

# 10.kapitola

## Boolove funkcie a logické obvody

### 10.1 Boolova algebra

Elektronické obvody v počítačoch a v podobných zariadeniach sú charakterizované binárnymi vstupmi a výstupmi (rovnajúcimi sa 0 alebo 1), transformácia vstupu na výstupu sa uskutočňuje prostredníctvom elektronického obvodu, ktorý tvorí jadro tohto „transformačného“ zariadenia, pozri Obr. 10.1. Elektronický obvod môže byť formálne simulovaný Boolovou funkciou, ktorá transformuje  $m$  binárnych vstupných premenných na  $n$  výstupných binárnych premenných.



**Obrázok 10.1.** Známenie elektronického obvodu, ktorý má  $m$  binárnych vstupov a  $n$  binárnych výstupov. Činnosť elektronického obvodu spočíva v transformácii binárnych vstupných hodnôt na binárne výstupné hodnoty.

Všeobecná definícia Boolovej funkcie je  $f : \{0,1\}^m \rightarrow \{0,1\}^n$ , táto funkcia transformuje binárny vektor dĺžky  $m$  na binárny vektor dĺžky  $n$ . Môžeme si položiť otázku, ako realizovať túto Boolovu funkciu, aby mala vopred špecifikované vlastnosti? Tento problém je riešený pomocou Boolovej algebry, ktorá pomocou premenných s 0-1 ohodnotením (t. j. binárnych) premenných a pomocou niekoľkých binárnych algebraických operácií a jednej unárnej algebraickej operácie je schopná dostatočne všeobecne modelovať Boolove funkcie s vopred špecifikovanými vlastnosťami.

Poznamenajme, že Boolova algebra má dva známe modely, prvým je výroková logika a druhým algebra teórie množín. Pretože obe tieto zdanlivo odťažité disciplíny majú rovnakú „metateóriu“, existuje medzi zákonmi výrokovej logiky a formulami teórie množín „dualizmus“, pomocou ktorého ku každému zákonu výrokovej logiky priradíme 1-1-značne formulu teórie množín a naopak. Dobrým, ilustratívnym príkladom tohto dualizmu sú De Morganove formuly, ktoré vo výrokovej logike a v teórii množín majú tvary

$$\neg(p \wedge q) \equiv (\neg p \vee \neg q) \Leftrightarrow \overline{A \cap B} = \bar{A} \cup \bar{B}$$

$$\neg(p \vee q) \equiv (\neg p \wedge \neg q) \Leftrightarrow \overline{A \cup B} = \bar{A} \cap \bar{B}$$

V týchto formulách, vo výrokovej logike unárna logická spojka negácie má ekvivalent v teórii množín v unárnej algebraickej operácii doplnku, a podobne, vo výrokovej logike binárne spojky konjunkcie a disjunkcie majú ekvivalenty v teórii množín v binárnych algebraických operáciách prieniku resp. zjednotenia. Na záver je potrebné poznamenať, že výroková spojka ekvivalentnosti má v teórii množín ekvivalent v relácii rovnosti. Tieto priradenia vo výrokovej logike a v teórii množín môžeme zosumarizovať takto

$$\text{výrokové premenné } p, q, r, \dots \Leftrightarrow \text{množiny } A, B, C, \dots$$

spojka negácie  $\neg \Leftrightarrow$  operácia doplnku  $\bar{\quad}$   
 spojka konjunkcie  $\wedge \Leftrightarrow$  operácia prieniku  $\cap$   
 spojka disjunkcie  $\vee \Leftrightarrow$  operácia zjednotenia  $\cup$   
 spojka ekvivalentnosti  $\equiv \Leftrightarrow$  relácia rovnosti  $=$

**Definícia 10..1.** *Boolova algebra* je algebraická štruktúra špecifikovaná usporiadanou 6-ticou  $(B, +, \cdot, \bar{\quad}, \mathbf{0}, \mathbf{1})$ , kde  $B = \{a, b, \dots, x, y, \dots\}$  je neprázdna množina elementov (premenných Boolovej algebry), ktorá obsahuje dva špeciálne odlišené elementy - konštanty  $\mathbf{0}, \mathbf{1} \in B$  a nad ktorou sú definované binárne operácie súčinu a súčtu

$$\cdot : B \times B \rightarrow B \quad (10..1a)$$

$$+ : B \times B \rightarrow B \quad (10..1b)$$

a unárna operácia komplementu

$$\bar{\quad} : B \rightarrow B \quad (10..1c)$$

ktoré vyhovujú týmto podmienkam

(1) komutatívnosť:

$$x \cdot y = y \cdot x, \quad x + y = y + x \quad (10..1d)$$

(2) asociatívnosť:

$$(x \cdot y) \cdot z = x \cdot (y \cdot z), \quad (x + y) + z = x + (y + z) \quad (10..1e)$$

(3) distributívnosť:

$$x \cdot (y + z) = (x \cdot y) + (x \cdot z), \quad x + (y \cdot z) = (x + y) \cdot (x + z) \quad (10..1f)$$

(4) vlastnosť konštanty  $\mathbf{0}$ :

$$x = x + \mathbf{0}, \quad x \cdot \bar{x} = \mathbf{0} \quad (10..1g)$$

(5) vlastnosť konštanty  $\mathbf{1}$ :

$$x = x \cdot \mathbf{1}, \quad x + \bar{x} = \mathbf{1} \quad (10..1h)$$

V literatúre existuje mnoho alternatívnych notácií Boolovej algebry. Napríklad operácia súčinu sa alternatívne vyjadruje symbolmi  $\wedge$  alebo  $*$ , podobne, operácia súčtu symbolmi  $\vee$  a  $\oplus$ . Pre zjednodušenie notácie budeme vynechávať symbol súčinu, formulu  $x \cdot y$  budeme zjednodušene písať ako  $xy$ . Z formúl (10..1g-h) vyplýva, že konštanta  $\mathbf{1}$  ( $\mathbf{0}$ ) má úlohu neutrálneho (jednotkového) prvku pre súčin (súčet). Z bežného pohľadu na formuly (10..1g-h) by niekto mohol odvodiť záver, že výraz  $\bar{x}$  je inverzná formula pre  $x$ . Pripomeňme si, že v kapitole 6 bol inverzný element definovaný pomocou vlastnosti  $x \cdot \bar{x} = \mathbf{1}$ , avšak podľa pravej formuly (A.1g) platí  $x \cdot \bar{x} = \mathbf{0}$ , z čoho vyplýva, že výraz  $\bar{x}$  nemá vlastnosti inverzného prvku (tak vzhľadom k operácii súčtu, ako aj súčinu).

**Príklad 10..1.** Najjednoduchšia Boolova algebra (s veľkým významom v informatike a v logike) je založená na dvojprvkovej množine  $B = \{0, 1\}$ . Binárne operácie súčinu, súčtu a unárna operácia komplementu sú pomocou multiplikačných tabuliek definované takto

|   |   |   |
|---|---|---|
| + | 0 | 1 |
| 0 | 0 | 1 |
| 1 | 1 | 1 |

|   |   |   |
|---|---|---|
| · | 0 | 1 |
| 0 | 0 | 0 |
| 1 | 0 | 1 |

|               |     |           |
|---------------|-----|-----------|
| $\bar{\quad}$ | $b$ | $\bar{b}$ |
|               | 0   | 1         |
|               | 1   | 0         |

Jednoducho sa môžeme presvedčiť, že pre takto špecifikované operácie sú splnené podmienky (10..1a-h), t. j. algebraická štruktúra  $(B, +, \cdot, \bar{\quad}, \mathbf{0}, \mathbf{1})$  je Boolova algebra.

**Príklad 10..2.** Nech  $A = \{a, b, c, \dots\}$  je neprázdna množina, položme  $B = \mathcal{P}(A)$ . Operácie  $\cdot$  a  $+$  sú realizované pomocou množinových operácií  $\cap$  resp.  $\cup$ , operácia komplementu je realizovaná ako množinový komplement vzhľadom k množine  $A$ ,  $\bar{x} = A - x$ . Potom platí:

- (a) binárne operácie sú asociatívne, komutatívne,
- (b) medzi binárnymi operáciami platia distributívne zákony,
- (c) prázdna množina  $\emptyset$  má vlastnosti jednotkového elementu pre operáciu  $\cup$

$$(\forall X \in B)(X \cup \emptyset = \emptyset \cup X = X)$$

- (d) množina  $A$  má vlastnosti jednotkového elementu pre operáciu  $\cap$

$$(\forall X \in B)(X \cap A = A \cap X = X)$$

- (e) pre každé  $X \in B$  existuje komplement  $\bar{X} \in B$  taký, že

$$(\forall X \in B)(X \cap \bar{X} = \emptyset)$$

$$(\forall X \in B)(X \cup \bar{X} = A)$$

To znamená, že podmienky (10..1a-h) sú splnené, t. j. algebraická štruktúra  $(\mathcal{P}(A), \cup, \cap, \bar{\phantom{x}}, \emptyset, A)$  je Boolova algebra.

**Príklad 10..3.** Nech  $B = \{p, q, r, \dots\}$  je množina výrokových formúl, ktorá je uzavretá vzhľadom k binárnym operáciám konjunkcie ( $\wedge$ ), disjunkcie ( $\vee$ ) a k unárnej operácii negácie ( $\neg$ ). Pre túto množinu je definovaná aj relácia ekvivalentnosti  $\equiv$ , dve formuly sú ekvivalentné vtedy a len vtedy, ak majú rovnakú pravdivostnú interpretáciu (logicky ekvivalentné). Z množiny  $B$  vyberieme formulu kontradikciu (napr.  $p \wedge \neg p$ ) a označíme ju symbolom  $0$ ; podobne formula tautológia (napr.  $p \vee \neg p$ ) je označená symbolom  $1$ . To znamená, že symboly  $0$  a  $1$  patria do množiny  $B$ . Pre každú formulu  $p$  platia tieto vzťahy

$$p \vee \mathbf{0} = \mathbf{0} \vee p = p$$

$$p \wedge \mathbf{1} = \mathbf{1} \wedge p = p$$

Pretože logické spojky konjunkcie a disjunkcie sú komutatívne a asociatívne, pre tieto operácie platia taktiež distributívne zákony, podmienky z definície A.1 sú splnené, t. j. algebraická štruktúra  $(B, \vee, \wedge, \neg, 0, 1)$  tvorí Boolovu algebru.

