

CAPITOLUL 3

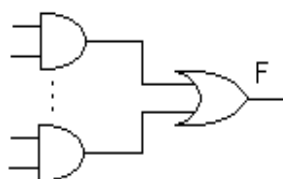
MINIMIZAREA FUNCȚIILOR DE COMUTARE

3.1. INTRODUCERE

În acest capitol sunt prezentate principalele metode de obținere a celei mai simple forme de exprimare a funcțiilor de comutare (expresii booleene), denumită *formă minimă*.

În practică suntem puși în fața a două probleme diferite: *analiza circuitului*, care constă în determinarea funcției de transfer, sau *sinteza circuitului*, în care pe baza funcției de transfer se determină structura circuitului. În cazul sintezei se urmărește realizarea circuitului a cărei expresie asociată este cea mai simplă. Pentru obținerea formei celei mai simple se va face minimizarea expresiei canonice date. Găsirea formei minime este importantă pentru analiză, dar mai ales pentru sinteza circuitelor de comutare care realizează funcția cerută, deoarece acestor forme le corespund circuite de comutare cu preț minim.

FND are următoarea structură:



FNC are următoarea structură:

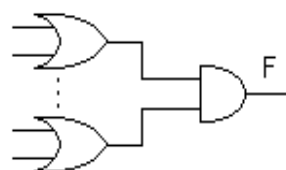


Figura 3.1. Tipuri de structuri ale circuitelor de comutare.

Structura formei normale disjunctive (FND), respectiv a formei normale conjunctive (FNC), este dată în figura 3.1.

În literatura de specialitate sunt prezentate mai multe metode de minimizare ale funcțiilor booleene, fiecare dintre acestea prezentând anumite avantaje. În cele ce urmează vor fi prezentate doar câteva dintre acestea, și anume cele mai reprezentative.

3.2. METODA MINIMIZĂRII FUNCȚIILOR PE BAZA AXIOMELOR ȘI TEOREMELOR ALGEBREI BOOLEENE

Folosind axiomele și teoremele algebrei booleene, o funcție dată sub formă canonică disjunctivă sau conjunctivă poate fi scrisă în general sub o formă mai simplă, cu un număr mai mic de termeni respectiv factori, căreia să îi corespundă o rețea cu cost mai mic. Această metodă de minimizare a funcției necesită însă multă experiență și îndemânare din partea proiectantului, motiv pentru care nu poate fi aplicată cu succes decât după o practică îndelungată în proiectarea circuitelor de comutare. De multe ori însă, forma funcției obținute în urma unor calcule laborioase nu este forma minimă.

3.3. METODA DIAGRAMELOR DE MINIMIZARE

3.3.1. DETERMINAREA FORMEI MINIME DISJUNCTIVE

Ideea folosirii unor diagrame Venn speciale în scopul minimizării funcțiilor de comutare aparține lui B.W. Veitch. La scurt timp după propunerea făcută de Veitch, Karnaugh propune și el o formă modificată a diagramelor Venn, cu același scop. Astfel, au rezultat diagramele care poartă numele de *diagrame Veitch* sau *diagrame Karnaugh*. Aceste diagrame sunt utile pentru minimizarea funcțiilor booleene deoarece permit evidențierea cu ușurință a unor identități de forma:

$$a + ab = a$$

$$ab + a\bar{b} = a$$

$$a + \bar{a}b = a + b$$

La baza acestei metode stau ideile aduse de către Karnaugh și Veitch privind reprezentarea unei funcții de comutare pe o suprafață închisă desfășurată în plan, astfel încât plasând pe această suprafață termenii canonici ai unei funcții, aceștia să fie vecini (pe linie sau pe coloană) dacă diferă printr-o singură variabilă. Variabila prin care diferă apare într-unul din termeni sub formă directă, iar în celălalt sub formă negată. Se consideră vecine și compartimentele aflate la capetele opuse ale unei linii, respectiv coloane.

Pentru a realiza această distribuție a termenilor canonici, suprafața diagramei se împarte în domenii, astfel încât într-o jumătate de diagramă una din variabile să apară înscrisă direct, iar în cealaltă jumătate să apară variabila negată. În general, o diagramă Veitch pentru o funcție booleană de n variabile, se desenează sub formă de pătrat sau dreptunghi, împărțit în 2^n compartimente. Fiecare compartiment este rezervat unui termen canonic al funcției, respectiv uneia dintre cele 2^n n-uple ale funcției, sau vârfuri ale cubului n -dimensional din reprezentarea geometrică a funcției.

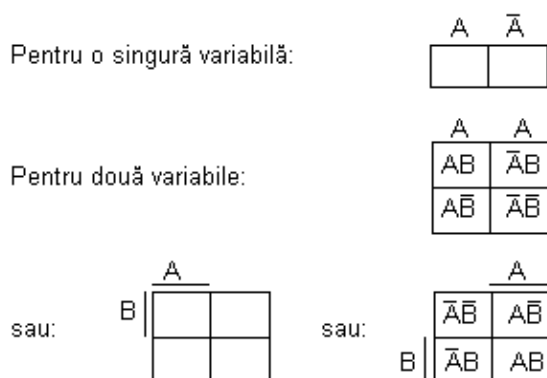


Figura 3.2. Diagrame pentru funcții cu una sau două variabile

Diagrama Veitch se notează fie indicând domeniul fiecărei variabile (figura 3.3.a), fie indicând pe linie și coloană n-uplul de zerouri și unități corespunzător unui compartiment din diagramă și ordinea variabilelor (figura 3.3.b). Prima notație se folosește în cazul funcțiilor date prin forma lor canonică sau normală, iar cea de a doua în cazul când se reprezintă funcții date prin tabelul de corespondență. Dacă funcțiile sunt exprimate prin indicii termenilor canonici, se poate nota fiecare compartiment cu indicele termenului canonic corespunzător, ținând cont de o anumită ordine a variabilelor (figura 3.3.c).

