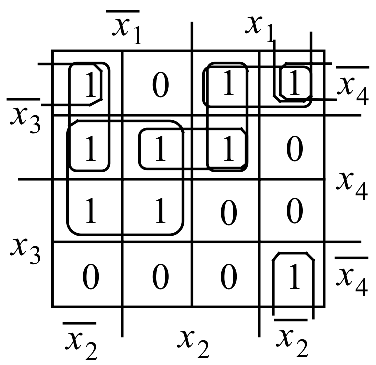
2.3. Minimizarea funcțiilor binare

- forma minimă a unei funcții binare
- criterii de minimizare

Metoda diagramelor Veitch-Karnaugh

- aplicabilă pentru funcții de până la 6 variabile
- **implicant prim** - orice grupare de mintermeni (compartimente) care nu poate fi inclusă într-o grupare mai mare
- **implicant prim esențial** - un implicant prim care conține cel puțin un compartiment care nu intră în componența altui implicant prim
- **implicant prim neesențial** - un implicant prim care nu este esențial
- mulțimea implicanților primi esențiali formează **nucleul** funcției
- **forma minimă** conține nucleul şi eventual o parte din implicanții primi neesențiali, pentru a acoperi toate compartimentele care conțin 1
- **exemplu**:



nucleul: $\overline{x_1}\,x_4$, $x_1\,\overline{x_2}\,\overline{x_4}$

implicanții primi neesențiali:

$x_2\,\overline{x_3}\,x_4$, $x_1\,x_2\,\overline{x_3}$, $x_1\,\overline{x_3}\,\overline{x_4}$

$\overline{x_1}\,\overline{x_2}\,\overline{x_3}$, $\overline{x_2}\,\overline{x_3}\,\overline{x_4}$

Fig. 2.4 *Lista implicanților primi pentru funcția dată*

$$f\left(x_1,x_2,x_3,x_4\right)=\overline{x}_1\cdot x_4+x_1\cdot\overline{x}_2\cdot\overline{x}_4+x_1\cdot x_2\cdot\overline{x}_3+\overline{x}_1\cdot\overline{x}_2\cdot\overline{x}_3=$$
$$=\overline{x}_1\cdot x_4+x_1\cdot\overline{x}_2\cdot\overline{x}_4+x_1\cdot x_2\cdot\overline{x}_3+\overline{x}_2\cdot\overline{x}_3\cdot\overline{x}_4$$

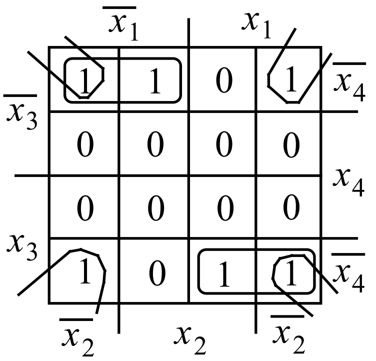
Funcția are 2 forme minime echivalente (cost de 5 porți, 15 intrări) Prin complementarea diagramei de mai sus, obținem o altă formă de scriere a funcției $f\left(x_1, x_2, x_3, x_4\right)$:

$$\bar{f}\left(x_1, x_2, x_3, x_4\right) = \bar{x}_1 \cdot x_2 \cdot \bar{x}_4 + \bar{x}_1 \cdot x_3 \cdot \bar{x}_4 + x_1 \cdot \bar{x}_2 \cdot x_4 + x_1 \cdot x_2 \cdot x_3$$

$$f\left(x_1, x_2, x_3, x_4\right) = \left(x_1 + \bar{x}_2 + x_4\right) \cdot \left(x_1 + \bar{x}_3 + x_4\right) \cdot \left(\bar{x}_1 + x_2 + \bar{x}_4\right) \cdot \left(\bar{x}_1 + \bar{x}_2 + \bar{x}_3\right)$$

Costul este de 5 porți şi 16 intrări, deci nici un avantaj.

• **exemplu**:



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

Fig. 2.5 *Minimizarea funcției reprezentate pe diagramă*

Costul reprezentării din figură este de 4 porți şi 11 intrări. Dacă evaluăm complementul funcției *f*:

$$\bar{f}\left(x_1, x_2, x_3, x_4\right) = x_4 + \bar{x}_1 \cdot x_2 \cdot x_3 + x_1 \cdot x_2 \cdot \bar{x}_3$$

$$f\left(x_1, x_2, x_3, x_4\right) = \bar{x}_4 \cdot \left(x_1 + \bar{x}_2 + \bar{x}_3\right) \cdot \left(\bar{x}_1 + \bar{x}_2 + x_3\right),$$

obținem un cost de 3 porți şi 9 intrări.

