilească un sistem de funcții care să poată reprezenta orice FB și în plus, dacă acest sistem există, să conțină un număr cât mai mic de funcții. Evident, problema formării FB pusă în acest mod are o mare importanță pentru sinteza sistemelor logice.

Definiție. Sistemul de funcții booleene (f_1, f_2, \dots, f_k) se numește *complet (bază)* în clasă R , dacă orice funcție φ aparținând lui R poate fi reprezentată prin superpoziția funcțiilor (f_1, f_2, \dots, f_k) .

În calitate de clasă R se poate considera clasa tuturor FB care depind de n argumente, B_2^n . Conform celor arătate în prima parte a cap. 2 numărul total al funcțiilor care depind de n argumente este egal cu 2^{2^n} . Deci, în clasa B_2^n există un sistem complet compus din cele 2^{2^n} funcții ale acestei clase, dar acest sistem este trivial. În paragraful 2.3.3 s-a stabilit că orice FB poate fi reprezentată sub FCD sau FCC . Rezultă că sistemul de funcții compus din conjuncție, disjuncție și negare constituie un sistem complet de funcții. Se pune problema dacă sistemul complet de funcții $(\cup, \cap, -)$ conține numărul strict necesar pentru reprezentarea oricărei FB . Pentru ca un sistem complet să fie minim (bază minimală) este necesar să satisfacă următoarea definiție:

Un sistem complet de funcții (f_1, f_2, \dots, f_k) este *minim (bază minimală)* dacă înlăturând oricare dintre funcții aparținând sistemului acesta devine incomplet.

Completitudinea sistemului de funcții $(\cup, \cap, -)$ permite demonstrarea completitudinii oricărui alt sistem de funcții arbitrar format. Pentru aceasta este suficient să se arate că funcțiile sistemului ales pot reprezenta funcțiile sistemului $(\cup, \cap, -)$.

Corespunzător celor arătate mai sus, interes practic prezintă demonstrarea următoarelor teoreme:

- *Teorema 2.4.1.* Sistemul de funcții format din conjuncție și negație, $(\cup, -)$, este un sistem complet în clasa B_2^n .

Demonstrație. Pentru a demonstra completitudinea sistemului $(\cup, -)$ este necesar să se demonstreze că funcția disjuncție poate fi reprezentată prin funcțiile conjuncție și negație. Considerînd cazul general al unei funcții SAU de n argumente, se poate scrie :

$$f(x_1, x_2, \dots, x_n) = x_1 \cup x_2 \cup \dots \cup x_n = \overline{\overline{x_1 \cup x_2 \cup \dots \cup x_n}} = \overline{\overline{x_1} \cap \overline{x_2} \cap \dots \cap \overline{x_n}}.$$

Rezultă că sistemul $(\cup, -)$ este un sistem complet, orice FB putînd fi reprezentată prin funcțiile conjuncție și negație.

Teorema 2.4.2. Sistemul de funcții format din disjuncție și negație este un sistem complet în clasa B_2^n .

Demonstrație. Se consideră o funcție SI de n argumente :

$$f(x_1, x_2, \dots, x_n) = x_1 x_2 \dots x_n = \overline{\overline{x_1 x_2 \dots x_n}} = \overline{\overline{x_1} \cap \overline{x_2} \cap \dots \cap \overline{x_n}},$$

de unde rezultă că funcția conjuncție poate fi reprezentată prin funcțiile disjuncție și negație.

Dacă conform teoremelor 2.4.1 și 2.4.2 sistemele $(\cap, -)$ și $(\cup, -)$ sînt complete, în același timp formează baze minimale în raport cu sistemul $(\cup, \cap, -)$.

Teorema 2.4.3. Funcția lui Sheffer formează în clasa B_2^n un sistem complet.

Demonstrație. Se va demonstra că funcția Sheffer (\uparrow) poate reprezenta sistemul complet $(\cap, -)$. În adevăr, pentru funcția negație se poate scrie :

$$\bar{x} = \bar{x} \cup \bar{x} = x \uparrow x,$$

iar pentru funcția conjuncție

$$x_1 x_2 = \overline{\overline{x_1 x_2}} = \overline{x_1 \uparrow x_2} = (x_1 \uparrow x_2) \uparrow (x_1 \uparrow x_2).$$

Întrucît sistemul $(\cap, -)$ este un sistem minim și funcția NUMAI formează un sistem complet și minimal.

Teorema 2.4.4. Funcția lui Pierce formează în clasa B_2^n un sistem complet.

Demonstrație. Se va demonstra că funcția NICI poate realiza prin superpoziție sistemul complet $(\cup, -)$. Astfel, pentru negație se scrie :

$$\bar{x} = \bar{x} \cdot \bar{x} = x \downarrow (x \downarrow x).$$

iar pentru disjuncție

$$x_1 \cup x_2 = \overline{\overline{x_1} \cap \overline{x_2}} = \overline{x_1 \downarrow x_2} = (x_1 \downarrow x_2) \downarrow (x_1 \downarrow x_2).$$

Deci și sistemul monofuncțional format de funcția lui Pierce este un sistem minim în clasa B_2^n .

Teoremele 2.4.3 și 2.4.4 sînt deosebit de importante pentru aplicarea practică la sinteza circuitelor logice, permițînd folosirea unui singur tip de circuit pentru materializarea oricărei funcții booleene. În acest context devine importantă trecerea de la FCD și FCC la forme cu funcții Pierce sau Sheffer. Această trecere este cunoscută și sub denumirea de *implementare*.

2.4.1. Implementarea FCD și FCC cu funcții Pierce și Sheffer

Prin analogie cu definirea funcțiilor elementare Pierce și Sheffer pentru două argumente, se vor defini aceste funcții pentru n argumente folosind tabelul de adevăr:

x_1	x_2	x_3	\dots	x_{n-1}	x_n	Pierce	Sheffer
0	0	0	\dots	0	0	1	1
0	0	0	\dots	0	1	0	1
0	0	0	\dots	1	0	0	1
\vdots	\vdots	\vdots	\dots	\vdots	\vdots	\vdots	\vdots
1	1	1	\dots	1	0	0	1
1	1	1	\dots	0	1	0	1
1	1	1	\dots	1	1	0	0

Conform acestui tabel, rezultă:

— funcția Pierce de n argumente: $P_n = x_1 \downarrow x_2 \downarrow \dots \downarrow x_n = \overline{x_1 \cup x_2 \cup \dots \cup x_n}$;

— funcția Sheffer de n argumente: $S_n = x_1 \uparrow x_2 \uparrow \dots \uparrow x_n = \overline{x_1 \cap x_2 \cap \dots \cap x_n}$.

Pentru $n=2$ din tabelul de adevăr se obțin funcțiile Pierce și Sheffer de două argumente, iar pentru $n=1$ ambele funcții se transformă în funcția negație: $P_1 = S_1 = \bar{x}$.

Exprimarea prin funcții Pierce a unei FB se obține considerînd FCC a acesteia și gînd de două ori termenii disjunctivi:

$$f(x_1, x_2, \dots, x_n) = \bigcap_{\theta} (x_1^{\theta} \cup x_2^{\theta} \cup \dots \cup x_n^{\theta}) = \bigcap_0 (\overline{x_1^{\bar{\theta}} \cup x_2^{\bar{\theta}} \cup \dots \cup x_n^{\bar{\theta}}})$$

Aplicînd relațiile De Morgan și ținînd cont de definirea funcției Pierce de n argumente, se obține :

$$(2.21) \quad f(x_1, x_2, \dots, x_n) = \bigcap_0 \overline{x_1^a x_2^a \dots x_n^a} = \downarrow_0 (x_1^a \downarrow x_2^a \downarrow \dots \downarrow x_n^a),$$

unde prin \downarrow_0 s-a notat faptul că se consideră numai n -uplele pe care funcția le aplică în zero.

Concluzie. Pentru implementarea unei FB cu funcții Pierce se pleacă de la FCC și se înlocuiesc operațiile \cup și \cap cu operația \downarrow .

Avînd în vedere cele demonstrate mai sus se poate stabili algoritmul implementării oricărei FB cu funcții NICI plecînd de la tabelul de adevăr sau de la diagrama Karnaugh.

Algoritmul 2.3. Implementarea unei FB de n argumente cu funcții NICI plecînd de la tabelul de adevăr se obține astfel :

1. Se consideră n -uplele pe care funcția le aplică în 0.

2. Fiecărui n -uplu considerat îi corespunde un termen implementat cu funcții NICI. În acești termeni fiecare argument intră ca atare sau negat după cum în combinația respectivă are valoarea 0 sau respectiv 1.

3. Toți termenii obținuți la pasul 2 se reunesc prin simbolul funcției NICI.

4. Excepție de la punctul 3 face situația cînd funcția are un singur n -uplu aplicat în 0. În acest caz termenul respectiv se neagă. Explicația rezultă din faptul că $x \downarrow x = \bar{x}$.

Exemplu. Să se implementeze cu funcții NICI funcția dată prin tabelul :

x_1	x_2	x_3	$f(x_1, x_2, x_3)$	x_1	x_2	x_3	$f(x_1, x_2, x_3)$
0	0	0	0	1	0	0	1
0	0	1	1	1	0	1	0
0	1	0	1	1	1	0	1
0	1	1	0	1	1	1	1

Conform algoritmului 2.3 se obține imediat :

$$f(x_1, x_2, x_3) = (x_1 \downarrow x_2 \downarrow x_3) \downarrow (x_1 \downarrow \bar{x}_2 \downarrow \bar{x}_3) \downarrow (\bar{x}_1 \downarrow x_2 \downarrow \bar{x}_3).$$

Dacă funcția ar aplica în 0 numai combinația (1, 0, 1) ar rezulta :

$$f(x_1, x_2, x_3) = \overline{\bar{x}_1 \downarrow x_2 \downarrow \bar{x}_3}.$$

Exprimarea prin funcții Sheffer a unei FB se obține considerând FCD și negînd de două ori termenii conjunctivi :

$$f(x_1, x_2, \dots, x_n) = \bigcup_1 (x_1^{a_1} x_2^{a_2} \dots x_n^{a_n}) = \bigcup_1 \overline{(x_1^{a_1} x_2^{a_2} \dots x_n^{a_n})}$$

Aplicînd relațiile De Morgan și ținînd cont de relația de definiție a funcției Sheffer de n argumente rezultă :

$$(2.22) f(x_1, x_2, \dots, x_n) = \bigcup_1 \overline{(x_1^{a_1} \uparrow x_2^{a_2} \uparrow \dots \uparrow x_n^{a_n})} = \uparrow_1 (x_1^{a_1} \uparrow x_2^{a_2} \uparrow \dots \uparrow x_n^{a_n}).$$

unde prin \uparrow_1 s-a notat faptul că se consideră numai n -uplele aplicate de funcție în 1.

Concluzie. Pentru implementarea cu funcții Sheffer a unei FB se pleacă de la FCD în care se înlocuiesc simbolurile \cup și \cap cu simbolul \uparrow .

Avînd în vedere (2.22) și algoritmul 2.2 se poate stabili și în acest caz un algoritm pentru implementarea cu funcții NUMAI a unei FB plecînd de la tabel de adevăr sau diagramă Karnaug.

Algoritmul 2.4. Implementarea unei FB de n argumente cu funcții NUMAI plecînd de la tabelul de adevăr se obține astfel :

1. Se consideră toate n -uplele pe care funcția le aplică în 1.
2. Fiecărui n -uplu îi corespunde un termen implementat cu funcții NUMAI, în care fiecare variabilă se ia ca atare sau negată după cum în combinația considerată are valoarea 1 sau respectiv 0.
3. Termenii obținuți la pasul 2 se reunesc prin simbolul funcției NUMAI.
4. Excepție de la punctul 3 face cazul cînd funcția aplică în 1 numai un n -uplu ; termenul implementat cu funcții NUMAI se neagă deoarece $x \uparrow x = \bar{x}$.

Exemplu. Să se implementeze cu funcții NUMAI funcția de la exemplul precedent.

Considerînd combinațiile valorilor argumentelor pe care funcția le aplică în 1 și ținînd cont de algoritmul 2.4, rezultă :

$$f(x_1, x_2, x_3) = (\bar{x}_1 \uparrow \bar{x}_2 \uparrow x_3) \uparrow (\bar{x}_1 \uparrow x_2 \uparrow \bar{x}_3) \uparrow (x_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3) \uparrow (x_1 \uparrow x_2 \uparrow \bar{x}_3) \uparrow (x_1 \uparrow x_2 \uparrow x_3).$$

Dacă însă funcția ar aplica în 1 numai combinația $(0, 1, 0)$, atunci : $f(x_1, x_2, x_3) = \bar{x}_1 \uparrow x_2 \uparrow \bar{x}_3$.

2.4.3. Alte sisteme complete de funcții

Pentru realizarea sistemelor complete de funcții s-au considerat pînă acum numai acele funcții care necesită minimum două argumente. De mare interes practic mai sînt și alte funcții care necesită însă minimum trei argumente, operațiile cu aceste funcții fiind ternare. Din această categorie face parte *funcția majoritară* :

$$(2.23) \quad \text{Maj}(x_1, x_2, \dots, x_n) = x_1 \# x \#_2 \dots \# x_n.$$

Funcțiile majoritare sînt de un număr impar de argumente. Aceste funcții se aplică în 1 numai atunci cînd majoritatea argumentelor au valoare 1. Negata funcției majoritare este funcția minoritară :

$$(2.24) \quad \text{Min}(x_1, x_2, \dots, x_n) = \overline{\text{Maj}}(x_1, x_2, \dots, x_n).$$

Funcțiile majoritară și minoritară elementare sînt definite prin tabelul următor :

x_1	x_2	x_3	$\text{Maj}(x_1, x_2, x_3)$	$\text{Min}(x_1, x_2, x_3)$	x_1	x_2	x_3	$\text{Maj}(x_1, x_2, x_3)$	$\text{Min}(x_1, x_2, x_3)$
0	0	0	0	1	1	0	0	0	1
0	0	1	0	1	1	0	1	1	0
0	1	0	0	1	1	1	0	1	0
0	1	1	1	0	1	1	1	1	0

Conform tabelului și definiției, rezultă :

$$(2.25) \quad \begin{aligned} \text{Maj}(x_1, x_2, x_3) &= x_1 x_2 \cup x_1 x_3 \cup x_2 x_3 = \\ &= (x_1 \cup x_2)(x_1 \cup x_3)(x_2 \cup x_3) \end{aligned}$$

și respectiv

$$(2.26) \quad \begin{aligned} \text{Min}(x_1, x_2, x_3) &= \bar{x}_1 \bar{x}_2 \cup \bar{x}_1 \bar{x}_3 \cup \bar{x}_2 \bar{x}_3 = \\ &= (\bar{x}_1 \cup \bar{x}_2)(\bar{x}_1 \cup \bar{x}_3)(\bar{x}_2 \cup \bar{x}_3). \end{aligned}$$

Legat de aceste funcții interes are următoarea :

Teoremă. Funcția majoritară împreună cu funcțiile negație și constanta 0 formează un sistem complet în clasa B_2^n .

Demonstrație. Se va arăta că sistemul (Maj., —, 0) poate reprezenta sistemul complet (\cup , \cap , —). În adevăr: $x_1 x_2 = \text{Maj}(x_1, x_2, 0) = x_1 x_2 \cup x_1 0 \cup x_2 0$; $x_1 \cup x_2 = \text{Maj}(x_1, x_2, \bar{0}) = (x_1 \cup x_2)(x_1 \cup 1)(x_2 \cup 1)$.

2.5. Clase de funcții booleene

Pentru rezolvarea unor probleme legate de aplicarea teoriei *FB* la sinteza sistemelor logice este util a se vedea care sînt principalele clase de funcții booleene.

2.5.1. Funcții degenerare și funcții nedegenerate

La definirea funcțiilor elementare de două variabile s-a văzut că din cele $2^2 = 16$ funcții numai 10 depind de ambele argumente, restul de 6 fiind ori funcții constante ori funcții care depind de un singur argument.

Definiție. O funcție booleană de n argumente $f(X)$, $X = (x_1, x_2, \dots, x_n)$ se numește *nedegenerată* dacă pentru oricare $x_i (i = 1, 2, \dots, n)$ are loc relația :

$$f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) \neq f(x_1, x_2, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n).$$

Dacă însă pentru un anumit argument (sau mai multe) are loc identitatea :

$$f(x_1, \dots, x_{i-1}, 0, x_{i+1}, \dots, x_n) = f(x_1, \dots, x_{i-1}, 1, x_{i+1}, \dots, x_n),$$

funcția se numește *degenerată* sau *vidă* de argumentul x_i , iar acest argument (argumentele) se numește *argument fictiv* al funcției.

Numărul *FB* nedegenerate de n argumente se poate determina cu următoarea relație recurentă :

$$(2.27) \quad N_n = 2^{2^n} - \sum_{i=0}^{i=n-1} C_n^i N_i,$$

unde C_n^i sînt combinațiile de n argumente luate cîte i , iar N_i — numărul funcțiilor nedegenerate de $i < n$ argumente.

Exemplu. Să se stabilească numărul *FB* nedegenerate de trei argumente. Deoarece pentru $n=0$ avem $f(0)=0$ și $f(0)=1$, deci $N_0=2$ și folosind relația (2.27), rezultă :

$$- \text{ pentru } n=1, N_1 = 2^{2^1} - N_0 = 2.$$

- pentru $n=2$, $N_2=2^{2^2}-(C_2^1N_1+N_0)=16-(2\cdot 2+2)=10$.
- pentru $n=3$, $N_3=2^{2^3}-(C_3^2N_2+C_3^1N_1+N_0)=256-(3\cdot 10+3\cdot 2+2)=218$.

Din definiția funcțiilor degenerare și nedegenerate rezultă că pentru funcțiile degenerare argumentele fictive pot fi eliminate. Ca un corolar, la o FB pot fi adăugate un număr oarecare de argumente fictive fără ca aceasta să se modifice.

2.5.2. Funcții duale și autoduale

Principiul dualității pentru latici se păstrează și în algebra booleană, fiind necesar să se schimbe în afara operațiilor \cup cu \cap și 1 cu 0. Astfel, dualul lui $0^a=1$ iar dualul lui $1^a=0$.

Definiție. Funcția $f^a(x_1, x_2, \dots, x_n)$ se numește *duala* funcției $f(x_1, x_2, \dots, x_n)$ dacă se obține din aceasta prin aplicarea principiului dualității.

Conform definiției duala FCD este funcția următoare :

$$f(x_1, x_2, \dots, x_n) = \bigcup_1 (x_1^{a_1} x_2^{a_2} \dots x_n^{a_n}) \rightarrow f^a(x_1, \dots, x_n) = \bigcap_1 (x_1^{a_1} \cup \dots \cup x_n^{a_n}),$$

iar pentru FCC :

$$f(x_1, \dots, x_n) = \bigcap_0 (x_1^{\bar{a}_1} \cup x_2^{\bar{a}_2} \cup \dots \cup x_n^{\bar{a}_n}) \rightarrow f^a(x_1, \dots, x_n) = \bigcup_0 (x_1^{a_1} x_2^{a_2} \dots x_n^{a_n}).$$

Lemă. Între funcția duală și funcția negată ale unei FB există relația :

$$(2.28) \quad f^a(x_1, x_2, \dots, x_n) = \bar{f}(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n).$$

Demonstrație. Este suficient să se verifice relația pentru una din formele funcției duale și una din formele funcției negate ale formelor canonice ; celelalte forme sînt *echivalente*. Astfel, pentru FCD :

$$\begin{aligned} \bar{f}(x_1, x_2, \dots, x_n) &= \overline{\bigcup_1 (x_1^{a_1} x_2^{a_2} \dots x_n^{a_n})} = \bigcap_1 (\overline{x_1^{a_1} \cup x_2^{a_2} \cup \dots \cup x_n^{a_n}}) \rightarrow \\ \rightarrow \bar{f}(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n) &= \bigcap_1 (x_1^{a_1} \cup x_2^{a_2} \cup \dots \cup x_n^{a_n}) = f^a(x_1, x_2, \dots, x_n). \end{aligned}$$

Exemplu. Să se stabilească duala funcției $f(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 \bar{x}_3 \cup x_1 \bar{x}_2 x_3 \cup \bar{x}_1 x_2 \bar{x}_3$. Conform definiției rezultă $f^a(x_1, x_2, x_3) = (\bar{x}_1 \cup \bar{x}_2 \cup \bar{x}_3)(x_1 \cup \bar{x}_2 \cup x_3)(\bar{x}_1 \cup x_2 \cup \bar{x}_3)$. Folosind relația (2.28) se obține : $f(x_1, x_2, x_3) = (x_1 \cup x_2 \cup x_3)(\bar{x}_1 \cup x_2 \cup \bar{x}_3)(x_1 \cup \bar{x}_2 \cup x_3) \rightarrow \bar{f}(\bar{x}_1, \bar{x}_2, \bar{x}_3) = (\bar{x}_1 \cup \bar{x}_2 \cup \bar{x}_3)(x_1 \cup \bar{x}_2 \cup x_3)(\bar{x}_1 \cup x_2 \cup \bar{x}_3)$, care coincide cu f^a .

Definiție. Funcția $f(x_1, x_2, \dots, x_n)$ se numește *autoduală*, dacă ea coincide cu duala funcției, adică are loc identitatea $f(x_1, x_2, \dots, x_n) = \bar{f}(\bar{x}_1, \dots, \bar{x}_n)$. Conform definiției, expresia unei funcții autoduale este identică cu duala funcției considerate.

2.5.3. Funcții cu prag

Definiție. O FB de n argumente este denumită *funcție cu prag*, dacă există o mulțime de numere reale $\{w_1, w_2, \dots, w_n, P\}$, astfel încît :

$$f(x_1, x_2, \dots, x_n) = \begin{cases} 1, & \text{dacă } \sum_{i=1}^n w_i x_i \geq P \\ 0, & \text{dacă } \sum_{i=1}^n w_i x_i < P, \end{cases}$$

unde $x_i (i=1, 2, \dots, n)$ sînt variabile booleene.

Numerele w_1, w_2, \dots, w_n se numesc *ponderi*, iar numărul P se numește *pragul* funcției. Simbolul Σ reprezintă o sumă aritmetică în care w_i se înmulțește aritmetic cu x_i .

Exemplu. Să se stabilească dacă funcția $f(x_1, x_2, x_3) = x_1 x_2 x_3 \cup x_1 \bar{x}_2 x_3 \cup \bar{x}_1 x_2 x_3$ este o funcție cu prag pentru $w_1=0,75$, $w_2=0,5$, $w_3=1$ și $P=1,5$.

Conform definiției este necesar să se stabilească valorile funcției și să se compare $\sum_{i=1}^3 w_i x_i = w_1 x_1 + w_2 x_2 + w_3 x_3$ cu valoarea lui P . Rezultatele calculelor sînt trecute în tabelul următor :

x_1	x_2	x_3	$f(x_1, x_2, x_3)$	$w_1 x_1 + w_2 x_2 + w_3 x_3$
0	0	0	0	$0 + 0 + 0 = 0 < P = 1,5$
0	0	1	0	$0 + 0 + 1 = 1 < P$
0	1	0	0	$0 + 0,5 = 0,5 < P$
0	1	1	1	$0 + 0,5 + 1 = 1,5 = P$
1	0	0	0	$0,75 + 0 + 0 = 0,75 < P$
1	0	1	1	$0,75 + 0 + 1 = 1,75 > P$
1	1	0	0	$0,75 + 0,5 + 0 = 1,25 < P$
1	1	1	1	$0,75 + 0,5 + 1 = 2,25 > P$

Deoarece se respectă relațiile de definiție pentru toate combinațiile posibile ale valorilor argumentelor, funcția considerată este o funcție cu prag pentru ponderile și pragul dat.

Deși nu toate FB sînt funcții cu prag, aceste funcții au mare importanță practică deoarece pot reprezenta orice FB . Astfel, sistemele complete monofuncționale NICI și NUMAI pot fi reprezentate prin funcții cu prag. Afirmăția se va verifica pentru funcțiile Pierce și Sheffer de două argumente. Alegînd $w_1 = w_2 = -1$ și $P = -0,5$, pentru funcția Pierce este necesar ca :

$$f(x_1, x_2) = x_1 \downarrow x_2 = \overline{x_1 \cup x_2} = \begin{cases} 1, & \text{dacă } w_1 x_1 + w_2 x_2 \geq P \\ 0, & \text{dacă } w_1 x_1 + w_2 x_2 < P \end{cases}$$

De asemenea, păstrînd aceleași valori pentru ponderi dar luînd $P = -1,5$, funcția NUMAI devine o funcție cu prag. Verificarea celor afirmate se găsește în tabelul următor :

x_1	x_2	$x_1 \downarrow x_2$	$(w_1 x_1 + w_2 x_2)_{-0,5}$	x_1	x_2	$x_1 \uparrow x_2$	$(w_1 x_1 + w_2 x_2)_{-1,5}$
0	0	1	$0 > -0,5$	0	0	1	$0 > -1,5$
0	1	0	$-1 < -0,5$	0	1	1	$-1 > -1,5$
1	0	0	$-1 < -0,5$	1	0	1	$-1 > -1,5$
1	1	0	$-2 < -0,5$	1	1	0	$-2 < -1,5$

Ca un corolar al celor demonstrate mai sus este faptul că clasa funcțiilor cu prag formează un sistem complet în B_2^n .

Funcția majoritară poate fi de asemenea reprezentată prin funcții cu prag. Astfel, se poate verifica ușor că $\text{Maj}(x_1, x_2, \dots, x_n)$ poate fi reprezentată prin funcția cu prag avînd $w_1 = w_2 = \dots = w_n = 1$ și $P = (n+1)/2$.

Din cele prezentate rezultă concluzia că funcțiile cu prag au o *potențialitate structurală* mai mare decît sistemele complete formate cu funcțiile NICI, NUMAI sau Maj.

2.5.4. Funcții simetrice

Definiție. O funcție booleană de n argumente se numește *simetrică* dacă rămîne neschimbată la orice permutare a argumentelor.

Exemplu. Funcția suma modulo 2 este o funcție simetrică. În adevăr $f(x_1, x_2) = x_1 \oplus x_2 = x_1 \bar{x}_2 \cup \bar{x}_1 x_2 = x_2 \bar{x}_1 \cup \bar{x}_2 x_1$.

De asemenea, funcția Maj este o funcție simetrică. Orice permutare a argumentelor din tabelul de definiție (v. paragraful

2.4.3) nu afectează valoarea acesteia ; toate tripletele de valori ale argumentelor formate de doi de 1 și un 0 se aplică în 1 indiferent de poziția pe care apar în tabel.

2.5.5. Funcții monotone

Se consideră o *FB* de n argumente $f(X)$, $X = (x_1, x_2, \dots, x_n)$ și două n -uple oarecare $\tilde{X}_1 = (\tilde{x}_1^1, \tilde{x}_2^1, \dots, \tilde{x}_n^1)$ și $\tilde{X}_2 = (\tilde{x}_1^2, \tilde{x}_2^2, \dots, \tilde{x}_n^2)$. Se va spune că n -uplul \tilde{X}_1 este mai mare decât n -uplul \tilde{X}_2 dacă între toate componentele acestor n -uple există relația $x_i^1 \geq x_i^2$ ($i=1, 2, \dots, n$).

Exemplu. Fie $\tilde{X}_1 = (1, 0, 1, 1)$, $\tilde{X}_2 = (1, 0, 1, 0)$ și $\tilde{X}_3 = (0, 1, 1, 1)$. Între primele două 4-uple există relația $\tilde{X}_1 \geq \tilde{X}_2$, însă între primul și ultimul 4-uplu nu se poate face comparație.

Definiție. O funcție booleană de n argumente se numește *monoton crescătoare* dacă și numai dacă $\tilde{X}_1 \leq \tilde{X}_2$ implică $f(\tilde{X}_1) \leq f(\tilde{X}_2)$, respectiv *monoton descrescătoare* dacă și numai dacă $\tilde{X}_1 \leq \tilde{X}_2$ implică $f(\tilde{X}_1) \geq f(\tilde{X}_2)$.

Exemplu. Să se stabilească dacă funcțiile conjuncție și NICI de două argumente sînt funcții monotone.

x_1	x_2	$f(x_1, x_2) = x_1 x_2$	$f(x_1, x_2) = x_1 \downarrow x_2$
0	0	0	1
0	1	0	0
1	0	0	0
1	1	1	0

Folosind tabelul de adevăr se poate stabili ușor monotonia acestor funcții. Comparînd valorile funcției pentru diferitele combinații ale valorilor argumentelor rezultă că funcția conjuncție este monoton crescătoare, iar funcția NICI monoton descrescătoare. Din exemplul considerat rezultă că funcțiile monotone se pot realiza ușor.

2.5.6. Funcții liniare

Definiție. Funcția booleană $f(x_1, x_2, \dots, x_n)$ se numește *liniară* dacă se poate scrie sub forma :

$$f(x_1, x_2, \dots, x_n) = c_0 \oplus c_1 x_1 \oplus \dots \oplus c_n x_n,$$

unde $c_i \in \{0, 1\}$.

Exemplu. Să se arate că funcția echivalență de două argumente este liniară. Din definiția acestei funcții a rezultat :

$$f(x_1, x_2) = \overline{x_1 \sim x_2} = \overline{(\bar{x}_1 \cup x_2)(x_1 \cup \bar{x}_2)} = \overline{x_1 \oplus x_2}$$

Dar, $\bar{x} = x \oplus 1 = \bar{x} \cdot 1 \cup x \cdot 0 = \bar{x}$, astfel că $f(x_1, x_2) = 1 \oplus x_1 \oplus x_2 = c_0 \oplus c_1 x_1 \oplus c_2 x_2$, unde $c_1 = c_2 = c_0 = 1$.

3. Minimizarea funcțiilor booleene

În capitolul precedent s-a considerat problema reprezentării *FB* prin sisteme complete avînd un număr minim de funcții elementare. Acest lucru vizează posibilitatea folosirii unui număr cît mai redus de tipuri de circuite logice pentru materializarea oricărei *FB*. În continuare se va prezenta și alt aspect al problemei și anume cel care privește utilizarea unui număr cît mai redus de circuite standard. Din punct de vedere teoretic această problemă se reflectă în simplitatea funcțiilor booleene. În acest sens, trebuie menționat că formele canonice ale *FB* sînt în general neeconomice.

Exemplu. Se dă următoarea funcție sub *FCD*: $f(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 x_3 \cup x_1 \bar{x}_2 \bar{x}_3 \cup x_1 \bar{x}_2 x_3 \cup x_1 x_2 \bar{x}_3 \cup x_1 x_2 x_3$. Asociind convenabil termenii și aplicînd apoi proprietatea de distributivitate, se obține :

$$f(x_1, x_2, x_3) = \bar{x}_1 \bar{x}_2 x_3 \cup x_1 \bar{x}_2 (x_3 \cup \bar{x}_3) \cup x_1 x_2 (x_3 \cup \bar{x}_3) = \bar{x}_1 \bar{x}_2 x_3 \cup x_1 (\bar{x}_2 \cup x_2) = \bar{x}_1 \bar{x}_2 x_3 \cup x_1.$$

Aplicînd din nou proprietatea de distributivitate, rezultă :

$$f(x_1, x_2, x_3) = (\bar{x}_1 \cup x_1)(x_1 \cup \bar{x}_2 x_3) = x_1 \cup \bar{x}_2 x_3.$$

Cele două forme mai simple obținute din forma canonică sînt evident mai economice. Dintre ultimile două forme, evident ultima este cea mai simplă, avînd numărul minim de argumente. Din cele prezentate mai sus rezultă faptul că obținerea celei mai simple forme a unei *FB* prin aplicarea proprietăților algebrei Boole depinde de experiența operatorului. Acest fapt a condus la căutarea unor metode sistematice pentru obținerea expresiilor minimale pentru reprezentarea funcțiilor booleene.

Problema simplificării *FB* conduce la problema alegerii sistemului complet și la problema reprezentării cît mai economice

în acest sistem. Pînă în prezent rezultate esențiale s-au obținut numai pentru sistemul complet format din funcțiile conjuncție, disjuncție și negație. Pentru precizarea problemei minimizării FB se consideră forma canonică disjunctivă.

Definiție. Conjuncția $x_1^{a_1} x_2^{a_2} \dots x_k^{a_k} (k < n)$, în care fiecare variabilă se întâlnește numai o singură dată, se numește *termen normal disjunctiv*.

Definiție. Numărul literelor unui termen normal conjunctiv se numește *rangul* termenului normal.

Definiție. Disjuncția termenilor normal conjunctivi se numește *formă normal disjunctivă (FND)*.

Din definițiile date rezultă că FCD a unei funcții de n argumente este forma la care toți termenii sînt de rang n . Deci, forma canonică disjunctivă a unei FB este forma normală cea mai complexă.

Definiție. Forma normal disjunctivă care conține cel mai mic număr de litere $x_i^{a_i}$ în comparație cu toate celelalte FND ale unei funcții date se numește *formă disjunctivă minimă (FDM)*.

Într-un mod analog se pot face definiții similare dacă se pleacă de la forma canonică conjunctivă a unei FB . În continuare se va trata problema minimizării FB plecînd de la FCD , rezultatele putîndu-se extinde și pentru FCC .

După cum s-a arătat în exemplul precedent, o posibilitate de simplificare a unei FB constă în aplicarea proprietăților algebrei Boole cu eliminarea succesivă a variabilelor fictive. Dar, acest procedeu prezintă dezavantajul că nu se știe cu certitudine dacă s-a obținut forma minimă.

În continuare se va prezenta principiul unor metode sistematice de obținere a expresiilor minime pentru FB și care utilizează noțiunea de *implicant prim*.

Definiție. Se numesc *implicanți primi* ai unei funcții booleene de n argumente, termenii conjunctivi de forma $\varphi_k = x_1^{a_1} x_2^{a_2} \dots x_k^{a_k} (k \leq n)$ care implică funcția respectivă fără a se mai putea elimina vreo variabilă. Din definiție rezultă că implicanții primi sînt termeni de rang minim. De exemplu, dacă pentru o funcție de patru argumente $f(x_1, x_2, x_3, x_4)$ au loc relațiile de implicație :

$$x_1 x_2 \bar{x}_3 x_4 \rightarrow f(x_1, x_2, x_3, x_4) ,$$

$$x_2 \bar{x}_3 x_4 \rightarrow f(x_1, x_2, x_3, x_4) ,$$

$$x_2 \bar{x}_3 \rightarrow f(x_1, x_2, x_3, x_4) ,$$

$$x_2 \rightarrow f(x_1, x_2, x_3, x_4),$$

$$\bar{x}_3 \rightarrow f(x_1, x_2, x_3, x_4),$$

atunci $x_2\bar{x}_3$ este un implicanț prim al funcției.

Implicanții primi ai unei FB se obțin plecînd de la FCD și aplicînd sistematic la cîte doi termeni adiacenți identitatea evidentă :

$$(3.1) \quad Ax_i \cup A\bar{x}_i = A$$

Aplicarea identității (3.1) necesită combinarea a cîte doi termeni canonici adiacenți, operație denumită *alipirea parțială* sau *combinare a vecinilor*.

Exemplu. Să se stabilească implicanții primi pentru funcția $f(x_1, x_2, x_3) = x_1x_2\bar{x}_3 \cup \bar{x}_1x_2\bar{x}_3 \cup x_1x_2x_3 \cup \bar{x}_1\bar{x}_2x_3$.

Folosind procedeul mai sus menționat și aplicînd (3.1), rezultă : $x_1x_2\bar{x}_3 \cup \bar{x}_1x_2\bar{x}_3 = x_2\bar{x}_3$, $x_1x_2\bar{x}_3 \cup x_1x_2x_3 = x_1x_2$. Termenul $\bar{x}_1\bar{x}_2x_3$ nu se poate alipi. Astfel implicanții primi ai funcției considerate sînt : $x_2\bar{x}_3$, x_1x_2 , $\bar{x}_1\bar{x}_2x_3$.

Implicanții primi ai unei FB se bucură de următoarele proprietăți. Cînd funcția ia valoarea 0 toți implicanții primi iau valoarea 0 iar cînd funcția ia valoarea 1 cel puțin unul din implicanții primi ia valoarea 1. Dacă unul din implicanții primi ia

Tab. 3.1.

x_1	x_2	x_3	$\bar{x}_2\bar{x}_3$	x_1x_2	$\bar{x}_1\bar{x}_2x_3$	$f(x_1, x_2, x_3)$
0	0	0	0	0	0	0
0	0	1	0	0	1	1
0	1	0	1	0	0	1
0	1	1	0	0	0	0
1	0	0	0	0	0	0
1	0	1	0	0	0	0
1	1	0	1	1	0	1
1	1	1	0	1	0	1

valoarea 1 și funcția ia valoarea 1. Folosind funcția din exemplul precedent se vor verifica proprietățile enumerate. În tabelul 3.1 sînt trecute valorile funcției și ale implicanților săi primi pentru toate combinațiile posibile ale valorilor argumentelor.

Se observă că atunci când $f=0$ toți implicații primi sînt 0 iar când $f=1$ cel puțin unul din implicații primi are valoarea 1. De asemenea, când implicații primi sînt 1 și funcția are valoarea 1.

Este posibil ca în urma alipirilor parțiale și aplicării relației (3.1) să se mai poată aplica implicațiilor rezultați și proprietatea de absorbție :

$$(3.2) \quad A \cup A x_i = A.$$

De mare importanță pentru abordarea sistematică a minimizării FB este :

Teorema lui Quine. Dacă în forma canonică disjunctivă a unei FB se fac toate operațiile de alipire parțială și apoi toate operațiile de absorbție, se obține disjuncția implicațiilor primi.

Demonstrație. Fie un sistem de implicații primi φ_k ai unei funcții de n argumente. Conform teoremei are loc relația :

$$(3.3) \quad f(x_1, x_2, \dots, x_n) = \bigcup_k \varphi_k$$

Relația (3.3) trebuie să fie adevărată atât pentru $f=0$ cât și pentru $f=1$. Atunci când $f=0$, așa cum s-a arătat mai sus, toți implicații primi ai funcției sînt 0, deci și $\bigcup \varphi_k = 0$. Când $f=1$, va exista cel puțin un implicant $\varphi_j=1$, astfel că întreaga disjuncție din partea dreaptă a relației (3.3) va avea valoarea 1.

Relația (3.3) este o formă normal disjunctivă a funcției și se numește *formă disjunctivă prescurtată (FDP)*. Această formă nu este minimă deoarece în general există implicații primi care implică suplimentar funcția. După eliminarea implicațiilor primi redundanți (de prisos), rămîn numai implicații strict necesari care sînt denumiți *implicații esențiali* ai funcției. Disjuncția implicațiilor esențiali conduce la forma disjunctivă minimă. Din cele prezentate rezultă că minimizarea unei FB dată sub FCD comportă două etape :

- 1) determinarea FDP prin căutarea implicațiilor primi,
- 2) căutarea reuniunii minimale care conține cel mai mic număr de implicații primi.

Pentru stabilirea reuniunii minimale se construiește *tabelul implicațiilor primi*, în care fiecare linie corespunde unui implicant prim, iar fiecare coloană unui termen canonic conjunctiv. Corespondențele între termenii canonici și implicații se marchează la intersecția liniilor cu coloanele respective. Se rețin numai acei

implicanți primi necesari să acopere toate coloanele ; aceștia sînt implicanții esențiali.

Exemplu. Să se stabilească forma disjunctivă minimă a funcției $f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2\bar{x}_3 \cup \bar{x}_1\bar{x}_2x_3 \cup \bar{x}_1x_2x_3 \cup x_1\bar{x}_2\bar{x}_3 \cup x_1x_2\bar{x}_3 \cup x_1x_2x_3$.

1) Se caută implicanții primi prin realizarea tuturor operațiilor de alipire parțială a termenilor canonici :

$$\bar{x}_1\bar{x}_2\bar{x}_3 \cup \bar{x}_1\bar{x}_2x_3 = \bar{x}_1\bar{x}_2,$$

$$\bar{x}_1\bar{x}_2\bar{x}_3 \cup x_1\bar{x}_2\bar{x}_3 = \bar{x}_2\bar{x}_3,$$

$$\bar{x}_1\bar{x}_2x_3 \cup \bar{x}_1x_2x_3 = \bar{x}_1x_3,$$

$$\bar{x}_1x_2x_3 \cup x_1x_2x_3 = x_2x_3,$$

$$x_1\bar{x}_2\bar{x}_3 \cup x_1x_2\bar{x}_3 = x_1\bar{x}_3,$$

$$x_1x_2\bar{x}_3 \cup x_1x_2x_3 = x_1x_2.$$

Se continuă cu alipirea termenilor normali de rang 2. Deoarece în cazul de față între aceștia nu se pot face alipiri parțiale rezultă FDP :

$$f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2 \cup \bar{x}_2\bar{x}_3 \cup \bar{x}_1x_3 \cup x_2x_3 \cup x_1\bar{x}_3 \cup x_1x_2.$$

2) Pentru stabilirea numărului minim de implicanți (acoperirea minimă) se construiește tabelul implicanților primi :

Implicanți primi	Termeni canonici					
	$\bar{x}_1\bar{x}_2\bar{x}_3$	$\bar{x}_1\bar{x}_2x_3$	$\bar{x}_1x_2x_3$	$x_1\bar{x}_2\bar{x}_3$	$x_1x_2\bar{x}_3$	$x_1x_2x_3$
$\bar{x}_1\bar{x}_2$	✓	—————	✓			
$\bar{x}_2\bar{x}_3$	✓	-----				✓
\bar{x}_1x_3			✓	-----	✓	
x_2x_3					✓	—————
$x_1\bar{x}_3$					✓	—————
x_1x_2						✓

Din tabel rezultă că fiecare TCC este implicat de câte doi implicați primi. Rezultă astfel două grupuri de implicați esențiali, marcați diferit în tabel și deci două forme disjunctive minime pentru funcția considerată :

$$f_1(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2 \cup x_2x_3 \cup x_1\bar{x}_3 \quad \text{și} \quad f_2(x_1, x_2, x_3) = \bar{x}_2\bar{x}_3 \cup \bar{x}_1x_3 \cup x_1x_2.$$

Din exemplul considerat se poate desprinde concluzia că, în general, o funcție booleană poate avea mai multe forme minime.

Metodologia de stabilire a formei disjunctive minime prezentată mai sus aparține lui Quine și-i poartă numele. În metoda lui Quine este un neajuns determinat de necesitatea comparării complete a perechilor de termeni în prima etapă. Cu creșterea numărului termenilor canonici care definesc FCD a funcției considerate crește numărul acestor comparații. Această creștere este caracterizată de o funcție factorială. Din acest motiv, la un număr suficient de mare de mintermi folosirea metodei Quine devine greoaie. Mc Cluskey a îmbunătățit prima etapă a metodei lui Quine prin transcrierea binară a termenilor canonici. În acest mod se poate face o sistematizare a comparării mintermilor grupându-i după numărul de biți 1. Astfel, în grupa i intră toate numerele corespunzătoare mintermilor care au în transcriere binară i biți 1. Compararea perechilor se poate face numai între grupe vecine, deoarece numai aceste grupe diferă între ele cu un singur bit 1. La reprezentarea termenilor normali rezultați prin eliminarea variabilelor, în locul acestor variabile se trece o linie.

Exemplu. Să se minimizeze funcția de patru argumente

$$f(x_1, x_2, x_3, x_4) = \bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4 \cup \bar{x}_1\bar{x}_2\bar{x}_3x_4 \cup \bar{x}_1\bar{x}_2x_3\bar{x}_4 \cup \bar{x}_1\bar{x}_2x_3x_4 \cup \\ \cup \bar{x}_1x_2\bar{x}_3\bar{x}_4 \cup \bar{x}_1x_2x_3\bar{x}_4 \cup \bar{x}_1x_2x_3x_4 \cup x_1\bar{x}_2\bar{x}_3\bar{x}_4 \cup x_1\bar{x}_2\bar{x}_3x_4 \cup x_1\bar{x}_2x_3\bar{x}_4 \cup \\ \cup x_1x_2x_3x_4.$$

Toți termenii canonici se scriu în cod binar după numărul de biți 1 :

- grupa zero : 0000
- grupa unu : 0001, 0010, 0100, 1000
- grupa doua : 0011, 0110, 1001
- grupa treia : 0111, 1011
- grupa patra : 1111.

Comparând grupele vecine se obțin termenii de rangul trei următori :

- grupa zero : 000—, 00—0, —000
- grupa unu : 00—1, —001, 001—, 0—10, 100—
- grupa doua : 0—11, —011, 011—, 10—1
- grupa treia : —111, 1—11.

Procedînd în același mod se obțin termeni de rangul doi :

- grupa zero : 00— —, —00—, 0— —0
- grupa unu : —0—1, 0—1—
- grupa doua : — —11.

Deoarece nu se mai pot realiza alte alipiri se trece la etapa a doua, construind tabelul implicantilor primi :

	0000	0001	0010	0011	0100	0110	0111	1000	1001	1011	1111
00---	√	√	√	√							
-00-	√	√	-----	-----	-----	-----	-----	√	-----	√	
0--0	√	-----	√	-----	√	-----	√				
-0-1		√		√							
0-1-			√	√		√	√		√	√	
--11			√	-----	-----	-----	√	-----	-----	√	-----

În tabel sînt indicați implicantii esențiali care realizează acoperirea minimală a funcției considerate. Rezultă astfel forma disjunctivă minimă :

$$f(x_1, x_2, x_3, x_4) = \bar{x}_2\bar{x}_3 \cup \bar{x}_1\bar{x}_4 \cup x_3x_4.$$

Completată în modul menționat de către Mc Cluskey metoda de minimizare descrisă este cunoscută ca *metoda Quine-Mc Cluskey*. Minimizarea *FB* prin metoda Quine-Mc Cluskey necesită considerarea funcțiilor sub *FCD*. Problema poate fi tratată și pentru *FB* date sub *FCC*, în care caz operațiile de alipire parțială a termenilor disjunctivi sînt urmate de aplicarea sistematică a formelor duale pentru relațiile (3.1) și (3.2) :

$$(3.4) \quad (A \cup x_i) (A \cup \bar{x}_i) = A \text{ și respectiv } A(A \cup x_i) = A.$$

Etapa a doua, de căutare a intersecției minimale, se realizează similar ca și pentru *FCD*, obținîndu-se în final *forma conjunctivă minimă (FCM)*.

O altă modalitate de obținere a FCM pentru o FB dată prin FCC constă în considerarea funcției negate :

$$(3.5) \quad \overline{f}(x_1, x_2, \dots, x_n) = \overline{\bigcap_0 (x_1^a \cup x_2^a \cup \dots \cup x_n^a)} = \bigcap_0 (\overline{x_1^a} \overline{x_2^a} \dots \overline{x_n^a}),$$

care se poate trata ca și FCD . Se obține astfel forma disjunctivă minimă a funcției negate. Negînd expresia obținută se revine la forma conjunctivă, corespunzînd formei conjunctive minime.

Metoda Quine-Mc Cluskey începe să devină greu de manipulat și pentru un număr redus de variabile ($n \geq 4$) însă prezintă avantaje în sensul realizării unor algoritmi pentru calculul mecanic. De asemenea, principiul metodei expuse poate fi transpus pe diagrame Karnaugh, căpătînd un caracter intuitiv și ușurință în aplicarea pentru funcții avînd 6—7 argumente. Avînd aceleași principii ca și metoda Quine-Mc Cluskey dar mai avantajoasă decît aceasta pentru calcul manual, metoda simbolică va fi prezentată de asemenea în detaliu.

În afara metodelor menționate mai există și alte metode de minimizare a FB , cum ar fi *metoda coeficienților nedeterminați* [34] sau *metoda celui mai mare divizor comun* [29], dar sînt mai greu de manipulat chiar pentru un număr redus de variabile.

3.1. Minimizarea FB folosind diagramele Karnaugh

Spre deosebire de metoda Quine-Mc Cluskey, metoda diagramelor Karnaugh este o *metodă globală*, în sensul că ambele etape din metoda Quine se efectuează simultan. Conform celor arătate în cap. 2, într-o diagramă Karnaugh cîmpurile cu unități corespund mintermilor unei FB iar cîmpurile cu zerouri corespund maxtermilor funcției. Doi termeni canonici conjunctivi plasați în cîmpuri vecine se pot alipi, diagrama Karnaugh asigurînd adiacența acestora. Din cele expuse mai sus rezultă că relațiile (3.1) și (3.2) se pot aplica în diagrama Karnaugh astfel : grupînd două cîmpuri cu unități adiacente se elimină o variabilă (cea care-și schimbă valoarea la trecerea de la un cîmp la celălalt) ; grupînd patru cîmpuri adiacente cu unități se elimină două variabile ; grupînd opt cîmpuri adiacente se elimină trei variabile, ș.a. Deci, pentru eliminarea unui număr cît mai mare de argumente fictive trebuie să se grupeze un număr cît mai mare de cîmpuri adiacente. Un același cîmp (sau mai multe) poate interveni în mai multe grupări, dar pentru ca implicanții corespun-

zătorii reunirilor să fie și esențiali este necesar ca fiecare grupare să aibă cel puțin un câmp propriu.

Exemplu. Să se minimizeze funcția dată în *FCD* folosind diagrama Karnaugh $f(a, b, c, d) = \overline{a}bcd \cup a\overline{b}cd \cup ab\overline{c}d \cup abc\overline{d} \cup \overline{a}bcd \cup \overline{a}bcd \cup \overline{a}bcd \cup \overline{a}bcd \cup \overline{a}bcd \cup \overline{a}bcd \cup \overline{a}bcd \cup \overline{a}bcd \cup \overline{a}bcd$.

În fig. 3.1 este reprezentată diagrama Karnaugh corespunzătoare. Reunind convenabil câmpurile cu unități (cu linie plină) rezultă forma disjunctivă minimă: $f(a, b, c, d) = \overline{a}bc \cup acd \cup bd \cup \overline{a}ad \cup \overline{a}bc$. Există uneori mai multe posibilități de grupare a câmpurilor, rezultând mai multe forme minime. Astfel, pentru funcția considerată se poate face și reunirea marcată cu linie întreruptă, rezultând: $f(a, b, c, d) = \overline{a}bd \cup cd \cup ad \cup abc \cup bcd$. Din punct de vedere al formei minime ambele expresii sînt echivalente deoarece au, conform definiției, același număr de litere.

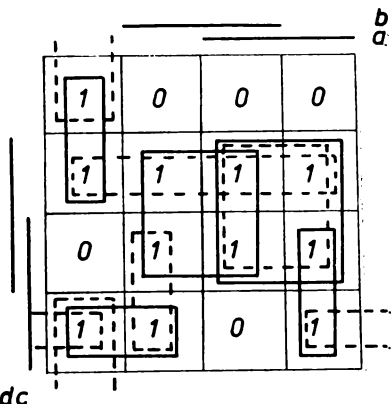


Fig. 3.1

Avînd în vedere relațiile (3.4) și algoritmul de trecere de la reprezentarea tabelară a *FB* la reprezentarea algebrică (v. algoritmul 2.2) se poate transpune în diagramă Karnaugh și minimizarea funcțiilor date prin *FCC*. În acest caz se reunesc în diagrama Karnaugh câmpurile cu zerouri care sînt adiacente, eliminîndu-se variabilele care-și schimbă valoarea între câmpuri. Implicanții esențiali se obțin sub forma termenilor disjunctivi minimali. În final se obține forma conjunctivă minimă realizată cu termeni minimali disjunctivi corespunzători reunirilor câmpurilor adiacente cu zerouri.

Exemplu. Să se minimizeze prin metoda diagramei Karnaugh funcția dată prin *FCC*: $f(x_1, x_2, x_3, x_4) = (x_1 \cup x_2 \cup x_3 \cup x_4) (x_1 \cup \overline{x}_2 \cup x_3 \cup x_4) (\overline{x}_1 \cup x_2 \cup x_3 \cup x_4) (x_1 \cup \overline{x}_2 \cup x_3 \cup \overline{x}_4) (\overline{x}_1 \cup \overline{x}_2 \cup x_3 \cup \overline{x}_4) (\overline{x}_1 \cup x_2 \cup x_3 \cup \overline{x}_4) (x_1 \cup x_2 \cup \overline{x}_3 \cup \overline{x}_4) (x_1 \cup \overline{x}_2 \cup \overline{x}_3 \cup \overline{x}_4) (\overline{x}_1 \cup \overline{x}_2 \cup \overline{x}_3 \cup \overline{x}_4)$.

În fig. 3.2a este reprezentată diagrama Karnaugh pentru funcția considerată. Reunind câmpurile adiacente cu zerouri se

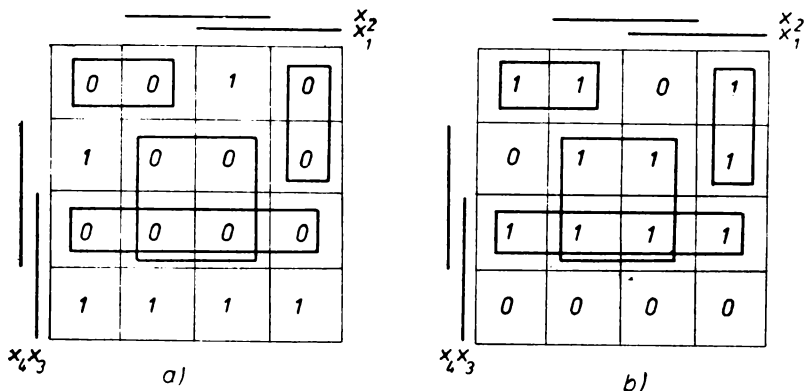


Fig. 3.2

obține următoarea expresie conjunctivă minimă : $f(x_1, x_2, x_3, x_4) = (x_1 \cup x_3 \cup x_4) (\bar{x}_1 \cup x_2 \cup x_3) (\bar{x}_2 \cup \bar{x}_4) (\bar{x}_3 \cup \bar{x}_4)$.

Forma conjunctivă minimă se poate obține și dacă se consideră negata FCC a funcției. Evident, forma conjunctivă canonică negată corespunde diagramei Karnaugh complementate (0 se înlocuiește cu 1 și invers). Aplicînd diagramei complementate metodologia corespunzătoare FCD se obține expresia minimă pentru funcția negată. În fig. 3.2b este prezentată diagrama Karnaugh complementată pentru funcția din exemplul dat. Pentru acest caz rezultă :

$$\bar{f}(x_1, x_2, x_3, x_4) = \bar{x}_1 \bar{x}_3 \bar{x}_4 \cup x_1 \bar{x}_2 \bar{x}_3 \cup x_2 x_4 \cup x_3 x_4.$$

Negînd expresia obținută rezultă

$$f(x_1, x_2, x_3, x_4) = (x_1 \cup x_3 \cup x_4) (\bar{x}_1 \cup x_2 \cup x_3) (\bar{x}_2 \cup \bar{x}_4) (\bar{x}_3 \cup \bar{x}_4).$$

3.1.1. Formele minime implementate cu funcții NICI, NUMAI

Mare importanță practică prezintă obținerea formelor minime ale FB direct implementate cu funcții NICI sau NUMAI. Folosind cele demonstrate în cazul implementării formelor canonice cu funcții Pierce și Sheffer este posibilă exprimarea formelor minime cu funcții NICI sau NUMAI direct din diagrama Karnaugh. Se pot enunța următoarele reguli :

1) Prin reunirea în diagrama Karnaugh a cîmpurilor adiacente cu zerouri se obțin paranteze cu funcții NICI în care se iau variabilele negate. Excepți : fac termenii care se reduc la o singură variabilă, cînd aceasta se ia necomplementată.

2) Prin reunirea cîmpurilor adiacente cu unități ale funcției se obțin paranteze cu funcții NUMAI în care variabilele se iau ca atare. Excepție fac termenii care se reduc la o singură variabilă cînd aceasta se ia negată.

Exemplu. Să se stabilească formele minime implementate cu funcții NICI și NUMAI pentru funcția din exemplul precedent.

În fig. 3.3 este prezentată diagrama Karnaugh a funcției cu reunitirile cîmpurilor adiacente care conduc la forma minimă implementată cu funcții NICI (cu linie plină) sau funcții NUMAI (cu linie întreruptă). Avînd în vedere regulile stabilite, rezultă :

$$f(x_1, x_2, x_3, x_4) = (x_1 \downarrow x_3 \downarrow x_4) \downarrow (\bar{x}_1 \downarrow x_2 \downarrow \bar{x}_3) \downarrow (\bar{x}_2 \downarrow \bar{x}_4) \downarrow (\bar{x}_3 \downarrow \bar{x}_4).$$

$$f(x_1, x_2, x_3, x_4) = (x_1 \uparrow x_2 \uparrow \bar{x}_4) \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3 \uparrow x_4) \uparrow (x_3 \uparrow \bar{x}_4).$$

3.1.2. Minimizarea funcțiilor incomplet definite

Minimizarea funcțiilor incomplet definite are mare importanță deoarece cea mai mare parte a comenzilor discrete conțin situații de nedefinire. Luarea în considerație a stărilor de nedeterminare, prin atribuirea de valori 0 sau 1 funcției, conduce în general la forme minime mai economice decît în cazul neconsiderării lor. În diagrama Karnaugh minimizarea funcțiilor incomplet definite este avantajos de realizat deoarece permite asocierii ușoară a stărilor nedefinite pentru eliminarea a cît mai

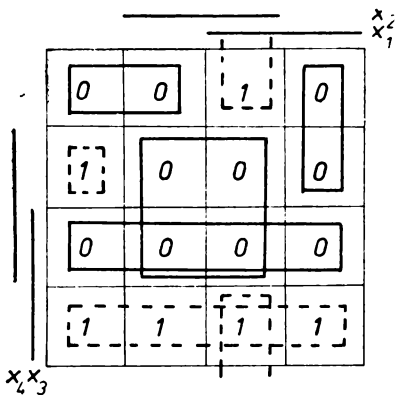


Fig. 3.3

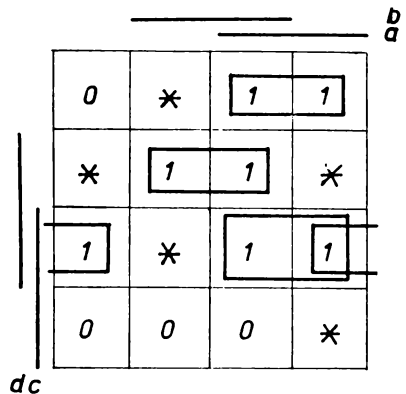


Fig. 3.4

multe variabile. În rest procedura de obținere a formelor minime este identică cu a funcțiilor complet definite.

Exemplu. Să se minimizeze funcția de patru argumente dată prin diagrama Karnaugh din fig. 3.4. Stările nedefinite sînt notate cu asterisc.

În cazul cînd se dorește obținerea FDM și nu se ține cont de situațiile de indiferență, rezultă expresia :

$$f_1(a, b, c, d) = \overline{a}c\overline{d} \cup \overline{b}c\overline{d} \cup acd \cup \overline{b}cd.$$

Dacă însă se adoptă pentru stările indiferente convenabil valoarea 1 se pot face reunirile din fig. 3.5a. În acest caz rezultă :

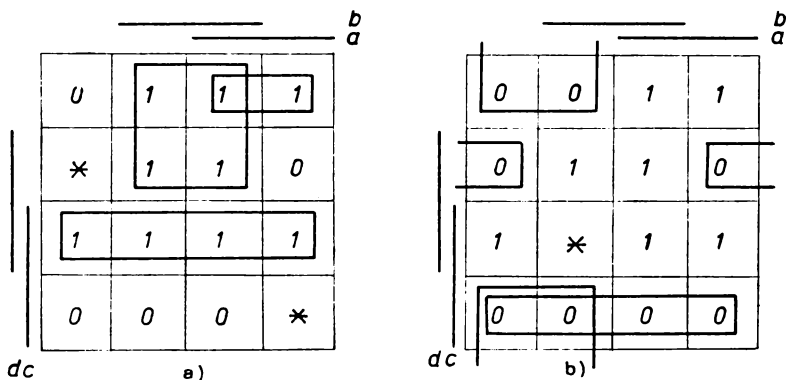


Fig. 3.5

$$f_2(a, b, c, d) = \overline{b}c \cup cd \cup \overline{a}c\overline{d},$$

care evident este mai simplă decît f_1 . Dacă se dorește obținerea formei conjunctive minime se atribuie convenabil valoarea 0 unor situații de nedefinire. În acest caz se obține diagrama din fig. 3.5b și în final

$$f(a, b, c, d) = (a \cup d) (\overline{c} \cup \overline{d}) (b \cup c \cup \overline{d}).$$

Folosirea diferită a stărilor indiferente în obținerea formelor minime disjunctivă și conjunctivă conduce la rezultate diferite. Din acest motiv în cazul funcțiilor incomplet definite este bine să se minimizeze în ambele forme și să se păstreze forma cea mai avantajoasă.

3.2. Minimizarea FB prin metoda simbolică

Deși folosirea diagramelor Karnaugh prezintă avantajele menționate, pentru cazul funcțiilor cu mai mult de 8 argumente nu mai prezintă siguranță în obținerea formei minime.

Metoda simbolică [40] folosește reprezentarea funcțiilor prin simbol de marcare (v. cap. 2), permițând minimizarea sigură și a *FB* de 10—12 argumente. După cum s-a mai menționat metoda simbolică este de tipul metodei Quine-Mc Cluskey; diferă de aceasta prin modul de stabilire a implicanților primi și prin faptul că este tot o metodă globală. În această metodă stabilirea implicanților primi esențiali se realizează prin intermediul *listelor de adiacențe*. Fiecărei liste îi corespunde un implicanț esențial pentru funcția considerată.

Se consideră o *FB* dată prin simbol de marcare *D* în care numerele de stare sînt ordonate în sens crescător. Primul număr de stare va constitui *capul* primei liste de adiacențe. Lista de adiacențe se formează din numerele de stare adiacente cu capul de listă. Numerele de stare fiind în cod octal, două numere de stare vor fi adiacente dacă diferă între ele cu 1, 2, 4, 10, 20, 40, 100, 200, ..., adică în binar diferă între ele printr-un singur bit 1. Întocmirea primei liste de adiacențe se realizează astfel: se caută numărul de stare adiacent cu capul de listă testînd în ordine crescătoare celelalte numere de stare cuprinse în simbolul de marcare; numărul de stare care va corespunde uneia din diferențele de adiacență (1, 2, 4, 10, ...) poate fi alipit cu capul de listă. Prin alipirea celor două numere de stare, care corespund termenilor canonici conjunctivi, se va elimina acea variabilă care corespunde diferenței de adiacență între numerele de stare considerate. Considerînd cele două numere de stare ca fiind n_1 și n_k iar diferența de adiacență dintre ele d_i , în lista de adiacențe

se simbolizează astfel: $n_1 \overset{d_i}{/} n_k$. Pentru a se putea elimina două variabile trebuie să se găsească alte două numere de stare adiacente cu numerele n_1 și n_k prin altă diferență de adiacență d_j . Fie aceste două numere n_m și n_n , lista de adiacențe completîndu-se astfel:

$n_1 \overset{d_i}{/} n_k \overset{d_j}{/} n_m n_n$. Într-un mod similar se testează în ordine crescătoare toate numerele de stare cuprinse în simbolul de marcare, similar reunirii cîmpurilor de unități în diagrama Karnaugh. Pentru a nu fi luat de mai multe ori în formarea listelor de adiacențe,

fiecare număr de stare adiacent cu capul de listă se bifează în simbolul de marcare prin subliniere. Dacă cu capul de listă nu se mai pot face alipiri, lista de adiacențe stabilită formează primul implicanț esențial al funcției. Se consideră în continuare primul număr de stare nebifat în simbolul de marcare; acesta va fi capul celei de a doua liste de adiacențe, ș.a. până la cuprinderea tuturor numerelor de stare în liste de adiacențe. Avînd în vedere corespondența dintre simbolul de marcare D și FCD (v. relația 2.18) pentru fiecare listă de adiacențe stabilită se poate explicita sub formă analitică implicanții esențiali. Pentru aceasta se scrie cîte un simbol de marcare pentru fiecare listă realizată. În locul în care erau scrise numerele de stare în simbol, se trece valoarea în cod binar a capului de listă; fiecare bit fiind trecut în dreptul fiecărui argument începînd cu bitul cel mai puțin semnificativ. Dispar din componența implicanțului acele argumente care corespund diferențelor de adiacență marcate în liste, deoarece diferențele de adiacență corespund rangului argumentelor. În expresia algebrică a implicanților esențiali argumentele intră ca atare sau negate, după cum corespund unui bit 1 sau 0 din desfășurarea binară a capului de listă. Disjuncția implicanților obținuți corespunde formei disjunctive minime.