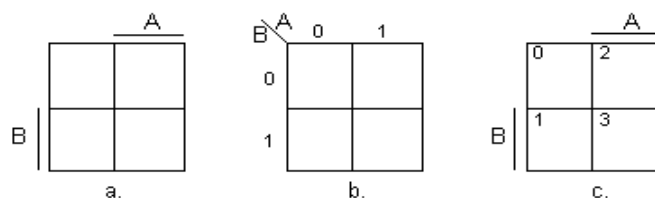


Figura 3.3. Moduri de notare ale diagramelor Veitch

Ne vom referi mai întâi la funcțiile complet definite. În acest caz funcția de două variabile, $f(A, B)$, dată în tabelul 3.1 se reprezintă ca în figura 3.4.

Tabelul 3.1

A	B	$f(A,B)$
0	0	0
0	1	1
1	0	1
1	1	0

sau:

Figure 3.4 shows the representation of the function $f(A,B)$ from the truth table. It consists of two 2x2 Veitch diagrams. The first diagram has top labels '0' and '1' and left labels '0' and '1'. The values in the compartments are 0, 1, 1, 0. The second diagram has top labels '0' and '2' and left labels '1' and '3'. The values in the compartments are 0, 1, 1, 0.

Figura 3.4. Reprezentarea funcției din tabelul

Pentru trei variabile, diagrama are 8 căsuțe, așa cum se vede și în figura 3.5.a.

Funcția: $f(A,B,C)=ABC+\bar{A}\bar{B}C+AB\bar{C}+\bar{A}\bar{B}\bar{C}$
se va reprezenta ca în figura 3.5.b.

Minimizarea funcțiilor de comutare

Pentru patru variabile se folosește o diagramă ca în figura 3.6, care reprezintă de fapt gruparea a două diagrame de trei variabile.

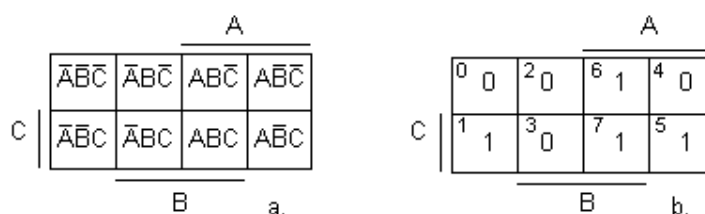


Figura 3.5. Reprezentarea unei funcții de trei variabile

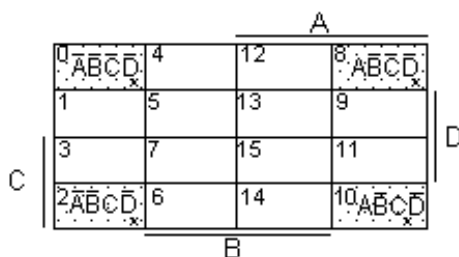


Figura 3.6. Diagrama Veitch pentru funcția de patru variabile

S-au notat cu “*” termenii vecini din diagramă (de exemplu, se consideră vecini: $\bar{A}\bar{B}\bar{C}\bar{D}$ și $A\bar{B}\bar{C}\bar{D}$).

O funcție booleană, dată sub formă canonică disjunctivă, poate fi reprezentată pe o diagramă Veitch marcând, de exemplu cu 1, compartimentele corespunzătoare termenilor canonici ai funcției.

Pe baza relației:

$$\bar{A}\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}\bar{D} = \bar{B}\bar{C}\bar{D}$$

doi termeni vecini din diagramă, pot fi înlocuiți în expresia funcției cu un termen elementar care va conține cu o variabilă mai puțin.

De asemenea, grupând patru termeni reciproc vecini dintr-o diagramă Veitch, $(\overline{A}\overline{B}\overline{C}D, \overline{A}B\overline{C}D, \overline{A}\overline{B}CD, \overline{A}BCD)$, ei se pot înlocui cu un singur termen elementar, care conține cu două variabile mai puțin:

$$\overline{A}\overline{B}\overline{C}D + \overline{A}B\overline{C}D + \overline{A}\overline{B}CD + \overline{A}BCD = \overline{A}\overline{B}D + \overline{A}BD = \overline{A}D$$

Termenii elementari pot fi scriși direct, urmărind pe diagramă toate variabilele comune ale termenilor canonici care îi compun.

În mod asemănător se pot grupa 2^r compartimente reciproc vecine, unde $r=1, \dots, n$, iar n reprezintă numărul de variabile.

În cazul reprezentării funcției de comutare se caută să se formeze grupe cât mai mari de compartimente vecine marcate cu 1. Modul în care se pot grupa compartimentele vecine sunt arătate în figura 3.7.

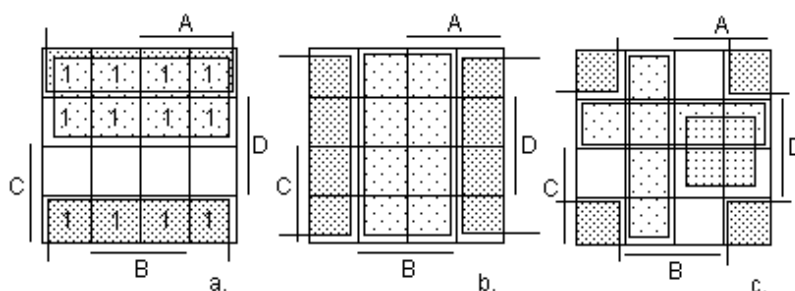


Figura 3.7. Modalități de grupare ale compartimentelor vecine marcate cu 1. a. și b - grupe de câte 8; c - grupe de câte 4

