Universitatea Politehnica Timişoara
P-ţa Victoriei nr. 2
RO 300006 - Timişoara
Tel: +4 0256 403000
Fax: +4 0256 403021
rector@rectorat.upt.ro
www.upt.ro

Logică digitală

-Curs 4-

Reprezentarea functiilor

-2021-

Algebra booleană și logica digitală

- Forma canonică;
- Forma standard;
- Aspecte legate de implementarea funcțiilor booleene cu porților logice;
- Hărți Karnaugh

Funcții booleene

- Tabel de adevăr prin care este specificată

x	y	z	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

mintermi

Un **minterm** este o funcţie elementară de n variabile notată m_i^n unde n indică numărul de variabile ale funcţiei iar i este echivalentul zecimal al **mintermului.**

E_z	X_2	X_1	X_0	m_0^3	m_1^3	m_2^3	m_3^3	m_4^3	m_5^3	m_6^3	m_7^3
0	0	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0	0
2	0	1	0	0	0	1	0	0	0	0	0
3	0	1	1	0	0	0	1	0	0	0	0
4	1	0	0	0	0	0	0	1	0	0	0
5	1	0	1	0	0	0	0	0	1	0	0
6	1	1	0	0	0	0	0	0	0	1	0
7	1	1	1	0	0	0	0	0	0	0	1

mintermi

A	B	C	minterms	
0	0	0	$\overline{A}\ \overline{B}\ \overline{C}$	m0
0	0	1	$\overline{A}\ \overline{B}\ C$	m1
0	1	0	$\overline{A}\ B\ \overline{C}$	m2
0	1	1	$\overline{A}\ B\ C$	m3
1	0	0	$A\ \overline{B}\ \overline{C}$	m4
1	0	1	$A\ \overline{B}\ C$	m5
1	1	0	$A\ B\ \overline{C}$	m6
1	1	1	$A\ B\ C$	m7

Sumă de mintermi

- funcţia minterm $m_2^3(X_0, X_1, X_2) = \overline{X_2} \cdot X_1 \cdot \overline{X_0}$

are expresia 1 dacă X_2 =0, X_1 =1 şi X_0 =0, și valoarea 0 în rest;

- orice funcţie booleană de n variabile poate fi

reprezentată ca **sumă logică de funcţii minterm**

$$f(X_{0,}X_1.....X_{n-1}) = \sum_{i \in K} m_i^n$$

Sumă de mintermi (SOP)

F = xy + xy'z + x'yz

$$F = \sum(3,5,6,7)$$

$$F = m_3^3 + m_5^3 + m_6^3 + m_7^3$$

x	y	z	F
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Formă canonică disjunctivă

$$F = \sum (3, 5, 6, 7)$$

$$F = m_3^3 + m_5^3 + m_6^3 + m_7^3$$

- **forma canonică disjunctivă** a funcţiei:- termenii produs logic ai funcţiei conţin **toate** variabilele funcţiei, între termeni realizându-se operaţia **SAU** (disjuncție).

Maxterm

- maxterm este o functie elementară de n variabile notate M_i^n unde i este echivalentul zecimal al n-uplului funcţiei, aplicat in „0”, interpretat ca un număr binar pe n poziţii.
- Functiei maxterm îi corespunde o expresie de n variabile în formă M_i^n care în urma evaluării pentru toate n-uplurile, ia aceeasi valoare ca si $M_i^n = \overline{m_i^n}$.

2.Reprezentarea funcţiilor de comutaţie

E_z	X_2	X_1	X_0	M_0^3	M_1^3	M_2^3	M_3^3	M_4^3	M_5^3	M_6^3	M_7^3
0	0	0	0	0	1	1	1	1	1	1	1
1	0	0	1	1	0	1	1	1	1	1	1
2	0	1	0	1	1	0	1	1	1	1	1
3	0	1	1	1	1	1	0	1	1	1	1
4	1	0	0	1	1	1	1	0	1	1	1
5	1	0	1	1	1	1	1	1	0	1	1
6	1	1	0	1	1	1	1	1	1	0	1
7	1	1	1	1	1	1	1	1	1	1	0

Funcţiile maxterm de 3 variabile

2.Reprezentarea funcţiilor de comutaţie

- Funcţia maxterm M_3^3 de exemplu are expresia $M_3^3 = X_2 + \overline{X_1} + \overline{X_0}$ =0 pentru $X_2 = 0$, $X_1 = 1$,$X_0 = 1$ pentru celelalte atribuiri având valoarea „1”.
- O funcţie de comutaţie de n variabile poate fi reprezentată printr-un produs de maxtermi:

$$f(X_{0,} X_1 .....X_{n-1}) = \prod_{i \in K0} M_i^n$$

unde K_0 este mulţimea indicilor M_i^n pt care care funcţia ia valoarea „0”.