## 10.2 Vlastnosti Boolovej algebry

V úvodnej časti kapitoly 10..1 bol zmienený princíp duality medzi algebrou teórie množín a výrokovou logikou. Ukážeme, že tento princíp je aplikovateľný aj pre rôzne Boolove algebry.

Jednoduchými prostriedkami je možné dokázať jednoznačnosť jednotkového (neutrálneho) elementu a taktiež aj jednoznačnosť komplementu..

**Veta 10.1.** Jednotkové elementy  $\mathbf{1}$  a  $\mathbf{0}$  existujú jednoznačne.

Podobne sa dá dokázať aj jednoznačnosť existencie inverzných elementov v Boolovej algebry.

**Veta 10.2.** Pre každý element  $x \in B$  existuje jednoznačne element  $\bar{x} \in B$  taký, že  $x \cdot \bar{x} = \mathbf{0}$  a  $x + \bar{x} = \mathbf{1}$  (t. j. sú splnené podmienky 10.1g-h).

**Veta 10.3.** Nech  $(B, +, \cdot, \bar{\phantom{x}}, \mathbf{0}, \mathbf{1})$  je Boolova algebra, potom platia tieto formuly

(1) Involutivnosť komplementu

$$(\forall x \in B)(\overline{\overline{x}} = x) \quad (10.2a)$$

(2) Idempotentnosť

$$(\forall x \in B)(x \cdot x = x) \quad (10.2b)$$

$$(\forall x \in B)(x + x = x) \quad (10.2c)$$

(3) De Morganove zákony

$$(\forall x, y \in B)(\overline{x + y} = \overline{x} \cdot \overline{y}) \quad (10.2d)$$

$$(\forall x, y \in B)(\overline{x \cdot y} = \overline{x} + \overline{y}) \quad (10.2e)$$

(4) Nulitnosť

$$(\forall x \in B)(x + \mathbf{1} = \mathbf{1}) \quad (10.2f)$$

$$(\forall x \in B)(x \cdot \mathbf{0} = \mathbf{0}) \quad (10.2g)$$

(5) Absorpcia

$$(\forall x, y \in B)(x + (x \cdot y) = x) \quad (10.2h)$$

$$(\forall x \in B)(x \cdot (x + y) = x) \quad (10.2i)$$

(6) Komplementy konštant

$$\overline{\mathbf{0}} = \mathbf{1} \quad (10.2j)$$

$$\overline{\mathbf{1}} = \mathbf{0} \quad (10.2k)$$

(7) Vlastnosti konštant vzhľadom k binárnym operáciám

$$\mathbf{0} + \mathbf{0} = \mathbf{0}, \mathbf{0} + \mathbf{1} = \mathbf{1}, \mathbf{1} + \mathbf{0} = \mathbf{1}, \mathbf{1} + \mathbf{1} = \mathbf{1} \quad (10.2l)$$

$$\mathbf{0} \cdot \mathbf{0} = \mathbf{0}, \mathbf{0} \cdot \mathbf{1} = \mathbf{0}, \mathbf{1} \cdot \mathbf{0} = \mathbf{0}, \mathbf{1} \cdot \mathbf{1} = \mathbf{1} \quad (10.2m)$$

Dôkaz týchto vlastností prenecháme na cvičenie.

### 10.3 Boolove funkcie

V úvodnej časti tejto kapitole bola Boolova funkcia definovaná ako funkcia nad binárnymi premennými  $\{0,1\}$ . Tento pomerne zjednodušený pohľad na Boolovu funkciu bude teraz rozšírený tak, aby koncepcia Boolovej funkcie bola časťou Boolovej algebry. Základný pojem pre definíciu Boolovej funkcie je pojem Boolovej premennej. Použijeme analogický prístup, aký sa používa pre definíciu reálnej premennej, je to veličina, ktorá môže nadobúdať hodnoty z množiny reálnych čísel.

**Definícia 10.2.** Nech  $(B, +, \cdot, \bar{\phantom{x}}, \mathbf{0}, \mathbf{1})$  je Boolova algebra. Potom,

(1) **Boolova premenná** je taká premenná, ktorá nadobúda hodnoty z množiny  $B$ ,

(2) **komplement premennej**  $x$ , označený  $\overline{x}$ , je taká premenná, ktorej hodnota sa rovná komplementu hodnoty premennej  $x$  (t. j. ak  $x = b \in B$ , potom  $\overline{x} = \overline{b} \in B$ ,

(3) **literál** je Boolova premenná  $x$  alebo jej komplement  $\overline{x}$ .

V ďalšom texte budeme používať notáciu, ktorá umožní rozlíšiť literál

$$x^e = \begin{cases} x & (\text{pre } e = 1) \\ \bar{x} & (\text{pre } e = 0) \end{cases} \quad (10.3)$$

Podobne ako pre reálnu premennú, aj Boolova premenná môže byť kombinovaná do tvaru Boolových formúl použitím binárnych operácií súčinu, súčtu a komplementu.

**Definícia 10.3.** Nech  $(B, +, \cdot, \bar{\phantom{x}}, \mathbf{0}, \mathbf{1})$  je Boolova algebra. Potom **Boolova formula**, obsahujúca Boolove premenné  $x_1, x_2, \dots, x_n$ , je definovaná takto:

- (1) konštanty  $\mathbf{0}$  a  $\mathbf{1}$  sú Boolove formuly,
- (2) Boolove premenné  $x_1, x_2, \dots, x_n$  sú Boolove formuly,
- (3) ak  $X$  a  $Y$  sú Boolove formuly, potom aj výrazy  $(X \cdot Y)$ ,  $(X + Y)$ ,  $\bar{X}$  a  $\bar{Y}$  sú Boolove formuly.

V tejto definícii bod (3) môže byť rozšírený ešte o ďalšie Boolove spojky, napr. spojku  $\oplus$ , ktorá sa nazýva exkluzívna disjunkcia (XOR) a je definovaná takto:  $x \oplus y = 1$  vtedy a len vtedy, ak buď  $x = 1$  alebo  $y = 1$ , ale nie súčasne ( $x \neq y$ );  $x \oplus y = 0$  vtedy a len vtedy, ak premenné  $x$  a  $y$  majú rovnakú hodnotu ( $x = y$ ).

V ďalšom texte budeme používať konvenciu, že ak bude jasné o akú formulu sa jedná, tak termín '*Boolova formula*' budeme skracovať na '*formula*'. Podobne, ako vo výrokovej logike môžeme si definovať rastúcu prioritu operácií takto: (1) súčet, (2) súčin a (3) komplement. Napríklad, formulu  $((x \cdot y) + z)$  môžeme pomocou tejto konvencie vyjadriť v zjednodušenom tvare bez zátvoriek  $x \cdot y + z$ . Konečne, podobne ako v štandardnej algebre, budeme vynechávať znak súčinu, napríklad predchádzajúci ilustračný príklad má tvar  $xy + z$ .

**Príklad 10.4.** Zjednodušte formulu  $((x + y) \cdot (\bar{x} + \bar{y}))$ .

Použitím distributívneho zákona a (A.1g-h)

$$((x + y) \cdot (\bar{x} + \bar{y})) = (x \cdot \bar{x}) + (x \cdot \bar{y}) + (y \cdot \bar{x}) + (y \cdot \bar{y}) = \underset{0}{x\bar{x}} + x\bar{y} + y\bar{x} + \underset{0}{y\bar{y}} = x\bar{y} + \bar{x}y$$

**Definícia 10.4.** Dve Boolove formule sú **ekvivalentné** (alebo **rovné**) vtedy a len vtedy, ak jedna pomocou konečného počtu aplikácií axióm Boolovej algebry je pretransformovaná na druhú formulu.

Podľa príkladu A.4 formule  $\varphi_1 = (x + y) \cdot (\bar{x} + \bar{y})$  a  $\varphi_2 = x\bar{y} + \bar{x}y$  sú ekvivalentné, pretože druhú formulu získame z prvej použitím konečného počtu aplikácií axióm Boolovej algebry, potom  $\varphi_1 = \varphi_2$ .

Konečne sa dostávame k definícii Boolovej funkcie  $f(x_1, x_2, \dots, x_n)$  ako Boolovej formuly, ktorá obsahuje premenné  $x_1, x_2, \dots, x_n$ . Napríklad

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

**Definícia 10.5.** Nech  $(B, +, \cdot, \bar{\phantom{x}}, \mathbf{0}, \mathbf{1})$  je Boolova algebra.

- (1) **Boolova funkcia**  $f(x_1, x_2, \dots, x_n)$  premenných  $x_1, x_2, \dots, x_n$ , je zobrazenie  $f: B^n \rightarrow B$ ,

pričom  $f(x_1, x_2, \dots, x_n)$  je špecifikovaná ako Boolova formula.

(2) Všetky Boolove formuly, ktoré sú navzájom ekvivalentné, definujú rovnakú funkciu.

Z tejto definície vyplýva, že ekvivalentné Boolove formuly špecifikujú rovnakú Boolovu formulu. Napríklad, máme dve funkcie

$$f : B^2 \rightarrow B \quad f(x_1, x_2) = x_1(\bar{x}_1 + x_2)$$

$$g : B^2 \rightarrow B \quad g(x_1, x_2) = x_1x_2$$

Použitím distribučného zákona ľahko dokážeme, že formuly sú ekvivalentné,  $x_1(\bar{x}_1 + x_2) = x_1x_2$ , potom funkcie  $f$  a  $g$  sú rovnaké.