În reprezentarea geometrică, doi termeni canonici care diferă printr-o singură variabilă corespund la două noduri adiacente, deci definesc o latură a cubului n -dimensional. Din acest motiv se spune că două compartimente vecine sau adiacente pe diagrama Veitch reprezintă un subcub 1-dimensional. Un grup de patru compartimente vecine din diagrama Veitch corespunzătoare unei funcții de patru

variabile, reprezintă un subcub 2-dimensional (figura 3.7.c), iar cei patru termeni canonici pot fi înlocuiți cu partea lor comună, formată din două variabile. În cazul în care gruparea conține 8 compartimente vecine (figura 3.7.a, b.) se pot forma cuburi tridimensionale, iar termenii canonici corespunzători vor fi înlocuiți cu o singură variabilă în expresia funcției de patru variabile. Se observă că se pot forma subcuburi de diferite dimensiuni. Un subcub care nu este inclus într-un subcub de dimensiune mai mare se numește *implicant prim* al funcției date. Făcând suma booleană a tuturor implicantilor primi ai unei funcții date se obține o formă disjunctivă a acesteia, care în general este mult mai simplă decât forma canonică a funcției, dar nu este încă forma minimă a acesteia.

Termenii formei minime disjunctive ai funcției se află printre implicantii primi ai funcției. Nu toți implicantii primi apar în mod obligatoriu în această formă a funcției. Cei care apar în mod obligatoriu se numesc *implicanți primi esențiali*. Pentru a stabili dacă un implicant prim este esențial sau nu, se inspectează termenul respectiv în vederea detectării termenilor canonici componenți care nu mai sunt incluși în alți implicați. Dacă există astfel de termeni, implicantul prim este esențial (IPE), iar dacă nu, el este implicant prim neesențial (IPNE).

3.3.2. DETERMINAREA FORMEI MINIME CONJUNCTIVE

Există proceduri asemănătoare de determinare a FMC ca și pentru FMD, dar vom recurge la aceeași metodă ca și la determinarea forme canonice conjunctive. Se determină forma minimă disjunctivă a negatei funcției și apoi se neagă rezultatul obținut.

3.3.3. FOLOSIREA METODEI DIAGRAAMELOR LA MINIMIZAREA FUNCȚIILOR INCOMPLET DEFINITE

În rețelele de comutare a căror comportare este dată printr-o funcție de comutare parțială, cu precizarea condițiilor

“nu ține cont”, sinteza se efectuează considerând că rețeaua trebuie să realizeze acea funcție din clasa funcțiilor booleene corespondente funcției de comutare incomplet definite, care are cea mai simplă formă minimă disjunctivă sau conjunctivă. În acest caz, în diagramă apar unele elemente ale căror valoare nu este precizată și sunt marcate cu x.

Găsirea formei cele mai avantajoase corespunde practic în interpretarea compartimentelor "indiferente" în așa fel încât să contribuie la formarea unor contururi cât mai mari. Având în vedere că acești termeni (“nu ține cont”) nu sunt obligatorii pentru funcție ei nu vor determina esențialitatea unui implicant prim. Modul în care sunt date condițiile "nu ține cont" depinde de forma de exprimare a funcției. Când funcția este dată sub formă de expresie, condițiile "nu ține cont" se exprimă tot prin expresii logice.

3.3.4. MINIMIZAREA FUNCȚIILOR CARE AU MAI MULT DE PATRU VARIABLE

În general diagramele de minimizare pentru mai mult de patru variabile se construiesc folosind diagramele de 4 variabile ca module.

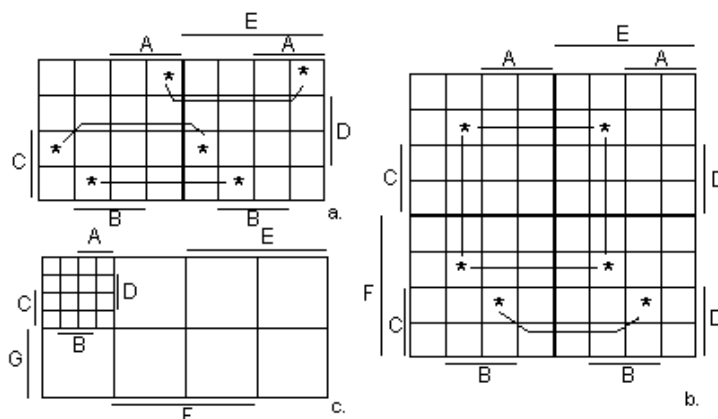


Figura 3.8. Diagrame Veitch cu mai mult de 4 variabile. a-5 variabile; b- 6 variabile; c- 7 variabile

De exemplu, pentru cinci variabile se pot folosi 2 diagrame de patru variabile. Din comoditate ele se pot desena alăturate, cu observația că 2 compartimente din diagrame vecine sunt vecine dacă ocupă aceeași poziție în cele două diagrame elementare (figura 3.8.a).

Pentru 6 variabile se folosesc 4 diagrame de câte 4 variabile, așa cum se arată în figura 3.8.b. Pentru 7 variabile se folosesc 8 diagrame de 4 variabile (figura 3.8.c). Se observă faptul că odată cu creșterea numărului de variabile diagramele devin din ce în ce mai complexe.

3.4. METODA QUINE – Mc. CLUSKEY

În cazul expresiilor cu un număr mai mare de variabile, când diagramele Veitch devin greu de folosit, se utilizează o metodă algebrică, numită *metoda Quine - Mc.Cluskey*.

Punctul de plecare în această metodă este tot forma canonică disjunctivă (FCD) sau conjunctivă (FCC) a funcției, respectiv tabelul de adevăr. Într-o primă etapă se caută implicații primi ai funcției, iar apoi se determină mulțimea minimă de implicații primi care acoperă termenii canonici ai funcției. Considerăm cazul în care se pleacă de la n -uplele aplicate în 1, sau de la mulțimea indicilor termenilor canonici prezenți în forma canonică a funcției.

