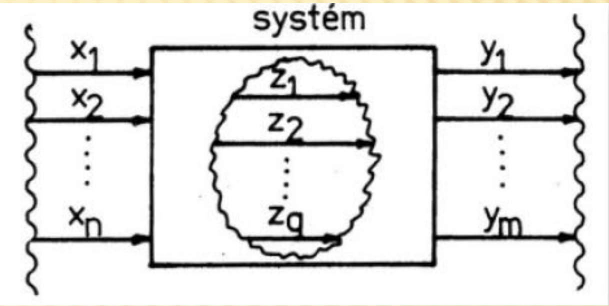
LOGICKÉ SYSTÉMY

Prednáška 2, 2014-2015

Ing. Adam Jaroš, PhD – prednášky, cvičenia
Ing. Michal Chovanec –cvičenia
Katedra technickej kybernetiky
Web predmetu: http://frtk.fri.uniza.sk

OPAKOVANIE – DISKRÉTNY SYSTÉM



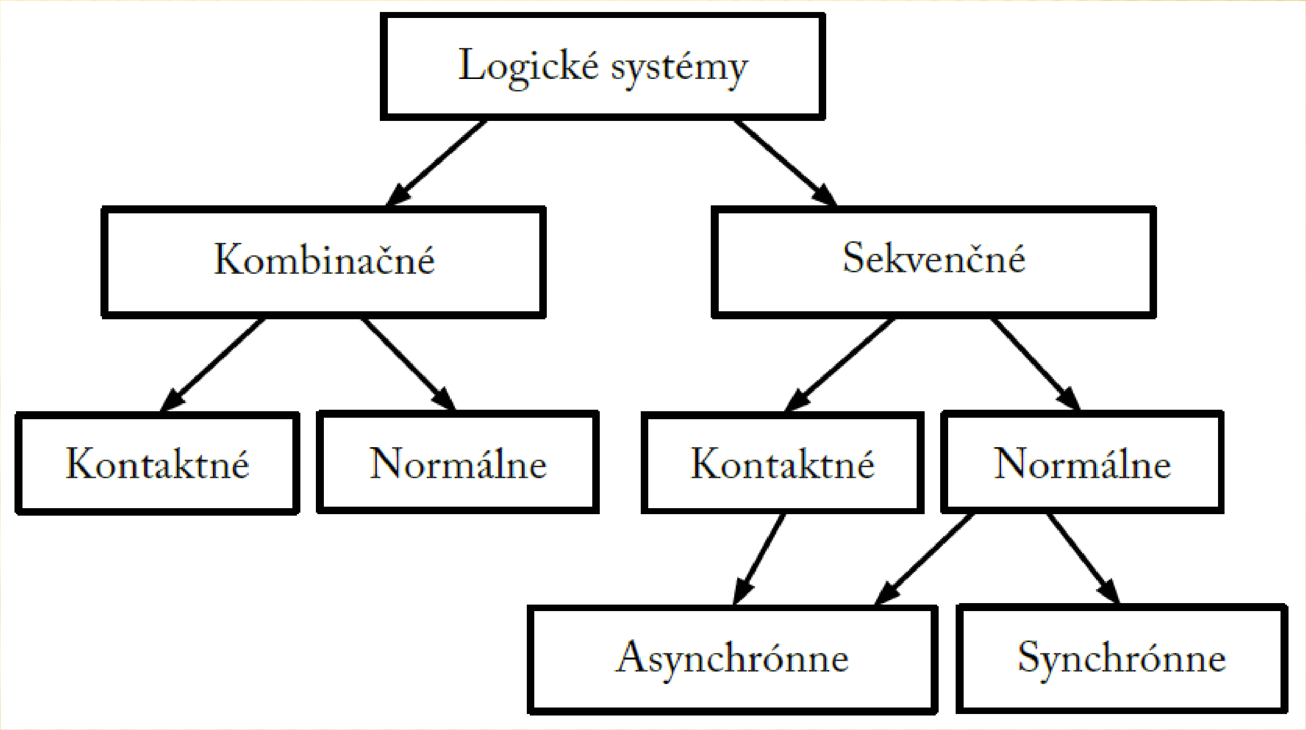
Diskrétny systém má konečný počet stavov, v ktorých sa môže nachádzať.

Budeme sa zaoberať výlučne *diskrétnymi systémami*, kde každý vstupný a výstupný symbol môže nadobúdať len dve hodnoty - *log. 0* a *log. 1 (true, false)*. *Číslicové logické systémy*–*logické systémy*.

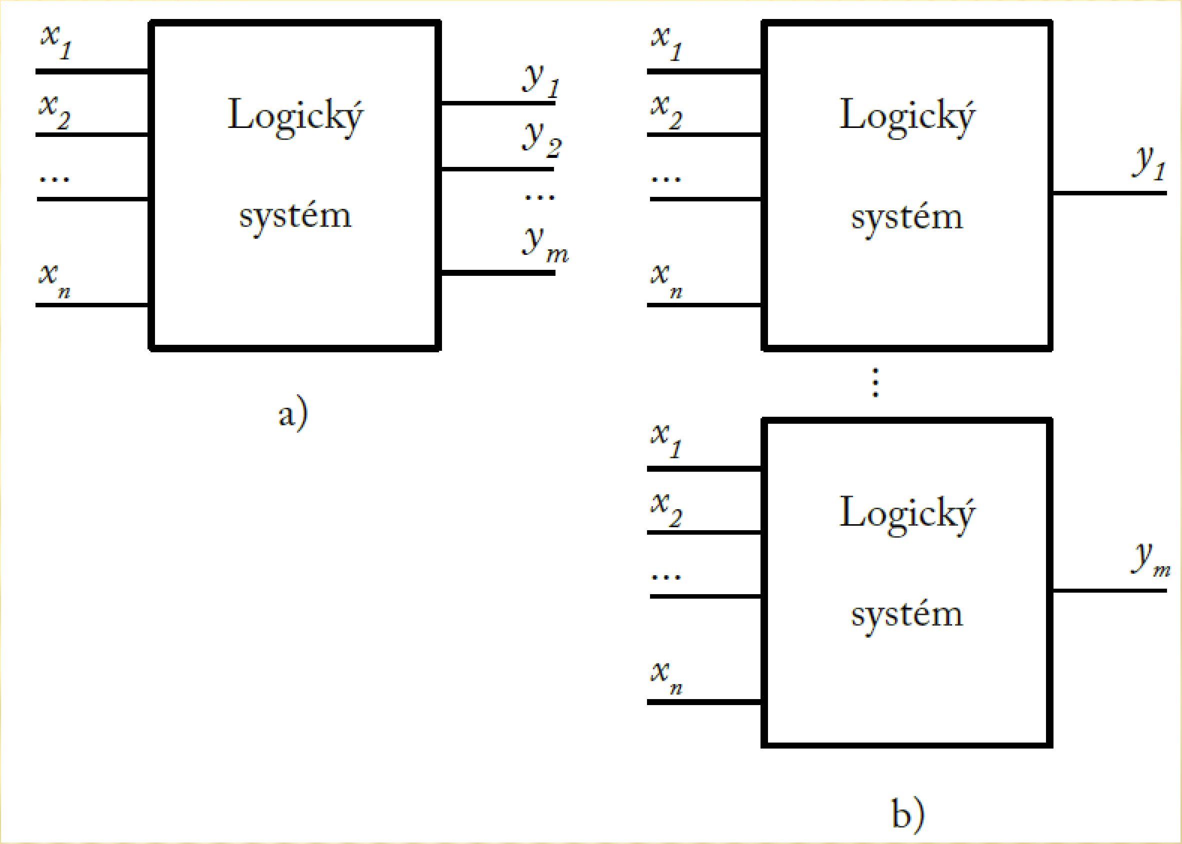
OPAKOVANIE – ROZDELENIE LOGICKÝCH SYSTÉMOV

Základné hľadiská.