Maxtermi & Mintermi: $M_i^n = \overline{m_i^n}$

A	B	C	maxterms		minterms	
0	0	0	$A + B + C$	M0	$\overline{A}\ \overline{B}\ \overline{C}$	m0
0	0	1	$A + B + \overline{C}$	M1	$\overline{A}\ \overline{B}\ C$	m1
0	1	0	$A + \overline{B} + C$	M2	$\overline{A}\ B\ \overline{C}$	m2
0	1	1	$A + \overline{B} + \overline{C}$	M3	$\overline{A}\ B\ C$	m3
1	0	0	$\overline{A} + B + C$	M4	$A\ \overline{B}\ \overline{C}$	m4
1	0	1	$\overline{A} + B + \overline{C}$	M5	$A\ \overline{B}\ C$	m5
1	1	0	$\overline{A} + \overline{B} + C$	M6	$A\ B\ \overline{C}$	m6
1	1	1	$\overline{A} + \overline{B} + \overline{C}$	M7	$A\ B\ C$	m7

Produs de maxtermi

$$F = \sum(3,5,6,7)$$

$$F = m_3^3 + m_5^3 + m_6^3 + m_7^3 = \sum(3,5,6,7)$$

$$F = M_0^3 \cdot M_1^3 \cdot M_2^3 \cdot M_4^3 = \prod(0,1,2,4)$$

- **forma canonică conjunctivă** a funcţiei:- termenii sumă logică ai funcţiei conţin **toate** variabilele funcţiei, între termeni realizându-se operaţia **ȘI**.

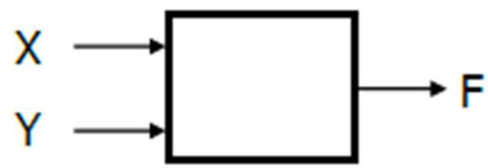
Forme canonice

- Două forme canonice:
  - Sumă de mintermi
  - Produs de maxtermi
- Formele canonice sunt unice!
- Conversia dintr-o formă canonică în alta se face:
  - Interschibând Σ și Π
  - Listând toți indicii lipsă

Glossar

Termen	Definiţie
Literal	Variabilă booleană sau complementul ei
Termen Produs (Product Term)	Literal sau produs logic (ŞI) între mai mulţi literali
Termen Sumă (Sum Term)	Literal sau sumă logică (OR) între mai mulţi literali
Sum of Products (SOP)	Sumă logică (OR) între mai mulţi termeni produs
Products of Sums (POS)	Produs logic (ŞI) între mai mulţi termeni sumă
Minterm	Caz particular de termen produs, care conţine toate variabilele de intrare o singură dată
Maxterm	Caz particular de termen sumă, care conţine toate variabilele de intrare o singură dată
Sumă de produse canonică	Sumă logică (OR) de acei mintermi aferenţi rândurilor din tabelul de adevăr al funcţiei de ieşire unde aceasta are valoarea 1 logic
Produs de sume canonic	Produs logic (ŞI) de acei mintermi aferenţi rândurilor din tabelul de adevăr al funcţiei de ieşire unde aceasta are valoarea 0 logic

n variabile $\rightarrow \quad 2^{2^n}$ funcții



X	Y	16 possible functions (F_0–F_{15})															
0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
0	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
1	0	0	0	1	1	0	0	1	1	0	0	1	1	0	0	1	1
1	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1	0	1
		0	X AND Y		X		Y	X XOR Y	X OR Y	X NOR Y NOT (X OR Y)	X = Y	NOT Y		NOT X		X NAND Y NOT (X AND Y)	1

NAND

X	Y	Z
0	0	1
0	1	1
1	0	1
1	1	0

$$Z = \overline{X \bullet Y}$$

AND

X	Y	Z
0	0	0
0	1	0
1	0	0
1	1	1

$$Z = X \bullet Y$$

NOR

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	0

$$Z = \overline{X + Y}$$

OR

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	1

$$Z = X + Y$$

Porți logice (cont.)

XOR
$(X \oplus Y)$

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

$Z = X\,\overline{Y} + \overline{X}\,Y$
X or Y but not both
("inequality", "difference")

XNOR
$\overline{(X \oplus Y)}$

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	1

$Z = \overline{X}\,\overline{Y} + X\,Y$
X and Y the same
("equality")

Porți logice

- Fiecare poartă logică realizează una sau mai multe funcții logice;
- Colecția de porți logice folosită în realizarea unui circuit se numește **bibliotecă de porți**, iar porțile din cadrul ei **porți standard**;
- Bibliotecile moderne conțin zeci de porți a.î. să scadă costul cu întreținerea și să simplifice realizarea tool-urilor CAD

Minimizarea funcțiilor logice

- se înţelege simplificarea/rescrierea ecuaţiilor logice booleene în vederea:
  - Unui cost mai mic și/sau;
  - Performanță mai ridicată;
- Cheia simplificării este: $y\left(x+\overline{x}\right)=y$
  - distributivitatea - *x(y+z)=xy+xz*
  - Proprietatea complementului $x+\overline{x}=1$