La baza metodei stau următoarele observații:

1. Doi termeni canonici pot fi absorbiți de un termen cu o variabilă mai puțin dacă n -uplele corespondente sunt adiacente.

De exemplu:

$$\left. \begin{array}{l} (3) \quad 0 \ 1 \ 1 \\ (2) \quad 0 \ 1 \ 0 \end{array} \right\} \Rightarrow 0 \ 1 \ x \ (\bar{A}B)$$

Aceeași observație făcută asupra indicilor ne conduce la concluzia că doi termeni canonici pot fi absorbiți de un

termen cu o variabilă mai puțin dacă diferența indicilor este o putere a lui 2. Exponentul puterii indică poziția de unde va lipsi variabila.

2. Pentru ca două n-uple să fie adiacente ponderea lor trebuie să difere cu o unitate. Pentru exemplul anterior ponderile sunt:

$$0\ 1\ 1 \quad w = 2$$

$$0\ 1\ 0 \quad w = 1$$

Generalizând observația, de exemplu asupra indicilor, se poate spune că patru termeni canonici pot fi absorbiți de un singur termen cu 2 variabile mai puțin, dacă indicii acestora diferă între ei prin aceeași putere a lui 2. De exemplu:

$$\left. \begin{array}{l} (3) \ 0\ 1\ 1 \\ (2) \ 0\ 1\ 0 \\ (0) \ 0\ 0\ 0 \\ (1) \ 0\ 0\ 1 \end{array} \right\} \begin{array}{l} (3-2)=1=2^0 \\ (2-1)=1=2^0 \\ (0-1)=1=2^0 \\ (1-0)=1=2^0 \end{array} \left\{ \begin{array}{l} 0\ 1\ x \\ 0\ 0\ x \end{array} \right\} \Rightarrow \bar{A}$$

Având la bază observațiile de mai sus, se poate stabili un algoritm de aflare al implicanților primi pornind de la FCD sau tabelul de corespondență al funcției. Pentru acest lucru se vor parcurge următoarele etape:

- 1) Fiecare termen canonic este reprezentat sub formă de număr binar, prin n-uplul de 0 și 1 corespondent termenului respectiv.
- 2) Se împart termenii canonici ai funcției în grupe, în funcție de ponderea acestora, adică de numărul de 1-uri cuprinse în n-uplul respectiv.
- 3) Se aranjează grupele, pe o coloană, în ordinea crescătoare a ponderilor. Acest lucru este util deoarece doi termeni canonici se pot asocia, formând un subcub 1-dimensional, numai dacă fac parte din grupe ale căror ponderi diferă cu o unitate.

4) Se compară fiecare termen al unei grupe cu toți termenii grupei de pondere mai mare cu o unitate, în vederea stabilirii adiacenței, respectiv a posibilității de absorbție. Dacă numerele binare sunt adiacente cei doi termeni se pot asocia, formând un subcub 1-dimensional. Acesta este notat cu un număr binar care are pe poziția prin care cei doi termeni corespundenți diferă un x , ceea ce semnifică faptul că variabila corespondentă acelei poziții lipsește. Toți termenii care s-au putut asocia se marchează, iar termenul normal care reprezintă subcubul rezultat se înscrie pe o nouă coloană, într-un nou tabel. Toți termenii normali rezultați (subcuburile 1-dimensionale) în urma comparării a două grupe din coloana termenilor canonici, formează o nouă grupă în tabelul subcuburilor 1-dimensionale. Deci coloana subcuburilor 1-dimensionale conține, în cazul general, cu o grupă mai puțin decât coloana termenilor canonici (a subcuburilor 0-dimensionale).

5) Se compară fiecare termen al unei grupe din tabelul subcuburilor r -dimensionale cu toți termenii grupei cu pondere mai mare cu o unitate. Pentru ca doi asemenea termeni să se poată asocia trebuie ca în ambii termeni simbolurile x să fie pe aceleași poziții. Doi termeni care îndeplinesc această condiție și sunt adiacenți se asociază, formând un subcub $(r+1)$ -dimensional, care se notează cu un număr binar în care apare încă un x pe poziția prin care cei doi termeni diferă. Se bifează termenii care s-au asociat, iar subcubul $(r+1)$ -dimensional se înscrie într-un nou tabel, care în cazul general, are cu o grupă mai puțin decât tabelul subcuburilor r -dimensionale. Dacă se obține de mai multe ori un termen, acesta se consideră o singură dată.

6) Se mărește r cu o unitate și se repetă algoritmul până când subcuburile ultimului tabel nu se mai pot

asocia în scopul formării unui subcub de dimensiune superioară. În acest moment prima etapă a algoritmului este terminată.

Eventualii termeni rămași nemarcați la sfârșitul operației de comparare, devin *implicanți primi*.

3.5. MINIMIZAREA SISTEMELOR DE FUNCȚII BOOLEENE

În general circuitele de comutare combinaționale sunt circuite cu mai multe intrări și mai multe ieșiri (figura 3.9). Formele minime pentru un sistem de funcții booleene sunt acele expresii booleene disjunctive sau conjunctive în care apare un număr minim de termeni, respectiv factori normali diferiți, având un număr minim de literale.

Acestor forme le corespunde o rețea de comutare cu două niveluri, cu un număr minim de elemente logice, deci cu un preț minim.

