

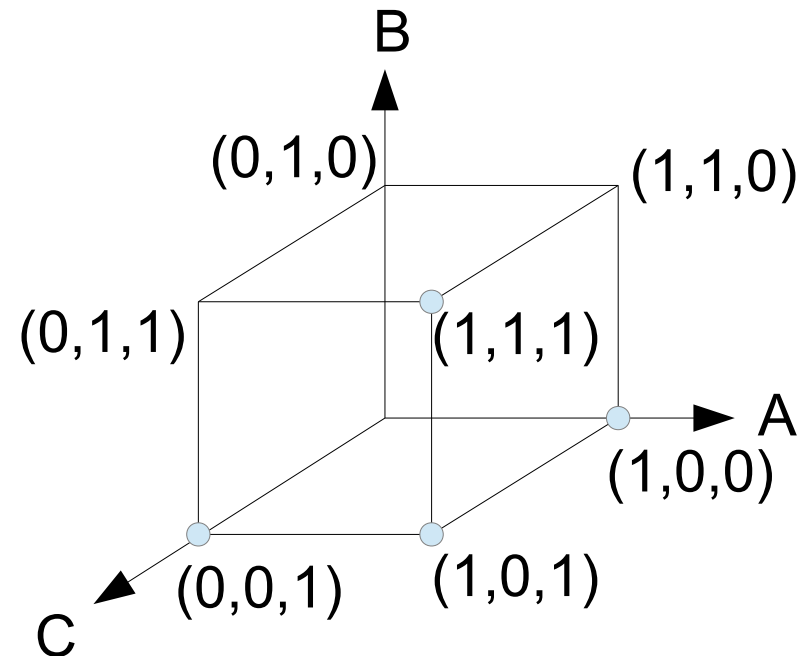
Reprezentarea funcțiilor logice

Expresii algebrice

- forma normală (canonică) disjunctivă:
 - $f(x_1, x_2, \dots, x_n) = A_1 + A_2 + \dots + A_n$, unde A_i este minterm
 - $Y(A,B,C) = \bar{A} * \bar{B} * C + A * \bar{B} * \bar{C} + A * \bar{B} * C + A * B * C$
- forma echivalențelor zecimale a mintermilor
 - $f(x_1, x_2, \dots, x_n) = \{i \mid A_i = 1\}$
 - $f(A,B,C) = \{1,4,5,7\}$
- forma normală (canonică) conjunctivă:
 - $f(x_1, x_2, \dots, x_n) = B_1 * B_2 * \dots * B_n$, unde B_i este maxterm
 - $Y(A,B,C) = (A+B+\bar{C}) * (\bar{A}+B+C) * (\bar{A} + B + \bar{C}) * (\bar{A} + \bar{B} + \bar{C})$

Cuburi booleene

- Greu de utilizat pentru funcții cu mai multe variabile
- Reprezentare grafică, important din punct de vedere istoric – a condus la diagrame Karnaugh
- Exemplu:
 - O funcție Y , cu trei variabile A , B și C



Tabel de adevăr

- Coloane cu combinațiile posibile ale valorilor + o coloană cu valoarea funcției
- Exemplu:
 - O funcție Y, cu trei variabile de intrare A, B și C