Minimizarea funcțiilor logice

- Găsirea a doi termeni (suma sau produs funcție de reprezentarea dorită SOP/POS) pentru care:
  - funcția ia valoare 1
  - numai o variabilă își modifică valoarea

A	B	F
0	0	1
0	1	0
1	0	1
1	1	0

B are acеși valoare → B este păstrat

A are valori diferite → A este eliminat

$$F = \bar{A}\,\bar{B} + A\bar{B} = (\bar{A} + A)\bar{B} = \bar{B}$$

Metoda de minimizare Karnaugh

- Diagramele Karnaugh:
  - Metodă alternativă tabelelor de adevăr și ecuațiilor logice de a vizualiza o funcție
  - se aplică atât pentru ecuaţiile logice descrise sub formă canonică de sumă de produse (SOP), cât şi pentru ecuaţiile logice descrise sub formă canonică de produs de sume (POS)

Diagramele Karnaugh

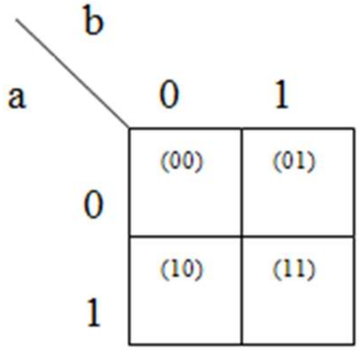
- constituie o matrice de pătrate cu proprietatea ca două celule vecine corespund unor mintermi **adiacenţi**.
- doi vectori sunt adiacenţi dacă diferă valoric printr-un singur bit
- în diagramă se marchează acei mintermi care au valoarea logică 1 în tabelul de adevăr

Diagrame Karnaugh

- Numerele adiacente numărului 0100 sunt: 0101; 0110; 0000; 1100.
- Numerele adiacente numărului 000 sunt: 001; 010; 100.
- Vectorii adiacenţi mintermului $ab\bar{c}$ sunt: $\bar{a}b\bar{c}, a\bar{b}\bar{c}, abc$

Construcție diagrame Karnaugh

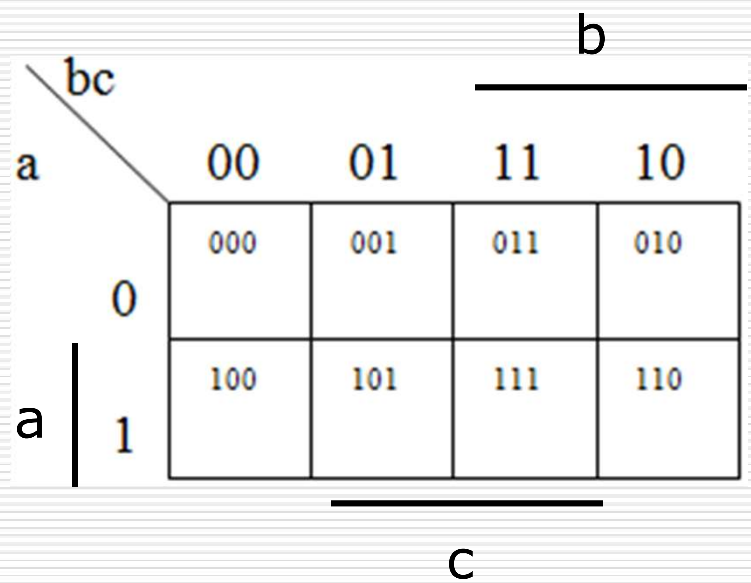
- Diagrame Karnaugh pentru funcții logice cu 2 variabile a, b