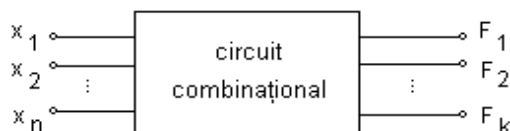


Figura 3.9. Schema bloc a unui circuit de comutare cu intrări și ieșiri multiple

Pentru a realiza cât mai economic un astfel de sistem, fiind date funcțiile F_1, F_2, \dots, F_k , se poate proceda în două moduri principal diferite:

a) Se face sinteza circuitului, tratând independent fiecare funcție. Deși mai simplă ca procedură, metoda nu asigură decât în cazuri rare obținerea unui circuit cu cost minim.

b) Se face sinteza globală a circuitului, caz în care se caută realizarea tuturor funcțiilor cu un set minim de circuite

comune. Se procedează la minimizarea sistemului de funcții booleene.

Unul dintre procedeele de *minimizare corelată a funcțiilor*, respectiv de găsire ai IP ai sistemului de funcții, constă în căutarea acestora printre IP ai funcțiilor separate (F_1, F_2, \dots, F_k) respectiv printre implicantii primi ai funcțiilor produs ($F_1F_2, F_1F_3, \dots, F_1F_k, \dots, F_{k-1}F_k, F_1F_2F_3, \dots, F_{k-2}F_{k-1}F_k, \dots, F_1F_2F_3\dots F_k$). Având acest set de implicantii primi, se calculează acoperirile posibile pentru fiecare dintre funcții iar apoi se alege cea mai avantajoasă combinație de acoperiri, din punct de vedere al costului, care reprezintă acoperirea minimală a sistemului.

Pentru a minimiza sistemul de funcții se caută în continuare un set minim de implicantii primi, care să acopere toate funcțiile. Se găsesc în literatură metode practice - de exemplu cu diagrame, metode tabelare sau metode pentru calculator. Cheltuiala de muncă fiind foarte mare, în practică se face un compromis între minimizarea independentă și cea globală. Acest lucru se justifică cu atât mai mult cu cât pentru creșterea fiabilității sistemului este de dorit o anumită redundanță în construcția sistemului.

Pentru obținerea acoperirii minimele cu această metodă, se parcurg etapele descrise mai jos cu ajutorul unui exemplu.

3.6. PROBLEME

3.6.1. Să se determine FMD și FMC pentru următoarele funcții:

- a) $f = p_3 + p_7 + p_8 + p_9 + p_{12} + p_{13} + p_{15};$
- b) $f = p_0 + p_2 + p_8 + p_{10} + p_{11};$
- c) $f = p_5 + p_7 + p_{13} + p_{10} + p_{11} + p_{14} + p_{15};$
- d) $f = p_0 + p_2 + p_4 + p_6 + p_8 + p_{10} + p_{12} + p_{14};$

e) $f = p_0 + p_1 + p_5 + p_7;$

f) $f = p_0 + p_2 + p_4 + p_6.$

3.6.2. Să se minimizeze funcțiile:

a) $f = \overline{a}b\overline{d} + ab\overline{c}d + abc\overline{d};$

b) $f = b\overline{c} + ad + \overline{a}c.$

3.6.3. Să se determine FMD, FMC și să se exprime utilizând funcția ȘI-NU următoarele funcții:

a) $f = p_5 + p_7 + p_9 + p_{10} + p_{14};$

b) $f = p_1 + p_2 + p_5 + p_7 + p_{15}.$

3.6.4. Să se minimizeze funcțiile logice asociate unui decodificator BCD/7 segmente utilizând termenii canonici redundanți.

3.6.5. Să se minimizeze și să se exprime cu ajutorul operatorului ȘI-NU:

a) $f = p_1 + p_2 + p_8 + p_{13};$

b) $f = p_0 + p_1 + p_2 + p_6 + p_8;$

c) $f = p_0 + p_2 + p_4;.$

3.6.6. Să se minimizeze și să se exprime cu ajutorul operatorului ȘI-NU următoarele funcții:

a) $f = p_8 + p_9 + p_{12} + p_{13};$

b) $f = p_3 + p_4 + p_5 + p_6 + p_{12} + p_{13} + p_{14};$

c) $f = p_1 + p_2 + p_5 + p_6 + p_9 + p_{10} + p_{13} + p_{14};$

d) $f = p_1 + p_3 + p_5 + p_7 + p_9 + p_{11} + p_{13} + p_{15}.$

3.6.7. Să se minimizeze funcțiile;

a) $f = p_0 + p_2 + p_3 + p_5 + p_7 + p_8 + p_{10} + p_{11} + p_{13} + p_{15};$

b) $f = p_0 + p_1 + p_2 + p_5 + p_9 + p_{10} + p_{11} + p_{12} + p_{13} + p_{14};$

- c) $f = \bar{a}bc + a\bar{b}c + abc$;
- d) $f = p_1 + p_2 + p_4 + p_5 + p_7 + p_8 + p_9 + p_{10} + p_{15}$;
- e) $f = p_0 + p_1 + p_2 + p_5 + p_8 + p_9 + p_{10} + p_{11} + p_{12} + p_{13} + p_{14}$;
- f) $f = p_0 + p_1 + p_3 + p_4 + p_5 + p_{12} + p_{13}$.