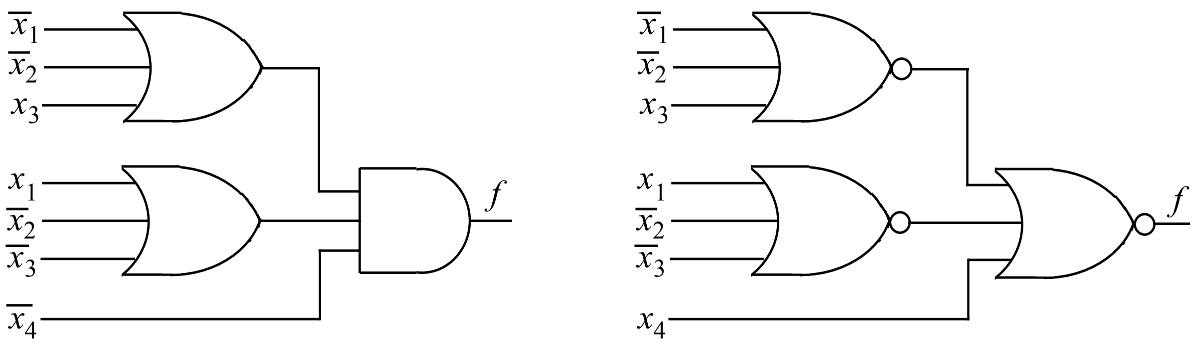


Fig. 2.6 *Două soluții de implementare a funcției* $f\left(x_1, x_2, x_3, x_4\right)$

- **<u>funcții incomplet specificate</u>:** valoarea funcției este indiferentă pentru anumite combinații ale variabilelor de intrare; se notează prin simbolul X - don't care.

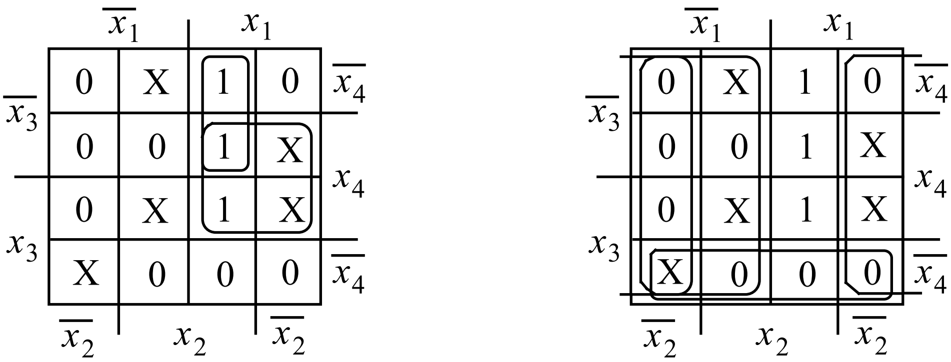


Fig. 2.7 *Minimizarea funcțiilor incomplet specificate*

Rezultă: $f(x_1, x_2, x_3, x_4) = x_1 \cdot x_4 + x_1 \cdot x_2 \cdot \overline{x_3}$. Prin evaluarea complementului se obține: $\overline{f}(x_1, x_2, x_3, x_4) = \overline{x_1} + \overline{x_2} + x_3 \cdot \overline{x_4}$, adică $f(x_1, x_2, x_3, x_4) = x_1 \cdot x_2 \cdot \overline{x_3 \cdot \overline{x_4}} = x_1 \cdot x_2 \cdot (\overline{x_3} + x_4)$.

- **<u>diagrame condensate</u>:** concepute în scopul minimizării unor funcții cu număr mai mare de variabile. Să considerăm funcția: $f(x_1, x_2, x_3) = x_1 \cdot x_2 + x_1 \cdot x_3 + \overline{x_1} \cdot \overline{x_2} \cdot \overline{x_3}$.

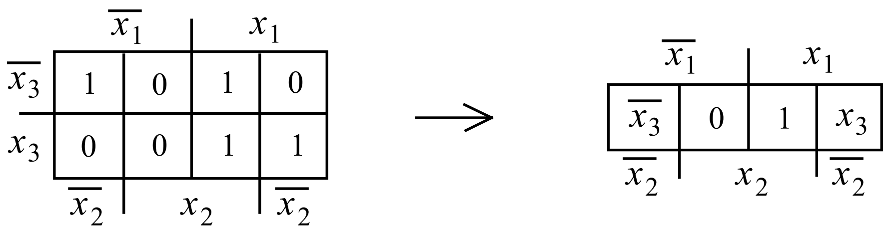


Fig. 2.8 *Condensarea diagramei Veitch - Karnaugh*

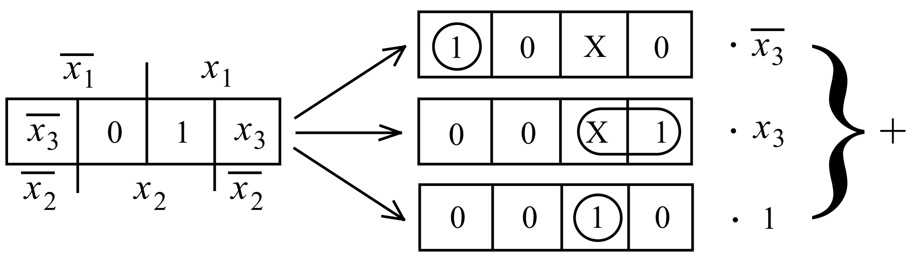


Fig. 2.9 *Descompunerea diagramei condensate*

Descompunerea diagramei condensate:

1. Într-un compartiment care conține o variabilă sau o expresie booleană se trece 1, în celelalte compartimente similare se trece 0, compartimentele care conțin 0 își păstrează valoarea, iar cele care conțin 1 capătă valoarea X (don't care). Se minimizează funcția reprezentată în diagramă şi variabila sau expresia booleană se aplică prin operatorul logic ŞI formei minime a funcției.

2. Se repetă punctul 1 pentru toate compartimentele care conțin variabile sau expresii boolene.

3. În toate compartimentele care conțin variabile sau expresii boolene se trece 0, iar compartimentele care conțineau 0 sau 1 își păstrează valorile.

4. Se aplică operatorul SAU expresiilor intermediare obținute la punctele 1, 2 şi 3.

Compartimentele care conțin simbolul "don't care" nu se modifică.

• **exemplu:**

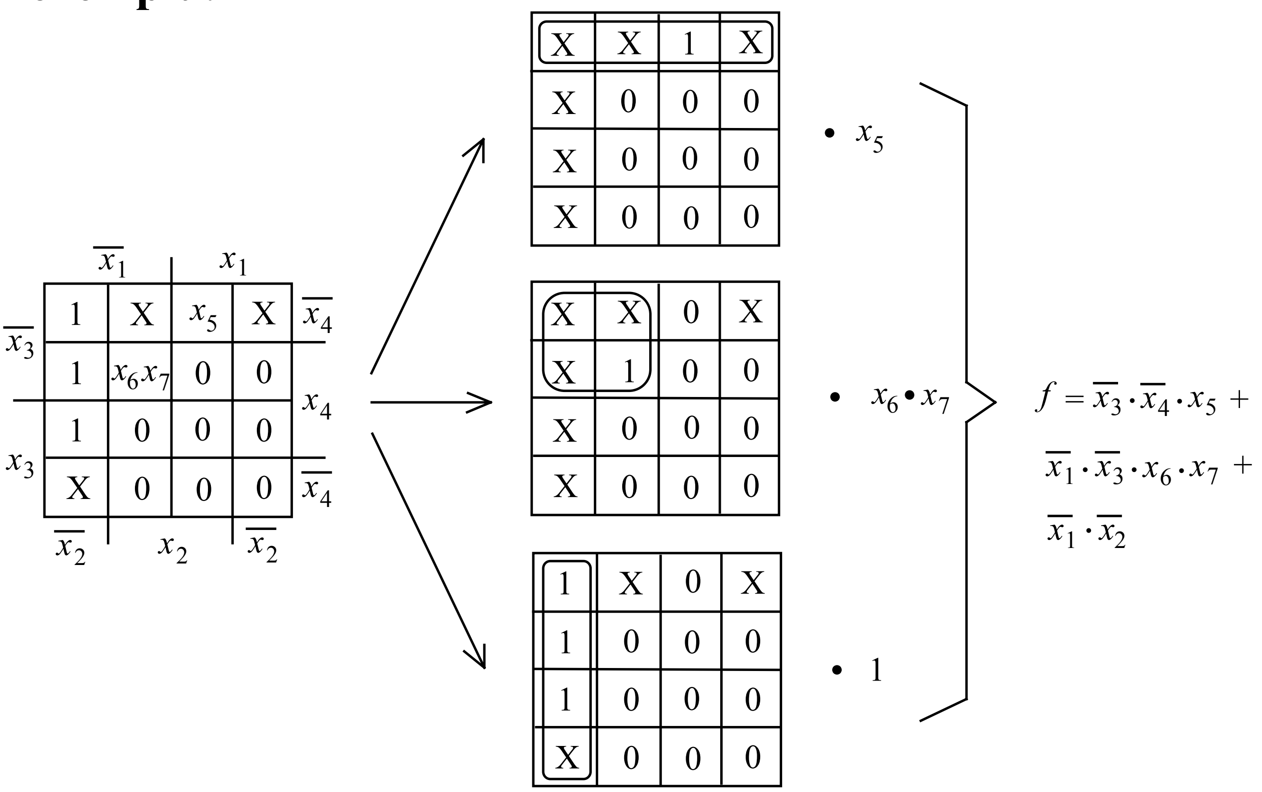


Fig. 2.10 *Descompunerea unei diagrame condensate*

<u>Metoda Quine - McCluskey</u>

- minimizarea funcțiilor binare cu număr mare de variabile
- minimizarea automată a funcțiilor binare pe calculator

- **<u>ponderea</u>** numărului binar x este un număr zecimal notat cu $w(x)$ care este egal cu numărul de biți de valoare 1 din reprezentarea binară a numărului x.
- **exemplu:** ponderea lui $x = 1011$ este $w(x) = 3$

<u>Descrierea algoritmului:</u>

1. Pentru funcția binară dată în formă canonică disjunctivă se face conversia din zecimal în binar a indicilor mintermenilor.