Normálne sú *polovodičové* integrované obvody (IO).



OPAKOVANIE – KOMBINAČNÉ LOGICKÉ SYSTÉMY



Princíp dekompozície.

OPAKOVANIE – ZÁPIS SPRÁVANIA SA KOMBINAČNÝCH OBVODOV

Pravdivostná tabuľka

- tabuľkový zápis

Všetky kombinácie vstupných signálov - ***úplný zápis***.

V reálnych prípadoch je redukovaná - ***skrátený zápis***.

H_1	H_2	H_3	Výsledok
N	N	N	Z
N	N	A	Z
N	A	N	Z
N	A	A	P
A	N	N	Z
A	N	A	P
A	A	N	P
A	A	A	P

Príklad 1.2 **Hlasovací systém pre troch hlasujúcich**

– často sa vyskytujúce funkcie majú svoj názov: majorita z troch, značíme M3

OPAKOVANIE – ZÁPIS SPRÁVANIA SA KOMBINAČNÝCH OBVODOV

Publikačný spôsob zápisu

$deV \rightarrow 0 = \{0, 1, 2, 4\}$,

Pozn. vstupné hodnoty v tabuľke sú usporiadané
Podľa binárneho kódu vzostupne.

Zápis čítame: ***„dekadický ekvivalent, kedy hodnota výstupnej premennej „V“ vedie na nulu***“.

H_1	H_2	H_3	Výsledok
N	N	N	Z
N	N	A	Z
N	A	N	Z
N	A	A	P
A	N	N	Z
A	N	A	P
A	A	N	P
A	A	A	P

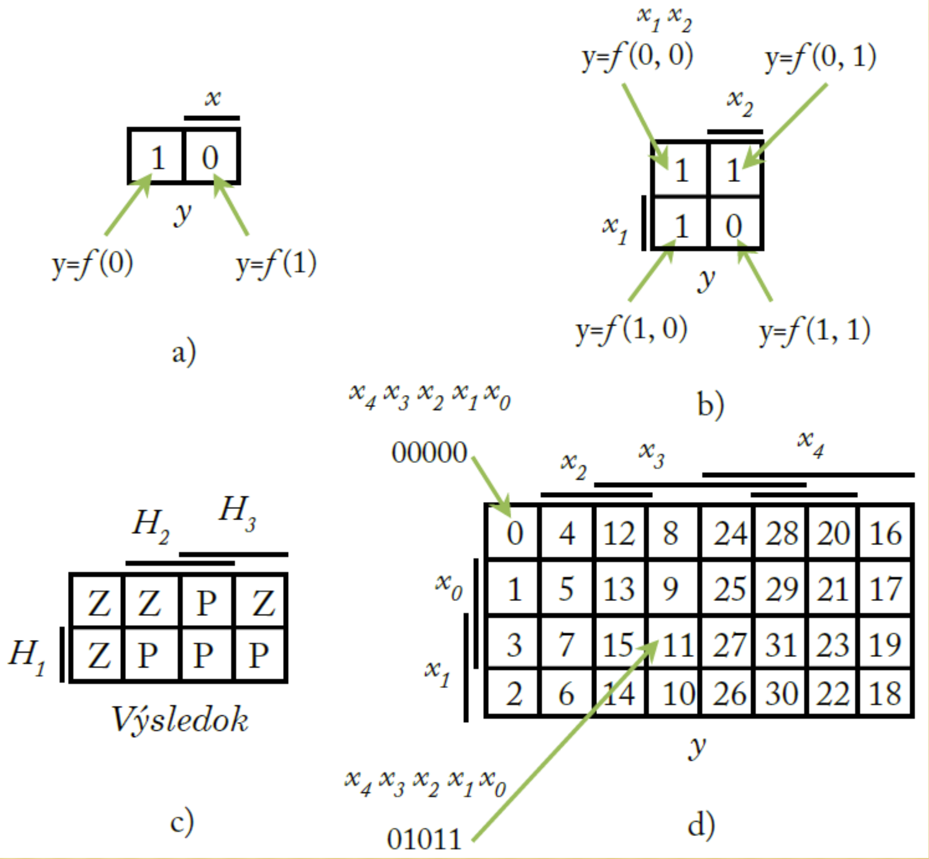
OPAKOVANIE – ZÁPIS SPRÁVANIA SA KOMBINAČNÝCH OBVODOV – KARAUGHOVA MAPA

Predstavuje grafickú reprezentáciu *úplnej pravdivostnej tabuľky*.

Vhodná pre malý počet premenných.

Vytvorenie K-mapy

- zrkadlovým preklopením mapy podľa ľubovoľnej hrany z mapy o jeden rád nižší.



OPAKOVANIE – ZÁKLADNÉ LOGICKÉ ČLENY–LOGICKÉ HRADLÁ

Musíme vedieť akú súčiastkovú základňu pri realizácii použijeme.

Logické obvody

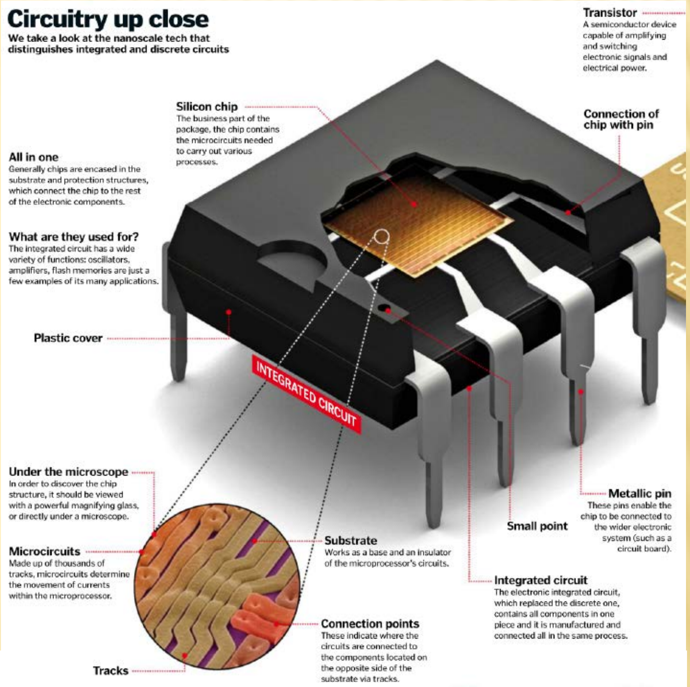
- s pevnou štruktúrou (predmet Logické systémy)
- s programovateľnou štruktúrou:
  - PLD - PAL, GAL programovateľné logické obvody (Programmable Logic Device, PLD)
  - FPLD - zložitejšie obvody, kde niekoľko uzavretých blokov podobných GAL je prepojených
  - FPGA - programovateľná verzia hradlového poľa, kde prepoje jednotlivých buniek poľa ako aj samotná funkcia malých buniek je daná obsahom RAM buniek, ktorý je do obvodu „nahraný“ obvykle po zapnutí napájania