A	B	C	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	1

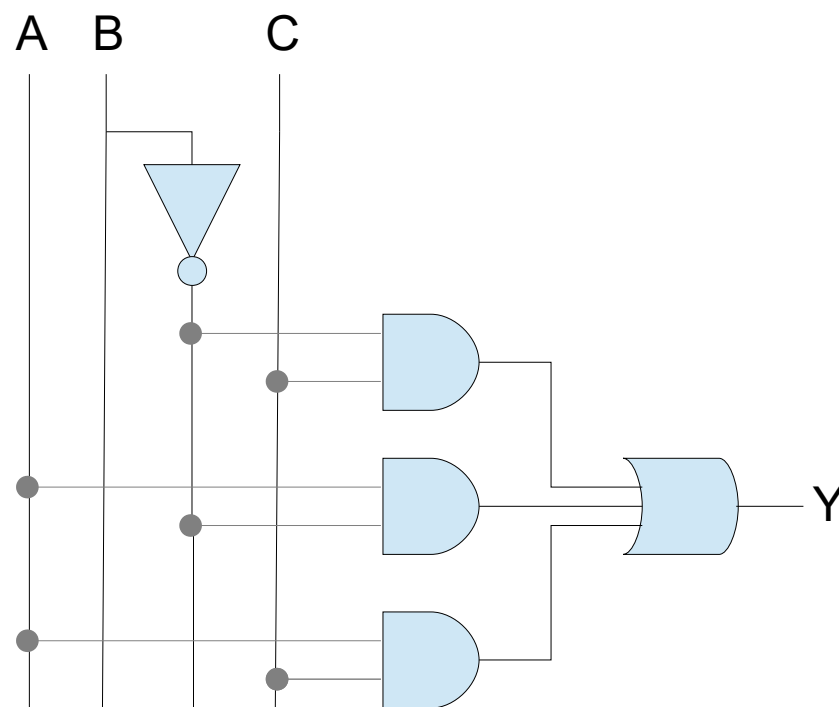
Diagrame Karnaugh

- O reprezentare tabelară a proiecției unui cub boolean
- “sistem de coordonate”, pe axe valorile sunt aranjate în codare Gray
- Exemplu:
 - O funcție Y , cu trei variabile de intrare A , B și C

A	BC				
		0 0	0 1	1 1	1 0
0		$\overline{A}\overline{B}\overline{C}$	$\overline{A}\overline{B}C$	$\overline{A}BC$	$\overline{A}B\overline{C}$
1		$A\overline{B}\overline{C}$	$A\overline{B}C$	ABC	$AB\overline{C}$

Circuite logice

- O reprezentare garfică, ușor de obținut atât din reprezentări algebrice, cât și din diagrame Karnaugh
- Exemplu:
 - O funcție Y , cu trei variabile de intrare A , B și C



Minimizarea funcțiilor logice

Minimizarea funcțiilor logice

- Obiectivul minimizării logice este reducerea mărimii reprezentării funcțiilor Booleene în oricare din formele de reprezentare sume de produse ori produse de sume.
- Se poate remarca faptul că oricare din cele două forme ale funcțiilor Booleene poate fi dedusă din cealaltă cu ajutorul legilor De Morgan iar această transformare păstrează numărul de termeni și de literalii.
- Categorii
 - Pe două nivele (minimizarea Karnaugh, Quine-McCluskey, minimizarea euristică)
 - Multinivel (metode simbolice)

Definiții

- Literal: variabilă sau forma negată a variabilei
- Implicant: este un grup ce se poate încercui într-o diagrama Karnaugh, un termen (chiar un minterm) care face parte din funcție logică
- Implicanți prim: scoaterea unui literal din implicant, grupul rezultat nu mai acopera doar valori de 1, ci conține și valori de 0
- Acoperire: o acoperire a unei funcții booleane este o multime de implicanți care conține toți termenii a funcției
- Acoperire minimă: acoperire cu număr minim de implicanți

A \ BC	0 0		0 1		1 1		1 0	
	0	0	0	1	1	1	0	0
0	1	1	0	0	0	0	0	0
1	1	1	1	1	0	0	0	0

Metode grafice (diagrame Karnaugh)