Pretože Boolova funkcia môže byť vyjadrená mnohými rôznymi formulami, ktoré sú navzájom ekvivalentné, vzniká otázka, ako efektívne rozhodnúť, či dve Boolove formuly sú ekvivalentné, alebo či dve Boolové funkcie sú rovnaké. Ukážeme postup, ktorý nie je založený na transformácii jednej formuly na druhú, aby sme rozhodli, či funkcie sú rovnaké, ale navrhne sa „kanonická“ reprezentácia Boolovej funkcie, podľa ktorej môžeme jednoducho rozhodnúť, či dve Boolove funkcie sú rovnaké alebo nie.

**Definícia 10.6. Konjunktívna klauzula** premenných  $x_1, x_2, \dots, x_n$  je Boolova formula, ktorá obsahuje súčin  $n$  literálov (t. j. premennú alebo jej komplement) pre každú premennú.

Ako príklad pre konjunktívnu premenných  $x_1, x_2, x_3$  sú tieto formuly:  $x_1x_2x_3$ ,  $x_1x_2\bar{x}_3$ ,  $x_1\bar{x}_2x_3$ ,  $\bar{x}_1x_2x_3$ , ...,  $\bar{x}_1\bar{x}_2\bar{x}_3$ . Ak použijeme formalizmus  $x^e$  (pozri (10.3)), potom konjunktívnu klauzulu premenných  $x_1, x_2, \dots, x_n$ , ktorá je špecifikovaná binárnym vektorom  $e = (e_1, e_2, \dots, e_n)$ , má tvar

$$l_e = x_1^{e_1}x_2^{e_2} \dots x_n^{e_n} \quad (10.4)$$

Napríklad, pre  $e = (11011)$  konjunktívna klauzula má tvar

$$l_{(11011)} = x_1^1x_2^1x_3^0x_4^1x_5^1 = x_1x_2\bar{x}_3x_4x_5$$

Pretože binárnym vektorom  $e = (e_1, e_2, \dots, e_n)$  je  $2^n$ , potom aj **rôznych** konjunktívnych klauzúl je  $2^n$ .

**Definícia 10.7. Disjunktívna klauzula** premenných  $x_1, x_2, \dots, x_n$  je Boolova formula, ktorá obsahuje disjunkciu  $n$  literálov (t. j. premennú alebo jej komplement) pre každú premennú.

Podobne ako pre konjunktívnu klauzulu, môžeme aj súčtovú klauzulu pre premenné  $x_1, x_2, \dots, x_n$  špecifikovať binárnym vektorom  $e = (e_1, e_2, \dots, e_n)$

$$L_e = x_1^{e_1} + x_2^{e_2} + \dots + x_n^{e_n} \quad (10.5)$$

Pre  $e = (10100)$  disjunktívna klauzula má tvar

$$L_e = x_1^1 + x_2^0 + x_3^1 + x_4^0 + x_5^0 = x_1 + \bar{x}_2 + x_3 + \bar{x}_4 + \bar{x}_5$$

Pretože každá disjunktívna klauzula premenných  $x_1, x_2, \dots, x_n$  je špecifikovaná binárnym vektorom  $e = (e_1, e_2, \dots, e_n)$ , potom počet súčtových klauzúl je taktiež  $2^n$ .

Týmto sa dostávame k formulácii hlavného výsledku tejto kapitoly, že každá Boolova funkcia môže byť jednoznačne vyjadrená ako sumácia súčinových klauzúl (tento tvar sa nazýva vo výrokovej logike **disjunktívna normálna forma**, skratka DNF).

**Príklad 10.5.** Vyjadrite Boolovu funkciu  $x_1x_2(x_1+x_3)$  pomocou súčtu konjunktívnych klauzúl (DNF)

$$\begin{aligned}
 x_1x_2(x_1+x_3) &= x_1x_2x_1 + x_1x_2x_3 \\
 &= x_1x_1x_2 + x_1x_2x_3 = x_1x_2 + x_1x_2x_3 \\
 &= x_1x_2\mathbf{1} + x_1x_2x_3 = x_1x_2(x_3 + \bar{x}_3) + x_1x_2x_3 \\
 &= \underbrace{x_1x_2x_3 + x_1x_2x_3}_{x_1x_2x_3} + x_1x_2\bar{x}_3 \\
 &= x_1x_2x_3 + x_1x_2\bar{x}_3
 \end{aligned}$$

**Veta 10.4.** Každá Boolova funkcia  $f(x_1, x_2, \dots, x_n)$ , ktorá sa identicky nerovná nule, môže byť špecifikovaná ako disjunkcia konjunktívnych klauzúl

$$\begin{aligned}
 f(x_1, x_2, \dots, x_n) &= \sum_e f(e_1, e_2, \dots, e_n) x_1^{e_1} x_2^{e_2} \dots x_n^{e_n} \\
 &= \sum_e f(e_1, e_2, \dots, e_n) l_{(e_1, e_2, \dots, e_n)} \\
 &= \sum_{(f(e)=1)} f(e) l_e
 \end{aligned} \tag{10.6}$$

V poslednej formule berieme v úvahu len také binárne hodnoty argumentov  $e$  pre ktoré sú funkčné hodnoty jednotkové, nulové funkčné hodnoty,  $f(e) = 0$ , môžu byť ignorované.

Naznačíme jednoduchý konštruktívny dôkaz. Boolova funkcia  $f(x_1, x_2, \dots, x_n)$  je alternatívne špecifikovaná pomocou jej funkčných hodnôt  $f(e_1, e_2, \dots, e_n)$ , pre všetky hodnoty binárneho vektora  $e = (e_1, e_2, \dots, e_n) \in \{0, 1\}^n$ .

| #     | $e = (e_1, e_2, \dots, e_n)$               | $f(e_1, e_2, \dots, e_n)$ |
|-------|--|---------------------------|
| 1     | (00.....00)                                | 0                         |
| 2     | (00.....01)                                | 1                         |
| ..... |  |                           |
| $i$   | $(e_1^{(i)}, e_2^{(i)}, \dots, e_n^{(i)})$ | 1/0                       |
| ..... |  |                           |
| $2^n$ | (11.....11)                                | 0                         |

Konjunktívna klauzula  $l_{(e_1, e_2, \dots, e_n)}(x_1, x_2, \dots, x_n) = x_1^{e_1} x_2^{e_2} \dots x_n^{e_n}$ , ktorá je priradená binárnemu vektoru  $e = (e_1, e_2, \dots, e_n) \in \{0, 1\}^n$  má zaujímavú vlastnosť, jej funkčná hodnota sa rovná 1 len pre  $(x_1, x_2, \dots, x_n) = (e_1, e_2, \dots, e_n)$ , kde  $e_i \in \{0, 1\}$ , pre všetky iné prípady funkčná hodnota je 0

$$l_{(e_1, e_2, \dots, e_n)}(x_1, x_2, \dots, x_n) = \begin{cases} 1 & (\text{pre } (x_1, x_2, \dots, x_n) = (e_1, e_2, \dots, e_n)) \\ 0 & (\text{pre } (x_1, x_2, \dots, x_n) \neq (e_1, e_2, \dots, e_n)) \end{cases} \tag{10.7}$$

To znamená, že pre konštrukciu (10.6) sú pre nás dôležité len funkčné hodnoty 1, funkčné hodnoty 0 nie sú podstatné pre náš konštruktívny dôkaz. Zostrojíme Boolovu formulu ako sumáciu týchto klauzúl (t. j. v DNF tvare)

$$F(x_1, x_2, \dots, x_n) = \sum_e f(e_1, e_2, \dots, e_n) l_{(e_1, e_2, \dots, e_n)} \quad (10.8)$$

Z konštrukcie tejto Boolovej funkcie, vyplýva, že jej funkčné hodnoty sú špecifikované tabuľkou funkčných hodnôt Boolovej funkcie  $f(x_1, x_2, \dots, x_n)$ . To znamená, že Boolove funkcie  $f(x_1, x_2, \dots, x_n)$  a  $F(x_1, x_2, \dots, x_n)$  sú ekvivalentné, t. j. majú rovnaké funkčné hodnoty pre rôzne hodnoty argumentov. Týmto sme zavřšili jednoduchý intuitívny konštruktívny dôkaz vety 10.4.

Poznamenajme, že DNF tvar Boolovej funkcie je určený jednoznačne až na permutácie argumentov v súčtových klauzulách, alebo až na permutácie súčtových klauzúl. Táto nejednoznačnosť DNF tvaru vyplýva zo skutočnosti, že binárne operácie súčtu a súčinu sú komutatívne. Môžeme teda konštatovať, že DNF sú základné charakteristiky (niečo ako odtlačky prstov alebo zloženie DNA) Boolových funkcií. Aby sme odstránili prípadné nejednoznačnosti zapisujeme DNF v tzv. kanonickom tvare, t. j. jednotlivé argumenty sa zapisujú postupne podľa rastúceho indexu (tým sme odstránili nejednoznačnosti v dôsledku komutatívnosti súčinu) a potom jednotlivé súčtové klauzule sú písané v poradí rastúcej číselnej hodnoty „indexu“  $e = (e_1, e_2, \dots, e_n)$ .

