PPI 2. Logická úroveň a stavba počítačových systémov

2.1. LOGICKÉ OBVODY

2.1.2. ANALÝZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

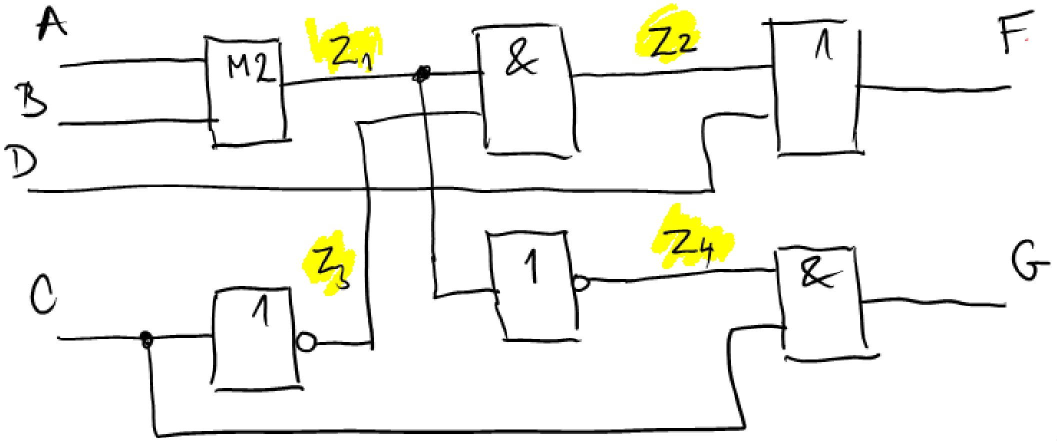
Je to riešenie tejto úlohy:

Je známa štruktúra logického obvodu a je potrebné nájsť a opísať jeho správanie (musí byť známe správanie sa prvkov, ktoré obsahuje štruktúra logického obvodu).

- *úloha má* ***vždy jednoznačné*** *riešenie*
- *správanie obvodu možno zapísať niektorým spôsobom uvedeným v predchádzajúcej kapitole (v 2.1.1.1)*

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.2. ANALÝZA LOGICKÝCH KOMBINAČNÝCH OBVODOV



Nájdite: $F = f_1(A,B,C,D)$

$G \quad f_2(A,B,C,D)$

$$F = \underbrace{Z_1 \cdot Z_3}_{Z_2} + D = (\underbrace{A \oplus B}_{Z_1}) \cdot \bar{C} + D$$

$$F = (\bar{A}B + A\bar{B}) . \bar{C} + D$$

$$\underline{\underline{F = \bar{A}B\bar{C} + A\bar{B}\bar{C} + D}}$$ → DNF

$Z_1 = A \oplus B$

$Z_2 = Z_1 \cdot Z_3$

$F = Z_2 + D$

$Z_3 = \bar{C}$

$Z_4 = \bar{Z}_1$

$G = Z4 . C$

princip substitúcie

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.2. ANALÝZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

$$G = C \cdot Z_4 = C \cdot \overline{Z}_1 = C \cdot (\overline{\underbrace{A \oplus B}_{Z_1}}) = C \cdot (\overline{A\overline{B} + \overline{A}B}) =$$

$$= C \cdot (\overline{A}+B)(A+\overline{B}) =$$

$$= C \cdot (\overline{A}\overline{B} + AB) = \underline{\underline{\overline{A}\overline{B}C + ABC}} \leftarrow \text{DNF}$$

a/ MAPOVÝ ZÁPIS

			C		
				D	
		$^{0}0$	$^{2}0$	$^{3}1$	$^{1}1$
B		$^{4}1$	$^{6}0$	$^{7}1$	$^{5}1$
	A	$^{12}0$	$^{14}0$	$^{15}1$	$^{13}1$
		$^{8}1$	$^{10}0$	$^{11}1$	$^{9}1$

F

ABCD

(0000) → DE „0"

(0011) → DE „3"

			C		
				D	
		0	1	1	0
B		0	0	0	0
	A	0	1	1	0
		0	0	0	0

G

ČÍSELNÝ ZÁPIS

$F = D(1,3,4,5,7,8,9,11,13,15)$

$F = K(0,2,6,10,12,14)$

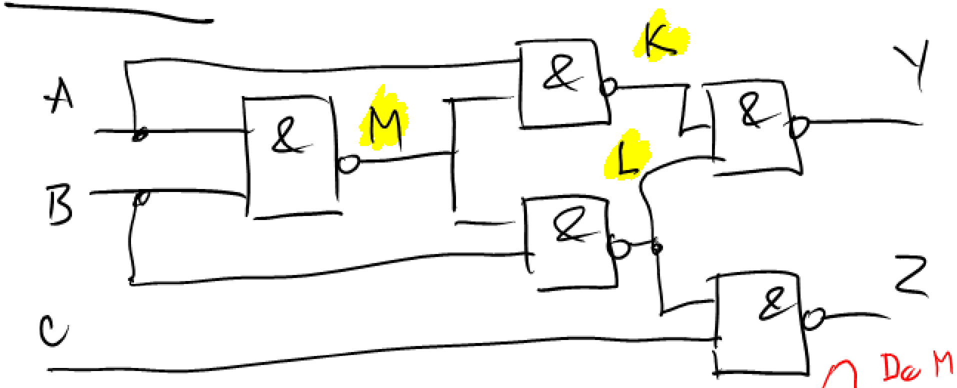
$G = D(2,3,14,15)$

$G = K(0,1,4,5,6,7,8,9,10,11,12,13)$

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.2. ANALÝZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