Exemplu. Să se minimizeze prin metoda simbolică funcția dată prin următorul simbol de marcare:

$$f(a, b, c, d) = D_{1: \underline{0}, \underline{1}, \underline{2}, \underline{5}, \underline{6}, \underline{7}, \underline{11}, \underline{12}, \underline{13}, \underline{15}, \underline{17}}$$

Simbolul de marcare corespunde funcției reprezentată prin diagrama Karnaugh din fig. 3.1.

Listele de adiacențe corespunzătoare sînt următoarele:

$$\begin{aligned}
 & \text{— prima listă} \quad -0 \overset{1}{/} 1 \quad \rightarrow D_{0000}^{abcd} = \overline{abc} \\
 & \text{— lista a doua} \quad -2 \overset{4}{/} 6 \quad \rightarrow D_{0010}^{abcd} = \overline{ac\bar{d}} \\
 & \text{— lista a treia} \quad -5 \overset{2}{/} 7 \overset{10}{/} 15 \quad 17 \rightarrow D_{0101}^{abcd} = b\bar{d} \\
 & \text{— lista a patra} \quad -11 \overset{2}{/} 13 \overset{4}{/} 15 \quad 17 \rightarrow D_{1001}^{abcd} = ad \\
 & \text{— lista a cincea} \quad -12 \overset{1}{/} 13 \quad \rightarrow D_{1010}^{abcd} = \overline{a\bar{b}c}.
 \end{aligned}$$

Forma disjunctivă minimă: $f(a, b, c, d) = \overline{a}bc \cup \overline{a}c\overline{d} \cup bd \cup ad \cup a\overline{b}c$, identică cu cea obținută prin diagrama Karnaugh.

Din exemplul considerat se observă că unele numere de stare se folosesc de mai multe ori în liste diferite, similar cu folosirea unui câmp în mai multe asocieri în diagrama Karnaugh. De asemenea, trebuie menționat faptul că dacă sondările de adiacențe în sens crescător au epuizat numerele de stare, testările se pot face și în sens descrescător.

Prin metoda simbolică este posibilă obținerea și a formei conjunctive minime, plecând de la simbolul C al funcției considerate. În acest caz se întocmesc liste de adiacențe cu numerele de stare aplicate în 0 dar la explicitarea algebrică a implicanților se completează valoarea binară a capului de listă. Acest fapt este justificat de relația de definiție a simbolului C în care numerele de stare se iau complementate (v. relația 2.19).

Exemplu. Să se minimizeze prin metoda simbolică funcția reprezentată prin diagramă Karnaugh în fig. 3.2a. Simbolul de marcare C corespunzător este următorul:

$$f(x_1, x_2, x_3, x_4) = C_{0: \overline{0}, \overline{3}, \overline{4}, \overline{5}, \overline{7}, \overline{10}, \overline{11}, \overline{13}, \overline{15}, \overline{17}}$$

Rezultă următorii implicanți:

$$\begin{array}{ll} -0/4 & \rightarrow C_{1 \underline{1} \underline{1} \underline{1}} = x_1 x_3 x_4 \\ -3/7/13 \ 17 & \rightarrow C_{1 \underline{1} \underline{0} \underline{0}} = \overline{x_3} \overline{x_4} \\ -5/7/15 \ 17 & \rightarrow C_{1 \underline{0} \underline{1} \underline{0}} = \overline{x_2} \overline{x_4} \\ -10/11 & \rightarrow C_{0 \underline{1} \underline{1} \underline{1}} = \overline{x_1} x_2 x_3. \end{array}$$

În final, forma conjunctivă minimă $f(x_1, x_2, x_3, x_4) = (x_1 \cup x_3 \cup x_4)(\overline{x_3} \cup \overline{x_4})(\overline{x_2} \cup \overline{x_4})(\overline{x_1} \cup x_2 \cup x_3)$.

3.2.1. Minimizarea funcțiilor incomplet definite

Cele arătate mai sus pentru minimizarea funcțiilor complet definite se aplică și pentru funcțiile incomplet definite. În plus, la stabilirea listelor de adiacențe se consideră și numerele de stare nedefinite pentru a elimina cât mai multe variabile. Numerele de stare nedefinite nu se vor lua nici odată drept cap de listă.

De asemenea, se caută mai întâi să se cuprindă în liste numerele de stare obligatorii din simbolul de marcare și apoi să se folosească și numerele de stare indiferente.

Exemplu. Să se minimizeze prin metoda simbolică funcția incomplet definită dată prin următorul simbol de marcare :

$$f(a, b, c, d) = D_{\substack{1 : (3, 5, 10, 13, 14, 15, 17) \\ 0 : (0, 2, 6, 11, 16)}^{abcd}}$$

Simbolul de marcare considerat corespunde funcției reprezentată prin diagrama Karnaugh din fig. 3.4.

Listele de adiacențe și implicații corespunzătoare sînt :

$$\begin{aligned} - 3 \diagdown 13 \diagup 7 * 17 & \rightarrow D_{\underline{0011}}^{abcd} = cd \\ - 5 \diagdown 15 \diagup 4 * 14 & \rightarrow D_{\underline{0101}}^{abcd} = \bar{b}c \\ - 10 \diagdown 14 & \rightarrow D_{1000}^{abcd} = \bar{a}cd. \end{aligned}$$

Prin asterisc s-au notat numerele de stare indiferente care nu apar în simbolul de marcare. Ca urmare rezultă forma disjunctivă minimă : $f(a, b, c, d) = cd \cup b\bar{c} \cup a\bar{c}\bar{d}$. Într-un mod similar se poate stabili FCM, considerînd numerele de stare pe care funcția le aplică în 0.

Metoda simbolică devine net avantajoasă pentru calcul manual în raport cu alte metode pentru funcții cu un număr mare de argumente și mai ales în cazul funcțiilor incomplet definite. Pentru a dovedi siguranța obținerii formei minime se va considera ca exemplu o funcție incomplet definită de 9 argumente.

Exemplu. Să se minimizeze funcția dată prin simbol de marcare D :

$$X = D_{\substack{1 : (1, 20, 21, 40, 60, 220, 400, 420) \\ 0 : (0, 11, 50, 100, 102, 110, 111, 140, 142, 160, 250, 300, 304, 320, 340, 344, 3E0)}}^{x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9}$$

Funcția considerată a rezultat în cadrul sintezei unui automat secvențial.

Prin sondarea sistematică a adiacențelor se obține prima listă :

$$\begin{array}{r}
\begin{array}{cccc}
20 & 2 & 4 & 40 \\
-1/21/3*23* & 5*25*7*27* & /41*61*.. & \rightarrow D_{\underline{00000000} \underline{0000} \underline{1}}^{\overline{x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9}} = \overline{x_6 x_9} \\
100 \underline{101*} \dots \\
200 \underline{201*} \\
400 \underline{301*} \\
\underline{401*} \\
501* \\
601* \\
701*
\end{array}
\end{array}$$

Dacă nu s-ar fi considerat și numerele de stare indiferente, s-ar fi eliminat numai o singură variabilă (x_6), deoarece între stările obligatorii de 1 nu se mai pot realiza alipiri. După realizarea adiacențelor posibile cu numerele de stare obligatorii s-a început sondarea pentru diferențele de adiacență rămase: 1, 2, 4, 10, 10, 40, 100, 200, 400. Deoarece în cazul de față cu numerele indiferente se pot realiza foarte multe adiacențe, nu se scriu toate acestea în listă pentru a nu o complica; se verifică însă ca să nu fie vreunul în stările de 0. Astfel, pentru diferența de adiacență 40 s-au trecut numai primele două numere din cele 8 adiacente cu primele 8. De asemenea, pentru a nu se extinde prea mult lista, aceasta se poate dezvolta și pe verticală. Și în acest caz, pentru simplitate, se indică numai primele numere de stare care formează adiacență prin diferențele 100, 200 și 400. Procedînd similar, se mai obțin următorii implicați:

$$\begin{array}{r}
\begin{array}{cccc}
40 & 200 & 400 & 1 \\
-20/60/220 & 260/420 & 460 \dots /21 & 61 \dots \rightarrow D_{\underline{00000100} \underline{0000} \underline{00}}^{\overline{x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9}} = \overline{x_3 x_5} \\
2 \underline{22} \\
4 \underline{24} \\
10 \underline{26} \\
\underline{30} \\
\dots
\end{array} \\
\begin{array}{cccc}
1 & 2 & 4 & 20 \\
-40/41/42 & 43/44 & 45 \dots /60 & 61 \dots \rightarrow D_{\underline{00001000} \underline{0000} \underline{00}}^{\overline{x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9}} = \overline{x_3 x_4 x_6} \\
200 \underline{240} \\
400 \underline{440} \\
\underline{640}
\end{array}
\end{array}$$

$$\begin{array}{r}
 -400 \text{ / } \overset{1}{401} \text{ / } \overset{2}{402} \quad 403 \text{ / } 404 \quad \dots \text{ / } \overset{10}{410} \quad \dots \text{ / } \overset{20}{420} \quad \dots \text{ / } \overset{40}{440} \quad \dots \\
 \hline
 \overset{100}{500} \\
 \hline
 \overset{200}{600} \\
 \hline
 700
 \end{array}
 \rightarrow D_{\underset{1}{0} \underset{0}{0} \underset{0}{0} \underset{0}{0} \underset{0}{0} \underset{0}{0} \underset{0}{0}}^{x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9} = x_1.$$

Rezultă următoarea formă disjunctivă minimă: $X(x_1, x_2, x_3, x_4, \dots, x_9) = x_1 \cup \bar{x}_3 x_5 \cup \bar{x}_6 x_9 \cup \bar{x}_3 x_4 \bar{x}_6$.

3.3. Minimizarea FB date sub forma normală

Minimizarea FB prin metoda Quine-Mc Cluskey precum și prin metodele globale rezultate din această metodă impune cunoașterea funcțiilor sub una din formele canonice. Această necesitate este determinată de posibilitatea omiterii unor implicații primi dacă se pleacă de la o formă normală a unei FB . Rezultă că în cazul FB date sub formă normală, minimizarea lor este posibilă dacă în prealabil sînt aduse la una din formele canonice. Acest lucru este posibil prin operații de dezvoltare a formelor normale în forme canonice. Astfel, pentru aducerea FND a unei funcții la FCD se ține cont de principiul terțului exclus, introducînd argumentele lipsă din termenii normali (care sînt în fapt argumente fictive) prin termeni de forma $x_k \cup \bar{x}_k = 1$. Aplicînd apoi proprietățile algebrei Boole se obțin termenii canonici conjunctivi și deci și FCD .

Exemplu. Să se dezvolte în formă canonică funcția următoare dată sub FND : $f(a, b, c) = ab \cup \bar{b}c$. Introducînd variabilele care lipsesc din fiecare termen normal, se obține: $f(a, b, c) = a\bar{b}(c \cup \bar{c}) \cup (a \cup \bar{a})\bar{b}c = a\bar{b}c \cup a\bar{b}\bar{c} \cup a\bar{b}c \cup \bar{a}\bar{b}c$.

Într-un mod analog se poate dezvolta în formă canonică conjunctivă o FB dată sub FNC . În acest caz argumentele lipsă se introduc prin folosirea principiului contradicției, conform căruia $x_k \bar{x}_k = 0$.

După obținerea formelor canonice minimizarea funcțiilor se realizează după una din metodele prezentate în paragrafele anterioare.

Necesitatea reprezentării FB sub formă canonică devine un neajuns pe măsura creșterii numărului argumentelor. Eliminarea volumului suplimentar de operații implicat de aducerea formelor normale sub forme canonice se poate face dacă se poate construi forma prescurtată plecînd de la forma normală. După

cum au stabilit A. Black și P.S. Porețki acest lucru este posibil și decurge din următoarea leamnă.

Lemă. Dacă în *FND* a funcției $f(x_1, x_2, \dots, x_n)$ intră două conjuncții de forma Ax_i și $B\bar{x}_i$ atunci are loc identitatea

$$(3.6) \quad P = P \cup AB,$$

unde P este *FND* a funcției f .

Demonstrație. Pentru demonstrare este necesar a se arăta că relația (3.6) este adevărată pentru toate n -uplele de zerouri și unități ale argumentelor funcției. Se consideră un n -uplu oarecare pentru care $f(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n) = 1$. În virtutea echivalenței între f și P are loc identitatea $f = 1 \cup AB$ care decurge din (3.6). Evident, egalitatea este adevărată, deoarece disjuncția din partea dreaptă este egală cu 1. Considerînd acum un n -uplu oarecare pentru care $f(\bar{x}_1, \bar{x}_2, \dots, \bar{x}_n) = 0$, în baza relației (3.6) se scrie $0 = 0 \cup AB$. Identitatea poate fi îndeplinită numai cînd $AB = 0$. Astfel, trebuie arătat că pentru n -uplele aplicate în zero sînt nule A sau B sau simultan și A și B . Conform lemei forma normal disjunctivă P are forma: $P = S \cup Ax_i \cup B\bar{x}_i$. Deoarece pentru n -uplul considerat $P = 0$, atunci sînt satisfăcute simultan trei egalități: $S = 0$, $Ax_i = 0$ și $B\bar{x}_i = 0$. Din ultimele două identități decurge că ori A ori B , ori simultan A și B sînt zero deoarece x_i și \bar{x}_i nu pot fi simultan zero.

Din lema demonstrată decurge metodologia construirii *FDP*. Pentru aceasta este necesară completarea *FND* cunoscută cu noi termeni conform relației (3.6). După aceea trebuie realizate operațiile de absorbție și din nou repetată completarea *FND*. Procesul se continuă atît timp cît apar termeni conjunctivi noi; în momentul cînd astfel de termeni noi nu mai apar s-a obținut *FDP*. După obținerea *FDP* se poate folosi metoda lui Quine, începînd cu etapa a doua de căutare a implicanților esențiali.

Exemplu. Să se stabilească *FDP* pentru funcția $f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2 \cup \bar{x}_2x_3 \cup x_1x_2\bar{x}_3 \cup \bar{x}_1x_2\bar{x}_3$, dată sub formă normal disjunctivă.

Perechile de termeni care satisfac lema sînt: $(\bar{x}_1\bar{x}_2, x_1x_2\bar{x}_3)$, $(\bar{x}_1\bar{x}_2, \bar{x}_1x_2\bar{x}_3)$, $(\bar{x}_2x_3, x_1x_2\bar{x}_3)$, $(\bar{x}_2x_3, \bar{x}_1x_2\bar{x}_3)$, $(x_1x_2\bar{x}_3, \bar{x}_1x_2\bar{x}_3)$. Conform lemei se completează *FND* cu termeni de forma AB , renunțînd la cei care devin zero:

$$f(x_1, x_2, x_3) = \overbrace{\bar{x}_1\bar{x}_2 \cup \bar{x}_2x_3 \cup x_1x_2\bar{x}_3 \cup \bar{x}_1x_2\bar{x}_3}^{\text{FND}} \cup \bar{x}_1\bar{x}_3 \cup x_2\bar{x}_2.$$

După simplificări rezultă :

$$f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2 \cup \bar{x}_2x_3 \cup \bar{x}_1\bar{x}_3 \cup x_2\bar{x}_3.$$

În ultima expresie mai există o singură pereche care satisface condițiile lemei ($\bar{x}_1\bar{x}_2, x_2\bar{x}_3$), dar prin care nu se mai obține nimic nou în expresia funcției. Rezultă că expresia obținută este forma disjunctivă prescurtată a funcției.

Continuînd cu etapa a doua din metoda lui Quine se obține următoarea formă disjunctivă minimă a funcției : $f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2 \cup \bar{x}_2x_3 \cup x_2\bar{x}_3$.

Foarte apropiată de metoda Black-Porețki este *metoda consensurilor*, în care simplificarea se bazează nu pe alipirea parțială a termenilor adiacenți ca în metoda Quine-Mc Cluskey ci pe implicația $AB \rightarrow Ax_i \cup B\bar{x}_i$. Implicația menționată se numește *implicația consensului* iar produsul logic AB -*consensul implicanților* Ax_i și $B\bar{x}_i$. Se poate remarca că implicația consensului este o generalizare a relației $A \rightarrow Ax_i \cup A\bar{x}_i$. În adevăr, se poate verifica ușor că pentru $A=B$ implicația consensului se reduce la relația de echivalență $A = Ax_i \cup A\bar{x}_i$. Folosind implicația consensului Tison a realizat o metodă [29, 31] de obținere a *FDP* plecînd de la *FND*, asemănătoare metodei Black-Porețki. Din cele prezentate mai sus rezultă că metoda consensurilor constituie o generalizare a metodei lui Quine.

3.4. Reprezentarea absolut minimală a FB

Pînă acum s-a studiat problema minimizării *FB* prin căutarea unei astfel de reprezentări a funcțiilor pentru care numărul de litere să fie minim. Dar, adesea formele disjunctive sau conjunctive minime nu dau expresii absolut minimale. Astfel, pentru funcția din exemplul precedent s-a obținut *FDM* sub forma : $f(x_1, x_2, x_3) = \bar{x}_1\bar{x}_2 \cup \bar{x}_2x_3 \cup x_2\bar{x}_3$. Dacă se scoate în factor \bar{x}_2 se obține $f(x_1, x_2, x_3) = x_2(x_1 \cup x_3) \cup \bar{x}_2x_3$, adică o expresie care conține 5 litere în loc de 6 și conform definiției este reprezentarea minimă. Apare astfel problema găsirii reprezentării absolut minimale prin realizarea de scoateri în factor în formele minimale.

Definiție. Q constituie o *reprezentare absolut minimală* pentru funcția f dacă în sistemul complet ($\cup, \cap, -$) nu există o reprezentare mai simplă decît Q .

Problema stabilirii expresiei absolut minimale prin toate scoaterile în factor în *FDM* a fost studiată de Abhyankar,

Burthart și alții. În 1952 Burthart a arătat că există o expresie absolut minimală dar care nu poate fi obținută pe baza operației de scoatere în factor în FDM . De asemenea, Abhyankar a dat un algoritm al aflării expresiei absolut minimale pentru o FB dată, dar necesită un număr excesiv de operații. Mai practică este obținerea reprezentării celei mai simple prin scoaterea în factor în formele minime obținute în sistemul complet $(\cup, \cap, -)$.

•

Partea II

CIRCUITE LOGICE

Circuitele logice constituie modelul fizic al funcțiilor logice bivalente. Orice element fizic, organ sau sistem care în funcționarea sa poate avea două stări distincte sub acțiunea unei comenzi exterioare este capabil, în principiu, să materializeze variabile sau funcții boolene. În funcționarea lor, elementele capabile de o funcționare binară pot implica intervenția unor fenomene de naturi diferite: mecanice, electrice, magnetice, electronice, pneumatice, criogenice etc.

Elementele fizice care stau la baza realizării circuitelor logice se pot împărți în două categorii:

- elemente dinamice sau cu piese în mișcare,
- elemente statice sau fără piese în mișcare.

Prima categorie este definită în special de contactele releelor electromagnetice, dar se mai pot aminti și alte elemente cum ar fi elementele pneumatice cu piese în mișcare. Dacă stării unui contact (închis sau deschis) sau obturării sau nu a unui orificiu prin care circulă un fluid li se asociază valorile binare 0 sau 1, funcționarea lor permite materializarea variabilelor și funcțiilor binare. Deoarece prin funcționarea elementelor dinamice se conectează sau se deconectează un circuit, electric sau fluidic, se spune că aceste elemente realizează o *logică de conexiuni*.

Elementele din a doua categorie, datorită avantajelor pe care le prezintă în raport cu primele, au căpătat o utilizare foarte mare în domeniul circuitelor logice. Din această clasă fac parte: dispozitivele electronice, elementele pneumatice fără piese în mișcare, materialele magnetice cu ciclu de histerzis dreptunghiular, elemente care folosesc tehnica frigului foarte înaintat (criotroane), elemente optoelectronice (optroni) etc. Elementelor din această categorie, care în funcționare se

bazează pe fenomene electrice, li se asociază cifrele binare 0 și 1 fie nivelelor de tensiune, fie semnalelor sub formă de impulsuri. În cazul nivelelor de tensiune se atribuie convențional valoarea 0 unui nivel de tensiune în jurul valorii de zero volți iar valoarea 1 altui nivel în valoare absolută distinct față de primul. În cazul semnalelor sub formă de impulsuri, absenței impulsului i se atribuie cifra 0, iar prezenței acestuia cifra 1. Elementele statice care au ca agent de lucru un fluid li se asociază valorile 0 și 1 nivelelor de presiune coborâtă sau nulă și respectiv unor nivele de presiune distincte de primele. Se spune că elementele statice cu acțiune discretă funcționează cu o *logică de nivele*.

4. Circuite logice cu contacte de rele

Circuitele logice realizate cu contacte de rele au fost primele tipuri și mult timp au fost singurele. Referitor la aceste circuite s-a dezvoltat și o teorie matematică puternică la care cercetătorii români și-au adus o importantă contribuție [24, 26].

Deși prezintă multe dezavantaje în raport cu elementele statice (piese în mișcare, durată redusă de funcționare, viteză mică de comutare, gabarite mari, întreținere periodică) aceste elemente se utilizează încă suficient de mult datorită unei serii de avantaje (număr mare de contacte acționate simultan, robustețe la suprasarcini). Faptul că diferiți constructori fac eforturi, pe de o parte să reducă dimensiunile, iar pe de altă parte să le mărească siguranța și longevitatea (rele miniaturizate cu contacte în vid, tip „trestie” cu eliminarea vibrației contactelor etc.) confirmă utilizarea și în viitor a acestor elemente la realizarea comenzilor de complexitate mică și medie.

Cea mai utilizată convenție de atașare a cifrelor binare 0 și 1 celor două stări distincte a unor contacte de releu este următoarea : când înfășurarea releului este alimentată contactelor închise li se atribuie cifra 1 iar celor deschise cifra 0 ; deci în stare de repaus contactelor normal închise (n.î.) cifra 0 iar contactelor normal deschise (n.d) cifra 1. În fig. 4.1 sînt prezentate circuitele logice care materializează sistemul complet de funcții SI, SAU, NU, împreună cu reprezentările simbolice mai mult utilizate. În mod curent înfășurării releului i se atașează o literă mare iar contactelor sale minusculele corespunzătoare.

Evident, folosind circuitele prezentate se poate materializa orice funcție logică, deci orice circuit logic în care nu intervine timpul (combi-național). Folosirea releelor temporizate sau polari-zate permite introducerea timpului în prelucrarea logică, obți-

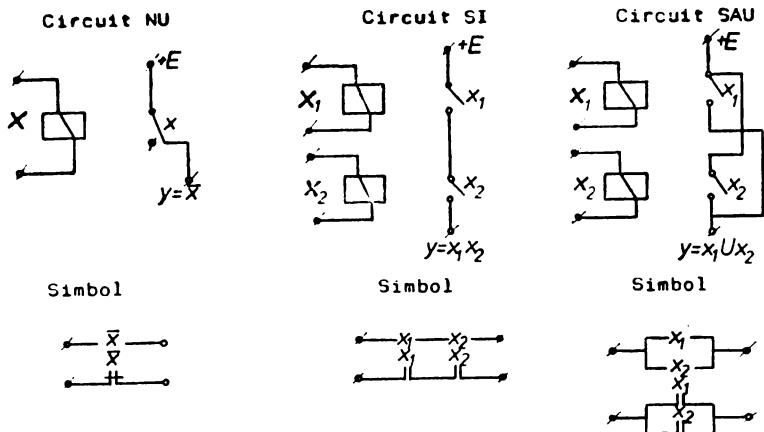


Fig. 4.1

nînd astfel posibilitatea realizării cu contacte și a circuitelor secvențiale.

Simplitatea deosebită în realizarea circuitelor logice cu contacte este împietată de un mare neajuns și anume că nu sînt *unidirecționale*, fapt ce permite realizarea unor legături false. Se consideră, pentru exemplificare, circuitul cu contacte din Fig. 4.2a. Acest circuit materializează funcțiile: $y_1 = x_1 x_2 x_3 \cup x_4$ și $y_2 = x_1 x_2 \cup \bar{x}_2 \bar{x}_3$, folosind în comun contactele x_1 și x_2 . Dar, datorită bidirecționalității în transmiterea informației calea $x_3 - x_4$ realizează o legătură falsă pentru y_2 . În cazul de față

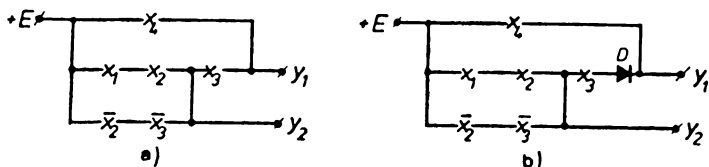


Fig. 4.2

neajunsul se poate elimina prin folosirea a două contacte în plus (x_1 și x_2) sau prin utilizarea diodelor semiconductoare (fig. 4.2b). Deci, la realizarea comenzilor cu astfel de elemente sînt necesare precauții pentru a evita legăturile false.

5. Circuite cu dispozitive semiconductoare

Posibilitatea dispozitivelor semiconductoare de a funcționa în regim de comutație, cu două stări stabile, le conferă acestora proprietatea de element cu acțiune discretă apt de a materializa funcții logice.

După cum s-a menționat, în cazul acestor elemente se folosește logica de nivele: unui nivel de tensiune ridicat în valoarea absolută i se atribuie cifra binară 1 iar unui nivel coborît în valoare absolută cifra 0 (se poate aplica și convenția inversă). Pentru a ține cont de imperfecțiunile montajelor precum și de variația parametrilor celor două nivele, se acordă toleranțe acestor nivele (fig. 5.1). Nivelul înalt mai este notat cu litera H (high) și respectiv cel coborît cu L (low). De asemenea, se utilizează noțiunile de *logică pozitivă* și *logică negativă*. În primul caz cifra binară 1 se atașează celui mai pozitiv nivel de tensiune, iar în al doilea caz celui mai negativ nivel de tensiune. Cifra binară 0 se atașează unui nivel pozitiv apropiat de zero volți pentru logica pozitivă și unui nivel negativ apropiat de zero volți pentru logica negativă.

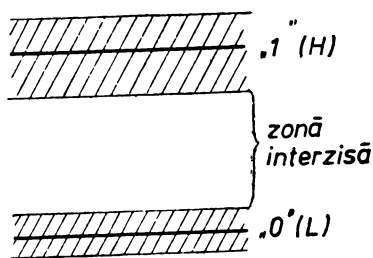


Fig. 5.1

5.1. Circuite logice cu diode semiconductoare

Neliniaritatea caracteristicii volt-amperice (fig. 5.2) permite diodelor semiconductoare o funcționare discretă în regim de comutare. Comutarea se realizează între regiunea de conducție directă A și de conducție inversă B . Deși asigură parametri mai buni, comutația între regiunile B și C nu se utilizează datorită pericolului străpungerii joncțiunii în regiunea Zener (porțiunea C).

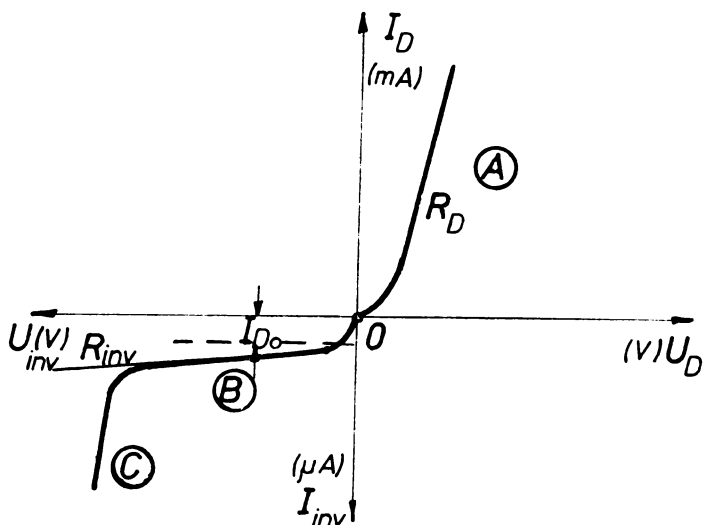


Fig. 5.2

5.1.1. Circuit elementar cu diodă semiconductoră

În fig. 5.3a este prezentată schema unui circuit elementar cu diodă semiconductoră comandat de un generator de rezistență internă R_g . Semnalul de comandă $e(t)$ se modifică în salt (fig. 5.3b) între două nivele — unul ridicat E_1 și unul coborât E_2 .

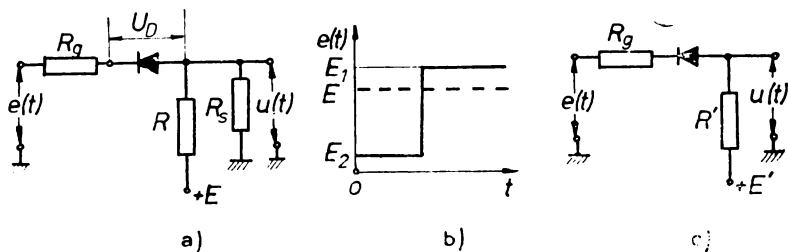


Fig. 5.3

E_2 . Polarizarea diodei se face prin rezistorul R de la sursa E a cărei valoare se poate afla, în general, între nivelele de comandă E_1 și E_2 . Dacă circuitul are sarcina R_s , schema echivalentă este

mai comodă pentru studiu (fig. 5.3. c). Conform teoremei generatorului echivalent

$$E' = \frac{R_s}{R_s + R} \bar{E} \text{ și } R' = R \parallel R_s = \frac{RR_s}{R + R_s}.$$

Pentru început se consideră R_D neglijabilă, deci nivelele E_1 și E_2 nu depind de sarcină.

Regimul staționar. În acest regim dioda poate fi polarizată direct sau invers. La o *polarizare inversă* ($u_D < 0$) prin diodă trece curentul invers I_{inv} care are două componente: curentul termic I_{D_0} determinat de generarea termică de purtători și curentul de scăpări I_s condiționat de rezistența de scăpări și care depinde de valoarea tensiunii inverse. Curentul I_{D_0} depinde de temperatură și se poate considera că se dublează la o creștere a temperaturii cu 10°C , iar I_s practic nu depinde de temperatură. La tensiuni inverse mici I_s este neînsemnat pentru diodele cu Ge și $I_{inv} \approx I_{D_0}$; la diodele cu Si de obicei I_{D_0} este neglijabil și se poate considera $I_{inv} \approx I_s$. Dar, în general

$$I_{inv} = I_{D_0} + I_s,$$

sau dacă se liniarizează caracteristica inversă a diodei (v. fig. 5.2).

$$(5.1) \quad I_{inv} = I_{D_0} + \frac{U_{inv}}{R_{inv}},$$

unde R_{inv} este panta dreptei care aproximează caracteristica.

La o *polarizare directă* ($u_D > 0$) prin diodă trece curentul direct I_D . În acest caz dioda se poate echivala prin rezistența directă R_D care caracterizează panta medie a ramurii din cadranul I.

Analiza funcționării în regim staționar

În studiul comportării circuitului elementar în regim staționar interesează valorile semnalului la ieșirea circuitului. Acestea depind de nivelele semnalului de comandă, E_1 și E_2 , și de raportul între acestea și tensiunea de polarizare.

a) Sub acțiunea semnalului E_2 dioda este în conducție. În acest caz schema echivalentă este dată în fig. 5.4a. La ieșire se obține nivelul coborât U_2 . Corespunzător schemei echivalente se pot scrie relațiile:

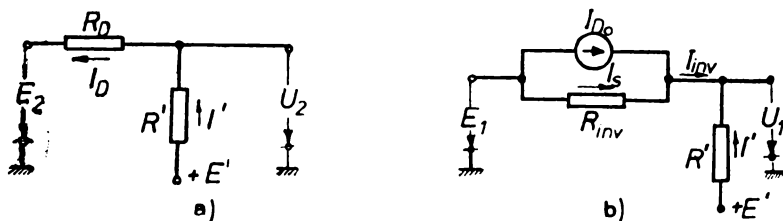


Fig. 5.4

$$\begin{cases} E' = U_2 + R'I' \\ E_2 = U_2 - R_D I_D \end{cases} \text{ sau } \begin{cases} U_2 G_D = E_2 G_D + I_D \\ U_2 G' = E' G' - I' \end{cases}$$

de unde :

$$U_2 = \frac{E_2 G_D + E' G'}{G_D + G'}, \text{ cu } G_D = \frac{1}{R_D} \text{ și } G' = \frac{1}{R'}$$

Înlocuind conductanțele cu rezistențele corespunzătoare, rezultă :

$$U_2 = \frac{E_2}{\frac{R_D}{R'} + 1} + \frac{E'}{\frac{R_D}{R'} + 1}$$

Dacă se ține cont de R_g , atunci în (5.2) în loc de R_D se introduc $R'_D = R_D + R_g$. Dacă $R \gg R_D$, din (5.2) rezultă $U_2 \cong E_2$.

b) Sub acțiunea semnalului E_1 , în funcție de valoarea sa față de E' , sînt posibile trei situații distincte :

1) Dacă $E_1 > E'$ dioda este blocată. Avînd în vedere relația (5.1) schema echivalentă a circuitului este cea din fig. 5.4b. Notînd semnalul la ieșire cu U_1 , valoarea acestuia rezultă din următoarele ecuații :

$$\begin{cases} U_1 = E_1 - R_{inv} I_s \\ U_1 = E' - R'I' \\ I_{inv} = I_{D0} - I_s \end{cases}$$

de unde :

$$(5.3) \quad U_1 = \frac{E_1}{\frac{R_{inv}}{R'} + 1} + \frac{E' + I_{D0} R'}{\frac{R_{inv}}{R'} + 1}$$

În acest caz evident $R_g \ll R_{inv}$ (obișnuit $R_{inv} \geq 0,1 - 1 \text{ M}\Omega$). Dacă $R' \ll R_{inv}$ atunci

$$U_1 \cong E' + I_{D_0} R'.$$

Deci, nivelul U_1 depinde de temperatură prin I_{D_0} . Pentru a nu exista această dependență este necesar să se asigure $R' I_{D_{0max}} \ll E'$, unde $I_{D_{0max}}$ este valoarea componentei termice a curentului invers la temperatura maximă de funcționare. La îndeplinirea acestei condiții $U_1 \cong E'$.

2) Când $E_1 < E'$, dioda este în conducție și nivelul semnalului la ieșire se determină similar cazului a) :

$$U_1 = \frac{E_1}{\frac{R_D}{R'} + 1} + \frac{E'}{\frac{R'}{R_D} + 1}.$$

Deoarece în mod obișnuit $R_D \ll R'$ rezultă că $U_1 \cong E_1$. Dacă se ia în considerare și R_g se înlocuiește în relația de mai sus R_D cu $R'_D = R_D + R_g$.

3) Dacă $E_1 = E'$ dioda este blocată și curentul prin ea este nul. Semnalul la ieșire are valoarea $U_1 = E_1 = E'$.

Din punct de vedere al regimului staționar, cea mai avantajoasă situație este ultima deoarece nu se solicită curent de la generatorul de comandă.

Regimul tranzitoriu. Acest regim are loc la comutarea circuitului dintr-o stare staționară în cealaltă și este condiționat de :

— inerția diodei, caracterizată prin : capacitatea joncțiunii (C_j), constantele de timp de stocare (τ_s) și de blocare (τ_{bl}) care determină duratele de stabilire a rezistenței directe și de restabilire a rezistenței inverse ;

— capacitatea sarcinii, C_s ;

— capacitatea montajului, C_m .

La actualele diode de comutație influența timpilor de stabilire și restabilire a rezistențelor directă și inversă este mică, valoarea acestora nedepășind $0,1 \mu\text{sec}$.

Din acest motiv, la studiul regimului tranzitoriu se poate considera schema echivalentă din fig. 5.5 unde $C = C_j + C_s + C_m$.

Dacă la momentul t_1 se aplică la intrarea circuitului din fig. 5.5 o

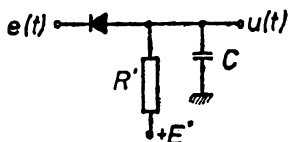


Fig. 5.5

treaptă de tensiune $E_1 > E'$ (fig. 5.6 a) de la un generator ideal ($R_g = 0$) și se consideră că dioda se blochează instantaneu, semnalul la ieșirea circuitului crește exponențial cu constanta de timp $\tau_{cr} = C(R' \parallel R_{inv}) \cong CR'$. În decursul unui timp de cca. 3–4 constante de timp: $t_{cr} \cong (3-4)CR'$, procesul tranzitoriu se

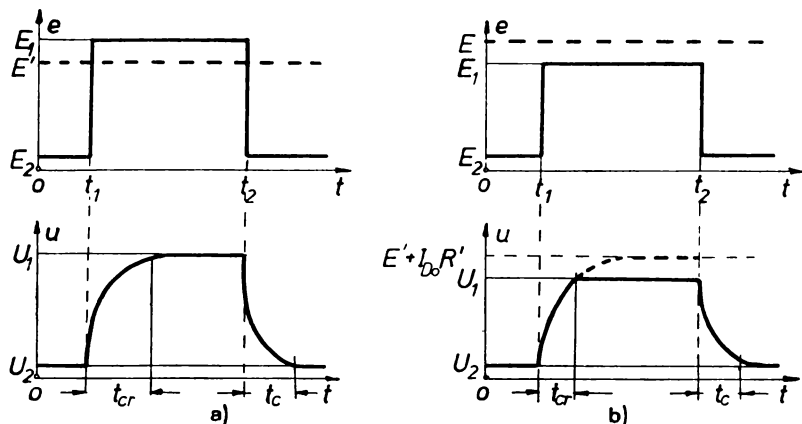


Fig. 5.6

stabilizează, la ieșire obținându-se valoarea $U_1 \cong E' + I_{D0}R'$. La aplicarea în momentul t_2 a unei trepte negative, prin revenirea semnalului de comandă la E_2 , capacitatea C se descarcă cu constanta de timp $\tau_c = C(R' \parallel R_D) \cong CR_D$. În decursul unui interval de timp $t_c \cong (3-4)\tau_c$, semnalul la ieșire atinge nivelul $U_2 \cong E_2$.

Dacă la momentul t_1 circuitului comandat cu semnal E_2 i se aplică o treaptă de tensiune cu $E_1 < E'$ (fig. 5.6 b), în primul moment dioda se blochează și condensatorul C se încarcă. Semnalul la ieșire tinde exponențial către valoarea $E' + I_{D0}R'$. Când $u(t)$ devine aproximativ egală cu E_1 dioda se deschide și procesul tranzitoriu practic încetează, semnalul la ieșire fixându-se la nivelul $U_1 \cong E_1$. La aplicarea în momentul t_2 a unei trepte negative procesul tranzitoriu decurge ca și în cazul precedent când după timpul t_c , $U_2 \cong E_2$.

Din cele prezentate rezultă că din punct de vedere al regimului tranzitoriu cea mai avantajoasă situație este când $E_1 < E'$ deoarece conduce la micșorarea timpului de creștere t_{cr} care este predominant (evident $t_{cr} > t_c$).

5.1.2. Circuite logice cu diode semiconductoare cu mai multe intrări

Se consideră un circuit cu n intrări a cărui schemă este prezentată în fig. 5.7. În această schemă s-au considerat direct parametrii echivalenți ai circuitului de polarizare și sarcinii, E' și R' . Semnalele de intrare e_i se modifică între două nivele: unul coborât E_2 și unul ridicat E_1 . De asemenea, se menționează

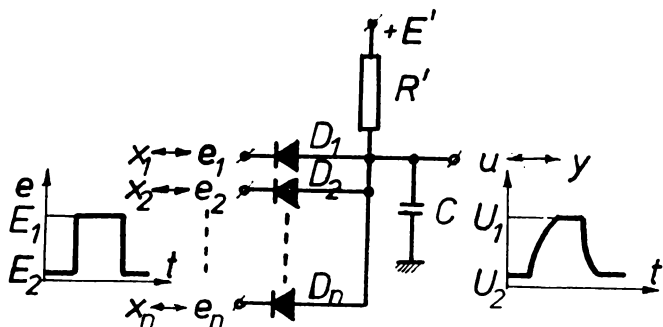


Fig. 5.7

că $E' > E_2$. Dacă la toate intrările se aplică semnale înalte E_1 , semnalul la ieșire va fi înalt, U_1 . Dacă însă la o singură intrare se aplică semnal coborât E_2 , semnalul la ieșire va fi de asemenea coborât și apropiat de E_2 ; dioda în conducție va polariza invers celelalte $n-1$ diode. Este evident că la ieșire se va obține semnal coborât și în cazurile în care nu numai la una ci la $k > 1$ sau la toate cele n intrări se aplică semnale coborâte E_2 .

Atașind semnalelor de intrare e_i argumentele binare x_i iar semnalului de ieșire u mărimea binară y , din descrierea funcționării circuitului rezultă că acesta materializează în logică pozitivă funcția booleană SI de n argumente:

$$y = x_1 x_2 \dots x_n.$$

Dacă s-ar accepta convenția inversă: valoarea 1 atașată celui mai coborât semnal pozitiv, schema din fig. 5.7 ar materializa funcția SAU de n argumente.

Dacă se inversează diodele și sursa de polarizare va fi negativă, cu $-E' < -E_2$, circuitul obținut va realiza funcția SI în logică negativă când $-E_1 < -E_2$. De asemenea, circuitul din fig. 5.7 cu diodele inversate și cu sursa de polarizare negativă

constituie un circuit SAU cu n intrări în logică pozitivă ($-E' < < E_2 < E_1$). Ultima situație se folosește mult la realizarea circuitelor industriale cu alimentare de la surse duble.

Circuitul din fig. 5.7 poate funcționa atât cu semnale sub formă de nivele de tensiune cât și sub formă de impulsuri. În ultimul caz, când este denumit și *circuit de coincidență* datorită faptului că la ieșire se obține semnal 1 numai la coincidența impulsurilor la intrare, devine esențială luarea în considerare a proceselor tranzitorii. Procesele tranzitorii determină și viteza de acționare a schemei.

Analiza regimului staționar al circuitului



Se consideră cazul când circuitul din fig. 5.7 funcționează ca circuit SI în logică pozitivă. Se va analiza cazul general când la $k < n$ intrări acționează semnale E_1 iar la $r = n - k$ semnale coborâte E_2 . Evident, diodele din cele r ramuri vor fi deschise polarizând invers diodele din celelalte k ramuri. Drept urmare la ieșire se va obține un semnal coborât. Considerând diodele blocate de pe cele k intrări definite de curentul și rezistența inversă, se pot scrie următoarele ecuații :

$$\begin{cases} kUG_{inv} = kE_1G_{inv} - kI_{inv} \\ rUG_D = rE_2G_D + rI_D \\ UG' = E'G' - I' \end{cases}$$

unde $G_{inv} = 1/R_{inv}$ $G_D = 1/R_D$ și $G' = 1/R'$.

Din sistemul de ecuații rezultă :

$$(5.4) \quad U(k, r) = \frac{E' + k \frac{R'}{R_{inv}} E_1 + r \frac{R'}{R_D} E_2}{1 + k \frac{R'}{R_{inv}} + r \frac{R'}{R_D}}$$

Dacă se ține cont de rezistența internă a generatoarelor de semnal în (5.4) se înlocuiește R_D cu $R'_D = R_D + R_g$.

Folosind relația (5.4) se pot stabili nivelele semnalului la ieșire pentru diferite regimuri posibile. În funcție de raportul dintre E' și E_1 sînt posibile următoarele trei regimuri staționare :

- a) Dacă $E_1 > E'$, în cele mai importante cazuri se obține :
 - 1) când toate diodele sînt în conducție ($k=0$, $r=n$)

$$U(0, n) = U_{2min} = \frac{E' + n \frac{R'}{R_D} E_2}{1 + n \frac{R'}{R_D}}$$

2) pentru o singură diodă în conducție ($k=n-1, r=1$)

$$U(n-1, 1) = U_{2max} = \frac{E' + (n-1) \frac{R'}{R_{inv}} E_1 + \frac{R'}{R_D} E_2}{1 + (n-1) \frac{R'}{R_{inv}} + \frac{R'}{R_D}};$$

3) cînd toate diodele sînt blocate ($k=n, r=0$)

$$U(n, 0) = (U_1)_a = \frac{E' + n \frac{R'}{R_{inv}} E_1}{1 + n \frac{R'}{R_{inv}}}$$

Dacă $R' \gg R_D(R'_D)$ și $nR' \ll R_{inv}$, se obține :

$$U(0, n) = U_{2min} \cong E_2; \quad U(n-1, 1) \cong U_{2max} \cong E_2;$$

$$U(n, 0) = (U_1)_a \cong E'.$$

b) Pentru cazul cînd $E_1 < E'$ relațiile stabilite mai sus pentru situațiile 1) și 2) se păstrează însă pentru cazul 3) nu, deoarece și în acest caz diodele sînt deschise. Pentru ultima situație se obține o relație analoagă cu cea din cazul 1) înlocuind însă pe E_2 cu E_1 :

$$U(n, 0) = (U_1)_b = \frac{E' + n \frac{R'}{R_D} E_1}{1 + n \frac{R'}{R_D}}$$

și care pentru $R_D(R'_D) \ll R'$ devine $U(n, 0) = (U_1)_b \cong E_1$.

c) În cazul cînd $E_1 = E'$ rămîn valabile relațiile din cazul precedent întrucît pentru cazul 3) se obține $U(n, 0) = (U_1)_c = E' = E_1$.

Cunoscînd nivelele la ieșirea circuitului, se poate stabili pentru fiecare caz în parte valoarea saltului la ieșire, $U_{se} = U_1 - U_2$, parametru important în special pentru circuitul de coinci-

dență. Acest salt trebuie să fie cât mai mare pentru a nu fi influențat de necoincidența semnalelor la intrare. Valoarea saltului la ieșire în condițiile cele mai dezavantajoase (cînd $U_2 = U_{2max}$) este:

$$U_{se} = U(n; 0) - U(n-1, 1).$$

Înlocuind expresiile semnalelor la ieșire, se obține :

$$(U_{se})_a = \frac{E' + n \frac{R'}{R_{inv}} E_1}{1 + n \frac{R'}{R_{inv}}} - \frac{E' + \frac{R'}{R_D} E_2 + (n-1) \frac{R'}{R_{inv}} E_1}{1 + \frac{R'}{R_D} + (n-1) \frac{R'}{R_{inv}}}$$

și

$$(U_{se})_{b,c} = \frac{E' + n \frac{R'}{R_D} E_1}{1 + n \frac{R'}{R_D}} - \frac{E' + \frac{R'}{R_D} E_2 + (n-1) \frac{R'}{R_{inv}} E_1}{1 + \frac{R'}{R_D} + (n-1) \frac{R'}{R_{inv}}}$$

Dacă se păstrează constant E_1 și se modifică E' , pentru $R_{inv} \gg \gg R_D$ se poate verifica ușor că $(U_{se})_a = f_1(E')$, $E' < E_1$ este monoton crescătoare iar $(U_{se})_b = f_2(E')$, $E' > E_1$ este monoton descrescătoare. Pentru $E' = E_1$ ambele funcții sînt egale și ca urmare în regimul c) la ieșire se obține semnalul maxim. Pentru acest regim rezultă :

$$(U_{se})_c = (U_{se})_{max} = \frac{E_1 + n \frac{R'}{R_D} E_1}{1 + n \frac{R'}{R_D}} - \frac{E_1 + \frac{R'}{R_D} E_2 + (n-1) \frac{R'}{R_{inv}} E_1}{1 + \frac{R'}{R_D} + (n-1) \frac{R'}{R_{inv}}}$$

sau după operații simple

$$(5.5) \quad U_{se \max} = \frac{U_{si}}{1 + \frac{R_D}{R'} + (n-1) \frac{R_D}{R_{inv}}},$$

unde $U_{si} = E_1 - E_2$ este saltul la intrare.

Relația (5.5) permite evaluarea numărului maxim de intrări ale circuitului astfel ca acesta să nu influențeze valoarea saltului la ieșire :

$$(n-1) \frac{R_D}{R_{inv}} < 1, \text{ de unde } n_{max} \leq \frac{R_{inv}}{R_D} + 1$$

Curenții absorbiți de la sursa de polarizare și de la generatoarele de comandă sînt de asemenea parametri care interesează în calculul acestor circuite. Curentul minim care trece prin R' este atunci cînd tensiunea la ieșire are valoarea U_1 :

$$I'_{min.E} = \frac{E' - U_1}{R'} \text{ sau } I_{min.E} = I'_{min.E} \frac{R_s}{R + R_s} = \frac{E' - U_1}{R}.$$

Curentul maxim solicitat de la sursa E va fi atunci cînd $U = U_{2min}$:

$$I_{max.E} = \frac{E' - U_{2min}}{R}.$$

Solicitarea maximă a generatoarelor de comandă este atunci cînd numai o singură intrare este acționată cu semnal coborît, deci pentru o singură diodă deschisă — $U(n-1, 1)$. De la generatorul care comandă intrarea respectivă se absoarbe un curent egal cu suma curenților care circulă prin R' și prin celelalte $(n-1)$ diode blocate:

$$I_{gmax} = \frac{E' - U_{2max}}{R} + (n-1)I_{inv.max}.$$

Curentul $I_{inv.max}$ pentru diodă este pentru tensiunea inversă maximă $E_1 - U_{2max}$ și pentru temperatura maximă de lucru.

Analiza regimului tranzitoriu

Regimul tranzitoriu este similar celui prezentat pentru circuitul elementar. Se consideră de asemenea că rezistențele directă și inversă ale diodelor se stabilesc instantaneu. În acest caz procesele tranzitorii sînt influențate de capacitățile joncțiunilor diodelor, sarcinii și montajului și se iau în considerare printr-o capacitate totală C la ieșire (fig. 5.7).

Se consideră inițial că la toate intrările circuitului acționează semnale ridicate $E_1 > E'$, astfel că la ieșirea circuitului SI se obține semnal ridicat U_1 . Dacă la un moment dat la o intrare se modifică semnalul în salt pînă la nivelul coborît E_2 , semnalul la ieșire scade exponențial pînă la nivelul $U_2 \approx E_2$. Dioda comandată cu semnal 0 se va deschide și celelalte $n-1$ diode vor fi

polarizate invers. Timpul de stabilire a nivelului 0 logic va fi $t_c = (3-4)CR_{ech}$, unde $R_{ech} = R' \parallel R_D \parallel \frac{R_{inv}}{n-1} \cong R_D$. Dacă semna-

lul la intrarea cu nivel E_2 revine din nou la valoarea E_1 , dioda se va bloca și tensiunea la ieșire va crește exponențial cu constanta de timp $\tau_{cr} = C \left(R' \parallel \frac{R_{inv}}{n} \right)$, tinzînd spre valoarea $U_1 \cong \cong E' + nI_{D_0}R'$.

Dacă $E_1 < E'$, în momentul în care se atinge nivelul $U_1 = E_1$ dioda se deschide și procesul tranzitoriu încetează.

Concluzii. Din studiul regimurilor staționare și tranzitorii se pot desprinde concluzii asupra situațiilor mai avantajoase în funcționarea circuitelor logice cu diode. Din cele arătate rezultă că se pot utiliza trei regimuri — a), b) sau c), dar la alegerea unuia din ele trebuie să se țină cont de faptul că pentru regimul b) se obține viteza maximă de funcționare (regim tranzitoriu minim), iar în regim c) se obține semnal maxim la ieșire.

5.1.3. Circuite logice cu diode semiconductoare cu mai multe etaje

Prin conectarea circuitelor logice de tip SI și SAU se pot materializa funcții logice complexe. În fig. 5.8 este prezentat un circuit logic SI—SAU în două nivele pentru semnale pozitive.

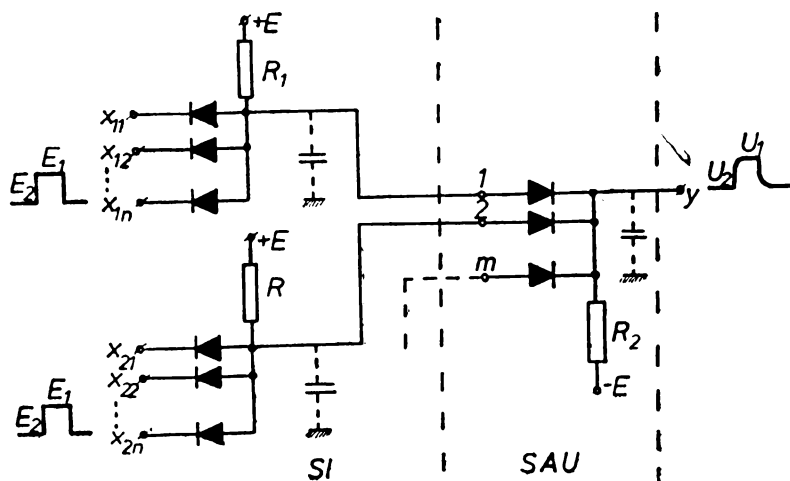


Fig. 5.8

Cele m intrări ale circuitului SAU sînt comandate de cître semnalele de ieşire ale circuitelor SI din primul nivel logic.

Pentru ca la ieşirea y să se obţină semnal ridicat ($U_1 \cong E_1$) este necesar ca cel puţin la o intrare din cele m ale circuitului SAU să fie semnal ridicat E_1 . Deci, cel puţin cînd la unul din cele m circuite SI la toate cele n intrări acţionează semnale înalte. Semnal coborît la ieşirea circuitului se obţine cînd la toate intrările circuitului SAU se aplică semnale coborîte. Circuitul realizează astfel funcţii de forma :

$$y = (x_{11}x_{12} \dots x_{1n}) \cup (x_{21}x_{22} \dots x_{2n}) \cup \dots \cup (x_{m1}x_{m2} \dots x_{mn}) = \bigcup_{i=1}^m \bigcap_{j=1}^n x_{ij}.$$

Datorită pierderilor care se produc în fiecare etaj, are loc o diminuare puternică a semnalelor (a treptelor) între acestea. Din acest motiv practic nu se utilizează circuite complexe cu diode cu mai mult de două nivele logice SI—SAU sau SAU—SI.

5.1.4. Proiectarea circuitelor logice cu diode

Proiectarea circuitelor logice urmăreşte obţinerea unui număr cît mai mare de circuite similare comandate de la ieşirea unică a circuitului considerat (număr maxim de ieşiri sau „fan out”) pentru un anumit număr de intrări („fan in”), realizînd totodată o funcţionare sigură. Numărul maxim de ieşiri se realizează urmărind obţinerea valorii maxime a curentului disponibil în circuitul de ieşire şi a valorii minime a curentului de comandă pe fiecare intrare. O funcţionare sigură se obţine proiectînd circuitul pentru convergenţa celor mai dezavantajoase condiţii din punctul de vedere al toleranţelor componentelor, surselor şi de mediu ambiant.

Calculul circuitelor logice cu diode semiconductoare urmăreşte alegerea elementelor componente astfel încît pentru o funcţionare corectă în situaţia cea mai dezavantajoasă să se obţină la ieşire un curent maxim, iar la intrare un curent minim. În cadrul proiectării acestor circuite se va neglija căderea de tensiune în sens direct pe diode şi timpul finit de comutare al acestora. Aproximările făcute introduc erori relativ mici, permiţînd simplificarea proiectării.

Calculul circuitului SI

În fig. 5.9 este prezentat un circuit SI în logică pozitivă, comandat pe cele n intrări cu semnale $E_2=0$ şi E_1 , deci $U_{si} = =E_1 \cong U_{se} = U_s$. Valoarea curentului disponibil la ieşire este

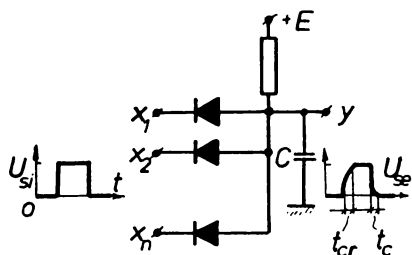


Fig. 5.9

determinat de parametrii schemei, fiind minim cînd ieșirea este la nivel ridicat U_s :

$$I_y = \frac{E - U_s}{R}$$

Avînd în vedere dispersia parametrilor elementelor schemei și a sursei, limita inferioară a curentului disponibil la ieșire va fi:

$$(5.6) \quad I_y = \frac{(1 - \delta_E)E - U_s}{(1 + \delta_R)R}$$

unde δ_E și δ_R sînt toleranțele pentru sursă și rezistoare.

Curentul de comandă în cazul cel mai dezavantajos, cînd una din intrări este acționată cu semnal coborît, iar celelalte cu semnal ridicat U_s , este dat de relația:

$$(5.7) \quad \bar{I}_x = \frac{(1 + \delta_E)E}{(1 - \delta_R R)} + (n - 1)\bar{I}_{inv}$$

în care n este numărul de intrări a circuitului, iar \bar{I}_{inv} — curentul invers al diodelor la temperatura maximă de lucru și la tensiunea inversă U_s .

Pierderile în circuit pot fi evaluate prin raportul între curenții de intrare și de ieșire ai schemei. Astfel, raportul \bar{I}_x/\bar{I}_y constituie o măsură a pierderilor și este de dorit să fie cît mai mic. Din studiul dependenței raportului \bar{I}_x/\bar{I}_y în funcție de E/U_s , avînd ca parametru $\bar{I}_{inv}R/(E - U_s)$ rezultă că pierderile scad cu creșterea raportului E/U_s . Dar, mărirea raportului peste valoarea 5 nu este avantajoasă [7] deoarece crește puterea disipată în rezistorul R .

Regimul tranzitoriu, după cum s-a văzut, este determinat de capacitatea echivalentă C a punctului de ieșire față de masă. La comutarea semnalului de la 0 la U_s se poate obține un timp de creștere impus, t_{cr} , dacă se ca'culează valoarea lui R astfel ca în regim tranzitoriu să furnizeze un curent mediu I_{tr} suficient pentru încărcarea în timpul impus a capacității C :

Notă. Prin subliniere sau supraliniere se va indica în continuare valoarea minimă sau maximă a parametrului considerat.

$$(5.8) \quad I_{tr} = C \frac{\Delta U}{\Delta t} = C \frac{U_s}{t_{cr}}$$

Din relațiile (5.6) și (5.8) se poate determina valoarea rezistorului R , astfel ca $\underline{I}_v = I_{tr}$:

$$(5.9) \quad R = \frac{(1 - \delta_E)E - U_s}{\frac{C U_s}{t_{cr}} (1 + \delta_R)}$$

Timpul de cădere a semnalului, t_c , este determinat de generatorul de comandă cînd diodele sînt deschise. În acest caz generatorul trebuie să asigure și curentul din regimul tranzitoriu necesar realizării timpului de cădere impus:

$$(5.10) \quad I_{x.tot.} = \bar{I}_x + C \frac{U_s}{t_c}$$

În cazul cînd circuitul are și o sarcină R_s , în locul lui E se introduce valoarea echivalentă E' . După stabilirea valorii rezistorului R se determină și valoarea sursei.

Exemplul 1. Să se proiecteze un circuit SI cu 5 intrări ($n=5$) în logică pozitivă, capabil să furnizeze un semnal $U_s=5V$ pe o sarcină capacitivă $C=100 \text{ pF}$ și avînd $t_{cr}=1,2 \text{ } \mu\text{sec}$ și $t_c=1 \text{ } \mu\text{sec}$. Circuitul trebuie să funcționeze corect în domeniul de temperatură cuprins între 0 și 65°C . Toleranțele elementelor circuitului vor fi $\delta_E = \delta_R = 5\%$.

1. Considerînd satisfăcător din punct de vedere al pierderilor raportul $E/U_s=4$, rezultă $E=20V$.

2. Se alege dioda de comutație 1N4009(1N 4154) cu $U_{inv} = 25V > \bar{U}_{inv} = U_s = 5V$ și $\bar{I}_{inv} = 100 \text{ nA}$ (la $U_{inv} = 25V$ și $T_{amb} = 150^\circ\text{C}$).

3. Din relația (5.9) rezultă:

$$R = \frac{(1 - 0,05)20 - 5}{\frac{100 \cdot 10^{-12} \cdot 4}{1,2 \cdot 10^{-6}} (1 + 0,05)} = 32 \text{ k}\Omega$$

Se alege valoarea standardizată $R=33 \text{ k}\Omega$.

4. Cu relația (5.7) se determină curentul necesar la intrare în regim staționar:

$$\bar{I}_x = \frac{(1+0,05)20}{(1-0,05)33} + 4 \cdot 100 \cdot 10^{-6} = 0,67 \text{ mA},$$

iar cu relația (5.10) curentul minim necesar de la generatorul de comandă în timpul comutării pentru realizarea unui $t_c = 1 \mu\text{sec}$:

$$\underline{I}_{x, \text{tot.}} = 0,67 + 100 \cdot 10^{-12} \frac{5 \cdot 10^3}{1 \cdot 10^{-6}} = 1,17 \text{ mA}.$$

5. Curentul disponibil la ieșirea circuitului se stabilește cu relația (5.6):

$$I_y = \frac{(1-0,05)20-5}{(1+0,05)33} = 0,404 \text{ mA}.$$

Rezultă astfel $\bar{I}_x/I_y = 1,66$.

Exemplul 2. Să se proiecteze circuitul SI din exemplul precedent, avînd și sarcina $R_s = 100 \text{ k}\Omega$ cu $C_s = 50 \text{ pF}$.

În acest caz rezultă $C_t = C + C_s = 150 \text{ pF}$.

1. Considerînd $E'/U_s = 4$ rezultă $E' = 20\text{V}$.

$$2. \quad R = \frac{0,95 \cdot 20 - 5}{\frac{150 \cdot 10^{-12} \cdot 5}{1,2 \cdot 10^{-6}}} = 21,33 \text{ k}\Omega. \text{ Se alege } R = 22 \text{ k}\Omega.$$

3. Valoarea tensiunii sursei de alimentare:

$$E = E' \left(1 + \frac{R}{R_s} \right) = 20 \left(1 + \frac{22}{100} \right) = 24,4\text{V}.$$

Calculul circuitului SAU

În fig. 5.10 este prezentată schema circuitului SAU în logică pozitivă, comandat pe cele n intrări cu nivelele $E_2 = 0$ volți și $E_1 = U_s$. Valoarea minimă a curentului disponibil la ieșirea schemei este atunci cînd la ieșire este semnal 0 logic, adică atunci cînd la intrare toate diodele sînt acționate cu semnal coborît:

$$(5.11) \quad \underline{I}_y = \frac{(1 - \delta_E)E}{(1 + \delta_R)R}.$$

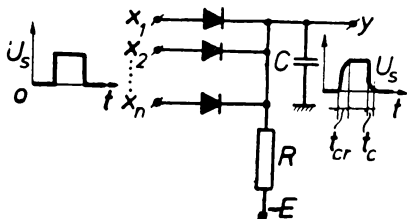


Fig. 5.10

În regim staționar curentul de comandă necesar în situația cea mai dezavantajoasă este atunci cînd o singură intrare este acționată cu semnal ridicat U_s , iar celelalte intrări sînt acționate cu semnal 0, deci polarizate invers :

$$(5.12) \quad \bar{I}_x = \frac{(1 + \delta_E)E + U_s}{(1 - \delta_R)R} + (n-1)\bar{I}_{inv.}$$

Și în acest caz stabilind dependența raportului $\bar{I}_x/\underline{I}_y$ de raportul E/U_s , cu $\bar{I}_{inv}R/E$ ca parametru, rezultă că pentru valorile medii tipice $n=3$ și $\delta_E = \delta_R = 5\%$ nu este economic să se depășească pentru E/U_s valoarea $5/7$.