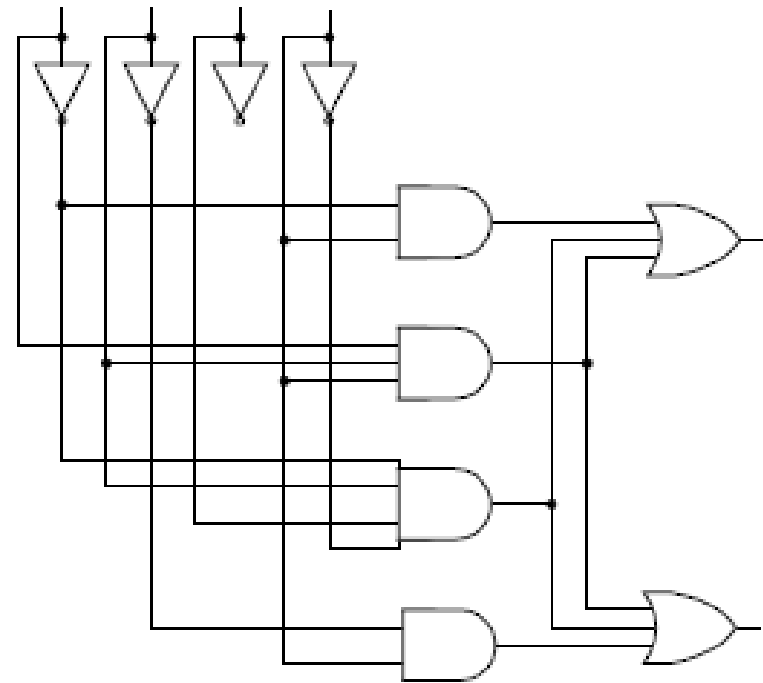
- Valorile în diagramele Karnaugh așa sunt plasate, încât două poziții diferă doar într-un singur bit
- Factorizarea mintermilor este mai ușor:
 - $\overline{A}BC + ABC = AC(\overline{B}+B)$

CD \ AB	0 0	0 1	1 1	1 0
0 0	0	0	0	0
0 1	1	1	1	0
1 1	1	1	1	0
1 0	0	1	0	0

CD \ AB	0 0	0 1	1 1	1 0
0 0	0	0	0	0
0 1	1	0	1	1
1 1	1	0	1	1
1 0	0	1	0	0

Metode grafice (diagrame Karnaugh)

- Minimizarea grafică a funcțiilor cu mai multe ieșiri:
 - se caută minimizarea unei perechi de funcții logice:
 - $Y_1 = \bar{A} * D + B * D + \bar{A} * B * C$
 - $Y_2 = A * D + B * D + \bar{A} * B * C * \bar{D}$.
 - Realizările individuale costă 29 de unități:
 - Y_1 costă 1 poartă SAU, 3 porți ȘI și 14 fire de legătură
 - Y_2 costă 1 poartă SAU, 3 porți ȘI și 11 fire de legătură



Metode grafice (diagrame Karnaugh)

- Minimizarea grafică a funcțiilor cu mai multe ieșiri:
 - Indetificarea părților comune a celor două funcții
 - $Y_1 = \bar{A} * D + A * B * D + \bar{A} * B * C * \bar{D}$
 - $Y_2 = A * D + A * B * D + \bar{A} * B * C * \bar{D}$
 - implementarea comună a funcțiilor necesită doar 2 porți SAU, 4 porți ȘI și 17 fire, per total costul fiind 23

Metoda Quine – McCluskey

- Minimizarea prin această metodă presupune parcurgerea pașilor următoare:
 - ordonarea mintermilor
 - determinarea implicațiilor primi
 - determinarea tabelului de acoperire a funcției.
 - calculul formal de determinare a tuturor soluțiilor funcției
- Exemplu
 - $F(A,B,C,D,E) = \{1,3,5,7,8,11,12,17,19,21,23,24,28\}$

Metoda Quine – McCluskey

- Exemplu
 - Ordonarea termenilor

1.	ponderea
1	1
8	
3	2
5	
12	
17	
24	
7	3
11	
19	
21	
28	
23	4

Metoda Quine – McCluskey

- Exemplu
 - determinarea implicaților primi
 - sunt comparate implicantii care figurează în grupuri consecutive
 - toate elementele grupului cu pondere mai mică sunt comparate cu toate elementele grupului cu pondere mai mare

1.	2.
1✓	1,3 (2)
8	
3✓	
5	
12	
17	
24	
7	
11	
19	
21	
28	
23	

Metoda Quine – McCluskey

- Exemplu

- determinarea implicațiilor primi

- condițiile de unificare a doi implicantți:

- diferența ponderilor este 1;
 - diferența valorilor decimale ale implicantților este egală cu o putere a lui 2 (adică 1, 2, 4, 8, 16, etc);
 - valoarea decimală a mintermului cu pondere mai mare trebuie să fie mai mare;
 - toate aceste condiții trebuie să se îndeplinească în aceeași moment.