PRÍKLAD



NAND

x_1 x_2	f
0 0	1
0 1	1
1 0	1
1 1	0

NAND $f = \overline{x_1 \cdot x_2}$

De Morgan

$$Y = \overline{K \cdot L} = \overline{\overline{A \cdot M} \cdot \overline{M \cdot B}} = \overline{\overline{A . \overline{A . B}}} \; . \; \overline{\overline{\overline{A . B} . B}}$$

$$Y = \overline{\overline{A \cdot \overline{A \cdot B}}} + \overline{\overline{\overline{A \cdot B} \cdot B}}$$

$$Y = A . \overline{A \cdot B} + B . \overline{A \cdot B}$$

$$Y = A . (\bar{A} + \bar{B}) + B (\bar{A} + \bar{B})$$

$$Y = A\bar{B} + B\bar{A}$$

	A	A	
0	1	0	1
C 0	1	0	1

Y

nezávisí od C

	A
0	1
B 1	0

Y menšia mapa

2.1.2. ANALÝZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

$$Z = \overline{L \cdot C} = \overline{\overline{M \cdot B} \cdot C} = \overline{\overline{\overline{A \cdot B} \cdot B} \cdot C}$$

$$= \overline{\overline{B \cdot \overline{A}} \cdot C}$$

$$= \overline{\overline{B \cdot \overline{A}}} + \overline{C}$$

$$= B \cdot \overline{A} + \overline{C}$$

A B

	1	1	1	1
C	0	0	0	1

Z

Výsledok v normovanom tvare DNF :

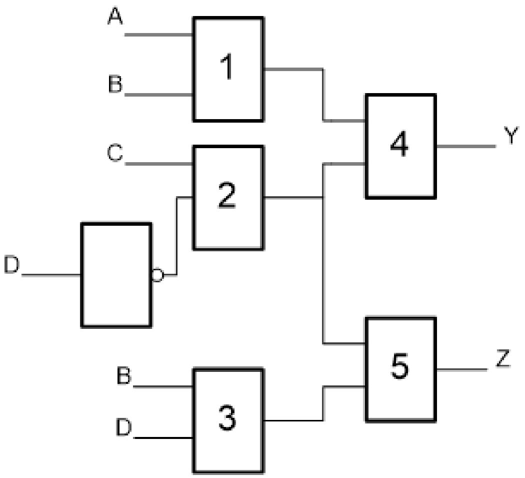
$$Y = A\overline{B} + B\overline{A}$$

$$Z = B\overline{A} + \overline{C}$$

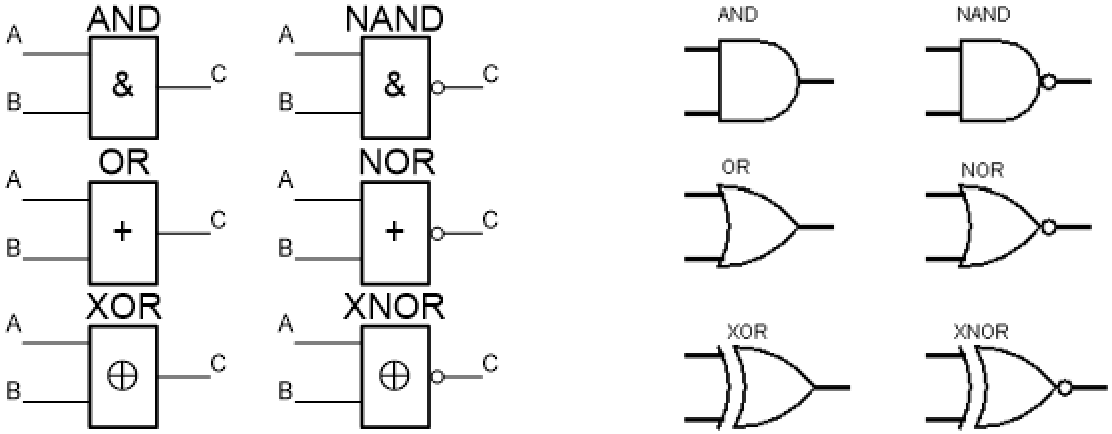
PPI 2. Logická úroveň a stavba počítačových systémov

2.1.2. ANALÝZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Všeobecná schéma obvodu



Logické členy



PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Je to riešenie tejto úlohy:

- *je daná skupina B-funkcií*
- *je daný súbor typov logických členov*

a je potrebné nájsť (navrhnúť) štruktúru logického obvodu, ktorý realizuje danú skupinu funkcií

(môžu byť dané aj kritéria optimálnosti riešenia)

- *najčastejším kritériom optimálnosti je minimálny počet logických členov (optimálna syntéza)*
- *úloha nemá **jednoznačné** riešenie*

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Postup:

- *hľadá sa skupina výrazov zodpovedajúcich danej skupine B-funkcií*
- *algoritmické riešenie existuje iba pre niektoré typy výrazov a obvodov*
    - *najčastejšie iba pre DNF , KNF a* ***dvojstupňové*** *obvody*

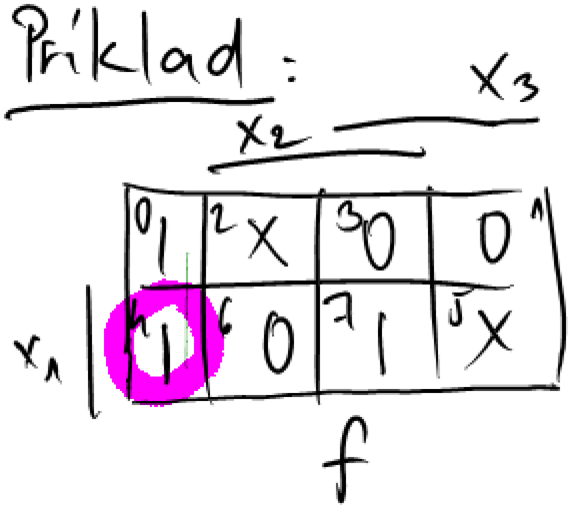
2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.1. VYJADROVANIE IDNF (IKNF) Z MAPY

Implikant funkcie f je súčin, ktorý má hodnotu 1 v bodoch funkcie, v ktorých má táto funkcia hodnotu 1 alebo nie je definovaná.

Prostý implikant funkcie f je taký implikant funkcie, ktorého nijaká vlastná časť nie je implikantom funkcie f.