O valoare determinată a timpului de creștere a semnalului la ieșire se poate obține stabilind astfel valoarea lui R încît să se asigure curentul mediu de încărcare a condensatorului C în procesul tranzitoriu. Avînd în vedere (5.11), din condiția $I_y = I_{tr}$ rezultă :

$$(5.13) \quad R = \frac{(1 - \delta_E)t_{cr} E}{(1 + \delta_R)C U_s}.$$

Pentru a obține un timp de cădere impus este necesar ca generatorul de comandă să fie capabil să furnizeze în regim tranzitoriu un curent mediu suplimentar CU_s/t_c . Astfel, curentul total minim necesar pe o intrare este dat de relația :

$$(5.14) \quad \underline{I}_{x.tot.} = \bar{I}_x + C \frac{U_s}{t_c}.$$

Exemplu. Să se proiecteze un circuit SAU cu $n=4$ în logică pozitivă capabil să furnizeze un semnal $U_s=6V$ pe o sarcină capacitivă $C=100 \text{ pF}$ cu $t_{cr}=t_c=1 \text{ } \mu\text{sec}$. Circuitul trebuie să funcționeze între 0 și $55^\circ C$ cu $\delta_E = \delta_R = 5\%$.

1. Considerînd satisfăcător $E/U_s=4$ rezultă $E=-24 V$.

2. Folosind dioda de comutație 1N4009 cu $U_{inv}=25V$ și $\bar{I}_{inv}=100 \text{ nA}$, cu relațiile stabilite mai sus se calculează circuitul.

3. Din relația (5.13) rezultă :

$$R = \frac{0,95 \cdot 1 \cdot 10^{-6}}{1,05 \cdot 100 \cdot 10^{-12}} 4 = 36,2 \text{ k}\Omega.$$
 Se alege valoarea normalizată $R=36 \text{ k}\Omega$.

4. Curentul de comandă în regim staționar, în condițiile cele mai dezavantajoase, va fi :

$$\bar{I}_x = \frac{1,05 \cdot 24 + 6}{0,95 \cdot 36} + 3 \cdot 100 \cdot 10^{-6} = 0,913 \text{ mA},$$

iar curentul minim necesar în timpul comutării cu $t_c = 1 \mu\text{sec}$

$$\underline{I}_x = 0,913 + 100 \cdot 10^{-12} \frac{6}{1 \cdot 10^{-6}} 10^3 = 1,513 \text{ mA}.$$

5. Curentul disponibil la ieșirea circuitului se determină cu relația (5.11)

$$\underline{I}_y = \frac{0,95 \cdot 24}{1,05 \cdot 36} = 0,603 \text{ mA},$$

de unde $\bar{I}_x / \underline{I}_y = 1,514$.

Dacă circuitul are și o sarcină, de exemplu $R_s = 100 \text{ k}\Omega$ și $C_s = 50 \text{ pF}$, considerînd $E' / U_s = 4$ rezultă $E' = -24 \text{ V}$ iar $R = \frac{0,95 \cdot 1 \cdot 10^{-6}}{1,05 \cdot 150 \cdot 10^{-12}} 4 = 24,2 \text{ k}\Omega$. Alegînd $R = 24 \text{ k}\Omega$ rezultă valoarea sursei de alimentare :

$$E = E' \left(1 + \frac{R}{R_s} \right) = 24 \left(1 + \frac{24}{100} \right) = 30 \text{ V}.$$

Calculul circuitelor cu două nivele

La proiectarea circuitelor logice obținute din conectarea de circuite *SI* cu circuite *SAU* se utilizează relațiile de calcul stabilite pentru fiecare tip de circuit. Calculul începe de la ieșirea schemei complexe spre intrare. În acest mod se pot stabili curenții necesari în punctele de conectare dintre circuitele cu funcții logice diferite.

Exemplu. Să se proiecteze un circuit *SI-SAU* cu $n_{SI} = n_{SAU} = 4$ în logică pozitivă, capabil să furnizeze un semnal la ieșire $U_s = 4 \text{ V}$, cu un timp de creștere și de cădere de $1 \mu\text{sec}$. Capacitățile echivalente ale circuitelor sînt $C_{SI} = C_{SAU} = 100 \text{ pF}$, toleranțele $\delta_E = \delta_R = 5\%$, iar domeniul de temperatură cuprins între 0 și 45°C .

1. Se adoptă $E / U_s = 5$. Rezultă $E = 20 \text{ V}$. Se vor folosi diode de comutație cu *Ge* de tip *EFD 105* cu : $U_{inv} = 30 \text{ V} > \bar{U}_{inv} = U_s = 4 \text{ V}$ și $I_{inv} = 3 \mu\text{A}$ la $T_{amb} = 25^\circ\text{C}$ și $U_{inv} = 5 \text{ V}$. Rezultă $\bar{I}_{inv} = I_{inv 25^\circ} \cdot 2^{(T - 25)/10} = 3 \cdot 2^{(45 - 25)/10} = 12 \mu\text{A}$.

2. Se calculează circuitul SAU de la ieșire ca un circuit SAU izolat. Cu relația (5.13) și pentru $E = -20$ V rezultă $R_{SAU} = 45,2$ k Ω . Se adoptă $R_{SAU} = 43$ k Ω . Curentul disponibil la ieșirea circuitului se calculează cu (5.11)

$$\underline{I}_{y, SAU} = \frac{0,95 \cdot 20}{1,05 \cdot 43} = 0,42 \text{ mA.}$$

Curentul necesar la intrarea circuitului SAU ținând cont și de asigurarea regimului tranzitoriu va fi:

$$\underline{I}_{x, tot, SAU} = \frac{1,05 \cdot 20 + 4}{0,95 \cdot 43} + 3 \cdot 12 \cdot 10^{-3} + 100 \cdot 10^{-12} \frac{4}{1 \cdot 10^{-6}} 10^3 = 1,05 \text{ mA}$$

3. Circuitul SI trebuie să asigure curentul la ieșire pentru comanda circuitului SAU precum și pentru încărcarea capacității C_{SI} :

$$\underline{I}_{y, SI} = \underline{I}_{x, tot, SAU} + C_{SI} \frac{U_s}{t_{cr}} = 1,05 + 100 \cdot 10^{-12} \frac{4}{1 \cdot 10^{-6}} 10^3 = 1,45 \text{ mA.}$$

4. Adoptînd același raport și pentru circuitul SI, rezultă $E = +20$ V. Folosind relația (5.6) rezultă:

$$R_{SI} = \frac{(1 - \delta_E)E - U_s}{(1 + \delta_R) \underline{I}_{y, SI}} = \frac{0,95 \cdot 20 - 4}{1,05 \cdot 1,45} = 9,85 \text{ k}\Omega. \text{ Se alege } R_{SI} = 10 \text{ k}\Omega.$$

5. Curentul necesar la intrarea circuitului SI în regim staționar:

$$\bar{I}_{x, SI} = \frac{1,05 \cdot 20}{0,95 \cdot 10} + 3 \cdot 12 \cdot 10^{-3} = 2,24 \text{ mA,}$$

ar în regim tranzitoriu $\underline{I}_{x, tot, SI} = 2,24 + 100 \cdot 10^{-12} \frac{4}{1 \cdot 10^{-6}} 10^3 = 2,64 \text{ mA.}$

6. Pierderile introduse de cele două etaje se pot aprecia ca fiind date de $\bar{I}_{x, SI} / \underline{I}_{y, SAU} = 5,34$ în regim staționar.

5.2. Circuite logice cu tranzistoare bipolare

Posibilitatea de a lucra în regim de comutație (blocat-saturat) conferă și tranzistoarelor proprietatea de element cu acțiune discretă, apt pentru a fi utilizat la materializarea funcțiilor

booleene. Avînd, spre deosebire de diode, și proprietatea de amplificare, numărul de etaje logice nu este limitat. De asemenea, avînd proprietatea de inversare a semnalului, cu ajutorul lor se poate materializa funcția negație.

Existența celor două tipuri constructive — *pnp* și *npn* — a condus la diferențierea circuitelor logice cu tranzistoare în circuite cu logică pozitivă și cu logică negativă. Astfel, pentru circuitele cu tranzistoare *pnp* se folosește logica negativă: semnalul 1 logic se atașează celui mai negativ nivel de tensiune. În schimb, circuitele cu tranzistoare *npn* lucrează obișnuit în logică pozitivă: semnalul 1 logic se atașează celui mai pozitiv nivel de tensiune. În continuare se vor considera circuitele logice realizate cu tranzistoare *npn* cu cea mai mare utilizare în prezent, trecerea la scheme cu tranzistoare *pnp* ne reprezentînd dificultăți. De asemenea, se vor studia numai circuitele avînd tranzistoare conectate în montaj emitor comun (EC), celelalte conexiuni posibile fiind rar folosite.

În mod obișnuit, tranzistorul dintr-un circuit logic este comandat prin intermediul unui circuit de cuplaj tot de un circuit care funcționează în regim de comutație (fig. 5.11). Circuitul

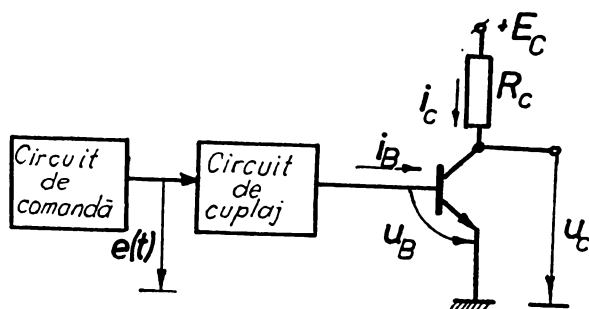


Fig. 5.11

de comandă poate fi realizat cu tranzistor sau cu diodă. Valorile semnalului de comandă $e(t)$ și ale rezistenței interne R_g depind în acest caz de starea circuitului: dacă este conectat (tranzistorul saturat, dioda în conducție) semnalul $e(t)$ are valoare redusă iar R_g este mică; dacă circuitul este deconectat (tranzistorul sau dioda blocate), $e(t)$ are valoare mare iar R_g este de asemenea de valoare ridicată. Cuplajul între cele două circuite, de comandă și comandat, se poate face direct sau prin intermediul unor circuite cu parametri bine determinați. Din punctul de vedere al modului

de realizare a cuplajului circuitele logice cu tranzistoare bipolare pot fi împărțite în următoarele tipuri principale :

- circuite logice cu rezistoare și tranzistoare (*RTL*) ;
- circuite logice cu diode și tranzistoare (*DTL*) ;
- circuite logice cu tranzistoare cu cuplaj direct (*DCTL*).

Primele două tipuri sînt cel mai mult utilizate în comanda discretă industrială, rezistoarele și diodele realizînd funcțiile logice *SI*, *SAU* sau combinații ale acestora iar tranzistorul amplificarea și negarea. În ultimul caz prin intermediul tranzistoarelor se realizează și funcțiile *SI* și *SAU*.

5.2.1. Circuite logice RTL

Cu ajutorul rezistoarelor și tranzistoarelor se materializează în mod obișnuit funcțiile logice *NU* și *NICI*. În fig. 5.12 sînt prezentate circuitele de negare (fig. 5.12a) și *NICI* (fig. 5.12b). Aplicînd la intrarea circuitului din fig. 5.12a semnal logic 0 (nivel de tensiune apropiat de zero volți) tranzistorul se blochează și

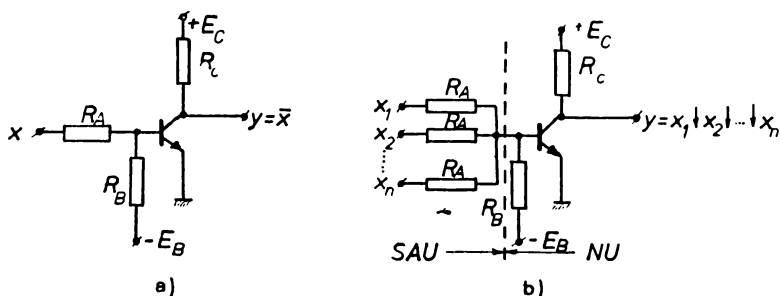


Fig. 5.12

la ieșire se obține semnal logic 1 (nivel de tensiune apropiat de $+E_c$). Dacă la intrare se aplică semnal logic 1 (nivel apropiat de $+E_c$) tranzistorul se saturează și la ieșire se obține semnal logic 0 (nivel apropiat de zero volți). Avînd în vedere funcționarea descrisă, un astfel de circuit asigură complementarea semnalului aplicat la intrarea sa.

Dacă la intrările circuitului din fig. 5. 12b se aplică semnale 0 logic tranzistorul se blochează și la ieșire se obține semnal 1 logic. Cînd la cel puțin una din intrări se aplică semnal 1 logic, tranzistorul se saturează și la ieșire se obține semnal logic 0.

Circuitul descris materializează funcția *NICI* pentru n argumente, rezistoarele realizând funcția *SAU* iar tranzistorul funcția *NU*.

Deoarece circuitele funcționează cu nivele mari de tensiune (de valoarea lui E_c), rezistoarele R_A asigură limitarea curentului în baza tranzistoarelor. Blocarea sigură se realizează prin intermediul sursei de polarizare inversă $-E_B$ și a rezistoarelor R_B .

Analiza funcționării circuitului N U (inversor logic)

Considerînd și rezistența internă a circuitului de comandă, schema folosită pentru analiza circuitului inversor logic este prezentată în fig. 5.13. Semnalul de comandă se consideră că se modifică în salt între nivelele $E_2 \cong 0$ și $E_1 \cong E_c$.

Regimul staționar. Pentru ca circuitul prezent să funcționeze corect este necesar ca parametrii circuitului de cuplaj să fie astfel determinați ca tranzistorul să lucreze în regim de comutație, adică să se asigure următoarele stări staționare :

a) tranzistorul să fie blocat cînd la intrare acționează semnalul $e = E_2$;

b) tranzistorul să fie saturat cînd la intrare acționează semnalul $e = E_1$.

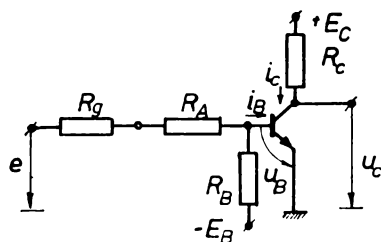


Fig. 5.13

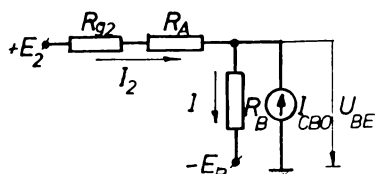


Fig. 5.14

Pentru prima stare, schema echivalentă este prezentată în fig. 5.14. Tranzistorul blocat este reprezentat prin generatorul de curent rezidual I_{CB0} , iar rezistența internă a generatorului de comandă pentru $e = E_2$ se notează prin R_{g2} . Avînd în vedere că pentru o blocare sigură este necesar ca $u_{BE} \leq 0$ (pentru tranzistoarele *pnp* $u_{BE} \geq 0$) în schema echivalentă s-au stabilit sensurile curenților, în care caz se pot scrie următoarele ecuații :

$$\begin{cases} E_2 = (R_{g2} + R_A)I_2 - U_{BE} \\ E_B = U_{BE} + R_B I. \end{cases}$$

Avînd în vedere că $I = I_2 + I_{CB0}$, se obține :

$$(5.15) \quad U_{BE} = \frac{-E_2 R_B + E_B (R_A + R_{g2}) - I_{CB0} (R_A + R_{g2})}{R_A + R_{g2} + R_B}$$

Blocarea sigură a tranzistoarelor s-a constatat că se realizează pentru valori $|U_{Bb1}| = (0,1 - 0,2)$ V, numită valoare prag pentru blocarea profundă. Evident, asigurarea blocării se poate face dacă se calculează circuitul de cuplaj astfel încît $U_{BE} \geq U_{Bb1}$.

În fig. 5.15 este prezentată schema echivalentă pentru a doua stare staționară a circuitului. În acest caz tranzistorul trebuie să se satureze; $U_{B_s} > 0$ fiind tensiunea de saturație bază-emitor, iar $U_{C_s} > 0$ ($U_{B_s} > U_{C_s}$) — tensiunea la saturație colector-emitor. Prin R_{g1} s-a notat rezistența prezentată de circuitul de comandă pentru $e = E_1$.

Pentru ca tranzistorul să se satureze este necesar ca

$$I_{B1} \geq I_{B_s} = \frac{I_{C_s}}{h_{21E}} = \frac{E_c}{h_{21E} R_c},$$

unde I_{B_s} și I_{C_s} sînt curenții de bază și de colector la saturație, iar h_{21E} factorul static de amplificare în curent. Avînd în vedere schema echivalentă, se poate scrie :

$$(5.16) \quad I_{B1} = \frac{E_1 - U_{B_s}}{R_A + R_{g1}} - \frac{E_B + U_{B_s}}{R_B} \geq \frac{E_c}{h_{21E} R_c}$$

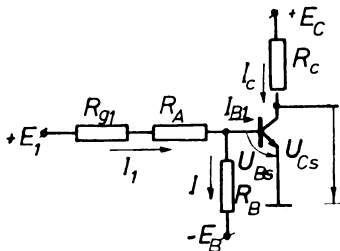


Fig. 5.15

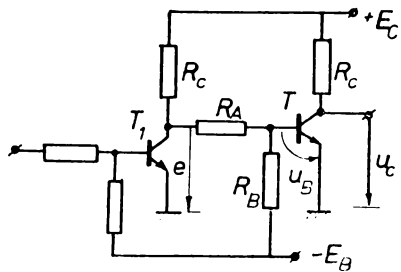


Fig. 5.16

În practica comenzilor secvențiale circuitul de comandă este un circuit identic cu cel comandat (fig. 5.16). În acest caz E_2 este tocmai semnalul la ieșirea circuitului cu tranzistorul T_1 cînd este saturat, deci $E_2 = U_{C_s}$ iar $R_{g2} = R_c \parallel R_{ies}$. Dar R_{ies}

(rezistența prezentată de tranzistor în stare saturată) este mult mai mică decât R_C și determină valoarea lui $R_{g2} \cong R_{tes.s}$. În comparație cu R_A valoarea stabilită a lui R_{g2} se poate neglija (kiloohmi în comparație cu zeci de ohmi). Semnal E_1 la ieșirea circuitului de comandă va exista când T_1 va fi blocat, adică $E_1 = E_c - I_{CB0} R_c$. În acest caz $R_{g1} = (R_c \parallel R_{tes.bl}) \cong R_c$.

Pentru situația considerată relațiile (5.15) și (5.16), devin :

$$(5.17) \quad U_{BE} = \frac{-U_{Cs} R_B + E_B R_A - I_{CB0} R_A R_B}{R_A + R_B} \geq U_{Bbl}$$

$$(5.18) \quad I_{B1} = \frac{E_c - I_{CB0} R_c - U_{Bs}}{R_A + R_c} - \frac{E_B + U_{Bs}}{R_B} \geq \frac{E_c}{R_c h_{21E}}$$

Relațiile stabilite nu țin cont însă de condițiile cele mai defavorabile în ceea ce privește dispersia parametrilor și regimul termic cel mai dificil. Pentru ca cele două regimuri staționare să poată fi realizate sigur și în cel mai defavorabil caz este necesar să fie îndeplinite următoarele condiții :

$$(5.19) \quad \underline{U}_{BE} = \frac{-\bar{U}_{Cs} \bar{R}_B + \bar{E}_B \bar{R}_A - \bar{I}_{CB0} \bar{R}_A \bar{R}_B}{\bar{R}_A + \bar{R}_B} \geq U_{Bbl}$$

$$(5.20) \quad I_{B1} = \frac{E_c - \bar{I}_{CB0} \bar{R}_c - \bar{U}_{Bs}}{\bar{R}_c + \bar{R}_A} - \frac{\bar{E}_B + \bar{U}_{Bs}}{\bar{R}_B} \geq \frac{E_c}{\bar{R}_c h_{21E}}$$

Pentru un anumit tranzistor, ales astfel ca $E_c < U_{CEadm.}$, rămân necunoscute E_c, E_B, R_c, R_A, R_B . În mod obișnuit se alege E_c, E_B (între 12 și 24 V pentru a fi mult mai mari decât valorile tensiunilor reziduale \bar{U}_{Bs} și \bar{U}_{Cs}) și R_c prin impunerea unui curent I_c în domeniul $h_{21E} \cong \bar{h}_{21E}$. Rezolvînd inegalitățile (5.19) și (5.20) în raport cu R_B , rezultă :

$$(5.21) \quad \bar{R}_B \leq \frac{E_B - U_{Bbl}}{\bar{U}_{Cs} + U_{Bbl}} + \bar{I}_{CB0} R_A$$

$$(5.22) \quad R_B \geq \frac{\bar{E}_B + \bar{U}_{Bs}}{E_c - \bar{I}_{CB0} \bar{R}_c - \bar{U}_{Bs}} - \frac{E_c}{\bar{R}_c h_{21E}}$$

Dacă se consideră semnele de egalitate, aceste inecuații definesc frontierele domeniului de funcționare din planul (R_A, R_B) . În fig. 5.17a este reprezentat domeniul de funcționare definit de frontiera domeniului de blocare $\bar{R}_B = f_1(R_A)$ și a domeniului de saturare $\underline{R}_B = f_2(R_A)$. Punctele situate în domeniul delimitat de cele două curbe au coordonate care satisfac condițiile (5.21) și (5.22). Valorile rezistoarelor trebuie astfel determinate încât

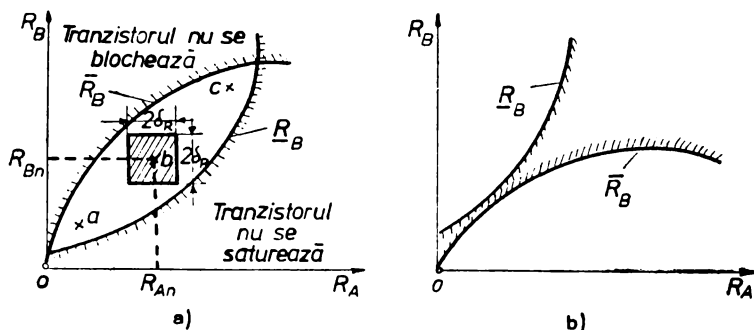


Fig. 5.17

ținând cont de toleranțe (δ_R) dreptunghiul toleranțelor să nu intersecteze frontierele. În alegerea punctului de funcționare trebuie să se țină cont și de următoarele :

- alegerea funcționării în punctul „a” conduce la valori mici pentru R_A și R_B , fiind necesare puteri mari pentru sursele de alimentare și de comandă. În schimb, datorită capacităților parazite mici circuitele pot funcționa la frecvențe mari de comutare ;

- în punctul „c”, dimpotrivă, R_A și R_B au valori mari și efectele sînt contrarii cazului precedent ;

- punctul „b” de funcționare constituie o situație de compromis între consum acceptabil și viteză de comutare corespunzătoare.

Este posibil ca parametrii circuitului să fie aleși într-un mod atît de nereușit încît curbele să nu se intersecteze, deci să nu existe un domeniu de funcționare (fig. 5.17b). În acest caz, este necesar să se modifice unii parametri impuși inițial. După cum se observă din relațiile (5.12) și (5.22), frontiera de blocare se ridică la mărirea lui E_B iar frontiera de saturare coboară

la mărirea lui R_c , E_c și h_{21E} și la micșorarea lui E_B . Rezultă că modificarea lui E_B nu este indicată deoarece ambele frontiere se modifică în același sens cu modificarea sursei de polarizare inversă. Pentru obținerea unui domeniu de funcționare trebuie să se acționeze asupra lui E_c , R_c sau h_{21E} (alegerea unui alt tranzistor) în sensul măririi lor.

Regimul tranzitoriu. Procesele tranzitorii determină viteza de acționare a circuitului. Aceste procese sînt determinate de fenomenele care au loc în tranzistor și care determină pentru regimul de comutație duratele fronturilor la saturare și la blocare.

Regimul tranzitoriu la saturare. Inițial se consideră că intrarea circuitului este acționată cu semnal coborît E_2 și ca urmare tranzistorul este blocat. În momentul t_1 semnalul la intrare se modifică în salt pînă la valoarea E_1 (fig. 5.18). Dacă se neglijează timpul foarte mic, denumit *timp de pregătire*, necesar încărcării capacității joncțiunii bază-emitor sub acțiunea semnalului E_1 , curentul în bază crește pînă la valoarea $I_{B1} \geq I_{Bs}$. Ca urmare sarcina în bază și curentul de colector vor crește exponențial cu constanta de timp $\tau_B = h_{21E} \tau_\alpha$ ($\tau_\alpha = \frac{1}{2\pi f_\alpha}$, în care f_α este frec-

vența de tăiere a tranzistorului). În momentul în care se atinge

valoarea $h_{21E} I_{Bs} = I_{Cs}$ procesul tranzitoriu de saturare este practic terminat, determinînd durata frontului la saturare. Curentul de colector nu se mai modifică, dar sarcina în bază continuă să crească sub acțiunea semnalului de comandă pînă cînd se încheie procesul de stocare (determinat de constanta de stocare τ_s). Semnalul la ieșire este determinat de potențialul de colector, acesta modificîndu-se de la valoarea $E_c - I_{CB0} R_c$ la valoarea U_{Cs} . Considerînd comanda tranzistorului prin sarcină și aproximînd procesul de modificare a sarcinii $Q(t)$ a purtătorilor minoritari (proces condiționat de difuzie) din bază la variația curentului $i_B(t)$ prin ecuația diferențială

$$\frac{dQ(t)}{dt} + Q(t) = i_B(t),$$

se pot stabili relații cantitative asupra timpului de comutare. În cazul semnalelor mari de comandă, durata frontului de saturare se poate determina cu relația/12/

$$(5.23) \quad t_{f1} \cong \tau_B \frac{I_{B_s}}{I_{B_1}} = h_{21E} \tau_\alpha \frac{I_{B_s}}{I_{B_1}} = \tau_\alpha \frac{I_{C_s}}{I_{B_1}}$$

Conform cu (5.23) scurtarea duratei de conectare se poate obține prin mărirea curentului I_{B_1} sau prin folosirea unui tranzistor de frecvență ridicată.

Regimul tranzitoriu la blocare. Tranzistorului saturat sub acțiunea semnalului E_1 , aplicat la intrarea circuitului i se aplică o treaptă negativă, prin revenirea semnalului la valoarea E_2 (fig. 5.19). Ca urmare, și sub influența sursei de polarizare in-

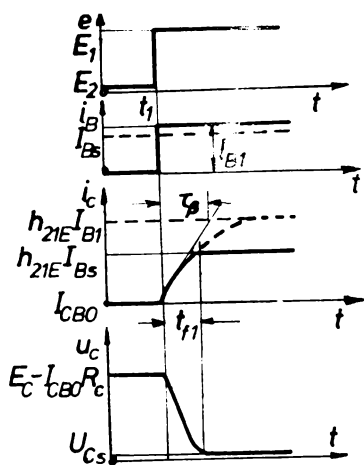


Fig. 5.18

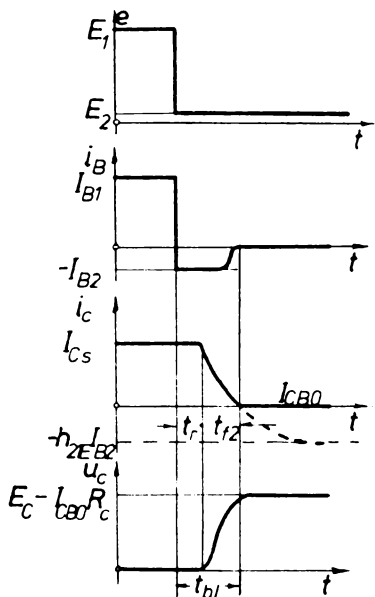


Fig. 5.19

versă $-E_B$, curentul în bază se modifică în salt de la I_{B_1} la $-I_{B_2}$. Curentul $-I_{B_2}$ provoacă micșorarea sarcinii stocate în bază. Această micșorare se face exponențial cu constanta de timp τ_s . Atît timp cît are loc resorbția sarcinii stocate în bază (t_r) curentul și tensiunea de colector nu se modifică. Durata de resorbție scade cu scăderea gradului de saturare al tranzistorului ($s = I_{B_1}/I_{B_s}$). După încheierea resorbției sarcinii din bază tran-

zistorul ajunge la limita regiunii active. Din acest moment începe scăderea exponențială a curentului de colector, de la I_{C_s} la I_{CB0} , cu constanta τ_B , determinînd durata frontului t_{f2} . Concomitent cu scăderea curentului de colector crește potențialul de colector pînă la valoarea $E_c - I_{CB0}R_c$. Pe măsura blocării tranzistorului impedanța sa de intrare crește și în momentul în care devine comparabilă cu rezistența R_A începe scăderea spre zero a curentului de bază. Pentru semnal mare, durata totală a timpului de blocare a tranzistorului se poate determina cu expresia :

$$(5.24) \quad t_{bl} = t_r + t_{f2} \cong \tau_B \frac{I_{B1}}{I_{B2}}$$

Rezultă că pentru obținerea unui timp de blocare cît mai redus este necesar ca I_{B1} să fie cît mai mic iar $-I_{B2}$ cît mai mare.

Din cele prezentate rezultă că scurtarea timpului de conectare al tranzistorului se poate realiza prin mărirea curentului I_{B1} , iar pentru scurtarea timpului de blocare I_{B1} să fie cît mai mic iar I_{B2} cît mai mare. În fig. 5.20a este reprezentată modifi-

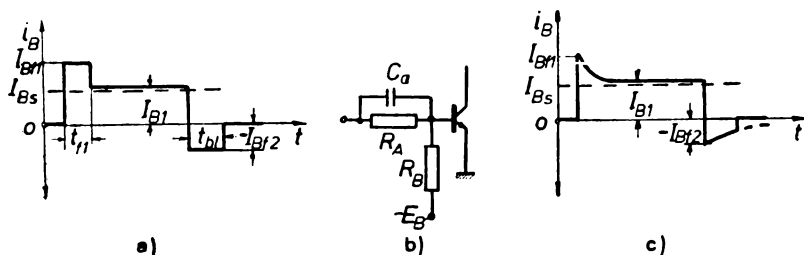


Fig. 5.20

care necesară a curentului în bază pentru a satisface condițiile impuse. Astfel, în cazul ideal la deblocare $I_{Bf1} \gg I_{B_s}$ și de durată t_{f1} revenind apoi la valoarea $I_{B1} \geq I_{B_s}$. La blocare $i_B = -I_{Bf2}$ pe durata t_{bl} și apoi să scadă la zero. Deși condițiile par contradictorii se poate obține o modificare a curentului în bază apropiată de cea ideală prin conectarea unui condensator convenabil (100—300 μF) în paralel cu R_A (fig. 5.20b), denumit condensator de accelerare. Modificarea curentului în bază după conectarea condensatorului C_a este prezentată în fig. 5.20c. Încheierea procesului de descărcare a condensatorului la blocare este determinat de creșterea impedanței de intrare a tranzistorului.

Analiza funcționării circuitului RTL—NICI

În fig. 5.21 este prezentată schema unui circuit RTL—NICI cu n intrări și cu m ieșiri. Se consideră cea mai întâlnită situație în practică când semnalele de intrare ale circuitului sînt semnalele de ieșire ale altor circuite similare. De asemenea, se consideră că circuitul în studiu comandă de la ieșirea unică m circuite similare.

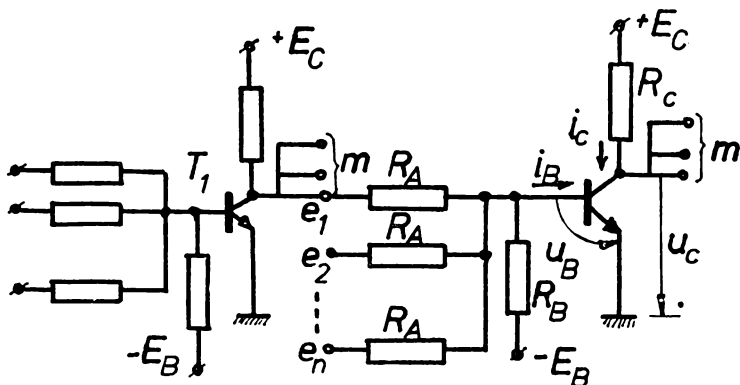


Fig. 5.21

Regimul staționar. Tranzistorul din circuitul considerat se poate afla într-unul din regimurile staționare :

a) *blocat*, în cazul când toate semnalele de intrare sînt nivele reduse de tensiune, respectiv tensiunile U_{c_s} preluate de la colectoarele tranzistoarelor T_1 saturate ;

b) *saturat*, când cel puțin la una din intrări se aplică un nivel ridicat de tensiune, respectiv dacă cel puțin unul dintre tranzistoarele de comandă este blocat.

În continuare se vor stabili condițiile care trebuie îndeplinite de circuitul de cuplaj pentru ca să fie realizate cele două stări ale schemei.

a) Schema echivalentă pentru cazul când tranzistorul T trebuie să fie blocat este prezentată în fig. 5.22. Semnalele e_1, e_2, \dots, e_n se consideră

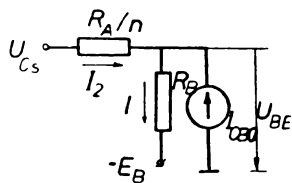


Fig. 5.22

identice și egale cu U_{C_s} . Prin analogie cu regimul similar de la circuitul inversor se poate scrie imediat :

$$U_{BE} = \frac{-U_{C_s} R_B + E_B \frac{R_A}{n} - I_{CB0} R_B \frac{R_A}{n}}{\frac{R_A}{n} + R_B} \geq U_{Bbl}$$

Pentru asigurarea regimului de blocare și în cazul cel mai defavorabil în ceea ce privește dispersia parametrilor componentelor și surselor, trebuie îndeplinită condiția :

$$\underline{U}_{BE} = \frac{-\bar{U}_{C_s} \bar{R}_B + \frac{R_A}{n} \bar{E}_B - \bar{I}_{CB0} \bar{R}_B \frac{R_A}{n}}{\frac{R_A}{n} + \bar{R}_B} \geq U_{Bbl},$$

de unde

$$(5.25) \quad \bar{R}_B \leq \frac{E_B - U_{Bbl}}{\frac{\bar{U}_{C_s} + U_{Bbl}}{R_A} n + \bar{I}_{CB0}}$$

b) Pentru a doua stare staționară se consideră cazul cel mai defavorabil din punctul de vedere al interconectării și anume atunci cînd numai la o singură intrare se aplică semnal logic 1, la toate celelalte acționînd semnal 0 logic (U_{C_s}). În fig. 5.23 este reprezentată schema echivalentă a circuitului. Se observă că s-a considerat situația cea mai dezavantajoasă din punct de vedere al semnalului 1, cînd tranzistorul de comandă T_1 este încărcat pe toate cele m ieșiri ale sale cu circuite similare. Conform schemei echivalente, condiția de saturare se scrie :

$$I_{B1} = I_1 - I_R - I \geq \frac{I_{C_s}}{h_{21E}} = \frac{1}{h_{21E}} (I_{Rc} + I')$$

sau

$$I_{B1} = \frac{U_{Cbl} - U_{B_s}}{R_A} - \frac{U_{B_s} - U_{C_s}}{R_A} (n-1) - \frac{E_B + U_{B_s}}{R_B} \geq \frac{1}{h_{21E}} \left(\frac{E_C - U_{C_s}}{R_c} + m \frac{U_{B_s} - U_{C_s}}{R_A} \right).$$

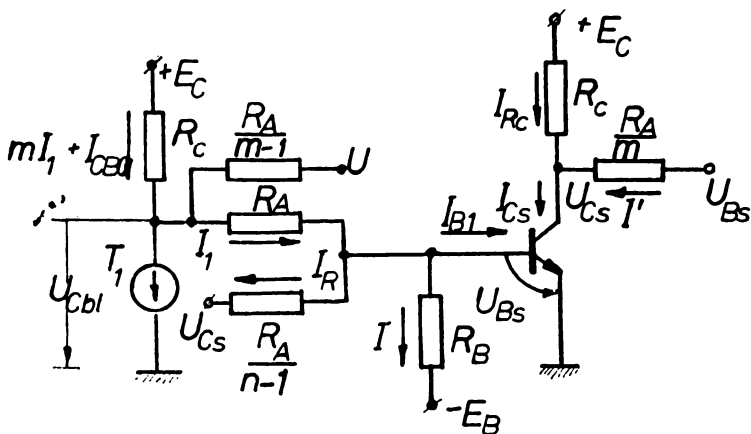


Fig. 5.23

Avînd în vedere dispersia parametrilor și influența temperaturii :

$$\begin{aligned}
 \underline{I_{B1}} &= \frac{E_c - \bar{I}_{CB0} \bar{R}_c - \bar{U}_{B_s}}{\bar{R}_A + m \bar{R}_c} - \frac{\bar{U}_{B_s} - U_{C_s}}{\bar{R}_A} (n-1) - \frac{\underline{E}_B + \bar{U}_{B_s}}{\bar{R}_B} \geq \\
 (5.26) \quad &\geq \frac{1}{h_{21E}} \left(\frac{E_c - U_{C_s}}{\bar{R}_c} + m \frac{\bar{U}_{B_s} - U_{C_s}}{\bar{R}_A} \right),
 \end{aligned}$$

în care s-a explicat U_{cbl} . Din (5.26) și păstrînd în expresie pe 1 logic minim sub forma \underline{U}_{cbl} , se obține :

$$\begin{aligned}
 (5.27) \quad &\underline{R}_B \geq \\
 &\frac{\bar{E}_B + \bar{U}_{B_s}}{\frac{U_{C_{bl}} - \bar{U}_{B_s}}{\bar{R}_A} - \frac{\bar{U}_{B_s} - U_{C_s}}{\bar{R}_A} (n-1) - \frac{1}{h_{21E}} \left(\frac{E_c - U_{C_s}}{\bar{R}_c} + m \frac{\bar{U}_{B_s} - U_{C_s}}{\bar{R}_A} \right)}
 \end{aligned}$$

Ca și în cazul circuitului inversor limitele domeniului de funcționare sînt determinate luînd semnul egal în relațiile (5.25) și (5.27). Coordonatele punctelor din planul (R_B, R_A) delimitat de cele două frontiere satisfac în cele mai defavorabile situații condiții de blocare și de saturare pentru tranzistor. Dar, de această dată mărimea domeniului de funcționare nu depinde numai

de E_c , E_B și R_c și de numărul de intrări și numărul de ieșiri. Se observă că creșterea lui n și m conduce la micșorarea domeniului de funcționare, deci la o îndeplinire mai dificilă a condițiilor de blocare și de saturare decât la circuitul inersor.

Estimarea numărului maxim de ieșiri. În cazul circuitelor logice interesează numărul maxim de circuite similare care pot fi comandate de la ieșirea unică a unui circuit în condițiile unei funcționări sigure și corecte. Pentru estimarea numărului maxim de ieșiri se folosește relația (5.26) din care rezultă :

$$m \leq \bar{m} = \frac{\underline{E}_c - \bar{I}_{CB0} \bar{R}_c - \bar{U}_{Bs}}{\frac{\bar{R}_c}{R_B} (\bar{E}_B + \bar{U}_{Bs}) + \frac{\bar{R}_c}{R_A} (\bar{U}_{Bs} - \underline{U}_{Cs})(n-1) + \frac{\bar{R}_c}{h_{21E}} \left(\frac{\underline{E}_c - \underline{U}_{Cs}}{R_c} + \frac{\bar{U}_{Bs} - \underline{U}_{Cs}}{R_A/m} \right)} \cdot \frac{\bar{R}_A}{R_c}$$

Dacă într-o primă aproximație se neglijează \underline{U}_{Cs} , \underline{U}_{Bs} și $\bar{I}_{CB0} R_c$ în comparație cu \underline{E}_c și \bar{E}_B , se obține :

$$m \leq \bar{m} = \frac{\underline{E}_c}{\frac{\underline{E}_c}{h_{21E}} + \frac{\bar{R}_c \bar{E}_B}{R_B}} - \frac{\bar{R}_A}{R_c} = \frac{h_{21E}}{\frac{\bar{R}_c \bar{E}_B}{R_B \underline{E}_c} h_{21E} + 1} - \frac{\bar{R}_A}{R_c}$$

În relația obținută se poate considera că $\underline{E}_c / \bar{R}_c = \bar{I}_{Cs}$, iar $\bar{E}_B / R_B \cong \bar{I}_{CB0}$. Dacă $\bar{I}_{Cs} \gg \bar{I}_{CB0}$, valoarea maximă a numărului de ieșiri va fi :

$$m \leq \bar{m} = h_{21E} - \frac{\bar{R}_A}{R_c}$$

Întrucît $R_A / R_c > 1$, rezultă că numărul maxim de ieșiri nu poate depăși valoarea minimă a factorului static de amplificare în curent. Din relațiile scrise mai sus rezultă că prin creșterea lui \underline{E}_c și h_{21E} crește numărul de ieșiri pentru un număr de intrări dat. De asemenea, micșorarea toleranțelor surselor și pieselor componente conduce la mărirea lui m .

Regimul tranzitoriu. Regimul tranzitoriu al circuitului RTL cu mai multe intrări este similar cu al circuitului de negare. În cazul cînd se folosesc mai multe etaje are loc o întîrziere a declanșării diverselor etaje din cauza duratei finite a fronturilor ; fiecare etaj este declanșat după o anumită modificare

a tensiunii de colector a etajului precedent. Dacă se consideră că tranzistorul următor se saturează când cel precedent se blochează complet, întârzierea pentru două etaje este $t_{f1} + t_r + t_{f2}$ și deci pe etaj va fi $\frac{t_{f1}}{2} + \frac{t_r + t_{f2}}{2}$. Mai corect este însă a

se considera că un tranzistor se saturează încă din momentul în care tensiunea de colector a tranzistorului precedent a ajuns la jumătatea valorii finale. În acest caz întârzierea totală pentru k etaje va fi $k \frac{t_{f1}}{2} + k \frac{t_r + t_{f2}}{2}$. Relațiile de mai sus permit nu-

mai o evaluare a ordinului de mărime a întârzierilor în transmisia informației.

Pentru circuitele cu un număr mare de intrări mărirea vitezei de acționare cu ajutorul condensatoarelor este adesea neindicată. Astfel, dacă la toate cele n intrări acționează semnale 1, iar la un moment dat la k intrări reprezentând majoritatea ($n/2 < k \leq n-1$) semnalele trec în 0, prin baza tranzistorului va trece un curent invers datorat descărcării celor k condensatoare. În aceste caz este posibilă o blocare temporară a tranzistorului, deși la una sau mai multe intrări acționează semnal 1. În colectorul tranzistorului va apare un semnal 1 fals care poate conduce la perturbarca funcționării întregii scheme. Din acest motiv se folosesc condensatoare de accelerare numai pentru circuite cu maximum două intrări.

Mărirea vitezei de acționare în circuitele *RTL* cu număr mare de intrări ($n > 2$) este posibilă prin folosirea unor tranzistoare de înaltă frecvență sau prin folosirea unor montaje care să nu permită intrarea în saturație profundă a tranzistoarelor. În ultimul caz se obține o reducere a timpului de resorbție. În fig. 5.24 sînt prezentate două scheme care asigură funcționarea nesaturată a tranzistoarelor: cu sursă de fixare (fig. 5.24a) și cu reacție negativă (fig. 5.24b). În primul caz potențialul colectorului se fixează la valoarea $E_f \ll E_c$, însă $E_f > U_{cs}$. Pe măsura deschiderii tranzistorului, potențialul colectorului scade și cînd $u_c < E_f$ dioda D se deschide și potențialul colectorului se fixează la un nivel apropiat de E_f . În al doilea caz, atît timp cît tensiunea bază emitor este mai mică decît tensiunea colector-emitor, dioda este blocată și reacția negativă nu acționează. Atunci cînd $U_{Bs} \geq U_c$, creșterea curentului i_{in} influențează puțin regimul

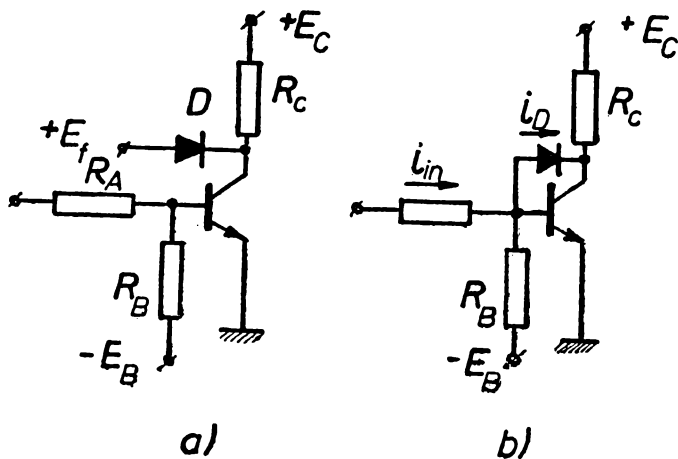


Fig. 5.24

tranzistorului deoarece o parte însemnată a acestui curent trece prin diodă. În acest mod se evită și în acest caz saturarea profundă a tranzistorului.

5.2.1.1. Proiectarea circuitelor RTL

Conform celor arătate în paragrafele precedente, în cadrul proiectării se va urmări obținerea numărului maxim de ieșiri (m) pentru un număr dat de intrări (n), cu condiția asigurării saturării și blocării tranzistorului în situațiile cele mai dezavantajoase din punctul de vedere al interconectării cu alte circuite similare și al toleranțelor surselor și componentelor.

Metodologia de calcul a unui circuit RTL se conduce după cum urmează :

1. În funcție de sursele de alimentare (E_c și E_B se aleg între 12 și 24 V) se alege tranzistorul astfel ca U_{CE} admisibil $> E_c$. De asemenea, se aleg toleranțele pentru surse și componente (δ_E și δ_R) și se stabilește domeniul de temperatură în care circuitul trebuie să funcționeze sigur.

2. Din condiția ca numărul de ieșiri să fie cât mai mare rezultă necesitatea unui curent de colector cât mai mare. Dar, în același timp creșterea curentului de colector atrage o viteză redusă de acționare și un consum mare de energie. Un compromis între număr mare de ieșiri și viteză / consum acceptabile se obține

prin adoptarea unui curent de colector $I_{c_{opt}} \cong 10$ mA. Se stabilește $\bar{R}_c \geq E_c / I_{c_{opt}}$.

Condiția ca m să fie cât mai mare impune ca $E_c \geq U_{cb1}$. Dar, în același timp este de dorit ca valoarea minimă pentru semnalul logic 1, U_{cb1} să fie mult mai mare decât tensiunile reziduale ale tranzistorului. Situația optimă se realizează pentru $U_{cb1, opt} \cong E_c / 2$.

3. Din condiția de realizare a valorii \bar{U}_{cb1} pentru numărul maxim de ieșiri, conform fig. 5.23 se obține:

$$\bar{U}_{cb1} = \frac{E_c \bar{R}_A + m \bar{R}_c \bar{U}_{Bs} - \bar{I}_{CB0} \bar{R}_A \bar{R}_c}{m \bar{R}_c + \bar{R}_A}$$

Având în vedere că trebuie asigurată valoarea U_{cb1} optimă, din relația precedentă se obține:

$$(5.28) \quad \frac{m}{R_A} = \frac{(1 - \delta_u) E_c (1 - \delta_E) - U_{cb1} - I_{CB0} R_c (1 + \delta_R)}{R_c (U_{cb1} - \bar{U}_{Bs})}$$

4. Din condiția de saturare pentru situația cea mai dezavantajoasă se stabilește valoarea curentului în baza tranzistorului:

$$I_B \geq \bar{I}_{Bs} = \frac{1}{h_{21E}} \left[\frac{(1 + \delta_E) E_c - \bar{U}_{Cs}}{(1 - \delta_R) R_c} + m \frac{\bar{U}_{Bs} - \bar{U}_{Cs}}{(1 - \delta_R) R_A} \right]$$

Valoarea definitivă a curentului în bază se stabilește din relația

$$I_B = s \bar{I}_{Bs}$$

în care s este valoarea factorului de suprainjecție și care, pentru o funcționare sigură, se ia $s = 2 - 2,5$.

5. Având în vedere (5.26), expresia curentului minim în bază pentru valoarea U_{cb1} stabilită se poate scrie:

$$(5.29) \quad I_B = \frac{U_{cb1} - \bar{U}_{Bs}}{\bar{R}_A} - \frac{\bar{U}_{Bs} - \bar{U}_{Cs}}{\bar{R}_A} (n - 1) - \frac{\bar{E}_B + \bar{U}_{Bs}}{\bar{R}_B}$$

Impunând ca valoarea minimă a acestui curent să fie cea stabilită la punctul 4 și folosind (5.25), se poate determina R_A :

$$(5.30) \quad R_A \leq \frac{U' - k(U_{Bb1} + \bar{U}_{Cs})n}{I_{CB0}(1 - \delta_R)k + I_B}$$

în care

$$U' = \frac{(U_{cb1} - \bar{U}_{Bs})(1 - \delta_R) - (\bar{U}_{Bs} - \bar{U}_{Cs})(n-1)(1 + \delta_R)}{1 - \delta_R^2}$$

și

$$k = \frac{\bar{E}_B + \bar{U}_{Bs}}{\bar{E}_B - U_{Bbl}} \frac{(1 + \delta_R)}{(1 - \delta_R)^2}$$

6. Din relația (5.25) se calculează R_B :

$$R_B \leq \frac{1 - \delta_R}{1 + \delta_R} \frac{R_A(E_B - U_{Bbl})}{\bar{I}_{CBO} R_A (1 - \delta_R) + n(U_{Bbl} + \bar{U}_{Cs})}$$

7. Cunoscînd m/R_A de la punctul 3 și valoarea lui R_A se calculează numărul de ieșiri ale circuitului proiectat.

Exemplu. Să se proiecteze un circuit *RTL-NICI* cu $n=5$ folosind un tranzistor *BC 107A* cu $h_{21E} = 90$, $\bar{U}_{Cs} = 0,09$ V, $\bar{U}_{Cs} = 0,2$ V, $\bar{U}_{Bs} = 0,5$ V, $\bar{U}_{Bs} = 0,8$ V, $\bar{I}_{CBO} = 0,2$ nA (la 25°C). Circuitul trebuie să funcționeze în domeniul de temperatură $0-55^\circ\text{C}$, avînd $\delta_E = \delta_R = 5\%$.

1. Se alege $E_c = E_B = 24$ V, $I_C = 10$ mA. Rezultă $R_c = 2,4$ k Ω .
2. Pentru $U_{cb1} = E_c/2 = 12$ V, rezultă :

$$\frac{m}{R_A} = \frac{0,95}{1,05} \frac{24 \cdot 0,95 - 12 - 4 \cdot 10^{-6} \cdot 2,4 \cdot 1,05}{2,4(12 - 0,5)} = 0,354,$$

în care $\bar{I}_{CBO_{55^\circ}} = I_{CBO_{25^\circ}} \cdot 2^{(55-25)/7} = 4$ nA — la componentele din Si curentul rezidual se dublează la fiecare creștere cu 7°C .

3. Valoarea maximă a curentului necesar în bază la saturație :

$$\bar{I}_{Bs} = \frac{1}{90} \left[\frac{24 \cdot 1,05 - 0,09}{2,4 \cdot 0,95} + (0,8 - 0,09) \cdot 0,354 \right] = 0,125 \text{ mA},$$

de unde $I_B = \beta I_{Bs} = 2,5 \cdot 0,125 = 0,312$ mA.

4. Folosind relația (5.30) rezultă :

$$R_A \leq \frac{7,68 - 1,34 \cdot 5(0,15 + 0,2)}{0,312 + 1,34 \cdot 0,95 \cdot 4 \cdot 10^{-6}} = 17 \text{ k}\Omega.$$

în care $U_{Bbl} = 0,15$ V,

$$U' = \frac{(12 - 0,8) \cdot 0,95 - (0,8 - 0,09) \cdot 4 \cdot 1,05}{1 - 0,05^2} = 7,68 \text{ V,}$$

$$k = \frac{24 \cdot 1,05 + 0,8}{24 \cdot 0,95 - 0,15} \cdot \frac{1,05}{0,95^2} = 1,34.$$

Se alege $R_A = 15 \text{ k}\Omega$.

5. Se calculează R_B :

$$R_B \leq \frac{0,95}{1,05} \frac{15(24 \cdot 0,95 - 0,15)}{4 \cdot 10^{-6} \cdot 15 \cdot 0,95 + 5(0,15 + 0,2)} = 176 \text{ k}\Omega.$$

Se alege valoarea normalizată $R_B = 160 \text{ k}\Omega$.

6. Numărul de ieșiri $m = m/R_A \cdot R_A = 0,354 \cdot 15 = 5,31$.

Deci circuitul NCI proiectat cu 5 intrări poate comanda 5 circuite similare, avînd domeniul de funcționare pentru 1 logic cuprins între $U_{cb1} \div E_c = 12 \div 24 \text{ V}$.

5.2.2. Circuite logice cu tranzistoare și diode (DTL)

Combinarea circuitelor cu diode cu tranzistoare funcționînd în regim de comutație este mult utilizată în prezent. Aceste circuite logice au o viteză de acționare mai mare decît a circuitelor *RTL*. Din acest motiv tranzistoarele din circuitele *DTL* pot fi cu un ordin de mărime mai coborît în ceea ce privește frecvența de lucru decît pentru circuitele *RTL*. În mod obișnuit se asociază atît circuite SAU cît și circuite SI (sau combinații ale acestora) cu diode, cu tranzistoare (fig. 5.25).

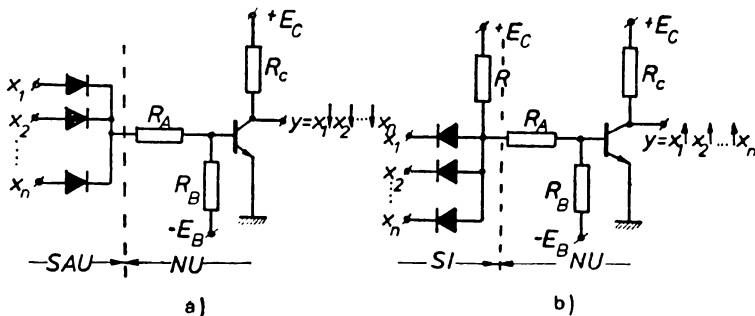


Fig. 5.25

În logică pozitivă circuitul din fig. 5.25a realizează negarea funcției SAU de n argumente, deci materializează funcția NICI :

$$y = \overline{x_1 \cup x_2 \cup \dots \cup x_n} = x_1 \downarrow x_2 \downarrow \dots \downarrow x_n,$$

iar circuitul din fig. 5.25b realizează negarea funcției SI de n argumente materializînd funcția NUMAI :

$$y = \overline{x_1 x_2 \dots x_n} = x_1 \uparrow x_2 \uparrow \dots \uparrow x_n.$$

Deoarece procesele în schemele NICI—DTL sînt similare cu cele din schemele NICI—RTL, în continuare se va analiza funcționarea circuitelor NUMAI—DTL.

Analiza funcționării circuitului DTL — NUMAI

Se consideră circuitul logic NUMAI cu n intrări și m ieșiri din fig. 5.26 comandat de circuite similare. Se va analiza funcționarea circuitului în regim staționar.

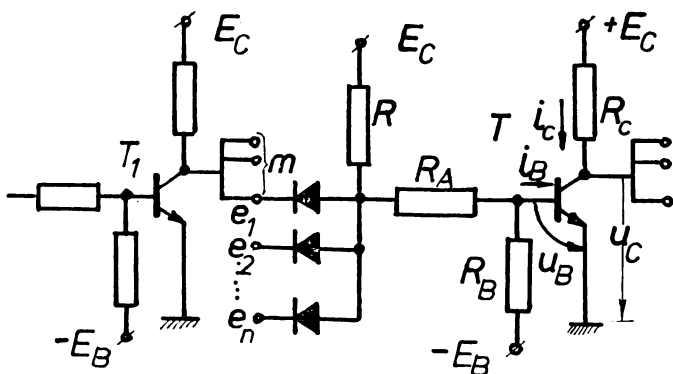


Fig. 5.26

a) Prima stare staționară se obține cînd la cel puțin una din intrări se aplică semnal logic 0, de la unul din tranzistoarele de comandă T_1 saturate. În acest caz tranzistorul din circuitul studiat trebuie să fie blocat și în cele mai dezavantajoase condiții.

Din punctul de vedere al interconectării cu circuitele de comandă, cea mai dezavantajoasă situație este atunci cînd un singur circuit de comandă are tranzistorul saturat, celelalte $n-1$ fiind blocate. Schema echivalentă pentru situația consi-

derată este prezentată în fig. 5.27. Conform cu această schemă se pot scrie următoarele ecuații de echilibru :

$$\begin{cases} E_B = U_{BE} + I R_B \\ U_2 = I_2 R_A - U_{BE} \\ I = I_2 + I_{CB0} \end{cases}$$

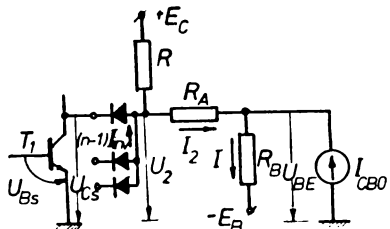


Fig. 5.27

Avînd în vedere că $U_2 = U_D + U_{Cs}$, în care U_D este căderea de tensiune pe dioda în conducție, se obține :

$$U_{BE} = \frac{E_B R_A - (U_D + U_{Cs}) R_B - I_{CB0} R_A R_B}{R_A + R_B}$$

Blocarea tranzistorului trebuie asigurată și pentru cazul cel mai defavorabil în ceea ce privește dispersia parametrilor :

$$(5.31) \quad \underline{U_{BE}} = \frac{E_B \underline{R_A} - (\bar{U}_D + \bar{U}_{Cs}) \bar{R}_B - \bar{I}_{CB0} \underline{R_A} \bar{R}_B}{\underline{R_A} + \bar{R}_B} \geq U_{Bb1}$$

b) A doua stare staționară, cu tranzistorul saturat, se obține cînd toate intrările circuitului sînt conectate la circuite de comandă cu tranzistoare blocate (fig. 5.28). Din schema echivalentă se observă că se consideră cea mai dezavantajoasă situație din punctul de vedere al interconectării cînd la toate cele n intrări se aplică semnal 1 logic de nivel minim (toate tranzistoarele de comandă sînt încărcate și pe celelalte $m - 1$ ieșiri). De asemenea,

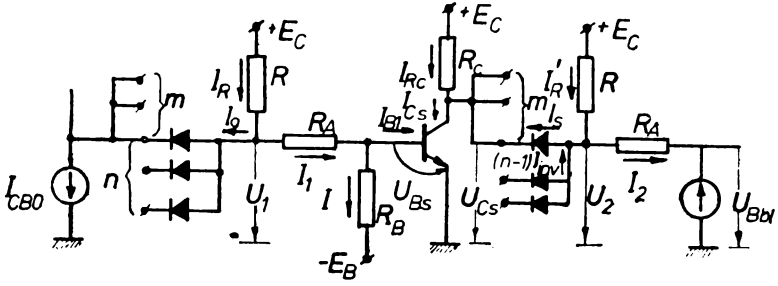


Fig. 5.28

tranzistorul din circuitul studiat este încărcat pe toate cele m ieșiri, comandînd circuite similare. Pentru saturarea tranzistorului trebuie îndeplinită condiția :

$$(5.32) \quad h_{21E} I_{B1} = h_{21E} (I_1 - I) \geq I_{Cs} = I_{Rc} + m I_s = I_{Rc} + m I'_R - I_2 + (n-1) I_{inv}.$$

Dar, conform schemei echivalente

$$I_1 = \frac{U_1 - U_{Bs}}{R_A} = \frac{E_c - (I_0 + I_1)R - U_{Bs}}{R_A}, \text{ de unde}$$

$$I_1 = \frac{E_c - I_0 R - U_{Bs}}{R_A + R}$$

Înlocuind pe I_1 în (5.32) și ținînd cont de schema echivalentă, pentru cele mai dezavantajoase condiții se obține :

$$(5.33) \quad \begin{aligned} h_{21E} \frac{E_c - \bar{I}_0 \bar{R} - \bar{U}_{Bs}}{\bar{R}_A + \bar{R}} - h_{21E} \frac{\bar{E}_B + \bar{U}_{Bs}}{\bar{R}_B} &\geq \frac{E_c - U_{Cs}}{R_c} + \\ + m \frac{E_c - (U_D + U_{Cs})}{R} - m \frac{U_D + U_{Cs} + U_{Bb1}}{\bar{R}_A} + m(n-1) \bar{I}_{inv}. \end{aligned}$$

În relația (5.33) valoarea maximă pentru I_0 se obține atunci cînd toți cei $n(m-1)$ curenți inverși ai diodelor și curenții reziduali ai celor n tranzistoare de comandă trec prin R :

$$(5.34) \quad \bar{I}_0 \cong n [\bar{I}_{CB0} + (m-1) \bar{I}_{inv}].$$

Din (5.31) și (5.33), la limită, se obțin ecuațiile pentru frontierele domeniului de funcționare în planul (R_A, R_B) . Alegînd convenabil pe R_c (astfel ca I_c să fie în zona $h_{21E} \cong \bar{h}_{21E}$) și R (astfel ca I_R — fig. 5.28 — să fie cît mai mare pentru un consum acceptabil), se determină R_A și R_B astfel ca să fie îndeplinite condițiile (5.31) și (5.33).

5.2.2.1. Proiectarea circuitelor DJL

În continuare se va prezenta proiectarea celor două tipuri de circuite logice DTL mai mult folosite.

Proiectarea circuitelor DTL—NICI

Proiectarea circuitelor DTL—NICI este apropiată de a circuitelor RTL—NICI și se poate efectua în următoarea ordine :

1. În funcție de tipul tranzistorului utilizat (U_{CE} admisibil) se aleg E_c și E_B între 12 și 24 V. De asemenea, se aleg tipul de diodă (U_{inv} admisibil să fie mai mare decât $\bar{U}_{inv} \cong E_c$) și toleranțele δ_E, δ_R . Se stabilește domeniul de temperatură în care trebuie să funcționeze circuitul și numărul de intrări.

2. Funcționarea optimă a circuitului se obține pentru semnal 1 minim, $U_{Cb1} = (3-4) V \dots E_c/2$ și $I_{Copt} \cong 10$ mA. Se calculează R_c .

3. Se calculează curentul necesar în bază pentru saturarea tranzistorului în condițiile cele mai dezavantajoase :

$$(5.35) \quad \bar{I}_{B_s} = \frac{1}{h_{21E}} \left(\frac{\bar{E}_c}{R_c} + m I_{inv} \right),$$

unde \bar{I}_{inv} este curentul invers al unei diode la temperatura de lucru maximă și la tensiunea inversă din circuit ($\bar{U}_{inv} \cong E_c$).