1.	2.
1✓	1,3 (2)
8	
3✓	
5	
12	
17	
24	
7	
11	
19	
21	
28	
23	

Metoda Quine – McCluskey

- Exemplu
 - determinarea implicațiilor primi

1.	2.
1✓	1,3 (2)
8✓	1,5 (4)
3✓	1, 17 (16)
5✓	8, 12 (4)
12✓	8, 24 (16)
17✓	3, 7 (4)
24✓	3, 11 (8)
7✓	3, 19 (16)
11✓	5, 7 (2)
19✓	5, 21 (16)
21✓	12,28 (16)
28✓	17, 19 (2)
23✓	17, 21 (4)
	24, 28 (4)
	7, 23 (16)
	19, 23 (4)
	21,23 (2)

Metoda Quine – McCluskey

- Exemplu
 - determinarea implicațiilor primi

1.	2.	3.
1✓	1,3 (2)✓	1,3,5,7 (2,4)
8✓	1,5 (4)✓	1,3,17,19 (2,16)
3✓	1, 17 (16)✓	1,5,17,21 (4,16)
5✓	8, 12 (4)✓	8,12,24,28 (4,16)
12✓	8, 24 (16)✓	3,7,19,23 (4,16)
17✓	3, 7 (4)✓	5,7,21,23 (2,16)
24✓	3, 11 (8) (a)	17,19,21,23 (2,4)
7✓	3, 19 (16)✓	
11✓	5, 7 (2)✓	
19✓	5, 21 (16)✓	
21✓	12,28 (16)✓	
28✓	17, 19 (2)✓	
23✓	17, 21 (4)✓	
	24, 28 (4)✓	
	7, 23 (16)✓	
	19, 23 (4)✓	
	21,23 (2)✓	

Metoda Quine – McCluskey

- Example

– Consistent

1.	2.	3.	4.
1✓	1,3 (2)✓	1,3,5,7 (2,4)✓	1,3,5,7,17,19,21,23 (2,4,16). (c)
8✓	1,5 (4)✓	1,3,17,19 (2,16)✓	
3✓	1, 17 (16)✓	1,5,17,21 (4,16)✓	
5✓	8, 12 (4)✓	8,12,24,28 (4,16) (b)	
12✓	8, 24 (16)✓	3,7,19,23 (4,16)✓	
17✓	3, 7 (4)✓	5,7,21,23 (2,16)✓	
24✓	3, 11 (8) (a)	17,19,21,23 (2,4)✓	
7✓	3, 19 (16)✓		
11✓	5, 7 (4)✓		
19✓	5, 21 (16)✓		
21✓	12,28 (16)✓		
28✓	17, 19 (2)✓		
23✓	17, 21 (4)✓		
	24, 28 (4)✓		
	7, 23 (16)✓		
	19, 23 (4)✓		
	21,23 (2)✓		

Metoda Quine – McCluskey

- Exemplu
 - Determinarea unei acoperiri
 - Pe prima coloană se scriu implicantii găsiți, iar pe prima linie termenii funcției.
 - Pentru fiecare implicant prim se notează în tabel (matricea de incidenta) care sunt termenii acoperiți de el.

	1	3	5	7	8	11	12	17	19	21	23	24	28
a		x				x							
b					x		x					x	x
c	x	x	x	x				x	x	x	x		

Metoda Quine – McCluskey

- Exemplu
 - Determinarea unei acoperiri
 - Se poate construi o funcție complementară X pentru a determina implicantilor necesari:
 - $X = (c) (a + c) (c) (c) (b) (a) (b) (c) (c) (c) (c) (b) (b)$
 - $X = a b c$

	1	3	5	7	8	11	12	17	19	21	23	24	28
a		x				x							
b					x		x					x	x
c	x	x	x	x				x	x	x	x		

Metoda Quine – McCluskey

- Exemplu

- Transcierea implicantilor în variabilele funcției
- Primul implicant a 3,11 (8).
 - Se scrie un minterm în formă disjunctivă 3 este $\overline{A}BCDE$
 - Diferența calculată încă la gruparea mintermilor, arată care dintre variabilă a funcției se șterge: diferența fiind 8, se șterge variabila cu ponderea 8, adică variabila B
 - Se obține expresia implicantului: $\overline{A}CDE$.
- Al doilea implicant 8,12,24,28 (4,16)
 - se scrie echivalentul mintermului 8: $\overline{A}BCDE$.
 - se șterg variabilele cu ponderea 4 și 16: $\overline{B}DE$.
- A treilea implicant cu expresia 1,3,5,7,17,19,21,23 (2,4,16) va fi egal cu $\overline{B}E$.