**Príklad 10.6.** Jednoduchý dôkaz formuly (10.6) môžeme zostrojiť pomocou Shannonovej formuly [xx] Boolovej funkcie  $f(x_1)$

$$f(x_1) = \bar{x}_1 f(0) + x_1 f(1)$$

Ktorá je jednoduchým dôsledkom skutočnosti, že v Boolovej algebre funkcia  $f(x_1)$  má len dve možné hodnoty **0** alebo **1**. Túto formulu môžeme aplikovať k funkcii  $f(x_1, x_2)$  k premennej  $x_1$

$$f(x_1, x_2) = \bar{x}_1 f(0, x_2) + x_1 f(1, x_2)$$

Na komponenty  $f(0, x_2)$  a  $f(1, x_2)$  Shanonnovu formulu

$$f(0, x_2) = \bar{x}_2 f(0, 0) + x_2 f(0, 1)$$

$$f(1, x_2) = \bar{x}_2 f(1, 0) + x_2 f(1, 1)$$

Potom pre  $f(x_1, x_2)$  dostaneme

$$f(x_1, x_2) = \bar{x}_1 \bar{x}_2 f(0, 0) + \bar{x}_1 x_2 f(0, 1) + x_1 \bar{x}_2 f(1, 0) + x_1 x_2 f(1, 1)$$

Týmto sme dokázali, že pomocou Shanonnovej formuly pre  $n=1$  dokážeme aj formulu pre  $n=2$ . Týmto postupom (matematickou indukciou) dokážeme, že formula platí (10.6) aj pre ľubovoľné celočíselné  $n$ .

**Príklad 10.7.** Zostrojte Boolovu funkciu  $f(x_1, x_2) = x_1 + x_2$  v tvare DNF. Podľa vety 10.5 DNF tvar tejto funkcie je

$$f(x_1, x_2) = f(0, 0) \bar{x}_1 \bar{x}_2 + f(0, 1) \bar{x}_1 x_2 + f(1, 0) x_1 \bar{x}_2 + f(1, 1) x_1 x_2$$

kde jednotlivé funkčné hodnoty sú uvedené v tabuľke

| # | $e_1$ | $e_2$ | $f(e_1, e_2)$ |
|---|-------|-------|---------------|
| 1 | 0     | 0     | 0             |
| 2 | 0     | 1     | 1             |
| 3 | 1     | 0     | 1             |
| 4 | 1     | 1     | 1             |



Potom funkcia  $f$  má tvar

$$\begin{aligned} f(x_1, x_2) &= 0\bar{x}_1\bar{x}_2 + 1\bar{x}_1x_2 + 1x_1\bar{x}_2 + 1x_1x_2 \\ &= \bar{x}_1x_2 + x_1\bar{x}_2 + x_1x_2 \end{aligned}$$

Lahko dokážeme, že takto definovaná funkcia  $f(x_1, x_2) = \bar{x}_1x_2 + x_1\bar{x}_2 + x_1x_2$  je ekvivalentná s pôvodnou Boolovou funkciou  $f(x_1, x_2) = x_1 + x_2$

$$f(x_1, x_2) = \bar{x}_1x_2 + x_1\bar{x}_2 + x_1x_2 = \bar{x}_1x_2 + x_1\bar{x}_2 + x_1x_2 + x_1x_2 = \left( \begin{matrix} \bar{x}_1 + x_1 \\ 1 \end{matrix} \right) x_2 + x_1 \left( \underbrace{\bar{x}_2 + x_2}_1 \right) = x_1 + x_2$$

**Príklad 10.8.** Zostrojte Boolovu funkciu  $f(x_1, x_2, x_3) = x_2x_3 + x_1x_3$  v tvare DNF. Táto Boolova funkcia je určená tabuľkou funkčných hodnôt

| # | $e_1$ | $e_2$ | $e_3$ | $e_2e_3$ | $e_1e_3$ | $e_2e_3 + e_1e_3$ |
|---|-------|-------|-------|----------|----------|-------------------|
| 1 | 0     | 0     | 0     | 0        | 0        | 0                 |
| 2 | 0     | 0     | 1     | 0        | 0        | 0                 |
| 3 | 0     | 1     | 0     | 0        | 0        | 0                 |
| 4 | 0     | 1     | 1     | 1        | 0        | 1                 |
| 5 | 1     | 0     | 0     | 0        | 0        | 0                 |
| 6 | 1     | 0     | 1     | 0        | 1        | 1                 |
| 7 | 1     | 1     | 0     | 0        | 0        | 0                 |
| 8 | 1     | 1     | 1     | 1        | 1        | 1                 |

Potom funkcia  $f(x_1, x_2, x_3)$  (uvažujeme len jednotkové funkčné hodnoty) má DNF tvar

$$f(x_1, x_2, x_3) = \bar{x}_1x_2x_3 + x_1\bar{x}_2x_3 + x_1x_2x_3$$

Použijeme duálny princíp z vety 10.1, veta 10.5 má potom tento duálny tvar

**Veta 10.5.** Každá Boolova funkcia  $f(x_1, x_2, \dots, x_n)$ , ktorá sa identicky nerovná nule, môže byť špecifikovaná ako konjunkcia disjunktívnych klauzúl

$$\begin{aligned} f(x_1, x_2, \dots, x_n) &= \prod_e \left( f(e_1, e_2, \dots, e_n) + x_1^{1-e_1} + x_2^{1-e_2} + \dots + x_n^{1-e_n} \right) \\ &= \prod_e \left( f(e_1, e_2, \dots, e_n) + L_{(1-e_1, 1-e_2, \dots, 1-e_n)} \right) \\ &= \prod_{\substack{e \\ (f(e)=0)}} \left( f(e) + L_{\bar{e}} \right) \end{aligned} \tag{10.9}$$

kde  $\bar{e} = (\bar{e}_1, \bar{e}_2, \dots, \bar{e}_n) = (1 - e_1, 1 - e_2, \dots, 1 - e_n)$ .

Táto veta reprezentuje hlavný duálny výsledok tejto kapitoly, že každá Boolova funkcia môže byť jednoznačne vyjadrená ako súčin súčtových klauzúl (tento tvar sa nazýva vo výrokovej logike **konjunktívna normálna forma**, skratka KNF).

**Príklad 10.9 .** Vyjadrite  $f(x_1, x_2) = x_1(x_1 + x_2)$  v KNF tvare. V prvom kroku zostrojíme tabuľku funkčných hodnôt tejto Boolovej funkcie

| # | $e_1$ | $e_2$ | $e_1+e_2$ | $e_1(e_1+e_2)$ |
|---|-------|-------|-----------|----------------|
| 1 | 0     | 0     | 0         | 0              |
| 2 | 0     | 1     | 1         | 0              |
| 3 | 1     | 0     | 1         | 1              |
| 4 | 1     | 1     | 1         | 1              |

Použitím (10.9) dostaneme vyjadrenú Boolovu funkciu  $f(x_1, x_2) = x_1(x_1 + x_2)$  v KNF

$$f(x_1, x_2) = \underbrace{\left( f(0,0) + x_1 + x_2 \right)}_0 \cdot \underbrace{\left( f(0,1) + x_1 + \bar{x}_2 \right)}_0 \cdot \underbrace{\left( f(1,0) + \bar{x}_1 + x_2 \right)}_1 \cdot \underbrace{\left( f(1,1) + \bar{x}_1 + \bar{x}_2 \right)}_1$$

$$= (x_1 + x_2) \cdot (x_1 + \bar{x}_2)$$

Z tohto príkladu vyplýva, že pre konštrukciu KNF sú dôležité nulové funkčné hodnoty danej Boolovej funkcie. Táto vlastnosť je duálna k vlastnosti DNF, kde sú relevantné jednotkové funkčné hodnoty Boolovej funkcie. Z tohto faktu vyplýva skutočnosť, že si zvolíme DNF tvar Boolovej funkcie vtedy, keď tabuľka obsahuje v prevažnej miere nulové funkčné hodnoty, KNF si zvolíme vtedy, keď tabuľka obsahuje v prevažnej miere jednotkové funkčné hodnoty. V prípade, že tabuľka obsahuje rovnaký počet nulových a jednotkových funkčných hodnôt z pohľadu „zložitosti“ konštrukcie je jedno, aký tvar Boolovej funkcie sme zvolili.

**Príklad 10.10.** Zostrojte KNF Boolovej funkcie  $f(x_1, x_2, x_3) = (\bar{x}_1 + x_2) \cdot (\bar{x}_1 + \bar{x}_3)$ . Tabuľka funkčných hodnôt má tvar

| # | $e_1$ | $e_2$ | $e_3$ | $\bar{e}_1$ | $\bar{e}_3$ | $\bar{e}_1 + e_2$ | $\bar{e}_1 + \bar{e}_3$ | $(\bar{e}_1 + e_2) \cdot (\bar{e}_1 + \bar{e}_3)$ |
|---|-------|-------|-------|-------------|-------------|-------------------|-------------------------|---|
| 1 | 0     | 0     | 0     | 1           | 1           | 1                 | 1                       | 1   |
| 2 | 0     | 0     | 1     | 1           | 0           | 1                 | 1                       | 1   |
| 3 | 0     | 1     | 0     | 1           | 1           | 1                 | 1                       | 1   |
| 4 | 0     | 1     | 1     | 1           | 0           | 1                 | 1                       | 1   |
| 5 | 1     | 0     | 0     | 0           | 1           | 0                 | 1                       | 0   |
| 6 | 1     | 0     | 1     | 0           | 0           | 0                 | 0                       | 0   |
| 7 | 1     | 1     | 0     | 0           | 1           | 1                 | 1                       | 1   |
| 8 | 1     | 1     | 1     | 0           | 0           | 1                 | 0                       | 0   |

KNF má potom tvar (využívame len tri riadky s nulovou výslednou funkčnou hodnotou)

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

## 10.4 Spínacie obvody

Mnohé elektronické zariadenia, akými sú napr. počítače, telefónne ústredne, zariadenia na riadenie dopravy, obsahujú ako časť spínacie obvody. Ich teória bola vypracovaná v r. 1938 C. Shannonom v rámci jeho MSc. dizertácie [xx]. Spínač môže byť chápaný ako taký spoj v obvode, ktorý - ak je uzavretý - potom ním prechádza elektrický prúd, v opačnom prípade - ak je otvorený - elektrický prúd ním neprechádza. Spínač môžeme znázorniť takto:



Predpokladajme, že v spínacom obvode máme spínač A. Stav tohto spínača označíme premennou  $x$ , ak  $x = 1$  ( $x = 0$ ), potom spínač A je uzavretý (otvorený).