3.6.8. Să se minimizeze și să se exprime cu ajutorul operatorului ȘI-NU următoarele funcții:

- a) $f = p_3 + p_{11}$, termeni redundanți: $p_4 \div p_7$ și $p_{12} \div p_{15}$;
- b) $f = p_7 + p_{15}$, termeni redundanți: $p_0 \div p_3$ și $p_8 \div p_{11}$;
- c) $f = p_1 + p_5 + p_9 + p_{13}$, termeni redundanți: $p_2, p_3, p_6, p_7, p_{10}, p_{11}, p_{14}, p_{15}$;
- d) $f = p_3 + p_7 + p_{11} + p_{15}$, termeni redundanți: p_1, p_5, p_9, p_{13} ;
- e) $f = p_0 + p_2 + p_4 + p_6 + p_8 + p_{10} + p_{12} + p_{14}$, termeni redundanți: $p_1, p_3, p_5, p_7, p_9, p_{11}, p_{13}, p_{15}$;
- f) $f = p_1 + p_3 + p_5 + p_7 + p_9 + p_{11} + p_{13} + p_{15}$, termeni redundanți: $p_0, p_2, p_4, p_6, p_8, p_{10}, p_{12}, p_{14}$;
- g) $f = p_0 + p_1 + p_3 + p_4 + p_5 + p_6 + p_7 + p_8 + p_9$, termeni redundanți: $p_{10} \div p_{15}$;
- h) $f = p_0 + p_2 + p_6 + p_8$, termeni redundanți: $p_{10} \div p_{15}$;
- i) $f = p_0 + p_2 + p_3 + p_5 + p_7 + p_8 + p_9$, termeni redundanți: $p_{10} \div p_{15}$;
- j) $f = p_0 + p_2 + p_3 + p_5 + p_6 + p_8$, termeni redundanți: $p_{10} \div p_{15}$;
- k) $f = p_0 + p_4 + p_5 + p_6 + p_7 + p_{10}$, termeni redundanți: $p_2, p_3, p_8, p_9, p_{11}, p_{15}$;
- l) $f = p_0 + p_1 + p_2 + p_3 + p_4 + p_7 + p_8 + p_9$, termeni redundanți: $p_{10} \div p_{15}$;

m) $f = p_2 + p_3 + p_4 + p_5 + p_6 + p_8 + p_9$, termeni redundanți: $p_{10} \div p_{15}$;

n) $f = p_0 + p_4 + p_5 + p_6 + p_8 + p_9$, termeni redundanți: $p_{10} \div p_{15}$.

3.6.9. Să se determine FMD și FMC pentru următoarele funcții:

a) $f = p_0 + p_2 + p_3 + p_4 + p_6 + p_{14} + p_{16} + p_{18} + p_{19} + p_{20} + p_{22} + p_{24} + p_{26} + p_{30}$;

b) $f = p_0 + p_1 + p_2 + p_3 + p_7 + p_{14} + p_{15} + p_{22} + p_{23} + p_{29} + p_{31}$;

c) $f = p_5 + p_{10} + p_{11} + p_{30} + p_{31}$;

d) $f = p_0 + p_4 + p_{18} + p_{19} + p_{22} + p_{23} + p_{25} + p_{29}$;

e) $f = p_0 + p_2 + p_4 + p_6 + p_7 + p_8 + p_{10} + p_{11} + p_{12} + p_{13} + p_{14} + p_{16} + p_{18} + p_{19} + p_{29} + p_{30}$;

f) $f = p_0 + p_1 + p_2 + p_4 + p_5 + p_6 + p_{14} + p_{16} + p_{18} + p_{19} + p_{20} + p_{21} + p_{22} + p_{23} + p_{27}$;

g) $f = p_3 + p_{21} + p_{24} + p_{29} + p_{30}$;

h) $f = \bar{a}\bar{b}\bar{c} + \bar{b}cd + cde + \bar{a}bcd\bar{e} + abce$.

3.6.10. Să se minimizeze folosind metoda Quine – Mc Cluskey următoarele funcții:

a) $f = p_0 + p_2 + p_3 + p_5 + p_7 + p_8 + p_{10} + p_{11} + p_{13} + p_{15}$;

b) $f = p_0 + p_2 + p_4 + p_6$;

c) $f = p_0 + p_1 + p_4 + p_5 + p_{12} + p_{13}$;

d) $f = p_0 + p_4 + p_{18} + p_{19} + p_{22} + p_{23} + p_{25} + p_{29}$;

e) $f = p_0 + p_2 + p_4 + p_6 + p_7 + p_8 + p_{10} + p_{11} + p_{12} + p_{13} + p_{14} + p_{16} + p_{18} + p_{19} + p_{29} + p_{30}$;

f) $f = p_5 + p_{10} + p_{11} + p_{30} + p_{31}$;

Minimizarea funcțiilor de comutare

$$g) f = p_0 + p_1 + p_2 + p_3 + p_7 + p_{14} + p_{15} + p_{22} + p_{23} + p_{29} + p_{31};$$

$$h) f = p_0 + p_1 + p_2 + p_4 + p_5 + p_6 + p_{14} + p_{16} + p_{18} + p_{19} + p_{20} + p_{21} + p_{22} + p_{23} + p_{27}.$$

3.6.11. Să se implementeze cu un număr minim de porți ȘI-NU funcțiile:

$$a) f = p_3 + p_5 + p_7;$$

$$b) f = p_3 + p_7 + p_8 + p_9 + p_{12} + p_{13} + p_{15}.$$

3.6.12. Să se implementeze cu un număr minim de porți ȘI-NU funcția:

$$f = abc + cd + \bar{a}b + \bar{b}d + \bar{a}\bar{b}\bar{c}.$$

3.6.13. Să se implementeze cu un număr minim de porți ȘI-NU un sumator binar complet pe 2 biți.

3.6.14. Să se implementeze cu porți ȘI-NU un circuit de autoincidență cu trei intrări (este un circuit care oferă 1 logic la ieșire când cele trei variabile de intrare nu sunt identice, adică toate sunt 0 logic sau 1 logic).

3.6.15. Să se implementeze cu porți SAU-NU un circuit cu "vot majoritar" care are trei intrări de date (este un circuit care furnizează la ieșire valoarea logică a majorității variabilelor de intrare).