Metoda Quine – McCluskey

- Exemplu
 - Rezultatul este: $F = \overline{A}BCDE + \overline{B}DE + \overline{B}E$.
- Metoda Quine-McCluskey se poate utiliza si cu termeni nedeterminati
 - In prima faza de cautare a implicantilor termenii nedeterminati sunt incluse in cautare (sunt considerate termeni obisnuiti)
 - In a doua faza de determinare a acoperiiri in matricea de incidenta lipsesc coloanele termenilor nedeterminati

Metoda Quine – McCluskey

- Utilizarea metodei in cazul minimizarii functiilor logice cu multiple iesiri
 - Se consideră funcția

$$f_2 = m_1 + m_5 + m_6 + m_7;$$

$$f_1 = m_1 + m_4 + m_5 + m_6;$$

$$f_0 = m_0 + m_2 + m_5 + m_6 + m_7;$$

Metoda Quine – McCluskey

Generarea implicantilor primi pentru această funcție vectorială se face printr-un procedeu foarte puțin diferit în comparație cu cazul funcțiilor scalare.

Tabelul 4a
Inițializarea metodei Quine-
McCluskey, aplicată funcției
vectoriale din Exemplul 7.

Implicantii de ordinul 0			
Grupa	Indicii	$x_2x_1x_0$	$f_2f_1f_0$
0	0	000	001√
	1	001	110√
1	2	010	001√
	4	100	010√
2	5	101	111
	6	110	111
3	7	111	101√

Doi implicantii ai unei funcții vectoriale sunt adiacenți dacă îndeplinesc condiția de adiacență a implicantilor funcțiilor scalare, aplicată părților de intrare, la care se adaugă condiția de compatibilitate a părții de ieșire. Aceasta constă dintr-o intersecție nevidă a respectivelor părți de ieșire.

Astfel, spre exemplu, implicantii de ordinul 0: (000 | 001) și (001 | 110), au partea de intrare adiacentă dar funcțiile sunt disjuncte. În concluzie, acești implicantii de ordinul 0 nu sunt adiacenți, așa cum se poate remarca și din tabelul 4a.

Metoda Quine – McCluskey

Tabelul 4b

Implicanții de ordinul 1 ai funcției
vectoriale din Exemplul 7.

Grupa	Indicii	$x_2x_1x_0$	$f_2f_1f_0$
0	0,2	0-0	001
	1,5	-01	110
	2,6	-10	001
1	4,5	10-	010
	4,6	1-0	010
2	5,7	1-1	101
	6,7	11-	101

Deîndată ce un implicant are un succesor de index superior, acesta este marcat (cu caracterul \surd).

Metoda Quine – McCluskey

Astfel, $(010 \mid 001)$ și $(110 \mid 111)$ sunt adiacenți dar este marcat doar primul

Tabelul 4c

Matricea de incidență a implicantilor primi ai funcției vectoriale din exemplul 7.

Implicanții primi			Termenii canonici												
Simbol	Binar	Indicii	f ₂				f ₁				f ₀				
			1	5	6	7	1	4	5	6	0	2	5	6	7
p	(101 111)	5		*					*				*		
q	(110 111)	6			*						*			*	
r	(0-0 001)	0,2										*	*		
s	(-01 110)	1,5	*	*			*		*						
t	(-10 001)	2,6										*		*	
u	(10- 010)	4,5						*	*						
v	(1-0 010)	4,6						*		*					
w	(1-1 101)	5,7		*		*							*		*
x	(11- 101)	6,7			*	*								*	*

pentru că succesorul lor este în fapt doar succesorul primului, spre exemplu.

Deoarece, exceptând cei doi implicantii din grupa 0, nici un alt implicant de ordin 0 nu are partea de ieșire complet reprezentată, rezultă că implicantii grupei 0 nu au succesori si vor fi implicantii primi, spre exemplu.