00 - $\bar{a}\bar{b}$
01 - $a\bar{b}$
11 - ab
10 - $\bar{a}b$

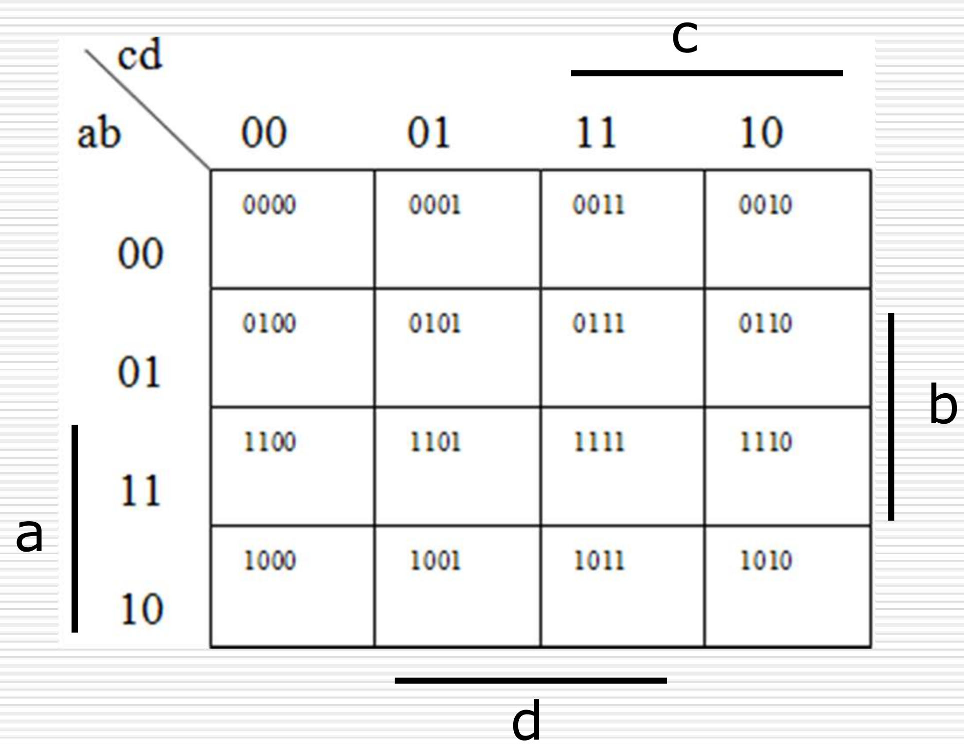
Construcție diagrame Karnaugh

- Diagrame Karnaugh pentru funcții logice cu 3 variabile a, b, c



Construcție diagrame Karnaugh

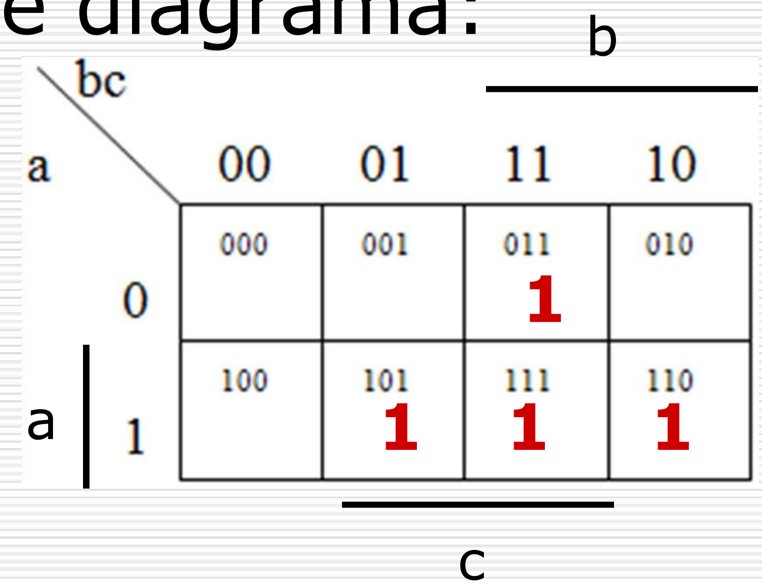
- construcţia diagramelor Karnaugh pentru o funcţie logică cu 4 variabile de intrare



Diagrame Karnaugh

- Ex. completare diagramă:

a	b	c	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



1. Introducerea mintermilor în diagramă conform tabelului de adevăr.

Diagrame Karnaugh

- Ex. completare diagramă:

a	b	c	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

b

a \ bc	00	01	11	10
0	000	001	011 **1**	010
a 1	100	101 **1**	111 **1**	110 **1**

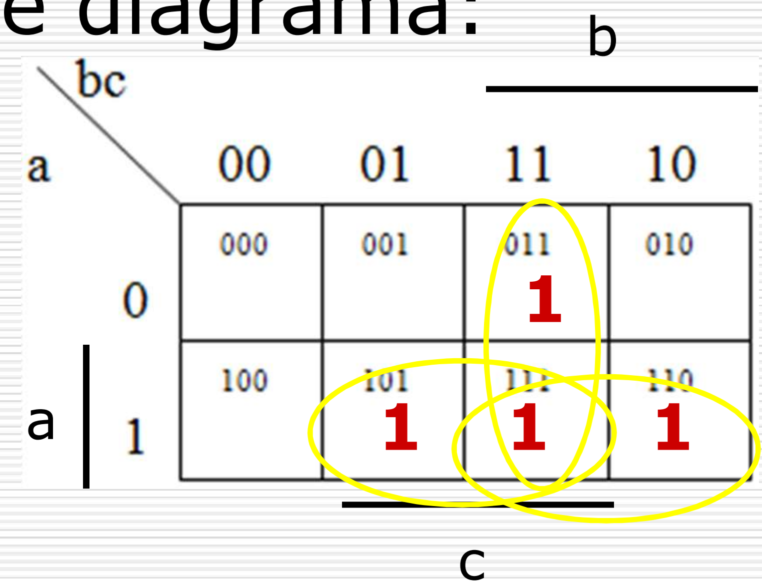
c

2. se încearcă formarea unor grupe de mintermi bazate pe reguli de adiacenţă

Diagrame Karnaugh

- Ex. completare diagramă:

a	b	c	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

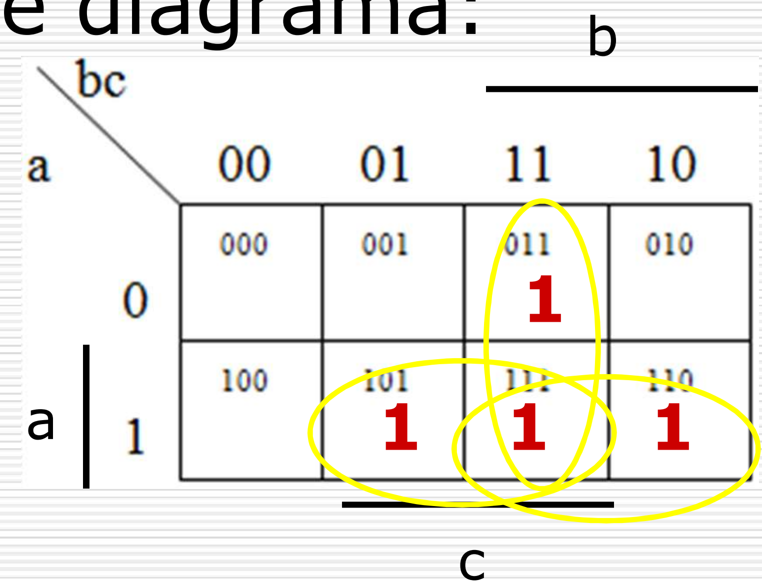


2. O grupare are forma unor dreptunghiuri/pătrate și conţine 2^n mintermi!

Diagrame Karnaugh

- Ex. completare diagramă:

a	b	c	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

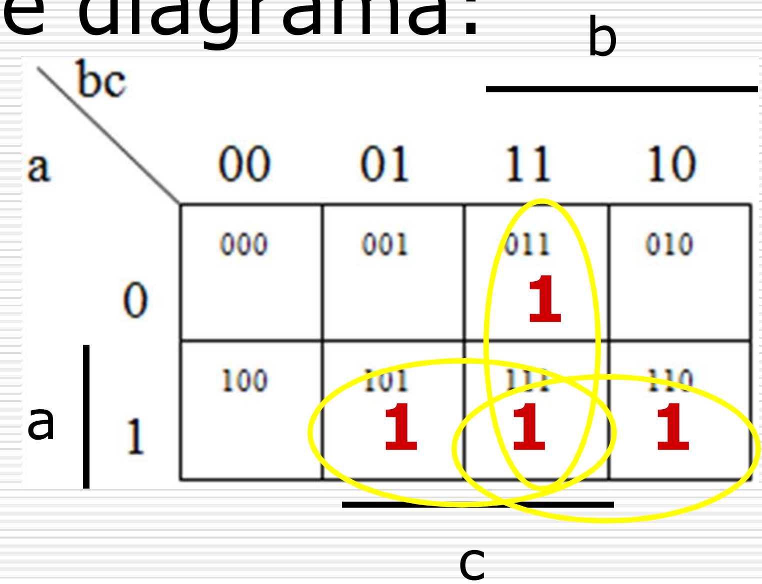


2. O grupare are forma unor dreptunghiuri/pătrate și conţine 2^n mintermi!

Diagrame Karnaugh

- Ex. completare diagramă:

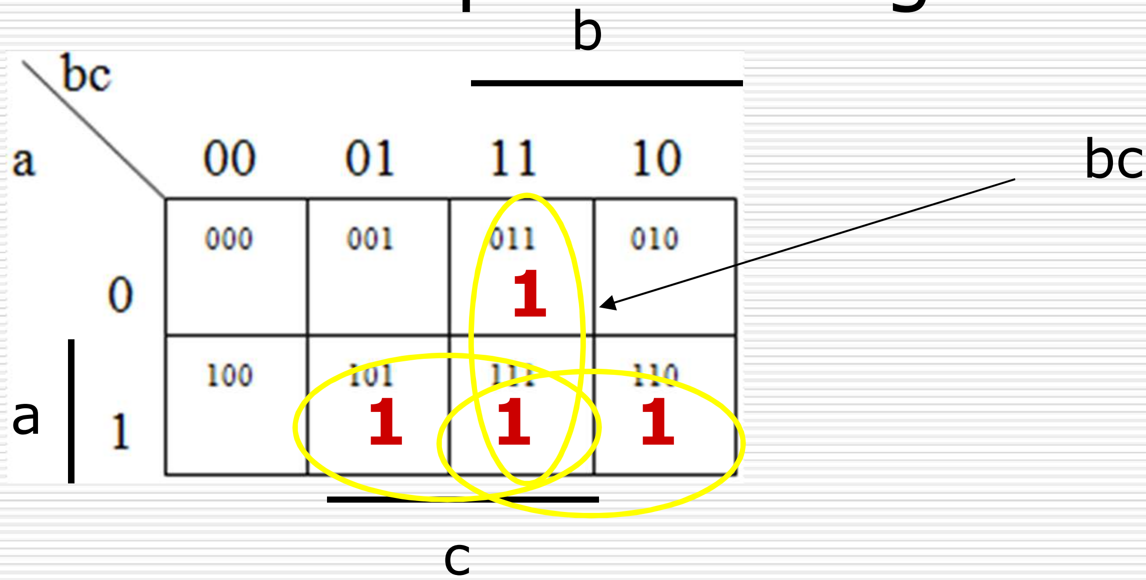
a	b	c	f
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