2. Se împart numerele binare care reprezintă indicii mintermenilor în grupuri, după ponderea fiecărui indice, în sens crescător al ponderilor. Grupurile se separă între ele prin linii orizontale şi rezultă tabelul 1.

3. Se compară fiecare număr binar din grupul de pondere i cu fiecare număr binar din grupul de pondere $i + 1$, pentru toate grupurile. Două numere binare formează o nouă grupare dacă diferă printr-un singur bit de acelaşi rang. Gruparea nouă se trece în tabelul 2.

4. De fiecare dată când ponderea i de la punctul 3 creşte cu o unitate, se trage o linie orizontală în tabelul 2. După completarea tabelului 2, se repetă comparațiile de la punctul 3 pentru elementele din tabel şi rezultă tabelul 3, ş.a.m.d.

5. Când se ajunge la un tabel care conține un singur număr binar, atunci termenul corespunzător este un implicant prim al funcției. Ceilalți implicanți primi se găsesc căutând în toate tabelele construite termenii care nu au fost folosiți în comparații.

• **exemplu:**

$$f\left(x_1,x_2,x_3\right)=x_1x_2x_3+x_1\bar{x}_2\bar{x}_3+\bar{x}_1x_2x_3+\bar{x}_1x_2\bar{x}_3+\bar{x}_1\bar{x}_2x_3+\bar{x}_1\bar{x}_2\bar{x}_3=$$
$$=P_7+P_4+P_3+P_2+P_1+P_0$$

Tabelul 1:

I	$(I)_2$	P_I
0	✓✓✓ 000	$\bar{x}_1\bar{x}_2\bar{x}_3$
2	✓ 010 ✓	$\bar{x}_1 x_2\bar{x}_3$
1	✓ 001 ✓	$\bar{x}_1\bar{x}_2 x_3$
4	100 ✓	$x_1\bar{x}_2\bar{x}_3$
3	✓ 011 ✓✓	$\bar{x}_1 x_2 x_3$
7	111 ✓	$x_1 x_2 x_3$

Pentru construcția tabelului 2, observăm că termenul de indice 0 poate fi grupat cu termenul de indice 2, rezultând reprezentarea: 0 – 0, o scriere echivalentă a relației: $\bar{x}_1\bar{x}_2\bar{x}_3+\bar{x}_1x_2\bar{x}_3=\bar{x}_1\bar{x}_3$. Semnele din tabelul 1 marchează termenii care au fost comparați.

Tabelul 2:

(0-2)	✓ 0-0	$\bar{x}_1\bar{x}_3$
(0-1)	✓ 00-	$\bar{x}_1\bar{x}_2$
(0-4)	-00	$\bar{x}_2\bar{x}_3$
(2-3)	01- ✓	$\bar{x}_1 x_2$
(1-3)	0-1 ✓	$\bar{x}_1 x_3$
(3-7)	-11	$x_2 x_3$

Tabelul 3:

(0-2-1-3)	0--	$\bar{x}_1$

$$f = \bar{x}_1+\bar{x}_2\bar{x}_3+x_2x_3$$

Forma minimă disjunctivă a funcției binare este disjuncția implicanților primi din toate tabelele construite, care nu au fost grupați: $f\left(x_1,x_2,x_3\right)=\bar{x}_1+\bar{x}_2\bar{x}_3+x_2x_3$.

Toți implicanții primi din expresia minimă a funcției f sunt esențiali și funcția are o formă minimă disjunctivă <u>unică</u>. Este posibil însă ca o funcție să aibă mai multe reprezentări minime.

- **exemplu:**

$$f\left(x_1,x_2,x_3,x_4\right)=P_{15}+P_{12}+P_{11}+P_9+P_8+P_7+P_6+P_4+P_3+P_0$$

Tabel 1:

0	√√ 0000	
8	√√ 1000	√
4	√√ 0100	√
12	1100	√√
9	√ 1001	√
6	√ 0110	√
3	√√ 0110	
11	√ 1011	√√
7	√ 0111	√√
15	1111	√√

Tabel 2:

(0-8)	√ -000	
(0-4)	√ 0-00	
(8-12)	1-00	√
(8-9)	100-	
(4-12)	-100	√
(4-6)	01-0	
(9-11)	10-1	
(6-7)	011-	
(3-11)	√ -011	
(3-7)	√ 0-11	
(11-15)	1-11	√
(7-15)	-111	√

Tabel 3:

(0-8-12)	--00
(0-4-12)	
(3-11-15)	--11
(3-7-15)	

f se poate scrie ca disjuncția implicanților primi:

$$f=\bar{x}_3\bar{x}_4+x_3x_4+x_1\bar{x}_2\bar{x}_3+\bar{x}_1x_2\bar{x}_4+x_1\bar{x}_2x_4+\bar{x}_1x_2x_3$$

dar nu este în formă minimă

Tabelul implicanților primi:

Mintermeni / Implicanți primi	P_{15}	P_{12}	P_{11}	P_9	P_8	P_7	P_6	P_4	P_3	P_0
	$x_1x_2x_3x_4$	$x_1x_2\bar{x}_3\bar{x}_4$	$x_1\bar{x}_2x_3x_4$	$x_1\bar{x}_2\bar{x}_3x_4$	$x_1\bar{x}_2\bar{x}_3\bar{x}_4$	$\bar{x}_1x_2x_3x_4$	$\bar{x}_1x_2x_3\bar{x}_4$	$\bar{x}_1x_2\bar{x}_3\bar{x}_4$	$\bar{x}_1\bar{x}_2x_3x_4$	$\bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4$
x_3x_4	√		√			√			√	
$\bar{x}_3\bar{x}_4$		√			√			√		√
$\bar{x}_1x_2x_3$						√	√			
$x_1\bar{x}_2x_4$			√	√						
$\bar{x}_1x_2\bar{x}_4$							√	√		
$x_1\bar{x}_2\bar{x}_3$				√	√					

Funcția are 4 forme minime:

$$f=x_3x_4+\bar{x}_3\bar{x}_4+\bar{x}_1x_2x_3+x_1\bar{x}_2x_4=x_3x_4+\bar{x}_3\bar{x}_4+\bar{x}_1x_2x_3+x_1\bar{x}_2\bar{x}_3=$$

$$=x_3x_4+\bar{x}_3\bar{x}_4+x_1\bar{x}_2x_4+\bar{x}_1x_2\bar{x}_4=x_3x_4+\bar{x}_3\bar{x}_4+\bar{x}_1x_2\bar{x}_4+x_1\bar{x}_2\bar{x}_3$$

Metoda consensurilor

- reprezentarea funcției într-o formă elementară disjunctivă
- reducerea dimensiunilor listelor de termeni prelucrați

- **consensul** a doi termeni produs P1 şi P2 care conțin aceeaşi variabilă, variabilă care este complementată într-unul din ei şi necomplementată în celălalt, se obține prin înlăturarea variabilei respective şi efectuarea produsului logic al celorlalte variabile din cei doi termeni produs.
- **exemple:** consensul termenilor $x_1x_2x_3$ şi $\bar{x}_3x_4x_5$ este $x_1x_2x_4x_5$.
  consensul termenilor $x_1x_2x_3$ şi $\bar{x}_2x_3x_4$ este $x_1x_3x_4$.

<u>Descrierea algoritmului:</u>

1. Se stabilesc perechile de termeni pentru care există consens şi se adaugă consensurile termenilor la forma elementară disjunctivă a funcției.

2. Se elimină termenii care sunt acoperiți de alți termeni existenți în expresia funcției.

3. Se repetă punctele 1 şi 2 până când nu se mai pot forma consensuri, sau toate consensurile care se pot forma sunt acoperite de termeni deja existenți.

- **exemplu:**

$$f\left(x_1,x_2,x_3,x_4\right)=\bar{x}_1x_2x_3x_4+\bar{x}_1\bar{x}_2x_3x_4+\bar{x}_1x_2x_3\bar{x}_4+\bar{x}_1\bar{x}_2x_3\bar{x}_4+x_1x_2x_3x_4=$$

$$=\bar{x}_1x_3x_4+\bar{x}_1x_2x_3x_4+\bar{x}_1\bar{x}_2x_3x_4+\bar{x}_1x_3\bar{x}_4+\bar{x}_1x_2x_3\bar{x}_4+\bar{x}_1\bar{x}_2x_3\bar{x}_4+x_1x_2x_3x_4=$$

$$=\bar{x}_1x_3x_4+\bar{x}_1x_3\bar{x}_4+x_1x_2x_3x_4=\bar{x}_1x_3+\bar{x}_1x_3x_4+\bar{x}_1x_3\bar{x}_4+x_1x_2x_3x_4=\bar{x}_1x_3+$$

$$+x_1x_2x_3x_4=x_2x_3x_4+\bar{x}_1x_3+x_1x_2x_3x_4=x_2x_3x_4+\bar{x}_1x_3.$$

Metoda ESPRESSO

- foloseşte o strategie euristică bazată pe aproximaţie, care evită construcţia listelor de termeni
- se aplică repetitiv şi generează de fiecare dată expresii boolene disjunctive mai simple decât precedentele lor
- fără a garanta forma minimă absolută, metoda oferă rezultate comparabile cu cele obţinute prin metoda Quine-McCluskey

Descrierea algoritmului:

1. Se aplică procedura EXPAND funcţiilor F şi $\overline{F}$, rezultatul fiind expresia redusă a funcţiei F , notată cu F^* .