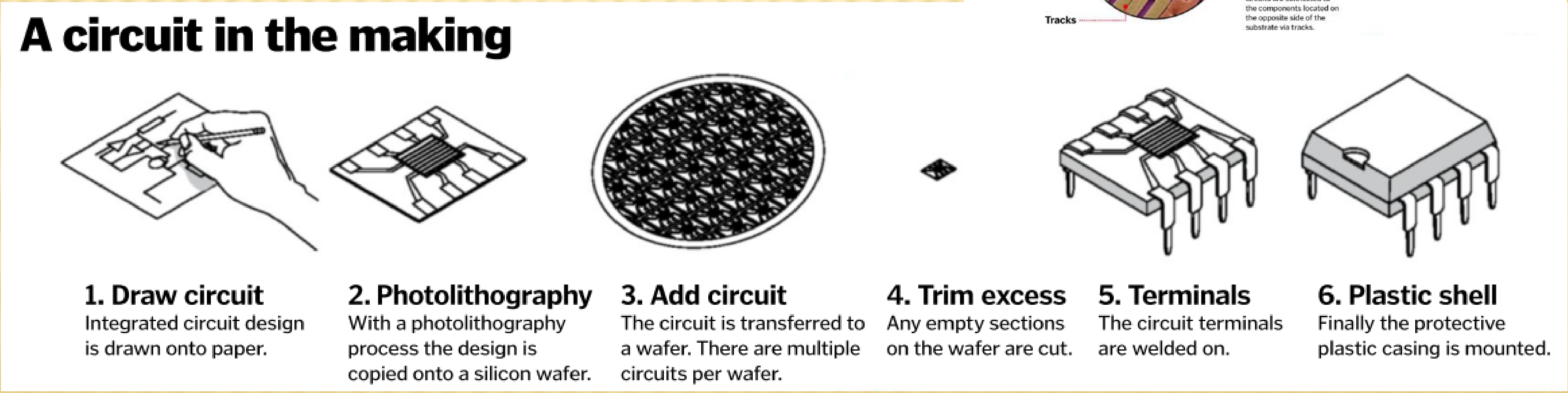
Návrhové prostriedky

- symbolický popis obvodu v jazyku podobnom vyšším programovacím jazykom; popis činnosti či štruktúry CPLD/FPGA s VHDL, Verilog a jazyk C
- schematický návrh podobný návrhu bežných elektronických obvodov

OPAKOVANIE – ZÁKLADNÉ LOGICKÉ ČLENY–LOGICKÉ HRADLÁ

My budeme používať z pohľadu zložitosti obvody tzv. nízkej integrácie (približne 100 tranzistorov).



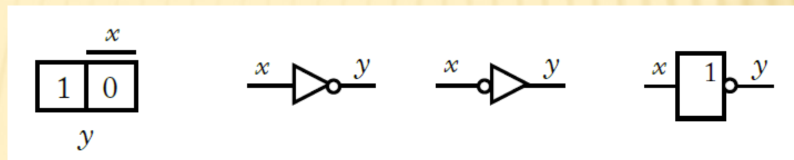


OPAKOVANIE – LOGICKÉ HRADLÁ

Negácia alebo inverzia (skratka INV alebo NOT)

Zápis logickej funkcie: $y = \overline{x}$

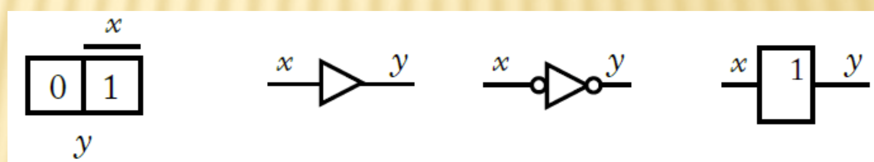
Schematická značka:



Buffer (skratka BUF)

Zápis logickej funkcie: $y = x$

Schematická značka:



OPAKOVANIE – LOGICKÉ HRADLÁ

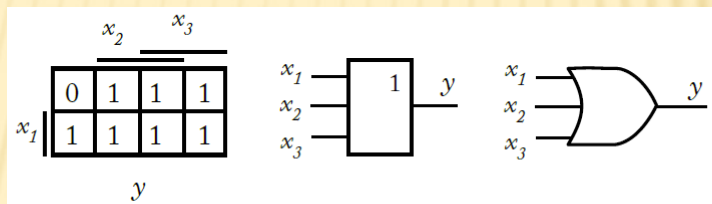
Logický súčet (skratka OR)

Zápis logickej funkcie:

$$y = x_1 + x_2 + \cdots + x_n$$

$$y = x_1 \vee x_2 \vee \ldots \vee x_n$$

Schematická značka:

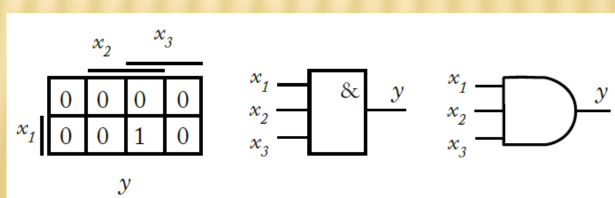


Logický súčin (skratka AND)

Zápis logickej funkcie:

$$y = x_1 \cdot x_2 \cdot \ldots \cdot x_n$$

Schematická značka:



OPAKOVANIE – LOGICKÉ HRADLÁ

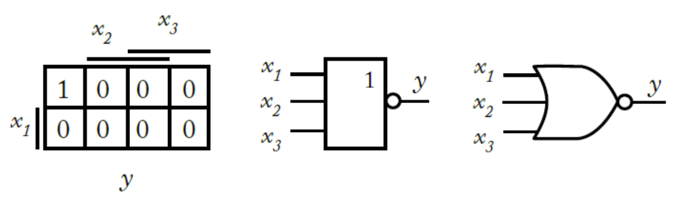
Negácia logického súčtu (skratka NOR), $\boxed{\downarrow\ (\textit{Pierceov operátor})}$

Zápis logickej funkcie:

$$y = \overline{x_1 + x_2 + \cdots + x_n}$$

$$y = \overline{x_1 \vee x_2 \vee \ldots \vee x_n} = x_1 \downarrow x_2 \downarrow \cdots \downarrow x_n$$

Schematická značka:

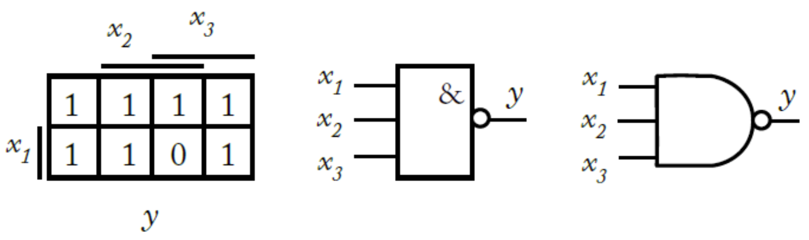


Negácia logického súčinu (skratka NAND), $\boxed{|\ (\textit{Shafferov operátor})}$