2. Din totalul de *m* variabile booleene a funcţiei, termenul asociat grupării formate va conţine *m-n* variabile

Diagrame Karnaugh

- Ex. completare diagramă:



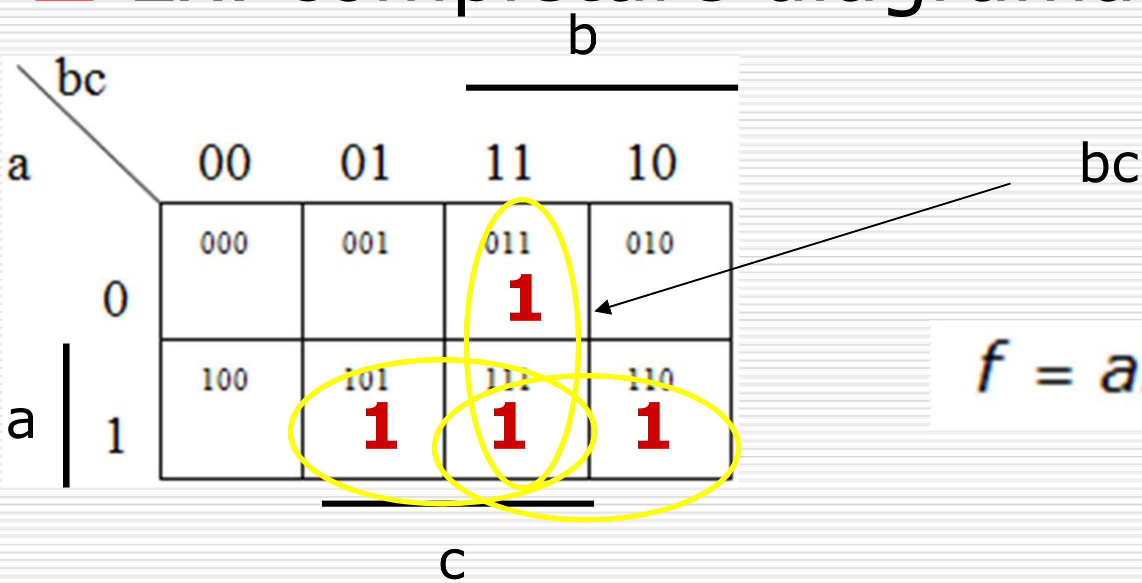
2. Din totalul de *m* variabile booleene a funcţiei, termenul asociat grupării formate va conţine *m*-*n* variabile

Minimizarea folosind diagrame Karnaugh

- Dacă la o astfel de grupare nu mai pot fi adăugaţi mintermi înseamnă că s-a obţinut un **implicant prim**.
- Dacă un anumit implicant prim conţine cel puţin un minterm care nu poate apare în alt implicanţi primi atunci acesta este un **implicant prim esenţial**