În relația (5.35) se estimează $m=6$, estimare care nu introduce o eroare mare deoarece $m \bar{I}_{inv} \ll h_{21E} \bar{I}_{B_s}$. Se stabilește curentul în bază $I_B = s \bar{I}_{B_s}$, cu $s = 2 - 2,5$.

4. Se calculează rezistorul R_A dintr-o relație de forma celeia obținută pentru circuitul RTL—NICI (relația 5.30) :

$$(5.36) \quad R_A \leq \frac{U' - k(U_{Bb1} + \bar{U}_{C_s} - \underline{U}_D)}{\bar{I}_{CB0}(1 - \delta_R)k + I_B},$$

în care

$$U' = \frac{U_{Cb1} - (\bar{U}_D + \bar{U}_{B_s})}{1 + \delta_R}, \quad k = \frac{\bar{E}_B + \bar{U}_{B_s}}{\bar{E}_B + U_{Bb1}} \frac{1 + \delta_R}{(1 - \delta_R)^2}, \text{ iar } U_D \text{ este}$$

căderea de tensiune pe dioda în conducție (valoarea maximă se stabilește pentru $I_D \cong 1 \text{ mA} \cong I_B$).

5. Din condiția de blocare a tranzistorului rezultă, ca și pentru circuitul RTL, valoarea pentru R_B :

$$(5.37) \quad R_B \leq \frac{1 - \delta_R}{1 + \delta_R} \frac{R_A(\bar{E}_B - U_{Bb1})}{\bar{I}_{CB0} R_A (1 - \delta_R) + U_{Bb1} + \bar{U}_{C_s} - \underline{U}_D}.$$

6. Numărul maxim de ieșiri pe care poate să-l asigure circuitul proiectat rezultă din condiția ca U_{Cbl} să nu scadă sub valoarea optimă impusă :

$$(5.38) \quad m \leq \frac{E_c - U_{Cbl} - \bar{I}_{CBO} R_C (1 + \delta_R)}{\left[\frac{U_{Cbl} - (\underline{U}_D + \underline{U}_{Bs})}{R_A (1 - \delta_R)} + (n-1) \bar{I}_{inv} \right] R_C (1 + \delta_R)} .$$

Exemplu. Să se proiecteze un circuit DTL—NICI cu $n = 5$ folosind un tranzistor BC 107 A cu $h_{21E} = 90$, $\underline{U}_{Cs} = 0,09V$, $\bar{U}_{Cs} = 0,2V$, $\underline{U}_{Bs} = 0,5V$, $\bar{U}_{Bs} = 0,8V$, $I_{CBO} = 0,2nA$ (la $25^\circ C$) și diode EFD 105. Circuitul trebuie să funcționeze în domeniul de temperatură $0-55^\circ C$, avînd $\delta = \delta_R = 5\%$.

1. Se aleg $E_c = E_B = 24V$, $I_C = 10$ mA, $U_{Cbl} = 10$ V. Rezultă $R_C = 2,4$ k Ω .

2. Estimînd $m = 6$, cu relația (5.35) se calculează

$$\bar{I}_{Bs} = \frac{1}{90} \left(\frac{24 \cdot 1,05}{2,4 \cdot 0,95} + 5 \cdot 100 \cdot 10^{-3} \right) = 0,128 \text{ mA},$$

unde $\bar{I}_{inv} = 100$ μA la $U_{inv} = 30$ V și $T_{amb} = 5^\circ C$.

Se stabilește valoarea curentului în bază $I_B = s \bar{I}_{Bs} = 2,5 \cdot 0,128 = 0,32$ mA.

3. Cu relația (5.36) se calculează R_A :

$$R_A \leq \frac{8,5 - 1,34(0,2 + 0,2 - 0)}{4 \cdot 10^{-6} \cdot 0,95 \cdot 1,34 + 0,32} = 24,9 \text{ k}\Omega,$$

$$\text{unde } U' = \frac{10 - (0,3 + 0,8)}{1,05} = 8,5V, \quad k = \frac{24 \cdot 1,05 + 0,8}{24 \cdot 0,95 - 0,2} \cdot \frac{1,05}{0,95^2} = 1,34,$$

în care $U_{Bbl} = 0,2$ V, $\bar{I}_{CBO} = 4 \cdot 10^{-6}$ mA (la $55^\circ C$) și $\underline{U}_D = 0$, $\bar{U}_D = 0,3V$ pentru $I_D = 1$ mA. Se alege valoarea normalizată $R_A = 24$ k Ω .

4. Folosind (5.37) se calculează R_B :

$$R_B \leq \frac{0,95}{1,05} \frac{24(24 \cdot 0,95 - 0,2)}{4 \cdot 10^6 \cdot 0,95 \cdot 24 + 0,2 + 0,2 - 0} = 1,13 \text{ M}\Omega.$$

Se alege valoarea normalizată $R_B = 1$ M Ω .

5. Numărul maxim de ieșiri se calculează cu relația (5.38) :

$$m \leq \frac{24.0,95 - 10 - 4.10^{-6} \cdot 1,05 \cdot 2,4}{\left[\frac{10 - (0 + 0,5)}{24.0,95} + 4.100 \cdot 10^{-3} \right] 2,4 \cdot 1,05} = 6,22.$$

Proiectarea circuitelor DTL-NUMAI

Din relația (5.31), se obține :

$$(5.39) \quad R_B \leq \frac{1 - \delta_R}{1 + \delta_R} \frac{R_A (\underline{E}_B - U_{Bbl})}{\bar{I}_{CB0}(1 - \delta_R) + U_{Bbl} + \bar{U}_D + \bar{U}_{Cs}},$$

în care \bar{U}_D se stabilește pentru $I_D \cong E_c/R$.

Condiționînd ca potențialul U_1 la saturare (v. fig. 5.28) să nu scadă sub o valoare impusă

$$\underline{U}_1 = \frac{\underline{E}_c \underline{R}_A - \bar{I}_0 \bar{R} \underline{R}_A + \underline{U}_{Bs} \bar{R}}{\underline{R}_A + \bar{R}} \geq U_1$$

și ținînd cont de (5.34), se obține :

$$(5.40) \quad R_A \geq \frac{1 + \delta_R}{1 - \delta_R} \frac{U_1 - \underline{U}_{Bs}}{\frac{E_c - U_1}{R} - n[\bar{I}_{CB0} + (m - 1)\bar{I}_{inv}](1 + \delta_R)}.$$

Din relația (5.40) rezultă că pentru obținerea unor valori pozitive pentru R_A , avînd un număr convenabil de ieșiri și intrări, este bine ca E_c/R să fie cît mai mare, iar I_{CB0} și I_{inv} cît mai mici (de preferat componentele cu siliciu). Ca un compromis între economicitate și capacitate de sarcină se acceptă $\frac{E_c}{R} \cong 10 \text{ mA}$, iar

$$U_{1opt} \cong E_c/2.$$

La saturare curentul minim în baza tranzistorului se poate scrie în funcție de potențialul U_1 (v. fig. 5.28) :

$$\underline{I}_{B1} = \frac{U_1 - \bar{U}_{Bs}}{\underline{R}_A} + \frac{\bar{E}_B + \bar{U}_{Bs}}{\underline{R}_B}$$

Avînd în vedere condiția de saturare (5.33), în care se poate neglija curentul foarte mic I_B , se poate stabili numărul maxim de ieșiri :

$$(5.41) \quad m \leq \frac{h_{21E} \left[\frac{U_1 - \bar{U}_{B_s}}{R_A(1 + \delta_R)} - \frac{\bar{E}_B + \bar{U}_{B_s}}{R_B(1 - \delta_R)} \right] - \frac{E_c - U_{C_s}}{R_c(1 - \delta_R)}}{\frac{E_c - (U_D + U_{C_s})}{R(1 - \delta_R)} + (n-1)\bar{I}_{inv}}$$

Din relația (5.41) se observă că m este cu atât mai mare cu cât R_c este mai mare. Din acest motiv I_C se alege astfel ca $h_{21E} \cong \bar{h}_{21E}$ și să fie în jurul a (10–20)% din valoarea curentului prin rezistorul R .

În condițiile menționate se poate efectua proiectarea circuitului NUMAI, estimînd în relația (5.40) pe m în jurul valorii 6.

Exemplu. Să se proiecteze un circuit logic DTL–NUMAI în logică pozitivă cu $n=6$ folosind un tranzistor BC 107 A cu $h_{21E}=90$, $\bar{U}_{C_s}=0,09V$, $\bar{U}_{C_s}=0,2V$, $\bar{U}_{B_s}=0,5V$, $\bar{U}_{B_s}=0,8V$. Circuitul trebuie să funcționeze între 0 și 55°C cu $\delta_E=\delta_R=5\%$. Se vor folosi diode de comutație EFD 105 cu $\bar{I}_{inv}=100 \mu A$. Pentru temperatura maximă de lucru $\bar{I}_{C_{B0}}=4 \cdot 10^{-6} mA$.

1. Se alege $E_c=E_B=24 V$, $E_c/R=10 mA$, $E_c/R_c=(1 \dots 2) mA$ și $U_1=E_c/2=12 V$. Rezultă $R=2,4 k\Omega$ și $R_c=(24 \dots 12) k\Omega$. Se alege $R_c=18 k\Omega$.

2. Estimînd $m=6$, din relația (5.40), rezultă :

$$R_A \geq \frac{1,05}{0,95} \frac{12 - 0,5}{\frac{24 \cdot 0,95 - 12}{2,4} - 6(4 \cdot 10^{-6} + 5 \cdot 100 \cdot 10^{-3})1,05} = 9,42 k\Omega.$$

Se alege valoarea standardizată $R_A=10 k\Omega$.

3. Folosind relația (5.39) se determină R_B :

$$R_B \leq \frac{0,95}{1,05} \frac{10(24 \cdot 0,95 - 0,2)}{4 \cdot 10^{-6} \cdot 10 \cdot 0,95 + 0,2 + 1,5 + 0,2} = 108 k\Omega,$$

în care $\bar{U}_D=1,5 V$ (pentru $I_D=E_c/R=10 mA$) și $U_{B_{bl}}=0,2 V$. Se alege valoarea normalizată $R_B=100 k\Omega$.

4. Cu relația (5.41) se stabilesc numărul de ieșiri :

$$m \leq \frac{90 \left(\frac{12 - 0,8}{10 \cdot 1,05} - \frac{24 \cdot 1,05 + 0,8}{100 \cdot 0,95} \right) - \frac{24 \cdot 0,95 - 0,09}{18 \cdot 0,95}}{\frac{24 \cdot 0,95 - (0 + 0,09)}{2,4 \cdot 0,95} + 5 \cdot 100 \cdot 10^{-3}} = 6,7.$$

5.2.3. Circuite logice cu tranzistoare cu cuplaj direct

În cazul circuitelor logice cu cuplaj direct cu tranzistoare (DCTL), tranzistorul este folosit atât pentru amplificare cât și pentru realizarea funcțiilor logice SI și SAU. Realizarea funcțiilor logice cu astfel de circuite se face după principiul schemelor cu contacte de releu, întrucât un tranzistor funcționând în regim de comutație poate fi asimilat în circuitul colector-emitor cu un contact închis la saturație și deschis la blocare. În fig. 5.29 sînt prezentate schemele circuitelor logice care materializează func-

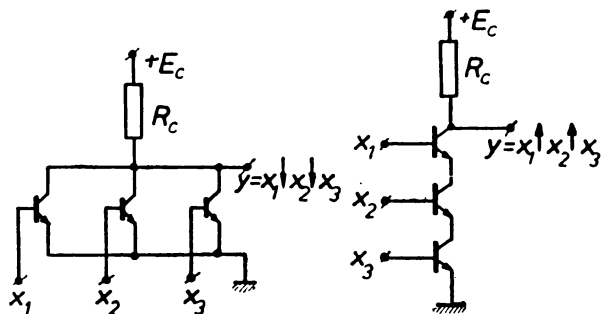


Fig. 5.29

țiile NICI și NUMAI pentru trei variabile prin conectarea în paralel și respectiv în serie a tranzistoarelor. Prin folosirea montajelor repetor pe emitor se pot obține și funcții SI și SAU, dar cel mai avantajos este a realiza sistemele complete monofuncționale de funcții.

Datorită similitudinii cu circuitele cu contacte, circuitelor DCTL li se mai spune circuite cu tranzistoare cu logică de contacte. Există însă o diferență importantă între circuitele DCTL și circuitele logice cu contacte: nu apare pericolul transmisiei bidirecționale a informației.

În schemele DCTL comanda tranzistoarelor se face direct pe bază de la colectoarele unor tranzistoare-contact similare. Posibilitatea de a comanda stările 0 și 1 prin cuplaj direct se bazează pe proprietatea tranzistoarelor (în special cele cu siliciu) de a prezenta la saturație un potențial de colector mai mic decât potențialul pe bază ($U_{C,s} \ll U_{B,s}$). Cu asemenea tranzistoare cuplate direct se realizează lanțuri în care se succed starea blocată și saturată. Principiul de funcționare se poate explica folosind

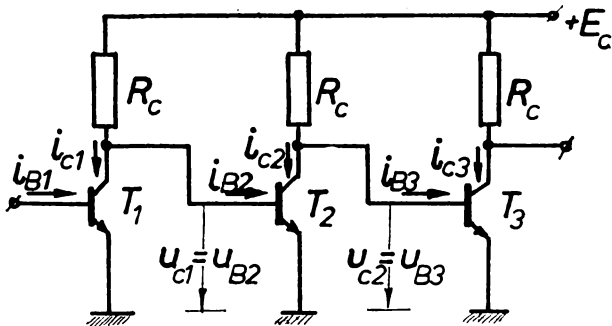


Fig. 5.30

schema din fig. 5.30. Se consideră T_1 deschis și saturat. Pentru un curent de bază i_{B1} de valoare însemnată tensiunea U_{Cs} are va'oare foarte mică (zeci de milivolți). Tensiunea de bază a tranzistorului T_2 fiind egală cu tensiunea de colector a tranzistorului T_1 , se realizează blocarea lui T_2 ($U_{CsT_1} \ll U_{BsT_1}$). Totuși, prin colectorul tranzistorului T_2 trece un curent mic $I_C > I_{CBO}$ deoarece $U_{Cs} > U_{Bb1}$, care asigură o polarizare directă mică a joncțiunii bază-emitor. Tranzistorul T_2 fiind blocat asigură tensiunea de saturație tranzistorului T_3 . În acest caz $u_{C2} = u_{B3} = = u_{Bs}$ și tensiunea pe colectorul tranzistorului nu va depăși valoarea u_{Bs} . Parametrii schemei trebuie aleși astfel încît să se asigure alternarea stărilor de blocare cu cele de saturare. Astfel, dacă T_2 este blocat T_3 trebuie să se satureze :

$$i_{B3} h_{21E} > i_{C3}.$$

Curentul în baza tranzistorului T_3 se poate considera că este aproximativ egal cu curentul care trece prin rezistorul de colector al tranzistorului T_2 blocat (se neglijează I_C), deci :

$$i_{B3} = \frac{E_c - u_{C2}}{R_c} = \frac{E_c - U_{Bs}}{R_c}.$$

Dacă nu se ia în considerare curentul de bază al tranzistorului următor blocat, condiția de saturare devine :

$$h_{21E} \frac{E_c - U_{Bs}}{R_c} > \frac{E_c - U_{Cs}}{R_c},$$

sau

$$(h_{21E}-1)E_c > h_{21E}U_{B_s} - U_{C_s}.$$

Deoarece $U_{C_s} \ll U_{B_s}$, treapta de tensiune la ieșirea unui astfel de circuit va fi $U_s = U_{B_s} - U_{C_s} \cong U_{B_s}$. Rezultă că pentru o funcționare corectă a acestor circuite este necesar ca

$$(5.42) \quad (h_{21E}-1)E_c > h_{21E}U_s.$$

Stabilirea parametrilor unui circuit DCTL poate fi făcută în următoarea ordine :

1. Se alege tensiunea de alimentare de valoare relativ redusă, dar totuși suficient de mare în raport cu tensiunile reziduale (v. relația 5.42). Obișnuit $E_c = 3 \dots 10V$.

2. Se alege valoarea curentului de colector la saturație din următoarele considerente :

— să fie cât mai mare pentru a se obține o capacitate de sarcină („fan out“) ridicată și totodată o tensiune U_{C_s} cât mai redusă ;

— cât mai mic pentru limitarea puterii disipate precum și pentru obținerea unei viteze de funcționare ridicate.

Compromisul pentru cele două condiții contradictorii se realizează pentru un curent de colector $I_{c_{opt}} = 3 \dots 5 \text{ mA}$.

3. Se alege tranzistorul cu h_{21E} cât mai mare pentru a avea U_{C_s} cât mai redusă și în funcție de frecvența de lucru.

$$4. \text{ Se determină rezistorul } R_c = \frac{E_c - U_{C_s}}{I_{c_{opt}}} \cong \frac{E_c}{I_{c_{opt}}}$$

5. Cunoscînd I_c , din caracteristicile tranzistorului se stabilesc U_{B_s} și U_{C_s} și deci și U_s .

Circuite DCTL complexe

Plecînd de la schemele de bază NICI sau NUMAI se pot realiza ușor circuite logice complexe în tehnică DCTL. În fig. 5.31 este prezentată schema unui circuit DCTL realizat pe baza circuitului NICI, cu n intrări și m ieșiri. Se urmărește să se obțină, și în acest caz, un număr cât mai mare de circuite similare care să poată fi conectate la un asemenea circuit. Dacă se consideră tranzistoarele T_1 blocate, curentul prin baza unuia din cele m tranzistoare conectate la ieșire va fi :

$$i_B = \frac{E_c - U_{B_s}}{mR_c}.$$

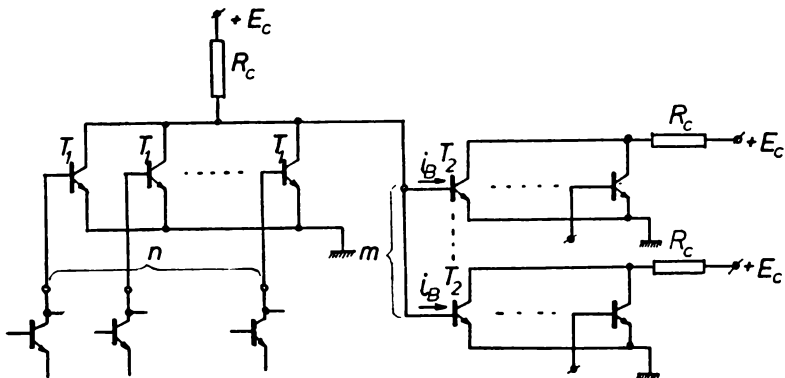


Fig. 5.31

Saturarea tranzistoarelor T_2 are loc dacă

$$h_{21E} \frac{E_c - U_{Bs}}{m R_c} > \frac{E_c - U_{Cs}}{R_c} \cong \frac{E_c}{R_c}$$

de unde rezultă

$$m < h_{21E} \left(1 - \frac{U_{Bc}}{E_c} \right)$$

Dacă se consideră $U_{Bs} = 0,6 \text{ V}$ și $E_c = 3 \text{ V}$, rezultă $m < 0,8 h_{21E}$. În realitate numărul de ieșiri este mult mai mic din cauza dispersiei caracteristicilor tranzistoarelor precum și a influenței temperaturii asupra curenților reziduali și mărimii lui h_{21E} . Practic, numărul maxim de ieșiri este de 2–3 mai mic decât valoarea minimă a lui h_{21E} .

În comparație cu celelalte tipuri de circuite logice circuitele DCTL prezintă următoarele *avantaje*: simplitate extremă datorită lipsei elementelor de cuplaj; folosirea unei singure surse de valoare coborâtă; disipație redusă în tranzistoare (cca. 1 mW) și în rezistoare; frecvență de lucru mult mai mare decât a celorlalte tipuri (tipic de ordinul megahertzilor). În schimb prezintă *dezavantajele*: sînt neeconomice datorită consumului mare de tranzistoare (un tranzistor pe intrare); cerințele impuse tranzistoarelor mult mai severe în ceea ce privește dispersia caracteristicilor și stabilitatea în timp; sînt sensibile la perturbații datorită nivelului scăzut al semnalelor logice.

5.3. Circuite logice integrate cu tranzistoare bipolare

În prezent se tinde tot mai mult să se înlocuiască circuitele logice realizate cu componente discrete (individuale) prin circuite realizate monolitic (cu corp solid). Introducerea acestor dispozitive conduce nu numai la micșorarea dimensiunilor, greutateii și puterii consumate ci și la creșterea substanțială a *siguranței în funcționare* a circuitelor. Ultimul avantaj a fost factorul decisiv care a condus la dezvoltarea și asimilarea circuitelor integrate.

În circuitele integrate (CI) nu există piese discrete separate, funcțiile acestora fiind îndeplinite de anumite domenii ale unui corp solid (de exemplu un cristal de siliciu), domenii care sînt analoge rezistoarelor, condensatoarelor, diodelor sau tranzistoarelor. Domeniile amintite ale corpului solid se formează prin procedee tehnologice speciale/20, 48/.

Primul circuit integrat a fost realizat în 1959 în S.U.A. și a constituit o copie exactă, în versiune integrată, a unui circuit logic cu componente discrete. În primii ani de fabricație industrială a CI (1960—1963) schemele electrice ale acestora erau copii ale circuitelor logice cu componente discrete. Din aproape în aproape tehnica CI a evoluat de la familia RTL prin RTCL (cu condensator de accelerare), DCTL, la diferite tipuri DTL. De la circuitele DTL au derivat circuitele logice integrate de tip TTL (Transistor-Transistor-Logic) care constituie baza circuitelor logice integrate din prezent cu tranzistoare bipolare. În cazul CI TTL (denumite uneori și T²L) diodele din circuitele DTL au fost înlocuite cu *tranzistoare multiemitor* care se pot realiza avantajos numai în tehnologie integrată. În fig. 5.32 sînt prezentate circuite logice DTL și TTL pentru construcție monolitică. În cazul circuitului DTL (fig. 5.32 a) se observă că

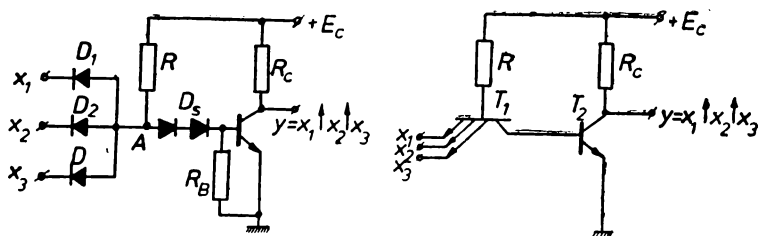


Fig. 5.32

rezistorul de limitare a curentului în bază a fost înlocuit cu diodele D_s . Acest lucru permite creșterea gradului de imunitate la perturbații, tranzistorul saturându-se numai atunci cînd se asigură în punctul A un potențial suficient pentru deschiderea diodelor D_s . În construcție TTL simplificată (fig. 5.32 b) locul diodelor de intrare și din bază îl iau joncțiunile tranzistorului multiemitor (convențional) T_1 : fiecare diodă bază-emitor constituie o intrare, iar dioda bază-colector funcționează ca diodă serie. Cele două circuite din fig. 5.32 realizează în logică pozitivă funcția NUMAI, care este predominant materializată în versiune integrată. Astfel, la circuitul TTL dacă toate intrările circuitului sînt la 1 logic (nivel apropiat de $+E_c$) diodele de intrare sînt blocate. Dioda bază-colector fiind polarizată direct, curentul injectat în baza tranzistorului (convențional) T_2 prin rezistorul R îl aduce în saturație și la ieșire se obține semnal 0 logic (apropiat de potențialul masei). Dacă cel puțin una din intrări este acționată cu semnal 0, dioda respectivă conduce și curentul din baza lui T_2 este comutat la masă. Din acest motiv aceste circuite sînt cunoscute ca funcționînd cu logică de comutare a curentului de bază sau prescurtat BCSL (Base Current Switching Logic). Ca urmare tranzistorul T_2 se blochează și la ieșire se obține semnal 1 logic.

Datorită cuplajului direct între tranzistoare, CI descrie lucrarea cu tensiuni de alimentare coborîte (tipic $E_c = \pm 5$ V). Pentru a avea și în acest caz o capacitate ridicată de sarcină la un număr acceptabil de intrări, în locul tranzistorului T_2 se folosește un montaj cu sarcină activă, cu funcționare în contra-timp denumit și „totem pole stage”. În fig. 5.33 este prezentată schema CI de tip SI-NU care stă la baza circuitelor logice actuale.

Diodele $D_1 \dots D_3$ realizează protejarea circuitului la semnale negative aplicate la intrare. Tranzistorul convențional T_2 are rol de defazor, asigurînd comanda în contra-timp a tranzistoarelor convenționale T_3 și T_4 . Rezultă că cele două tranzistoare nu trebuie să fie deschise simultan, ele funcționînd alternativ blocat-saturat.

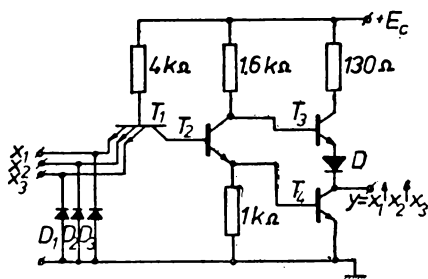


Fig. 5.33

1) În cazul cînd toate intrările sînt acționare cu semnale 1, diodele de intrare sînt blocate. În acest caz, prin joncțiunea bază-colector a tranzistorului multiemitor T_1 , se injectează în baza lui T_2 un curent suficient pentru saturare. Fiind în montaj receptor pe emitor, T_2 asigură saturarea și a tranzistorului T_4 și la ieșire se obține semnal 0. Colectorul tranzistorului T_2 prezintă un potențial determinat de tensiunile reziduale $U_{C_sT_2}$ și $U_{B_sT_4}$. Dacă pentru început nu se consideră dioda D , valoarea tensiunii bază-emitor a tranzistorului T_3

$$U_{BET_3} = U_{B_sT_2} + U_{C_sT_2} - U_{C_sT_4} \cong U_{B_sT_4} \cong 0,7 \text{ V},$$

este apropiată de valoarea de saturație. Este deci posibil ca și T_3 să se mențină în conducție. Pentru a evita această situație și a realiza o blocare sigură a lui T_3 s-a introdus o tensiune de prag prin dioda D , cînd tensiunea bază-emitor devine :

$$U_{BET_3} = U_{B_sT_2} + U_{C_sT_2} - (U_{C_sT_4} + U_D) \cong 0 \text{ V}.$$

2) Dacă la o intrare sau mai multe se aplică semnal 0, curentul din baza lui T_2 este comutat la masă prin tranzistorul multiemitor. Blocarea lui T_2 atrage și blocarea lui T_4 . Simultan, în baza tranzistorului T_3 se injectează un curent suficient pentru saturare. La ieșire se obține în acest caz semnal logic 1.

În regim tranzitoriu, determinat de trecerea semnalului la intrări din 0 în 1, toate tranzistoarele conduc. Pentru ca sursa de alimentare să nu fie scurtcircuitată prin T_3 - D - T_4 , s-a introdus rezistorul de 130 de ohmi.

Avantajele circuitelor TTL

Folosirea tranzistorului multiemitor asigură cîteva avantaje greu de obținut în alt mod. Cînd semnalul la orice intrare este 0, joncțiunea bază-emitor a tranzistorului T_1 este polarizată direct, fapt ce conduce la saturarea acestuia. În acest caz, prin T_1 se realizează o cale de impedanță minimă care asigură eliminarea rapidă a sarcinii stocate în baza lui T_2 . Rezultă un timp de comutare inversă mult mai bun decît la alte tipuri de circuite logice integrate. Un alt avantaj al folosirii tranzistorului multiemitor constă în înlocuirea combinațiilor de diode, rezistoare și tranzistoare din alte tipuri de circuite. Aceasta atrage o geometrie redusă cu capacități parazite mici și viteze mari de comutare. De asemenea, dimensiunea redusă conduce la un preț mai redus sau la posibilitatea realizării mai multor circuite pe

aceeași structură (cip) de Si. În tabelul 5.1 sînt prezentate comparativ diferite variante de circuite logice integrate cu principalele caracteristici: întârziere pe circuit (t), putere disipată (P), factor de calitate ($m \cdot 10^3 / tP - m$ fiind numărul de ieșiri), prag de perturbare, fan in (n), fan out (m) și integrabilitatea.

Tab. 5.1

Tipul circuitului logic	t (ns)	P (mW)	Factor de calitate $\left(\frac{m \cdot 10^3}{n \cdot mW} \right)$	Prag de perturbare (mV)	Fan in (n)	Fan out (m)	Integrabilitate
RTL	50	50	30	700	7	4	nu
DCTL	10	20	100	300	2	4	bun/mulțumitor
DTL	10	15	60	800	2	5	mulțumitor
TTL	5	15	120	750	7	20	bun/foarte bun

Din analiza tabelului se explică de ce în prezent cel mai construit tip de CI este TTL. Construcția etajului de ieșire în contratimp asigură o impedanță redusă a sursei de semnal. Aceasta explică timpul foarte redus de comutare și capacitatea mare de comandă. Chiar și în cazul sarcinilor capacitive impedanța redusă de ieșire asigură timpi de comutare foarte buni. În țara noastră, la I.P.R.S.—Băneasa, se construiesc CI TTL avînd la bază circuitul din fig. 5.33, în seria CDB. În afară de circuite SI-NU cu 2, 3, 4 și 8 intrări se construiesc CI care realizează funcțiile NU, SI și SI-SAU-NU. Alimentarea se realizează la $V_{cc} = +5V \pm 5\%$ și asigură un fan out de 10 pentru 0 logic și 20 pentru 1 logic. Semnalul 0 logic corespunde unor nivele de tensiune cuprinse între 0V și 0,8V la intrare și între 0V și 0,4V la ieșire, iar 1 logic corespunde unor nivele cuprinse între 2V și V_{cc} la intrare și 2,4V și V_{cc} la ieșire. Se realizează de asemenea și circuite avînd colectorul tranzistorului de ieșire în gol (cu posibilitatea conectării în exterior a rezistorului și realizarea funcției SI-cablat) precum și circuite de putere (asigură un curent la ieșire în starea 0 logic de 48 mA față de 16 mA la circuitele normale). Detalii privind performanțele statice și dinamice se obțin consultînd [48, 52]. Circuitele menționate sînt fabricate pentru realizarea echipamentelor numerice dispuse în medii cu nivel coborît al perturbațiilor. În industrie, datorită nivelului ridicat de perturbații aceste circuite nu sînt sigure în funcționare,

creșterea insensibilității realizându-se greu / 28, 48/. Din acest motiv în industrie se preferă și se utilizează circuite mai lente dar care să aibă un grad ridicat de imunitate la perturbații. În fig. 5.34 este prezentată schema unui astfel de circuit cu alimentare între 12 și 18 V și prag static de perturbare de 5–7 V ca urmare a înserării unei diode Zener. Aceste circuite sînt cunoscute sub denumirea de circuite DTLZ sau HLL (High-Level-Logic).

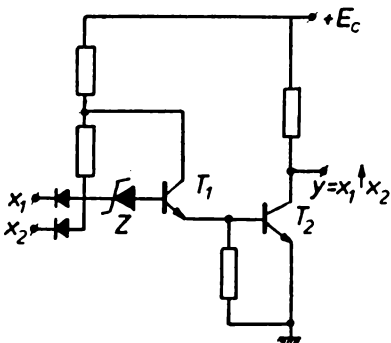


Fig. 5.34

Cînd la ambele intrări se aplică semnal 1 (apropiat de +E) T_1 și T_2 sînt deschise și la ieșire se obține semnal 0. La aplicarea la intrare a semnalului 0 cele două tranzistoare se blochează și la ieșire se obține semnal 1. La trecerea semnalului la intrare din 0 în 1 tranzistorul T_1 se deschide abia după ce se depășește tensiunea de deschidere a diodei Zener, asigurîndu-se astfel pragul de perturbare. Tranzistorul T_1 lucrînd ca repetor asigură o impedanță mare de intrare și o capacitate de sarcină ridicată. Reacția negativă colector-bază asigură stabilitatea circuitului la dispersia de fabricație a amplificării în curent a tranzistorului T_1 .

5.4. Circuite logice cu tranzistoare unipolare

După modul în care se realizează trecerea curentului electric, tranzistoarele se pot împărți în două categorii: *bipolare* și *unipolare*. Deosebirea dintre cele două tipuri constă în faptul că la tranzistoarele bipolare la conducția curentului electric participă atât electronii cît și golurile, în timp ce la tranzistoarele unipolare participă numai un tip de purtători-majoritari, fie electroni, fie goluri. La tranzistorul bipolar controlul curentului care îl străbate se realizează ca o consecință a unor fenomene de injecție în bază a purtătorilor minoritari și de difuzie a acestora către colector. În tranzistoarele unipolare, reprezentate prin *tranzistorul cu efect de cîmp* (TEC sau FET-Field Effect Transistor), controlul curentului se realizează cu ajutorul unui cîmp electric care modulează conductanța căii de trecere a curentului. În toate TEC actuale cîmpul electric este dirijat perpendicular pe

calea de trecere a curentului. Calea de trecere a curentului se realizează la suprafața sau în volumul unui semiconductor (de obicei Si) și se numește *canal*. Canalul poate fi de tip *n* sau tip *p*, TEC-ul purtînd numele corespunzător de *TEC cu canal n* sau *TEC cu canal p*. Acest canal reprezintă partea activă a unui TEC și este delimitat de două contacte ohmice care servesc la introducerea și extragerea curentului: *sursa* (*S*) și *drena* (*D*). Prin aplicarea unei diferențe de potențial celor două contacte, prin canal va circula un curent. Valoarea curentului este determinată de cîmpul electric aplicat din exterior perpendicular pe canal prin intermediul unui electrod denumit *grilă* sau *poartă* (*G-gate*).

După modul în care se obține un canal cu conductanță dependentă de cîmpul electric există două tipuri de TEC:

- cu grilă joncțiune (TEC-J sau JFET),
- cu grilă izolată.

La primul tip canalul se realizează în volumul semiconductorului, iar la al doilea la suprafață. Cel de al doilea tip de TEC se obține prin izolarea grilei față de substrat printr-un izolator. Din acest motiv TEC cu grilă izolată se numește și TEC-MIS (Metal-Insulator-Semiconductor). În tehnologiile moderne acest izolator este bioxidul de Si fapt care a condus la denumirea consacrată de TEC-MOS (Metal-Oxide-Semiconductor) pentru TEC cu grilă izolată. Tranzistoarele MOS sînt la rîndul lor de două tipuri: *cu canal indus* și *cu canal inițial*. Din motive tehnologice în prezent se fabrică mai mult TEC-J cu canal *n* și TEC-MOS cu canal *p* indus, cu tendința de revenire la TEC-MOS cu canal *n* în cazul integrării pe scară largă (LSI).

Comparație între TEC și tranzistorul bipolar

Tranzistoarele cu efect de cîmp prezintă o serie de proprietăți care le conferă avantaje în raport cu tranzistoarele bipolare. Comparația se va face privitor la montajul cu sursă comună, montaj similar celui cu emitor comun la tranzistoarele bipolare:

1) Caracteristica statică $I_D = f(U_{D.})$ trece prin originea axelor deoarece la TEC nu există joncțiuni în calea curentului de drenă. Din acest motiv TEC este foarte bun pentru utilizare ca element cu acțiune discretă, semnalul 0 logic fiind chiar 0 volți.

2) Impedanța de intrare a unui TEC este mult mai mare decît a tranzistoarelor bipolare: 10^8 — 10^9 ohmi pentru TEC-J și 10^9 — 10^{14} ohmi pentru TEC cu grilă izolată.

3) Cel mai mare avantaj pe care îl prezintă TEC față de tranzistorul bipolar este integrabilitatea sa superioară. În prezent cel mai bine se pretează integrării TEC-MOS care prezintă următoarele avantaje în raport cu CI cu tranzistoare bipolare :

a) Circuitele integrate cu TEC-MOS sînt mai simple. Simplitatea se datorește faptului că nu conține rezistoare și condensatoare convenționale, ci numai tranzistoare MOS. Acest fapt conduce la creșterea densității funcțiilor electronice pe unitatea de suprafață deoarece : 1) CI cu TEC-MOS exclud necesitatea izolării tranzistoarelor realizate pe același cip (excepție fac numai TEC complementare) ; în cazul tranzistoarelor bipolare realizate integrat circa 30% din suprafața utilă a cip-ului este ocupată cu regiuni de izolare a componentelor schemei. 2) În cazul CI cu tranzistoare bipolare rezistoarele ocupă un spațiu foarte mare, cu un ordin de mărime mai mare decît la CI cu TEC-MOS.

b) Tehnologia de realizare a CI cu TEC-MOS este mai simplă decît pentru CI cu tranzistoare bipolare (o etapă de difuzie și 4 de fotomascare, față de 4 etape de difuzie și minimum 6 de fotomascare).

c) Existența unei tensiuni de prag (pinch-off) ridicate $U_p = 0,6 - 6$ V (în funcție de tehnologie) conduce la o mai mare imunitate la perturbații a CI cu TEC-MOS.

d) CI cu TEC-MOS folosesc tensiuni de alimentare și nivele logice mai ridicate decît CI cu tranzistoare bipolare. În același timp consumul lor de energie este mai redus și au dimensiuni mai mici.

e) Se pot realiza cu mare ușurință circuite complexe, cu creșterea randamentului de fabricație și siguranței în funcționare.

Regimul de comutare al TEC

Folosirea TEC la realizarea circuitelor logice implică funcționarea lor în regim de comutație cu trecere rapidă din starea de conducție în cea de blocare și invers. Ca element de comutație TEC prezintă avantaje în raport cu dispozitivele semiconductoare clasice :

— în stare de conducție nu introduce tensiuni reziduale în calca semnalului ; circuitul drenă-sursă este echivalent cu un simplu rezistor ;

— deci rezistențele minime care se realizează în stare de conducție la TEC sînt mai mari decît la tranzistoarele bipolare ($r_{Dson} = 2 \dots 1\ 000$ ohm față de $r_{CEon} = 1 \dots 30$ ohm), în stare blo-

cată rezistența este cu 2—3 ordine de mărime superioară tranzistoarelor bipolare ($r_{D_{off}}=10^8-10^{12}$ ohm). Astfel, raportul între rezistența în stare blocată și conductoare, care caracterizează calitatea unui dispozitiv semiconductor ca element de comutare, este pentru TEC $r_{off}/r_{on} \cong 10^7-10^{10}$, apropiindu-se de funcționarea unui comutator ideal.

Comanda TEC se face cu nivele de tensiune aplicate între grilă și sursă în cazul montajelor cu sursă comună. Pentru TEC-J aducerea în conducție se face simplu prin reducerea la zero a tensiunii grilă-sursă ($U_{GS}=0$). Blocarea unui TEC-J se face aplicând între grilă și sursă o tensiune de polarizare inversă mai mare decât valoarea de prag $-U_p$. La TEC-MOS posibilitățile de comandă sînt mai variate, depinzînd de tipul constructiv. Cel mai avantajos se comandă TEC-MOS cu canal indus care se blochează cînd $U_{GS}=0$ și se deschide pentru semnale mult mai mari decât valoarea de prag ($U_{GS} \gg U_p$).

Datorită integrabilității foarte bune în prezent circuitele logice cu TEC se realizează cu predilecție integrate. Am amintit că din motive tehnologice se preferă TEC-MOS cu canal p indus. În continuare se vor prezenta principalele posibilități de folosire a TEC la materializarea circuitelor logice folosind TEC-MOS în varianta integrată.

TEC-MOS utilizat ca rezistor de sarcină

Datorită faptului că pentru CI la realizarea unei rezistențe este necesar un număr mare de operații și că ocupă o arie apreciabilă, în CI cu TEC-MOS rezistoarele sînt înlocuite cu structuri TEC-MOS. Acestea permit obținerea unor valori mari de rezistență pe o suprafață mică și sînt mai simplu de realizat. Rezistorul de tip TEC-MOS are o structură similară cu a tranzistorului MOS. Drept rezistor se utilizează rezistența canalului TEC-MOS în conducție care funcționează fie în regiunea saturată, fie în cea nesaturată a caracteristicii statice. În afară de realizarea ușoară sub formă integrată, folosirea TEC-MOS ca rezistor prezintă avantajul controlului valorii rezistenței prin tensiunea de grilă.

Constructiv, rezistorul de tip TEC-MOS se realizează fie prin conectarea grilei cu dreana, fie prin conectarea grilei la o sursă separată (Fig. 5.35 a). Sursa TEC de sarcină este conectată cu TEC-ul care realizează prelucrarea logică. Cînd grila se conectează la drenă, deoarece $U_{GS}=U_{DS}=E_D$ TEC-ul de sarcină

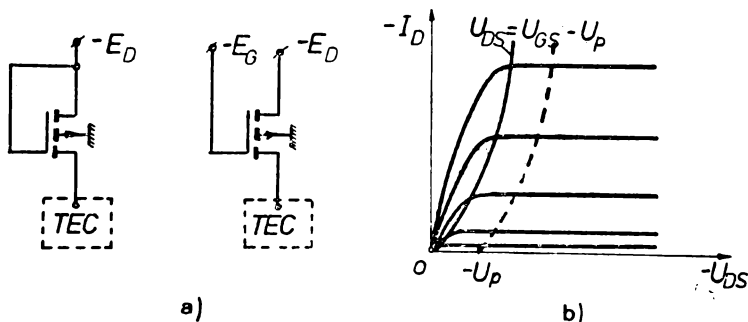


Fig. 5.35

va funcționa saturat întrucît $|U_{D_s}| > |U_{G_s} - U_p|$ (fig. 5.35 b). În acest caz tensiunea obținută pe drenea TEC de prelucrare logică la blocarea acestuia va fi mai mică decît E_D cu valoarea U_p . Dacă grila se conectează la sursa separată de valoare E_G mai mare decît E_D cu cel puțin U_p , în așa fel ca să fie satisfăcută relația $|U_{D_s}| \leq |U_{G_s} - U_p|$, TEC-ul de sarcină va funcționa pe porțiunea nesaturată a caracteristicii. În acest caz semnalul la ieșire va fi apropiat de E_D . Dezavantajul necesității unei surse suplimentare este compensat de faptul că aceasta trebuie să asigure practic numai polarizarea, consumul fiind neglijabil.

5.4.1. Circuite logice statice cu TEC-MOS

Circuitele logice care sînt conectate permanent la sursele de alimentare au primit denumirea de *circuite logice statice*. Elementul de bază al acestor circuite îl constituie circuitul de negare cu ajutorul căruia se realizează funcții logice complexe.

Circuitul logic NU static

În fig. 5.36 sînt prezentate schemele circuitului de negare în versiunea integrată pentru cele două moduri de polarizare ale grilei tranzistorului de sarcină. Deoarece TEC-MOS sînt cu canal p se lucrează în logică negativă. Se consideră circuitul din fig. 5.36 a și la intrare se aplică semnal 0 (potențialul masei). În acest caz T_1 se blochează și singurul curent care străbate circuitul este curentul rezidual al celor două dispozitive integrate, drenea lui T_1 și sursa lui T_2 . Acest curent îndreptat de la substrat spre $-E_D$ este suficient să producă pe TEC de sarcină o cădere de tensiune egală cu valoarea de prag. Rezultă că la ieșirea cir-

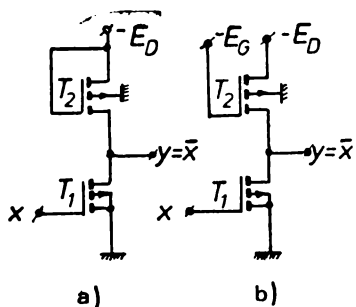


Fig. 5.36

cuitului semnalul va fi mai mic decât $-E_D$ cu valoarea U_p a lui T_2 . Deoarece pentru comanda etajului următor semnalul la ieșire trebuie să depășească valoarea U_p iar semnalul de comandă este $E_D - U_p$, semnalul necesar pentru comanda etajului următor trebuie să fie minimum $E_D - 2U_p$. Dacă $E = 2U_p$ curentul prin TEC de sarcină tinde spre zero și timpul de comutare al inversorului crește la infinit. Din acest motiv pentru aceste conectări

ale TEC de sarcină este necesar ca $E_D \geq 3U_p$. În cazul circuitului din fig. 5.36 b, la ieșire se obține semnal 1 egal cu $-E_D$.

Dacă la intrare se aplică semnal 1 (nivel de tensiune negativ în valoarea absolută mai mare decât U_p), T_1 se saturează și la ieșire se obține semnal logic 0.

Circuite logice statice NICI și NUMAI

În fig. 3.57a este prezentat un circuit NICI cu două intrări. Tranzistoarele T_1 și T_2 realizează prelucrarea logică iar T_3 este tranzistorul de sarcină. Când la intrări semnalul este 0 ambele tranzistoare de prelucrare sînt blocate și la ieșire se obține semnal logic 1 (apropiat de $-E_D$). Dacă una sau ambele intrări sînt acționate cu semnal 1 ($-E_D$), unul sau ambele tranzistoare de prelucrare logică sînt în conducție și la ieșire se obține semnal 0. Schema unui circuit NUMAI cu două intrări este prezentată

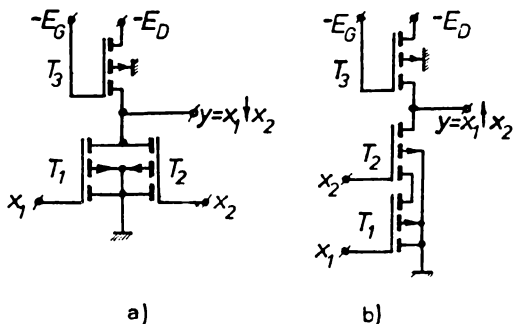


Fig. 5.37

în fig. 5.37b. La ieșire se va obține semnal 0 numai atunci când la ambele intrări se aplică semnal logic 1, deci când T_1 și T_2 vor fi în conducție. În orice altă situație unul sau ambele tranzistoare de prelucrare vor fi blocate și la ieșire se va obține semnal logic 1.

Se observă că realizarea circuitelor logice cu TEC-MOS se face ca și în cazul circuitelor DCTL, după o logică de conexiuni. Deosebirea în realizarea integrată constă în utilizarea unui tranzistor în locul rezistorului de sarcină. La realizarea circuitelor logice mai complexe se preferă circuitul NICI care este mai economic de realizat integrat decât circuitul NUMAI. Acest lucru se datorește faptului că la circuitul NUMAI, care are două TEC în serie (sau mai multe), este necesar ca dimensiunile canalului să fie aproximativ duble pentru a se obține semnal 0 de aceeași valoare ca și pentru circuitul NICI. Rezultă pentru circuitul NUMAI o suprafață mai mare cu o capacitate mai mare și deci o viteză de comutare mai redusă.

5.4.2. Circuite logice dinamice cu TEC-MOS

Spre deosebire de circuitele logice statice, *circuitele logice dinamice* sînt alimentate numai în timpul transmiterii informației. Rezultă astfel circuite logice cu un consum mult mai mic decât al circuitelor clasice. Circuitele dinamice sînt ușor de realizat integrat datorită posibilității de stocare a informației de către un TEC-MOS în condensatorul format de sistemul grilă-sursă. Principial, un element de stocare cu TEC-MOS utilizat la realizarea circuitelor logice dinamice este reprezentat în fig. 5.38a. Semnalul se aplică pe grila tranzistorului T prin intermediul întrerupătorului I , încărcînd condensatorul C format din grila izolată și masă. Când I se deschide C rămîne încărcat polarizînd tranzistorul. Deoarece în circuitul de grilă se pierde numai o cantitate extrem de mică de sarcină, condensatorul păstrează

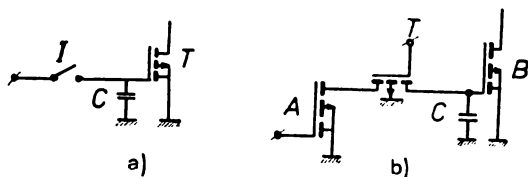


Fig. 5.38

informația pînă la aplicarea unui nou semnal prin intermediul întrerupătorului. Întrerupătorul este realizat tot cu un TEC-MOS și se numește *poartă de transmisie*. În fig. 5.38 b este reprezentată poarta de transmisie cu TEC care servește la izolarea, controlată prin impulsuri de tact, a circuitelor *A* și *B*. Semnalul de la circuitul *A* se transmite circuitului *B* numai în cazul cînd poarta de transmisie este deschisă, adică atunci cînd semnalul de tact *T* se află în starea logică 1.

Circuit logic NU^* dinamic

În fig. 5.39a este prezentată schema circuitului inversor logic dinamic, iar în fig. 5.39b simbolizările lui. Dacă semnalul de tact este 0, tranzistoarele T_2 (de sarcină), și T_3 (de transmisie) sînt blocate. Dacă semnalul de tact este 1, atît T_2 cît și T_3 sînt în conducție și la ieșire apare semnal în funcție de semnalul aplicat la intrare. Astfel, dacă T_1 este blocat, potențialul drenei sale

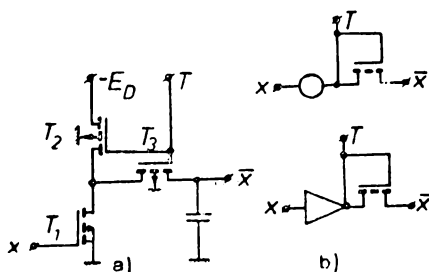


Fig. 5.39

este $-E_D$ care se transmite prin T_3 încărcînd condensatorul etajului următor. Dacă T_1 conduce, prin aplicarea semnalului 1 la intrare, potențialul drenei sale este apropiat de al masei și dacă anterior condensatorul circuitului următor a fost încărcat cu semnal 1 acum este adus la 0. Datorită conectării tranzistorului de sarcină numai în timpul semnalului de tact consumul de energie este foarte redus.

Circuite logice dinamice NICI și NUMAI

În fig. 5.40 a și b sînt reprezentate circuite logice dinamice NICI și NUMAI pentru două variabile cu TEC-MOS cu canal p indus. De asemenea, sînt indicate și simbolurile acestor cir-

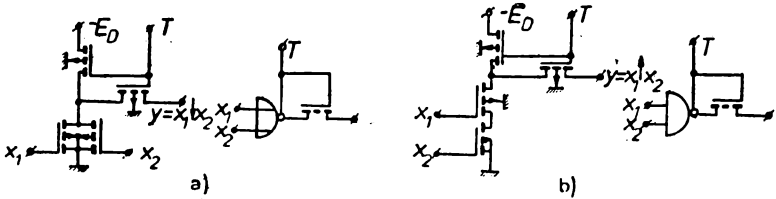


Fig. 5.40

cuite care evidențiază și poarta de transmisie. Funcționarea este similară circuitelor NICI și NUMAI statice cu excepția că informația este transmisă numai sincron cu tactul.

5.4.3. Circuite logice cu TEC-MOS complementare

Spre deosebire de circuitele logice cu TEC-MOS prezentate pînă acum care utilizau tranzistoare de aceeași polaritate, circuitele cu TEC-MOS complementare (TEC-COSMOS, de la Complementary-Simmetry-MOS sau TEC-CMOS) conțin dispozitive cu canale de ambele polarități pe același substrat. În comparație cu TEC-MOS, TEC-CMOS prezintă avantajul unui consum mult mai redus de energie deoarece indiferent de starea în care se află, un tranzistor este în conducție iar complementarul este blocat. Astfel, circuitul format dintr-un cuplu de tranzistoare complementare este parcurs numai de curentul rezidual al tranzistorului blocat. Această particularitate face ca aceste circuite să fie foarte convenabile în aplicații cu consum redus: aparatură portabilă, instalații aerospațiale etc. Principalul dezavantaj al acestor circuite constă în faptul că ocupă suprafață mare din cauza necesității izolării între ele a celor două structuri. Din acest motiv densitatea pe unitatea de suprafață este redusă iar costul mai ridicat decît la circuitele cu tranzistoare cu un singur tip de canal.

Circuit logic de negare cu TEC-CMOS

În fig. 5.41a este prezentată schema unui circuit inversor logic cu tranzistoare MOS complementare. Sursa și substratul tranzistorului cu canal n sînt conectate la masă, iar ale tranzistorului cu canal p sînt conectate la plusul sursei de alimentare $+E_s$. Intrarea se face pe ambele grile ale tranzistoarelor. Circuitele logice cu TEC-CMOS lucrează în logică pozitivă. Dacă la intrare se aplică semnal 0 (masa), tensiunea grilă-sursă a TEC cu canal p este negativă și îi asigură conducția. Tranzistorul T_1

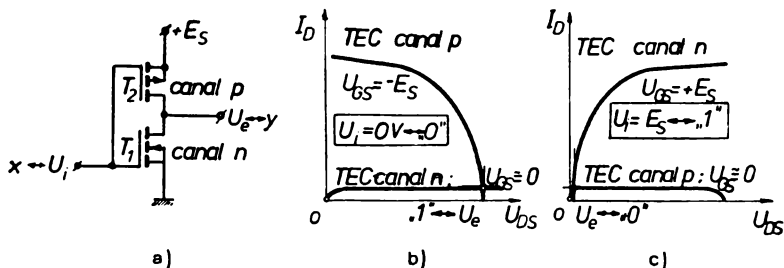


Fig. 5.41

cu canal n este blocat deoarece $U_{G,S}=0$. Drept urmare potențialul drenei tranzistorului cu canal n este egal cu al sursei de alimentare și la ieșire se obține semnal 1. Starea circuitului este definită de intersecția caracteristicilor statice a celor două structuri și pentru semnal 0 la intrare sînt prezentate în fig. 5.41b. Dacă la intrare se aplică semnal 1 ($+E_s$) TEC cu canal p se blochează deoarece $U_{G,S}=0$ iar TEC cu canal n va fi în conducție ($U_{G,S}=+E_s$) și la ieșire se obține semnal logic 0. În fig. 5.41c sînt prezentate caracteristicile celor două tranzistoare pentru această stare. Din funcționarea prezentată rezultă că fiecare din cele două tranzistoare constituie sarcină pentru cel complementar. Dacă se ține cont de consumul necesar încărcării și descărcării capacităților parazite în timpul comutării, consumul de putere al unui astfel de circuit este :

$$P_a = P_{a0} + E_s^2 f C,$$

unde f este frecvența de comutare, iar P_{a0} — puterea consumată în regim staționar (pentru $E_s=10V$ și $I_D \approx 1$ nA, rezultă $P_{a0} \approx 10$ nW).

Circuite NICI și NUMAI cu TEC-CMOS

În fig. 5.42 sînt prezentate circuite logice NICI și NUMAI cu două întrări cu tranzistoare TEC-CMOS. Pentru ca circuitul NICI (fig. 5.42a) să prezinte semnal 1 la ieșire este necesar ca la ambele intrări să acționeze semnal 0, deci T_3 și T_4 să conducă iar T_1 și T_2 să fie blocate. Este suficient ca la o intrare să se aplice semnal 1 astfel ca prin deschiderea tranzistorului corespunzător cu canal n la ieșire să se obțină semnal 0. În cazul circuitului NUMAI (fig. 5.42b) este necesar ca la ambele intrări

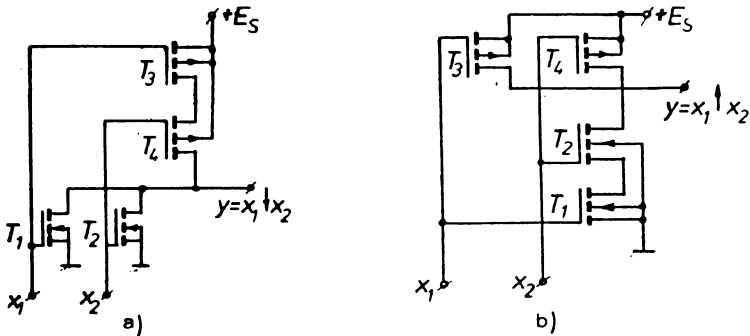


Fig. 5.42

să acționeze semnal 1 pentru ca la ieșire să se obțină semnal 0 ; în toate celelalte situații la ieșire se obține semnal 1. Se poate observa că la aceste circuite tranzistoarele cu canal p se conectează dual față de tranzistoarele cu canal n .

5.5. Circuite logice cu tranzistoare cu performanțe deosebite

După apariția CI dezvoltarea acestora a marcat două tendințe : de creștere a vitezei de comutare și de creștere a densității componentelor convenționale pe un cip.

I) Necesitatea măririi vitezei de funcționare a circuitelor a fost impusă de creșterea complexității tehnicilor numerice care reclamă un timp de răspuns din ce în ce mai redus. În acest sens, pornind de la circuitul „totem pole” (fig. 5.34) s-au elaborat CI cu tranzistoare bipolare cu viteză mare de comutare — seria *TTL rapidă*. În fig. 5.43 este prezentată schema circuitului SI-

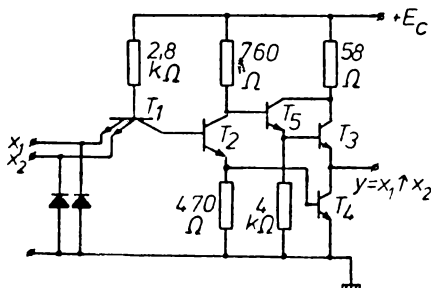


Fig. 5.43

NU cu două intrări din seria rapidă. Se observă că tranzistorul T_3 de la seria normală (v. fig. 5.34) s-a înlocuit cu perechea de tranzistoare T_3-T_4 în montaj Darlington. Această configurație asigură o viteză mai mare de comutație decât a circuitului standard (6 ns față de 11 ns la trecerea din 0 în 1 și 6 ns față de 7 ns la comutarea din 1 în 0). Tranzistorul T_4 are și rolul diodei de decalare din circuitul standard, asigurând blocarea fermă a tranzistorului T_3 când T_4 este în conducție. Circuitele din această serie prezintă dezavantajul unui consum de putere mai mare decât circuitul de viteză normală. La I.P.R.S.—Băneasa se produc circuite logice din seria rapidă / 52/ și care se disting de cele normale prin introducerea în codul de marcare a literei H (prescurtare de la „High speed“). Datorită faptului că seria TTL H funcționează cu tranzistoare saturate nu se pot obține timpi de comutare sub 5 ns. Pentru aplicațiile care impun timpi de comutare sub limita menționată s-au realizat *circuite logice cu cuplaj în emitor* (ECL-Emitter Coupled Logic). Circuitele ECL funcționează cu tranzistoare nesaturate fapt ce conduce la obținerea unor timpi de comutare de 1—4 ns. În fig. 5.44 este pre-

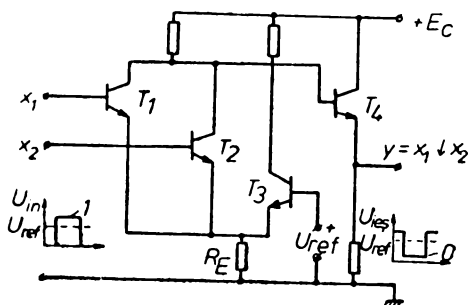


Fig. 5.44

zentată schema unui circuit ECL-NICI cu două intrări, în care tranzistoarele T_1 și T_2 sînt de prelucrare logică iar T_3 de referință. Baza lui T_3 este conectată la o sursă de referință U_{ref} asigurînd conducția acestuia. Valoarea sursei de referință se află sub nivelul logic 1. Rezistorul R_E de cuplaj are o valoare suficient de mare pentru a se realiza o sursă de curent constant. Dacă ambele intrări sînt acționate cu semnal logic 0 (la masă), atît T_1 cît și T_2 sînt blocate. Ca urmare, în baza tranzistorului T_4

se injectează curent suficient pentru a-l aduce în conducție și la ieșire se obține semnal logic 1 (apropiat de $+E_C$). Dacă la una din intrări se aplică nivel logic 1 (superior lui U_{ref}) tranzistorul respectiv intră în conducție și comută o parte din curentul care parcurge circuitul $R_E - T_3$. În consecință potențialul de colector al tranzistorului de prelucrare logică scade și totodată începe blocarea tranzistorului T_4 . Pragul de comutare este egal cu tensiunea de referință, după care la ieșire se obține semnal logic 0. Cuplajul prin emitor nu permite saturarea tranzistoarelor iar comutarea curentului asigură o excursie coborîită (cca. 0,8 V) la trecerea de la un nivel logic la celălalt ; ambele situații conduc la un timp de comutare foarte redus. Folosirea repetorului pe emitor la ieșire asigură un „fan out” în jur de 25. Dezavantajul circuitelor ECL constă în consumul de putere mult mai mare decît al celorlalte circuite logice (circa 50 mW/circuit). Prin folosirea diodelor Schottky conectate între baza și colectorul unui tranzistor (fig. 5.45 a) s-a obținut un dispozitiv care permite îmbinarea vitezei de comutație a circuitelor ECL cu consumul relativ redus al circuitelor TTL. Folosirea conexiunii menționate nu permite saturarea tranzistorului, timpul de comutare fiind extrem de redus datorită timpului de stocare practic nul (cca. 1 ns) al diodei Schottky. În fig. 5.45 b este prezentat un circuit SI-NU

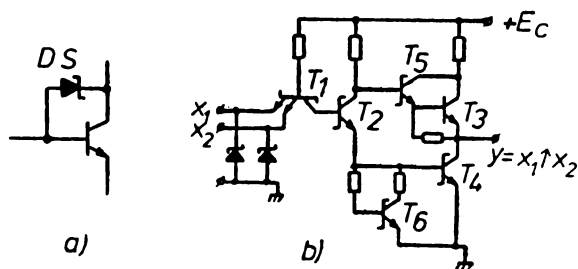


Fig. 5.45

cu două intrări cu „tranzistoare Schottky”. Denumirea și simbolizarea specifică definesc folosirea configurației diodă Schottky-tranzistor bipolar. Se observă că structura circuitului este asemănătoare circuitului TTL H, cu deosebirea că tranzistoarele T_1 , T_2 , T_4 și T_5 sînt tranzistoare Schottky. În plus, se folosește tranzistorul T_6 care accelerează comutarea tranzistorului T_4 .

Tranzistorul T_3 , nu este necesar să fie de tip Schottky deoarece conexiunea Darlington cu T_5 , nu-i permite saturarea. Circuitul prezentat realizează timpi de comutare de aproximativ 3 ns și puteri disipate de cca. 20 mW. Circuitele cu tranzistoare MOS asigură o complexitate mult mai mare decât circuitele cu tranzistoarele bipolare dar sînt mult mai lente (timpi de comutare în medie mai mari de 50 ns).