2. Se aplică procedura IRREDUNDANT expresiilor F^* şi $\emptyset$, rezultând o nouă expresie redusă notată cu F^* .

3. Se aplică procedura ESSENTIAL_PRIMES expresiei F^* , iar rezultatul este expresia *E*.

4. Rezultatul procedurii anterioare se notează cu *D* şi se elimină din expresia F^* , iar rezultatul este notat tot cu F^* .

5. Se aplică procedura REDUCE expresiei notate cu F^* şi expresiei *D*, iar rezultatul este notat cu F^* .

6. Se aplică procedura EXPAND funcţiilor F^* şi $\overline{F}$, iar rezultatul se notează cu F^* .

7. Se aplică procedura IRREDUNDANT expresiilor F^* şi *D*, rezultând o nouă expresie redusă notată cu F^* .

8. Dacă expresia redusă F^* este mai simplă decât cea folosită la pasul 5, atunci se reia de la pasul 5; în caz contrar, se introduce o "perturbaţie" pentru a scoate procesul dintr-un eventual minim local. Dacă se reuşeşte acest lucru, atunci se reia de la pasul 5; în caz contrar, forma minimă a funcţiei este $F = F^* + E$.

- Procedura **EXPAND** înlocuieşte fiecare termen din expresia funcţiei F cu un set alcătuit din toţi termenii distincţi care formează o acoperire de rang imediat superior şi care nu intersectează pe $\overline{F}$:
- **exemplu:** $F = \overline{x}_1\overline{x}_2\overline{x}_3\overline{x}_4 + \overline{x}_1x_2\overline{x}_3 + x_1x_2x_4 + x_1x_3x_4$

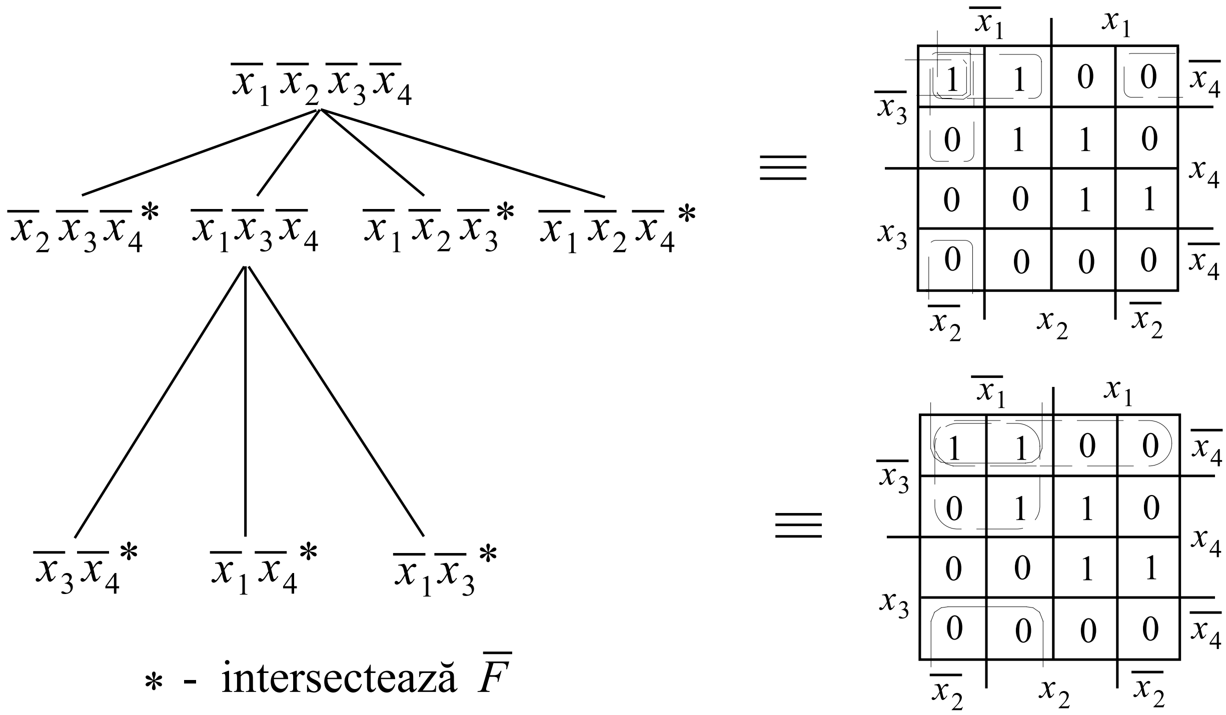


Fig. 2.11 *Exemplificarea procedurii EXPAND*

Complementul lui F este:

$$\overline{F} = \left(x_1 + x_2 + x_3 + x_4\right)\cdot\left(x_1 + \overline{x}_2 + x_3\right)\cdot\left(\overline{x}_1 + \overline{x}_2 + \overline{x}_4\right)\cdot\left(\overline{x}_1 + \overline{x}_3 + \overline{x}_4\right) =$$

$$= \left(x_1 + x_3 + \overline{x}_2x_4\right)\cdot\left(\overline{x}_1 + \overline{x}_4 + \overline{x}_2\overline{x}_3\right) =$$

$$= x_1\overline{x}_4 + x_1\overline{x}_2\overline{x}_3 + \overline{x}_1x_3 + x_3\overline{x}_4 + \overline{x}_1\overline{x}_2x_4 + \overline{x}_2\overline{x}_3x_4 .$$