Príklad:



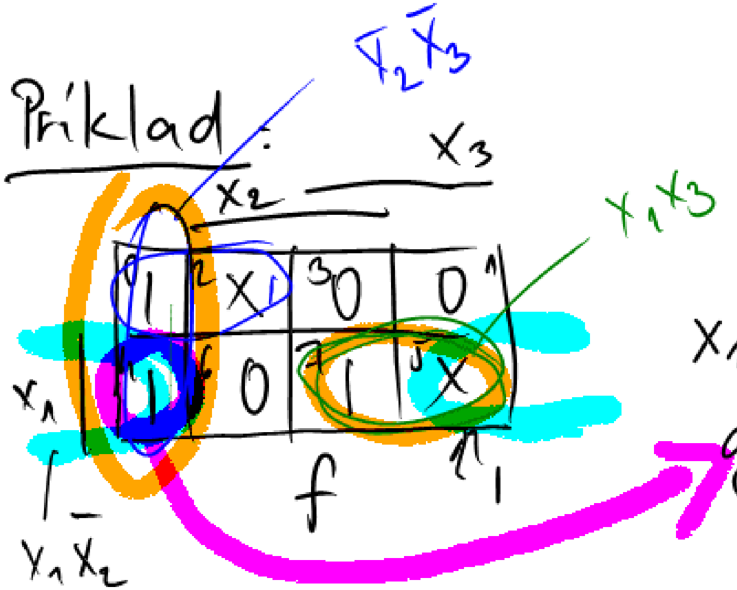
	0 1	2 X	3 0	0 1
x_1	4 1	6 0	7 1	5 X

$x_1\bar{x}_2$ -> je implikant f

$g_1 = x_1\bar{x}_2\cancel{\bar{x}_3}$ - je implikant f ale nie je prostý

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.1. VYJADROVANIE IDNF (IKNF) Z MAPY



$x_1\bar{x}_2$ -> je implikant f

$g_1 = x_1 \bar{x}_2 \cancel{\bar{x}_3}$ - je implikant f ale nie je prostý implikant

$g_1' = x_1 \bar{x}_2$ => je prostý implikant

$f = x_1\bar{x}_2 + \boxed{\bar{x}_2\bar{x}_3 + x_1x_3} + \bar{x}_1\bar{x}_3$ MDNF IDNF

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.1. VYJADROVANIE IDNF (IKNF) Z MAPY

Dôsledky:

- **SDNF** (SKNF) obsahuje **všetky prosté implikanty** (implicenty) funkcie
- **IDNF** (IKNF) obsahuje **iba prosté implikanty** (implicenty), ale **nie všetky** (iba tie prosté implikanty, aby bolo zabezpečené, že každý bod funkcie s hodnotou 1 je pokrytý **aspoň jedenkrát**)

Pravidelná konfigurácia – vlastnosti:

- Obsahuje **2^s** štvorčekov v mape (**s** je stupeň konfigurácie)
- Každý štvorček v konfigurácii má práve **s susedných** štvorčekov („susedný štvorček" - opisy sa líšia iba v hodnote jednej vstupnej premennej)

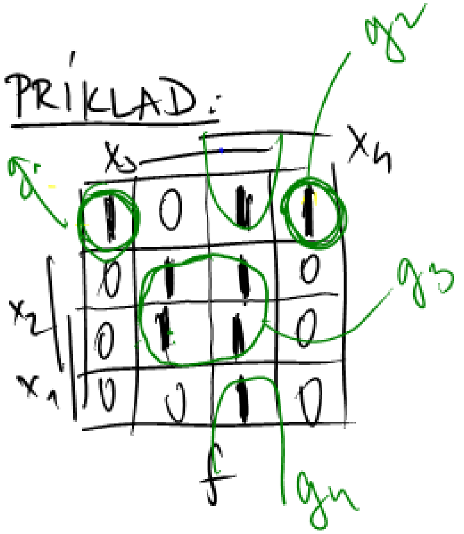
Postup hľadania **IDNF** (MDNF) z mapy:

1. Pokrývajú sa všetky body funkcie s hodnotou 1 pravidelnými konfiguráciami tak, aby každý **jednotkový** bod funkcie bol pokrytý aspoň **jedenkrát**
2. Každá konfigurácia sa vyjadrí **súčinom** vstupných premenných
3. IDNF je **logickým súčtom** týchto súčinov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.1. VYJADROVANIE IDNF (IKNF) Z MAPY

PRÍKLAD:



1	0	1	1
0	1	1	0
0	1	1	0
0	0	1	0

$$f = \underbrace{\bar{x}_1\bar{x}_2\bar{x}_3\bar{x}_4}_{g_1} + \underbrace{\bar{x}_1\bar{x}_2\bar{x}_3x_4}_{g_2} + \underbrace{x_2x_3}_{g_3} + \underbrace{\bar{x}_2x_3x_4}_{g_4}$$

$r=4$

Pravidelná konfigurácia

$$g_i = \underbrace{\tilde{x}_{i1}\cdot\tilde{x}_{i2}\cdot\ldots\cdot\tilde{x}_{ir}}_{r}\underbrace{\ldots\ldots}_{n-r}$$

$$\tilde{x}_i = \begin{cases} x_i \\ \bar{x}_i \end{cases}$$

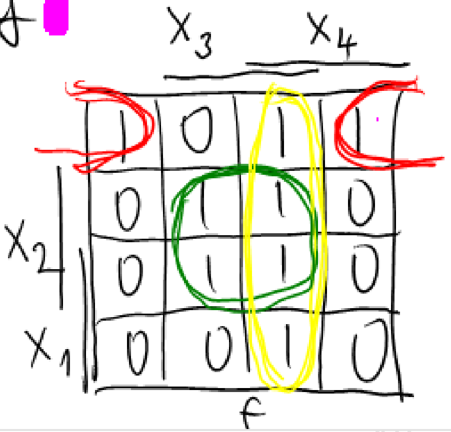
S pomocou pravidelných konfigurácií vieme napísať výraz pre funkciu

Vlastnosti:

1. obsahuje 2^s štvorčekov
   s - stupeň konf
2. každý štvorček má s susedných štvorčekov

$$g_1 + g_2 = g_5 = \bar{x}_1\bar{x}_2\bar{x}_3$$

$$f = \underset{\uparrow}{\bar{x}_1\bar{x}_2\bar{x}_3} + \underset{\uparrow}{x_2x_3} + \underset{\uparrow}{x_3x_4}$$



1	0	1	1
0	1	1	0
0	1	1	0
0	0	1	0

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.1. VYJADROVANIE IDNF (IKNF) Z MAPY