Metoda Quine – McCluskey

Implicanții primi ai funcției f sunt:

$(101 \mid 111), (110 \mid 111), (0-0 \mid 001), (-01 \mid 110), (-10 \mid 001), (10- \mid 010), (1-0 \mid 010), (1-1 \mid 101), (11- \mid 101).$

Se atașează fiecărui implicant prim din lista inițială câte un simbol distinct:

$p = (101 \mid 111), q = (110 \mid 111), r = (0-0 \mid 001), s = (-01 \mid 110), t = (-10 \mid 001),$
 $u = (10- \mid 010), v = (1-0 \mid 010), w = (1-1 \mid 101)$ și $x = (11- \mid 101).$

Matricea de incidență a acestei funcții vectoriale este prezentată în tabelul 4c.

Tabelul 4d

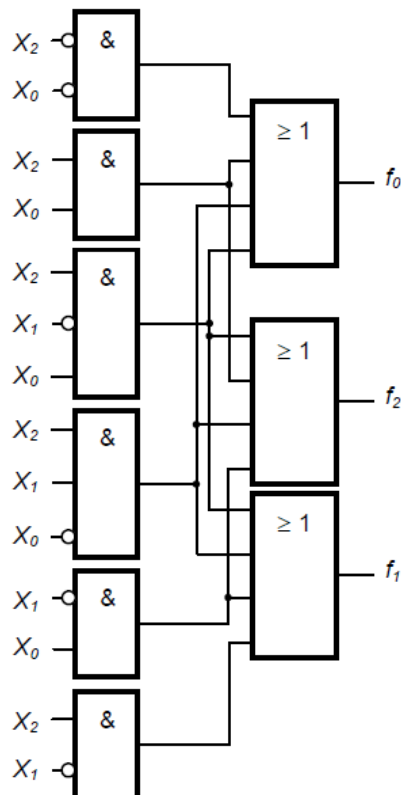
Matricea de incidență după îndepărtarea implicantilor primi p și q
 aparținând celor trei funcții f_2, f_1 și f_0 .

Implicanții primi			Termenii canonici						
simbol	binar	indicii	f_2		f_1		f_0		
			1	7	1	4	0	2	7
r	$(0-0 \mid 001)$	0,2					*	*	
s	$(-01 \mid 110)$	1	*		*				
t	$(-10 \mid 001)$	2						*	
u	$(10- \mid 010)$	4				*			
v	$(1-0 \mid 010)$	4				*			
w	$(1-1 \mid 101)$	7		*					*
x	$(11- \mid 101)$	7		*					*

Metoda Quine – McCluskey

Raționând după apartenența la funcții, implicantii primi $p = (101 \mid 111)$, având indicele 5, și $q = (110 \mid 111)$, având indicele 6, aparțin intersecției celor trei funcții $f_2 \cdot f_1 \cdot f_0$ și vor avea ramificații spre cele trei linii de ieșire reprezentând f_2 , f_1 și f_0 .
Cu alte cuvinte, implicantii primi p și q , aparțin oricărei soluții de acoperire minimă primă care poate fi alcătuită prin setul actual de implicantii primi.

Trebuie menționat faptul că odată acoperiți mintermii intersecției celor trei funcții (m_5 și m_6), coloanele acestora (5 și 6) vor fi îndepărtate din matricea de incidență împreună cu liniile celor doi implicantii primi.
Această transformare a matricei de incidență este ilustrată în tabelul 4d.



Metoda Quine – McCluskey

Considerând matricea de incidență rezultată după îndepărtarea implicantilor primi p și q , aparținând celor trei funcții f_2 , f_1 și f_0 , se remarcă faptul că implicantul prim s aparține intersecției funcțiilor f_2 și f_1 .

Acest implicant prim acoperă termenul canonic m_1 , iar acest termen canonic este chiar intersecția celor două funcții, $m_1 = f_2 \cdot f_1$.

Implicantul prim s va avea ramificații atât spre linia de ieșire a funcției f_2 , cât și spre linia de ieșire a funcției f_1 . În această manieră se dovedește că implicantul prim s va aparține oricărei acoperiri minime prime a acestei funcții vectoriale.