Diagrame Karnaugh

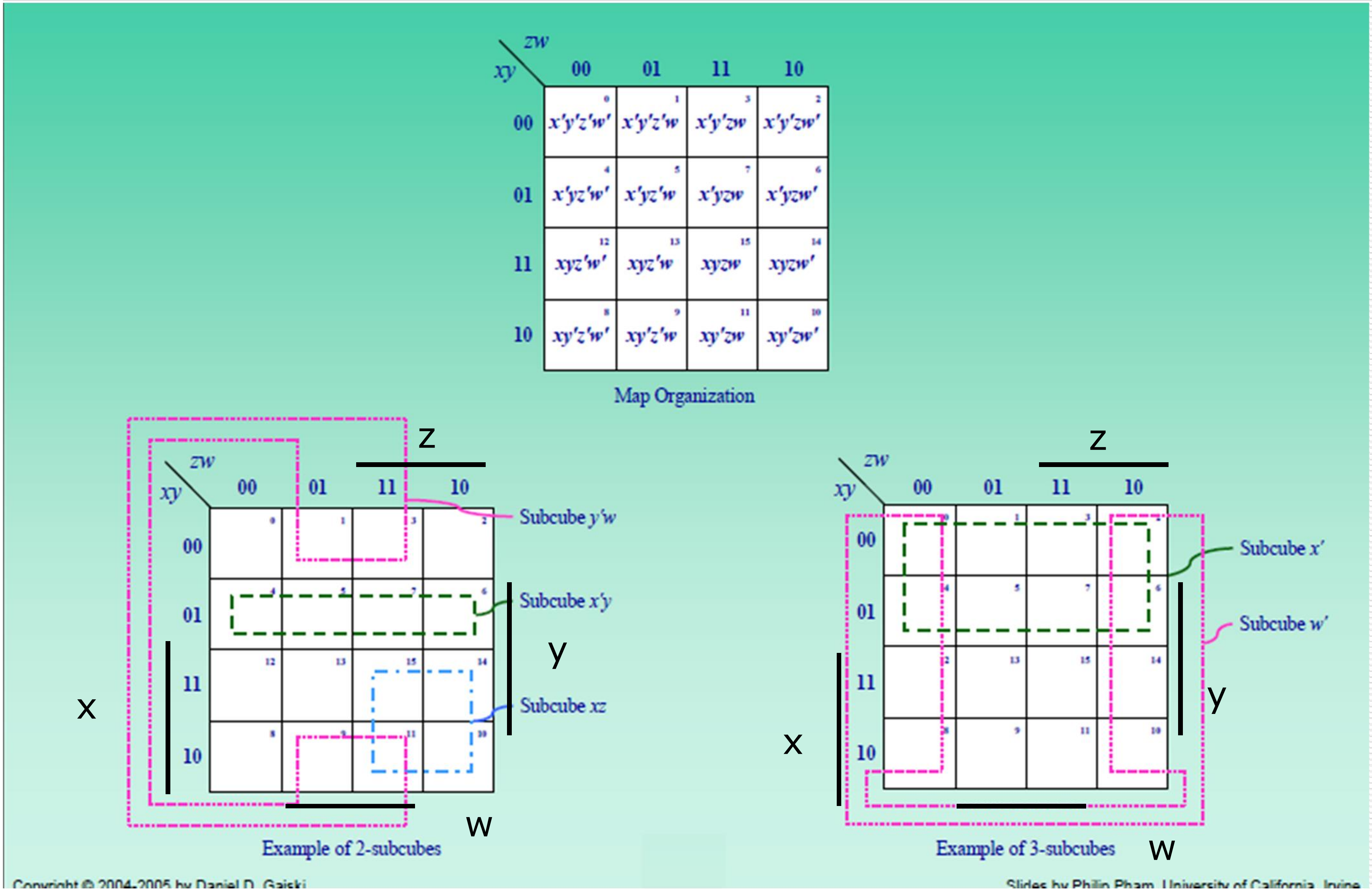
- Ex. completare diagramă:



$$f = ab + ac + bc$$

3. Ecuaţia minimizată va conţine toţi implicanţii primi esenţiali, si uneori si implicanţi primi neesenţiali, astfel încât toate celule marcate cu 1 logic să fie acoperite.

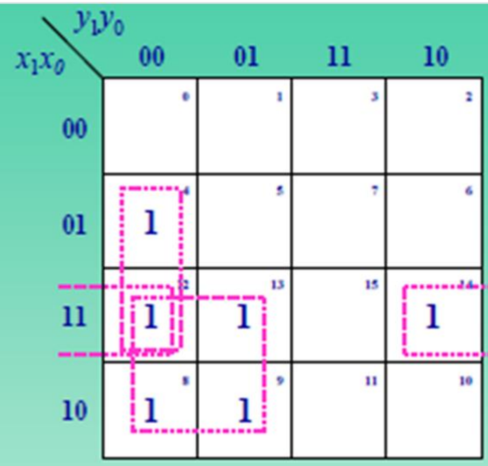
Diagrame Karnaugh



Diagrame Karnaugh

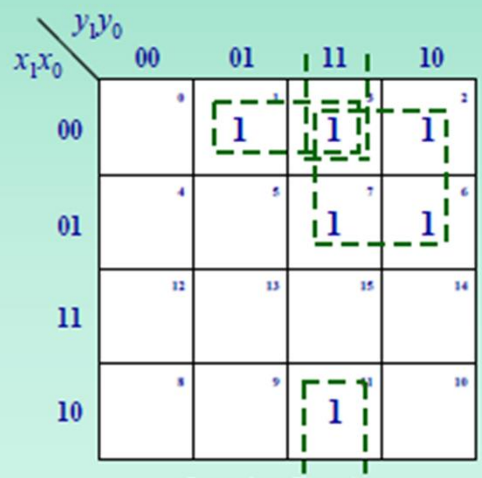
x_1	x_0	y_1	y_0	Greater Than	Equal	Less Than
0	0	0	0	0	1	0
0	0	0	1	0	0	1
0	0	1	0	0	0	1
0	0	1	1	0	0	1
0	1	0	0	1	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	1
0	1	1	1	0	0	1
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	1	0
1	0	1	1	0	0	1
1	1	0	0	1	0	0
1	1	0	1	1	0	0
1	1	1	0	1	0	0
1	1	1	1	0	1	0

Truth Table



$$G = x_1y_1' + x_0y_1'y_0' + x_1x_0y_0'$$

Greater-than Function



$$L = x_1'y_1 + x_1'x_0'y_0 + x_0'y_1y_0$$

Less-than Function

Diagrame Karnaugh – don’t care

- Nu toate funcţiile logice sunt definite complet.
- Pentru unele valori ale variabile de intrare funcţia este nu specificată (funcţia are „valoarea” don’t care - d)
- Pt.”d” în diagrama Karnaugh se va lua în considerare valoarea care ne convine pentru d (0 sau 1) a.î. să permită o acoperire mai largă a minternilor.