O trochu zložitejší prípad spínacieho obvodu obsahuje dva spínače  $A_1$  a  $A_2$



Hovoríme, že v tomto prípade sú spínače zapojené *sériovo*. Nech  $x_1$  a  $x_2$  sú premenné popisujúce stavy spínačov  $A_1$  resp.  $A_2$ , tieto premenné ak sa rovnajú 1 (0), potom daný spínač je uzavretý (otvorený). Nech  $f(x_1, x_2)$  je funkcia, ktorej hodnota sa rovná 1 (0) pre tie hodnoty  $x_1$  a  $x_2$ , ktoré umožňujú (znemožňujú) tok prúdu. Táto funkcia môže byť chápaná ako binárna funkcia  $f : \{0,1\}^2 \rightarrow \{0,1\}$ , ktorej funkčné hodnoty sú určené tabuľkou

| # | $x_1$ | $x_2$ | $f(x_1, x_2)$ |
|---|-------|-------|---------------|
| 1 | 0     | 0     | 0             |
| 2 | 0     | 1     | 0             |
| 3 | 1     | 0     | 0             |
| 4 | 1     | 1     | 1             |

Z tejto tabuľky vyplýva, že funkcia  $f$  je vyjadrená ako súčin premenných  $x_1$  a  $x_2$

$$f(x_1, x_2) = x_1 \cdot x_2 \quad (10.12)$$

Táto funkcia je definovaná nad Boolovou algebrou  $(\{0,1\}, +, \cdot, \mathbf{0}, 1)$ , t. j. premenné patria do množiny  $\{0,1\}$ .

Nový druh spínacieho obvodu, ktorý je podobný sériovému obvodu (10.11), ale v paralelnom zapojení, má tvar



Podobne ako v predchádzajúcom príklade (10.11), nech spínače  $A_1$  a  $A_2$  sú popísané premennými  $x_1$  a  $x_2$ , tento obvod je popísaný funkciou  $g(x_1, x_2)$  nad Boolovou algebrou  $(\{0,1\}, +, \cdot, \mathbf{0}, 1)$ , ktorá je špecifikovaná tabuľkou

| # | $x_1$ | $x_2$ | $g(x_1, x_2)$ |
|---|-------|-------|---------------|
| 1 | 0     | 0     | 0             |
| 2 | 0     | 1     | 1             |
| 3 | 1     | 0     | 1             |
| 4 | 1     | 1     | 1             |

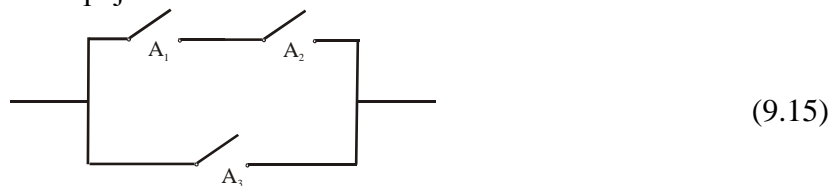
Potom funkcia  $g(x_1, x_2)$  je špecifikovaná ako súčet premenných

$$g(x_1, x_2) = x_1 + x_2 \quad (10.14)$$

ktorá je definovaná nad Boolovou algebrou  $(\{0,1\}, +, \cdot, \mathbf{0}, 1)$

Funkcie, ktoré sú podobné (11.12) a (10.14) popisujú vlastnosti spínacieho obvodu pomocou stavov spínačov, ktoré sú súčasťou daného obvodu. Takéto funkcie sa nazývajú **spínacie funkcie**. Majme  $n$  spínačov, ktorých stavy sú špecifikované premennými  $x_1, x_2, \dots, x_n$ . Spínacia funkcia  $f : \{0,1\}^n \rightarrow \{0,1\}$  popisuje správanie sa spínacieho obvodu pre všetky možné  $2^n$  stavy spínačov. Ako už bolo ukázané na predchádzajúcich ilustračných príkladoch, funkcia  $f$  môže byť reprezentovaná Boolovou formulou a teda aj Boolovou funkciou.

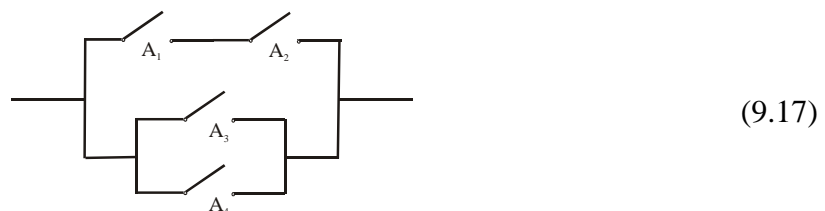
Nasledujúci príklad spínacieho obvodu bude zložitejší spínací obvod, ktorý obsahuje tri spoínače v sériovo-paralelnom zapojení



Nech jednotlivé spínače  $A_1, A_2$  a  $A_3$  sú špecifikované premennými  $x_1, x_2$  resp.  $x_3$ . Nech funkcia  $f(x_1, x_2)$  popisuje vlastnosti hornej časti obvodu, ktorý obsahuje dva sériovo zapojené spínače  $A_1$  a  $A_2$ . Pomocou predchádzajúceho príkladu (10.11) funkcia, ktorá špecifikuje takýto obvod má tvar  $f_1(x_1, x_2) = x_1 x_2$ . Celkový obvod potom môžeme zložiť z dvoch paralelných podštruktúr, horná je reprezentovaná funkciou  $f_1(x_1, x_2) = x_1 x_2$  a dolná je reprezentovaná funkciou  $f_2(x_3) = x_3$ . Spojením týchto dvoch funkcií pomocou (10.14) dostaneme Boolovu funkciu celého spínacieho obvodu (10.15)

$$f(x_1, x_2, x_3) = f_1(x_1, x_2) + f_2(x_3) = x_1 x_2 + x_3 \quad (10.16)$$

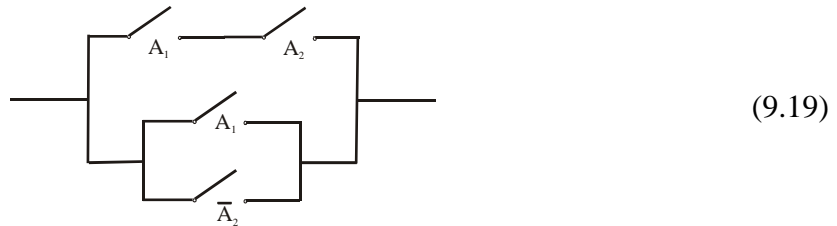
**Príklad 10.11.** Zostrojte spínaciu funkciu  $f$  spínacieho zariadenia



Nech  $x_1, x_2, x_3$  a  $x_4$  sú premenné označujúce stavy spínačov  $A_1, A_2, A_3$  resp.  $A_4$ . Potom celková spínacia funkcia zariadenia má tvar  $f(x_1, x_2, x_3, x_4)$ . V prvom kroku určíme pomocné spínacie funkcie  $f_1(x_1, x_2)$  a  $f_2(x_3, x_4)$ , ktoré sú priradené hornej časti obsahujúcej spínače  $A_1, A_2$  resp. dolnej časti obsahujúcej spínače  $A_3, A_4$ . Tieto funkcie sú určené spínacími funkciami (10.12) a (10.14),  $f_1(x_1, x_2) = x_1 x_2$  resp.  $f_2(x_3, x_4) = x_3 + x_4$ . Hľadaná spínacia funkcia má potom tvar

$$f(x_1, x_2, x_3, x_4) = f_1(x_1, x_2) + f_2(x_3, x_4) = x_1 x_2 + x_3 + x_4 \quad (10.18)$$

**Príklad 10.12.** Zostrojte spínaciu funkciu  $f$  zariadenia, ktoré vznikne malou modifikáciou spínacieho zariadenia (10.17)



kde pôvodné spínače  $A_1$  a  $A_3$  sú teraz spolu spriahnuté, t. j. obe sú súčasne zapnuté alebo vypnuté (preto sú obe označené  $A_1$ ). Ďalšie dva pôvodné spínače  $A_2$  a  $A_4$  sú teraz spolu taktiež spriahnuté, ale opačným spôsobom, t. j. ak je jeden spínač zapnutý, druhý je vypnutý a naopak (preto sú oba označené  $A_2$  a  $\bar{A}_2$ ). Spínaciu funkciu takto špecifikovaného zariadenia ľahko zostrojíme pomocou spínacej funkcie (10.18) pôvodného spínacieho zariadenia, keď položíme  $x_3 = x_1$  a  $x_4 = \bar{x}_2$

$$g(x_1, x_2) = f(x_1, x_2, x_1, \bar{x}_2) = x_1 x_2 + x_1 + \bar{x}_2 \quad (10.20)$$

Tabuľka funkčných hodnôt tejto spínacej funkcie má tvar

| $x_1$ | $x_2$ | $\bar{x}_2$ | $x_1 x_2$ | $x_1 x_2 + x_1 + \bar{x}_2$ |
|-------|-------|-------------|-----------|-----------------------------|
| 0     | 0     | 1           | 0         | 1                           |
| 0     | 1     | 0           | 0         | 0                           |
| 1     | 0     | 1           | 0         | 1                           |
| 1     | 1     | 0           | 1         | 1                           |

Ak použijeme vetu 10.6, zostrojíme Boolovu funkciu v tvare KNF, ktorá simuluje túto tabuľku, z ktorej si vyberieme riadok s nulovou funkčnou hodnotou, potom  $f(x_1, x_2) = x_1 + \bar{x}_2$ . Spínacie zariadenie (10.19) má túto zjednodušenú spínaciu funkciu

$$g(x_1, x_2) = x_1 + \bar{x}_2 \quad (10.21)$$

Ľahko dokážeme, že funkcie (10.20) a (10.21) sú ekvivalentné

$$g(x_1, x_2) = x_1 x_2 + x_1 + \bar{x}_2 = \begin{pmatrix} x_2 + 1 \\ 1 \end{pmatrix} x_1 + \bar{x}_2 = 1x_1 + \bar{x}_2 = x_1 + \bar{x}_2$$