Tabelul 4e

Matricea de incidență după îndepărtarea
implicantului prim s și a coloanelor m_1 .

Implicantii primi			Termenii canonici					
			f_2	f_1	f_0			
simbol	binar	indicii	7	4	0	2	7	
r	(0-0 001)	0,2			*	*		
t	(-10 001)	2				*		
u	(10- 010)	4		*				
v	(1-0 010)	4		*				
w	(1-1 101)	7	*				*	
x	(11- 101)	7	*				*	

Metoda Quine – McCluskey

Tabelul 4e prezintă matricea de incidență după excluderea implicantului prim s și a coloanelor corespunzătoare termenului canonic m_1 , atât din dreptul funcției f_2 cât și din dreptul funcției f_1 .

Matricea de incidență din tabelul 4e a scăzut considerabil în complexitate.

Astfel, termenul canonic m_7 aparținând funcției f_2 și funcției f_0 poate fi acoperit de implicantul prim w sau de implicantul prim x , unul dintre aceștia.

În mod analog, termenul canonic m_4 , aparținând funcției f_1 , poate fi acoperit de implicantul prim u sau de implicantul prim v , oricare dintre aceștia.

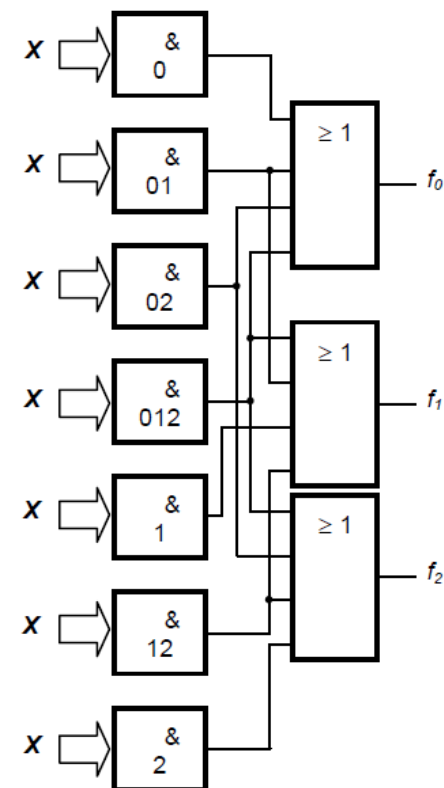
Implicantul prim r domină implicantul prim t , în consecință acesta din urmă poate fi îndepărtat din matrice.

Acoperirea primă minimă a funcției vectoriale arată astfel:

$(101|111)$, $(110|111)$, $(-01|110)$, $(1-1|101)$, $(10-|010)$, $(0-0|001)$.

Implementarea prin circuite logice ȘI și SAU a descrierii simbolice corespunzătoare acoperirii prime minime a acestei funcții vectoriale este prezentată în figura 5.

◇



Metoda Quine – McCluskey

În cazul unei funcții vectoriale, având trei componente scalare, de forma:

$$F: B^n \rightarrow B^3,$$

structura minimizată generică a funcției F arată ca în figura 6.

Toate porțile $\mathcal{S}I$ sunt generice și sunt indexate.

Indexul unei porți $\mathcal{S}I$ este înscris, în figura 6, la baza simbolului porții respective.

Circuitele $\mathcal{S}I$ sunt generice în sensul că pot fi mai multe circuite având același index dar realizând fiecare, în parte, alt termen produs. Doi termeni produs, având același index, pot diferi atât prin paritatea variabilelor cât și prin faptul că liniile lor de intrare pot defini două submulțimi distincte ale domeniului de definiție.

Codul indexului unui circuit $\mathcal{S}I$ este determinat în raport cu ramificațiile pe care le are linia de ieșire a respectivului circuit. Astfel, un circuit $\mathcal{S}I$ având indexul 012 va avea linia de ieșire ramificată spre circuitele SAU aparținând liniilor funcțiilor scalare f_0 , f_1 și f_2 .

Metoda Quine – McCluskey

În mod similar, o poartă ȘI având indexul 12 își ramifică liniile de ieșire spre porțile SAU ale funcțiilor scalare f_1 și f_2 . În timp ce o poartă ȘI cu indexul 0 se va conecta doar la poarta SAU a cărei linie de ieșire este etichetată prin f_0 , spre exemplu.