Zápis logickej funkcie:

$$y = \overline{x_1 \cdot x_2 \cdot \ldots \cdot x_n} = x_1 | x_2 | \ldots | x_n$$

Schematická značka:



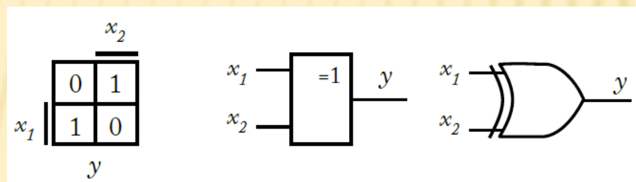
OPAKOVANIE – LOGICKÉ HRADLÁ

Neekvivalencia, nerovnoznačnosť (skratka XOR, eXclusive OR) - kryptografia

Zápis logickej funkcie: $y = x_1 \oplus x_2 \oplus \dots \oplus x_n$

pre dve premenné: $y = x_1 \oplus x_2 = \overline{x}_1 \cdot x_2 + x_1 \cdot \overline{x}_2$

Schématická značka:

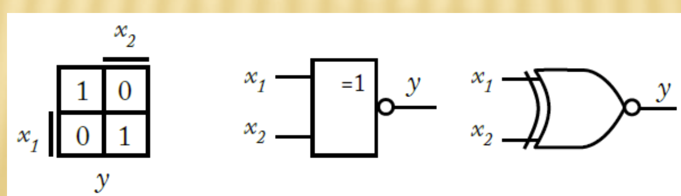


Ekvivalencia, rovnoznačnosť (skratka XNOR , eXclusive NOR)

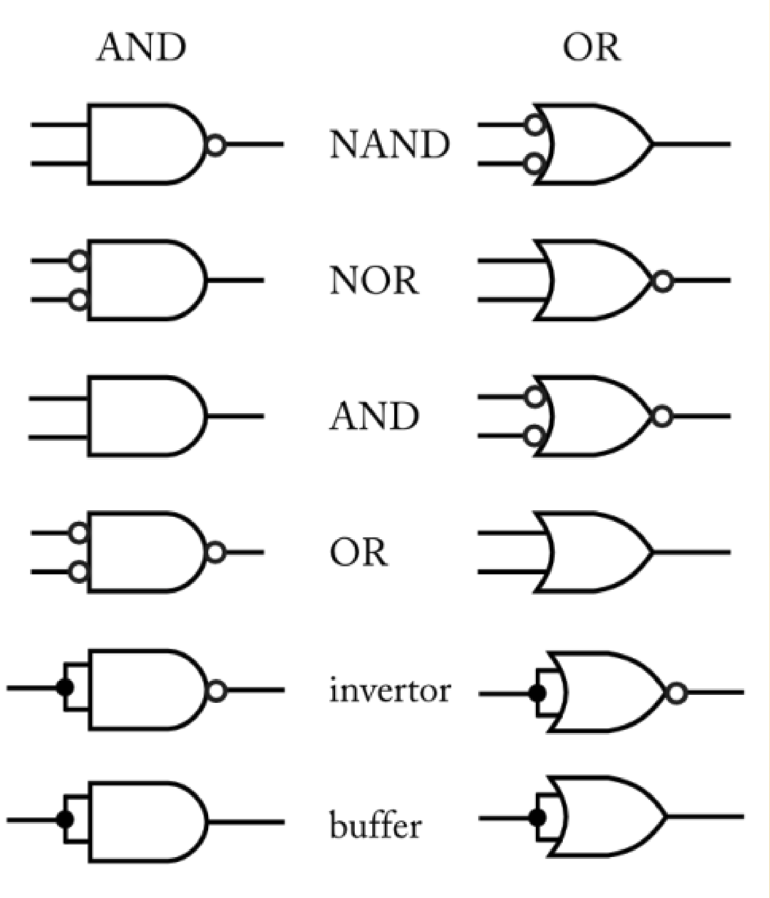
Zápis logickej funkcie: $y = x_1 \odot x_2 \odot \dots \odot x_n$

pre dve premenné: $y = x_1 \odot x_2 = \overline{x}_1 \cdot \overline{x}_2 + x_1 \cdot x_2 = (\overline{x}_1 + x_2) \cdot (x_1 + \overline{x}_2)$

Schématická značka:

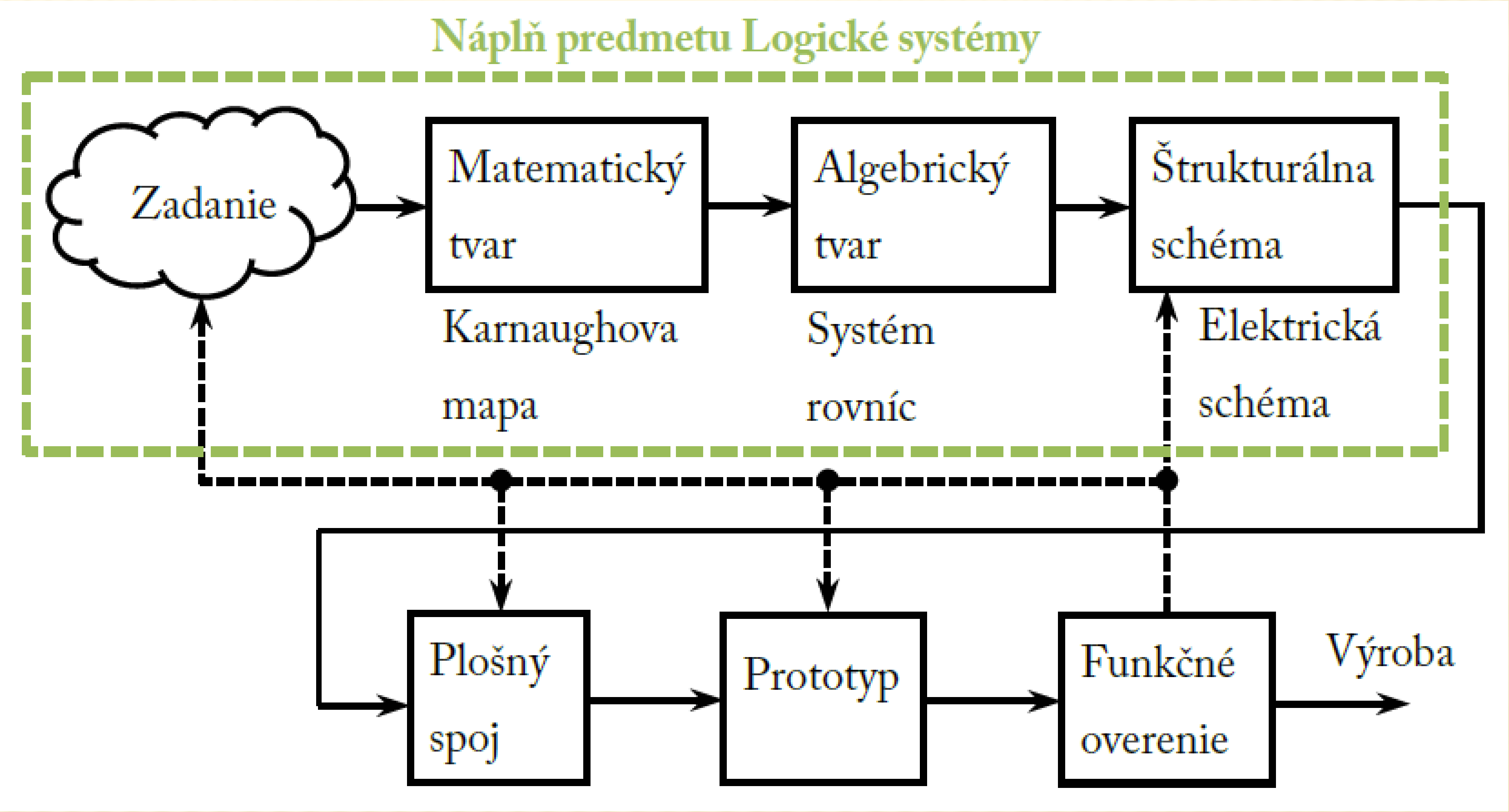


OPAKOVANIE – EKVIVALENTNÉ ZAPOJENIA



Aplikovaním Booleovej algebry, zákonov overte rovnosť zapojení.

OPAKOVANIE – FÁZY VÝVOJA ČÍSLICOVÉHO LOGICKÉHO SYSTÉMU



OPAKOVANIE – BOOLEOVA ALGEBRA

Zákony Booleovej algebry.

$a+a=a$ $\quad\quad$ $a.a=a$

Zákon absorpcie:

$a+a.b=a$ $\quad\quad$ $a.(a+b)=a$

Zákon absorpcie negácie:

$a+\bar{a}.b=a+b$ $\quad\quad$ $a.(\bar{a}+b)=a.b$

Distributívny zákon:

$a+(b.c)=(a+b).(a+c)$ $\quad\quad$ $a.(b+c)=a.b+a.c$

Napr.: $a+(a.b)=a$ $\quad\quad$ $a.(a+b)=a$

$a.b+\bar{a}.b=b$ $\quad\quad$ $(a+b).(\bar{a}+b)=b$

Neutrálnosť nuly a jednotky:

$a+0=a$ $\quad\quad$ $a.1=a$

Agresívnosť nuly a jednotky:

$a+1=1$ $\quad\quad$ $a.0=0$

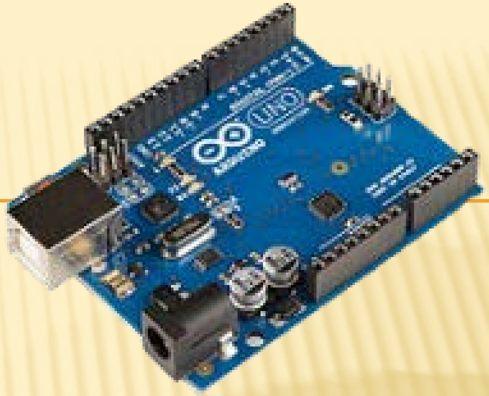
Zákon vylúčenia tretieho:

$a+\bar{a}=1$ $\quad\quad$ $a.\bar{a}=0$

De Morganove zákony:

$\overline{a+b}=\bar{a}.\bar{b}$ $\quad\quad$ $\overline{a.b}=\bar{a}+\bar{b}$

PREDNÁŠKA 2 ARDUINO



Témy prednášky:

1) *Normálne disjunktívne formy*
2) *Normálne konjunktívne formy*
3) *Kreslenie elektrickej (štrukturálnej) schémy*
4) *Oneskorenia v logických obvodoch*
5) *Kontaktné systémy*
6) *Minimalizácia logických výrazov*
7) *Vytváranie pravidelných konfigurácii v mape (grafická metóda)*
8) *Metóda Quine-Mc Cluskeyho*
9) *Neúplne definované logické funkcie*

ZÁPIS KARNAUGHOVEJ MAPY DO ALGEBRICKEJ FORMY

Karnaughovu mapu možno popísať viacerými spôsobmi.

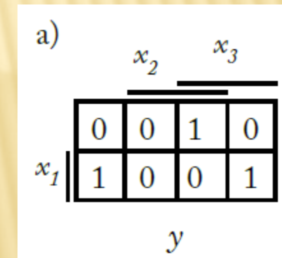
Najčastejší spôsob je **popis „jednotiek“ - disjunktívna forma**
alebo **popis „núl“ - konjunktívna forma**.

Na základe **voľby použitých logických hradiel** ďalej upravujeme získanú formu.

Častá požiadavka je implementácia riešenia s použitím jediného typu logických obvodov (hradiel). To spĺňajú logické funkcie: NAND, NOR a XOR.

DISJUNKTÍVNE FORMY

Príklad 1: Zapíšte disjunktívnu formu Karnaughovej mapy, kde *y=f* (*h1*, *h2*, *h3*).



Riešenie:

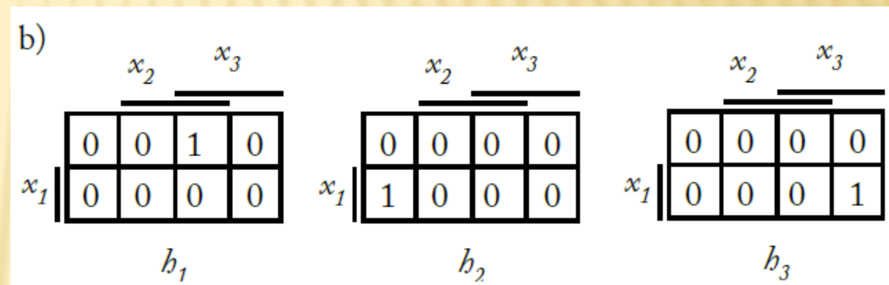
Zjednotením funkcií *h1*, *h2* a *h3* získame *y*.

Môžeme teda zapísať

$$y = h1 + h2 + h3$$

alebo

$$y = h1 \vee h2 \vee h3$$



Samostatné jednotky v mapách popíšeme disjunkciou:

$$h1 = \overline{x}1 \cdot x2 \cdot x3 \qquad h2 = x1 \cdot \overline{x}2 \cdot \overline{x}3 \qquad h3 = x1 \cdot \overline{x}2 \cdot x3$$