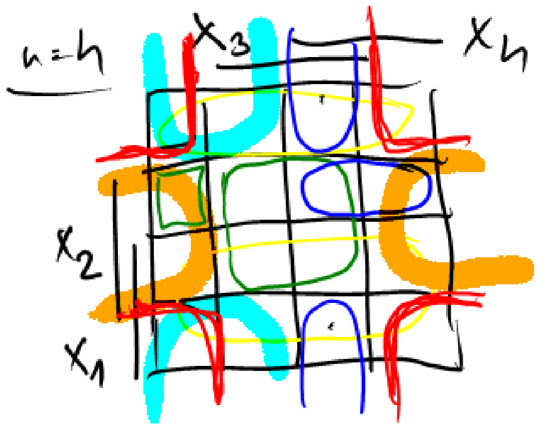
n – počet premenných danej funkcie

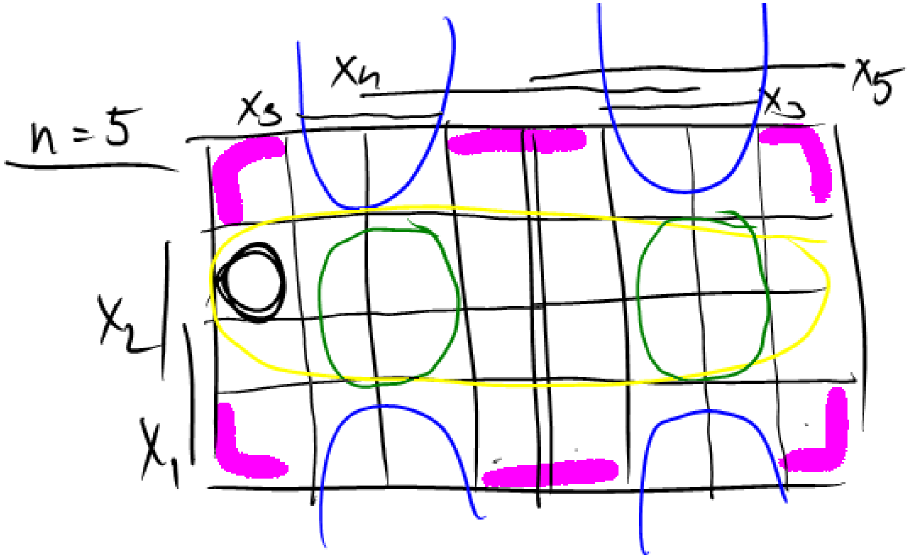
s – stupeň konfigurácie

r – rád súčinu

r = n-s

Príklady pravidelných konfigurácií





PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.1. VYJADROVANIE IDNF (IKNF) Z MAPY

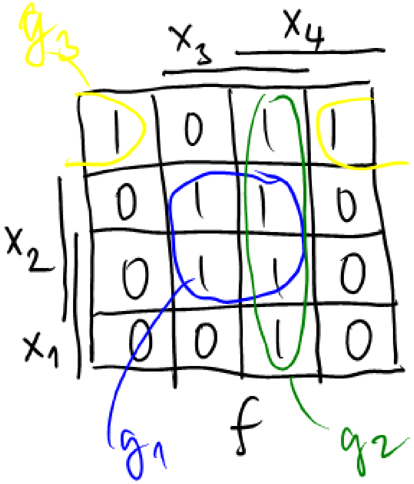
Základná stratégia pri hľadaní DNF (minimálnej skupiny DNF):

- Pokryť **všetky jednotkové body** funkcie (skupiny funkcií) minimálnym počtom čo najväčších (**maximálnych**) konfigurácií
- Vyjadriť každú konfiguráciu súčinom
- Súčtom spojiť všetky súčiny do výrazu funkcie (funkcií) DNF

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.1. VYJADROVANIE IDNF (IKNF) Z MAPY

Príklad



IDNF:

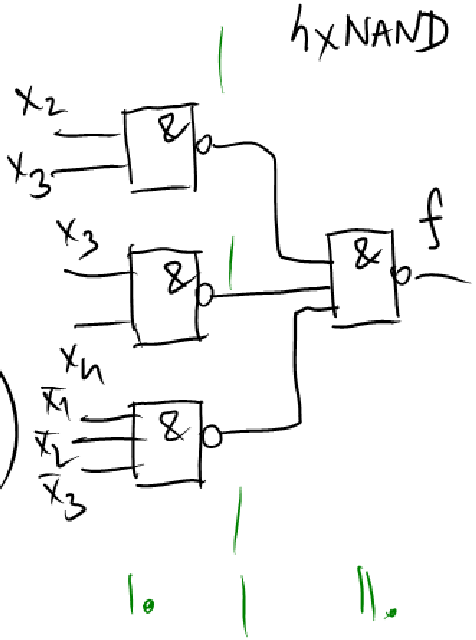
$$f = \underbrace{x_2 x_3}_{g_1} + \underbrace{x_3 x_4}_{g_2} + \underbrace{\bar{x}_1 \bar{x}_2 \bar{x}_3}_{g_3}$$

MDNF:

$$f = x_2 x_3 + x_3 x_4 + \bar{x}_1 \bar{x}_2 \bar{x}_3$$

NF ↑/↑:

$$f = \underbrace{(x_2 \uparrow x_3)} \uparrow \underbrace{(x_3 \uparrow x_4)} \uparrow \underbrace{(\bar{x}_1 \uparrow \bar{x}_2 \uparrow \bar{x}_3)}$$

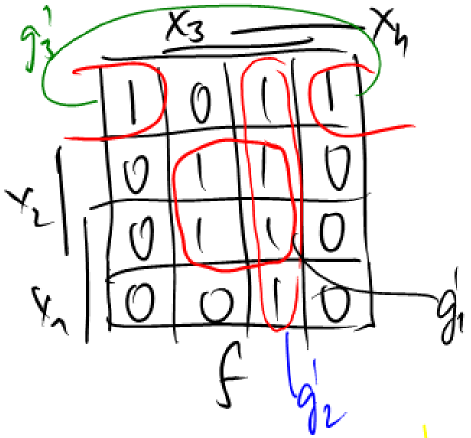


Realizácia s členmi NAND

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.1. VYJADROVANIE IDNF (IKNF) Z MAPY