**Príklad 10.13.** Budeme riešiť veľmi praktickú úlohu, ktorá pre mnohých z nás je záhadou ako vlastne funguje. Predstavme si schodište, na začiatku a konci ktorého sú umiestnené stenové vypínače  $S_1$  a  $S_2$ , pomocou ktorých zapneme alebo vypneme svetlo nad schodišťom. Hlavná požiadavka je taká, aby sa na jednom konci mohlo svetlo buď vypnúť, ak na druhom konci je zapnuté, alebo zapnúť, ak je na druhom konci vypnuté. Túto podmienku môžeme formulovať alternatívne tak, že ak sú oba spínače  $S_1$  a  $S_2$  vypnuté alebo zapnuté, potom zariadením nepreteká prúd, ale stačí, aby bolo zapnuté práve jedno, potom zariadením preteká prúd, čo môžeme vyjadriť touto tabuľkou

| $S_1$   | $S_2$   | prúd |
|---------|---------|------|
| zapnuté | zapnuté | nie  |
| zapnuté | vypnuté | áno  |
| vypnuté | zapnuté | áno  |
| vypnuté | vypnuté | nie  |

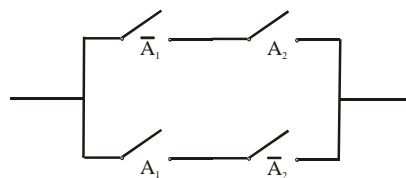
Predpokladajme, že vypínače  $S_1$  a  $S_2$  sú realizovaný pomocou dvoch spínačov  $A_1$  a  $A_2$ . Ak použijeme premenné  $x_1$  a  $x_2$ , ktoré označujú stavy spínačov  $A_1$  a  $A_2$ , potom vyššie uvedená tabuľka môže byť prepísaná do tvaru

| $x_1$ | $x_2$ | $f(x_1, x_2)$ |
|-------|-------|---------------|
| 0     | 0     | 0             |
| 0     | 1     | 1             |
| 1     | 0     | 1             |
| 1     | 1     | 0             |

Podľa vety 10.5, táto tabuľka špecifikuje Boolovu funkciu v DNF

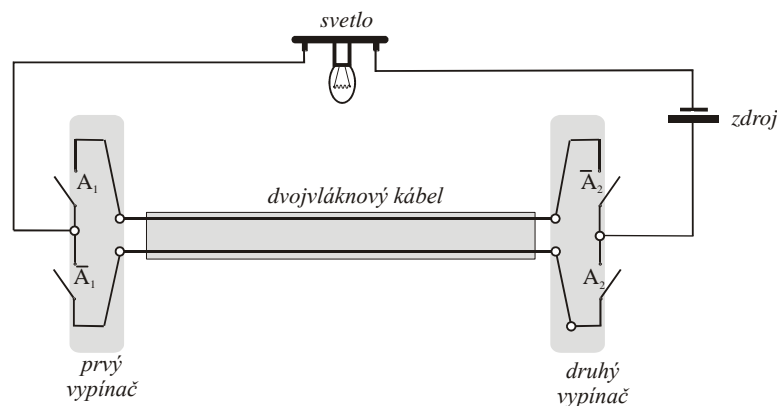
$$f(x_1, x_2) = \bar{x}_1 x_2 + x_1 \bar{x}_2 \quad (10.22)$$

Spínací obvod so spínacou funkciou takto špecifikovanou má tvar



(11.23)

Diagram (10.24) znázorňuje realizáciu tohto spínacieho obvodu, kde vytieňované oblasti tvoria stenové vypínače, ktoré sú spojené dvojvláknovým káblom.



(11.24)

Tento jednoduchý príklad demonštruje užitočnosť teórie spínacích obvodov vybudovanej pomocou Boolových funkcií. Vyriešili sme praktický príklad, ako realizovať zapínanie a vypínanie svetla na schodišti (alebo na dlhej chodbe) tak, že každým vypínačom môžeme svetlo zapnúť alebo vypnúť, nezávisle od polohy druhého vypínača.

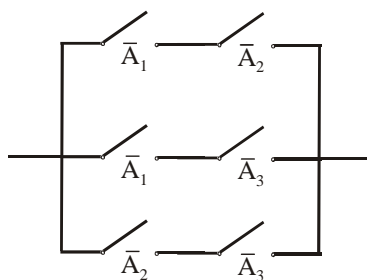
**Príklad 10.14.** Ústredné kúrenie v rodinnom dome je riadené troma termostatmi, ktoré sú umiestnené v každej izbe domu. Termostaty sú nastavené na  $18\text{ }^{\circ}\text{C}$ , pričom z dôvodu šetrenia energiou sa požaduje, aby systém ústredného kúrenia bol zapnutý len ak teplota aspoň v dvoch izbách je menšia ako  $18\text{ }^{\circ}\text{C}$ , v opačnom prípade je systém vypnutý. Navrhnete spínacový systém, ktorý prijíma signály z termostatov a ktorý riadi ústredné kúrenie. Pokúste sa minimalizovať navrhnutý systém, aby bol čo najjednoduchší.

| $x_1$ | $x_2$ | $x_3$ | $F(x_1, x_2, x_3)$ |
|-------|-------|-------|--------------------|
| 0     | 0     | 0     | 1                  |
| 0     | 0     | 1     | 1                  |
| 0     | 1     | 0     | 1                  |
| 0     | 1     | 1     | 0                  |
| 1     | 0     | 0     | 1                  |
| 1     | 0     | 1     | 0                  |
| 1     | 1     | 0     | 0                  |
| 1     | 1     | 1     | 0                  |

Boolova funkcia špecifikovaná touto tabuľkou má tvar

$$\begin{aligned}
 F(x_1, x_2, x_3) &= \bar{x}_1 \bar{x}_2 \bar{x}_3 + \bar{x}_1 \bar{x}_2 x_3 + \bar{x}_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 \bar{x}_3 = \\
 &= \underbrace{\bar{x}_1 \bar{x}_2 \bar{x}_3 + \bar{x}_1 \bar{x}_2 x_3 + \bar{x}_1 x_2 \bar{x}_3}_{\bar{x}_1 \bar{x}_2 \bar{x}_3} + \bar{x}_1 \bar{x}_2 x_3 + \bar{x}_1 x_2 \bar{x}_3 + x_1 \bar{x}_2 \bar{x}_3 = \\
 &= \bar{x}_1 \bar{x}_2 (\underbrace{\bar{x}_3 + x_3}_1) + \bar{x}_1 (\underbrace{x_2 + \bar{x}_2}_1) \bar{x}_3 + (\underbrace{\bar{x}_1 + x_1}_1) \bar{x}_2 \bar{x}_3 \\
 &= \bar{x}_1 \bar{x}_2 + \bar{x}_1 \bar{x}_3 + \bar{x}_2 \bar{x}_3
 \end{aligned}$$

Potom spínací obvod pre automatické zapínanie a vypínanie ústredného kúrenia má podobu

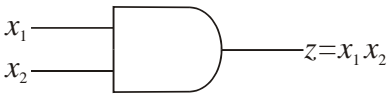
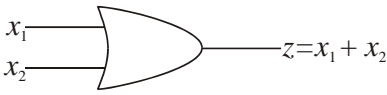
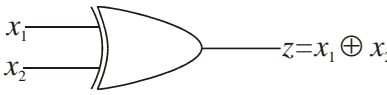
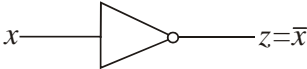


## 10.5 Logické obvody

Na tomto mieste je vhodné pripomenúť si pioniersku prácu McCullocha a Pittsa z r. 1943, v ktorej boli formulované teoretické základy neurónových sietí s logickými neurónmi a ktorá je v súčasnosti pokladaná za jednu prvých prác, na základe ktorých vznikol nový informatický vedný odbor **umelá inteligencia** (pozri Kap. 7). Zmienení autori dokázali, že ľubovoľná Boolova funkcia môže byť simulovaná pomocou obvodu – neurónovej siete – obsahujúcej elementárne obvody – neuróny – ktoré simulujú disjunktívne a konjunktívne klauzuly. V prípade logických obvodov sa jedná o abstrakciu, ktorá ignoruje vnútornú architektúru neurónov a postuluje existenciu elementárnych *logických brán*<sup>1</sup> pre operácie disjunkcie, konjunkcie a negácie, pozri Tab. 10.1.

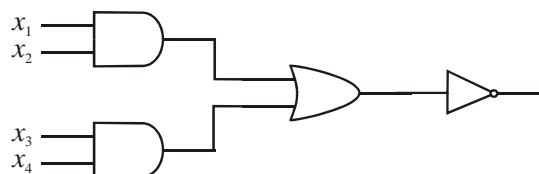
<sup>1</sup> V staršej počítačovej literatúre sa používa termín “hradlo”, v tejto publikácii dôsledne používame termín “brána”, ktorý je prekladom štandardného anglického termínu “gate”.