Po dosadení do výrazu pre *y* dostávame

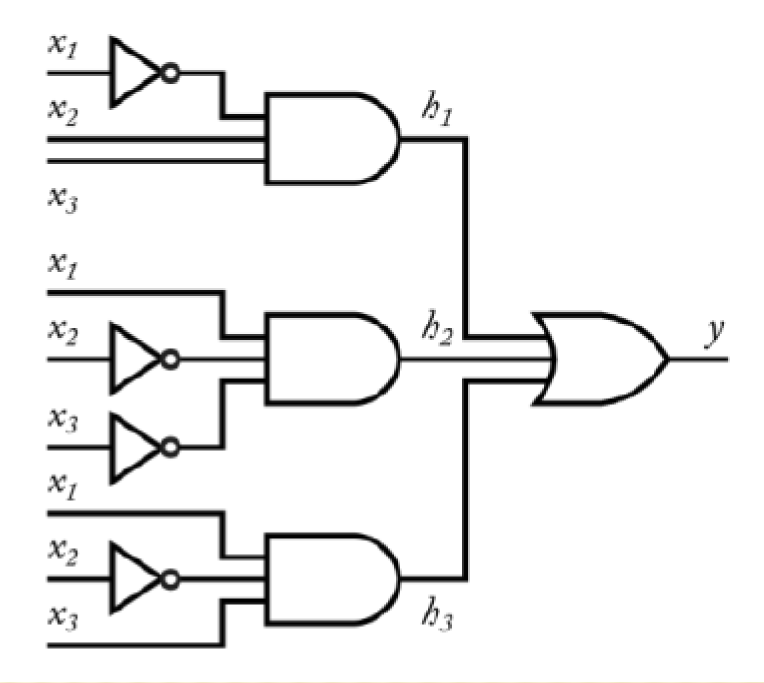
$$y = \overline{x}1 \cdot x2 \cdot x3 + x1 \cdot \overline{x}2 \cdot \overline{x}3 + x1 \cdot \overline{x}2 \cdot x3$$

Takýto algebrický zápis nazývame *úplná normálna disjunktívna forma* – ÚNDF

NORMÁLNA SIEŤ – *ELEKTRICKÁ SCHÉMA*

Elektrická schéme *úplnej normálnej disjunktívnej formy:*

$$y = \overline{x1}{\cdot}x2{\cdot}x3 + x1{\cdot}\overline{x2}{\cdot}\overline{x3} + x1{\cdot}\overline{x2}{\cdot}x3$$



Vlastnosti normálnej siete sú:

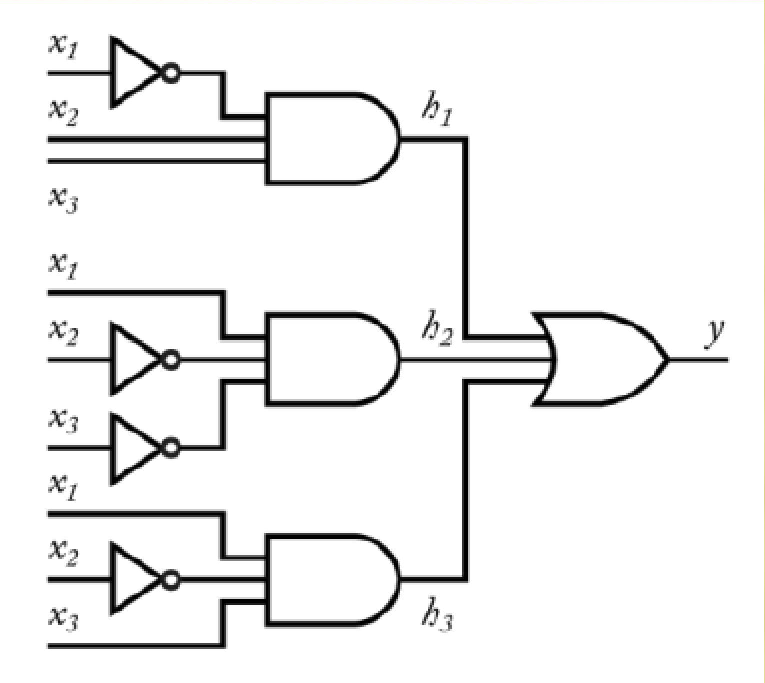
- je bez „spätnej“ väzby
- obsahuje vetvenie signálov (fan-out)
- zaťažiteľnosť výstupov (je limitovaná)

ONESKORENIE NORMÁLNEJ SIETE

Pri zmene hodnôt nezávislých (vstupných) premenných sa výstup logického obvodu nezmení okamžite.

Je k tomu potrebný určitý čas.

Zjednodušenie: uvažujeme „*jednotkové*“ oneskorenie pre každé logické hradlo.



MINIMALIZÁCIA LOGICKÝCH VÝRAZOV

Minimalizácia zložitosti elektrickej schémy je **významnou požiadavkou** pri vytváraní logických obvodov.

Spôsoby minimalizácie:

- úprava logických výrazov s použitím pravidiel Booleovej algebry
- hľadanie takého zápisu hodnôt Karnaughovej mapy, ktorý je minimálny

<u>Cieľ:</u>

Znižujeme tak rozmery, nároky na výkon výstupov (fan-out), vyžarovanie tepla a cenu. Nevýhody?

PRAVIDELNÁ KONFIGURÁCIA V MAPE (GRAFICKÁ METÓDA)

Pravidelná konfigurácia v Karnaughovej mape zahŕňa skupinu bodov s rovnakou hodnotou. Stupeň pravidelnej konfigurácie označíme ***s***.

Vlastnosti pravidelnej konfigurácie:

- zahŕňa práve 2^s bodov,
- každý bod má práve s - susedných bodov, ktoré sú súčasťou konfigurácie.

Dva body sú susedné, keď sa líšia v hodnote jednej premennej.

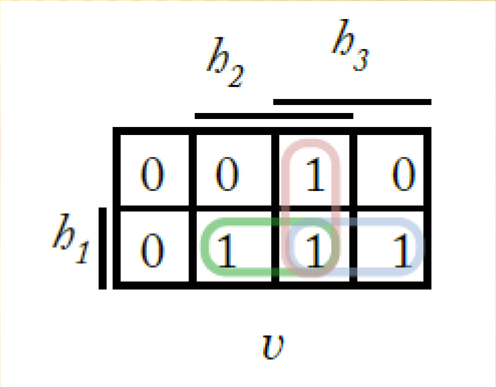
Nech je n počet premenných, R rád súčinu a s je stupeň konfigurácie.
Potom platí

$$R = n - s$$

PRAVIDELNÁ KONFIGURÁCIA V MAPE (GRAFICKÁ METÓDA)

Príklad 2:

Nájdite optimálne (pravidelné) konfigurácie v Karnaughovej mape funkcie M3 (majorita z troch, príklad hlasovacieho systému).



Riešenie:

Zapíšme ÚNDF pre funkciu M3

$$v = h1 \cdot h2 \cdot \overline{h3} + \overline{h1} \cdot h2 \cdot h3 + h1 \cdot h2 \cdot h3 + h1 \cdot \overline{h2} \cdot h3$$

a teraz výraz pre pravidelné konfigurácie (sú zakreslené farebne):

$$v = h1 \cdot h2 + h2 \cdot h3 + h1 \cdot h3$$

ktorý predstavuje zároveň *optimálne* konfigurácie.

Ak vieme, že sme vytvorili „**najlepšie**“ - optimálne konfigurácie tak hovoríme o *iredudantnej normálnej disjunktívnej forme* – INDF.

Inak hovoríme o ***normálnej disjunktívnej forme - NDF***.

METÓDA QUINE – MC CLUSKEY PRE MINIMALIZÁCIU LOGICKÉHO VÝRAZU

Pri určovaní optimálnych konfigurácií v počítači je grafická metóda nevhodná.

Autori Quine a Mc Cluskey zostavili tabuľkovú metódu, ktorá je prehľadná a hľadanie konfigurácií pozostáva z niekoľkých krokov.

Popis metódy v učebnici *Logické systémy, 2. vydanie z roku 1986*
od autorov Frištacký, Kolesár a kol.