II) Din constatarea că prin integrare costul pe circuit scade substanțial cu creșterea complexității cip-ului a apărut imediat tendința dezvoltării corespunzătoare a tehnologiilor de integrare. Dacă la început se obțineau pe un singur cip 1—12 circuite logice similare, etapă cunoscută ca integrare pe scară mică (SSI), în prezent se construiesc curent pînă la 100 de circuite pe o așchie de siliciu (integrare pe scară medie — MSI). De asemenea, în prezent asistăm la o depășire a complexității de 100 circuite/cip (integrare pe scară largă — LSI), prin rafinarea tehnologiilor de integrare atingîndu-se o densitate de 10 000—15 000 de tranzistoare pe o pastilă de siliciu. Această creștere spectaculoasă a complexității de integrare se datorește integrabilității foarte bune a tranzistoarelor MOS. În prezent s-a trecut la circuite VLSI (Very Large Scale Integration) cu o densitate de pînă la 50 000 de tranzistoare/cip și apoi urmînd circuite SLSI (Super Large Scale Integration) cu densități ce depășesc valoarea menționată pentru tipul VLSI (cca. 10^6 tranzistori/cip). În ultimii ani, din „con competiția” dintre tranzistorul bipolar care asigură viteze mari dar un grad scăzut de complexitate și tranzistorul MOS care asigură o complexitate deosebit de mare dar sînt relativ lente, s-a obținut o nouă familie de CI avînd la bază *circuitul logic cu injecție* — I^2L (Integrated Injection Logic). Aceste circuite folosesc tranzistoare bipolare cu introducerea unor perfecționări care să asigure creșterea gradului de complexitate la integrare. Pentru aceasta s-au înlocuit rezistoarele cu tranzistoare de sarcină (ca la TEC-MOS) și s-a căutat să se elimine necesitatea izolării componentelor realizate pe același cip. Punctul de plecare a fost circuitul DCTL (v. fig. 5.32). În cazul circuitelor I^2L toate joncțiunile bază-emitor în paralel ale tranzistoarelor din circuitele comandate (tranzistoarele T_2 din circuitele DCTL — fig. 5.32) sînt înlocuite cu un *tranzistor multicolector*, iar rezistorul de sarcină cu un generator de curent (fig. 5.46 a). Ca generator de curent poate fi folosită orice sursă capabilă să injecteze purtători minoritari în baza tranzistorului multicolector. Soluția cea

mai avantajoasă a fost folosirea unui tranzistor pnp ca generator de curent (fig. 5.46 b). Referitor la structura prezentată se poate observa că :

- baza tranzistorului pnp este identică cu emitorul tranzistorului npn ;
- colectorul tranzistorului pnp este identic cu baza tranzistorului npn ;
- componentele nu trebuie izolate între ele ;
- tranzistorul multicolector se poate obține ca un tranzistor multiemitor cu dopare inversă.

Datorită celor evidențiate mai sus, aria ocupată de structura din fig. 5.46 b este egală cu aria ocupată de un tranzistor, fără nici o izolare între componente. Acest fapt conduce la obținerea unor grade de complexitate foarte ridicate ; se estimează realizarea unei densități de pînă la 3 000 circuite pe un cip. Alimentația unei structuri I^2L se face cu tensiuni mai mari de 0,85 V, impusă de deschiderea joncțiunii emitor-bază a tranzistorului pnp . Emitorul tranzistorului pnp poate fi comun pentru toate circuitele de pe un cip și a primit denumirea de *injector*.

O proprietate deosebit de avantajoasă a structurii I^2L constă în faptul că prin modificarea curentului injectorului se poate modifica după dorință puterea disipată și ca urmare și timpul de comutare. Atît puterea cît și timpul de propagare al unui circuit I^2L se pot regla pe 3... 4 ordine de mărime : puterea între 1 nW și 10 μ W/circuit, iar timpul de comutare între zeci

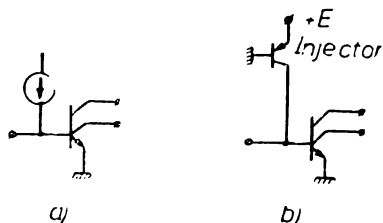


Fig. 5.46

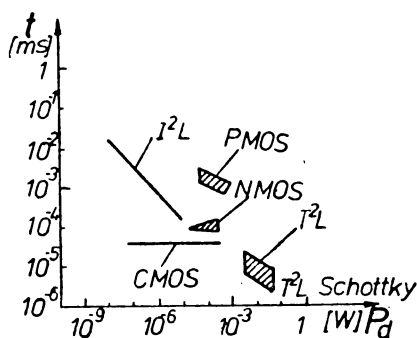


Fig. 5.47

de nanosecunde și zeci de milisecunde. În fig. 5.47 este prezentată o diagramă comparativă în ceea ce privește raportul viteză-putere consumată pentru diferite tipuri de circuite. Se observă că tehnologia I^2L este competitivă cu tehnologiile MOS, păstrând un ușor avantaj de viteză (s-au obținut deja circuite cu timpi de propagare între 1 și 10 ns). Folosind structura din fig. 5.46 b se pot materializa simplu funcții logice complexe. Astfel, în fig. 5.48 a este prezentat un circuit SI-NU cu două intrări. Dacă la una din intrări se aplică semnal logic 0 (U_{CEs} , a tran-

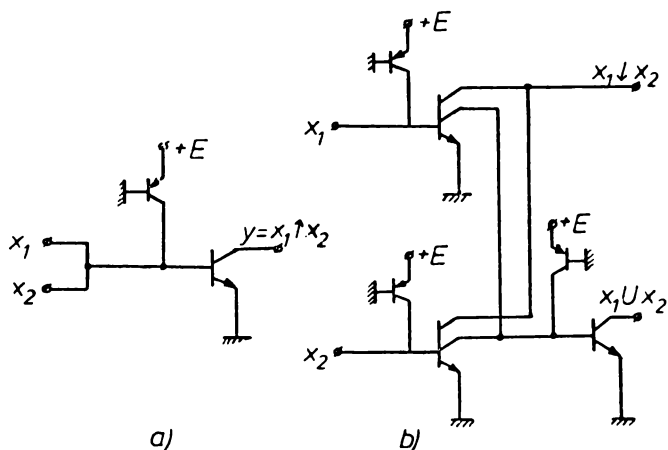


Fig. 5.48

zistorului de comandă) curentul de injector este comutat prin intrarea considerată la masă. Ca urmare, tranzistorul $n\phi n$ este blocat și potențialul colectorului conectat la circuitul de sarcină crește-semnal logic 1. La ieșire se obține potențial coborât (U_{CEs}) dacă la ambele intrări se aplică potențiale ridicate (U_{BEs}), curentul injectorului asigurând saturarea tranzistorului $n\phi n$. Datorită tranzistorului multicolector cu ieșiri izolate galvanic, circuitele I^2L permit materializarea funcțiilor complexe prin conectarea în paralel a ieșirilor (logică cablată). În fig. 5.48 b se prezintă un circuit cu două intrări și două ieșiri la care se obțin funcțiile NICI și respectiv NICI—NEGAT=SAU.

5.6. Circuite logice cu dispozitive semiconductoare cu caracteristică non-monotonă

Toate dispozitivele semiconductoare utilizate pînă acum la materializarea *FB* prezintă o *caracteristică statică monotonă*. Există dispozitive semiconductoare în a căror caracteristică statică apar porțiuni de instabilitate, cu rezistență negativă, denumite dispozitive cu *caracteristică non-monotonă*. Avantajul acestora constă în comutarea extrem de rapidă între porțiunile stabile cu conductanță pozitivă. Datorită existenței a două stări stabile de funcționare aceste dispozitive sînt apte să materializeze funcții și variabile bivalente. Cele mai utilizate dispozitive cu caracteristică non-monotonă în realizarea circuitelor logice sînt : dioda tunel (*DT*), tiristorul și dioda cu două baze sau tranzistorul unijoncțiune (*TUJ*).

5.6.1. Circuite logice cu diode tunel

Dioda tunel sau Esaki prezintă o caracteristică volt-amperică de tip *N*, cu o porțiune de conductanță negativă (reprezentată cu linie întreruptă în fig. 5.49). Aceste dispozitive prezintă unele proprietăți deosebite, printre care : timp de comutare extrem de redus-tipic 1 ns, funcționare sigură într-un domeniu larg de temperatură și nu sînt influențate de doze importante de radiație. Principalii parametri ai unei diode tunel sînt : curentul de vîrf I_p (denumit și curent de prag), curentul de vale I_v și tensiunile corespunzătoare acestor curenți U_p , U_v și U_{fp} (tensiune de vîrf avansat). În domeniul circuitelor logice se folosește comutarea între regiunile stabile de funcționare : *regiunea tunel* cuprinsă între 0 și U_p căreia i se asociază cifra binară 0 și *regiunea de difuzie* cuprinsă între U_v și U_{fp} căreia i se asociază cifra binară 1.

În circuitele logice *DT* asigură funcționarea discretă de mare viteză, iar alte elemente interpușe între diodă și generatoarele de comandă realizează prelucrarea logică. Cele mai utilizate circuite atunci cînd nu se urmărește o viteză foarte mare de comutație sînt cele cu rezistoare și *DT*. În fig. 5.50 a este prezentat

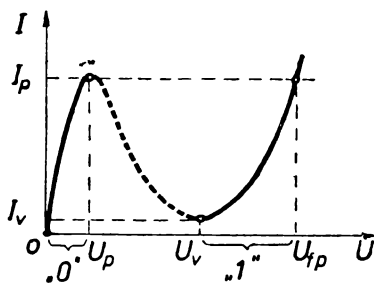


Fig. 5.49

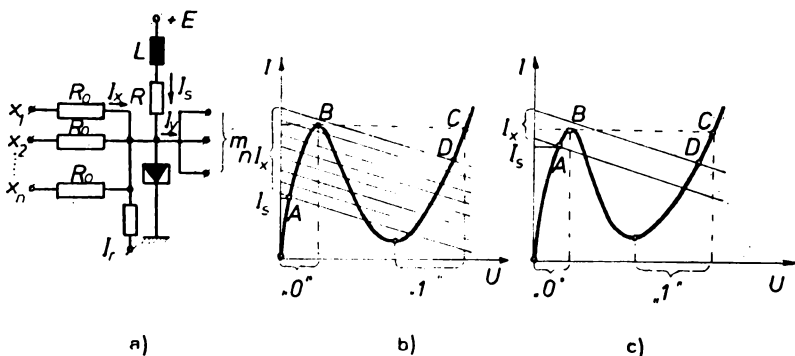


Fig. 5.50

un circuit tipic cu rezistoare și DT , cu n intrări și m ieșiri. Pentru revenirea în stare inițială de pe ramura tunel este prevăzută o intrare pentru aplicarea unui impuls negativ de rebasculare, I_r . Prin alegerea corespunzătoare a valorilor sursei și rezistorului R se poate alege poziția punctului inițial A în regim staționar. Dacă acest punct se alege astfel încît să fie îndeplinită condiția: $nI_r + I_s \geq I_p + I_v$, circuitul poate materializa funcția SI pentru cele n semnale de intrare (fig. 5.50 b). În adevăr, curenții pe cele n intrări determinați de semnalele de comandă asigură depășirea punctului de vîrf ce conduce la comutarea DT la nivel de tensiune ridicat. În cazul cînd se folosește o inductanță L dimensionată corespunzător ($\tau = L/R \gg \tau_D = U_p C_D / I_p$ — în care C_D este capacitatea DT) comutarea se realizează practic la curent constant cu deplasarea punctului de funcționare din A în B și apoi din C în D . Dacă punctul inițial de funcționare de pe ramura tunel se alege astfel încît la aplicarea semnalului 1 la orice intrare să aducă DT pe ramura de difuzie (fig. 5.50c), circuitul realizează funcția SAU pentru n argumente.

Se remarcă deosebita simplitate a circuitului și posibilitatea modificării facile a acestuia pentru a realiza funcții SI sau funcții SAU. Principalele dezavantaje ale circuitelor cu DT constau în: dependența severă de toleranțele pieselor, lipsa amplificării, semnale reduse ca valoare. Asocierea tranzistoarelor cu DT permite eliminarea neajunsurilor menționate și realizarea unor circuite logice cu performanțe deosebite. Este evident că aceste circuite hibrid vor avea o viteză de funcționare mai redusă

decît a schemelor numai cu DT , în schimb tranzistorul „este forțat” să funcționeze cu viteză mai mare. Se pot atinge în mod curent cu astfel de circuite viteze de comutație în jur de 50 MHz.

În afara circuitelor menționate mai sus, cu ajutorul DT și tranzistoarelor se pot materializa funcții mult mai complexe, cum sînt funcții cu prag.

Circuite logice cu prag (CLP)

În paragraful 2.5.3. s-au definit funcțiile booleene cu prag. În continuare se va prezenta modul de materializare al acestor funcții prin circuite logice cu prag. Prima dată aceste circuite au fost evidențiate în 1960 și datorită avantajelor pe care le prezintă față de circuitele clasice se impun atenției. Un circuit cu prag cu n intrări are structura din fig. 5.51a și conține un sumator linear (Σ) cu n intrări ponderate (w_1, w_2, \dots, w_n — ponderile asociate intrărilor) și un element nelinier EN cu caracteristica de transfer din fig. 5.51b. Elementului nelinier i se aplică excitația E de la ieșirea sumatorului: $E = \sum_{i=1}^n w_i x_i$. Parametrii unui CLP sînt valorile ponderilor și a pragului, iar simbolul acestuia

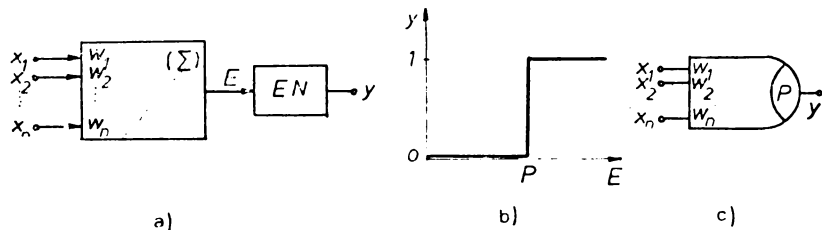


Fig. 5.51

este în fig. 5.51c. Conform modelului matematic al unei funcții cu prag, pentru un CLP se poate scrie :

$$y = (w_1 x_1 + w_2 x_2 + \dots + w_n x_n)_P .$$

Circuitele cu prag pot fi materializate foarte simplu atît cu tranzistoare bipolare cît și cu diode tunel. În fig. 5.52a este prezentată schema unui CLP cu tranzistor în care rezistoarele R_1, R_2, \dots, R_n realizează sumarea analogică a semnalelor de intrare

u_1, u_2, \dots, u_n , iar tranzistorul în comutație constituie elementul neliniar. Ponderile sînt date de valoarea rezistoarelor iar pragul de sursa de polarizare — E_B și de R_B . Comutarea tranzistorului din stare blocată în stare saturată are loc cînd suma ponderată a curenților de la intrare depășește curentul prag de blocare :

$$\sum_{i=1}^n \frac{u_i}{R_i} \geq \frac{E_B}{R_B},$$

în care s-a neglijat curentul rezidual I_{CB0} . Întrucît $R_i > 0$ și u_i sînt pozitive și depind de mărimea lui R_i și u_i . În particular, cînd $u_1 = u_2 = \dots = u_n = u$ și $R_1 = R_2 = \dots = R_n = R$, circuitul realizează funcția NICI în logică pozitivă dacă $P=1$ sau funcția NUMAI dacă $P=n$.

Existența punctului de vîrf al caracteristicii statice a unei DT permite realizarea foarte ușoară a CLP . În fig. 5.52b este prezentată schema unui CLP cu DT și rezistoare. Se consideră că dreapta de sarcină intersecțiază caracteristica statică numai într-un punct (funcționare monostabilă a DT) și se află inițial pe ramura tunel (fig. 5.52c). Rezistoarele formează sumatorul

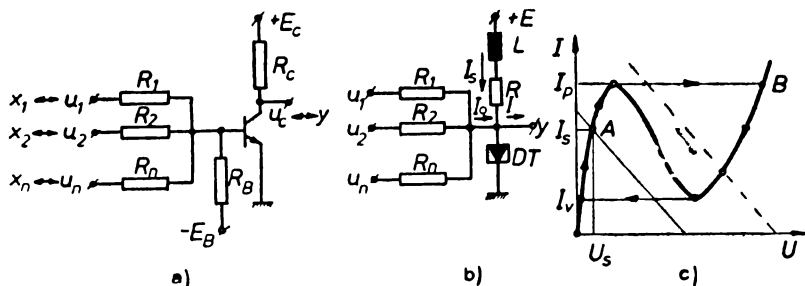


Fig. 5.52

analogic, iar DT este elementul neliniar. Pragul este determinat de curentul de vîrf al diodei tunel. Pentru acest circuit se pot defini condițiile de funcționare ca CLP :

$$y = \begin{cases} 0, & \text{dacă } I_0 + I_s < I_p + I \\ 1, & \text{dacă } I_0 + I_s \geq I_p + I. \end{cases}$$

sau

$$y = \begin{cases} 0, & \text{dacă } \sum_{i=1}^n \frac{u_i}{R_i} < I_p + I - \frac{E}{R} \\ 1, & \text{dacă } \sum_{i=1}^n \frac{u_i}{R_i} \geq I_p + I - \frac{E}{R}, \end{cases}$$

$$\text{unde } I_0 = \sum_{i=1}^n \frac{u_i}{R_i} \text{ iar } I_s = \frac{E - U_s}{R} \cong \frac{E}{R}.$$

În cazul particular cînd $R_1 = R_2 = \dots = R_n = R$ și $u_1 = u_2 = \dots = u_n = u$, circuitul din figura 5.52b materializează funcția SAU dacă $P=1$ și funcția SI de n argumente dacă $P=n$.

Din cele prezentate rezultă că circuitele logice cu prag se bucură de două proprietăți remarcabile față de circuitele logice clasice :

1) posedă o mare *flexibilitate structurală*, care se manifestă prin posibilitatea de modela cu un același CLP diferite funcții booleene prin modificarea parametrilor acestuia. Această proprietate le face apte de a fi utilizate în structuri de sisteme adaptive și instruibile ;

2) au o mare *potențialitate structurală*, care se manifestă prin posibilitatea de a modela cu un CLP, de cost și dimensiuni similare cu cel ale circuitelor logice convenționale, funcții booleene foarte complexe. Această proprietate asigură o sinteză foarte economică a circuitelor combinaționale.

Circuite logice multiprag (CLMP)

Circuitele multiprag constituie o generalizare a circuitelor cu prag și posedă o *potențialitate absolută*, adică orice FB poate fi realizată fizic printr-un astfel de circuit. $CLMP$ au fost evidențiate încă din 1962, dar prima tratare aprofundată cu elaborarea unei metode de sinteză au fost efectuate de către D.R. Haring în 1966. Proprietățile deosebite ale $CLMP$ au făcut ca acestea să fie studiate intens în ultimul timp. Și în țara noastră există preocupări în domeniul CLP și $CLMP$, prezentate în [32].

Structura unui $CLMP$, la fel ca și a unui CLP , este alcătuită dintr-un sumator linear cu intrări ponderate și dintr-un element neliniar. Dar, spre deosebire de CLP , caracteristica de transfer a elementului neliniar la un $CLMP$ conține mai multe praguri (fig. 5.53a). Reprezentarea simbolică a unui CMP este dată în

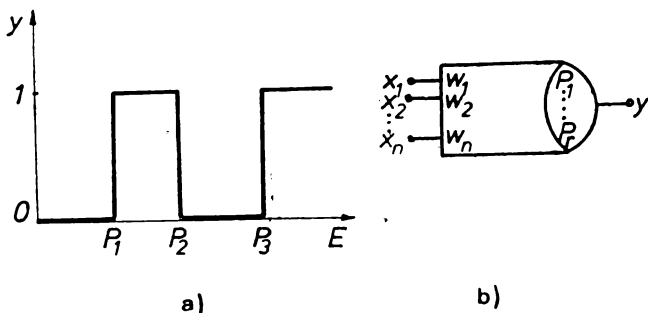


Fig. 5.53

fig. 5.53b. Un *CLMP* este complet definit prin mulțimea ponderilor $\{w_1, w_2, \dots, w_n\}$ și mulțimea pragurilor $\{P_1, P_2, \dots, P_r\}$. Modelul matematic al unui *CLMP* este:

$$y = \begin{cases} 1, & \text{dacă } E > P_1 \text{ sau dacă } P_{2j} > E > P_{2j+1} \\ 0, & \text{în caz contrar,} \end{cases}$$

unde $E = \sum_{i=1}^n w_i x_i$, $P_j \in \{P_1, P_2, \dots, P_r\}$, $P_j < P_{j+1}$, $j = 1, 2, 3, \dots$

Prin asocierea diodelor tunel cu tranzistoare bipolare se pot obține circuite logice cu mai multe praguri. În fig. 5.54a este prezentată schema unui circuit logic cu trei praguri. S-au imaginat circuite cu mai multe praguri (4,5) prin folosirea mai multor *DT*, dar consumul suplimentar de diode nu este compensat de avantaje deosebite. Pentru circuitul considerat se presupune că inițial *DT* se află pe ramura tunel cu funcționare

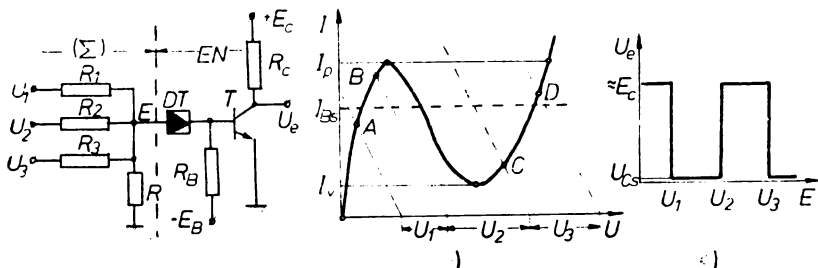


Fig. 5.54

În A , astfel ca prin diodă curentul să fie mai mic decât I_{B_s} a tranzistorului (fig. 5.54b). Nivelul prag de saturare se fixează prin sursa de polarizare inversă. Deoarece tranzistorul este blocat, la ieșire se obține semnal 1 (fig. 5.54c). Prin aplicarea semnalului U_1 se asigură depășirea curentului de saturare a tranzistorului și la ieșire se obține semnal 0, realizându-se primul prag. La alegerea corespunzătoare a semnalului U_2 și a rezistorului R_2 punctul de funcționare trece din B în C (fig. 5.54b), dioda tunel comutînd pe ramura de difuzie dar cu $I_{DT} < I_{B_s}$ și la ieșire se obține al doilea prag. Prin aplicarea și a semnalului U_3 se obține din nou saturarea tranzistorului și astfel și al treilea prag.

Pentru a ilustra potențialitatea unui $CLMP$ se consideră un astfel de circuit cu trei praguri, prezentat simbolic în fig. 5.55. Circuitul are ponderile egale $w_1 = w_2 = w_3 = 1$ iar pragurile $P_1 = 0,5$, $P_2 = 1,5$ și $P_3 = 2,5$. Funcția care se materializează cu circuitul multiprag considerat este stabilită în tabelul 5.2 și are expresia :

$$y = \bar{x}_1 \bar{x}_2 \bar{x}_3 \cup x_1 x_2 \bar{x}_3 \cup x_1 \bar{x}_2 x_3 \cup \bar{x}_1 x_2 x_3.$$

Tab. 5.2

x_1	x_2	x_3	E_{111}	y
0	0	0	0	1
0	0	1	1	0
0	1	0	1 (2)	0 (1)
0	1	1	2 (3)	1 (0)
1	0	0	1	0
1	0	1	2	1
1	1	0	2 (3)	1 (0)
1	1	1	3 (4)	0

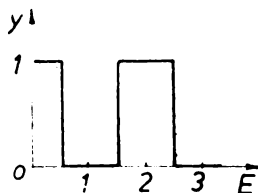
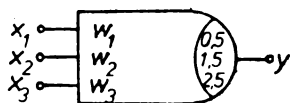


Fig. 5.55

Realizarea acestei funcții cu circuite SI—SAU—NU necesită 3 tranzistoare pentru negare și 16 diode pentru realizarea circuitelor SI și SAU în două nivele. Aceeași funcție realizată cu un $CLMP$ cu trei praguri necesită numai o DT și un tranzistor. În acest mod s-a evidențiat potențialitatea structurală a $CLMP$. Prin modificarea ponderilor sau pragurilor se pot materializa funcțiile dorite. Pentru a ilustra flexibilitatea structurală se

modifică numai ponderea $w_2 = 2$. În tab. 5.2 în paranteze sînt indicate noile valori datorate modificării lui w_2 . Se poate stabili imediat că noua funcție are expresia : $y = \bar{x}_1 \bar{x}_3 \cup x_1 \bar{x}_2 x_3$.

5.6.2. Circuite logice cu dispozitive semiconductoare cu caracteristică non-monotonă de tip S

În categoria dispozitivelor semiconductoare cu caracteristică volt-amperică în formă de S care pot fi utilizate la materializarea circuitelor logice sînt tiristoarele și TUJ-urile.

Folosirea tiristoarelor, a căror caracteristică este arătată în fig. 5.56, prezintă avantajele amplificării mari în putere (cu puteri de ordinul mW...W se pot comanda puteri de ordinul a

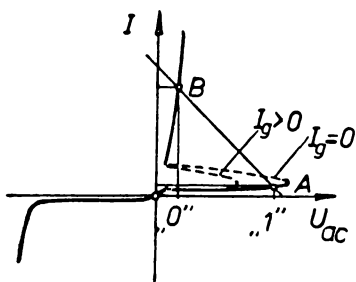


Fig. 5.56

sute de KW și chiar MW), posibilitatea funcționării atât în c.c. cât și în c.a. și o comutare rapidă cu două stări stabile (fapt ce permite comanda în impulsuri). Caracteristica volt-amperică prezintă o porțiune instabilă de conductanță negativă (indicată întrerupt) unde se face comutarea. Folosind ca intrare grila (g) și la ieșire potențialul anod-catod (se poate considera și curentul prin tiristor), în fig. 5.57 sînt prezentate trei circuite logice cu tiristoare care materializează funcțiile NU, NICI și NUMAI.

Prin notația $+E \approx$ s-a dorit menționarea posibilității de funcționare atât în c.c. cât și în c.a. în alternața pozitivă. În afara tiristoarelor convenționale, denumite și nesimetrice, se mai pot folosi *fototiristoare* la care comutarea se comandă prin fascicule luminoase (avantajul lipsei legăturii galvanice intrare-

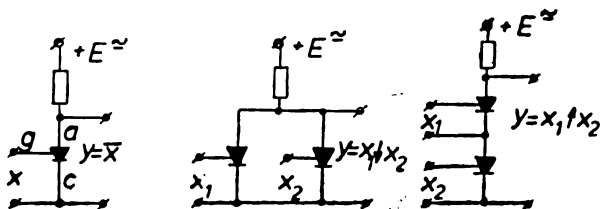


Fig. 5.57

ieșire) sau *tiristoare simetrice (TRIAC)*, cu unul sau doi electrozi de comandă. Folosirea tiristoarelor în domeniul circuitelor logice este limitată de unele dezavantaje: dificultate în comutarea stărilor în c.c., viteză de comutare redusă în comparație cu alte dispozitive (sub 100 KHz), cost unitar ridicat și necesitatea folosirii unor circuite speciale de comandă. În categoria elementelor cu caracteristică de tip S sînt și *diodele cu două baze*, denumite impropriu tranzistor-unijoncțiune (TUJ). Caracteristica volt-amperică a acestor dispozitive este asemănătoare cu a tiristorului, putîndu-se realiza similar circuite logice. Deși sînt mai ieftine decît tiristoarele TUJ-urile sînt limitate de viteză (zeci de KHz) și sînt folosite numai în aplicații singulare.

6. Circuite logice cu dispozitive optoelectronice

Prin dezvoltarea optoelectronicii, în care se folosește cuplajul prin fascicule luminoase între o sursă de lumină și un receptor de lumină, au apărut în ultimul timp și noi dispozitive optoelectronice de comutație. Elementele de bază ale optoelectronicii sînt sursele de lumină, mediile optice de propagare și elementele fotoreceptoare. Sursa de lumină și receptorul cuplate prin intermediul unui mediu optic de propagare reprezintă elementul de bază al optoelectronicii, denumit *optron*. Folosirea cuplajului optic, cu izolarea între comandă și ieșire, permite eliminarea dezavantajului cuplajului galvanic din circuitele de comutație cu dispozitive semiconductoare clasice. În plus, cuplajul optic permite mărirea considerabilă a vitezei de comutație, creșterea cantității de informație și îmbunătățirea stabilității la perturbații.

6.1. Optron de comutație

În opronii de comutație se folosesc drept surse de lumină diodele și condensatoarele electroluminescente comandate electric, iar ca fotoreceptoare fotorezistoare, fotodiode sau fototranzistoare. Semnalele de comandă și ieșire ale unui opron pot fi semnale electrice sau optice. În fig. 6.1a este prezentat un opron de comutație compus dintr-o diodă luminescentă *DL* și un fototranzistor *FT*. Elementele opronului sînt asamblate cu

rășină epoxidică în aceeași incintă. Sub influența curentului de comandă I_i , dioda luminescentă produce un flux luminos de strălucire B care asigură saturarea fototranzistorului. Semnalul de ieșire poate fi curentul I_c sau potențialul de colector al fototranzistorului. În cazul în care se înseriază în circuitul de colector o diodă luminescentă semnalul de ieșire este optic. În fig. 6.1b este prezentată schema unui optron cu intrare și ieșire sub formă de semnal optic. Fotorezistorul FR primește fluxul luminos de comandă de strălucire B_i , sub a cărei influență își micșorează foarte mult rezistența (de la 10^6 — 10^8 ohmi la 300—1 000 ohmi). În acest mod condensatorul electroluminescent CL poate emite lumină cu strălucirea B_e . Optronii din această categorie pot fi folosiți acolo unde nu sînt necesare viteze de comutație ridicate, fotorezistoarele avînd o inerție ridicată (tipic zeci de milisecunde). În prezent cel mai mult fabricat este optronul cu diode luminescente și fototranzistoare. Frecvența de comutare a acestor optroni se situează în jurul a 1 MHz.

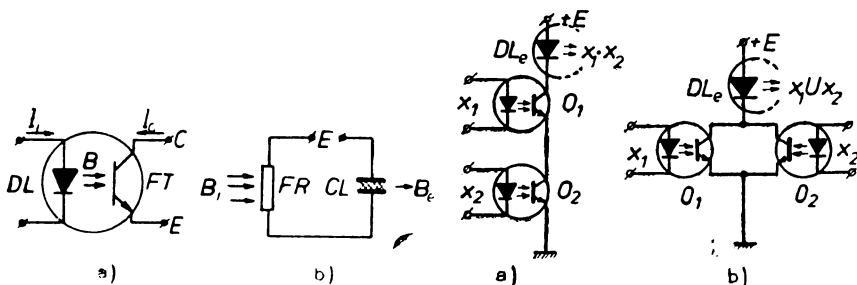


Fig. 6.1

Fig. 6.2

6.2. Realizarea circuitelor logice cu optroni

Cu ajutorul optronilor de comutație se pot realiza circuite logice care să materializeze funcțiile sistemelor complete. În fig. 6.2 sînt prezentate schemele circuitelor logice SI (fig. 6.2a) și SAU (fig. 6.2b) cu două intrări, cu comandă electrică și ieșire optică. În acest mod se obține o separare completă a optronilor de prelucrare logică, O_1 și O_2 , de circuitul de sarcină. Dacă în locul diodelor luminescente din colectorul fototranzistoarelor se folosesc rezistoare, se pot materializa funcțiile NUMAI și NICI. Cuplajul optic în interiorul optronului se poate realiza

nu numai direct ci și prin reacție, folosind o parte din fluxul luminos de ieșire pentru a comanda un alt element din interiorul optronului. În fig. 6.3 sînt prezentate două circuite cu optroni cu reacție internă care permit realizarea atît a funcțiilor directe SI (fig. 6.3a) și SAU (fig. 6.3b) cît și a complementelor acestora. Circuitele sînt realizate cu optroni cu condensatoare electro-luminescente și fotorezistoare. Prin intermediul condensatoarelor luminescente CL_1 în cele două scheme se obțin funcțiile directe. Fotorezistoarele FR_2 comandate prin reacție de la CL_2 asigură mărirea vitezei de comutare iar condensatoarele CL_2 materializarea funcțiilor NIUMAI și respectiv NICI.

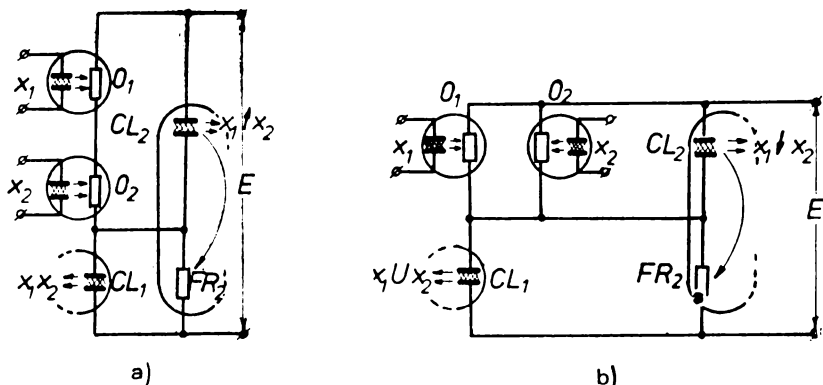


Fig. 6.3

Prin folosirea unor ecrane și a unor optroni auxiliari, convenabil plasați, se pot realiza module universale care să permită trecerea simplă de la un tip de circuit logic la altul. În prezent există tendința de extindere a optoelectronicii prin împletire din ce în ce mai strînsă cu microelectronica, din acest mixaj rezultînd și elemente noi de prelucrare logică.

7. Circuite logice cu miezuri magnetice

La realizarea circuitelor logice se folosesc și materialele magnetice cu ciclu de histeresis dreptunghiular (CHD). În circuitele de comutație se folosesc astfel de materiale sub formă

de tor din bandă de permalloy sau permivar foarte subțire (zeci de μm) sau din ferite. În țara noastră, la Întreprinderea de ferite din Urziceni, se realizează miezuri din ferită pentru folosirea în circuitele de comutație.

Dacă pe un miez magnetic inelar sînt înfășurate w spire străbătute de curentul I , solenația $\theta = wI$ determină un cîmp magnetic de intensitate $H = \frac{wI}{l}$, în care $l = \frac{(d_1 + d_2)}{2}$ este

linia medie de cîmp (fig. 7.1 a). Fluxul magnetic Φ printr-un astfel de miez depinde nelinier de solenație. Dacă se consideră că fluxul este uniform distribuit în secțiunea miezului, inducția magnetică în miez va fi legată de intensitate printr-o relație

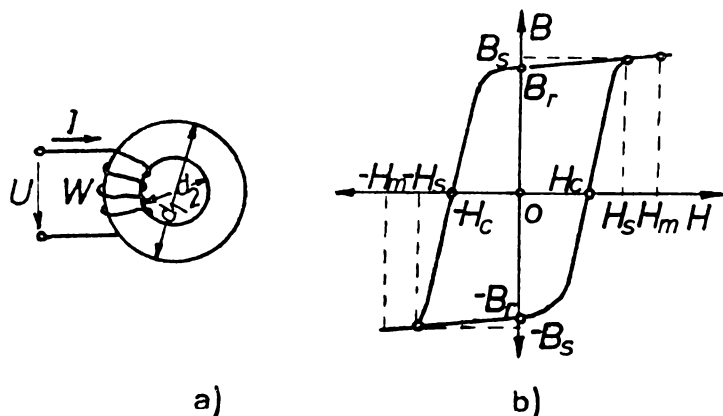


Fig. 7.1

analogă relației dintre flux și solenație. Aceste dependențe în reprezentare grafică constituie caracteristicile statice ale miezului sub formă de cicluri de histererezis. În fig. 7.1b este reprezentată o astfel de caracteristică obținută la modificarea lentă a curentului de magnetizare. Mărimile caracteristice ale miezului sînt valorile cîmpului coercitiv H_c , inducției remanente B_r , inducției de saturație B_s și a cîmpului corespunzător H_s . Pentru $H = H_m > H_s$ practic $B = B_s$ și se obține *ciclul limită*, iar pentru $H < H_s$ au loc *cicluri parțiale* de magnetizare. În lipsa unui cîmp de magnetizare ($H = 0$) miezul magnetizat pe ciclul limită se află în una din cele două stări posibile de magnetizare care

corespund valorilor de inducție remanentă $+B_r$ și $-B_r$. Dacă se atașează stării stabile $+B_r$, cifra binară 1, iar celei de a doua stări stabile $-B_r$, cifra 0, prin comutarea miezului cu ajutorul unui câmp $H > H_s$, un astfel de dispozitiv poate funcționa ca element discret. În plus, datorită păstrării naturale a stării în care este comutat, aceste dispozitive nu necesită o comandă continuă.

Cu cât ciclul de magnetizare a acestor dispozitive se apropie mai mult de forma dreptunghiulară, funcționarea lor se va apropia mai mult de un comutator ideal. În acest sens se definește un *coeficient de rectangularitate*: $k = B_r / B_s$. Se consideră miez cu CHD dacă $k > 0,8$. Pentru miezurile actuale din permalloy $k = 0,8 - 0,87$ și $B_r = 0,8 - 1 T$, iar pentru cele din ferite $k = 0,9 - 0,98$ însă $B_r = 0,2 - 0,27 T$. Datorită abaterii de la forma ideală de ciclu dreptunghiular există dezavantajul apariției *semnalelor parazite*. Astfel, dacă miezul se află în una din stările stabile și i se aplică semnale de magnetizare în același sens, inducția se modifică cu $\Delta B = B_s - B_r$, provocând în înfășurările de pe miez t.e.m. parazite. Un alt dezavantaj al miezurilor magnetice este dependența formei ciclului de temperatură: la o modificare a temperaturii cu $60^\circ C$, k se modifică la ferite între 5 și 30% iar la permalloy pînă la 35%.

Pentru a putea fi utilizate la materializarea funcțiilor logice miezurile magnetice necesită pe lîngă posibilitatea de comutare a miezului pentru înscrierea informației și posibilitatea de evidențiere și transmitere a acesteia. Datorită stocării naturale a informației majoritatea circuitelor logice cu miezuri magnetice funcționează cu impulsuri. Din cele arătate mai sus rezultă necesitatea unui număr minim de trei înfășurări pe un miez:

- o înfășurare de intrare (sau mai multe) care să permită introducerea informației în miez (de obicei prin aducerea miezului în starea 1);

- o înfășurare de evidențiere a informației stocată în miez (prin acțiunea acestei înfășurări, denumită *de tact*, miezul este adus în starea 0);

- o înfășurare de ieșire, care permite transformarea informației miezului în semnal electric și conectarea sarcinii. În fig. 7.2a este prezentată schema simbolică a unui miez magnetic cu înfășurările menționate. Începutul înfășurărilor se marchează, făcîndu-se convenția că tensiunea la bornele unei înfășurări este astfel încît atunci cînd curentul parcurge înfășurarea de la început spre sfîrșit cîmpul magnetic creat de acesta tinde să

comute miezul din 0 în 1. Se observă că înfășurării de tact i se aplică curentul astfel ca miezul să fie adus în zero. Înfășurarea de ieșire preia semnalul atunci când acționează impulsul de tact I_T . De multe ori se structurează această reprezentare a miezului ca în fig. 7.2b. Cifrele binare indică stările care tind să fie stabilite de impulsurile de comandă și de tact. Când numărul de înfășurări pe miez este mai mare se folosește *reprezentarea prin oglinzi* (fig. 7.2c). Miezul este reprezentat prin bara grosă

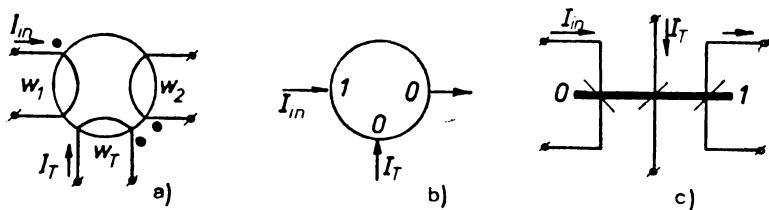


Fig. 7.2

intersectată perpendicular de linii subțiri care reprezintă înfășurările. La intersecția miezului cu înfășurările sînt plasate segmente înclinate la 45° și care au rolul unor oglinzi imaginare. Ele reflectă curenții, asimilați cu raze luminoase, spre 0 sau 1 după cum aceștia tind să aducă miezul în starea 0 sau 1. Tensiunea electromotoare indusă în înfășurarea de ieșire se opune acțiunii impulsului de tact, tinzînd să mențină miezul în starea 1. Pentru realizarea circuitelor logice între miezuri se folosesc circuite de cuplaj realizate cu diode sau tranzistoare.

7.1. Circuite logice cu miezuri magnetice și diode

În cazul acestor circuite singura sursă de energie o constituie sursa pentru impulsurile de tact. După modul cum se conectează această sursă se disting *scheme paralel și serie*.

7.1.1. Circuite logice cu scheme paralel

Conectarea miezurilor magnetice în aceste scheme se face conform cu schema din fig. 7.3. Sursele de tact se conectează la fiecare miez *în paralel* cu înfășurările de intrare și ieșire ale acestora. Se consideră că miezul M_1 se află în starea 1 datorită semnalului aplicat la intrare, iar celelalte miezuri se află

în starea 0. La aplicarea impulsului de tact miezului M_1 , acesta este adus în 0 iar variația de flux produce o astfel de tensiune încît curentul care trece prin dioda D_1 determină comutarea miezului M_2 în starea 1. La aplicarea impulsului de tact lui M_2 , acesta revine în 0, iar prin D_2 circulă un curent care comută pe M_3 în starea 1, ș.a. Diodele asigură transmiterea informației

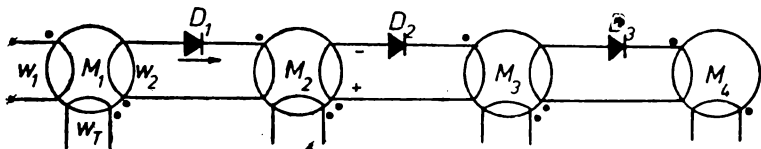


Fig. 7.3

la miezul următor numai la comanda prin tact. În adevăr, în timpul înscrierii informației în M_2 , prin comutarea sa din 0 în 1, în înfășurările de ieșire se induce o t.e.m. care polarizează invers dioda D_2 nepermițînd astfel circulația unui curent prin circuitul de cuplaj dintre M_2 și M_3 .

Din prezentarea funcționării schemei paralel rezultă că transmiterea informației de la un miez la altul trebuie să se facă la momente de timp diferite. Dacă impulsurile s-ar aplica simultan nu s-ar putea realiza transferul de informație deoarece efectul înfășurărilor de intrare ar fi anulat de înfășurărilor de tact. Ar fi astfel necesare atîtea generatoare de tact, cu semnale decalate în timp, cîte miezuri ar avea circuitul. Un astfel de circuit paralel se numește *circuit polifazat sau multitact*. Practic, se folosesc numai două generatoare de tact cu impulsurile decalate în timp sau un singur generator, în care caz însă circuitul de cuplaj asigură întîrzierea necesară. Primele tipuri se numesc *circuite bitact* iar celalalte *circuite monotact*. Atît cu scheme bitact cît și cu scheme monotact se pot realiza circuite logice.

În fig. 7.4 este prezentată schema unui circuit SAU cu două intrări. Miezul M_3 realizează prelucrarea logică iar miezu-

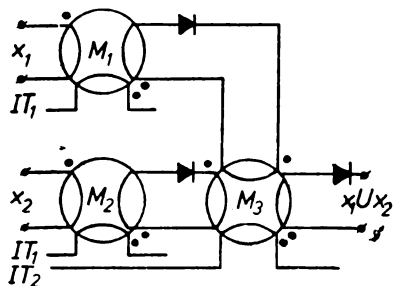


Fig. 7.4

rile M_1 și M_2 asigură separarea semnalelor de intrare de semnalele de ieșire. În primul tact se înscrie informația din primele miezuri în miezul M_3 , iar la aplicarea celui de al doilea tact rezultatul prelucrării se evidențiază la ieșire. Pentru realizarea funcțiilor mai complexe se folosește *generatorul de unități* (fig. 7.5a). Cu ajutorul acestui circuit se poate realiza negarea. Dacă înfășurării de intrare i se aplică tactul $1(IT_1)$, la apariția tactului $2(IT_2)$ se generează la ieșire semnal. În locul lui IT_1 se poate aplica și o

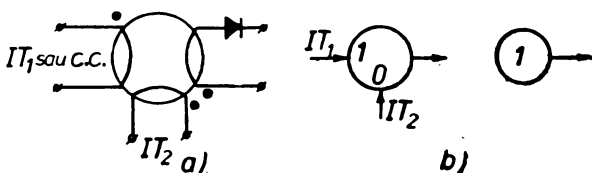


Fig. 7.5

tensiune continuă care să asigure în lipsa semnalului de tact menținerea miezului în starea 1. La aplicarea semnalului de tact IT_2 , care în acest caz trebuie să fie de câteva ori mai puternic decât valoarea minimă necesară comutării, la ieșire apare semnal 1. Astfel, circuitul generează la ieșire semnal 1 la fiecare apariție a semnalului în înfășurarea de tact. Reprezentarea simbolică a generatorului de unități este dată în fig. 7.5b. În fig. 7.6 este prezentat un circuit NICI cu trei intrări realizat cu ajutorul generatorului de unități. Prelucrarea logică se realizează în înfășurările de ieșire ale celor patru miezuri conectate în serie. Se observă că înfășurarea de ieșire a generatorului de unități este conectată în opoziție.

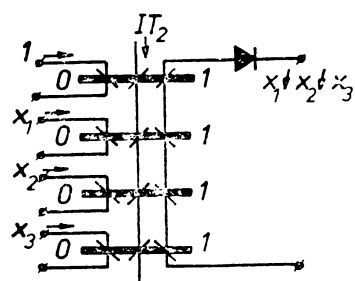


Fig. 7.6

Dacă $x_1 = x_2 = x_3 = 0$ la ieșire generatorul de unități determină apariția semnalului 1 la aplicarea lui IT_2 . Dacă la o intrare sau mai multe se aplică semnal 1 în timpul acționării lui IT_1 (înfășurările nu sînt figurate), la apariția lui IT_2 semnalul produs de generatorul de unități anulează efectul miezurilor de prelucrare logică și la ieșire se obține semnal 0 logic.

Existența a două generatoare de tact precum și a două miezuri pentru unitatea de informație constituie dezavantaje pentru schemele bitact. Circuitele monotact necesită un singur generator de tact și un miez pentru transmiterea unității de informație. În schimb, pentru a asigura separarea în timp între introducerea și transmiterea semnalelor, aceste circuite folosesc celule pasive (obișnuit RC) de întârziere între miezuri. În fig. 7.7a este prezentat un circuit SAU cu două intrări realizat cu schema monotact. Circuitul de întârziere reprezentat prin blocul notat cu τ realizează stocarea informației între două tacte, iar diodele elimină influența dintre miezuri. Prin folosirea generatorului de unități, care se obține similar celui de la schema bitact, se poate realiza negarea. De asemenea, pentru o funcționare sigură, cu ajutorul generatorului de 1 se realizează și circuitul SI (fig. 7.7b). Datorită introducerii negării variabilei x_1 prin generatorul de unități pe calea lui x_2 este necesară introdu-

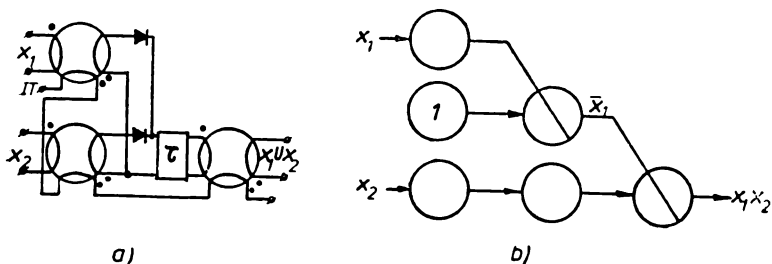


Fig. 7.7

derea unui miez suplimentar în vederea sincronizării în transferul de informație. Deși sînt mai simple decît circuitele logice cu schema bitact, circuitele logice cu schema monotact sînt critice la durata impulsului de tact și au o viteză redusă de funcționare.

7.1.2. Circuite logice cu scheme serie

În cazul acestor circuite, generatorul de tact se conectează în serie cu înfășurările de intrare și ieșire a două miezuri vecine. Alimentarea schemelor serie se realizează cu tensiuni sinusoidale decalate la 180° . O schemă serie este prezentată în fig. 7.8. Datorită diodelor de separare și defazajului între tensiunile de alimentare se asigură eliminarea completă a semnalelor parazite. Se consideră că toate miezurile sînt în starea 0. În prima alter-

nanță pozitivă a lui E_1 , miezurile M_1 și M_3 sînt menținute în 0 și înfășurările lor de ieșire prezintă impedanță minimă (inducțanță mică datorită modificării reduse a inducției de la $-B_r$ la $-B_s$). În schimb miezurile M_2 și M_4 comută în 1. La devenirea pozitivă a lui E_2 , miezurile M_2 și M_4 prezintă prin înfășurările de ieșire impedanțe mari. Din această cauză M_3, M_5, \dots nu sînt

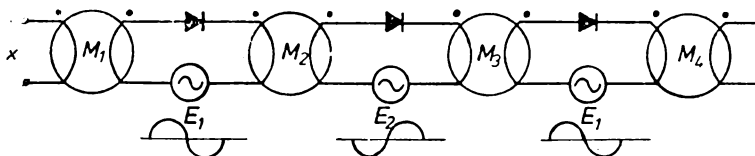


Fig. 7.8

comutate, rămînînd în 0. În schimb curenții determinați de E_2 în înfășurările de ieșire a miezurilor M_2, M_4, \dots aduc aceste miezuri în 0. Astfel, semnalul 0 de la intrare se menține și la ieșire în miezurile cu număr par. Dacă la intrare se aplică un semnal care aduce M_1 în starea 1, la acțiunea lui E_1 , miezul M_2 nu este comutat ci numai M_4, M_6, \dots . După 180° , cînd acționează E_2 , întrucît M_2 este în 0 comută pe M_3 în starea 1 realizînd transferul de informație. În același timp readuce pe M_4 în 0 asigurînd transmisia mai departe a semnalului. Se observă că aceste scheme funcționează în două tacte, semnalul fiind transmis sincron cu E_2 . Pe baza acestor scheme au fost construite sisteme industriale de comutație la frecvența rețelei (sistemele CYPAC și LOGIMAG) Folosirea schemelor serie la realizarea circuitelor logice pre-

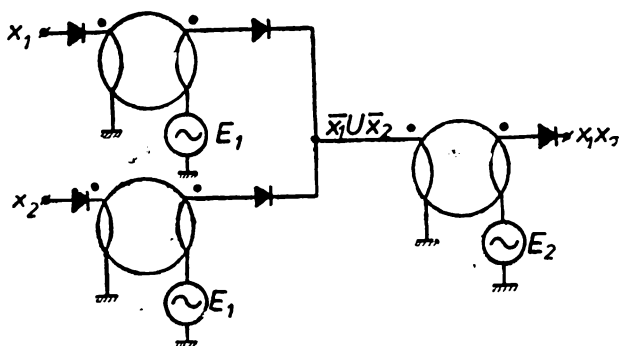


Fig. 7.9

zintă dezavantajul necesității unei stabilități ridicate a amplitudinii surselor de alimentare. În fig. 7.9 este prezentată schema unui circuit SI cu două intrări realizat cu scheme serie. Primele două miezuri realizează negarea variabilelor, la intrarea ultimului miez obținându-se $\overline{x_1} \cup \overline{x_2} = \overline{x_1 x_2}$. În tactul al doilea (E_2) se realizează negarea semnalului aplicat la intrare și se obține $x_1 x_2$ la ieșire.

7.2. Circuite logice cu miezuri și tranzistoare

Deși schemele și circuitele logice realizate cu miezuri și diode se remarcă prin simplitate constructivă și siguranță în exploatare, o serie de dezavantaje limitează folosirea lor :

- necesitatea reducerii transferului invers de informație care conduce la complicarea schemelor și la pierderi suplimentare :
- necesitatea de a dispune de surse de tact de putere pentru asigurarea comutării miezurilor ;
- datorită circuitului pasiv de cuplaj nu este posibil să se obțină un număr mare de ieșiri deoarece lipsește amplificarea.

Dezavantajele menționate pot fi înlăturate prin introducerea în circuitul de cuplaj a unui element activ, care de cele mai multe ori este un tranzistor. Tranzistoarele asigură separarea circuitelor de intrare și de ieșire și amplificarea semnalelor, fapt ce conduce la surse de tact de putere coborâtă. În practică se utilizează mult două scheme de conectare a tranzistorului în circuitul de cuplaj : fără reacție și cu reacție. În schema *fără reacție* tranzistorul îndeplinește rolul de amplificator de impulsuri la transmisia informației de la un miez la altul. În fig. 7.10 este prezentată schema

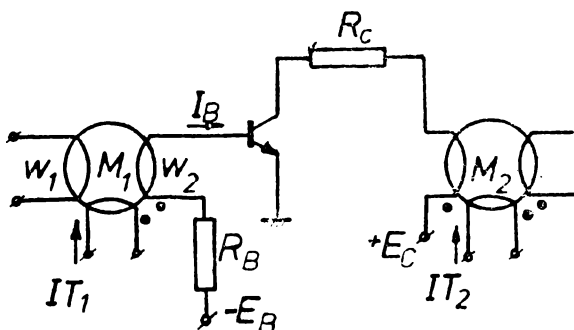


Fig. 7.10

unui astfel de circuit. După înscrierea semnalului 1 în miezul M_1 , la bornele înfășurării de ieșire poate apărea o tensiune perturbatoare care să deschidă tranzistorul și să producă un impuls fals în înfășurarea de intrare a miezului M_2 . Din acest motiv trebuie aleasă valoarea sursei de blocare astfel ca

$$E_B > c_p = w_2 A \frac{B_s - B_r}{t_f},$$

în care A este aria secțiunii miezului, t_f — durata impulsului parazit. Tranzistorul se deschide la aplicarea IT_1 , care determină o t.e.m. în înfășurarea de ieșire și un curent I_B în sensul deschiderii tranzistorului. În acest mod se transmite miezului M_2 informația. În fig. 7.11 este prezentată schema cu reacție, în care curen-

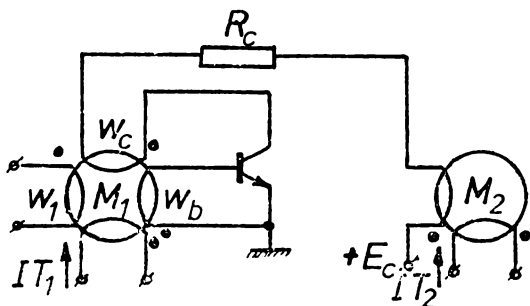


Fig. 7.11

mul de colector parcurge o înfășurare suplimentară w_c . Dacă M_1 se află în starea 1, la aplicarea IT_1 tranzistorul se deschide și prin înfășurarea w_c circulă curent. Această circulație conduce la o modificare a fluxului în miez în același sens ca și IT_1 , de aducere a miezului în 0. Urmează un proces cumulativ, de creștere a curenților în bază și de colector ai tranzistorului și a vitezei de modificare a inducției în miez. Se produce astfel o comutare rapidă a miezului. Când miezul ajunge în starea 0 tensiunile la bornele înfășurării lor w_b și w_c se anulează conducând la blocarea tranzistorului.

Realizarea circuitelor logice cu miezuri și tranzistoare se face simplu, utilizând schemele prezentate. Astfel, generatorul de unități este prezentat în fig. 7.12. La aplicarea semnalului

înfășurării de tact la ieșirea circuitului se generează semnal 1. În fig. 7.13 sînt prezentate circuite SAU și SI cu două intrări realizate cu schema cu reacție. În primul tact semnalele înscrise în miezurile de intrare sînt transmise miezului de ieșire. La aplicarea lui IT_2 rezultatul prelucrării logice este evidențiat la ieșirea circuitului.

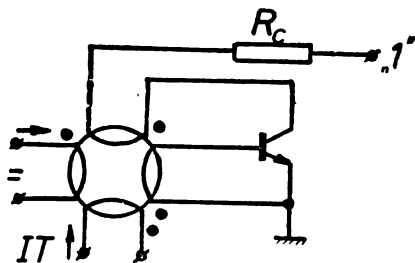


Fig. 7.12

Proprietățile miezurilor magnetice pot fi folosite în materializarea funcțiilor logice utilizînd *logica fluxurilor*. În acest caz miezurile au mai multe deschideri (ferestre) de diferite diametre și cu diferite dispuneri în interiorul miezului. Configurația geome-

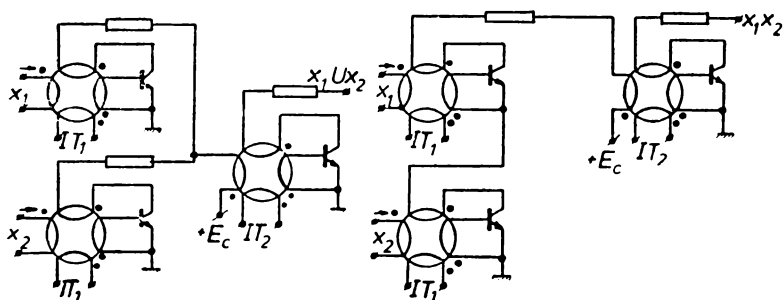


Fig. 7.13

trică a unui astfel de miez poate asigura evitarea cuplajului între diferite înfășurări, făcînd inutil elementele de decuplare (diode, tranzistoare) precum și a circuitelor de cuplaj. În acest caz modificarea stării miezului și prelucrarea logică se realizează numai prin intermediul fluxurilor create de înfășurările dispuse în diferite deschideri ale miezului. Amănunte despre aceste dispozitive, folosite destul de sporadic se pot găsi în [20, 44]. În categoria dispozitivelor apte de a materializa variabile și funcții booleene sînt și dispozitivele care nu folosesc miezuri, dar sînt controlate prin mărimi magnetice. În această categorie intră dispozitivele comandate prin inducția magnetică, cum ar fi:

generatoarele Hall și magnetorezistoarele (gaussitroane). În cazul primelor dispozitive (generatoare Hall), prin intermediul unor înfășurări se modifică tensiunea generată (tensiunea Hall la ieșire) Modificarea rezistivității unui conductor la modificarea câmpului magnetic aplicat transversal (descoperită de Thomson) se folosește la magnetorezistoare. În prezent, prin folosirea semiconductoarelor omogene speciale se pot obține modificări ale rezistivității de 5...15 ori prin modificarea inducției magnetice de la 0 la 1,0 T /20/.

Partea III

COMENZI SECVENȚIALE

Comenzile secvențiale sînt sisteme discrete de comandă. Ca noțiune abstractă acestea constituie o ramură a teoriei sistemelor (dinamice) cuprinsă în *teoria automatelor finite*.

8. Noțiuni de teoria automatelor finite

În acest capitol se vor prezenta principalele noțiuni de teoria automatelor finite necesare studiului teoretic al comenzilor secvențiale.

Definiție. Un automat (automaton) este un 5-uplu

$$A = (X, S, Z, f, g),$$

în care X, S, Z sînt mulțimi nevide avînd următoarele semnificații:

X — este o mulțime de intrări (alfabet de intrare),

S — o mulțime de stări ale automatului,

Z — o mulțime de ieșiri (alfabet de ieșire),

iar

$f: S \times X \rightarrow \mathcal{P}^*(S)$ — funcția de tranziție (de trecere).

$g: S \times X \rightarrow \mathcal{P}^*(Z)$ — funcția de ieșire (de răspuns).

Din definiție rezultă că funcțiile f și g sînt aplicații ale produsului cartezian al mulțimilor S și X pe mulțimea părților nevide (\mathcal{P}^*) ale lui S și respectiv Z . Avînd în vedere definiția produsului cartezian a două mulțimi rezultă că aceste funcții sînt mulțimi de perechi ordonate (s, x) , $s \in S, x \in X$.

Dacă $\forall s \in S, x \in X$ avem cardinalul mulțimilor $|f(s, x)| = 1$ și $|g(s, x)| = 1$, automatul A se numește *determinist*. În caz contrar A se numește *nedeterminist*. Dacă $\forall s \in S$ și $x \in X$, avem $|f(s, x)| = 1$, automatul A se numește *S-determinist*, iar dacă $|g(s, x)| = 1$ automatul se numește *Z-determinist*. Un automat determinist este *S-* și *Z-determinist*.

Definiție. Un automat A este *finit* dacă mulțimile X, S, Z aparținând automatului sînt finite.

Deci, un automat finit are un alfabet de intrare și de ieșire și o mulțime de stări finite. Un automat finit se poate imagina ca o „cutie neagră” (black box) prevăzută cu un număr finit de borne exterioare accesibile și capabilă să posede un număr finit de stări interne (elementele mulțimii S). Aplicînd un semnal de intrare (element al mulțimii X) la bornele de intrare, sistemul considerat elaborează un răspuns (element din alfabetul Z) și trece într-o nouă stare internă. Răspunsul, precum și starea în care trece automatul aparțin mulțimilor $f(s, x)$ și $g(s, x)$, s fiind starea prezentă.

Din cele prezentate pînă acum rezultă că evoluția în timp a unui automat finit este caracterizată de mulțimea stărilor interne S și funcțiile de tranziție și de ieșire. Aceste funcții sînt denumite și *funcții caracteristice* ale automatului.

În cazul mai multor automate cu aceleași mulțimi X și Z , dacă se notează $Aut(X, Z)$, orice automat $A \in Aut(X, Z)$ poate fi descris simplificat prin tripletul: $A = (S, f, g)$.

8.1. Modele de automate finite

Pentru a se evidenția evoluția în timp a automatelor finite s-au realizat diferite definiții ale funcțiilor caracteristice. A rezultat astfel noțiunea de *model de automat*. Primul model a fost elaborat de Mealy în anul 1955. În 1956 Moore a elaborat un alt model, care se va vedea că este un caz particular al modelului Mealy și care are numeroase aplicații în tehnică. Aceste modele matematice au primit denumirile de *model Mealy* și *model Moore*, fiind în prezent cele mai folosite.

Definiție. Un automat finit Moore este un 5-uplu.

$$A = (X, S, Z, f, g),$$

cu X, S, Z, f, g avînd semnificațiile enunțate mai înainte și cu următoarele expresii explicite pentru funcțiile caracteristice:

$$(8.1) \quad \begin{cases} s(v+1) = f[x(v), s(v)] \\ z(v) = g[s(v)], \end{cases}$$

unde $x(v)$, $s(v)$ și $z(v)$ sînt simbolul de intrare, starea și simbolul de ieșire la momentul t_v , $v=1, 2, 3, \dots$

N.B. Mulțimea de valori pe care le poate lua o variabilă v se numește *alfabetul variabilei*. Fiecare element al alfabetului se numește *simbol*. Din definiție rezultă că pentru automatele Moore starea următoare este definită de simbolul de intrare și de starea prezentă în timp ce ieșirea este determinată în prezent, numai de starea prezentă.

Definiție. Un *automat finit Mealy* este un 5-uplu

$$A = (X, S, Z, f, g)$$

cu semnificațiile cunoscute avînd funcțiile caracteristice de forma:

$$(8.2) \quad \begin{cases} s(v) + 1 = f[x(v), s(v)] \\ z(v) = g[x(v), s(v)]. \end{cases}$$

Conform definiției, la un automat Mealy ieșirea este determinată atît de starea prezentă cît și de intrare iar funcția de tranziție este aceeași ca la automatele Moore.

Din punct de vedere matematic putem lua :

$$g(s, x) = g_1(s), \quad s \in S, x \in X,$$

rezultînd că automatul Moore este un caz special de automat Mealy.

Cele două modele definite mai sus pot fi reprezentate unitar prin *modelul Starke* :

Definiție. Un *automat finit Starke* este un 4-uplu

$$A = (X, S, Z, k),$$

în care pentru X, S, Z se păstrează semnificațiile știute, iar

$$k : S \times X \rightarrow \mathcal{D}^*(S \times X).$$

Funcția k este mulțimea căreia îi aparține ieșirea la momentul t_v și starea la momentul t_{v+1} , în funcție de starea și intrarea automatului la momentul t_v . Evident, în cazul determinist automatele Mealy și Moore pot fi descrise prin automate Starke luînd :

$$k(s, x) = [f(s, x), g(s, x)]$$

pentru automatul Mealy și

$$k(s, x) = [f(s, x), g(s)]$$

pentru automatul Moore.