Primul termen care formează o acoperire de rang imediat superior lui $\overline{x}_1\overline{x}_2\overline{x}_3\overline{x}_4$ este $\overline{x}_2\overline{x}_3\overline{x}_4$. El intersectează $\overline{F}$ deoarece:

$$\overline{x}_2\overline{x}_3\overline{x}_4 \cap \overline{F} = \left(\overline{x}_2\overline{x}_3\overline{x}_4\right)\cdot\left(x_1\overline{x}_4 + x_1\overline{x}_2\overline{x}_3 + \overline{x}_1x_3 + x_3\overline{x}_4 + \overline{x}_1\overline{x}_2x_4 + \overline{x}_2\overline{x}_3x_4\right) =$$

$= x_1\overline{x}_2\overline{x}_3\overline{x}_4 + x_1\overline{x}_2\overline{x}_3\overline{x}_4 + 0 + 0 + 0 + 0 \neq 0$. Funcţia F devine:

$$F^* = \overline{x}_1\overline{x}_3\overline{x}_4 + \overline{x}_1x_2\overline{x}_3 + x_1x_2x_4 + x_1x_3x_4$$

- Procedura **IRREDUNDANT** încearcă să găsească o sumă logică neredundantă, adică să elimine termenii redundanţi din expresia disjunctivă a funcţiei F^*. Conţine 3 subproceduri care se aplică consecutiv: **REDUNDANT, PARTIALLY_REDUNDANT** şi **MINIMAL_COVER**.

  Subprocedura **REDUNDANT** separă toţi implicanţii primi care ar putea fi redundanţi de cei care sunt <u>relativ esenţiali</u>.

  Un implicant prim este relativ esenţial faţă de o anumită expresie dacă şi numai dacă pentru acele valori ale variabilelor pentru care el are valoarea logică 1, expresia care nu conţine implicantul respectiv este diferită de 1.

  - **exemplu:** termenul $\bar{x}_1\bar{x}_3\bar{x}_4$ este relativ esenţial pentru funcţia $F^* = \bar{x}_1\bar{x}_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3 + x_1x_2x_4 + x_1x_3x_4$ deoarece pentru $x_1 = 0$ $x_3 = 0$ şi $x_4 = 0$ termenul $\bar{x}_1\bar{x}_3\bar{x}_4 = 1$, iar expresia funcţiei din care se elimină termenul $\bar{x}_1\bar{x}_3\bar{x}_4$ devine: $1 \cdot x_2 \cdot 1 = x_2 \neq 1$.

  Observăm că toţi implicanţii din expresia lui F^* sunt relativ esenţiali, deci expresia lui F^* nu se modifică:

$$F^* = \bar{x}_1\bar{x}_3\bar{x}_4 + \bar{x}_1x_2\bar{x}_3 + x_1x_2x_4 + x_1x_3x_4$$

  Celelalte 2 subproceduri vor fi discutate ulterior, dar nici ele nu modifică expresia lui F^* din exemplul considerat.

- Procedura **ESSENTIAL_PRIMES** stabileşte care sunt implicanţii primi esenţiali. În lipsa listei de mintermeni, un implicant prim este esenţial dacă nu este acoperit în totalitate de orice alt set de implicanţi primi ai funcţiei *F*, indiferent dacă aceştia sunt sau nu incluşi în F^*. Acoperirea unui implicant de către alţi implicanţi primi poate fi găsită din consensul lui cu fiecare din termenii din F^*.

Pentru că nu întotdeauna se pot face consensuri, avem în vedere şi un <u>consens generalizat</u> (de exemplu, consensul lui $x_1x_2x_4$ cu $x_1x_3x_4$ este $x_1x_2x_3x_4$, adică mintermenul acoperit de fiecare din cei doi termeni)