**Tabuľka 10.1.** Logické brány

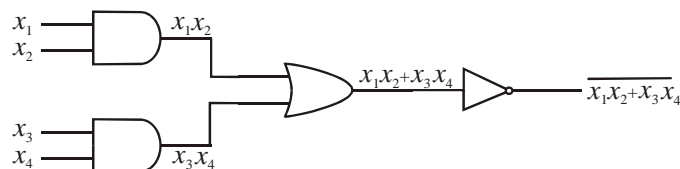
| logická brána konjunkcie  | Logická brána disjunkcie  | Logická brána exkluzívnej disjunkcie  |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|-----------|-------------|---|---|---|---|---|---|---|---|---|---|---|---|-------|-------|-------------|---|---|---|---|---|---|---|---|---|---|---|---|--|-------|-------|------------------|---|---|---|---|---|---|---|---|---|---|---|---|
|    |  |  |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>x_1 x_2</math></th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>0</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> | $x_1$   | $x_2$   | $x_1 x_2$ | 0           | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>x_1 + x_2</math></th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </tbody> </table> | $x_1$ | $x_2$ | $x_1 + x_2$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th><math>x_1</math></th> <th><math>x_2</math></th> <th><math>x_1 \oplus x_2</math></th> </tr> </thead> <tbody> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>0</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </tbody> </table> | $x_1$ | $x_2$ | $x_1 \oplus x_2$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| $x_1$   | $x_2$   | $x_1 x_2$   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0   | 0   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1   | 0   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0   | 0   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1   | 1   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| $x_1$   | $x_2$   | $x_1 + x_2$   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0   | 0   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1   | 1   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0   | 1   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1   | 1   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| $x_1$   | $x_2$   | $x_1 \oplus x_2$  |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 0   | 0   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1   | 1   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0   | 1   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 1   | 0   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| logická brána negácie   |   |   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
|    |   |   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th><math>x_1</math></th> <th><math>\bar{x}_1</math></th> </tr> </thead> <tbody> <tr><td>0</td><td>1</td></tr> <tr><td>1</td><td>0</td></tr> </tbody> </table>   |   |   | $x_1$     | $\bar{x}_1$ | 0 | 1 | 1 | 0 |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| $x_1$   | $\bar{x}_1$   |   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 0   | 1   |   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |
| 1   | 0   |   |           |             |   |   |   |   |   |   |   |   |   |   |   |   |       |       |             |   |   |   |   |   |   |   |   |   |   |   |   |  |       |       |                  |   |   |   |   |   |   |   |   |   |   |   |   |

Konjunktívna a disjunktívna brána má dva binárne vstupy a jeden binárny výstup, jednoduchšia brána negácie má jeden binárny vstup a jeden binárny výstup.

**Príklad 10.15.** Zostrojte Boolovu funkciu pre logický obvod



Použitím tabuľky 10.1 jednotlivé spoje tohto logického obvodu ohodnotíme takto

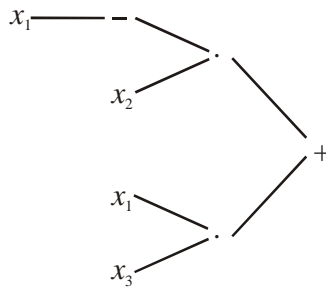


To znamená, že Boolova funkcia priradená tomuto obvodu má tvar

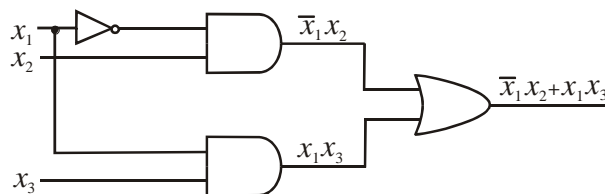
$$f(x_1, x_2, x_3, x_4) = \overline{x_1 x_2 + x_3 x_4}$$



**Príklad 10.16.** Zostrojte logický obvod, ktorý simuluje Boolovu funkciu  $f(x_1, x_2, x_3, x_4) = \bar{x}_1 \cdot x_2 + x_1 \cdot x_3$ . Pre ilustráciu zostrojíme syntaktický strom tejto funkcie



Logický obvod zostrojíme jednoducho z tohto syntaktického stromu, ktorý interpretuje Boolovu funkciu  $f(x_1, x_2, x_3, x_4) = \bar{x}_1 \cdot x_2 + x_1 \cdot x_3$ , tak, že jednotlivé vrcholy znázorňujúce algebraické operácie nahradíme príslušnými logickými bránami



### 10.5.1 Sumátor dvoch binárnych čísel (polosumátor)

Technika logických obvodov je aplikovateľná ku konštrukcii súčtu dvoch kladných celých čísel, ktoré sú prezentované v binárnej reprezentácii. Táto konštrukcia obsahuje tri etapy. Prvá etapa spočíva v návrhu logického obvodu (nazývaného *polosumátor*) ktorý sčíta dve jednobitové čísla  $x$  a  $y$

$$\begin{array}{r} x \\ y \\ \hline c \quad s \end{array} \quad (10.25)$$

Uvedieme tabuľku všetkých prípadov tejto schémy, ktoré môžu nastať

| Vstup |     | výstup |     |
|-------|-----|--------|-----|
| $X$   | $y$ | $c$    | $s$ |
| 0     | 0   | 0      | 0   |
| 0     | 1   | 0      | 1   |
| 1     | 0   | 0      | 1   |
| 1     | 1   | 1      | 0   |

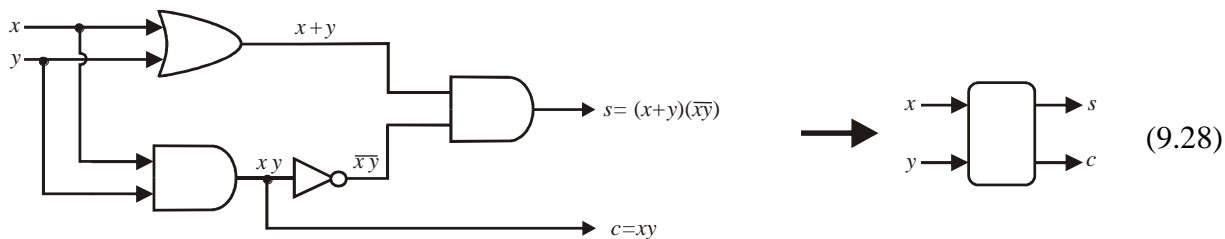
(11.26)

Ak použijeme vetu 10.5 výstupné premenné z tejto tabuľky sú určené pomocou sumácie klauzúl (v DNF tvare)

$$s = f(x, y) = \bar{x}y + x\bar{y} = (x + y)(\bar{x} + \bar{y}) = (x + y)(\overline{xy}) \quad (10.27a)$$

$$c = g(x, y) = xy \quad (10.27b)$$

kde pri konštrukcii alternatívnej pravej strany Boolovej funkcie  $f$  bol použitý distributívny zákon. Tieto dva výsledky znázorníme pomocou logického obvodu



### 10.5.2 Sumátor troch binárnych čísel (dvojité sumátor)

Konštrukcii logického obvodu (nazývaného *dvojité sumátor*) pre sčítanie troch jednobitových čísel

$$\begin{array}{r}
 x \\
 y \\
 \hline
 z \\
 u \ v
 \end{array}
 \quad (10.29)$$

Všetky možné prípady tejto schémy sú uvedené v tabuľke

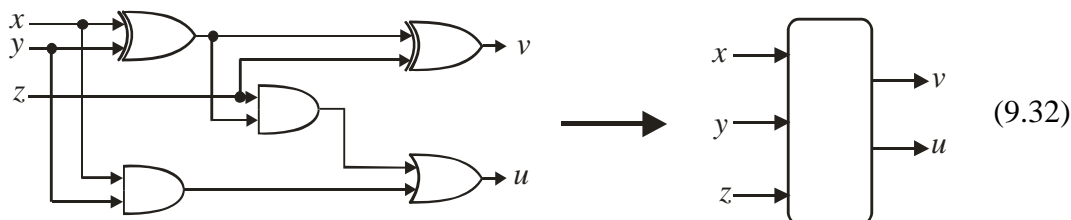
| Vstup |     |     | Výstup |     |
|-------|-----|-----|--------|-----|
| $x$   | $y$ | $z$ | $u$    | $v$ |
| 0     | 0   | 0   | 0      | 0   |
| 0     | 0   | 1   | 0      | 1   |
| 0     | 1   | 0   | 0      | 1   |
| 0     | 1   | 1   | 1      | 0   |
| 1     | 0   | 0   | 0      | 1   |
| 1     | 0   | 1   | 1      | 0   |
| 1     | 1   | 0   | 1      | 0   |
| 1     | 1   | 1   | 1      | 1   |

(11.30)

Pomocou formuly (10.6) môžeme zostrojiť z tejto tabuľky DNF Boolovu funkciu pre výstupy  $u$  a  $v$

$$\begin{aligned}
 u &= \bar{x}yz + x\bar{y}z + xy\bar{z} + xyz = (\bar{x}y + x\bar{y})z + xy(z + \bar{z}) \\
 &= (x \oplus y)z + xy \\
 v &= \bar{x}\bar{y}z + \bar{x}y\bar{z} + x\bar{y}\bar{z} + xyz = (\bar{x}y + x\bar{y})\bar{z} + (\bar{x}\bar{y} + xy)z \\
 &= (x \oplus y) \oplus z
 \end{aligned}
 \quad (10.31)$$

Výstupné veličiny  $u$  a  $v$  môžu byť reprezentované pomocou spoločnej Boolovej funkcie, ktorá je reprezentovaná obvodom



(9.32)

## 10.6 Optimalizácia logických obvodov

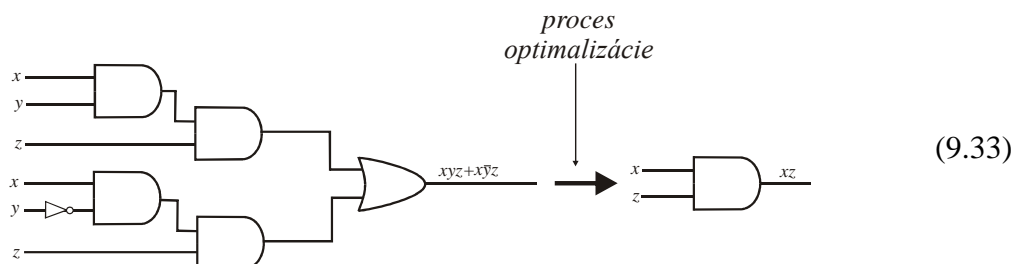
Efektívnosť logických obvodov závisí na počte a prepojení jej logických brán. Proces návrhu logického obvodu začína návrhom tabuľky špecifikujúcej Boolovu funkciu, ktorá transformuje vstupné binárne veličiny na výstupné binárne veličiny. Boolova funkcia je zostrojená podľa vety 10.5 pomocou súčtu konjunktívnych klauzúl. Aj keď je tento prístup ku konštrukcii Boolovej funkcie všeobecný, veta 10.5 nezaručuje optimálnosť zostrojenej funkcie. Pod optimálnosťou rozumieme to, že Boolova funkcia, ktorá simuluje danú tabuľku funkčných hodnôt pre všetky kombinácie vstupných hodnôt, má minimálny počet literálov.