Porțile indexate printr-o singură cifră (0, 1 sau 2, în cazul acesta) nu au linia de ieșire ramificată dar sunt generice. Aceste porți se conectează, fiecare, doar intrarea unei porți SAU a unei singure funcții scalare (respectiv f_0 , f_1 sau f_2).

Analog, sunt definibile structurile generice pentru funcțiile multi-ieșire având patru ori mai multe linii scalare de ieșire.

Minimizare euristică

- În cazul circuitelor reale, mărimea de ordin a numărul variabilelor de intrare a unei funcții booleene poate să ajungă la un ordin de 1000 sau și mai mare.
- Minimizarea euristică nu caută o acoperire minimă, ci o acoperire destul de bună.
- Minimizarea funcțiilor se face iterativ, prin aplicarea operatorilor de expandare, neredundanță și reducere

Minimizare euristică

- Exemplu
- Se pronește de la o acoperire oarecare, poate fi și definiția funcției

	00	01	11	10		abcd	F	implicant
00	1	1				0-0-	1	a
01	1	1	1	1		0-10	1	b
11			1	1		1001	1	c
10	1	1	1	1		1101	1	d
						1-1-	1	e

Minimizare euristică

- Exemplu
 - Expandare, acest operator calculează o acoperire primă și minimală în raport cu conținerea într-un singur implicant. Implicanții care formează acoperirea sunt procesați rând pe rând. Fiecare implicant care nu este prim este expandat într-un implicant prim, adică este înlocuit printr-un implicant prim care-l conține. Consecutiv, toți implicanții din acoperire (cei încă ne-procesați) care sunt conținuți în acest implicant prim sunt îndepărtați.

Minimizare euristică

- Exemplu
- După expandare

	00	01	11	10		abcd	F	implicant
00	1	1				0-0-	1	a
01	1	1	1	1		--10	1	b
11			1	1		--01	1	c
10	1	1	1	1		1-0-	1	d
						1-1-	1	e

Minimizare euristică

- Minimizare euristică
 - Exemplu
 - iredundant, are drept acțiune calculul unei acoperiri iredundante. Operatorul alege un subset al implicantilor astfel încât nici un implicant al acestui subset nu este acoperit de ceilalți implicanți din subset.

Minimizare euristică

- Exemplu
- După iredundant

	00	01	11	10		abcd	F	implicant
00	1	1				0-0-	1	a
01	1	1	1	1		--10	1	b
11			1	1		--01	1	c
10	1	1	1	1		1-0-	1	d
						1-1-	1	e

Minimizare euristică

- Minimizare euristică
 - Exemplu
 - Reduce, este un operator care realizează transformarea unei acoperiri oarecare, într-o acoperire ne-primă dar de aceeași cardinalitate cu cea inițială. Implicanții sunt procesați unul câte unul. Acest operator încearcă să înlocuiască fiecare implicant printr-un altul care este conținut în acesta, cu condiția că implicanții reduși împreună cu cei rămași continuă să acopere funcția.

Minimizare euristică

- Exemplu
- După reducere

	00	01	11	10		abcd	F	implicant
00	1	1				0-0-	1	a
01	1	1	1	1		0-10	1	b
11			1	1		1-01	1	e
10	1	1	1	1		1-01	1	d
						1-1-	1	e

Minimizare euristică

- Exemplu
- Încă o operație de expandare

	00	01	11	10		abcd	F	implicant
00	1	1				0-0-	1	a
01	1	1	1	1		--10	1	b
11			1	1		--01	1	c
10	1	1	1	1		1--1	1	d
						1-1-	1	e

Minimizare euristică

- Exemplu
- Încă o operație de iredundant

	00	01	11	10		abcd	F	implicant
00	1	1				0-0-	1	a
01	1	1	1	1		--10	1	b
11			1	1		--01	4	e
10	1	1	1	1		1--1	1	d
						1-1-	4	e

Unelte pentru minimizarea funcțiilor

Unelte pentru minimizarea funcțiilor

- Programul espresso