De multe ori nu interesează ieșirea unui automat ci numai evoluția sa internă cu tranzițiile între stări. În acest caz se definește *automatul de stare* folosind noțiunea de semiautomat.

Definiție. Un semiautomat este un triplet

$$S(A) = (X, S, f),$$

în care X , S și f au semnificațiile menționate anterior.

Este evident că noțiunea de semiautomat este mai largă înglobând noțiunea de automat. Unui automat A i se poate atașa un singur semiautomat $S(A)$ dar unui semiautomat, prin adoptarea alfabetului Z și a funcției g , mai multe automate. Astfel, pentru un semiautomat rămâne esențială funcționarea internă, automatale atașate nefiind unice.

Dacă se acceptă că stările unui automat finit determinist sînt realizabile fizic printr-un bloc denumit *memorie*, atunci modelul unui automat poate fi reprezentat prin schema bloc din fig. 8.1. Prin M s-a simbolizat blocul memorie care determină stările interne ale automatului. De asemenea, sînt prezentate blocurile care calculează funcțiile de tranziție și de ieșire. Schema bloc din fig. 8.1 reprezintă de fapt un automat Mealy. Dacă

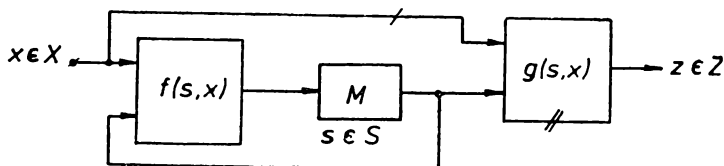


Fig. 8.1

lipsește conexiunea pentru semnalul de intrare $x()$ în blocul de calcul al funcției de ieșire se obține imaginea unui automat Moore. În cazul în care lipsește complet blocul care determină funcția $g(s, x)$, se obține imaginea unui automat de stare.

Un automat pentru care funcția de ieșire este de forma :

$$g[s(v), x(v)] = g[x(v)]$$

se numește *automat fără memorie* sau *trival*. În acest caz funcția de tranziție și variabilele intermediare determinate de legătura inversă nu au nici o influență și prin urmare noțiunea de stare este de prisos. Deoarece pentru automatele fără memorie ieșirea la un moment dat este determinată numai de combinația mărimilor aplicate la intrare în acel moment, astfel de automate se mai numesc *combinacionale*. Aceste automate sînt realizabile fizic prin *circuite logice combinacionale*.

Blocurile care elaborează funcțiile $f(s, x)$ și $g(s, x)$ într-un automat netrivial sînt de tip combinational. Automatele finite cu memorie sînt realizabile fizic prin *circuite logice secvențiale*, care, conform cu fig. 8.1, conțin circuite combinacionale și elemente de memorie. Convers, circuitele secvențiale de comandă sînt descrise prin modelul matematic de automat finit cu memorie iar circuitele combinacionale prin modelul de automat finit fără memorie.

8.2. Reprezentări ale automatelor finite

Studiul automatelor finite și deci și a comenzilor secvențiale este avantajos să se facă pe reprezentările acestora. Cele mai folosite reprezentări ale automatelor sînt reprezentările prin graf și prin tabel.

8.2.1. Reprezentarea automatelor finite prin grafuri

Unui automat finit i se poate atașa un graf orientat și marcat. Asocierea grafului G unui automat finit se face în modul următor :

a) mulțimii stărilor automatului $A = (X, S, Z, f, g)$ i se atașează mulțimea vîrfurilor (nodurilor) grafului G ;

b) dacă stările s_i și s_j , $s_i, s_j \in S$, sînt legate prin relația $s_j = f(s_i, x_i), x_i \in X$, atunci vîrfurile s_i este legat de vîrfurile s_j printr-un arc orientat de la s_i în s_j . Arcul (s_i, s_j) astfel obținut va fi marcat cu semnalul de intrare care a determinat tranziția între s_i și s_j ;

c) semnalele de ieșire ale automatului se specifică în graf în funcție de tipul automatului, și anume : pentru automatele Moore ieșirile se vor marca în graf pentru stările corespunzătoare, iar pentru automatele Mealy dacă $z_i = g(s_i, x_i)$ arcul (s_i, s_j) va fi marcat și cu ieșirea z_i .

Exemplul 1. Fie $A = (X, Z, f, g)$ cu $X = \{x_1, x_2\}$, $S = \{s_1, s_2, s_3\}$ $Z = \{z_1, z_2\}$ și funcțiile de tranziție și de ieșire date prin tabellele

f	s_1	s_2	s_3
x_1	s_2	s_3	s_1
x_2	s_1	s_3	s_2

g	s_1	s_2	s_3
x_1	z_1	z_1	z_2
x_2	z_1	z_2	z_1

Să se stabilească graful atașat automatului.

Conform cu (8.1) automatul este de tip Mealy. Graful va avea trei vîrfuri (fig. 8.2) și se construiește conform algoritmului dat mai sus. În graful atașat automatului se pot distinge arce *divergente*, *convergente* și *reflectate*, dacă acestea sînt privite în raport cu stările din care pleacă și în care ajung. Graful se marchează prin semnalele de intrare care determină tranzițiile între stări și prin semnalele de ieșire corespunzătoare acestor tranziții. Se observă că pentru un automat Mealy semnalele la ieșire apar în timpul tranzițiilor dintre stări. Arcele reflectate indică că acele stări sînt *stabile* pentru semnalele respective de intrare, în timp ce acele stări care nu posedă astfel de arce sînt stări de *tranzitie*.

Exemplul 2. Să se stabilească graful atașat automatului $A = (X, S, Z, f, g)$ cu $X = \{x_1, x_2\}$, $S = \{s_1, s_2, s_3\}$, $Z = \{z_1, z_2\}$ și cu funcțiile f și g date în tabelele următoare :

f	s_1	s_2	s_3
x_1	s_3	s_3	s_1
x_2	s_1	s_3	s_2

g	s_1	s_2	s_3
	z_1	z_1	z_2

Conform cu (8.2) automatul este de tip Moore și graful atașat este reprezentat în fig. 8.3. Se observă că în cazul acestor automate semnalele de ieșire apar numai la atingerea unei stări, arcele fiind marcate numai cu semnalele de intrare.

Referitor la reprezentarea prin graf a unui automat finit, mai pot fi definite următoarele stări particulare :

— *stare tranzitorie* este acea stare care nu are nici un arc convergent, ci cel puțin un arc divergent. Dintr-o astfel de stare se poate trece în cel puțin una din celelalte stări, dar nu mai poate fi atinsă odată părăsită ;

— *stare persistentă* — nu are arce divergente, ci cel puțin un arc convergent. O astfel de stare odată atinsă nu mai poate fi părăsită ;

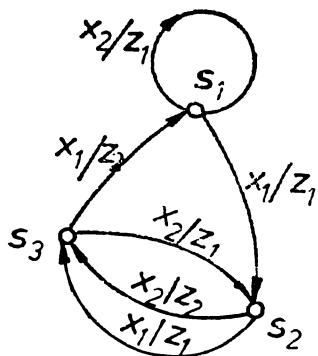


Fig. 8.2

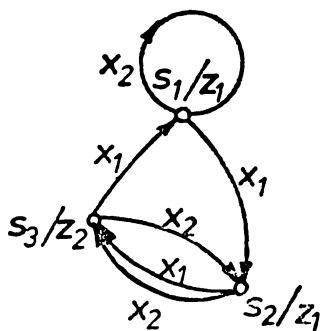


Fig. 8.3

— stare izolată — poate avea cel puțin un arc reflectat, dar nu are nici arce divergente și nici convergente. O astfel de stare nu poate fi nici atinsă și nici părăsită. Stările izolate, dacă apar în graf, fiind inaccesibile se pot elimina.

8.2.2. Reprezentarea automatelor finite prin tabele

Unui automat $A = (X, S, Z, f, g)$ i se poate atașa un tabel $T = (X, S, Z, f, g)$ construit în modul următor :

- liniile tabelului corespund stărilor automatului ;
- coloanele tabelului corespund semnalelor de intrare ;
- dacă s_i este o stare a automatului iar x_j unul din semnalele de intrare, la intersecția liniei „ i ” cu coloana „ j ” în tabel se trece funcția de tranziție $f(s_i, x_j)$;

d) pentru automatele Mealy la intersecția liniei „ i ” cu coloana „ j ” se trece funcția de ieșire $g(s_i, x_j)$ iar pentru automatele Moore se introduce o coloană suplimentară în care se trece funcția $g(s)$.

Exemplu. Să se reprezinte prin tabel automatele Mealy și Moore date în exemplele anterioare.

Corespunzător algoritmului formulat mai sus, tabele atașate celor două automate sînt următoarele :

Mealy		
Intrări	x_1	x_2
Stări		
s_1	s_2/z_1	s_1/z_1
s_2	s_3/z_1	s_3/z_2
s_3	s_1/z_2	s_2/z_1

Moore			
Intrări	x_1	x_2	$g(s)$
Stări			
s_1	s_2	s_1	z_1
s_2	s_3	s_3	z_1
s_3	s_1	s_2	z_2

Pînă acum s-au considerat cunoscute funcțiile de tranziție și de ieșire pentru toate perechile $(s, x) \in S \times X$. În acest caz se spune că un automat este *complet definit*. Dacă funcțiile f și g ale unui automat finit sînt definite numai pentru o submulțime a perechilor (s, x) automatul respectiv este *parțial definit*. În acest caz în reprezentarea tabelară situațiile de nedefinire sînt specificate printr-o linie.

9. Circuite logice combinaționale

Așa cum s-a arătat în capitolul precedent, circuitele logice combinaționale constituie modelul fizic al automatelor finite fără memorie. Absența memoriei face ca funcționarea acestui tip de automat să nu depindă de timp. În cadrul comenzilor secvențiale interesează acele automate la care elementele mulțimilor X și Z au valori conținute numai în mulțimea cu două valori $x, z \in \{0, 1\}$. În acest caz circuitele logice combinatorii realizează la ieșire funcții booleene care sînt combinații ale variabilelor binare de la intrări.

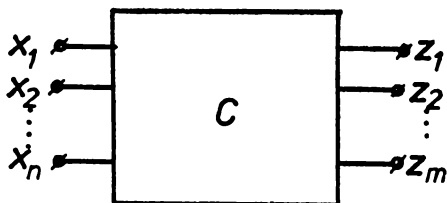


Fig. 9.1

Se consideră un circuit combinațional cu n intrări și m ieșiri. În fig. 9.1 este dată o reprezentare prin schemă bloc a circuitului. Pentru acest circuit se poate scrie setul de funcții:

$$\begin{cases} z_1 = f_1(x_1, x_2, \dots, x_n) \\ z_2 = f_2(x_1, x_2, \dots, x_n) \\ \dots \\ z_m = f_m(x_1, x_2, \dots, x_n), \end{cases}$$

unde f_1, f_2, \dots, f_m sînt funcții booleene de argumentele x_1, x_2, \dots, x_n , denumite și *funcții proprii* ale circuitului.

Legat de realizabilitatea fizică a circuitelor logice combinate se pot formula două tipuri de probleme: *de analiză și de sinteză*. În continuare se vor dezvolta pe rând cele două tipuri de probleme.

9.1. Analiza circuitelor combinate

În cadrul problemelor de analiză se cunoaște structura circuitelor și se cere să se stabilească valorile posibile la ieșiri pentru toate combinațiile posibile ale valorilor semnalelor de la intrări. Problemele de analiză se soluționează căutând expresiile funcțiilor booleene corespunzătoare semnalelor de ieșire, ca apoi să se poată stabili valorile lor pentru diferitele combinații ale valorilor argumentelor. În general, funcțiile proprii care se stabilesc plecând de la configurația circuitului se obțin sub formă normală. Pentru a putea stabili toate valorile acestor funcții pentru toate combinațiile valorilor argumentelor este necesară evidențierea termenilor canonici. Rezultă că din punct de vedere matematic problemele de analiză conduc la *necesitatea dezvoltării* funcțiilor booleene în forme canonice, pentru a le pune în evidență constituenții. Din cele expuse mai sus se pot formula etapele care trebuie parcurse în rezolvarea problemelor de analiză:

1. Cunoscând logigrama circuitului se determină din aproape în aproape funcțiile corespunzătoare ieșirilor.

2. Se dezvoltă funcțiile proprii ale circuitului sub formă canonică.

3. Folosind tabelul combinatorial, diagrama Karnaugh sau simbolul de marcare se stabilesc valorile funcțiilor pentru toate combinațiile posibile ale argumentelor.

4. Etapă neobligatorie, dar de multe ori interesantă din punct de vedere practic, de a stabili expresiile minimale ale funcțiilor canonice în vederea unor posibile simplificări. Această etapă vizează posibilitatea realizării unui circuit echivalent cu primul dar mai economic. Dar, în acest caz este necesar să se cunoască dacă nu sînt și alte criterii de realizare mai tari decît cel economic (fiabilitate, funcționare fără hazard).

9.1.1. Analiza circuitelor combinate realizate cu circuite logice NU, SI, SAU

Analiza acestor circuite se realizează conform algoritmului descris mai sus. Se va exemplifica aplicarea metodologiei considerînd circuitul dat prin logigrama din fig. 9.2.

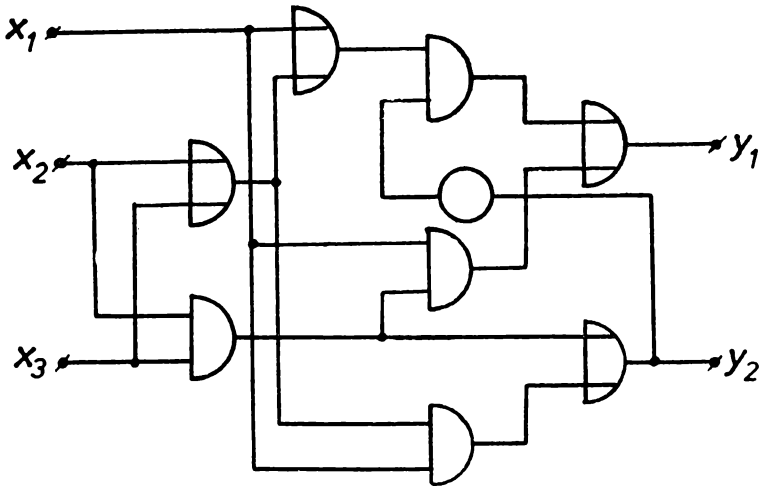


Fig. 9.2

1. Cunoscînd semnificațiile simbolurilor atașate circuitelor logice (v. tab. 2.2), plecînd de la intrări către ieșiri, se stabilesc expresiile funcțiilor booleene corespunzătoare celor două ieșiri:

$$y_1 = x_1 x_2 x_3 \cup (x_1 \cup x_2 \cup x_3) \bar{y}_2$$

$$y_2 = x_2 x_3 \cup x_1 (x_2 \cup x_3).$$

Aplicînd proprietățile algebrei booleene cele două funcții de mai pot scrie:

$$y_1 = x_1 x_2 x_3 \cup (x_1 \cup x_2 \cup x_3) (\bar{x}_1 \bar{x}_2 \cdot \bar{x}_1 \bar{x}_3 \cdot \bar{x}_2 \bar{x}_3)$$

$$y_2 = x_1 x_2 \cup x_1 x_3 \cup x_2 x_3,$$

sau

$$y_1 = x_1 x_2 x_3 \cup x_1 \bar{x}_2 \bar{x}_3 \cup \bar{x}_1 x_2 \bar{x}_3 \cup \bar{x}_1 \bar{x}_2 x_3$$

$$y_2 = x_1 x_2 \cup x_1 x_3 \cup x_2 x_3.$$

Se observă că funcția y_1 a rezultat sub formă canonică disjunctivă, înșă y_2 este sub formă normal-disjunctivă și trebuie dezvoltată.

2. Avînd în vedere cele arătate în capitolul 3.3, pentru funcția y_2 se obține:

$$y_2 = x_1 x_2 (x_3 \cup \bar{x}_3) \cup x_1 (x_2 \cup \bar{x}_2) x_3 \cup (x_1 \cup \bar{x}_1) x_2 x_3 = x_1 x_2 x_3 \cup x_1 x_2 \bar{x}_3 \cup x_1 \bar{x}_2 x_3 \cup \bar{x}_1 x_2 x_3.$$

Astfel, FCD ale celor două funcții sînt :

$$y_1 = x_1 x_2 x_3 \cup x_1 \bar{x}_2 \bar{x}_3 \cup \bar{x}_1 x_2 \bar{x}_3 \cup \bar{x}_1 \bar{x}_2 x_3$$

$$y_2 = x_1 x_2 x_3 \cup x_1 x_2 \bar{x}_3 \cup x_1 \bar{x}_2 x_3 \cup \bar{x}_1 x_2 x_3$$

3. Valorile funcțiilor y_1 și y_2 pentru toate combinațiile posibile ale argumentelor sînt prezentate în tabelul 9.1 sau folosind diagramele Karnaugh în fig. 9.3. Avînd în vedere cele arătate în cap. 2.3.4 se poate scrie și simbolul de marcarea pentru cele două funcții :

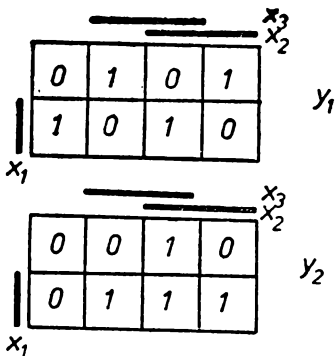


Fig. 9.3

Tab. 9.1

x_1	x_2	x_3	y_1	y_2
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

$$y_1 = D_{\substack{x_1, x_2, x_3 \\ 1 : (1, 2, 4, 7) \\ 0 : (0, 3, 5, 6)}} \quad \text{și} \quad y_2 = D_{\substack{x_1, x_2, x_3 \\ 1 : (3, 5, 6, 7) \\ 0 : (0, 1, 2, 4)}}$$

4. Folosind, spre exemplu, diagramele Karnaugh și minimizînd fiecare funcție se obțin expresii similare cu cele obținute în prima etapă. Deci, circuitul considerat are configurația cea mai economică.

Din inspectarea tabelului 9.1 se poate constata că y_1 corespunde sumei logice a argumentelor. Avînd în vedere și valorile lui y_2 se poate afirma că circuitul analizat este un sumator

pentru două cifre binare (de pe rangul k a două numere binare : y_1 corespunde sumei, iar y_2 transportului către rangul $k + 1$). Analiza circuitelor combinaționale folosind metoda simbolică

O posibilitate avantajoasă de analiză a acestor circuite constă în folosirea metodei simbolice [40], care permite operarea cu numere de stare și cu simbolul de marcarea al unei FB. Considerînd operațiile din algebra booleană, se poate arăta că simbolul de marcarea se poate folosi ca operator de calcul. Astfel, pentru simbolul D de marcarea sînt valabile relațiile :

$$(9.1) \quad \begin{cases} D_{(n_1, n_2, n_k)}^{x_1, x_2, \dots, x_n} \cup D_{(n_1, n_2)}^{x_1, x_2, \dots, x_n} = D_{(n_1, n_2, n_k)}^{x_1, x_2, \dots, x_n} \\ D_{(n_1, n_2, n_k)}^{x_1, x_2, \dots, x_n} \cap D_{(n_1, n_2)}^{x_1, x_2, \dots, x_n} = D_{(n_1)}^{x_1, x_2, \dots, x_n} \\ \bar{D}_{(N)}^{x_1, x_2, \dots, x_n} = D_{(\bar{N})}^{x_1, x_2, \dots, x_n} \end{cases}$$

Avînd în vedere relația de definiție a simbolului D (v. relația 2.18) primele două expresii din (9.1) se pot scrie :

$$(9.2) \quad (n_1 \cup n_2 \cup n_k)^* \cup (n_1 \cup n_3) = (n_1 \cup n_2 \cup n_3 \cup n_k) \\ (n_1 \cup n_2 \cup n_k) \cap (n_1 \cup n_3) = n_1,$$

a căror veridicitate este evidentă întrucît $n_i \cap n_j = 0, \forall i \neq j$. Pentru ultima relație din (9.1) considerînd simbolul D operator, conform principiului contradicției se poate scrie :

$$(9.3) \quad \overline{D}_{(N)}^{x_1, x_2, \dots, x_n} \cap D_{(N)}^{x_1, x_2, \dots, x_n} = 0.$$

Notînd $\overline{D}_{(N)}^{x_1, x_2, \dots, x_n} = D_{(N')}^{x_1, x_2, \dots, x_n}$, relația (9.3) devine :

$$D_{(N')}^{x_1, x_2, \dots, x_n} \cap D_{(N)}^{x_1, x_2, \dots, x_n} = 0,$$

fiind adevărată dacă și numai dacă numerele de ordine N' și N sînt disjuncte, deci dacă numerele de stare cuprinse în cele două simboluri de marcare sînt disjuncte. Deci, relația este adevărată dacă N' este complementul lui N , fapt ce implică $N' = \overline{N}$ și $\overline{D}_{(N)}^{x_1, x_2, \dots, x_n} = D_{(\overline{N})}^{x_1, x_2, \dots, x_n}$.

Conform relațiilor (9.1) și (9.2), folosirea simbolului de marcare ca operator conduce la operarea cu numerele de stare cuprinse în acesta. Admițînd o anumită ordine pentru variabilele de intrare ale unui circuit, conform celor arătate mai sus, fiecare modul logic realizează operarea cu numerele de stare atașate variabilelor. Pentru simbolul D interesează numai valorile de 1 ale variabilelor, care, pentru ordinea stabilită a acestora, se pot stabili din tabelul combinațional. Astfel, pentru trei variabile, conform cu tab. 9.2, rezultă numerele de stare atașate :

$$x_1 \leftrightarrow (4, 5, 6, 7) \quad x_2 \leftrightarrow (2, 3, 6, 7) \quad x_3 \leftrightarrow (1, 3, 5, 7)$$

Tab. 9.2

n_k	x_1	x_2	x_3	n_k	x_1	x_2	x_3
0	0	0	0	4	1	0	0
1	0	0	1	5	1	0	1
2	0	1	0	6	1	1	0
3	0	1	1	7	1	1	1

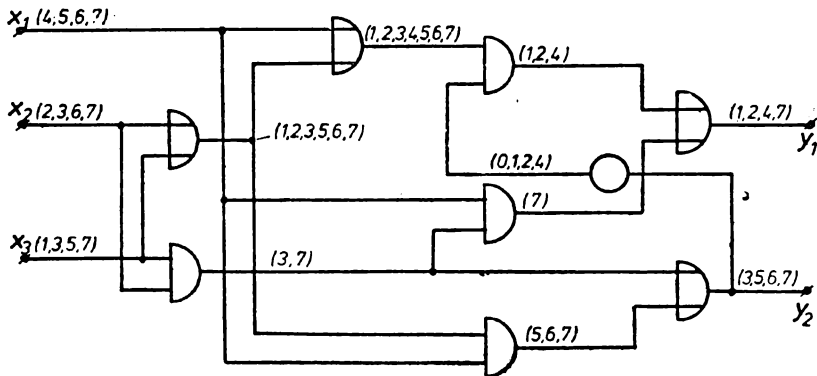


Fig. 9.4

Cunoscînd numerele de stare atașate variabilelor de intrare ale circuitului analizat și avînd în vedere (9.1), analiza circuitelor realizate cu module *SI*, *SAU*, *NU* comportă o singură etapă. Astfel, în raport cu metoda generală descrisă anterior, metoda simbolică de analiză este o metodă globală, permițînd obținerea directă a valorilor funcțiilor de ieșire ale circuitului. În mod obișnuit analiza se realizează direct pe logigramă. În cazul circuitului considerat în exemplul precedent în fig. 9.1, analiza realizată prin metoda simbolică este prezentată în fig. 9.4. Se observă că la ieșirea fiecărui circuit logic se indică numerele de stare rezultate în urma operării cu numerele de stare de la intrări folosind (9.1). Rezultă astfel numerele de stare pentru care y_1 și y_2 au valoarea 1 și care coincid cu cele stabilite prin metoda generală.

9.1.2. Analiza circuitelor combinaționale realizate cu circuite logice *NICI* sau *NUMAI*

Analiza acestor scheme se poate realiza într-un mod analog analizei schemelor realizate cu module *SI*, *SAU*, *NU*. Astfel, avînd în vedere relațiile de definiție ale funcțiilor *NICI* și *NUMAI* se pot stabili expresiile *FB* pentru ieșirile circuitului combinațional considerat. Aplicînd proprietățile algebrei booleene se aduc *FB* obținute la forme canonice. În acest mod se pot stabili valorile funcțiilor pentru toate combinațiile posibile ale semnalelor de intrare.

Exemplu. Să se analizeze circuitul combinațional din fig. 9.5 realizat cu module *NUMAI*.

1. Expresia funcției booleene pentru ieșirea circuitului se obține imediat :

$$y = \overline{b \cdot bc \cdot c \cdot bc \cdot a \cdot c \cdot bc} = (\overline{b \cup bc})(\overline{c \cup bc}) \cup a(\overline{c \cup bc}) = \overline{bc} \cup bc \cup a\overline{bc},$$

care fiind o formă normală trebuie dezvoltată.

2. Prin dezvoltare, se obține :

$$y = (a \cup \overline{a})\overline{bc} \cup (a \cup \overline{a})bc \cup a(b \cup \overline{b})\overline{c} \cup abc = abc \cup abc \cup \overline{a}\overline{bc} \cup \overline{a}bc \cup \overline{a}\overline{bc}.$$

3. Toate valorile pe care le poate lua funcția se pot determina folosind diagrama Karnaugh (fig. 9.6) sau simbolul de marcare :

$$y = D_1^{abc} : \begin{matrix} (0, 3, 4, 6, 7) \\ 0 : (1, 2, 5) \end{matrix}$$

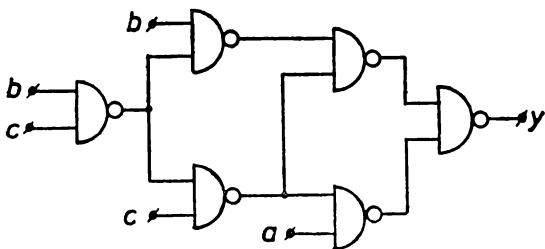


Fig. 9.5

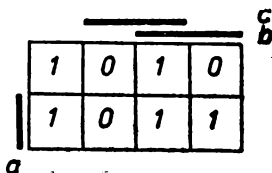


Fig. 9.6

Analiza circuitelor realizate cu module *NICI* sau *NUMAI* se poate efectua și prin metoda simbolică, structurând simbolul modulelor *NICI* sau *NUMAI* conform relațiilor de definiție a funcțiilor respective : $NICI = SAU - NU$ și $NUMAI = SI - NU$. Transformând schema dată cu module *NICI* sau *NUMAI* într-o schemă cu module *SI*, *SAU*, *NU* se poate aplica aceeași metodologie ca și la schemele combinaționale cu module din tipul primar.

Exemplu. Să se analizeze prin metoda simbolică circuitul combinațional din fig. 9.5.

În fig. 9.7 este prezentată schema echivalentă a circuitului cu module $NUMAI = SI - NU$. Adoptând ordinea naturală a-b-c pentru variabile, analiza se realizează direct pe logică.

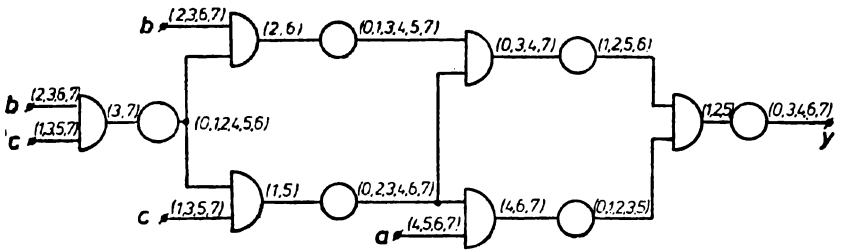


Fig. 9.7

În cazul circuitelor combinaționale complexe ambele metode de analiză prezentate pînă acum devin greu de folosit. Se poate folosi o altă posibilitate de analiză care decurge din proprietățile funcțiilor *NICI* și *NUMAI*. Astfel:

1) Deoarece $\bar{x} = \bar{x}$, $\bar{\bar{x}} = x \downarrow x$ și $\bar{x} = \bar{x} \cup \bar{x} = x \uparrow x$, orice circuit *NICI* sau *NUMAI* este un inversor logic. Rezultă că semnalul obținut după un număr impar de nivele logice se obține negat, iar după un număr par de nivele logice nenegat.

2) În cazul circuitelor logice *NICI* care realizează operația

$$x_1 \downarrow x_2 = \overline{x_1 \cup x_2} = \bar{x}_1 \cdot \bar{x}_2,$$

după parcurgerea unui număr par de nivele logice se obține funcția *SAU* a argumentelor, iar după un număr impar de nivele se obține funcția *SI* cu variabilele de intrare negate.

3) Pentru circuitele logice *NUMAI*, definite prin relațiile echivalente

$$x_1 \uparrow x_2 = \overline{x_1 x_2} = \bar{x}_1 \cup \bar{x}_2,$$

după un număr par de nivele logice se obține funcția *SI*, iar după un număr impar de nivele se obține funcția *SAU* cu variabilele negate.

În tabelul 9.3 sînt sintetizate cele stabilite mai sus. Avînd în vedere cele demonstrate, analiza oricărei scheme combinaționale realizată cu module *NICI* sau *NUMAI* poate fi redusă la analiza unei scheme echivalente cu module *SI*, *SAU*, *NU*. În acest caz este necesar ca în logigramă să se indice nivelele logice, numerotîndu-le de la ieșire spre intrare și începînd întotdeauna cu nivel logic impar.

Exemplu. Să se analizeze circuitul combinațional cu logigrama reprezentată în fig. 9.8.

Stabilind nivelele logice și ținând cont de tabelul 9.3, se obțin modulele echivalente *SI* și *SAU* și variabilele de intrare. Corespunzător logigramei echivalente se stabilește imediat funcția de ieșire :

$$y = (x_1 \cup x_2 \cup \bar{x}_3) [(x_1 \cup x_2) \bar{x}_3] \bar{x}_3 = x_1 \bar{x}_2 \bar{x}_3.$$

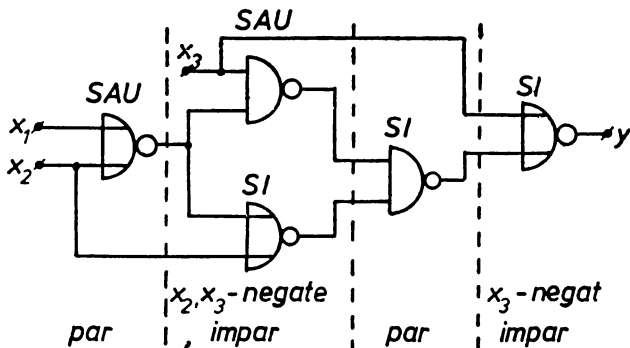


Fig. 9.8

Tab. 9.3

		Funcția care se realizează	
Tip circuit	Nivelul logic	par	impar
	NICI NUMAI		SAU ȘI
Variabile de intrare		nenegate	negate

În continuare analiza se desfășoară conform metodologiilor stabilite.

În cazul în care circuitul analizat are mai multe ieșiri în diferite nivele logice, pentru fiecare ieșire se stabilește schema echi-

valență. *Observație.* Dacă un modul *NICI* sau *NUMAI* comandă module atât din nivele pare cât și din cele impare se consideră că acesta comandă modulul din nivelul imediat superior ; modulul din nivelul logic mai îndepărtat îl va comanda un alt modul fictiv, echivalent cu cel inițial dar cu structura nivelului imediat inferior.

9.2. Sinteza circuitelor combinaționale

Problemele de sinteză ale circuitelor combinaționale se definesc în morderul următor : *cunoscînd semnalele de ieșire corespunzătoare diferitelor combinații ale semnelor de intrare se cere să se stabilească structura circuitului.* Din expunerea conținutului problemelor de sinteză rezultă că acestea comportă două etape :

I) *sinteza abstractă*, care constă în stabilirea expresiilor funcțiilor booleene ce corespund condițiilor impuse între semnalele de ieșire și intrare. Pentru ca schemele să fie cât mai simple și deci mai economice, se caută expresiile minime ale funcțiilor. Deci, din punct de vedere matematic, problemele de sinteză conduc la necesitatea minimizării *FB* în sistemul de funcții elementare ales din considerente practice ;

II) *sinteza structurală*, care constă în determinarea structurii fizice a circuitului sintetizat. Sinteza structurală se face în funcție de tipul circuitelor logice elementare (module) și de numărul de intrări ale acestora, de semnalele disponibile în diferite puncte ale sistemului etc.

Din cele stabilite mai sus se poate defini algoritmul sintezei circuitelor combinaționale :

1. Din condițiile problemei se stabilesc corespundențele între combinațiile semnalelor de intrare și ieșire folosind tabelul de adevăr, diagrama Karnaugh etc.

2. Se realizează minimizarea funcțiilor care rezultă din etapa precedentă.

3. Se implementează, dacă este cazul, cu funcțiile elementare impuse de realizarea practică.

4. Se stabilește logigrama, plecînd de la forma minimă obținută pentru funcțiile de ieșire în pasul al treilea.

5. Se analizează circuitul obținut, pentru a vedea dacă corespunde condițiilor impuse inițial (facultativ).

9.2.1. Sinteza circuitelor combinaționale cu module SI, SAU, NU

În cazul sintezei circuitelor combinaționale cu circuite logice elementare *SI*, *SAU*, *NU* este necesar ca expresiile minime să se obțină sub formele disjunctivă sau conjunctivă. În etapa de sinteză structurală mai intervine condiția de a realiza circuitul sintetizat cu module standard avînd un anumit număr de intrări. Dacă, corespunzător expresiilor minime, sînt necesare mai multe intrări decît are un modul, apare problema *expandării intrărilor*. Această operație se realizează în conformitate cu proprietatea de asociativitate a operațiilor *SI* și *SAU*, formînd în expresiile minimale asocieri cuprinzînd un număr de argumente sau expresii egal cu numărul de intrări ale circuitelor standard folosite.

Exemplu. Să se sintetizeze circuitul combinațional pentru care corespondențele între intrări și ieșiri sînt prezentate în tabelul 9.4. Circuitul se va realiza cu circuite logice *NU* și module *SAU* și *SI* cu două intrări.

Tab. 9.4

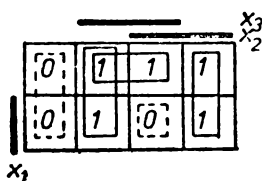


Fig. 9.9

x_1	x_2	x_3	y	x_1	x_2	x_3	y
0	0	0	0	1	0	0	0
0	0	1	1	1	0	1	1
0	1	0	1	1	1	0	1
0	1	1	1	1	1	1	0

Din tabelul de adevăr 9.4 se poate stabili expresia analitică canonică a funcției de ieșire. Mai avantajos este însă a folosi una din metodele globale de minimizare care permit obținerea directă a expresiei minimale. Astfel, în fig. 9.9 este prezentată diagrama Karnough corespunzătoare. Reunind cîmpurile cu unități rezultă :

$$y_D = \bar{x}_2 x_3 \cup \bar{x}_1 x_3 \cup x_2 \bar{x}_3 = (\bar{x}_1 \cup \bar{x}_2) x_3 \cup x_2 \bar{x}_3.$$

Reunind cîmpurile adiacente cu zerouri ale funcției se obține :

$$y_C = (x_2 \cup x_3)(\bar{x}_1 \cup \bar{x}_2 \cup \bar{x}_3).$$

Comparînd cele două expresii, conform definiției formelor minimale, avînd același număr de litere ar trebui ca și schemele structurale corespunzătoare să fie identice în ceea ce privește consumul

de materiale. Dar, fiind impuse module SI și SAU cu două intrări, pentru forma conjunctivă minimă este necesară expandarea termenului cu trei argumente :

$$y_c = (x_2 \cup x_3)[(\bar{x}_1 \cup \bar{x}_2) \cup \bar{x}_3],$$

care acum poate fi materializată numai cu module cu două intrări. În fig. 9.10a este prezentată logigrama corespunzătoare formei disjunctive minime (y_D) iar în fig. 9.10b logigrama corespunzătoare formei conjunctive minime (y_C). Se observă că ambele scheme conțin același număr de module.

Compararea a două scheme logice echivalente după criteriul material se poate face luînd în considerație numărul de module și numărul de intrări ale modulelor. În acest mod se ține cont de consumul de diode și tranzistoare, neglijîndu-se consumul de

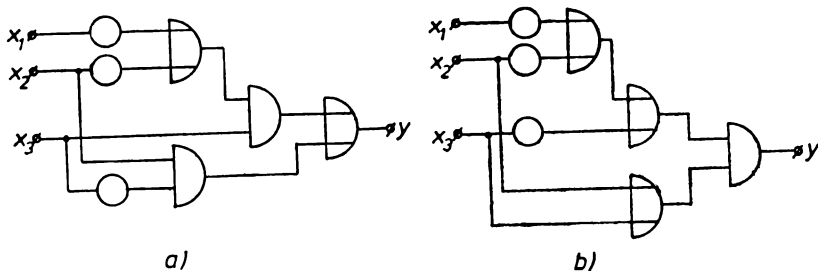


Fig. 9.10

rezistoare. Se poate defini în acest mod un *raport de structură* pentru logigrama considerată, ca raportul dintre numărul de module și numărul de intrări. Pentru logigramele din fig. 9.10

acest raport este : $\frac{7 \text{ module}}{11 \text{ intr.}}$.

Sinteza circuitelor combinaționale cu mai multe ieșiri

În acest caz sinteza se poate realiza ca și pentru circuitele cu o singură ieșire, minimizînd fiecare funcție booleană independent. Însă, la o astfel de abordare a problemei, în general, nu se obține soluția optimă. Din punct de vedere economic problema sintezei optime a circuitelor combinaționale cu mai multe ieșiri conduce la existența unui număr cît mai mare de module care să fie folosite în comun. De asemenea, este de dorit ca aceste

module comune să fie cît mai puține. Din punct de vedere teoretic această problemă conduce la minimizarea simultană a unui sistem de FB , denumită și *minimizare globală*.

Se consideră un sistem de FB de n argumente date prin FCD și care nu sînt disjuncte (aplică aceiași mintermeni în 1). Pentru acest set de funcții se pot stabili următoarele submulțimi de mintermeni:

— mintermeni comuni tuturor funcțiilor, denumit și nucleul sistemului;

— mintermeni comuni unor submulțimi a sistemului de funcții;

— mintermeni specifici fiecărei funcții din sistem.

Minimizînd fiecare din sub mulțimile de mintermeni stabilite se obțin implicații comune tuturor funcțiilor, comune numai unor funcții și implicații specifici fiecărei funcții.

Considerațiile de mai sus pot fi aplicate simplu la minimizarea prin diagrame Karnaugh sau prin simbol de marcare. În primul caz, în diagramele Karnaugh corespunzătoare diferitelor funcții sau într-o diagramă comună, se reunesc acele cîmpuri adiacente care sînt comune mai multor funcții. La folosirea metodei simbolice se realizează minimizarea submulțimilor de numere de stare comune diferitelor funcții.

Exemplu. Să se sintetizeze cu module SI , SAU , NU circuitul combinațional cu patru intrări și trei ieșiri pentru care condițiile de funcționare sînt date prin următorul simbol de marcare:

$$F(A, B, C) = D_{\substack{x_1 x_2 x_3 x_4 \\ (6, 7, 10, 11, 12, 13, 14, 15, 16, 17) \\ (1, 3, 6, 7, 10, 11, 12, 13) \\ (1, 3, 6, 7, 10, 14, 15, 16, 17)}}$$

Se consideră că se dispune de module logice cu numărul de intrări și ieșiri necesar.

Minimizînd fiecare funcție de ieșire separat, se obține:

— pentru funcția A

$$-6 \overset{1}{/} 7 \overset{10}{/} 16 \ 17 \quad \rightarrow D_{\underset{0}{0} \overset{1}{1} \overset{0}{0}}^{x_1 x_2 x_3 x_4} = x_2 x_3.$$

$$-10 \overset{1}{/} 11 \overset{2}{/} 12 \ 13 \overset{4}{/} 14 \ 15.. \quad \rightarrow D_{\underset{0}{0} \overset{0}{0} \overset{0}{0}}^{x_1 x_2 x_3 x_4} = x_1.$$

— pentru funcția B

$$-1 \overset{2}{/} 3 \overset{10}{/} 11 \ 13 \quad \rightarrow D_{\underset{0}{0} \overset{0}{0} \overset{1}{0}}^{x_1 x_2 x_3 x_4} = \bar{x}_2 x_4.$$

$$-6 \text{ / } \overset{1}{7} \quad \rightarrow D_{0\underline{1}1\underline{0}}^{x_1 x_2 x_3 x_4} = \bar{x}_1 x_2 x_3.$$

$$-10 \text{ / } \overset{2}{12} \text{ / } \overset{1}{11} \text{ / } 13 \quad \rightarrow D_{1\underline{0}0\underline{0}}^{x_1 x_2 x_3 x_4} = x_1 \bar{x}_2.$$

— pentru funcția C

$$-1 \text{ / } \overset{2}{3} \quad \rightarrow D_{0\underline{0}0\underline{0}}^{x_1 x_2 x_3 x_4} = \bar{x}_1 \bar{x}_2 x_4.$$

$$-6 \text{ / } \overset{1}{7} \text{ / } \overset{10}{16} \text{ / } 17 \quad \rightarrow D_{0\underline{1}1\underline{0}}^{x_1 x_2 x_3 x_4} = x_2 x_3.$$

$$-10 \text{ / } \overset{4}{14} \quad \rightarrow D_{1\underline{0}0\underline{0}}^{x_1 x_2 x_3 x_4} = \bar{x}_2 \bar{x}_3 \bar{x}_4.$$

$$-15 \text{ / } \overset{2}{17} \text{ / } \overset{1}{14} \text{ / } 16 \quad \rightarrow D_{1\underline{1}0\underline{0}}^{x_1 x_2 x_3 x_4} = x_1 x_2.$$

Rezultă expresiile disjunctive minime :

$$A = x_1 \cup x_2 x_3.$$

$$(9.4) \quad B = x_1 \bar{x}_2 \cup \bar{x}_1 x_2 x_3 \cup \bar{x}_2 x_4 = (x_1 \cup x_4) \bar{x}_2 \cup \bar{x}_1 x_2 x_3$$

$$C = x_1 x_2 \cup x_2 x_3 \cup \bar{x}_1 \bar{x}_2 x_4 \cup \bar{x}_2 \bar{x}_3 \bar{x}_4 = x_2 (x_1 \cup x_3) \cup \bar{x}_2 (\bar{x}_1 x_4 \cup \bar{x}_3 \bar{x}_4).$$

Minimizând global sistemul de funcții, rezultă :

— implicanții comuni funcțiilor A, B și C

$$-6 \text{ / } \overset{1}{7} \quad \rightarrow D_{0\underline{1}1\underline{0}}^{x_1 x_2 x_3 x_4} = \bar{x}_1 x_2 x_3.$$

$$-10 \quad \rightarrow D_{1\underline{0}0\underline{0}}^{x_1 x_2 x_3 x_4} = x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4.$$

— implicanții comuni funcțiilor A și B

$$-11 \text{ / } \overset{2}{13} \text{ / } \overset{1}{10} \text{ / } 12 \quad \rightarrow D_{1\underline{0}0\underline{0}}^{x_1 x_2 x_3 x_4} = x_1 \bar{x}_2.$$

— implicanții comuni funcțiilor A și C

$$-14 \text{ / } \overset{1}{15} \text{ / } \overset{2}{16} \text{ / } 17 \quad \rightarrow D_{1\underline{1}0\underline{0}}^{x_1 x_2 x_3 x_4} = x_1 x_2.$$

— implicanții comuni funcțiilor B și C

$$-1 \text{ / } \overset{2}{3} \quad \rightarrow D_{0\underline{0}0\underline{0}}^{x_1 x_2 x_3 x_4} = \bar{x}_1 \bar{x}_2 x_4.$$

Se observă că în acest caz nu este nici un implicant specific unei singure funcții. Expresiile disjunctive minime pentru minimizarea globală, rezultă :

$$(9.5) \quad \begin{aligned} A &= \bar{x}_1 x_2 x_3 \cup x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 \cup x_1 \bar{x}_2 \cup x_1 x_2. \\ B &= \bar{x}_1 x_2 x_3 \cup x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 \cup x_1 \bar{x}_2 \cup \bar{x}_1 \bar{x}_2 x_4. \\ C &= \bar{x}_1 x_2 x_3 \cup x_1 \bar{x}_2 \bar{x}_3 \bar{x}_4 \cup x_1 x_2 \cup \bar{x}_1 \bar{x}_2 x_4. \end{aligned}$$

În fig. 9.11a este prezentată logigrama corespunzătoare formelor (9.4), iar în fig. 9.11b logigrama corespunzătoare formelor (9.5). Stabilind raportul de structură pentru cele două scheme rezultă : pentru logigrama din fig. 9.11a $\frac{17 \text{ module}}{31 \text{ intrări}}$

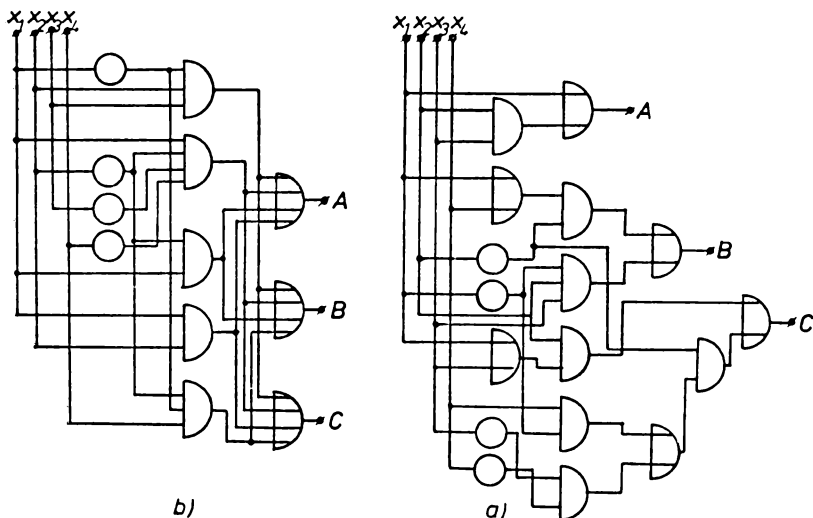


Fig. 9.11

iar pentru logigrama din fig. 9.11b $\frac{12 \text{ module}}{30 \text{ intrări}}$. Este evident

că minimizarea globală asigură un considerabil avantaj material (12 module față de 17) în cazul când funcțiile nu sînt disjuncte.

În cazul când sistemul de funcții proprii ale circuitului sînt disjuncte (nu conțin nici un constituent comun) se realizează minimizarea individuală.

9.2.2. Sinteza circuitelor combinaționale cu module NICI și NUMAI

În acest caz, în etapa de sinteză abstractă este necesară și implementarea formelor minime cu funcții NICI sau NUMAI, așa cum s-a stabilit în cap. 3. De asemenea, în etapa de sinteză structurală este necesar ca și funcțiile de negare să fie materializate tot cu module NICI sau NUMAI, precum și expandarea numărului de intrări.

Implementarea circuitelor de negare cu module NICI sau NUMAI

Deoarece o variabilă negată poate fi exprimată în următoarele forme :

$$(9.6) \quad \begin{aligned} \bar{x} &= \bar{x} \cdot \bar{x} \cdot \dots \cdot \bar{x} = x \downarrow x \downarrow \dots \downarrow x, \\ \bar{x} &= \overline{x \cup 0 \cup \dots \cup 0} = x \downarrow 0 \downarrow \dots \downarrow 0, \end{aligned}$$

rezultă că un circuit de negare poate fi realizat cu module NICI așa cum se arată în figurile 9.12 a sau 9.12b.

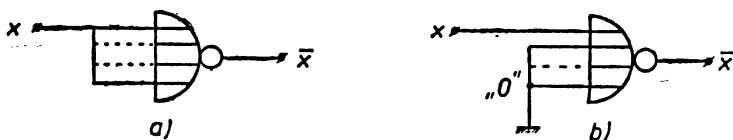


Fig. 9.12

De asemenea, întrucât o variabilă negată mai poate fi exprimată și prin funcții Sheffer :

$$(9.7) \quad \begin{aligned} \bar{x} &= \bar{x} \cup \bar{x} \cup \dots \cup \bar{x} = x \uparrow x \uparrow \dots \uparrow x, \\ x &= \overline{x \cdot 1 \cdot \dots \cdot 1} = x \uparrow 1 \uparrow \dots \uparrow 1, \end{aligned}$$

circuitele de negare pot fi realizate și cu module NUMAI așa cum se arată în fig. 9.13. În cazul circuitelor NICI cu rezistoare

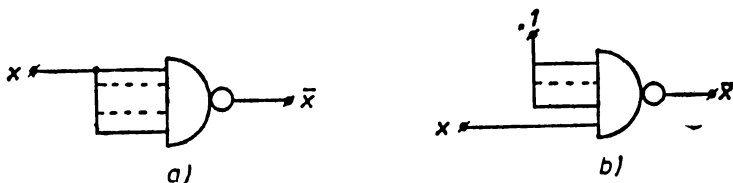


Fig. 9.13

și tranzistoare se preferă conectarea la masă (fig. 9.12b) a intrărilor nefolosite pentru a nu încărca suplimentar elementele de comandă. În cazul circuitelor DTL se preferă conexiunile din fig. 9.12a și respectiv 9.13a, avînd în vedere să nu se depășească „fan out“-ul circuitului de comandă.

Expandarea numărului de intrări a modulelor NICI și NUMAI

Se consideră următoarea funcție :

$$y = x_1^{a_1} \downarrow x_2^{a_2} \downarrow \dots \downarrow x_k^{a_k}, \quad a_j \rightarrow \{0, 1\}, \quad j=1, 2, \dots, k.$$

Se presupune că se dispune de circuite NICI cu „i“ intrări ($i < k$), cu care trebuie implementată funcția dată. Avînd în vedere definiția funcției NICI și (9.6) expresia funcției considerate se mai poate scrie :

$$\begin{aligned} y &= \overline{x_1^{a_1}} \cdot \overline{x_2^{a_2}} \cdot \dots \cdot \overline{x_i^{a_i}} \cdot \overline{x_j^{a_j}} \cdot \overline{x_k^{a_k}} = (\overline{x_1^{a_1}} \cdot \overline{x_2^{a_2}} \cdot \dots \cdot \overline{x_i^{a_i}}) (\overline{x_j^{a_j}} \cdot \overline{x_k^{a_k}} \cdot \overline{0} \dots \overline{0}) = \\ &= \underbrace{(x_1^{a_1} \downarrow x_2^{a_2} \downarrow \dots \downarrow x_i^{a_i})}_{\text{„i“ termeni}} \underbrace{(x_j^{a_j} \downarrow x_k^{a_k} \downarrow 0 \downarrow \dots \downarrow 0)}_{\text{„i“ termeni}}. \end{aligned}$$

Negînd de două ori fiecare paranteză din ultima expresie rezultă :

$$(9.8) \quad y = \overline{x_1^{a_1} \downarrow x_2^{a_2} \downarrow \dots \downarrow x_i^{a_i} \downarrow x_j^{a_j} \downarrow x_k^{a_k} \downarrow 0 \downarrow \dots \downarrow 0}.$$

Într-un mod analog se poate demonstra și identitatea următoare :

$$(9.9) \quad y = \overline{x_1^{a_1} \uparrow x_2^{a_2} \uparrow \dots \uparrow x_i^{a_i} \uparrow x_j^{a_j} \uparrow x_k^{a_k}} = \overline{x_1^{a_1} \uparrow x_2^{a_2} \uparrow \dots \uparrow x_i^{a_i} \uparrow x_j^{a_j} \uparrow x_k^{a_k} \uparrow \dots \uparrow 1}.$$

Exemplu. Să se implementeze următoarele funcții :

$$y_1 = x_1 \downarrow \overline{x_2} \downarrow x_3 \downarrow x_4 \downarrow \overline{x_5} \quad \text{și} \quad y_2 = \overline{x_1} \uparrow \overline{x_2} \uparrow x_3 \uparrow \overline{x_4} \uparrow x_5 \uparrow \overline{x_6} \uparrow x_7$$

cu module NICI și respectiv NUMAI cu trei intrări.

Conform cu (9.8) și (9.9), cele două funcții se pot scrie :

$$y_1 = \overline{x_1 \downarrow \overline{x_2} \downarrow x_3 \downarrow x_4 \downarrow \overline{x_5} \downarrow 0}.$$

$$y_2 = \overline{\overline{x_1} \uparrow \overline{x_2} \uparrow x_3 \uparrow \overline{x_4} \uparrow x_5 \uparrow \overline{x_6} \uparrow x_7 \uparrow 1 \uparrow 1} = \overline{\overline{x_1} \uparrow \overline{x_2} \uparrow x_3 \uparrow \overline{x_4} \uparrow x_5 \uparrow \overline{x_6} \uparrow x_7}.$$

Logigramele corespunzătoare celor două funcții sînt prezentate în fig. 9.14.

Din cele prezentate pînă acum rezultă dezavantajul materia 1 al expandării circuitelor pentru un număr mare de intrări și deci necesitatea realizării unor module logice tipizate cu un număr diferit de intrări. Astfel, apare justificată realizarea circuitelor

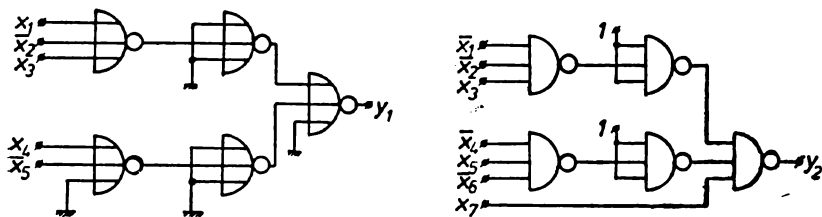


Fig. 9.14

integrate TTL — NUMAI cu 2, 3, 4 și 8 intrări, inclusiv inversorul logic.

Avînd în vedere cele stabilite pînă acum, sinteza circuitelor combinaționale cu module NICI sau NUMAI decurge după același algoritm definit la începutul capitolului. Astfel, după efectuarea minimizării se realizează implementarea funcțiilor obținute cu module NICI sau NUMAI și eventual expandarea lor.

Exemplu. Să se realizeze sinteza cu module NUMAI a circuitului combinațional din exemplul de minimizare prin metoda globală cu formele minime disjunctive (9.5). Se dispune de circuite integrate de tip CDB 404 E și CDB 410 E.

Expresiile disjunctive (9.5) se pot implementa imediat cu funcții NUMAI :

$$\begin{aligned}
 A &= (\bar{x}_1 \uparrow x_2 \uparrow x_3) \uparrow (x_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3 \uparrow \bar{x}_4) \uparrow (x_1 \uparrow \bar{x}_2) \uparrow (x_1 \uparrow x_2) \\
 B &= (\bar{x}_1 \uparrow x_2 \uparrow x_3) \uparrow (x_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3 \uparrow \bar{x}_4) \uparrow (x_1 \uparrow \bar{x}_2) \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow x_4) \\
 C &= (\bar{x}_1 \uparrow x_2 \uparrow x_3) \uparrow (x_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3 \uparrow \bar{x}_4) \uparrow (x_1 \uparrow x_2) \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow x_4).
 \end{aligned}$$

Ținînd cont că se dispune de module NUMAI cu 3 intrări (CDB 410 E) și avînd în vedere (9.9), setul de funcții devine :

$$\begin{aligned}
 A &= \overline{(\bar{x}_1 \uparrow x_2 \uparrow x_3) \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3 \uparrow \bar{x}_4) \uparrow (x_1 \uparrow \bar{x}_2 \uparrow 1) \uparrow (x_1 \uparrow x_2 \uparrow 1)} \\
 B &= \overline{(\bar{x}_1 \uparrow x_2 \uparrow x_3) \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3 \uparrow \bar{x}_4) \uparrow (x_1 \uparrow \bar{x}_2 \uparrow 1) \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow x_4)} \\
 C &= \overline{(\bar{x}_1 \uparrow x_2 \uparrow x_3) \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3 \uparrow \bar{x}_4) \uparrow (x_1 \uparrow x_2 \uparrow 1) \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow x_4)}.
 \end{aligned}$$

În fig. 9.15 este prezentată logigrama circuitului sintetizat, corespunzător expresiilor obținute mai sus. Pentru negare s-au folosit circuitele din capsula CDB 404 E. După cum rezultă din logigramă sînt necesare 11 module NUMAI cu 3 intrări și 7 circuite inversoare. Avînd în vedere numărul de circuite logice

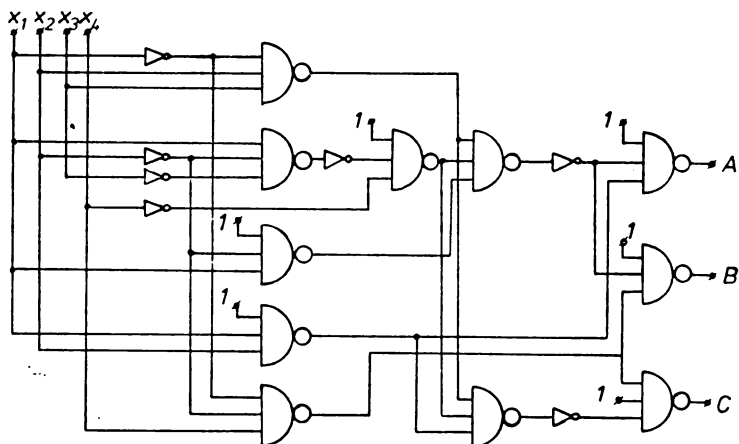


Fig. 9.15

conținute de fiecare capsulă folosită (CDB 404 E — 6 inversoare iar CDB 410 E — 3 circuite NUMAI cu trei intrări) rezultă un necesar de 4 capsule CDB 410 E și 2 capsule CDB 404 E.

În categoria circuitelor combinaționale intră multe circuite cu funcțiuni tipizate : codificatoare, decodificatoare, convertoare de cod, sumatoare și scăzătoare, multiplexoare etc., utilizate și în comenzile secvențiale. În prezent astfel de circuite sînt și integrate pe scară medie sau largă [35,52/.

9.3. Hazardul în circuitele combinaționale

În studiul circuitelor combinaționale pînă acum s-a considerat că circuitele logice cu care se materializează schemele sînt identice și comutarea lor este ideală. În realitate, datorită timpului finit de comutare fiecare modul logic introduce întîrzieri inerente. Din acest motiv apar fenomene calitativ noi care modifică comportarea reală a circuitelor de comandă, cu manifesta-

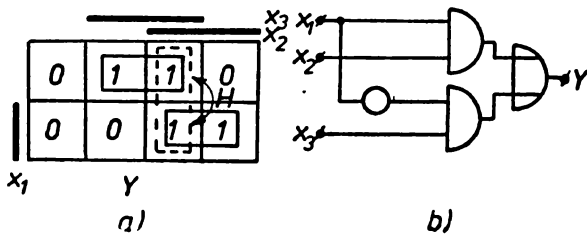


Fig. 9.16

rea hazardului. Pentru exemplificare se consideră sinteza circuitului cu corespondențele din tabelul 9.5. Diagrama Karnaugh corespunzătoare este prezentată în fig. 9.16a. Rezultă imediat ex-

Tab 9.5

x_1	x_2	x_3	y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	1

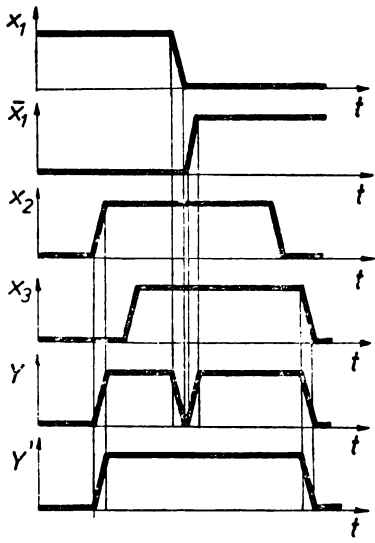


Fig. 9.17

presia disjunctivă minimă :

$$y = \bar{x}_1 x_3 \cup x_1 x_2,$$

căreia îi corespunde circuitul din fig. 9.16b. Se analizează în continuare funcționarea circuitului sintetizat pentru combinația semnalelor din fig. 9.17 prezentată prin diagrame în timp. S-a considerat funcționarea reală a circuitelor logice, cu întârzierile produse de timpul finit de comutare. Se observă că deși pentru combinațiile considerate ale semnalelor la intrare semnalul

la ieșire trebuie să se păstreze 1 logic, datorită funcționării neideale a elementelor schemei apare un impuls 0 fals. Acest semnal se poate propaga în schema de comandă și poate duce la o funcționare necorespunzătoare a acesteia. Din cele prezentate mai sus rezultă că astfel de fenomene pot apărea datorită decalajului în timp între semnalele care trebuie să se aplice simultan la intrarea unui element logic. Aceste fenomene se manifestă diferit, în funcție de semnalul care se aplică primul. Din acest motiv se numesc *fenomene de concurs* sau specific circuitelor combinaționale, *hazard static*.

Evidențierea hazardului static în circuitele combinaționale se poate face chiar în etapa de sinteză. Pentru exemplul considerat, din diagramele din fig. 9.17 rezultă că fenomenul de concurs este generat de negarea semnalului x_1 . Acest fapt în diagrama Karnaugh este echivalent cu tranziția de la câmpul care corespunde combinației 011 la cel care corespunde combinației 111 (fig. 9.16a). Deși aceste câmpuri conțin unități ale funcției minimize și sînt adiacente, ele nu au fost reunite deoarece nu formează un implicant prim esențial.

Din cele prezentate rezultă că hazardul static poate fi evidențiat în modul următor :

1) Hazardul static într-un circuit combinațional poate apărea dacă în diagramele Karnaugh există câmpuri adiacente necuprinse în reuniri comune.

2) Variabila care generează hazardul este cea care își schimbă valoarea la trecerea între câmpuri adiacente nereunite.

În fig. 9.16a s-a notat prin H posibilitatea de apariție a hazardului static generat de schimbarea valorii lui x_1 .

După cum s-a arătat în /14/ evidențierea hazardului static se poate face și în cazul cînd se folosește metoda simbolică și anume :

1) Hazardul static într-un circuit combinațional sintetizat prin metoda simbolică poate apărea dacă există numere de stare adiacente necuprinse în liste de adiacențe comune.

2) Variabila care corespunde diferenței de adiacență între numerele de stare de la punctul 1) este cea care generează hazardul.

Pentru exemplul considerat se stabilesc imediat simbolul de marcare și listele de adiacențe :

$$Y = D_1^{x_1 x_2} : (1, 3, 6, 7)$$

$$\left. \begin{array}{l} -1 \begin{array}{l} \nearrow 3 \\ \searrow 1 \end{array} \\ -6 \begin{array}{l} \nearrow 7 \\ \searrow 2 \end{array} \end{array} \right\} H \rightarrow 3 \begin{array}{l} \nearrow 4 \\ \searrow 7 \end{array} \rightarrow D_{011}^{\bar{x}_1 x_2 x_3} = x_2 x_3.$$

Din lista de adiacențe rezultă că poate să apară hazard datorită numerelor de stare 3 și 7 necuprinse în aceeași listă. Variabila care generează hazardul este cea care corespunde diferenței de adiacență între 3 și 7, adică x_1 .