NORMÁLNE FORMY

Naším cieľom je realizácia ***normálnej siete*** s použitím jediného typu logických členov.

Ukážme, že logické funkcie NAND respektíve NOR k tomu postačujú a predstavujú tak *úplný systém logických funkcií*.

Logická funkcia NAND

$\overline{a \cdot b} = a|b$

vytvorenie negácie: $\overline{a \cdot a} = a|a = \bar{a} = a|$

vytvorenie logického súčtu: $(a|\,)|(b|\,) = \bar{a}|\bar{b} = \overline{\bar{a} \cdot \bar{b}} = \bar{\bar{a}} \vee \bar{\bar{b}} = a \vee b$

vytvorenie logického súčinu: $(a|b)| = \overline{\overline{a \cdot b}} = a \cdot b$

Logická funkcia NOR

$\overline{x_1 \vee x_2} = x_1 \downarrow x_2$

vytvorenie negácie: $\overline{a \vee a} = a \downarrow a = \bar{a} = a \downarrow$

vytvorenie logického súčinu: $(a \downarrow) \downarrow (b \downarrow) = \overline{\bar{a} \vee \bar{b}} = \overline{\overline{a \cdot b}} = a \cdot b$

vytvorenie logického súčtu: $(a \downarrow b) \downarrow = \overline{\overline{a \vee b}} = a \vee b$

1. NORMÁLNA SHAFFEROVA FORMA

Zápis NDF(INDF) prevedieme do **Shafferovej** a **Pierceovej** funkcie úpravami výrazu podľa pravidiel Booleovej algebry a použitím De Morganových zákonov.

1. Normálna Shafferova forma (1. NSF), log. funkcia NAND

$$
\begin{aligned}
y &= (x_{11}|x_{12}|\ldots|x_{1a})|(x_{21}|x_{22}|\ldots|x_{2b})|\ldots|(x_{n1}|x_{n2}|\ldots|x_{nm}) \\
&= \overline{(\overline{x_{11}\cdot x_{12}\cdot\ldots\cdot x_{1a}})\cdot(\overline{x_{21}\cdot x_{22}\cdot\ldots\cdot x_{2b}})\cdot\ldots\cdot(\overline{x_{n1}\cdot x_{n2}\cdot\ldots\cdot x_{nm}})} \\
&= (\overline{\overline{x_{11}\cdot x_{12}\cdot\ldots\cdot x_{1a}}})+(\overline{\overline{x_{21}\cdot x_{22}\cdot\ldots\cdot x_{2b}}})+\cdots+(\overline{\overline{x_{n1}\cdot x_{n2}\cdot\ldots\cdot x_{nm}}}) \\
&= (x_{11}\cdot x_{12}\cdot\ldots\cdot x_{1a})+(x_{21}\cdot x_{22}\cdot\ldots\cdot x_{2b})+\cdots+(x_{n1}\cdot x_{n2}\cdot\ldots\cdot x_{nm})
\end{aligned}
$$

Pravidlá pre prepis NDF(INDF) do 1. NSF:

- súčiny uzavrieme do zátvoriek
- všetky operátory nahradíme Shafferovým operátorom

<u>**Výnimky:**</u>

Ak v NDF zápis nie je „úplný“ vieme si ho ľahko doplniť (jedná sa o prípady: jediná premenná v súčine; chýbajúci logický súčet aspoň dvoch logických súčinov).

2. NORMÁLNA PIERCEOVA FORMA

2. Normálna Pierceova forma (2. NPF), log. funkcia NOR

$$
\begin{aligned}
y &= [(x_{11} \downarrow x_{12} \downarrow \cdots \downarrow x_{1a}) \downarrow (x_{21} \downarrow x_{22} \downarrow \cdots \downarrow x_{2b}) \downarrow \cdots \\
&\qquad \downarrow (x_{n1} \downarrow x_{n2} \downarrow \cdots \downarrow x_{nm})] \downarrow \\
&= \overline{\overline{(\overline{x_{11} \vee x_{12} \vee \ldots \vee x_{1a}}) \vee (\overline{x_{21} \vee x_{22} \vee \ldots \vee x_{2b}}) \vee \ldots \vee (\overline{x_{n1} \vee x_{n2} \vee \ldots \vee x_{nm}})}} \\
&= (\overline{x_{11} \vee x_{12} \vee \ldots \vee x_{1a}}) \vee (\overline{x_{21} \vee x_{22} \vee \ldots \vee x_{2b}}) \vee \ldots \vee (\overline{x_{n1} \vee x_{n2} \vee \ldots \vee x_{nm}}) \\
&= (\bar{x}_{11} \cdot \bar{x}_{12} \cdot \ldots \cdot \bar{x}_{1a}) + (\bar{x}_{21} \cdot \bar{x}_{22} \cdot \ldots \cdot \bar{x}_{2b}) + \cdots + (\bar{x}_{n1} \cdot \bar{x}_{n2} \cdot \ldots \cdot \bar{x}_{nm})
\end{aligned}
$$

Pravidlá pre prepis NDF(INDF) do 1. NSF:

- súčiny uzavrieme do zátvoriek
- všetky operátory nahradíme Pierceovým operátorom
- negujeme každdú premennú
- na celý výraz aplikujeme Pierceov operátor (operácia negácie)

Výnimky:

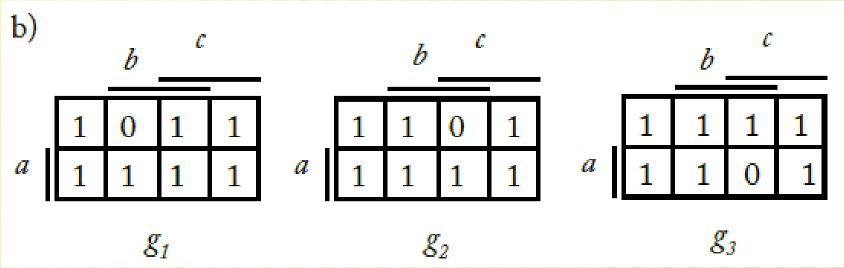
Pravidlá sú rovnaké ako pri 1. NSF.

KONJUNKTÍVNE FORMY

Príklad 3:
Zapíšte konjunktívnu formu nasledovnej Karnaughovej mapy, kde $y=f\,(a, b, c)$.

Riešenie:
Prienikom $g1$, $g2$ a $g3$ získame y.
Môžeme teda zapísať

$$y=g1\cdot g2\cdot g3.$$



Samostatné nuly v mapách popíšeme konjunkciou, teda

$$g1=a\vee\overline{b}\vee c \qquad g2=a\vee\overline{b}\vee\overline{c} \qquad g=\overline{a}\vee\overline{b}\vee\overline{c}$$

Po dosadení do výrazu pre y dostávame

$$y=(a\vee\overline{b}\vee c)\cdot(a\vee\overline{b}\vee\overline{c})\cdot(\overline{a}\vee\overline{b}\vee\overline{c})$$

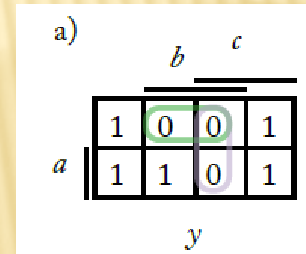
Algebrický zápis nazývame *úplná normálna konjunktívna forma* – ÚNKF.
Výrazy v zátvorkách voláme pre ÚNKF *mintermy* a pri ÚNDF *maxtermy*.