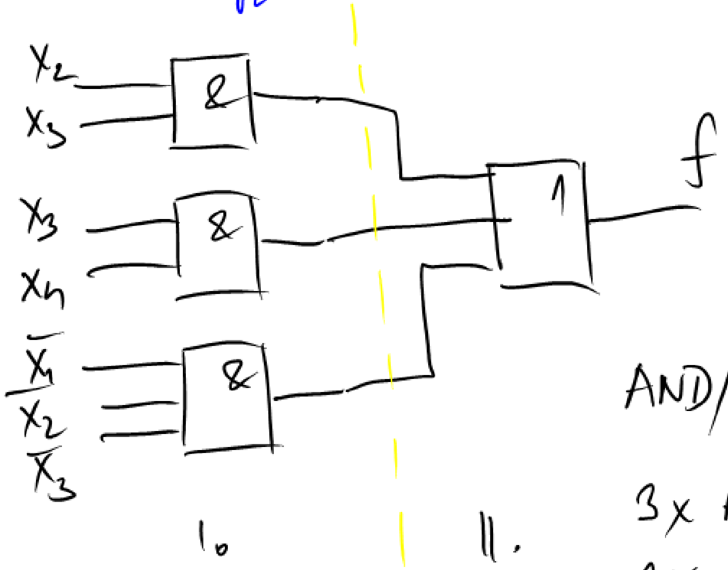
IDNF

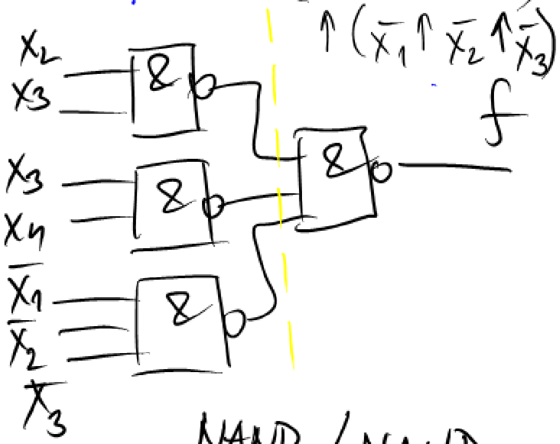
→ NF ↑/↑

$$f = \underbrace{x_2 x_3}_{g_1'} + \underbrace{x_3 x_4}_{g_2'} + \underbrace{\overline{x_1}\,\overline{x_2}\,\overline{x_3}}_{g_3'} =$$

⇒ MDNF!!

$$= (x_2 \uparrow x_3) \uparrow (x_3 \uparrow x_4) \uparrow (\overline{x_1} \uparrow \overline{x_2} \uparrow \overline{x_3})$$





AND/OR

3x AND

1x OR

NAND/NAND

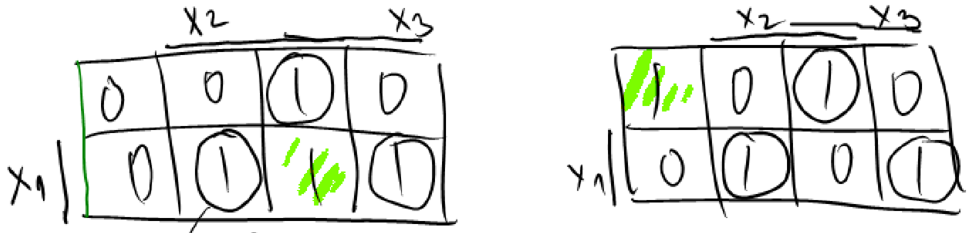
4x NAND

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Vyjadrovanie minimálnych skupín DNF (KNF) z máp a návrh obvodu

Príklad: *Vykonajte skupinovú minimalizáciu funkcií f1 a f2*

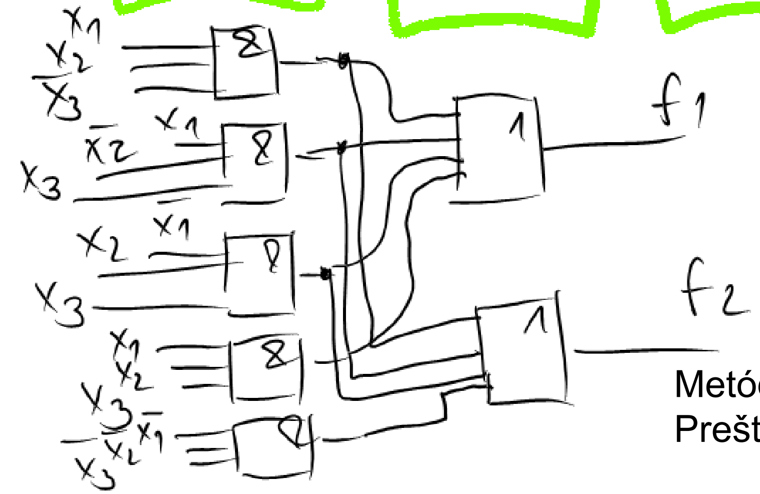


f_1 f_2

5 rôznych súčinov

$$f_1 = x_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 + \bar{x}_1 x_2 x_3 + x_1 x_2 x_3$$

$$f_2 = x_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 x_3 + \bar{x}_1 x_2 x_3 + \bar{x}_1 \bar{x}_2 \bar{x}_3$$