Eliminarea fenomenelor de hazard din schemele combinaționale se poate face pe două căi, și anume :

— prin introducerea unor elemente de întârziere pasive la ieșirea circuitelor respective care să filtreze semnalele de impulsurile datorate comutărilor false. În acest mod se împiedică transferul informației false. Deși această metodă este simplă poate afecta viteza de acționare a întregului circuit ;

— prin mărirea *redondanței* (redundanței) circuitului, folosind module logice suplimentare care să nu permită apariția fenomenelor de hazard. În acest caz se folosesc semnalele existente în schemă pentru a asigura o funcționare corectă și în timpul comutărilor. Dezavantajul acestei soluții este complicarea structurală a circuitului (crește redondanța) în schimb viteza de acționare practic nu este afectată.

Dacă pentru prima posibilitate modul de eliminare a hazardului static este evident, în continuare se va arăta modul cum se elimină aceste fenomene prin creșterea redondanței circuitului. Pentru aceasta se consideră exemplul studiat. Din diagrama în timp se observă că la ieșire semnalul poate deveni sub forma Y' dacă în expresia logică apare un nou termen determinat de existența simultană a semnalelor x_2 și x_3 (fig. 9.17). Termenul suplimentar $x_2 x_3$ nu modifică funcționarea circuitului, dar permite eliminarea hazardului. În adevăr, se poate verifica ușor că

$$Y = \bar{x}_1 x_3 \cup x_1 x_2 \quad \text{și} \quad Y' = \bar{x}_1 x_3 \cup x_1 x_2 \cup x_2 x_3$$

sînt echivalente. Termenul $x_2 x_3$ implică un modul SI suplimentar în logigrama din fig. 9.16b și un modul SAU cu trei intrări. În diagrama Karnaugh cele arătate mai sus sînt echivalente cu reunirea cîmpurilor adiacente care au determinat apariția hazardului (figurat cu linie întreruptă în fig. 9.16a). În metoda simbolică termenul suplimentar corespunde noii liste de adiacențe alcătuită cu numerele de stare care generează hazardul (3 și 7 în cazul exemplului considerat).

Din cele arătate mai sus rezultă că atunci cînd este necesar să se elimine hazardul static prin creșterea redondanței circuitului, termenii suplimentari se determină astfel :

1) Dacă sinteza circuitului s-a făcut cu ajutorul diagramelor Karnaugh, termenii redondanți necesari eliminării hazardului sînt cei care se obțin prin reunirea cîmpurilor adiacente care au generat hazardul.

2) Dacă sinteza s-a realizat prin metoda simbolică, termenii redondanți necesari pentru eliminarea hazardului se obțin prin formarea unor noi liste de adiacențe cu numerele de stare care au generat hazardul.

În fine, trebuie menționat faptul că chiar atunci cînd s-a evidențiat posibilitatea de apariție a hazardului static, să nu fie necesare măsuri de eliminare dacă în funcționarea circuitului nu apar combinațiile care generează acest fenomen.

10. Circuite logice secvențiale de comandă

Întrucît materializarea automatelor finite cu memorie o constituie circuitele logice secvențiale, acestea sînt în fapt chiar comenzile secvențiale. Existența memoriei la aceste automate face ca evoluția lor în timp să fie bine definită prin stările succesive în care se pot afla. Ca urmare, semnalele la ieșirea unui circuit secvențial vor fi funcții atît de semnalele aplicate la intrare la un anumit moment, precum și de semnalele aplicate la momente de timp anterioare. Astfel, spre deosebire de circuitele combinaționale, existența stărilor interne face ca timpul să apară explicit în funcționarea circuitelor secvențiale de comandă.

Avînd în vedere structura generală a unui automat finit cu memorie (fig. 9.1), rezultă că un circuit secvențial poate fi reprezentat printr-o parte combinațională care să furnizeze funcțiile de tranziție $f(s, x)$ și de ieșire $g(s, x)$ și o memorie care să definească stările circuitului. Particularizînd pentru cazul binar, atît variabilele cît și funcțiile caracteristice ale unui circuit secvențial vor lua valori în mulțimea $\{0, 1\}$. De asemenea, mulțimea stărilor poate fi explicitată prin atașarea unei mulțimi de variabile, operație denumită a s i g n a r e d e s t a r e. Dacă un circuit secvențial cu mulțimea de stări S are un număr $|S| = N_s$ stări, atunci vor fi necesare

$$(10.1) \quad 2^{n_k} \geq N_s \rightarrow n_k \geq \log_2 N_s$$

variabile binare de stare.

În condițiile menționate se poate defini structura și mărimile specifice ale unui circuit secvențial de comandă. În figura 10.1 este prezentată sub formă de schemă bloc structura unui astfel de circuit. Sînt evidențiate partea combinațională care asigură funcțiile de tranziție și de ieșire (C) și partea de memorie (M) care definește stările sistemului. Referitor la această structură a unui circuit secvențial se definesc :

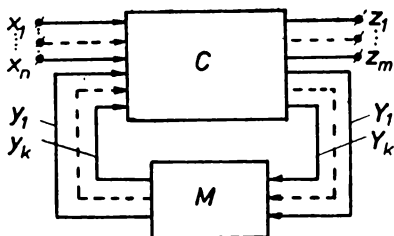


Fig. 10.1

(10.2) (x_1, x_2, \dots, x_n) — semnalele de intrare sau primare,
 (z_1, z_2, \dots, z_m) — semnalele de ieșire,
 (Y_1, Y_2, \dots, Y_k) — semnalele secundare de excitație a memoriei,
 (y_1, y_2, \dots, y_k) — semnalele secundare de stare.

În accepția teoriei automatelor finite mulțimea semnalelor de intrare constituie alfabetul de intrare al automatului, iar mulțimea semnalelor de ieșire alfabetul de ieșire. Stările automatului la momentul v de timp sînt definite prin mulțimea semnalelor secundare de stare, iar la momentul $(v+1)$ de semnalele secundare de excitație a memoriei.

Avînd în vedere cele spuse mai sus, evoluția unui circuit secvențial poate fi descrisă prin relații între semnalele de intrare, starea prezentă, starea următoare și semnalele de ieșire. Astfel, pentru circuitul din fig. 10.1 se pot scrie următoarele sisteme de funcții booleene :

$$(10.2) \quad \begin{aligned} z_1 &= g_1(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_k) \\ z_2 &= g_2(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_k) \\ &\dots \\ z_m &= g_m(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_k), \end{aligned}$$

pentru ieșiri și

$$(10.3) \quad \begin{aligned} Y_1 &= f_1(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_k) \\ Y_2 &= f_2(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_k) \\ &\dots \\ Y_k &= f_k(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_k), \end{aligned}$$

pentru stările următoare ale automatului.

Seturile de funcții (10.2) și (10.3) constituie explicitarea funcțiilor de ieșire și de tranziție pentru un circuit logic secvențial. S-a considerat o comportare deterministă a sistemului, adică pentru anumite valori ale variabilelor de intrare ($\tilde{x}_1, \tilde{x}_2, \dots, \tilde{x}_n$) la momentul v de timp și anumite stări ($\tilde{y}_1, \tilde{y}_2, \dots, \tilde{y}_k$) este posibilă tranziția într-o singură stare ($\tilde{Y}_1, \tilde{Y}_2, \dots, \tilde{Y}_k$). Comparând cele prezentate mai sus cu considerațiile din teoria automatelor finite deterministe rezultă că cele două seturi de funcții definesc un circuit secvențial de tip MEALY. Dacă funcțiile de ieșire depind numai de semnalele secundare de stare :

$$\begin{aligned}
 10.4) \quad z_1 &= g'_1(y_1, y_2, \dots, y_k) \\
 z_2 &= g'_2(y_1, y_2, \dots, y_k) \\
 &\dots \dots \dots \dots \dots \dots \dots \dots \dots \dots \\
 z_m &= g'_m(y_1, y_2, \dots, y_k),
 \end{aligned}$$

împreună cu sistemul de funcții (10.3) se definește un circuit secvențial de tip MOORE.

Tipuri de circuite secvențiale

Un criteriu de clasificare a circuitelor secvențiale îl constituie *modul de funcționare* al acestor circuite. Din acest punct de vedere se poate realiza o divizare a circuitelor secvențiale în două mari clase :

- circuite secvențiale asincrone,
- circuite secvențiale sincrone.

În *circuitele secvențiale asincrone* procesele de comutare au loc la momente arbitrare de timp. Elementele de memorie trebuie să rețină valorile semnalelor aplicate lor un timp relativ scurt, corespunzător intervalului dintre comutările succesive ale elementelor combinaționale. Obișnuit, ca elemente de memorie se folosesc circuite de întârziere (Delay unit). În fig. 10.2a este prezentată schema bloc a unui circuit asincron cu elemente de întârziere (D_1, D_2, \dots, D_k) pe circuitele de reacție. Elementele de întârziere nu modifică prelucrarea logică ci numai introduc întârzieri în transmiterea informației pe circuitele de reacție formate de memorie. Notînd cu $\Delta t_1, \Delta t_2, \dots, \Delta t_k$ întârzierile introduse de elementele menționate, pentru elementul „i” de memorie se poate scrie :

$$(10.5) \quad Y_i(t_v) = y_i(t_v + \Delta t_i), \quad v = 1, 2, \dots, i = 1, 2, \dots, k.$$

Un caz particular mult întâlnit în comanda secvențială industrială (comanda acționărilor electrice) este acela cînd se dorește ca întârzierile pe reacții să fie minime. În acest caz se poate scrie :

$$(10.6) \quad Y_i(t_v) = y_i(t_v), \quad v=1, 2, \dots, i=1, 2, \dots, k.$$

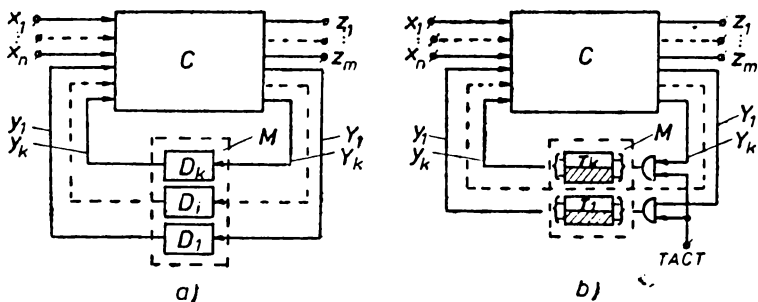


Fig. 10.2

Ca elemente de memorie se pot folosi și circuite basculante bistabile.

În *circuitele secvențiale sincrone* procesele de comutare au loc la momente de timp bine determinate denumite *tacte*, marcate prin impulsuri furnizate de un generator de *sincronizare* sau de *tact*. În cazul acestor circuite toate elementele de memorie trebuie să rețină valorile semnalelor aplicate un interval relativ lung de timp, egal cu cel puțin intervalul dintre două tacte. Din acest motiv se folosesc cu predilecție ca elemente de memorie circuite basculante bistabile (triggere) sincronizabile. În fig. 10.2b este prezentată schema bloc a unui circuit secvențial sincron avînd triggere drept celule de memorie. Semnalele de excitație se aplică sincron cu semnalele de la generatorul de tact.

Circuitele asincrone sînt mai simple și se preferă în comenzile de complexitate mai redusă. Dezavantajul acestor comenzi secvențiale constă în faptul că pot apare fenomene de concurs între semnalele secundare. Circuitele sincrone, implicînd circuite și generatoare de sincronizare, au un cost mai ridicat dar elimină complet fenomenele de concurs.

În funcție de *natura elementelor folosite pe buclele de reacție* se poate realiza o altă divizare a circuitelor secvențiale, și anume :

— *circuite secvențiale cu reacții directe*, în care funcția de memorie este îndeplinită de către partea combinațională prin întârzierile introduse;

— *circuite cu reacții prin celule de întârziere*, care joacă rolul de memorie temporară ;

— *circuite cu reacții prin celule de memorie binară propriuzisă* (triggere) cu rol de memorie permanentă.

Circuitele secvențiale cu reacții directe sînt totdeauna asincrone, cele cu memorie temporară (circuite de tip monostabil) sînt asincrone, iar cele cu memorie permanentă atît asincrone cît și sincrone.

Stări ale circuitelor secvențiale

Spre deosebire de variabilele de intrare și ieșire ale unui circuit secvențial, care pot fi evidențiate și măsurate de un observator extern, variabilele secundare pot fi măsurate și observate numai în anumite condiții. Dificultatea este determinată de faptul că aceste variabile secundare reprezintă rezultatul efectului combinat dintre variabilele primare și stările interne ale automatului. Or, determinarea precisă a mulțimii de stări este problema cea mai dificilă pentru care nu există reguli generale. Stabilirea mulțimii de stări se face prin aproximații succesive, apelînd la experimente (teoretice) asupra automatului considerat. În general experimentele necesită aplicarea unor succesiuni de combinații (secvențe) ale variabilelor de intrare și observarea semnalelor la ieșiri. Însă, în majoritatea cazurilor de interes practic există suficiente informații despre structura internă a sistemului pentru a defini variabilele ce caracterizează evoluția internă a acestuia. În continuare se vor defini unele stări posibile ale unui circuit secvențial, definiții care vor permite studiul teoretic și experimental al acestora.

Se pot defini următoarele stări ale circuitelor secvențiale, stări care pot fi evidențiate de un observator extern :

— *stare de start*, este starea pe care o acceptă sistemul în momentul punerii în funcțiune ;

— *stare inițială* — starea în care se aduce automatul pentru aplicarea unei secvențe de intrare. Dacă starea inițială nu este specificată, aceasta se consideră a fi starea de start ;

— *stare finală* — starea în care ajunge automatul în urma aplicării unei secvențe la intrare.

În studiul circuitelor secvențiale cunoașterea stării inițiale este foarte importantă în determinarea secvenței de ieșire și rezultă din : *Teorema stării inițiale*. Fie un automat netrivial cu funcțiile de tranziție $f(s, x)$ și de ieșire $g(s, x)$. Ieșirea este unic

determinată pentru o anumită intrare dacă și numai dacă se cunosc funcțiile caracteristice $f(s, x)$ și $g(s, x)$ și starea inițială.

Funcțiile f și g sînt analoge ecuațiilor de echilibru care definesc un sistem liniar, iar starea inițială a automatului este analogă energiei inițiale din sistem.

Stări stabile și stări instabile

Dacă pentru anumite valori ale semnalelor de intrare elementele de memorie ale unui circuit secvențial păstrează aceleași valori atît la intrări cît și la ieșiri

$$y_j = Y_j, \quad j = 1, 2, \dots, k,$$

se spune că circuitul se află într-o *stare stabilă*.

Dacă însă pentru anumite valori ale semnalelor de intrare semnalele de ieșire ale unor elemente de memorie diferă de semnalele corespunzătoare de intrare cel puțin pentru un element

$$y_j \neq Y_j, \quad j \in \{1, 2, \dots, k\},$$

se spune că circuitul se află într-o *stare instabilă*. În acest caz, fără modificarea valorii semnalelor la intrare automatul efectuează o tranziție spontană în altă stare. După o serie de tranziții spontane circuitul poate ajunge fie într-o stare stabilă fie că trece într-un regim ciclic. Dacă într-o stare instabilă se află mai mult decît un singur element de memorie apare fenomenul de concurs al semnalelor secundare. În acest caz semnalele la ieșire pot să depindă sau nu de întîrzierile din partea combinațională. Se spune că fenomenele de concurs sînt *critice* sau respectiv *necritice*. După cum s-a arătat în cazul circuitelor combinaționale, la apariția fenomenelor de concurs se manifestă *hazardul* — astfel de fenomene neputînd fi prevăzute apriori. După cum s-a mai menționat fenomenele de hazard pot apare în special la circuitele secvențiale asincrone; funcționarea sincronizată prin impulsuri de tact nu permite apariția acestor fenomene la circuitele sincrone. În cazul cînd apar astfel de fenomene, acestea pot conduce la o funcționare esențial incorectă a circuitelor nu numai în regim tranzitoriu, ca în circuitelor combinaționale. Din acest motiv fenomenele de concurs trebuie evitate cu grijă chiar de la proiectarea circuitelor asincrone.

Stări echivalente și stări compatibile

Într-un automat finit complet definit unele stări pot fi supra-abundente, în sensul că rolul lor poate fi îndeplinit de alte stări din interiorul sistemului. Evidențierea acestor stări este legată

de realizarea unor circuite secvențiale mai economice și deci este de dorit să se stabilească cînd două sau mai multe stări au roluri identice, adică sînt echivalente.

Definiție. Două stări s_i și s_j sînt *echivalente* dacă și numai dacă orice secvență aplicată la intrarea automatului furnizează o aceeași secvență la ieșire, indiferent dacă starea inițială este s_i sau s_j .

Echivalența stărilor s_i și s_j se notează $s_i = s_j$ și constituind o relație de echivalență se bucură de proprietățile acesteia : reflexivitatea, simetria și tranzitivitatea ($s_i = s_j$ și $s_j = s_k \rightarrow s_i = s_k$). Echivalența a două stări poate fi uneori stabilită ușor. Astfel, dacă pentru orice $x_i \in X$

$$g(s_i, x_i) = g(s_j, x_i)$$

și

$$f(s_i, x_i) = f(s_j, x_i),$$

atunci $s_i = s_j$. Din relațiile precedente rezultă că două stări sînt echivalente dacă pentru oricare din intrări stărilor considerate le corespund aceleași ieșiri, precum și stări următoare (succesori) identice. Aceste stări denumite și *stări redondante* sau *evident echivalente* pot fi ușor stabilite pe reprezentarea tabelară a unui circuit secvențial. În acest tabel al stărilor două stări sînt evident echivalente dacă au același conținut pentru toate valorile semnalelor de intrare. Liniile corespunzătoare se pot contopi într-una, operație denumită *fuzionare* sau *unificare a stărilor*. Se obține un circuit cu aceeași funcționare cu primul, dar cu un număr mai mic de stări (automat echivalent redus).

Dar, în cele mai multe cazuri eliminarea stărilor redondante nu conduce la obținerea numărului minim de stări. *Minimizarea numărului de stări* al unui circuit secvențial se bazează pe proprietatea de tranzitivitate a echivalenței stărilor, corespunzător căreia mulțimea stărilor circuitului se poate divide în *clase de echivalență disjuncte*. Căutarea claselor de echivalență se face pe modelul Mealy (deoarece conține un număr mai mic de stări decît modelul Moore) și are la bază *partiția mulțimii stărilor* introdusă de Hartmanis și Stearns [13, 20, 31]. Faull și Unger au stabilit o metodologie de căutare sistematică a claselor de echivalență folosind *tabelul implicațiilor*. Se va descrie această metodologie prin următorul :

Exemplu. Să se stabilească stările echivalente pentru circuitul secvențial complet definit dat prin tabelul 10.1.

Tab. 10.1

$s_k \backslash x$	0	1
s_1	$s_2/1$	$s_5/0$
s_2	$s_1/0$	$s_4/1$
s_3	$s_4/1$	$s_5/0$
s_4	$s_3/0$	$s_2/1$
s_5	$s_6/1$	$s_2/0$
s_6	$s_2/0$	$s_1/1$

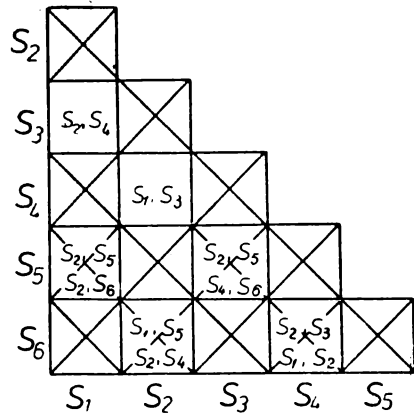


Fig. 10.3

Tabelul implicațiilor (fig. 10.3) se formează din linii și coloane egal cu numărul stărilor circuitului minus unu, astfel ca să poată fi studiată

echivalența tuturor perechilor de stări. În câmpul determinat de intersecția unei linii cu o coloană se înscriu condițiile privind echivalența stărilor respective. Completarea tabelului implicațiilor se realizează sistematic astfel:

— se caută perechile de stări neechivalente (incompatibile) care nu au ieșiri identice; aceste perechi se notează în tabel prin marcarea câmpului corespunzător cu două linii oblice încrucișate, de exemplu: $s_1 \neq s_2$, $s_1 \neq s_4$, ...

— se caută stările evident echivalente, cu același conținut în tabelul stărilor; aceste stări se notează în tabelul implicațiilor prin bifare sau se lasă câmpul respectiv liber. Din studierea tab. 10.1 se observă că nu există astfel de stări;

— se stabilesc condițiile de echivalență pentru stările rămase și care au ieșiri identice pentru toate valorile semnalelor de intrare. În cazul considerat: $s_1 = s_3$ dacă $s_2 = s_4$ — se specifică această condiție în tabel pentru câmpul respectiv, ș.a.

După completarea tabelului implicațiilor se caută *incompatibilitățile suplimentare* generate de implicațiile existente, examinând sistematic cuplurile implicate. Astfel, pentru exemplul considerat echivalența $s_1 = s_3$ implică echivalența $s_2 = s_4$ care la rândul său implică echivalența $s_1 = s_3$. În schimb $s_1 = s_5$ implică $s_2 = s_5$ care nu sînt compatibile, deci $s_1 \neq s_2$ ce se marchează prin barare în tabelul implicațiilor. Noua incompatibilitate afectează și echivalența dubletului (s_2, s_6) care implică echi-

valența stărilor s_1 și s_6 dovedită mai sus ca incompatibilă, ș.a. Se obțin în final perechile de stări echivalente : $s_1 = s_3$ și $s_2 = s_4$.

Problema minimizării numărului de stări impune, în general, nu numai găsirea perechilor de stări echivalente ci și căutarea *echivalențelor maxime*. Pentru aceasta, după ce în prealabil s-a aplicat proprietatea de tranzitivitate perechilor adiacente, se construiește *tabelul echivalențelor maxime*. Pentru exemplul considerat acesta este tab. 10.2 și se construiește formînd rîndu-

Tab. 10.2

Starea	Starea echivalentă	Echivalența maximală
s_5		
s_4		
s_3		
s_2	s_4	(s_2, s_4)
s_1	s_3	$(s_1, s_3)(s_2, s_4)$

Tab. 10.3

x	0	1
s_1	$s_2/1$	$s_5/0$
s_2	$s_1/0$	$s_2/1$
s_5	$s_6/1$	$s_2/0$
s_6	$s_2/0$	$s_1/1$

rilor prin citirea coloanelor tabelului implicațiilor de la dreapta la stînga. În prima coloană a tab. 10.2 se trec stările echivalente iar în a doua coloană se scriu echivalențele maxime. Se obține astfel partiția mulțimii stărilor circuitului în următoarele clase de echivalentă : (s_1, s_3) , (s_2, s_4) , (s_5) , (s_6) .

În tabelul stărilor se păstrează dintre două sau mai multe stări echivalente starea cu numărul cel mai mic, obținîndu-se astfel tabelul redus al stărilor. Pentru exemplul considerat tabelul redus este tab. 10.3.

În studiul automatelor secvențiale incomplet definite stabilirea numărului minim de stări nu se poate face ca și pentru automatele complete folosind noțiunea de echivalență a stărilor. În asemenea cazuri două stări nu mai pot fi complet echivalente întrucît pentru anumite intrări nu se cunosc ieșirile și stările următoare. Din acest motiv pentru automatele incomplete se folosește noțiunea de *compatibilitate a stărilor*, care este mai slabă decît cea de echivalență din cauza lipsei tranzitivității. *Definiție.* Două stări s_i și s_j ale unui automat incomplet definit sînt *compatibile* dacă respectă condiția de echivalență pentru toate cazurile cînd sînt specificate ieșirile și stările următoare. Se notează $s_i \approx s_j$.

O consecință imediată a lipsei tranzitivității este că împărțirea mulțimii stărilor în clase de compatibilitate nu va conduce la partiții ale mulțimii stărilor, ci la submulțimi oarecare nedisjuncte. Din acest motiv, prin aplicarea metodei implicațiilor, se obțin grupe care indică numai *compatibilitățile maxime*. Printre aceste grupe trebuie căutate grupele de stări echivalente cele mai avantajoase și care formează un ansamblu minimal închis/31, 35/. Întrucât în cazurile practice cea mai mare parte a nedefinirilor sînt opționale, metodologia reducerii numărului de stări permite aplicarea considerațiilor de la circuitele complete.

10.1. Analiza circuitelor secvențiale

Problemele de analiză ale circuitelor secvențiale se definesc astfel: *cunoscînd structura circuitului secvențial de comandă, se cere evoluția semnalelor la ieșire pentru anumite evoluții (secvențe) ale semnalelor de intrare*. Pentru ca problemele de analiză să poată fi rezolvate trebuie cunoscută starea inițială a circuitului analizat.

Atît pentru circuitele asincrone cît și pentru cele sincrone analiza se face principial în același mod și comportă următoarele etape principale:

1. Cunoscînd logigrama circuitului, se stabilesc părțile combinațională și de memorie și se definesc semnalele de intrare, ieșire și semnalele secundare.

2. Se stabilesc funcțiile booleene de ieșire din partea combinațională, denumite și funcții de control, de forma (10.2)... (10.4). Funcțiile stabilite se dezvoltă sub formă canonică.

3. Corespunzător combinațiilor valorilor variabilelor secundare de stare se realizează asignarea stărilor și se definesc stările circuitului.

4. Se construiește tabelul stărilor (tranzițiilor) sau graful atașat circuitului (graful tranzițiilor).

5. Pentru evoluția semnalelor de intrare (secvența) și starea inițială impuse, folosind graful sau tabelul tranzițiilor, se stabilește evoluția semnalelor la ieșire.

10.1.1. Analiza circuitelor secvențiale asincrone

Așa cum s-a arătat, analiza circuitelor secvențiale asincrone se face conform algoritmului enunțat mai sus. În plus față de circuitele sincrone, în acest caz este necesar să se stabilească

În cadrul analizei dacă nu se manifestă fenomene de concurs care să genereze hazardul, Modul de desfășurare a analizei se va exemplifica considerînd circuitul secvențial asincron cu logigrama din fig. 10.4. Se cere să se stabilească evoluția semnalelor la ieșire pentru o succesiune de 0 și 1 la intrare, considerînd că starea inițială corespunde situației cînd semnalele secundare au valoarea zero.

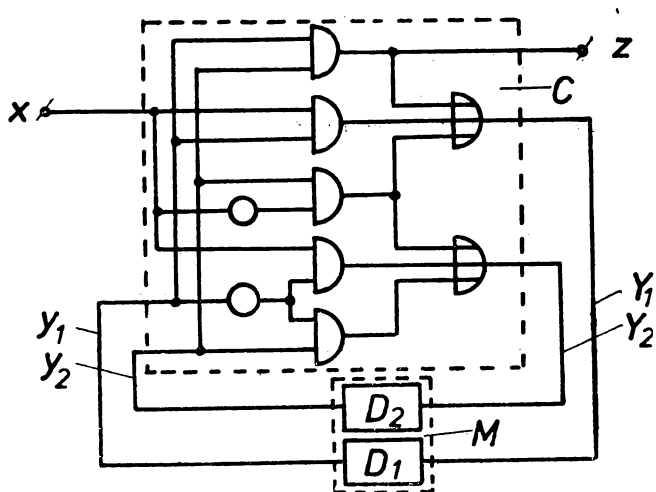


Fig. 10.4

1. Pe logigramă s-au evidențiat structura combinațională (C) și de memorie (M) și semnalele din circuit. Se observă că circuitul are o bornă de intrare (x), una de ieșire (z) și două bucle de reacție cu variabilele secundare de stare y_1 și y_2 .

2. Corespunzător structurii părții combinaționale, la ieșirea acestuia se obțin următoarele funcții booleene :

$$z = y_1 y_2$$

$$Y_1 = y_1 y_2 \cup x y_1 \cup \bar{x} y_2$$

$$Y_2 = x \bar{y}_1 \cup \bar{x} y_2 \cup \bar{y}_1 y_2.$$

Se observă că circuitul este de tip MOORE, funcția de ieșire z fiind de forma (10.4). Prin dezvoltare se obțin formele canonice ale funcțiilor :

$$z = xy_1y_2 \cup \bar{x}\bar{y}_1y_2$$

$$Y_1 = xy_1y_2 \cup \bar{x}y_1y_2 \cup xy_1\bar{y}_2 \cup \bar{x}\bar{y}_1y_2$$

$$Y_2 = x\bar{y}_1y_2 \cup x\bar{y}_1\bar{y}_2 \cup \bar{x}y_1y_2 \cup \bar{x}\bar{y}_1y_2$$

Această etapă poate fi realizată și direct pe diagramele Karnaugh. Astfel, pentru funcțiile y_1 și y_2 se realizează diagrama comună din fig. 10.5. În mod similar se poate stabili și pentru ieșire o diagramă Karnaugh.

3. Circuitul studiat avînd două variabile secundare de stare, cele patru combinații ale valorilor lor definesc stările. În acest caz se poate realiza următoarea asignare :

y_1y_2	s_k
0 0	$\rightarrow s_0$
0 1	$\rightarrow s_1$
1 0	$\rightarrow s_2$
1 1	$\rightarrow s_3$

Conform acestei asignări și condițiilor problemei starea inițială a circuitului studiat este starea s_0 .

4. Avînd în vedere asignarea realizată și diagrama din fig.

10.5 se poate stabili imediat tabelul stărilor (tab. 10.4). Tabelul corespunde unei reprezentări prin tabel a automatelor Moore cu o coloană suplimentară pentru ieșire. În tabel s-au evidențiat stările stabile ; stările stabile sînt identice cu stările inițiale. Corespunzător tabelului stărilor se poate stabili imediat și graful de tranziții, prezentat în fig. 10.6.

Tab. 10.4

	x		z
s_k	0	1	
s_0	s_0	s_1	0
s_1	s_3	s_1	0
s_2	s_0	s_2	0
s_3	s_3	s_2	1

5. Folosind tabelul sau graful de tranziții se poate determina secvența la ieșire pentru secvența impusă la intrare. Considerînd lungimea secvenței la intrare oarecare 0101010... și avînd în vedere că starea inițială este s_0 se obține graful din fig. 10.7a și secvența la ieșire 0010010... Este evidentă ciclicitatea în funcționarea circuitului studiat. Într-un mod similar se poate studia evoluția circuitului pentru orice altă secvență de intrare și orice altă stare inițială. În tab. 10.5 sînt prezentate alte două

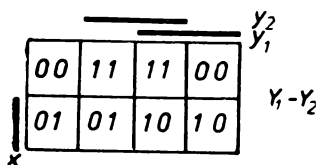


Fig. 10.5

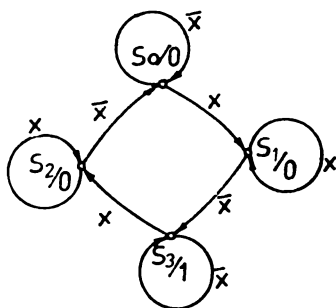


Fig. 10.6

exemple pentru stări inițiale și secvențe de intrare diferite. Grafurile de tranziții corespunzătoare sînt prezentate în fig. 10.7b și respectiv în fig. 10.7c.

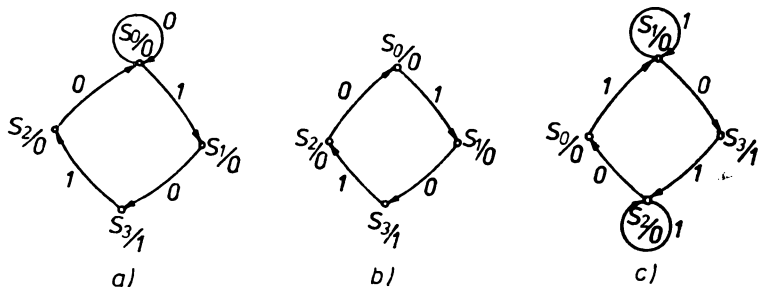


Fig. 10.7

În urma experimentelor efectuate cu circuitul considerat pentru diferite secvențe și diferite stări inițiale se observă că totdeauna tranzițiile se realizează în ordinea $s_0 - s_1 - s_3 - s_2 - s_0 \dots$

Tab. 10.5

Starea inițială	Secvența de intrare	Secvența de ieșire
s_0	101010	010001
s_1	0110110	1000001

Conform asignării făcută la punctul 3, rezultă că pentru toate experimentele realizate nu apar fenomene de concurs între semnalele secundare întrucît stările sînt adiacente.

Se poate studia în continuare o eventuală simplificare a părții combinaționale minimizând expresiile y_1 și y_2 . Folosind fig. 10.5, rezultă :

$$Y_1 = \bar{x}y_2 \cup xy_1 \text{ și } Y_2 = \bar{x}y_2 \cup xy_1,$$

evident mai simple decât expresiile obținute în etapa 2 (lipsesc termenii y_1y_2 și \bar{y}_1y_2). Dar, studiind diagrama din fig. 10.5 se observă că acești termeni permit tocmai eliminarea hazardului static din partea combinațională. Rezultă că circuitul studiat va funcționa fără fenomene de concurs.

Analiza circuitelor asincrone prin metoda simbolică

Prima etapă de analiză este identică cu cea stabilită în cadrul algoritmului general. În cadrul etapei a doua se folosește aceeași metodologie ca și în cazul analizei circuitelor combinaționale prin metoda simbolică (v. capitolul 9). Conform celor arătate atunci, pentru circuitul din fig. 10.4 considerând ordinea variabilelor de intrare în partea combinațională xy_1y_2 se obține următorul simbol de marcare, corespunzător cu fig. 10.8 :

$$(10.7) \quad F(z, Y_1, Y_2) = D_1^{xy_1y_2} \begin{matrix} 1 : (3, 7), 0 : (0, 1, 2, 4, 5, 6) \\ 1 : (1, 3, 6, 7), 0 : (0, 2, 4, 5) \\ 1 : (1, 3, 4, 5), 0 : (0, 2, 6, 7) \end{matrix}$$

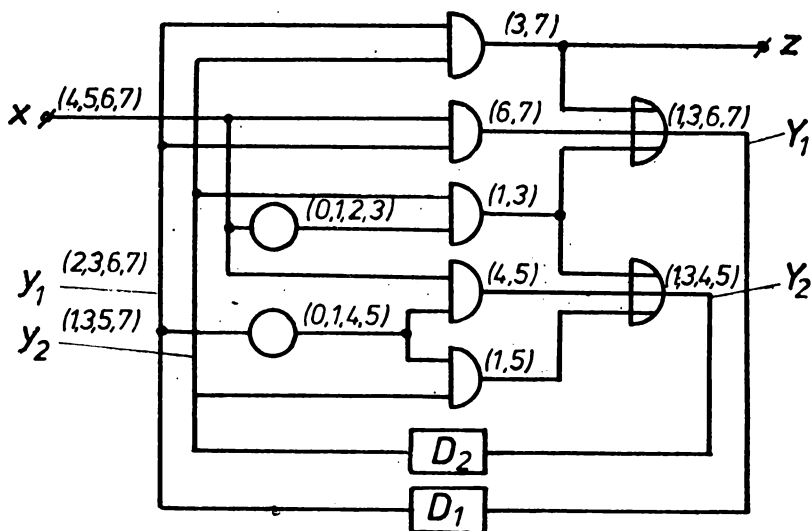


Fig. 10.8

Asignarea se realizează în același mod :

$$\begin{array}{l} y_1 y_2 \quad s_k \\ 0 \ 0 \rightarrow s_0 \\ 0 \ 1 \rightarrow s_1 \\ 1 \ 0 \rightarrow s_2 \\ 1 \ 1 \rightarrow s_3 \end{array}$$

Evoluția semnalelor la ieșire în funcție de evoluția semnalelor la intrare și de starea inițială se stabilește în acest caz folosind *raportul de stare* [40]. Raportul de stare este un raport simbolic între variabilele de intrare și de ieșire ale părții combinaționale. Pentru circuitul studiat raportul de stare este următorul :

$$\frac{x(y_1 y_2)}{z(Y_1 Y_2)}.$$

Din structura acestui raport se observă că variabilele secundare sînt evidențiate prin introducerea în paranteze. În acest mod se pot evidenția ușor stările stabile ($y_i = Y_i$) și cele instabile ($y_i \neq Y_i$). În continuare se va analiza funcționarea circuitului pentru secvența de intrare 01010... plecînd din starea inițială $s_0(y_1 = y_2 = 0)$. Aplicînd prima valoare din secvență ($x = 0$) se obține tranziția :

$$\frac{0(00)}{0(00)} \xrightarrow{x=0} \frac{0(00)}{0(00)},$$

cu păstrarea stării inițiale deoarece numărul de stare generat de $x = 0$ este $0(00)_2 = 0_8$, și care conform simbolului (10.7) aplică atît pe z cît și pe Y_1 și Y_2 în zero.

Dacă se aplică următoarea valoare din secvența de intrare ($x = 1$), rezultă la numărătorul raportului de stare combinația $1(00)_2 = 4_8$. Inspectînd simbolul de marcare (10.7) se observă că numărul de stare 4 aplică numai funcția y_2 în 1 fapt care generează următoarea tranziție :

$$\frac{0(00)}{0(00)} \xrightarrow{x=1} \frac{1(00)}{0(01)}.$$

Deoarece $y_2 = 0$ iar $Y_2 = 1$ starea obținută este instabilă și automatul efectuează o tranziție spontană în sensul atingerii unei

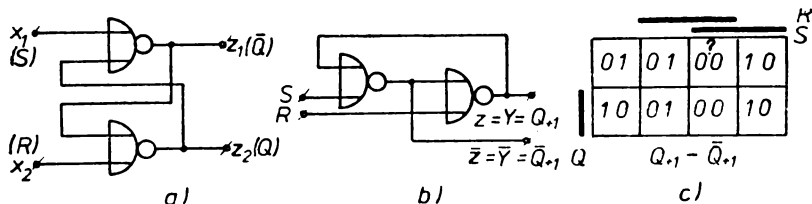


Fig. 10.9

intrări (x_1 și x_2). Un astfel de CBB este cunoscut și sub denumirea de CBB SR datorită denumirii acordate intrărilor circuitului (to Set-a pune, to Reset — a reveni). Deoarece aplicarea repetată a unui semnal de aceeași valoare pe o intrare nu schimbă starea circuitului, acestea se mai numesc și CBB „cu zăvorîre” (*latch*). Intrările R și S sînt uneori denumite și CLEAR și respectiv PRESET.

Studiul unui astfel de CBB ca circuit secvențial asincron este ușurată prin structurarea lui ca în fig. 10.9b. Se observă că ieșirea circuitului este identică cu semnalul de reacție ($z = Y = y$), fapt ce permite definirea CBB SR ca un circuit de tip MOORE. Existența unei singure legături de reacție conduce la concluzia că automatul are numai două stări. Notînd pentru simplitate $Y = Q(v+1) = Q_{+1}$ și $y = Q(v) = Q$, din logigrama din fig. 10.9b rezultă :

$$(10.8) \quad \begin{aligned} Q_{+1} &= (S \downarrow Q) \downarrow R = S\bar{R} \cup \bar{R}Q, \\ \bar{Q}_{+1} &= S \downarrow (\bar{Q} \downarrow R) = \bar{S}\bar{Q} \cup S\bar{R}. \end{aligned}$$

Diagrama Karnaugh corespunzătoare ieșirii și ieșirii inversoare este prezentată în fig. 10.9c. Aceasta constituie în fapt dezvoltarea în forme canonice a funcțiilor din (10.8).

Tab. 10.6

SR	00	10	01	11	Q_{+1}	\bar{Q}_{+1}
s_k						
s_0	$s_0 - s_1$	$s_1 - s_0$	$s_0 - s_1$?	0	1
s_1	$s_1 - s_0$	$s_1 - s_0$	$s_0 - s_1$	$s_0 - s_0$	1	0

Alocând $Q=0 \rightarrow s_0$, și $Q=1 \rightarrow s_1$, se obține tabelul 10.6 al stărilor în care sînt prezentate alături stările următoare pentru ieșirea directă (în stînga) și ieșirea complementară (în dreapta). Din studiul tabelului se observă că la aplicarea simultană a comenzii $S=R=1$ se realizează tranziția ambelor ieșiri în starea stabilă s_0 , cu $Q_{+1}=\bar{Q}_{+1}=0$, ceea ce nu este permis (?). Pentru ca func-

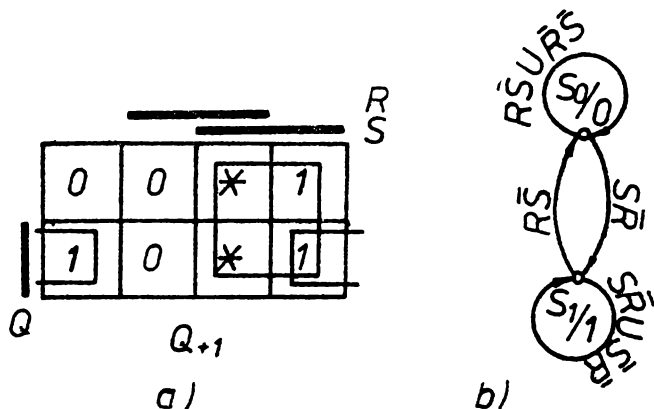


Fig. 10.10

ționarea să decurgă normal este necesară restricția $SR \neq 1 \rightarrow (SR)_v=0$. Avînd în vedere cele arătate, tabelul de adevăr al triggerului SR, denumit și *tabel caracteristic* se poate defini ca cel din tab. 10.7. Prin „neschimbat” se înțelege că se păstrează valoarea avută anterior aplicării comenzii $S=R=0$. În condițiile expuse diagrama Karnaugh pentru CBB SR latch este cea din fig. 10.10a din care rezultă imediat funcția de control a acestuia :

Tab. 10.7

S	R	Q	\bar{Q}	Q ₊₁
0	0	neschimbat		Q
0	1	0	1	0
1	0	1	0	1
1	1	0*	0*	?

*interzis

$$(10.9) \quad Q_{+1} = S \cup \bar{R} Q.$$

Graful de tranziții al circuitului SR pentru funcționarea normală (nu apare $S=R=1$) este cel prezentat în fig. 10.10b. În cazul în care CBB SR se realizează cu circuite NUMAI (fig. 10.11a, tabelul caracteristic este tab. 10.8. Din studiul circuitului rezultă în acest

caz restricția $S \cup R \neq 0 \rightarrow (\overline{SR})_v = 1$. În mod obișnuit CBB SR se simbolizează ca în fig. 10.11b.

Observație. Pentru ca ecuația de control (10.9) să rămână valabilă și în cazul realizării CBB SR cu circuite NUMAI este necesar să se complementeze atât intrările cât și ieșirile circuitului din fig. 10.11a.

Tab. 10.9

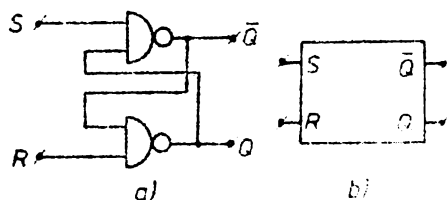


Fig. 10.11

S	R	Q	\overline{Q}	Q_{+1}
0	0	1*	1*	?
0	1	0	1	0
1	0	1	0	1
1	1	neschimbă		0

* interzis

Excitarea circuitului b_vasculant SR

Folosirea circuitului SR în calitate de celulă de memorie impune cunoașterea *condițiilor de excitație* ale acestui circuit astfel ca să se poată asigura funcționarea dorită. În cazul problemelor de sinteză se pot stabili funcțiile de excitație Y_i de forma (10.3). Aceste funcții trebuie realizate de celulele de memorie de tip SR ale căror intrări trebuie comandate astfel încât să se asigure stările următoare impuse. Deci, trebuie rezolvat sistemul de ecuații booleene :

$$(10.10) \quad \begin{cases} Y_i = f_i(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_k) \\ Q_{+1} = S_i \cup \overline{R}_i Q_i, \quad i = 1, 2, \dots, k. \end{cases}$$

Întrucât $Y_i = Q_{+1}$, și $y_i = Q_i$, rezolvarea sistemului (10.10) conduce la necesitatea rezolvării următorului sistem :

$$(10.11) \quad \begin{cases} S_i = h'_i(x_1, x_2, \dots, x_n, Q_1, Q_2, \dots, Q_k) \\ R_i = h''_i(x_1, x_2, \dots, x_n, Q_1, Q_2, \dots, Q_k). \end{cases}$$

Rezolvarea sistemului (10.11) se poate realiza folosind *tabelul de excitație* al CBB. Acest tabel permite stabilirea valorilor intrărilor când se cunoaște starea prezentă și starea următoare. Tabelul de excitație al circuitului SR se obține din graful acestuia (fig. 10.10 b) și este tab. 10.9. În tabel, pe coloanele intrărilor

Tab. 10.9

Q	Q ₊₁	S	R
0	0	0	*
0	1	1	0
1	0	0	1
1	1	*	0

S și R s-au înscris valorile 0 sau 1 dacă aceste valori sînt obligatorii pentru tranziția respectivă. Atunci cînd valoarea la o intrare este indiferentă s-a notat cu asterisc. Astfel, pentru prima linie din tabel deoarece atît Q (starea prezentă) cît și Q₊₁ (starea următoare) sînt 0 este obligatoriu ca S=0, valoarea lui R fiind indiferentă. În graficul CBB această situație este definită de arcul reflectat

pentru s₀ cînd $\overline{SR} \cup \overline{SR} = S$. Dacă însă Q=0 și se dorește ca în starea următoare CBB să comute în s₁(Q₊₁=1), conform grafului, obligatoriu S=1 și R=0, ș.a.

Circuitul basculant JK

Necesitatea eliminării restricțiilor impuse comenzii circuitului SR a condus la realizarea CBB JK. Acest CBB se obține din circuitul SR prin introducerea la intrări a două circuite SI

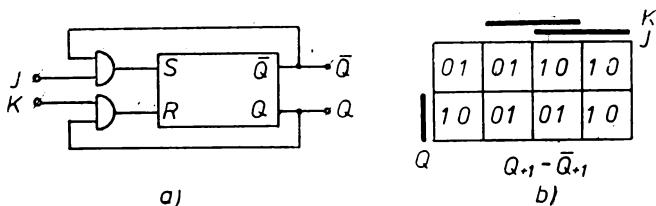


Fig. 10.12

conectate ca în fig. 10.12 a. Folosind relația (10.9) și logigrama din fig. 10.12a se poate stabili ecuația caracteristică a CBB JK :

$$\begin{cases} Q_{+1} = S \cup \overline{R}Q \\ S = J\overline{Q} \\ R = KQ. \end{cases}$$

de unde

$$(10.12) \quad Q_{+1} = J\overline{Q} \cup K\overline{Q}Q = J\overline{Q} \cup KQ.$$

Într-un mod similar, folosind relația (10.8), se obține funcția și pentru ieșirea inversoare :

$$\bar{Q}_{+1} = \bar{J}\bar{Q} \cup \bar{J}KQ \cup KQ.$$

Diagrama Karnaugh pentru circuitul JK este prezentată în fig. 10.12b. Deoarece și în acest caz ieșirea este identică cu variabila secundară de stare, circuitul JK este tot un automat MOORE cu două stări : $Q=0 \rightarrow s_0$ și $Q=1 \rightarrow s_1$. Tabelul stărilor este tab.

Tab. 10.10

	JK					
	00	01	10	11	Q_{+1}	\bar{Q}_{+1}
s_0	$s_0 - s$	$s_0 - s_1$	$s_1 - s_0$	$s_1 - s_0$	0	1
s_1	$s_1 - s_0$	$s_0 - s_1$	$s_1 - s_0$	$s_0 - s_1$	1	0

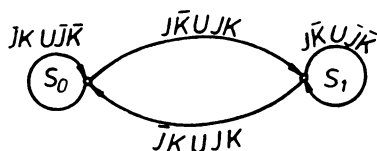


Fig. 10.13

10.10 iar graful de tranziții este prezentat în fig. 10.13. Tabelul stărilor s-a construit de asemenea pentru ambele ieșiri. Din acesta se observă că restricția impusă la circuitul SR s-a eliminat, aplicarea simultană a semnalului 1 la intrări conducând la ieșiri complementare.

Tabelul caracteristic al CBB JK este tab. 10.11 iar tabelul de excitație este tab. 10.12.

Tab. 10.11

J	K	Q_{+1}
0	0	Q
0	1	0
1	0	1
1	1	\bar{Q}

Tab. 10.12

Q	Q_{+1}	I	K
0	0	0	*
0	1	1	*
1	0	*	1
1	1	*	0

Comparând tabele de excitație 10.12 și 10.9 rezultă că circuitul JK prezintă avantajul unor condiții de indiferență mai numeroase și va permite o sinteză mai economică a părții combinaționale.

10.1.1.2. Analiza circuitelor secvențiale asincrone cu CBB ca elemente de memorie

Analiza circuitelor asincrone cu CBB pe buclele de reacție se realizează după același algoritm stabilit inițial. În plus, pentru a stabili evoluția semnalelor de ieșire este necesar să se stabilească evoluția CBB în funcție de excitarea lor. Analiza acestor circuite se va ilustra prin două exemple : unul cu circuite SR și celălalt cu JK.

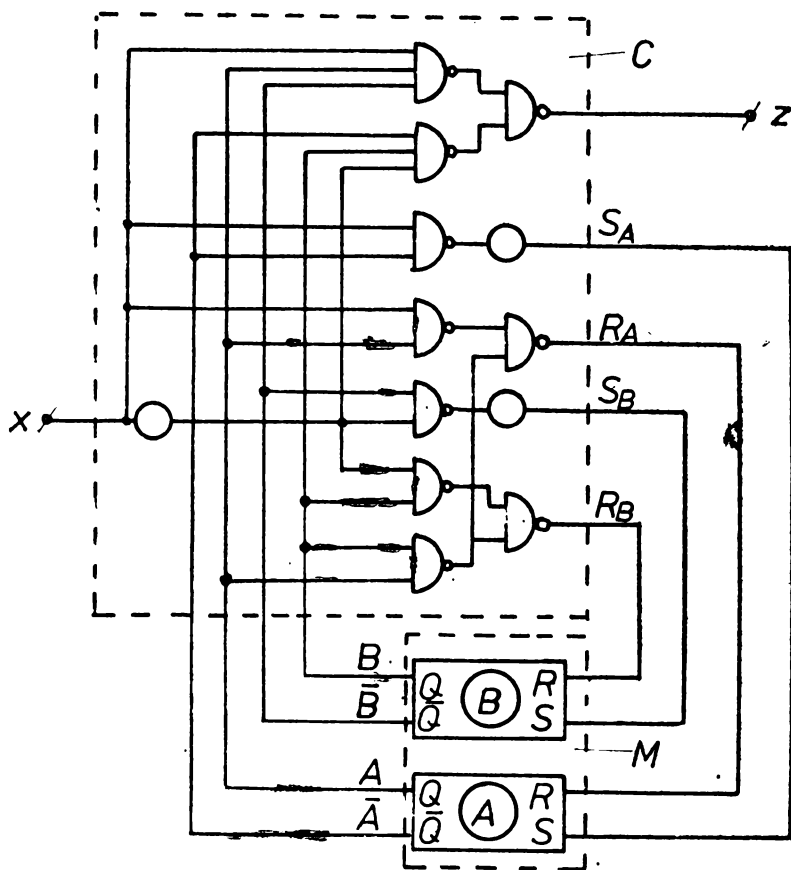


Fig. 10.14

Exemplul 1. Să se analizeze circuitul secvențial din fig. 10.14 pentru secvența de intrare 0110010 și starea inițială s_0 rezultată din asignare.

Se notează cele două circuite SR de pe reacții cu A și respectiv B . Funcțiile de excitație a circuitelor S_A, R_A, S_B, R_B împreună cu funcția de ieșire sînt ecuațiile de control ale circuitului analizat. Notînd, pentru simplitate, $Q_A = A, Q_{A+1} = A_{+1}$ și $Q_B = B, Q_{B+1} = B_{+1}$, din logigramă rezultă :

$$z = \overline{A\bar{B}x} \cdot \overline{A\bar{B}\bar{x}} = A\bar{B}x \cup \bar{A}B\bar{x}$$

$$S_A = \bar{A}x$$

$$R_A = \overline{A\bar{x}} \cdot \overline{A\bar{B}} = Ax \cup AB$$

$$S_B = \bar{B}\bar{x}$$

$$R_B = \overline{B\bar{x}} \cdot \overline{A\bar{B}} = B\bar{x} \cup AB$$

Deoarece $z = f(A, B, x)$ circuitul studiat este de tip MEALY. Dezvoltarea funcțiilor pentru cele două CBB conduce la diagramele Karnaugh din fig. 10.15 a și 10.15 b. În aceste diagrame

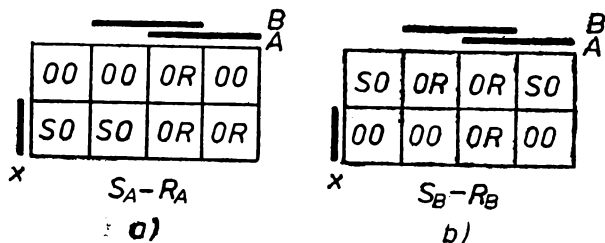


Fig. 10.15

prin S și R s-a notat existența semnalului 1 la intrările respective ale CBB. Evoluția următoare a CBB se determină în funcție de starea prezentă (fig. 10.15) și de tabelul caracteristic al circuitului SR (tab. 10.13). Avînd în vedere notațiile adoptate pentru starea următoare se obțin *diagramele de stare* din figurile 10.16 a și 10.16 b. Aceste diagrame se obțin avînd în vedere că :

$$Q_{+1,i} = f(x_1, x_2, \dots, x_n, Q_1, Q_2, \dots, Q_k), i = 1, 2, \dots, k.$$

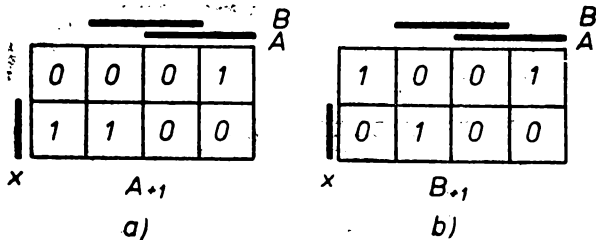


Fig. 10.16

Tab. 10.13

S	R	Q_{+1}
0	0	Q
0	1	0
1	0	1
1	1	?

Tab. 10.14

s_k AB

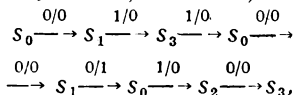
- $s_0 \rightarrow 00$
- $s_1 \rightarrow 01$
- $s_2 \rightarrow 10$
- $s_3 \rightarrow 11$

$s_k \backslash x$	0	1
s_0	$s_1/0$	$s_2/0$
s_1	$s_0/1$	$s_3/0$
s_2	$s_3/0$	$s_0/1$
s_3	$s_0/0$	$s_0/0$

Astfel, pentru CBB A starea următoare A_{+1} pentru $x=0$ și $A = B=0$ este determinată de valorile lui S și R corespunzătoare tab. 10.13 și diagramei din fig. 10.15a, conform căruia $Q_{+1}=Q$ deci $A_{+1}=A=0$. Pentru $x=1$ și $A=B=0$, $S=1$ și $R=0$ (fig. 10.15 a), rezultând $Q_{+1}=A_{+1}=1$ (fig. 10.16), ș.a.

Pentru cele două variabile secundare de stare A și B , se realizează asignarea alăturată. Tabelul 10.14 este tabelul stărilor, iar în fig. 10.17 este prezentat graful de tranziții corespunzător. Tabelul stărilor se determină folosind diagramele din fig. 10.16 și ținând cont de asigurarea de stare.

Din graful de tranziții pentru s_0 și secvența de intrare impusă (0110010) se obțin tranzițiile :



iar la ieșire se va obține secvența 0000100.

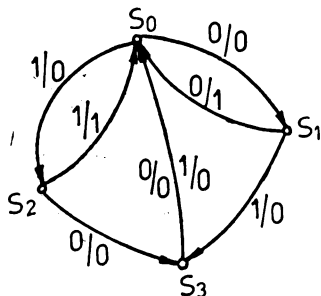


Fig. 10.17

Exemplul 2. Să se analizeze circuitul secvențial asincron cu logigrama din fig. 10.18, pentru aceeași stare inițială și aceeași secvență ca și pentru exemplul 1.

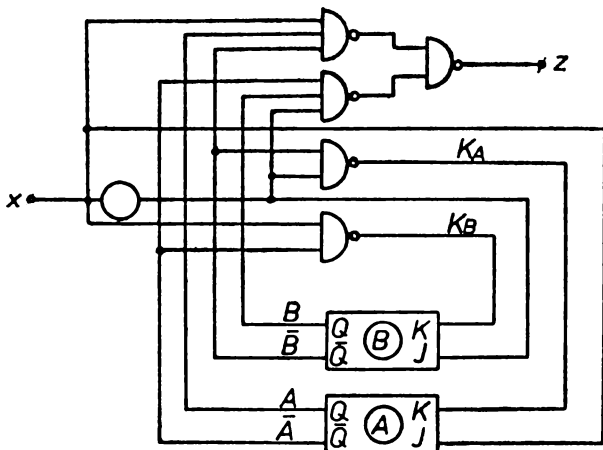


Fig. 10.18

Funcțiile booleene de control se obțin imediat :

$$z = \overline{A}Bx \cdot \overline{A}B\bar{x} = A\bar{B}x \cup \bar{A}B\bar{x}$$

$$J_A = x$$

$$K_A = \overline{B\bar{x}} = B \cup x$$

$$J_B = \bar{x}$$

$$K_B = \overline{Ax} = A \cup \bar{x}.$$

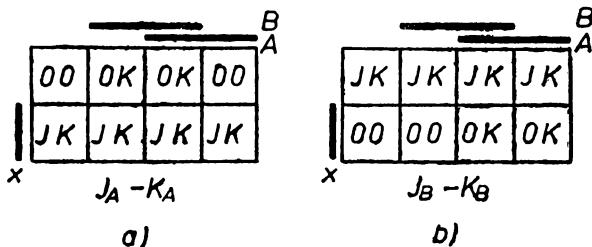


Fig. 10.19

Examinînd expresia funcției de ieșire se observă că și în acest caz circuitul este de tip MEALY. Diagramele Karnaugh pentru cele două CBB sînt prezentate în fig. 10.19, în care existența semnalului 1 la intrări s-a marcat prin denumirea intrării, J și K .

Avînd în vedere tabelul de excitație al circuitului JK (tab. 10.15) se determină diagramele de stare pentru cele două CBB ale circuitului (fig. 10.20).

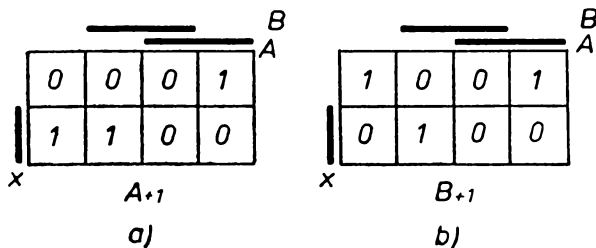


Fig. 10.20

Realizînd aceeași asignare ca și pentru exemplul precedent :

Tab. 10.15

Tab. 10.16

s_k	AB
$s_0 \rightarrow 00$	
$s_1 \rightarrow 01$	
$s_2 \rightarrow 10$	
$s_3 \rightarrow 11$	

	I	K	Q ₊₁
0	0	0	Q
1	0	1	0
0	1	0	1
1	1	1	\bar{Q}

	x	
	0	1
s_k		
s_0	$s_1/0$	$s_2/0$
s_1	$s_0/1$	$s_3/0$
s_2	$s_3/0$	$s_0/1$
s_3	$s_0/0$	$s_0/0$

se obține tab. 10.16 pentru stările următoare. Se observă că tabelul obținut este identic cu tabelul stărilor obținut pentru exemplul precedent, fapt ce conduce la concluzia că cele două circuite sînt echivalente. În ceea ce privește evoluția semnalului la ieșire, evident că va fi identică cu cea de la exemplul precedent.

Studiind evoluția stărilor pentru secvența impusă la intrare se poate observa că apare tranziția s_3-s_0 între stări neadiacente. În acest caz cele două CBB trebuie să comute simultan din starea cu $A=B=1$ în starea cu $A=B=0$ fapt ce poate conduce la fenomene de concurs cu manifestarea hazardului în funcționare.

10.1.2. Analiza circuitelor secvențiale sincrone

Analiza circuitelor sincrone se efectuează similar analizei circuitelor asincrone. Existența impulsurilor de sincronizare face ca funcționarea acestor circuite să fie independentă de intervalul de timp dintre două impulsuri de tact. Din acest motiv în circuitele sincrone fenomenele de concurs sînt complet eliminate. În schimb, pentru o funcționare corectă este necesar ca frecvența generatorului de tact să fie mai mare decît frecvența de comutare a semnalelor de intrare. În acest mod se realizează o descriere precisă a modificărilor de nivel de la intrări.

10.1.2.1. Analiza CBB sincrone

După cum s-a menționat la începutul capitolului, circuitele sincrone folosesc ca elemente de memorie CBB. Aceste circuite de memorie trebuie să fie astfel construite încît să poată funcționa sincron cu impulsurile de la generatorul de tact. Circuitele basculante bistabile folosite în circuitele secvențiale sincrone sînt denumite și CBB sincrone. Aceste CBB au pe lîngă intrările de comandă, denumite și intrări de date (Data inputs), intrări pentru impulsul de tact (Clock). O intrare de date nu modifică starea unui CBB decît în prezența impulsului de tact. Acest fapt permite comutarea sincronă, predictibilă, a mai multe celule de memorie care intră în componența unui circuit.

Sincronizarea CBB se poate face în trei moduri :

a) sincronizare de curent continuu sau pe front (edge triggered) ;

b) sincronizare de curent alternativ ;

c) sincronizare master-slave.

a) *Sincronizarea pe front* asigură comutarea CBB numai la tranziția din 0 în 1 sau din 1 în 0 (sincronizare pe front ascendent, respectiv pe front descendent). Această soluție se folosește mult și la CBB integrate TTL asigurînd o sincronizare de mare viteză. Sincronizarea pe front permite simultan accesul intrărilor de date și transferul datelor la ieșire. Circuitele care se sincronizează în acest mod se numesc și CBB cu comandă pe front.

b) *Sincronizarea de curent alternativ* asigură comutarea CBB numai în timpul modificării semnalului de comandă și se face prin intermediul unui cuplaj capacitiv. Astfel, spre deosebire de sincronizarea pe front unde comutarea se face la atingerea unor anumite nivele de tensiune, în acest caz comutarea se realizează prin impulsuri pozitive sau negative. CBB cu sincronizare de

c.a. mai sînt denumite și CBB *dinamice* și se utilizează în special în sistemele DTL.

c) *Sincronizarea master-slave* asigură o izolare a ieșirilor de intrările de date prin comanda succesivă a două CBB „latch” conectate în serie. Primul CBB se numește MASTER (M) iar al

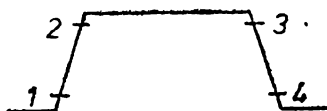


Fig. 10.21

doilea SLAVE (S). Comutarea unui CBB cu sincronizare M—S se face în 4 etape, în funcție de evoluția semnalului de sincronizare (fig. 10.21). Etapele de sincronizare sînt următoarele :

- 1 — circuitul S este izolat de circuitul M,
- 2 — accesul intrărilor de date la M,
- 3 — blocarea intrărilor de date,
- 4 — transferul informației de la M la S.