3.6.16. Să se implementeze cu porți SAU-EXCLUSIV funcția:

$$f = p_1 + p_2 + p_4 + p_7 + p_8 + p_{11} + p_{13} + p_{14}.$$

3.6.17. Să se implementeze cu porți ȘI-NU circuitul combinațional definit de următoarele funcții:

$$f_1 = p_0 + p_1 + p_2 + p_3 + p_{10} + p_{11} + p_{14} + p_{15};$$

$$f_2 = p_0 + p_1 + p_2;$$

$$f_3 = p_1 + p_2 + p_3 + p_5.$$

3.6.18. Să se minimizeze următoarele sisteme de funcții:

$$f_1 = p_0 + p_1 + p_2 + p_3 + p_{10} + p_{11} + p_{14} + p_{15}$$

a) $f_2 = p_0 + p_1 + p_5$

$$f_3 = p_1 + p_2 + p_3 + p_5;$$

$$f_1 = p_2 + p_3 + p_{10} + p_{13} + p_{15}$$

b) $f_2 = p_4 + p_5 + p_{10} + p_{13} + p_{15}$

$$f_3 = p_8 + p_9 + p_{10};$$

$$f_1 = p_0 + p_2 + p_3 + p_{10} + p_{13} + p_{15}$$

c) $f_2 = p_0 + p_4 + p_5 + p_9 + p_{10} + p_{13} + p_{15}$

$$f_3 = p_3 + p_9 + p_{10};$$

$$f_1 = \bar{a}bc + c\bar{d} + \bar{a}\bar{b}\bar{c}d + b\bar{d}$$

d) $f_2 = a\bar{c} + \bar{b}\bar{c}d + \bar{a}bcd$

$$f_3 = \bar{a}\bar{b}\bar{c} + ab\bar{c}d + \bar{c}\bar{d} + b\bar{d}.$$

3.6.19. Să se determine FMD și FMC pentru următoarele funcții:

a) $f = ac + b\bar{c};$

b) $f = ac + bd;$

c) $f = b\bar{c} + ad + \bar{a}c;$

d) $f = \overline{a \cdot b \cdot c}.$

3.6.20. Să se implementeze cu un număr minim de porți ȘI-NU:

$$f = p_0 + p_2 + p_3 + p_4 + p_6 + p_{14} + p_{16} + p_{18} + p_{19} + p_{20} + p_{22} + p_{24} + p_{26} + p_{30}.$$

Minimizarea funcțiilor de comutare

3.6.21. Să se implementeze cu un număr minim de porți SAU-NU funcția:

$$f = p_0 + p_3 + p_4 + p_5 + p_6 + p_7.$$

3.6.22. Să se implementeze un decodificator BCD/7 segmente:

- a) cu porți ȘI și SAU;
- b) cu porți ȘI-NU;
- c) cu porți ȘI-SAU-NU.

3.6.23. Simplificați următoarele funcții utilizând diagramele Veitch-Karnaugh:

- a) $X = \bar{a}bc + a\bar{b}\bar{c} + abc$;
- b) $Y = f(a, b, c) = \sum(1, 3, 5, 6, 7)$;
- c) $T = \bar{w}xy + w\bar{z} + xyz$;
- d) $P = f(w, x, y, z) = \sum(0, 2, 8, 10)$;
- e) $R = f(w, x, y, z) = \sum(1, 3, 4, 5, 6, 9, 11, 12, 13, 14)$.

3.6.24. Simplificați următoarele funcții utilizând diagrame Veitch-Karnaugh:

- a) $V = f(a, b, c, d) = \sum(2, 3, 4, 5, 13, 15) + \sum \text{t.r.}(8, 9, 10, 12)$;
- b) $Y = f(u, v, w, x) = \sum(1, 5, 7, 9, 13, 15) + \sum \text{t.r.}(8, 10, 11, 14)$;
- c) $V = f(r, s, t, u) = \sum(0, 2, 4, 8, 10, 14) + \sum \text{t.r.}(5, 6, 7, 12)$;
- d) $F = f(u, v, w, x, y) = \sum(0, 2, 8, 10, 16, 18, 24, 26)$;

- e) $H = f(a, b, c, d, e) = \sum(5, 7, 9, 12, 13, 14, 15, 20, 21, 22, 23, 25, 29, 31);$
- f) $M = f(v, w, x, y, z) = \sum(1, 3, 4, 6, 9, 11, 12, 14, 17, 19, 20, 22, 25, 27, 28, 30) + \sum \text{t.r.}(8, 10, 24, 26);$
- g) $J = f(a, b, c, d, e, f) = \sum(7, 12, 22, 23, 28, 34, 37, 38, 40, 42, 44, 46, 56, 58, 60, 62);$
- h) $K = f(r, s, t, u, v, w) = \sum(9, 11, 13, 15, 25, 27, 29, 31, 41, 43, 45, 47, 57, 59, 61, 63).$

3.6.25. Simplificați următoarele expresii folosind diagrame Veitch-Karnaugh:

- a) $T = \overline{a}\overline{b}\overline{c}de + \overline{a}b\overline{c}de + abcde + a\overline{b}\overline{c}de;$
- b) $P = \overline{v}\overline{w} + \overline{v}w\overline{y} + \overline{v}wz;$
- c) $G = \overline{y}z + \overline{w}x\overline{y} + \overline{w}xy + xyz.$

3.6.26. Identificați IPE și IPNE din următoarele expresii:

- a) $S = f(a, b, c, d) = \sum(1, 5, 7, 8, 9, 10, 11, 13, 15);$
- b) $T = f(a, b, c, d, e) = \sum(0, 4, 8, 9, 10, 11, 12, 13, 14, 15, 16, 20, 24, 28).$

3.6.27. Simplificați sistemele de funcții:

- a) $X = f(a, b, c) = \sum(1, 3, 7);$
 $Y = f(a, b, c) = \sum(2, 6, 7).$
- b) $X = f(a, b, c) = \sum(3, 4, 5, 7);$
 $Y = f(a, b, c) = \sum(3, 4, 6, 7).$