5 x AND

2 x OR

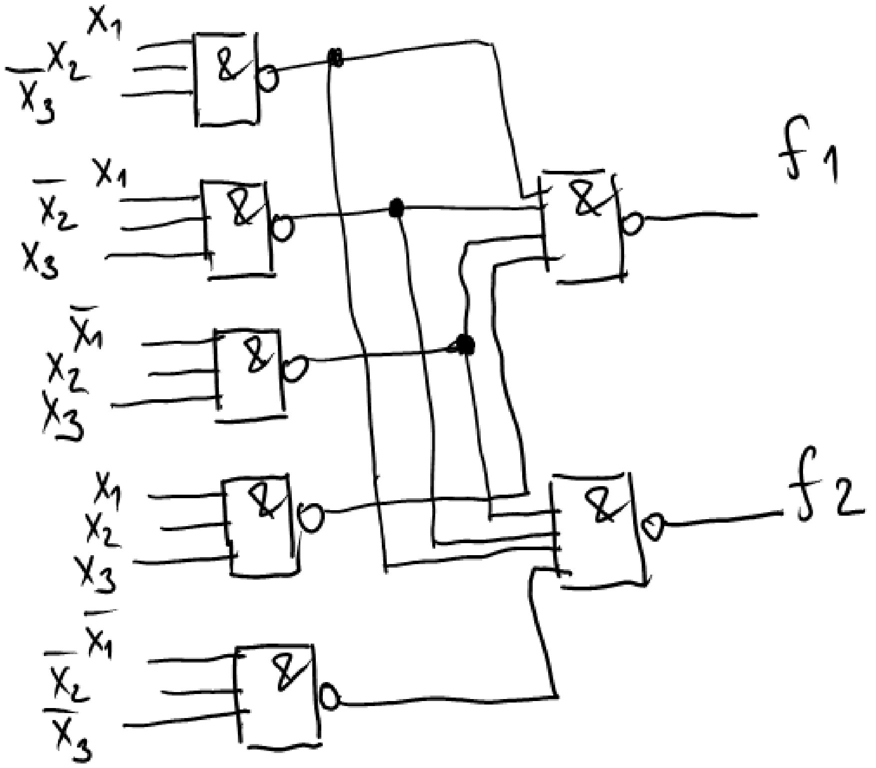
Metódy minimalizácie: Quine-McCluskey,Pattrick
Preštudovať v doporučenej literatúre

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Vyjadrovanie minimálnych skupín DNF (KNF) z máp a návrh obvodu

Realizácia s logickými členmi NAND:

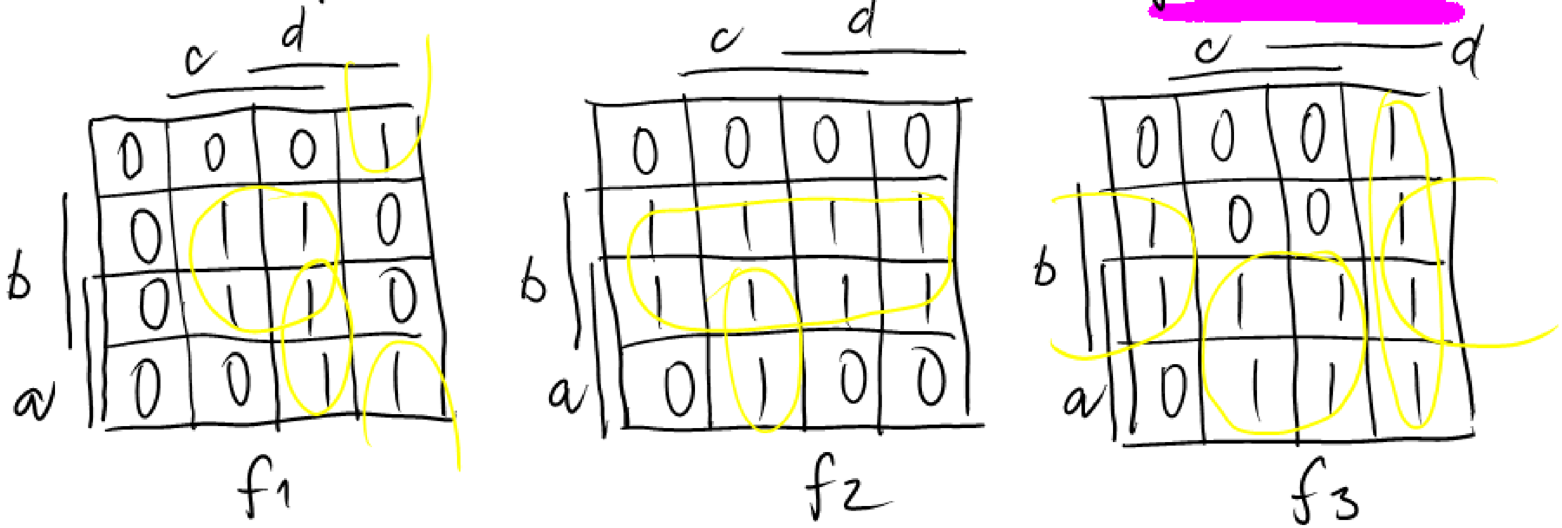


7 x NAND

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Vyjadrovanie minimálnych skupín DNF (KNF) z máp a návrh obvodu

Príklad: Nájdite DNF funkcií f_1, f_2, f_3 jednotlivo



f_1

0	0	0	1
0	1	1	0
0	1	1	0
0	0	1	1

f_2

0	0	0	0
1	1	1	1
1	1	1	1
0	1	0	0

f_3

0	0	0	1
1	0	0	1
1	1	1	1
0	1	1	1

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

$f_2 = b + ac\bar{d}$

$f_3 = \bar{c}d + ac + b\bar{c}$

7 súčinov

3 súčty

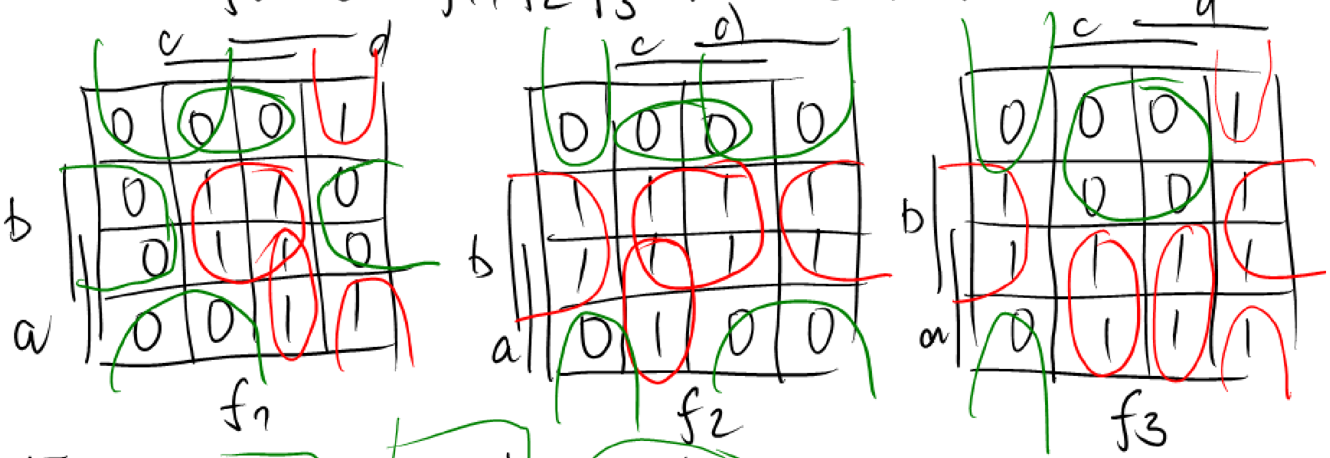
10 logických členov

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Vyjadrovanie minimálnych skupín DNF (KNF) z máp a návrh obvodu