CBB sincrone în marea majoritate sînt prevăzute cu încă două intrări asincrone (independente de tact): PRESET și CLEAR. Intrările asincrone se folosesc pentru alegerea stării inițiale și sînt prioritare în raport cu celelalte intrări de comandă. Totuși, pentru evitarea erorilor comenzile PRESET și CLEAR se aplică numai în lipsa impulsului de tact.

CBB care se realizează pentru funcționarea sincronă sînt circuitele basculante de tip *T*, *D* și *JK* master slave. Se folosește și sincronizarea circuitului *SR*, fiind denumit CBB de tip STR.

CBB de tip *T* sincron (trigger *T*).

Un trigger *T* se obține dintr-un CBB de tip SR „latch” avînd în plus trei circuite SI conectate ca în fig. 10.22a. Sînt prezentate toate intrările posibile, sincrone și asincrone și s-a

Tab. 1.17

PRESET	CLEAR	Q	\bar{Q}
0	0	1	1
0	1	1	0
1	0	0	1
1	1	fără modificare	

considerat cazul cînd la realizarea circuitului latch se folosesc circuite NUMAI. Efectul comenzilor asincrone este dat în tab. 10.17. Se observă că atunci cînd PRESET = 0, Q = 1, iar dacă CLEAR = 0, Q = 0. În fig. 10.22a se observă că circuitul SR împreună cu circuitele SI de comandă a intrărilor S și R nu este altceva decît un circuit *JK* latch. Rezultă că un trigger *T*

se poate obține și dintr-un circuit JK „latch” cu circuit SI de sincronizare cu impulsul de tact (fig. 10.22b).

Într-adevăr, având în vedere logigrama din fig. 10.22 a și ecuația caracteristică a CBB de tip SR latch, se obține :

$$\begin{cases} Q_{+1} = S \cup \bar{R}Q \\ S = T\bar{Q} \\ R = TQ \end{cases}$$

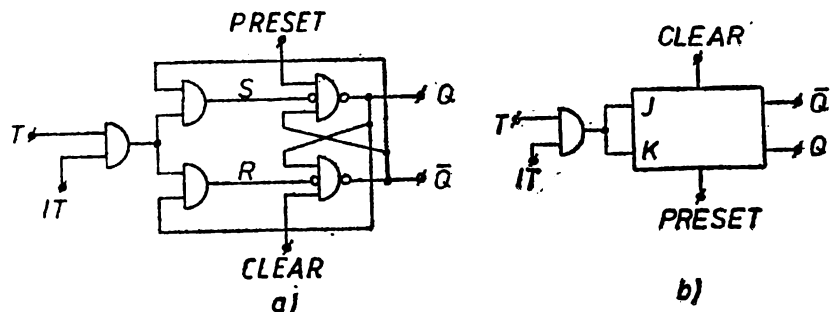


Fig. 10.22

de unde

$$(10.13) \quad Q_{+1} = T\bar{Q} \cup \bar{T}Q, Q = T\bar{Q} \cup \bar{T}Q.$$

Considerând circuitul din fig. 10.22b și ecuația caracteristică a circuitului JK, rezultă :

$$\begin{cases} Q_{+1} = J\bar{Q} \cup \bar{K}Q \\ J = Q = T. \end{cases}$$

de unde: $Q_{+1} = T\bar{Q} \cup \bar{T}Q.$

Cele demonstrate arată posibilitatea implementării triggerului T cu circuite SR sau JK, folosind un număr corespunzător de circuite suplimentare. Și triggerul T este un automat MOORE cu două stări. Ținând cont de (10.13) se obține tabelul 10.18 al stărilor și graful tranzițiilor din fig. 10.23. Conform tab. 10.10 și grafului de tranziții rezultă că triggerul T își schimbă starea odată cu aplicarea impulsului la intrarea de date ($T=1$) indiferent în ce stare inițială se află. Rezultă că un astfel de CBB se poate utiliza ca divizor de frecvență, starea lui schimbându-se o singură dată pentru un ciclu la intrare.

Tab. 10.18

$s_k \backslash T$	0	1	Q_{+1}
s_0	s_0	s_1	0
s_1	s_1	s_0	1

Tab. 10.19

T	Q	Q_{+1}
0	0	0
0	1	1
1	0	1
1	1	0

Tab. 10.20

Q	Q_{+1}	T
0	0	0
0	1	1
1	0	1
1	1	0

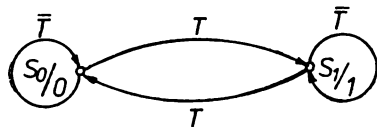


Fig. 10.23

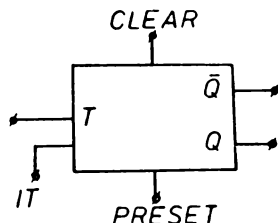


Fig. 10.24

Ecuția caracteristică (10.13) mai poate fi scrisă și astfel:

$$(10.14) \quad Q_{+1} = TQ \cup T\bar{Q} = T \oplus Q,$$

de unde rezultă imediat tabelul caracteristic 10.19 și care prin citire inversă conduce la tabelul de excitație (tab. 10.20). Triggerul T sincron se simbolizează obișnuit ca în fig. 10.24.

CBB de tip D sincron

În fig. 10.24a este prezentată schema logică a CBB de tip D obținut din circuitul SR iar în fig. 10.25b folosind circuitul JK. Pentru schema din fig. 10.25a rezultă:

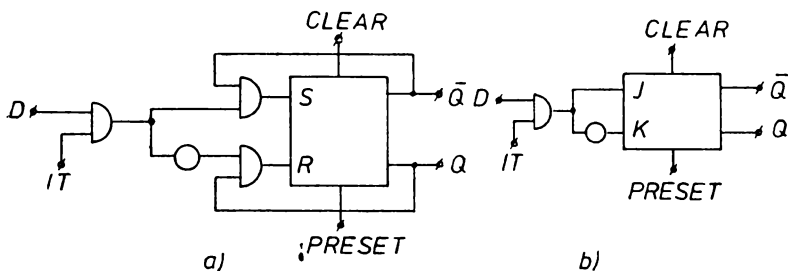


Fig. 10.25

$$(10.15) \quad Q_{+1} = S \cup \bar{R}Q = D\bar{Q} \cup \bar{D}Q = D\bar{Q} \cup DQ = D,$$

iar pentru cea din fig. 10.25 b: $Q_{+1} = \bar{J}\bar{Q} \cup \bar{K}Q = \bar{D}\bar{Q} \cup DQ = D.$

Comparând circuitele din fig. 10.22 cu cele din fig. 10.25 rezultă că CBB D diferă de CBB T prin faptul că intrarea de RESET din circuitul „latch“ (R respectiv \bar{K}) este acționată cu semnal complementat.

Tab. 10.21

$s_k \backslash D$		0		1		Q_{+1}
		s_0	s_1	s_0	s_1	
s_0	s_1	s_0	s_1	s_0	s_1	0
s_1	s_0	s_0	s_1	s_0	s_1	1

Tab. 10.22

D	Q	Q_{+1}
0	0	0
0	1	0
1	0	1
1	1	1

Tab. 10.23

Q	Q_{+1}	D
0	0	0
0	1	1
1	0	0
1	1	1

Tabelul și graful de tranziții ale CBB de tip D sînt prezentate în tab. 10.21 și în fig. 10.26.

Avînd în vedere (10.15) și tab. 10.21 rezultă tabelul caracteristic al CBB de tip D (tab. 10.22), care citit invers conduce la tabelul de excitație (tab. 10.23). În fig. 10.27 este prezentat simbolul triggerului D sincron.

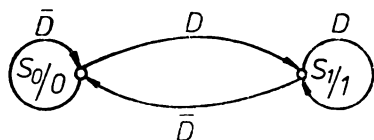


Fig. 10.26

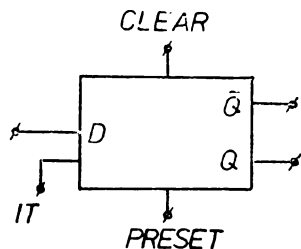


Fig. 10.27

În țara noastră, la I.P.R.S.—Băneasa, se produce capsula integrată CDB 474 E cu două circuite D cu sincronizare pe front pozitiv.

CBB sincrone Master-Slave

Din principiul sincronizării master-slave (fig. 10.21) rezultă că în momentul aplicării impulsului de tact circuitul M reține starea următoare în timp ce circuitul S memorează pe toată durata impulsului de sincronizare starea prezentă. Abia după dispariția impulsului de tact circuitul S trece în starea în care se află circuitul M care îl comandă. În acestmod se separă intrările de date, de ieșiri și se asigură o funcționare corectă independent de durata impulsului de tact. CBB de tip MS se pot realiza atât cu CBB de tip SR (fig. 10.28 a) cât și cu CBB de tip JK (fig. 10.28 b). Datorită lipsei restricțiilor în comenzi, în prezent se realizează cu predilecție circuite JK master-slave.

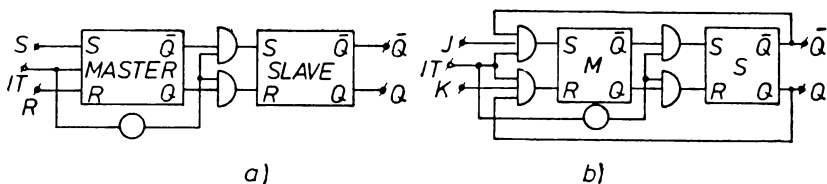


Fig. 10.28

Referitor la schemele din fig. 10.28, se consideră circuitul MS în starea 0 și se dorește înscrierea informației prin aplicarea la S (fig. 10.28 a) sau la J (fig. 10.28 b) a semnalului 1. La apariția semnalului de tact comută circuitul MASTER, memorînd informația. Atît timp cît acționează semnalul de tact circuitul SLAVE nu comută. Cînd $IT=0$ informația din MASTER se transmite în SLAVE și apare la ieșire.

Evident, circuitele MS au aceleași tabele caracteristice și de excitație ca și circuitele „latch” din componentă. În țara noastră se realizează CBB de tip JK master-slave integrate în trei variante :

— capsula CDB 472 cu un circuit JK master-slave avînd trei intrări disponibile pe J și K (J_1, J_2, J_3 și K_1, K_2, K_3) și intrări asincrone CLEAR și PRESET. Cele trei intrări pe J și K sînt introduse prin circuite SI controlate de impulsul de tact. Existența circuitelor SI la intrări permite renunțarea la circuite exterioare în unele aplicații ;

— capsula CDB 473 cu două circuite JK master-slave separate, cu câte o intrare de date și numai intrarea asincronă CLEAR ;

— capsula CDB 476 cu două circuite JK master-slave separate cu câte o intrare de date și cu ambele intrări asincrone.

10.1.2.2. Analiza circuitelor sincrone cu CBB

Analiza circuitelor sincrone se realizează similar circuitelor asincrone. Se va prezenta analiza unui circuit sincron cu CBB pe buclele de reacție printr-un exemplu.

Exemplu. Să se analizeze circuitul secvențial cu logigrama din fig. 10.29 la care atât ieșirea cât și cele trei CBB sînt controlate prin tact.

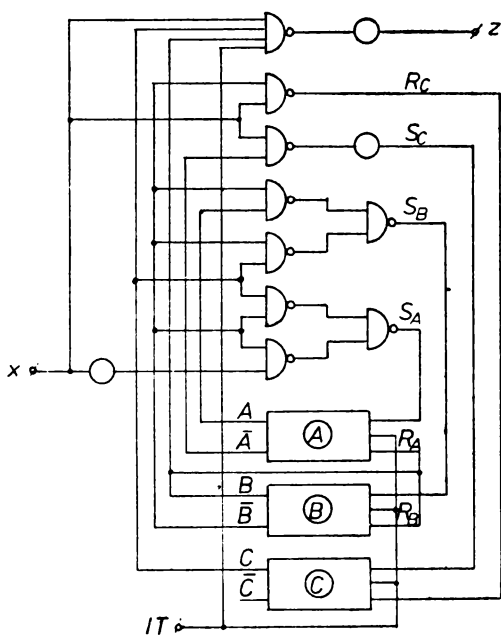


Fig. 10.29

Corespunzător logigramei se stabilesc funcțiile booleene pentru ieșirile din partea combinațională :

$$z = \overline{BC}x = BCx$$

$$S_A = \overline{BC} \cdot \overline{Bx} = \overline{BC} \cup \overline{Bx}$$

$$R_A = B$$

$$S_B = \overline{A} \overline{B} \cdot \overline{BC} = \overline{A} \overline{B} \cup \overline{BC}$$

$$R_B = B$$

$$S_C = \overline{A} x = \overline{A} x$$

$$R_C = \overline{B} x = B \cup \overline{x}$$

Din expresia lui z rezultă că circuitul este de tip MEALY, la ieșire obținându-se semnal 1 pentru $A=B=C=1/x=1$ și $A=0, B=C=1/x=1$. Diagramele Karnaugh corespunzătoare sînt prezentate în fig. 10.30.

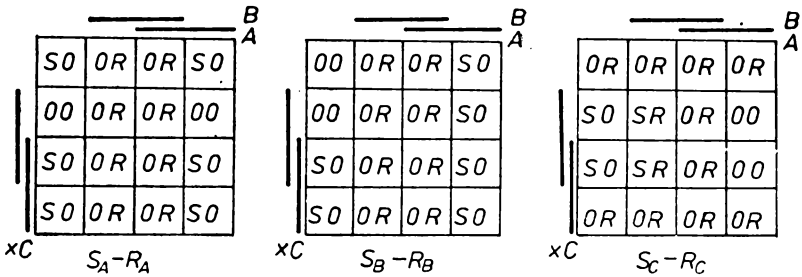


Fig. 10.30

Ținînd cont de tabelul caracteristic al CBB de tip SR (tab. 10.13), în fig. 10.31 sînt prezentate diagramele de stare ale celor trei CBB. Din diagrama pentru bistabilul C se observă că pentru combinațiile $A=0, B=1, C=0$ și $A=0, B=1, C=1$ apare nedeterminare. Astfel, din cele 8 stări posibile două sînt nedeterminate și se elimină. Pentru celelalte 6 stări se realizează asignarea de stare, de unde rezultă tabelul 10.24.

ABC s_k

Tab. 10.24

- 000 $\rightarrow s_0$
- 001 $\rightarrow s_1$
- 100 $\rightarrow s_2$
- 101 $\rightarrow s_3$
- 110 $\rightarrow s_4$
- 111 $\rightarrow s_5$

		s_k					
		s_0	s_1	s_2	s_3	s_4	s_5
x	0	$s_2/0$	$s_3/0$	$s_4/0$	$s_4/0$	$s_0/0$	$s_0/0$
	1	$s_1/0$	$s_5/0$	$s_4/0$	$s_5/0$	$s_0/0$	$s_0/1$

	B			
	A			
	1	0	0	1
	0	0	0	1
	1	0	0	1
	1	0	0	1
xC				
	A ₁			

	B			
	A			
	0	0	0	1
	0	0	0	1
	1	0	0	1
	1	0	0	1
xC				
	B ₁			

	B			
	A			
	0	0	0	0
	1	?	0	0
	1	?	0	1
	0	0	0	0
xC				
	C ₁			

Fig. 10.31

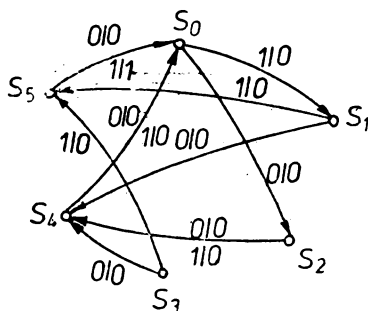


Fig. 10.32

Graful de tranziții atașat automatului este prezentat în fig. 10.32. Din acesta se observă că starea s_3 nu poate fi atinsă din nici o altă stare. În cazul în care această stare nu este inițială ea poate fi eliminată. Atât din tabelul 10.24 cât și din graf se pot stabili secvențele la ieșirea circuitului analizat pentru orice secvență de intrare și stare inițială.

10.2. Sinteza circuitelor secvențiale

Sinteza circuitelor secvențiale este mai complexă decât analiza acestora și comportă mai multe etape. Problemele de sinteză ale circuitelor secvențiale se definesc astfel : *cunoscând corespondența între secvențele de intrare și secvențele de ieșire, se cere să se determine structura circuitului.* Problemele de sinteză se rezolvă similar pentru circuitele sincrone și asincrone, cu diferența că în cazul circuitelor sincrone rezolvarea este chiar mai

simplă datorită lipsei fenomenelor de concurs. Ca și în cazul sintezei circuitelor combinaționale, sinteza circuitelor secvențiale comportă etapele de sinteză abstractă și de sinteză structurală.

I) Sinteza abstractă implică următoarele etape :

1. Stabilirea stărilor posibile în care se poate afla automatul, folosind datele din enunțul problemei de sinteză. Aceste stări se determină în așa fel încât să se țină cont de toate situațiile enunțate, condițiile să fie clare, necontradictorii și să nu se repete. Se numește și *etapa de stabilire primară a stărilor*.

2. Din condițiile impuse și avînd în vedere stările stabilite în etapa precedentă se stabilește graful de tranziții și apoi tabelul stărilor. De multe ori problemele de sinteză se dau prin graful de tranziții, cînd prima etapă este de prisos. După stabilirea tabelului stărilor, denumit și *tabelul primar al stărilor*, în acesta se definesc stările stabile și stările instabile ale automatului.

3. Se elimină stările redondante (evident echivalente) și apoi se minimizează numărul de stări ale automatului. Cele două etape pot fi realizate și simultan prin aplicarea metodei tabelului implicațiilor. Se obține tabelul redus al stărilor prin unificarea (fuzionarea) stărilor echivalente sau compatibile.

4. Se determină numărul variabilelor secundare de stare și se realizează asignarea de stare. Această etapă, denumită și *alocarea semnalelor secundare* sau *codificarea stărilor*, în cazul automatelor asincrone trebuie făcută cu grijă pentru a se evita fenomenele de concurs între semnalele secundare de stare.

5. Se stabilesc expresiile minime pentru funcțiile booleene de ieșire și de excitare a memoriilor. Atunci cînd se folosesc CBB pe buclele de reacție se ține cont de tabelul de excitație al CBB ales.

II) Sinteza structurală comportă următoarele etape :

6. Implementarea funcțiilor obținute în etapa 5 cu circuitele logice impuse și structurarea circuitului sintetizat.

7. Se analizează circuitul obținut, iar în cazul circuitelor asincrone cu reacții directe se verifică dacă nu apare hazardul în funcționare.

10.2.2. Sinteza circuitelor secvențiale asincrone

După cum s-a menționat, sinteza circuitelor asincrone impune respectarea unor anumite condiții care să permită eliminarea fenomenelor de concurs. În rest sinteza se desfășoară după algoritmul prezentat mai sus. Etapele de sinteză se vor parcurge considerând următorul

Exemplu. Să se sintetizeze circuitul secvențial asincron cu două borne de intrare și una de ieșire pentru care corespondența între semnalele de intrare și ieșire este prezentată în fig. 10.33. Circuitul va avea celule de întârziere pe reacții.

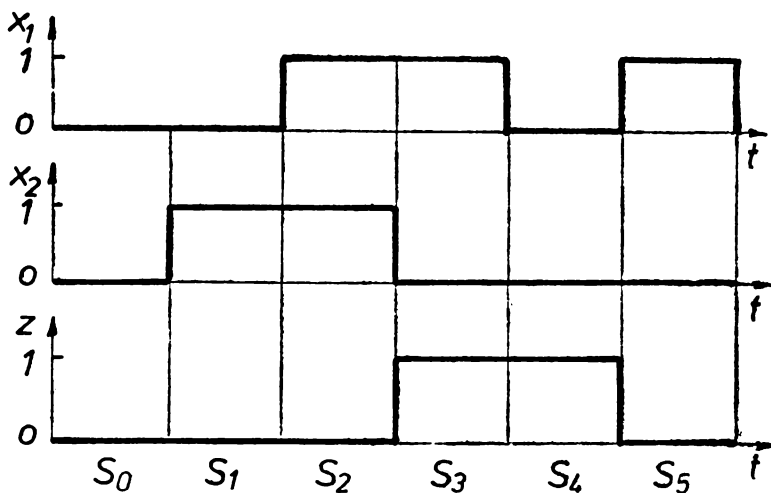


Fig. 10.33

1. Separînd în faze diagrama din fig. 10.33, se pot distinge 6 stări notate $s_0 \dots s_5$. Se observă că la ieșire apare semnal în stările s_3 și s_4 , circuitul sintetizat fiind de tip MOORE. Din condițiile problemei se poate defini funcția de tranziție $f(s, x_1x_2)$:

$$\begin{array}{lll}
 f(s_0, 00) = s_0, & f(s_0, 01) = s_1, & f(s_0, 11) = *, & f(s_0, 10) = * \\
 f(s_1, 00) = s_0, & f(s_1, 01) = s_1, & f(s_1, 11) = s_2, & f(s_1, 10) = * \\
 f(s_2, 00) = *, & f(s_2, 01) = s_1, & f(s_2, 11) = s_2, & f(s_2, 10) = s_3 \\
 f(s_3, 00) = *, & f(s_3, 01) = *, & f(s_3, 11) = s_2, & f(s_3, 10) = s_3 \\
 f(s_4, 00) = s_4, & f(s_4, 01) = *, & f(s_4, 11) = *, & f(s_4, 10) = s_5 \\
 f(s_5, 00) = *, & f(s_5, 01) = *, & f(s_5, 11) = * & f(s_5, 10) = s_5.
 \end{array}$$

Stările nedefinite au fost notate cu asterisc. Funcția de ieșire se poate defini, de asemenea :

$$g(s_0) = g(s_1) = g(s_2) = g(s_5) = 0,$$

$$g(s_3) = g(s_4) = 1.$$

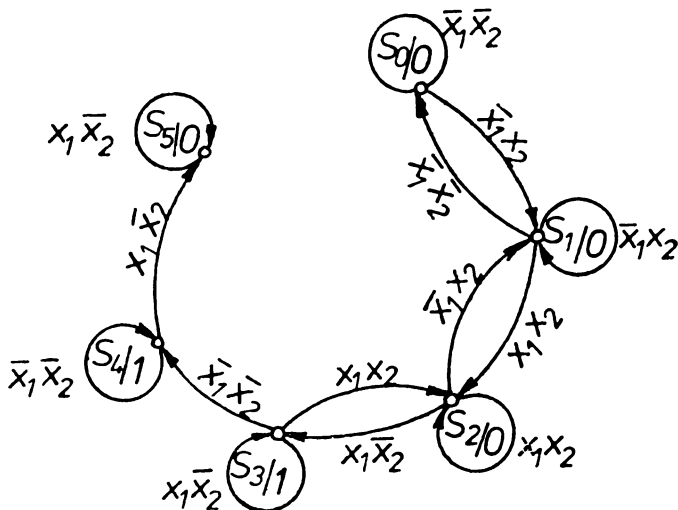


Fig. 10.34

2. Din analiza de la punctul 1 rezultă graful de tranziții din fig. 10.34. Din graful de tranziții se poate stabili imediat tab. 10.25, care constituie tabelul primar al stărilor. În acest tabel s-au marcat stările stabile, iar stările nedefinite cu linie.

Tab. 10.25

x_1x_2	00	01	11	10	z
s_0	s_0	s_1	—	—	0
s_1	s_0	s_1	s_2	—	0
s_2	—	s_1	s_2	s_3	0
s_3	s_4	—	s_2	s_3	1
s_4	s_4	—	—	s_5	1
s_5	—	—	—	s_5	0

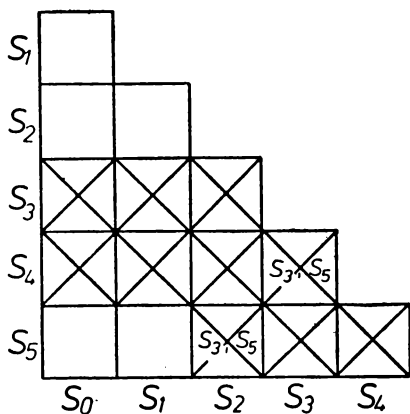


Fig. 10.35

clasa de compatibilități $(s_0s_1s_5)$. Dar întrucât (s_0s_1) este cuprinsă și în clasa $(s_0s_1s_2)$ nu s-a realizat o mulțime închisă. Realizarea clasei de compatibilități $(s_0s_1s_2s_5)$, deci un număr de trei

3. Conform celor arătate în prima parte a capitolului, pentru minimizarea numărului de stări se construiește tabelul implicațiilor din fig. 10.35. În acest tabel cîmpurile libere corespund situațiilor de compatibilitate evidentă. Corespunzător tabelului implicațiilor se construiește tab. 10.26 care permite stabilirea compatibilității maximele. Pentru ultimul rînd al tab. 10.26 se observă că deoarece $s_0 \approx s_1$, $s_1 \approx s_2$ și $s_0 \approx s_2$ se poate forma tripletul $(s_0s_1s_2)$. Se observă că se poate forma și

Tab. 10.26

s_r	stare compatibilă	compatibilitate maximală
s_4		
s_3		
s_2		
s_1	s_2, s_5	$(s_1s_2) (s_1s_5)$
s_0	s_1, s_2, s_5	$(s_0s_1s_2) (s_0s_5) (s_1s_5)$

stări pentru automatul redus, nu este posibilă deoarece implică s_2 compatibil cu s_5 și care conform cu fig. 10.35 $s_2 \not\approx s_5$. Rezultă următoarele clase de compatibilități:

$$\{(s_0, s_1, s_2), (s_3), (s_4), (s_5)\}.$$

Prin separarea clasei $(s_0s_1s_2)$ se mai poate defini următoarea mulțime

$$\{(s_0, s_5), (s_1, s_2), (s_3), (s_4)\}.$$

dar care nu aduce o micșorare a numărului de stări. Păstrînd prima mulțime de compatibilități se realizează fuzionarea rîndurilor 1—2—3 din tab. 10.25. Pentru fuzionare se aplică următoarele reguli evidente :

— în linia rezultată prin fuzionare, dintre stările de pe aceeași coloană, una stabilă și cealaltă instabilă sau nedefinită, se păstrează starea stabilă ; dintre o stare instabilă și una nedefinită se menține starea instabilă dar definită ;

— linia rezultată prin fuzionare păstrează indicele stării cu număr mai mic.

Ca urmare, pentru exemplul considerat se obține tabelul redus al stărilor care este tab. 10.27.

Tab. 10.27

$x_1 x_2$	00	01	11	10	z
s_r					
s_0	s_0	s_0	s_0	s_3	0
s_3	s_4	—	s_0	s_3	1
s_4	s_4	—	—	s_5	1
s_5	—	—	—	s_5	0

Observație. Dacă se transformă automatul *MOORE* dat într-un automat *MEALY* se poate arăta că se pot forma următoarele clase de compatibilități: $(s_0 s_1 s_2)$, (s_3) , (s_4, s_5) . Deci automatul echivalent *MEALY* are cu o stare mai puțin de cît automatul *MOORE* redus, fapt ce conduce la o parte combinațională mai simplă.

4. Rămînînd, pentru simplitatea și continuitatea prezentării, la automatul *MOORE* redus, pentru acesta vor fi necesare $n_k = \log_2 4 = 2$ variabile secundare. Fie y_1 și y_2 cele două variabile secundare și să realizăm asignarea de stare din tab. 10.28. Conform tab. 10.27, din starea stabilă s_0 circuitul trebuie să treacă în stare stabilă s_3 pentru combinația 10 a semnalelor de intrare. În cazul alocării din tab. 10.28, datorită întîrzierilor în funcționarea circuitelor este posibil ca cele două semnale secundare y_1 și y_2 să se modifice la momente diferite de timp, și anume :

— se consideră că y_1 se modifică mai repede decît y_2 ; atunci conform cu tab. 10.16 în loc ca din s_0 să se ajungă în s_3 pentru

Tab. 10.28

s_k	y_1	y_2
s_0	0	0
s_3	1	1
s_4	1	0
s_5	0	1

Tab. 10.29

s_k	y_1	y_2
s_0	0	0
s_3	0	1
s_4	1	1
s_5	1	0

$x_1=1$ și $x_2=0$, circuitul ajunge în s_4 cu $y_1=1$ și $y_2=0$. Dar, s_4 este o stare instabilă pentru combinația 10 și circuitul trece spontan în starea s_5 stabilă.

— dacă y_2 se modifică mai repede decât y_1 , din s_0 circuitul comută direct în s_5 .

Rezultă că pentru alocarea din tab. 10.28 circuitul nu va realiza tranziția s_0-s_3 ci va realiza tranzițiile $s_0-s_4-s_5$ sau s_0-s_5 . În acest caz, al existenței concursului de propagare între semnalele y_1 și y_2 , se manifestă hazardul în funcționare. Pentru o bună funcționare este necesar ca stările vecine să corespundă în asignare unor combinații adiacente pentru variabilele secundare. Alocarea din tab. 10.29 — conformă cu codul Gray — va asigura o funcționare fără fenomene de concurs între semnalele secundare de stare.

Trebuie menționat faptul că în cazul unui număr impar de stări nu se poate realiza o asignare fără apariția fenomenelor de concurs. În aceste situații se introduc stări suplimentare astfel ca să se păstreze adiacența între semnalele secundare corespunzătoare stărilor vecine.

5. Întrucât ieșirea depinde numai de stări, pentru asignarea din tab. 10.29 rezultă :

$$z = \bar{y}_1 y_2 \cup y_1 y_2 = y_2.$$

Avînd în vedere tabelul redus al stărilor (tab. 10.27) și asignarea din tab. 10.29 se stabilește *diagrama de excitații* din fig. 10.36a. În această diagramă s-au indicat și stările corespunzătoare. Din diagrama de excitații se determină imediat, prin separare, diagramele Karnaugh pentru fiecare funcție de excitație (fig. 10.36 b și c).

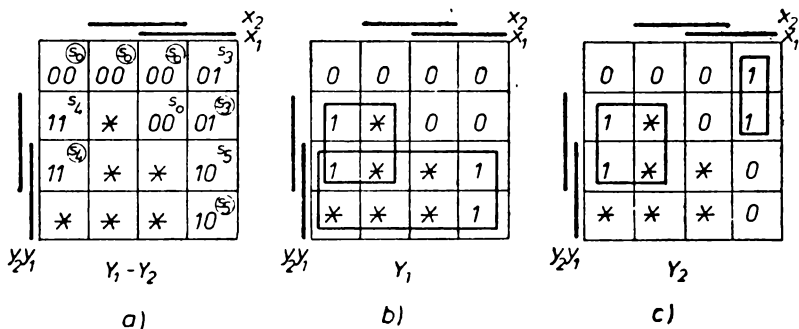


Fig. 10.36

6. Se dispune de module NUMAI. Din diagramele pentru Y_1 și Y_2 , rezultă :

$$Y_1 = (\bar{x}_1 \uparrow y_2) \uparrow \bar{y}_1$$

$$Y_2 = (\bar{x}_1 \uparrow y_2) \uparrow (x_1 \uparrow \bar{x}_2 \uparrow \bar{y}_1).$$

7. Avînd în vedere și expresia pentru ieșire, logigrama circuitului sintetizat este prezentată în fig. 10.37. Analiza funcționării circuitului obținut se efectuează așa cum s-a stabilit în cap. 10.1.

Sinteza circuitelor asincrone prin metoda simbolică

Primele două etape de sinteză se desfășoară conform algoritmului stabilit, analiza condițiilor de funcționare putîndu-se realiza și cu ajutorul raportului de stare (v. cap. 10.1). În cazul folosirii metodei simbolice, în loc de a se realiza tabelul stărilor sau graful de tranziții se folosește o desfășurare temporală a funcționării circuitului prin diagrame de faze [40]. Aceste diagrame conțin

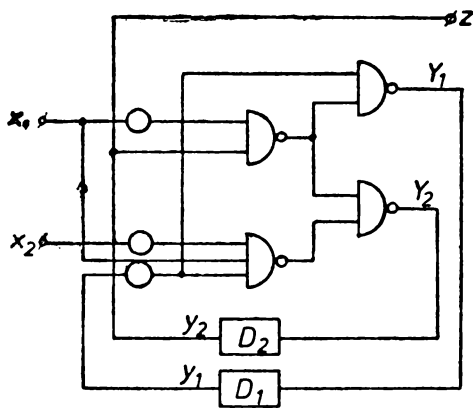


Fig. 10.37

semnalele din circuit-primare, secundare și de ieșire — în desfășurarea impusă de funcționarea dorită. În acest mod se evidențiază toate fazele prin care trece circuitul și se poate face o verificare directă și intuitivă a corectitudinii desfășurării procesului de sinteză. În partea stângă a diagramelor de faze se indică, în ordine, variabilele primare de comandă, secundare de stare și de excitație corespunzătoare, precum și variabilele de ieșire. Fiecare variabilă se separă prin linii orizontale pe care se vor indica, pe faze valorile luate de variabilele considerate : valoarea 1 prin marcare hașurată, iar valoarea 0 prin lipsa marcarului. La partea inferioară a diagramei se indică, în octal, combinațiile variabilelor de comandă și secundare de stare, combinații care constituie numerele de stare. În această etapă se poate face o verificare a *condiției de bună funcționare tehnică* — fazele vecine trebuie să fie marcate de numere de stare adiacente.

Pentru exemplul precedent, diagrama de faze pentru automatul redus este prezentată în fig. 10.38. Numerele de stare pentru fazele vecine fiind adiacente, rezultă că se respectă condiția

x_1			///	///	///			///	///
x_2		///	///						
y_1						///	///	///	///
y_2				///	///	///	///	///	
z					///	///	///	///	
n_k	0	4	14	10	11	1	3	13	12

Fig. 10.38

de bună funcționare. Se observă că în această diagramă se poate evidenția întârzierea de pe buclele de reacție între apariția semnalelor de excitație și semnalele secundare de stare. De asemenea, în diagrama de faze se pot evidenția stările stabile ($y_i = Y_i$), precum și cele instabile ($y_i \neq Y_i$). Se observă că stările stabile corespund numerelor de stare 0, 4, 14, 11, 3 și 12. De asemenea, se poate observa că efectul stărilor 0, 4 și 14, care corespund stărilor s_0 , s_1 și s_2 din tab. 10.25, este identic. Semnalul la ieșire apare în starea s_3 — numărul de stare 11 — și se păstrează în

starea s_4 (număr de stare 3) pînă cînd se realizează tranziția în starea s_5 , cînd $y_1=1$ și $y_2=0$. Din diagramă se poate stabili imediat simbolul de marcare pentru circuitul sintetizat. Considerînd ordinea variabilelor de intrare în partea combinațională ca fiind $x_1x_2y_1y_2$, se obține următorul simbol de marcare :

$$F(z, Y_1, Y_2) = D^{x_1x_2y_1y_2} \begin{matrix} 1: (1, 3, 11, 13), 0: (0, 4, 10, 12, 14) \\ 1: (1, 3, 12, 13), 0: (0, 4, 10, 11, 14) \\ 1: (1, 3, 10, 11), 0: (0, 4, 12, 13, 14). \end{matrix}$$

Minimizînd individual pentru a putea folosi la maximum stările nedefinite, se obține :

— pentru z

$$- 1/3/5^*7^*/11 \ 13 \ 15^*17^* \rightarrow D_{\underline{0} \underline{0} \underline{0} \underline{1}}^{x_1x_2y_1y_2} = y_2,$$

— pentru Y_1

$$- 1/3/5^*7^* \rightarrow D_{\underline{0} \underline{0} \underline{0} \underline{1}}^{x_1x_2y_1y_2} = \bar{x}_1y_2$$

$$- 12/13/16^*17^*/2^*3 \ 6^*7^* \rightarrow D_{\underline{1} \underline{0} \underline{1} \underline{0}}^{x_1x_2y_1y_2} = y_1.$$

— pentru Y_2

$$- 1/3/5^*7^* \rightarrow D_{\underline{0} \underline{0} \underline{0} \underline{1}}^{x_1x_2y_1y_2} = \bar{x}_1y_2$$

$$- 10/11 \rightarrow D_{\underline{1} \underline{0} \underline{0} \underline{0}}^{x_1x_2y_1y_2} = x_1\bar{x}_2\bar{y}_1$$

Rezultă setul de funcții :

$$z = y_2$$

$$Y_1 = \bar{x}_1y_2 \cup y_1 = (\bar{x}_1 \uparrow y_2) \uparrow \bar{y}_1$$

$$Y_2 = \bar{x}_1y_2 \cup x_1\bar{x}_2\bar{y}_1 = (\bar{x}_1 \uparrow y_2) \uparrow (x_1 \uparrow \bar{x}_2 \uparrow \bar{y}_1),$$

identic cu cel obținut anterior.

10.2.1.1. Hazardul în circuitele secvențiale asincrone cu reacții directe

În prima parte a capitolului s-a menționat faptul că în cazul circuitelor asincrone cu reacții directe se poate manifesta hazardul în funcționare chiar dacă asigurarea de stare s-a realizat

astfel ca să nu apară concurs între semnalele secundare. Datorită lipsei întârzierilor pe buclele de reacție și vitezei diferite de propagare a semnalelor în partea combinațională este posibil să apară comutări false ale modulelor componente. În cazul în care două din semnalele de intrare ale unui modul logic se modifică simultan în sensuri opuse, la ieșirea acestuia se manifestă hazardul. Aceste fenomene apar la trecerea circuitului dintr-o stare stabilă în alta.

Evidențierea hazardului în acest caz se poate face prin studierea sistematică a semnalelor la intrarea și la ieșirea modulelor din componența circuitului secvențial. Se pot utiliza diagramele în timp ale semnalelor, dar mai sistematic, se poate construi un tabel care să conțină stările circuitului și valorile semnalelor la intrările și ieșirile modulelor. Modulul la a cărui intrări are loc o modificare a două semnale în sensuri contrarii va fi cel care generează hazardul. *Eliminarea hazardului* de propagare menționat se poate face prin introducerea unor celule de întârziere care să egalizeze timpii de modificare a semnalelor, dar se afectează viteza de funcționare a circuitului. În cazul când este necesară o viteză ridicată de funcționare, eliminarea hazardului se face ca și pentru circuitele combinaționale, prin mărirea redonanței schemei. Modulele redondante introduc *semnale de corecție*, denumite și *semnale de blancare*, la intrarea modulelor care generează hazardul.

Se consideră că hazardul se manifestă la tranziția din starea s_i în starea s_j . Semnalul de corecție k_{ij} trebuie să apară în timpul tranziției s_{ij} . Dar, pentru a nu apare noi fenomene de concurs între semnalele din circuit și cele de corecție, semnalul k_{ij} trebuie să existe în starea s_i . De asemenea, este necesar ca semnalul de corecție să se mențină numai pînă în starea s_j , pentru a nu afecta tranziția în starea următoare. Din cele prezentate rezultă că semnalele de corecție se pot obține din combinațiile semnalelor existente în circuit, primare și secundare, corespunzătoare stărilor respective. Astfel, pentru semnalul k_{ij} se poate scrie :

$$(10.16) \quad k_{ij} = s_i \cup s_{ij} \cup s_j = f(x_1, x_2, \dots, x_n, y_1, y_2, \dots, y_k).$$

Semnalul obținut se introduce prin intermediul unor module suplimentare la intrarea modulului care generează hazardul.

Exemplu. Să se analizeze circuitul din fig. 10.39 obținut din circuitul din fig. 10.37 prin eliminarea celulelor de întârziere D_1 și D_2 pentru mărirea vitezei de funcționare a schemei. Să se

stabilească dacă se manifestă hazardul și în caz afirmativ să se introducă semnale de corecție.

În logigramă s-au numerotat modulele logice ale circuitului studiat. Cunoscând semnalele de comandă, stările și semnalele secundare, s-a întocmit tabelul 10.30 care conține semnalele la ieșirile modulelor componente ale circuitului.

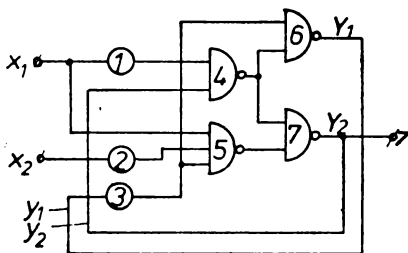


Fig. 10.39

Tab. 10.30

s_k	x_1	x_2	Semnalele la ieșirea modulelor							Obs.
			1	2	3	4	5	6	7	
			\bar{x}_1	\bar{x}_2	\bar{y}_1	$\bar{x}_1\bar{y}_2$	$\bar{x}_1\bar{x}_2\bar{y}_1$	y_1	y_2	
s_0	0	0	1	1	1	1	1	0	0	
	0	1	1	0	1	1	1	0	0	
	1	1	0	0	1	1	0	0	0	
s_3	1	0	0	1	1	1	0	0	1	H_{3-4}
s_4	0	0	1	1	0	0	1	1	1	
s_5	1	0	0	1	0	1	1	1	0	

Din studiul sistematic al semnalelor la intrarea celor 7 module logice rezultă că la ieșirea modulului numărul 7 se poate manifesta hazardul la tranziția din starea s_3 în starea s_4 . Modulul 7 este comandat de modulele 4 și 5. Or, la tranziția din s_3 și s_4 semnalele la ieșirea celor două module se modifică în sensuri contrarii. Dacă se modifică mai întâi semnalul de la intrarea modulului 4 semnalele la intrarea modulului 7 capătă valoarea 0, fapt ce nu perturbă semnalul 1 care trebuie să existe la ieșirea acestuia. Însă, dacă comută mai întâi modulul 5, la intrarea modulului 7 se aplică combinația 11 fapt care atrage la ieșirea sa apariția semnalului 0. Dacă modificarea se face când încă circuitul se află în starea s_3 , atât y_1 cât și y_2 vor avea valoarea 0, deci circuitul va trece în starea s_0 . Deoarece pentru $x_1=1$ și $x_2=0$ starea s_3

nu e stabilă se va reveni automat în starea s_3 (hazardul nu afectează esențial funcționarea). Dar dacă modificarea semnalelor se face cînd circuitul a trecut în s_4 , $y_1=1$ iar $y_2=0$ și circuitul va trece în starea s_5 care este stabilă pentru $x_1=1$ și $x_2=0$. În acest caz hazardul afectează *esențial* funcționarea circuitului și trebuie eliminat.

Pentru eliminare hazardului evidențiat trebuie introdus un semnal de corecție la tranziția între stările s_3 și s_4 . Conform cu (10.16), se poate scrie :

$$(10.17) \quad k_{34} = s_3 \cup s_{34} \cup s_4.$$

Combi-națiile semnalelor de intrare și secundare pentru stările s_3 și s_4 se cunosc, dar trebuie stabilite și semnalele în timpul

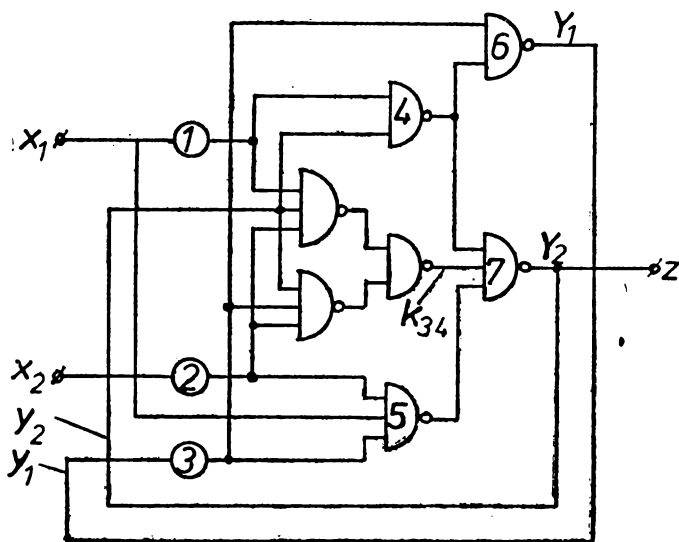


Fig. 10.40

tranzitiei s_{34} . Avînd în vedere că semnalele secundare prezintă întârzierea față de semnalele de excitație se obține tab. 10.31. A esta conține combinațiile semnalelor primare și secundare de stare pentru toate tranzițiile.

Tab. 10.31

s_r	x_1	x_2	y_1	y_2
s_0	0	0	0	0
s_{03}	1	0	0	0
s_3	1	0	0	1
s_{34}	0	0	0	1
s_4	0	0	1	1
s_{41}	1	0	1	1
s_5	1	0	1	0

Avînd în vedere relația (10.17), din tabel se obține imediat :

$$k_{34} = x_1 \bar{x}_2 \bar{y}_1 y_2 \cup \bar{x}_1 \bar{x}_2 \bar{y}_1 y_2 \cup \cup \bar{x}_1 \bar{x}_2 y_1 y_2 = (\bar{x}_2 \uparrow \bar{y}_1 \uparrow y_2) \uparrow \uparrow (\bar{x}_1 \uparrow \bar{x}_2 \uparrow y_2).$$

În fig. 10.40 este prezentată logigrama circuitului în care s-a eliminat hazardul prin introducerea semnalului de corecție k_{34} la intrarea modulului 7.

10.2.2. Sinteza circuitelor secvențiale sincrone

Sinteza circuitelor secvențiale sincrone se desfășoară după același algoritm stabilit inițial, cu deosebirea că în etapă a 5-a intervine și programarea *CBB* folosite ca celule de memorie. Datorită faptului că în aceste circuite nu pot apare fenomene de concurs, asignarea de stare poate fi oarecare. Mai mult, pentru circuitele sincrone se pot lua în considerație și alte criterii la asignare, cum ar fi obținerea unui număr minim de negări sau cele mai simple expresii pentru funcțiile de excitație /35, 45/.

Exemplu. Să se realizeze sinteza circuitului secvențial sincron cu o intrare (x) și o ieșire (z) care trebuie să funcționeze după graful din fig. 10.41. Se vor folosi ca elemente de memorie *CBB* de tip JK master-slave sau de tip *D*.

În tab. 10.32 este prezentat tabelul stărilor corespunzător grafului dat. Atît din tabel cît și din graf se observă că circuitul este de tip *MEALY*. Realizînd tabelul implicațiilor rezultă că nu se poate realiza nici o reducere a numărului de stări — tab. 10.32 fiind și tabelul redus.

Tab. 10.32

s_k	s_0	s_1	s_2	s_3	s_4	s_5
x						
0	$s_4/0$	$s_2/1$	$s_2/0$	$s_2/0$	$s_3/0$	$s_1/0$
1	$s_1/0$	$s_2/0$	$s_0/0$	$s_0/0$	$s_5/1$	$s_2/0$

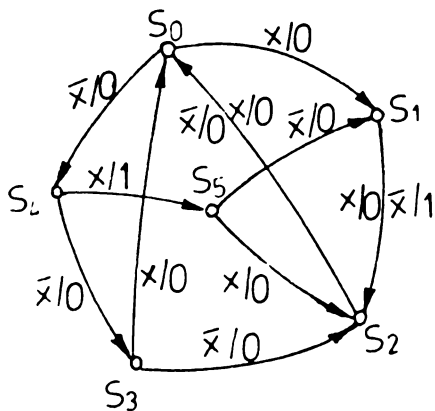


Fig. 10.41

	B A			
	100	000	*	011
	001	000	*	101
	010	000	*	010
	010	010	*	001
x C	A+1 - B+1 - C+1			

Fig. 10.42

Cunoscînd numărul de stări ale circuitului, rezultă numărul de CBB $n_k \geq \log_2 6 \rightarrow n_k = 3$; se notează A, B și C . Întrucît cu trei CBB se pot asigura $2^3 = 8$ stări, în cazul de față nu vor fi folosite decît 6 combinații. Notînd $Q_A = A, Q_B = B$ și $Q_C = C$, în tab. 10.33 este prezentată asignarea de stare. În fig. 10.42 este prezentată diagrama de excitații, în care s-a notat $Q_{A+1} = A+1, Q_{B+1} = B+1$ și $Q_{C+1} = C+1$.

Tab. 10.33

s_k	A	B	C
s_0	0	0	0
s_1	0	0	1
s_2	0	1	0
s_3	0	1	1
s_4	1	0	0
s_5	1	0	1

Tab. 10.34

Q	Q_{+1}	J	K
0	0	0	*
0	1	1	*
1	0	*	1
1	1	*	0

Sinteza cu CBB de tip JK. În tab. 10.34 este reluat tabelul de excitație al CBB de tip JK. Diagramele de excitație pentru cele trei CBB ale circuitului sintetizat sînt prezentate în fig. 10.43. Acestea s-au stabilit pentru fiecare CBB, ținînd cont de starea

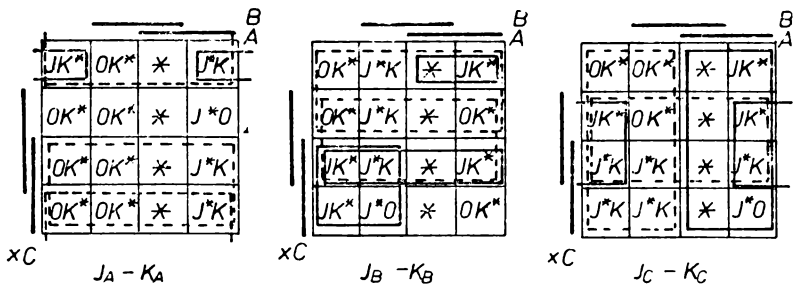


Fig. 10.43

prezentă (A, B, C) , starea următoare (A_{+1}, B_{+1}, C_{+1}) din fig. 10.42 și tabelul de excitație al CBB . În cele trei diagrame, prin J și K s-au notat valorile 1 la intrările respective iar prin J^* și K^* situațiile de nedefinire. Stările nedefinite ale circuitului s-au menționat în cîmpurile respective prin asterisc. Completarea diagramelor de excitație din fig. 10.43 se realizează în modul următor. Se consideră diagrama pentru CBB A : pentru $A = B = C = 0$ și $x = 0$, conform cu diagrama din fig. 10.42 $A_{+1} = Q_{+1}$ trebuie să fie 1. În tabelul de excitație al circuitului JK pentru $A = Q_A = 0$ și $A_{+1} = Q_{A+1} = 1$ rezultă $J = 1$ și $K = K^*$, valori care se trec în cîmpul corespunzător din diagrama de excitație pentru CBB A , ș.a.

Din diagramele din fig. 10.43 se stabilesc expresiile minime pentru intrările J și K ale fiecărui CBB . Cu linie continuă sînt realizate reunitrile cîmpurilor adiacente pentru J iar cu linie întreruptă pentru K . Pentru ieșire se obține diagrama Karnaugh din fig. 10.44, completată conform tab. 10.32 și asignării din tab. 10.33. Corespunzător reunitrilor realizate în diagramele din fig. 10.43 și fig. 10.44, rezultă funcțiile booleene implementate cu funcții *NUMAI*:

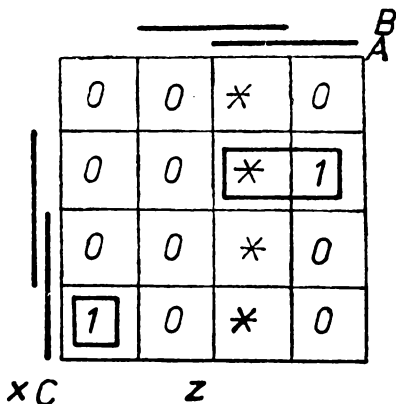


Fig. 10.44

$$\begin{aligned}
 J_A &= \overline{B \uparrow C \uparrow \bar{x}} \\
 K_A &= \bar{C} \uparrow x \\
 J_B &= (\bar{A} \uparrow \bar{C}) \uparrow (A \uparrow C \uparrow x) \uparrow (C \uparrow x) \\
 K_B &= C \uparrow \bar{x} \\
 J_C &= \bar{A} \uparrow (\bar{B} \uparrow x) \\
 K_C &= A \uparrow \bar{x} \\
 z &= (\bar{A} \uparrow \bar{B} \uparrow C \uparrow \bar{x}) \uparrow (A \uparrow \bar{C} \uparrow x).
 \end{aligned}
 \tag{10.18}$$

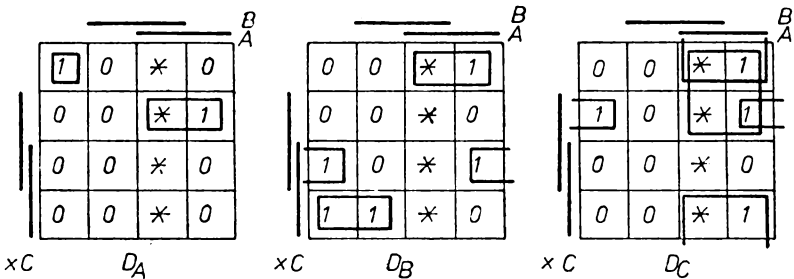


Fig. 10.45

Sinteza cu CBB de tip *D*. Corespunzător asignării din tab. 10.33 și avînd în vedere diagrama din fig. 10.42 și tabelul de excitație al CBB de tip *D* (tab. 10.35), se obțin diagramele de excitație ale celor trei CBB din fig. 10.45. Ieșirea nu depinde de tipul CBB, avînd expresia stabilită anterior. Corespunzător cu fig. 10.45 rezultă următoarele expresii minime implementate cu funcții NUMAI:

Tab. 10.35

<i>Q</i>	<i>Q</i> ₊₁	<i>D</i>
0	0	0
0	1	1
1	0	0
1	1	1

$$\begin{aligned}
 D_A &= (\bar{A} \uparrow \bar{B} \uparrow \bar{C} \uparrow \bar{x}) \uparrow (A \uparrow \bar{C} \uparrow x) \\
 D_B &= (A \uparrow \bar{C} \uparrow x) \uparrow (\bar{B} \uparrow C \uparrow x) \uparrow (\bar{A} \uparrow C \uparrow x) \\
 D_C &= (A \uparrow \bar{x}) \uparrow (A \uparrow \bar{C}) \uparrow (\bar{B} \uparrow \bar{C} \uparrow x) \\
 z &= (\bar{A} \uparrow \bar{B} \uparrow C \uparrow \bar{x}) \uparrow (A \uparrow \bar{C} \uparrow x)
 \end{aligned}
 \tag{10.19}$$

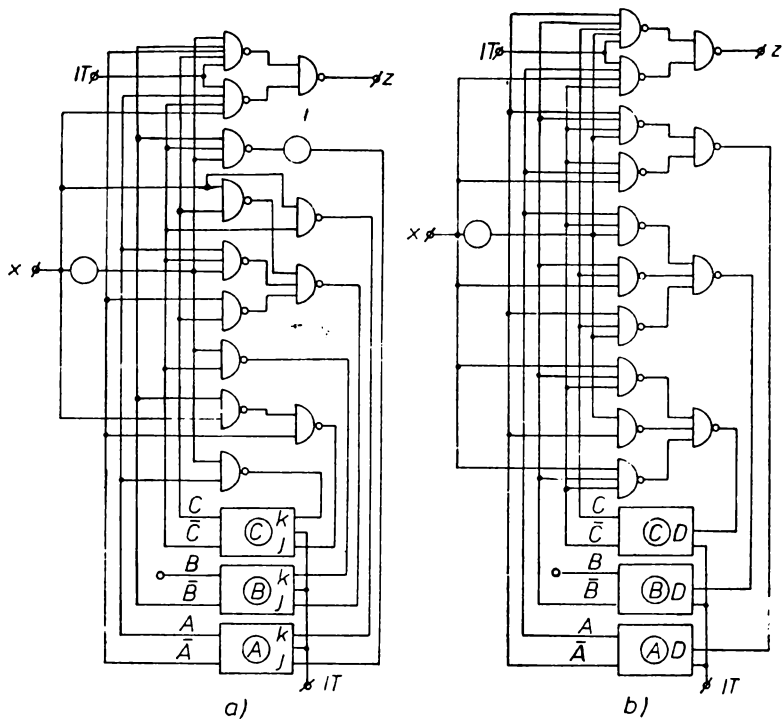


Fig. 10.46

Corespunzător seturilor de funcții (10.18) și (10.19) în figurile 10.46a și 10.46b sînt prezentate schemele logice ale circuitului sintetizat.

BIBLIOGRAFIE

1. Abhyankar, S., *Absolute minimal expressions of Boolean functions*. IRE Trans. on EC, vol. EC-8, nr. 1, 1959.
2. Bodea, M., Vild-Maior, A. A., *Circuite integrate I²L*. A.M.C., nr. 26. pp. 199.
3. Boțan, N. V., *Comanda sistemelor de acționare electrică*. Ed. Tehnică, Buc. 1977.
4. Caldwell, S. H., *Les circuits de comutation et leur conception*. Eyrolles Paris, 1967.
5. Creangă, I. ș.a., *Introducere algebrică în informatică*. Teoria automatelor, Ed. Junimea, Iași, 1973.
6. Denis-Papin, M., Malgrange, Y., *Exerciții de calcul boolean cu soluțiile lor*. Ed. Tehnică, București, 1970.
7. Felea, I. ș.a., *Circuite cu tranzistoare în industrie*. Vol. II. Ed. Tehnică, București, 1964.
8. Florine, J., *La synthèse des machines logiques*. Dunod, Paris, 1964.
9. Florine, J., *Automatismes à séquences et commandes numériques*. Dunod, Paris, 1969.
10. Gelder, E., *Le transistor en comutation*. Chiron, Paris, 1969.
11. Girard, P., Naslin P., *Constructions des machines séquentielles industrielles*. Dunod., Paris, 1973.
12. Goldenbrg, L. M., *Teoria și calculul circuitelor de impulsuri*. Ed. Tehnică, București, 1972.
13. Hartmanis, I., Stearns, R. E., *Algebraic Structure Theory of Sequential Machines*. Prentice-Hall. Englewood Cliffs, New York, 1966.
14. Huțanu, C., *Evidențierea hazardului static în circuitele logice combinatoriale sintetizate prin metoda simbolică*. Bul. Inst. polit. Iași (electrotehnică), fasc. 3-4, 1975.

15. Huțanu, C., Cuciureanu, D., *Sinteza unui automat secvențial cu ciclul cubic de funcționare*. Comunicările Ses. tehnico-științifice a I.U.G. Progresul, Brăila, 4–6 iulie, 1975, pp. 255.
16. Huțanu, C., Gheorghiescu, D., *Sinteza schemei secvențiale de comandă a unei mașini automate de găurit*. Lucrările Ses. de comunicări tehnico-științifice I.C.P.E., București, 4–6 noiembrie, 1976, secția D.
17. Huțanu, C., Reus, N., *Sinteza automatelor secvențiale cu trigger de diferite tipuri, prin metoda simbolică*. Bul. Inst. polit. Iași, (electrotehnică), fasc. 3–4, 1978.
18. Hurley, R. B., *Transistor Logic Circuits*. John Wiley and Sons, Inc., New Ycrk, 1961.
19. Ionescu, D. ș.a., *Baze matematice pentru calculatoare numerice*. Ed. Tehnică, București, 1978.
20. Jaczewski, I., *Logical Systems for Industrial Applications*. PWN, Warszawa, 1978.
21. Kalman, R. E., *Teoria sistemelor dinamice*. Ed. Tehnică, București, 1976.
22. Kaufman, A., Précigout, M., *Elemente de teoria mulțimilor și algebră modernă*. Ed. Tehnică, București, 1973.
23. Lagasse, J., *Logique combinatoire et séquentielle*. Dunod, Paris, 1971.
24. Livovschi, L., *Circuite cu contacte de rele*. Ed. Academiei R.S.R., București, 1968.
25. Marinescu, N., Barbu, E., *Teoria circuitelor de comutație*. Ed. Did. și ped., București, 1964.
26. Moisiu, G. C., *Teoria algebrică a schemelor cu contacte și rele*. Ed. Tehnică București, 1965.
27. Morris, N. M., *Circuits logiques*. Masson et C^{te}, Paris, 1971.
28. Morris, R. L., Miller, I. L., *Proiectarea cu circuite integrate TTL*. Ed. Tehnică, București, 1974.
29. Muntean, I., *Sinteza automatelor finite*. Ed. Tehnică, București, 1977.
30. Murgan, A., *Tehnici numerice*. Fasc. 1, Inst. polit. București, 1974.
31. Naslin, P., *Circuite logice și automatizări secvențiale*. Ed. Tehnică. București, 1967.
32. Necula, N. N., *Circuite logice. Sinteza automată. Circuite logice multi-prag*. Ed. Acad. R.S.R., București, 1972.
33. Pătrușescu, M., *Diodele tunel și utilizarea lor*. Ed. Acad. R.S.R., București, 1968.
34. Pospelov, D. A., *Loghiceskie metodî analiza i sinteza shem*. Izdatel'stvo Energiia, Moscova, 1974.
35. Pop, V., Popovici, V., *Circuite de comutare aplicate în calculatoare*. Ed. Facla. Timișoara, 1976.

36. Pressman, A. I., *Design of Transistorised Circuits for Digital Computers*. John F. Rider Publisher, Inc., New York, 1959.
37. De Palma, R., *Algebra binară a lui Boole și aplicațiile ei în informatică*, Ed. Tehnică, București, 1976.
38. Raymond, I. P., Minne, I., *Les schemas d'automatisme*. Dunod Paris, 1971.
39. Reus, N., *Tratarea simbolică a schemelor de comutație*. Ed. Acad. R.S.R. București, 1971.
40. Reus, N., Huțanu, C., *Sinteza automatelor finite cu memorie prin metoda simbolică*. Bul. Inst. polit. Iași (electrotehnică), fasc. 3—4, 1977.
41. Sprinceană, N. ș.a., *Automatizări discrete în industrie*. Culegere de probleme, Ed. Tehnică, 1978.
42. Săvescu, M. ș.a., *Circuite electronice*. Vol. 2 și 3. Ed. Tehnică, București, 1970 și 1973.
43. Roman, S. ș.a., *Tranzistoare cu efect de câmp*. Ed. Tehnică, București, 1972.
44. Simbotin, C., Tanasiciuc, Cl., *Comutația statică în autmatică*. Ed. Tehnică, București, 1970.
45. Unger, H. S., *Asynchronous Sequential Switching Circuits*. John Wiley & Sons, New York, 1969.
46. Varșavschii, V. I., *Aperiodiceskie automati*. Izdatelstvo Nauka, Moscova, 1976.
47. Vild-Maior, A. A., Bodea, M., *Circuite integrate I²L. Circuite elementare. Circuite complexe*. A.M.C., nr. 27, pp. 167.
48. Vătășescu, A. ș.a., *Dispozitive semiconductoare. Manual de utilizare*. Ed. Tehnică, București, 1975.
49. Siemens, A. G., *Optoelectronik halbleiter*. Datenbuch 1973/1974.
50. I.P.R.S.—Băneasa. *Catalog cu diode și tiristoare*, 1976—1977.
51. I.P.R.S.—Băneasa. *Catalog cu tranzistoare*, 1976—1977.
52. I.P.R.S.—Băneasa. *Catalog cu circuite integrate digitale*, 1978—1979.

Redactor : VICENȚIU DONOSE
Referent științific :
Prof. dr. doc. ing. N. V. BOȚAN
Tehnoredactor : MIHAI BUJDEI

Apărut 1983. Format 61×86/16. Coli tipo 15.
Bun de tipar la 1.III.1983.
Editura Junimea, str. Gheorghii Dimitrov nr. 1
IAȘI — ROMÂNIA



Tipărit la Intreprinderea poligrafică Iași
str. 7 Noiembrie nr. 49

Lucrarea, constituită din trei părți, prezintă principalele noțiuni care stau la baza tehnicilor numerice. După o definiție a elementelor teoretice, sînt prezentate principalele tipuri de circuite logice, pentru ca în final să se arate modul lor de utilizare în comenzile secvențiale.

Nivelul lucrării este accesibil unei arii largi de cititori și în conjuncție cu numeroase exemple permite abordarea gradată a analizei și sintezei sistemelor numerice.