KONJUNKTÍVNE FORMY

Konfigurácie v Karnaughovej mape z „núl“.

Príklad:

Zapíšte optimálne konfigurácie v *Karnaughovej mape*.

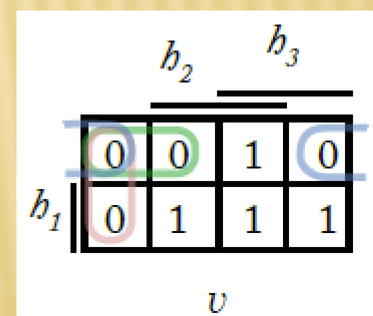


$$y = (a \vee \bar{b}) \cdot (\bar{b} \vee \bar{c})$$

Tento výraz predstavuje minimálnu konjunktívnu formu a označujeme ho *iredundantná normálna konjunktívna forma* – INKF. Inak hovoríme o ***normálnej konjunktívnej forme - NKF***.

Príklad 4:

Nájdite INKF v Karnaughovej mape funkcie M3.



Riešenie:

INKF pre funkciu M3 je nasledovný

$$v = (h2 \vee h3) \cdot (h1 \vee h2) \cdot (h1 \vee h3)$$

1. NORMÁLNA PIERCEOVA FORMA

Zápis NKF(INKF) prevedieme do **Pierceovej** a **Shafferovej** funkcie úpravami výrazu podľa pravidiel Booleovej algebry a použitím De Morganových zákonov.

1. Normálna Pierceova forma (1. NPF) , log. funkcia NOR

$$
\begin{aligned}
y &= (x_{11} \downarrow x_{12} \downarrow \cdots \downarrow x_{1a}) \downarrow (x_{21} \downarrow x_{22} \downarrow \cdots \downarrow x_{2b}) \downarrow \cdots \\
&\qquad \downarrow (x_{n1} \downarrow x_{n2} \downarrow \cdots \downarrow x_{nm}) \\
&= \overline{(\overline{x_{11} \vee x_{12} \vee \ldots \vee x_{1a}}) \vee (\overline{x_{21} \vee x_{22} \vee \ldots \vee x_{2b}}) \vee \ldots \vee (\overline{x_{n1} \vee x_{n2} \vee \ldots \vee x_{nm}})} \\
&= (\overline{\overline{x_{11} \vee x_{12} \vee \ldots \vee x_{1a}}}) \cdot (\overline{\overline{x_{21} \vee x_{22} \vee \ldots \vee x_{2b}}}) \cdot \ldots \cdot (\overline{\overline{x_{n1} \vee x_{n2} \vee \ldots \vee x_{nm}}}) \\
&= (x_{11} \vee x_{12} \vee \ldots \vee x_{1a}) \cdot (x_{21} \vee x_{22} \vee \ldots \vee x_{2b}) \cdot \ldots \cdot (x_{n1} \vee x_{n2} \vee \ldots \vee x_{nm})
\end{aligned}
$$

Pravidlá pre prepis NKF(INKF) do 1. NPF:

- súčiny uzavrieme do zátvoriek
- všetky operátory nahradíme Pierceovým operátorom

Výnimky:

Ak v NKF zápis nie je „úplný“ vieme si ho ľahko doplniť (jedná sa o prípady: jediná premenná v súčte; chýbajúci logický súčin aspoň dvoch logických súčtov−zátvoriek).

2. NORMÁLNA SHAFFEROVA FORMA

2. Normálna Shafferova forma (2. NSF), log. funkcia NAND

$$
\begin{aligned}
y &= [(x_{11}|x_{12}|\dots|x_{1a})|(x_{21}|x_{22}|\dots|x_{2b})|\dots(x_{n1}|x_{n2}|\dots|x_{nm})]| \\
&= \overline{\overline{(\overline{x_{11}\cdot x_{12}\cdot\dots\cdot x_{1a}})\cdot(\overline{x_{21}\cdot x_{22}\cdot\dots\cdot x_{2b}})\cdot\dots\cdot(\overline{x_{n1}\cdot x_{n2}\cdot\dots\cdot x_{nm}})}} \\
&= (\overline{x_{11}\cdot x_{12}\cdot\dots\cdot x_{1a}})\cdot(\overline{x_{21}\cdot x_{22}\cdot\dots\cdot x_{2b}})\cdot\dots\cdot(\overline{x_{n1}\cdot x_{n2}\cdot\dots\cdot x_{nm}}) \\
&= (\bar{x}_{11}+\bar{x}_{12}+\dots+\bar{x}_{1a})\cdot(\bar{x}_{21}+\bar{x}_{22}+\dots+\bar{x}_{2b})\cdot\dots \\
&\qquad \cdot(\bar{x}_{n1}+\bar{x}_{n2}+\dots+\bar{x}_{nm})
\end{aligned}
$$

Pravidlá pre prepis NKF(IKDF) do 2. NSF:

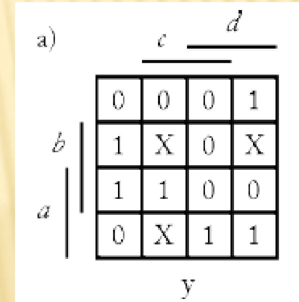
- súčty uzavrieme do zátvoriek
- všetky operátory nahradíme Shafferovým operátorom
- negujeme každdú premennú
- na celý výraz aplikujeme Shafferovým operátor (operácia negácie)

<u>Výnimky:</u>

Pravidlá sú rovnaké ako pri 1. NPF.

NEÚPLNE DEFINOVANÁ LOGICKÁ FUNKCIA

V praxi sa často stretávame s prípadmi, kedy výstup nie je definovaný pre všetky možné kombinácie vstupných hodnôt.



Potom zápis pravdivostnej tabuľky je *redukovaný* a v Karnaughovej mape máme „***prázdne***“ miesta.

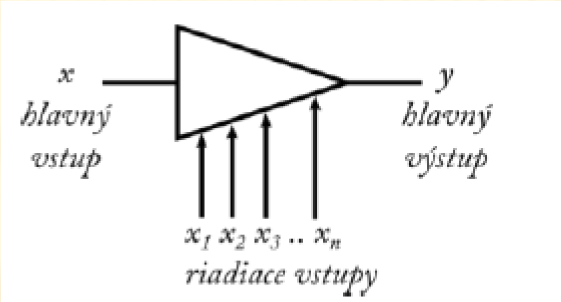
Tieto prípady umožňujú návrhárovi vhodne „***dodefinovať***“ prázdne miesta a to tak, aby sme dosiahli zjednodušenie riešenia.

V Karnaughovej mape si príslušné prázdne miesta označíme symbolom *X* (krížik). Určenie hodnoty tak prevedieme až pri vytváraní pravidelných konfigurácií.

KONTAKTNÉ SYSTÉMY

Stavebné prvky: ***kontakty***.

Reprezentácia logických úrovní: log. 0 – „*kľudový stav*“ (tlačidlo je uvoľnené), log. 1 – „*akcia*“.



Základnými stavebnými prvkami sú **invertor**, **logický súčet** a **logický súčin**.