3.6.28. Să se minimizeze următorul sistem de funcții:

$$X = f(a, b, c) = \sum(1, 2, 3, 7)$$

Minimizarea funcțiilor de comutare

$$Y = f(a, b, c) = \sum(1, 2, 3, 6)$$

$$Z = f(a, b, c) = \sum(2, 4, 6).$$

3.6.29. Scrieți următoarele expresii sub forma unei sume de produse și apoi a unui produs de sume:

- a) $\bar{x}\bar{z} + \bar{y}\bar{z} + y\bar{z} + xyz$
- b) $(a + \bar{b} + d)(\bar{a} + b + d)(c + d)(\bar{c} + \bar{d});$
- c) $(\bar{a} + \bar{b} + d)(a + \bar{b} + \bar{c})(\bar{a} + b + d)(b + \bar{c} + \bar{d});$
- d) $(\bar{a} + \bar{b} + d)(\bar{a} + \bar{d})(a + b + \bar{d})(a + \bar{b} + c + d);$
- e) $\bar{w}y\bar{z} + v\bar{w}\bar{z} + v\bar{w}x + \bar{v}w\bar{z} + \bar{v}\bar{w}\bar{y}\bar{z}.$

3.6.30. Implementați funcțiile de la punctul 3.6.29. cu minim de porți ȘI și SAU.

3.6.31. Simplificați funcțiile următoare și implementați-le cu porți ȘI-NU:

- a) $F_1 = A\bar{C} + ACE + ACE\bar{C} + \bar{A}CD + \bar{A}\bar{D}\bar{E};$
- b) $F_2 = (\bar{B} + \bar{D})(\bar{A} + \bar{C} + D)(A + \bar{B} + \bar{C} + D)(\bar{A} + B + \bar{C} + \bar{D}).$

3.6.32. Implementați următoarele funcții cu porți ȘI-NU:

- a) $bd + bcd + \bar{a}\bar{b}\bar{c}\bar{d} + \bar{a}\bar{b}c\bar{d}$, cu cel mult 6 porți, fiecare având 3 intrări ;
- b) $(ab + \bar{a}\bar{b})(c\bar{d} + \bar{c}d)$, cu porți cu 2 intrări.

3.6.33. Simplificați funcția booleană F sub formă de sumă de produse folosind condițiile nu ține cont:

- a) $F = \bar{y} + \bar{x}\bar{z}$, termeni redundanți = $yz + xy$;
- b) $F = \bar{b}\bar{c}\bar{d} + b\bar{c}\bar{d} + ab\bar{c}\bar{d}$,

$$\text{termeni redundanți} = \bar{b}\bar{c}\bar{d} + \bar{a}\bar{b}\bar{c}\bar{d}.$$

3.6.34. Simplificați funcția booleană F folosind condițiile „nu ține cont” în (la) sumă de produse și produs de sume:

a) $F = \bar{a}\bar{b}\bar{d} + \bar{a}c\bar{d} + \bar{a}bc$,

$$\text{termeni redundanți} = \bar{a}\bar{b}\bar{c}\bar{d} + \bar{a}c\bar{d} + \bar{a}\bar{b}\bar{c}\bar{d};$$

c) $F = \bar{w}(\bar{x}y + \bar{x}\bar{y} + xyz) + \bar{x}\bar{z}(y + w)$,

$$\text{termeni redundanți} = \bar{w}x(\bar{y}z + y\bar{z}) + w yz$$

d) $F = ace + \bar{a}c\bar{d}\bar{e} + \bar{a}cde$,

$$\text{termeni redundanți} = \bar{d}\bar{e} + \bar{a}de + a\bar{d}\bar{e};$$

e) $F = \bar{b}d\bar{e} + \bar{a}be + \bar{b}c\bar{e} + \bar{a}b\bar{c}\bar{d}$,

$$\text{termeni redundanți} = \bar{b}d\bar{e} + \bar{c}d\bar{e}.$$

3.6.35. Implementați următoarele funcții folosind condițiile „nu ține cont”:

a) $F = \bar{a}\bar{b}\bar{c} + \bar{a}\bar{b}d + \bar{b}c\bar{d}$, cu cel mult 2 porți NOR,

$$\text{termeni redundanți: } \bar{a}bc + \bar{a}\bar{b}\bar{d};$$

b) $F = (a + d)(\bar{a} + b)(\bar{a} + \bar{c})$, cu cel mult 3 porți NAND;

c) $F = \bar{b}d + \bar{b}c + abcd$, cu porți NAND,

$$\text{termeni redundanți} = \bar{a}bd + \bar{a}\bar{b}\bar{c}\bar{d}.$$

3.6.36. Implementați următoarea funcție cu porți NAND sau NOR, folosind doar 4 porți. Sunt valabile doar intrările normale.

$$F = \bar{w}xz + \bar{w}yz + \bar{x}y\bar{z} + w\bar{x}y\bar{z}, \text{ termen redundant } w yz.$$

3.6.37. Următoarea expresie booleană: $be + \bar{b}\bar{d}\bar{e}$, este forma simplificată a expresiei:

$$\bar{a}be + bcde + \bar{b}\bar{c}\bar{d}e + \bar{a}\bar{b}\bar{d}\bar{e} + \bar{b}\bar{c}\bar{d}\bar{e}.$$

Există condiții „nu ține cont”? Dacă da, care sunt acestea?

3.6.38. Arătați trei posibilități de a exprima funcția:

$$F = \bar{a}\bar{b}\bar{d} + \bar{a}bcd + \bar{a}bd + ab\bar{c}d,$$

cu opt sau mai puține litere.

3.6.39. Găsiți forma simplificată în sumă de produse a funcției

$$F = f \cdot g,$$

unde f și g sunt date astfel:

$$f = w\bar{x}\bar{y} + \bar{y}z + \bar{w}y\bar{z} + \bar{x}y\bar{z};$$

$$g = (w + x + \bar{y} + \bar{z})(\bar{x} + \bar{y} + z)(\bar{w} + y + \bar{z}).$$