Príklad: Vykonajte skupinovú minimalizáciu funkcií $f_1, f_2\, f_3$ v tvare KNF a DNF



0	0	0	1
0	1	1	0
0	1	1	0
0	0	1	1

f_1

0	0	0	0
1	1	1	1
1	1	1	1
0	1	0	0

f_2

0	0	0	1
1	0	0	1
1	1	1	1
0	1	1	1

f_3

MsDNF

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

$$f_2 = bc + b\bar{c} + ac\bar{d}$$

$$f_3 = b\bar{c} + \bar{b}\bar{c}d + ac\bar{d} + acd$$

5 súčinov
3 súčty
8 log. členov

MsKNF

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

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

$$f_3 = (b+c+d)(a+\bar{c})$$

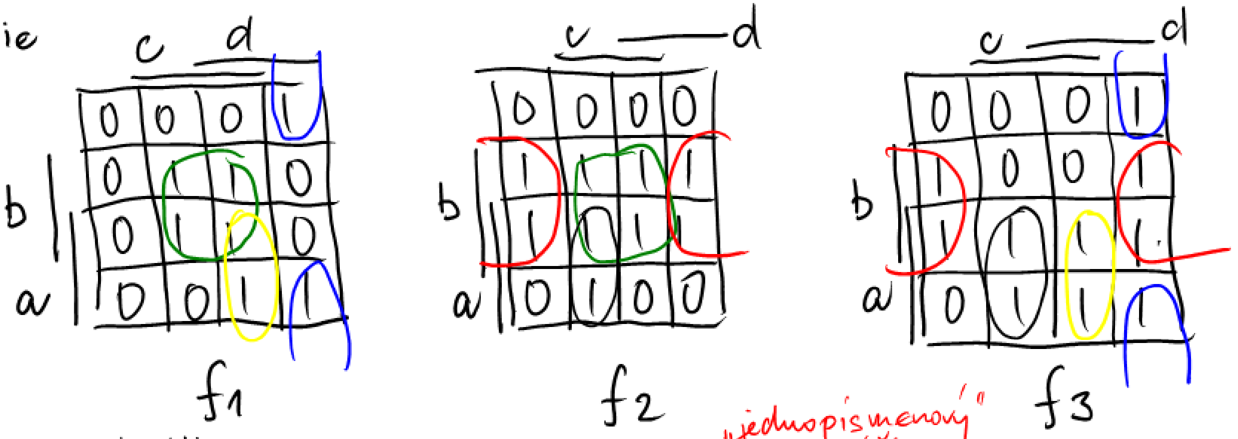
6 súčtov
3 súčiny
9 log. členov

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Vyjadrovanie minimálnych skupín DNF (KNF) z máp a návrh obvodu

Riešenie pre DNF:



f1:

0	0	0	1
0	1	1	0
0	1	1	0
0	0	1	1

f2:

0	0	0	0
1	1	1	1
1	1	1	1
0	1	0	0

f3:

0	0	0	1
1	0	0	1
1	1	1	1
0	1	1	1

DNF: jednotlivo

$$f_1 = bc + \bar{b}\bar{c}d + a\bar{b}d \qquad f_2 = b + ac\bar{d} \qquad f_3 = \bar{c}d + ac + b\bar{c}$$

„jednopísmenový" súčin

8 súčinov, 3 súčty $(7+\cancel{1})$

Jednotlivo: 10 logických členov

M$_s$DNF:

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

$$f_2 = bc + b\bar{c} + ac\bar{d}$$

$$f_3 = b\bar{c} + \bar{b}\bar{c}d + ac\bar{d} + acd$$

5 súčinov

3 súčty

8 logických členov

M$_s$DNF:

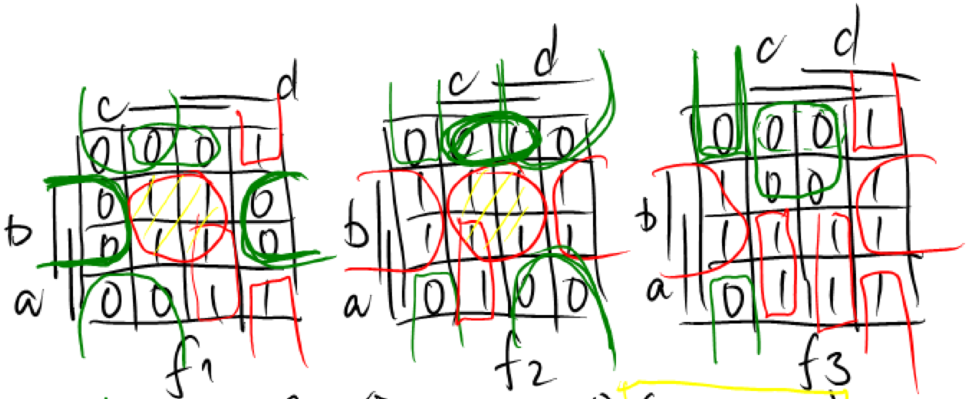
PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

Vyjadrovanie minimálnych skupín DNF (KNF) z máp a návrh obvodu

PRÍKLAD

Vykonajte skupinovú minimalizáciu pre funkcie f1, f2, f3 v tvare KNF a DNF:



f1:

0	0	0	1
0	1	1	0
0	1	1	0
0	0	1	1

f2:

0	0	0	0
1	1	1	1
1	1	1	1
0	1	0	0

f3:

0	0	0	1
1	0	0	1
1	1	1	1
0	1	1	1

NAND/NAND

AND/OR

NF g_1/g_2

Min. skupina KNF:

$$f_1 = (\bar{b}+c)(b+d)\boxed{(a+b+\bar{c})}$$

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

$$f_3 = (b+c+d)(a+\bar{c})$$

Min. skupina DNF:

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

$$f_2 = \boxed{bc} + b\bar{c} + acd$$

$$f_3 = b\bar{c} + \bar{b}\bar{c}d + ac\bar{d} + acd$$

Spotreba log. členov na realizáciu

Stupeň I.	6	5
II.	3	3
	↑	↑
	MsKNF	MsDNF

PPI 2. Logická úroveň a stavba počítačových systémov

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.2. ZMIEŠANÉ NORMÁLNE FORMY

Z NF sa dajú navrhnúť a zostaviť **dvojstupňové obvody**, ktoré sú **najrýchlejšie.**

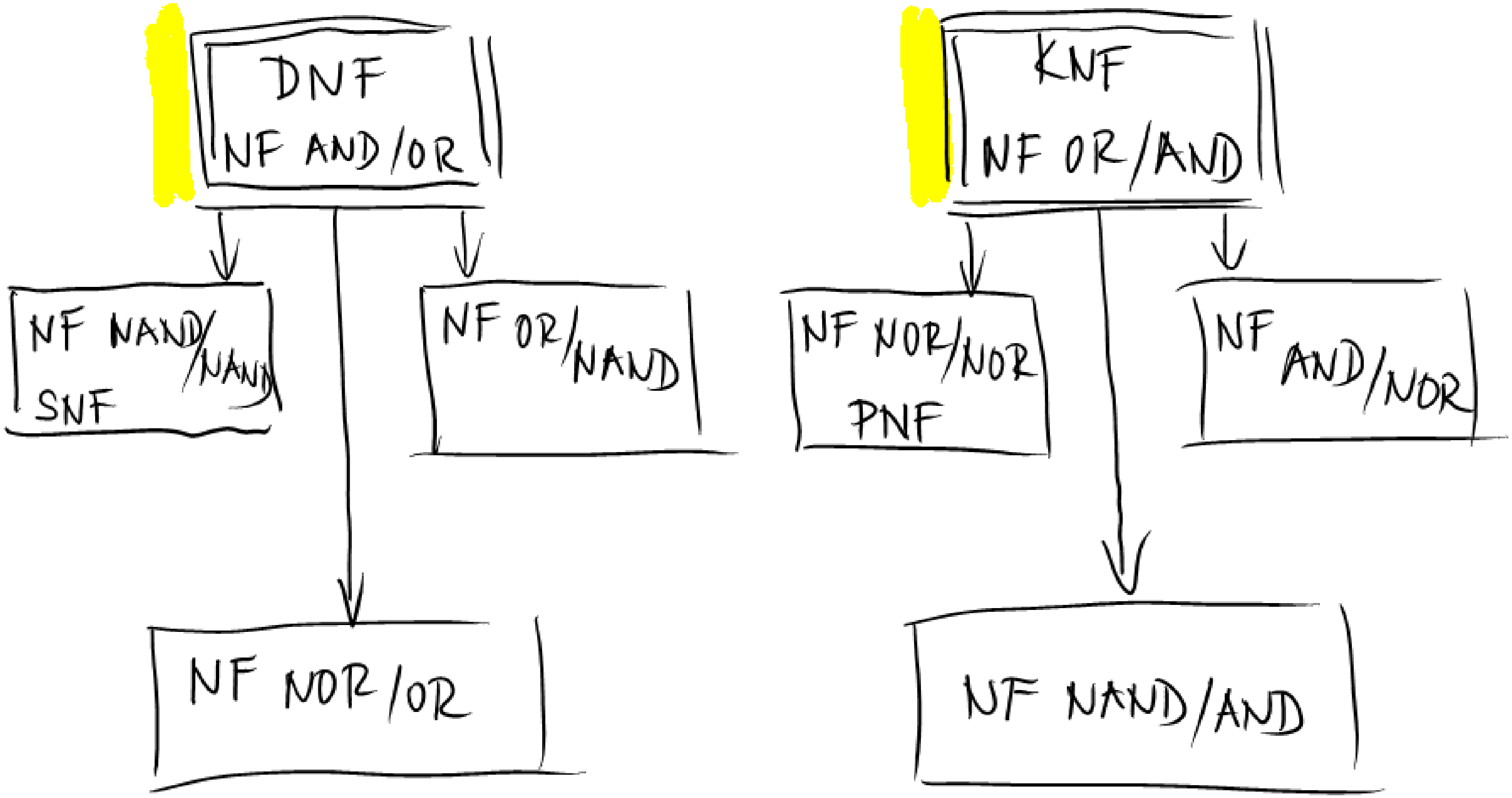
Predpoklady:

- sú k dispozícii negácie vstupných premenných
- je povolený neohraničený počet vstupov logických členov.

- Je možné DNF a KNF transformovať na ďalšie normálne formy NF s použitím logických členov NOR a NAND, čím vzniknú **zmiešané** NF.
- Základom pre ich tvorbu sú **DNF** a **KNF**.

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.2. ZMIEŠANÉ NORMÁLNE FORMY



Grafické znázornenie odvodených NF

2.1.3. SYNTÉZA LOGICKÝCH KOMBINAČNÝCH OBVODOV

2.1.3.2. ZMIEŠANÉ NORMÁLNE FORMY

PRÍKLAD – DNF:

napr. $y = a\bar{b} + \bar{c}d + e$

Odvodenie:

SNF: $y = (a \uparrow \bar{b}) \uparrow (\bar{c} \uparrow d) \uparrow \bar{e}$

NF NOR/OR: $y = (\bar{a} \downarrow b) + (c \downarrow \bar{d}) + e$

NF OR/NAND $y = (\bar{a} + b) \uparrow (c + \bar{d}) \uparrow \bar{e}$

PRÍKLAD - KNF: (iná funkcia)

napr. $y_1 = (\bar{a} + \bar{b}).(c + \bar{d}).e$

Odvodenie:

PNF: $y_1 = (\bar{a} \downarrow \bar{b}) \downarrow (c \downarrow \bar{d}) \downarrow \bar{e}$

NF NAND/AND: $y_1 = (a \uparrow b)(\bar{c} \uparrow d)e$

NF AND/NOR: $y_1 = ab \downarrow \bar{c}d \downarrow \bar{e}$