a	b	c	p	q
0	0	0	0	1
0	0	1	1	0
0	1	0	1	d
0	1	1	d	1
1	0	0	0	d
1	0	1	0	0
1	1	0	d	0
1	1	1	1	1

Tabelul 5.3

Funcţiile p şi q pot fi scrise şi altfel:

$$p = f(a,b,c) = \sum(1;2;7) + \sum d(3;6)$$

$$q = f(a,b,c) = \sum(0;3;7) + \sum d(2;4)$$

a \ bc	00	01	11	10
0		1	d	1
1			1	d

a \ bc	00	01	11	10
0	1		1	d
1	d		1	

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

$q = \bar{b}\bar{c} + bc$

Porți logice (cont.)

XOR
$(X \oplus Y)$

X	Y	Z
0	0	0
0	1	1
1	0	1
1	1	0

$Z = X\overline{Y} + \overline{X}Y$
X or Y but not both
("inequality", "difference")

XNOR
$\overline{(X \oplus Y)}$

X	Y	Z
0	0	1
0	1	0
1	0	0
1	1	1

$Z = \overline{X}\,\overline{Y} + XY$
X and Y the same
("equality")

Porți logice

- Fiecare poartă logică realizează una sau mai multe funcții logice;
- Colecția de porți logice folosită în realizarea unui circuit se numește **bibliotecă de porți**, iar porțile din cadrul ei **porți standard**;
- Bibliotecile moderne conțin zeci de porți a.î. să scadă costul cu mentenanta și să simplifice realizarea tool-urilor CAD

Porți logice

- Criteriile pt.selecția operatorilor:
  - Frecvența utilizării porții;
  - Extensibilitatea operatorului pentru mai mult de 2 variabile;
  - Costul prin prisma nr. de tranzistori, respectiv timpul de comutare al porții;
- 8 operatori sunt selectați: NOT(Inverter), Driver (comandarea multor circuite (load mare), linii lungi),And, Or, NOR, NAND, XOR, XNOR

PORȚI LOGICE ELEMENTARE

Name	Graphic Symbol	Functional Expression	Number of transistors	Delay in *ns*
Inverter		$F = x'$	2	1
Driver		$F = x$	4	2
AND		$F = xy$	6	2.4
OR		$F = x + y$	6	2.4
NAND		$F = (xy)'$	4	1.4
NOR		$F = (x + y)'$	4	1.4
XOR		$F = x \oplus y$	14	4.2
XNOR		$F = x \odot y$	12	3.2

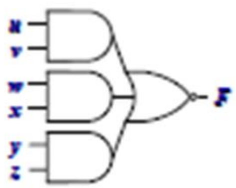
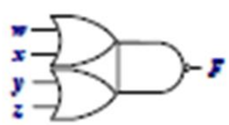
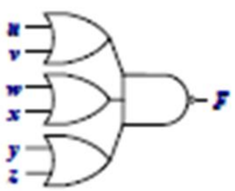
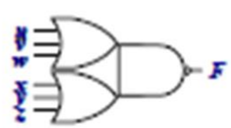
Porți logice

- Modul prin care pot fi optimizate:
  - Întârzierea (delay)
  - Costul (nr.tranzistori)
- 2 posibilități:
  - Porți cu mai multe intrări (operatori asociativi și comutativi). Ex.: AND, OR
  - Porți complexe:
    - mai multe intrări
    - Realizează operații multiple

PORȚI LOGICE CU MAI MULTE INTRĂRI

Name	Graphic Symbol	Functional Expression	Number of transistors	Delay in *ns*
3–input AND		$F = xyz$	8	2.8
4–input AND		$F = xyzw$	10	3.2
3–input OR		$F = x + y + z$	8	2.8
4–input OR		$F = x + y + z + w$	10	3.2
3–input NAND		$F = (xyz)'$	6	1.8
4–input NAND		$F = (xyzw)'$	8	2.2
3–input NOR		$F = (x + y + z)'$	6	1.8
4–input NOR		$F = (x + y + z + w)'$	8	2.2

PORȚI LOGICE COMPLEXE

Name	Graphic Symbol	Functional Expression	Number of transistors	Delay in ns
2–wide, 2–input AOI		$F = (wx + yz)'$	8	2.0
3–wide, 2–input AOI		$F = (uv + wx + yz)'$	12	2.4
2–wide, 3–input AOI		$F = (uvw + xyz)'$	12	2.2
2–wide, 2–input OAI		$F = ((w + x)(y + z))'$	8	2.0
3–wide, 2–input OAI		$F = ((u + v)(w + x)(y + z))'$	12	2.2
2–wide, 3–input OAI		$F = ((u + v + w)(x + y + z))'$	12	2.4

Întrebări?