	$\overline{x}_1\overline{x}_3\overline{x}_4$	$\overline{x}_1 x_2 \overline{x}_3$	$x_1 x_2 x_4$	$x_1 x_3 x_4$
$\overline{x}_1\overline{x}_3\overline{x}_4$	√	$\overline{x}_1 x_2 \overline{x}_3\overline{x}_4$	0	0
$\overline{x}_1 x_2 \overline{x}_3$	$\overline{x}_1 x_2 \overline{x}_3\overline{x}_4$	√	$x_2 \overline{x}_3 x_4$	0
$x_1 x_2 x_4$	0	$x_2 \overline{x}_3 x_4$	√	$x_1 x_2 x_3 x_4$
$x_1 x_3 x_4$	0	0	$x_1 x_2 x_3 x_4$	√
$\sum_j C^j$	$\overline{x}_1 x_2 \overline{x}_3\overline{x}_4$	$x_2\overline{x}_3(\overline{x}_1+x_4)$	$x_2x_4(x_1+\overline{x}_3)$	$x_1 x_2 x_3 x_4$
$C^i \cap (\Sigma) = C^i$	Nu	Da	Da	Nu
Esențial	Da	Nu	Nu	Da

Fig. 2.12 *Identificarea implicanților esențiali prin consensuri*

Dacă suma consensurilor pe coloane acoperă termenul de pe linia respectivă, atunci acesta nu este esențial. Se recomandă o vizualizare a implicanților primi folosind echivalentul grafic al diagramei Veitch-Karnaugh.

Suma implicanților primi esențiali este: $E = \overline{x}_1\overline{x}_3\overline{x}_4 + x_1x_3x_4$, iar noua expresie a lui F^* devine: $F^* = F^* - E = \overline{x}_1x_2\overline{x}_3 + x_1x_2x_4$.

- Procedura **REDUCE** încearcă să micşoreze acoperirea fiecărui implicant din F^*, fără a modifica funcția F^*. Pentru fiecare consens c^i, expresia $F^*(i) = \left(F^* - c^i\right) + D$ acoperă F^* prin alți implicanți. Procedura înlocuieşte consensul c^i cu $c^i \cap \overline{F}^*(i)$, o porțiune din c^i care este acoperită numai de c^i.

$$F^*\left(\overline{x}_1x_2\overline{x}_3\right) = \left(\overline{x}_1x_2\overline{x}_3 + x_1x_2x_4 - \overline{x}_1x_2\overline{x}_3\right) + \overline{x}_1\overline{x}_3\overline{x}_4 + x_1x_3x_4 =$$

$$= x_1x_2x_4 + \overline{x}_1\overline{x}_3\overline{x}_4 + x_1x_3x_4 =$$

$$\overline{F}^*\left(\overline{x}_1x_2\overline{x}_3\right) = x_3\overline{x}_4 + \overline{x}_1x_3 + \overline{x}_1x_4 + x_1\overline{x}_4 + \overline{x}_2\overline{x}_3x_4 + x_1\overline{x}_2\overline{x}_3$$

$$\left(\overline{x}_1x_2\overline{x}_3\right) \cap \overline{F}^*\left(\overline{x}_1x_2\overline{x}_3\right) = 0 + 0 + \overline{x}_1x_2\overline{x}_3x_4 + 0 + 0 + 0 = \overline{x}_1x_2\overline{x}_3x_4$$

Se repetă calculele şi pentru consensul $x_1x_2x_4$ şi se obţine:

$$F^* = \overline{x}_1x_2\overline{x}_3x_4 + x_1x_2\overline{x}_3x_4$$

- Procedura **EXPAND** foloseşte tot complementul lui *F* stabilit la început şi se obţine: $F^* = x_2\overline{x}_3x_4 + \overline{x}_1x_2\overline{x}_3 + x_1x_2x_4$

- Primele două subproceduri ale procedurii **IRREDUNDANT** folosesc funcţia "don't care" *D* ca parametru. De data aceasta, la expresia lui F^* din subprocedura **REDUNDANT** se adaugă şi expresia lui *D*. Observăm că nici unul din implicanţii primi nu este relativ esenţial, deci: $F^* = x_2\overline{x}_3x_4 + \overline{x}_1x_2\overline{x}_3 + x_1x_2x_4$.

  Subprocedura **PARTIALLY_REDUNDANT** stabileşte care sunt implicanţii acoperiţi de expresia *D* + *RE* pentru a fi declaraţi <u>total redundanţi</u> şi eliminaţi. *RE* este suma implicanţilor relativ esenţiali. Implicanţii din F^* care nu sunt nici relativ esenţiali şi nici total redundanţi se numesc <u>parţial redundanţi</u> şi formează expresia R_p:

$$R_p = x_2\overline{x}_3x_4 + \overline{x}_1x_2\overline{x}_3 + x_1x_2x_4$$

  Subprocedura **MINIMAL_COVER** caută un set minim de implicanţi din R_p care acoperă expresia neacoperită de *RE* şi setul *D*. Pentru fiecare implicant c^i din R_p va exista o sumă de implicanţi G_{nc} astfel încât expresia $(R_p - G_{nc}) + D + RE$ va acoperi implicantul c^i dacă şi numai dacă un implicant este eliminat din G_{nc}. Rezultă $F^* = x_2\overline{x}_3x_4$ şi forma minimă :

$$F = x_2\overline{x}_3x_4 + \overline{x}_1\overline{x}_3\overline{x}_4 + x_1x_3x_4$$