Uvažujme logický obvod, ktorý má výstup 1 vtedy a len vtedy, ak  $x = y = z = 1$  alebo ak  $x = z = 1$  a  $y = 0$ . Boolova funkcia zostrojená podľa vety 10.5, ktorá simuluje tento logický obvod má tvar  $f(x, y, z) = xyz + x\bar{y}z$ , môže byť podstatne zjednodušená takto

$$f(x, y, z) = xyz + x\bar{y}z = x(y + \bar{y})z = x \cdot 1 \cdot z = xz$$

To znamená, že táto funkcia  $xz$ , ktorá obsahuje dva literály, je ekvivalentná s pôvodnou funkciou obsahujúcej 6 literálov; môžeme teda povedať, že optimálny tvar Boolovej funkcie  $f(x, y, z) = xyz + x\bar{y}z$  je nová funkcia  $g(x, z) = xz$ , ktorá aj keď je s pôvodnou ekvivalentná, má podstatne menej literálov.

Tento jednoduchý príklad dostatočne jasne ukazuje dôležitosť hľadať optimálny tvar Boolovej funkcie, pôvodne zostrojenej podľa vety 10.5, ak má táto slúžiť ako podklad pre návrh logického obvodu. Optimálny tvar Boolovej funkcie môže v špeciálnych prípadoch podstatne zjednodušiť navrhovaný logický obvod. Ako ilustračný príklad budeme študovať logické obvody priradené Boolovej funkcii  $f(x, y, z) = xyz + x\bar{y}z$  a jej optimálnej formy  $g(x, z) = xz$



Logický obvod zostrojený pomocou optimálneho tvaru (vpravo), obsahujúceho minimálny počet literálov je podstatne jednoduchší ako logický obvod (vľavo), ktorý obsahuje šesť logických hradieľ.

### 10.6.1 Quinova a McCluskeyho optimalizačná metóda [2,5,6]

Táto metóda patrí medzi často používané prístupy k optimalizácii Boolových funkcií, jej hlavnou prednosťou pred ostatnými prístupmi je jej konceptuálna jednoduchosť a priamočiara algoritmizovateľnosť.

Quinovu a McCluskeyho metóda [2,5,6] bude ilustrovaná konkrétnym prípadom optimalizácie jednoduchej Boolovej funkcie

$$f(x, y, z) = xyz + x\bar{y}z + \bar{x}yz + \bar{x}\bar{y}z + \bar{x}\bar{y}\bar{z} \quad (10.34)$$

Každá konjunktívna klauzula (pozri definícia 10.6 a formulu (10.3)) môže byť reprezentovaná bitovým reťazcom  $e = (e_1, e_2, e_3) \in \{0, 1\}^3$

$$l_e(x, y, z) = x^{e_1} y^{e_2} z^{e_3} \rightarrow (e_1, e_2, e_3) \quad (10.35)$$

kde  $\xi^e = \xi$ , ak  $e = 1$ ,  $\xi^e = \bar{\xi}$ , ak  $e = 0$ , pre  $\xi = x, y, z$ . Pre takto definovanú binárnu reprezentáciu môžeme použiť metriku Hammingovej vzdialenosti ku kvantifikácii podobnosti medzi binárnymi vektormi. Nech  $\mathbf{e}_i = (e_1^{(i)}, e_2^{(i)}, \dots, e_n^{(i)})$  a  $\mathbf{e}_j = (e_1^{(j)}, e_2^{(j)}, \dots, e_n^{(j)})$  sú dve binárne reprezentácie klauzúl, potom

$$d_H(\mathbf{e}_i, \mathbf{e}_j) = \sum_{k=1}^n |e_k^{(i)} - e_k^{(j)}| \quad (10.36)$$

Táto vzdialenosť pre binárne vektory nám špecifikuje počet polôh v ktorých sa binárne vektory vzájomne odlišujú. Napríklad, ak Hammingova vzdialenosť medzi dvoma binárnymi vektormi je 2, potom tieto vektory sa navzájom odlišujú na dvoch miestach binárneho reťazca.

Každá Boolova formula v DNF forme je 1-1-značne reprezentovaná pomocou množiny bitových reťazcov, ktoré sú priradené jednotlivým klauzulám študovanej formuly. Pre ilustračný príklad (A.39) táto množina obsahuje 5 binárnych reťazcov dĺžky 3

$$U_f = \{(111), (101), (011), (001), (000)\} \quad (10.37)$$

Táto možnosť reprezentovať Boolovu funkciu v DNF forme pomocou množiny binárnych vektorov vyplýva zo skutočnosti, že operácia sumácie je komutatívna a asociatívna, čiže nezáleží v akom poradí sčítame jednotlivé klauzuly v Boolovej funkcii.

Dve klauzuly z množiny  $U_f$  môžu byť vzájomne sčítané do jednej klauzuly vtedy a len vtedy ak sa líšia ich binárne reťazce práve v jednej polohe, čiže ak ich vzájomná Hammingova vzdialenosť sa rovná 1. Napríklad, z množiny (10.42) vyberieme prvú a druhú klauzulu, ich binárne reprezentácie (111) a (101) sa líšia len hodnotou binárnej premennej v druhej polohe,  $d_H(111, 101) = 1$ . Tieto dve klauzuly sú sčítané takto

$$xyz + x\bar{y}z = x \left( \underset{1}{y + \bar{y}} \right) z = xz \quad (10.38a)$$

V binárnej reprezentácii tento proces zjednodušenia formálne vyjadríme takto

$$(111) + (101) = \text{sum}(\underbrace{(111), (101)}_{\#}) = (1\#1) \quad (10.38b)$$

kde bol použitý nový „prázdny“ symbol '#', ktorý reprezentuje prázdne miesto v binárnej reprezentácii novej klauzuly  $xz$ , pozri (A.43a). Takto zostrojené nové klauzuly obsahujúce jeden symbol '#' tvoria množinu

$$U_f^{(1)} = \{(1\#1), (\#11), (\#01), (0\#1), (00\#)\} \quad (10.39)$$

V ďalšej etape vytvárame z množiny  $U_f^{(1)}$  novú množinu  $U_f^{(2)}$ , ktorá obsahuje klauzuly s dvoma prázdnyimi symbolmi '#' a ktoré boli vytvorené operáciou súčtu klauzúl z množiny  $U_f^{(1)}$

$$U_f^{(2)} = \{(\#\#1)\} \quad (10.40)$$

Proces sčítania klauzúl obsahujúcich symboly '#' musí byť podrobnejšie špecifikovaný:

- (a) Sčítať môžeme len také dve klauzuly, ktoré obsahujú rovnaký počet symbolov '#', pričom tieto symboly v oboch použitých klauzulách musia byť umiestnené v rovnakých polohách v oboch binárných reprezentáciách.
- (b) Klauzuly, ktoré vyhovujú podmienke (a) môžeme sčítať len vtedy, ak ich binárne komponenty sa líšia len v jednej polohe.

Uvažujme dve klauzuly  $e_i = (e_1^{(i)}, e_2^{(i)}, \dots, e_n^{(i)})$  a  $e_j = (e_1^{(j)}, e_2^{(j)}, \dots, e_n^{(j)})$ , ktoré chceme sčítať. Podľa podmienky (a) musí existovať taká množina indexov  $I \subseteq \{1, 2, \dots, n\}$ , že

$$(\forall k \in I)(e_k^{(i)} = e_k^{(j)} = \#) \text{ a } (\forall k \notin I)(e_k^{(i)}, e_k^{(j)} \in \{0, 1\}) \quad (10.41)$$

Ako príklad uvidíme dvojicu klauzúl  $e_i = (10\#0\#11)$  a  $e_j = (11\#0\#11)$ , pre množinu indexov  $I = \{3, 5\}$  platia obe podmienky. V prípade, že takáto množina neexistuje, potom klauzuly  $e_i = (e_1^{(i)}, e_2^{(i)}, \dots, e_n^{(i)})$  a  $e_j = (e_1^{(j)}, e_2^{(j)}, \dots, e_n^{(j)})$  nemôžu byť použité v procese sčítania klauzúl. Zavedieme zovšeobecnenú Hammingovu vzdialenosť pre klauzuly, ktoré vyhovujú podmienke (a)

$$d_H(e_i, e_j) = \sum_{k \in I} |e_k^{(i)} - e_k^{(j)}| \quad (10.42)$$

t. j. v sumácii sú aktívne len binárne členy. Potom podmienka (b) požaduje, že Hammingova vzdialenosť medzi klauzulami musí byť rovná 1. Bez zníženia všeobecnosti môžeme postulovať, že Hammingova vzdialenosť medzi vektormi  $e_i = (e_1^{(i)}, e_2^{(i)}, \dots, e_n^{(i)})$  a  $e_j = (e_1^{(j)}, e_2^{(j)}, \dots, e_n^{(j)})$ , ktoré nevyhovujú podmienke (a) (majú buď rôzny počet symbolov symbol '#', alebo ak majú rovnaký počet týchto symbolov, potom ich polohy nie sú rovnaké) je nekonečne veľká. Týmto jednoduchým predpokladom máme zabezpečené, že podmienka (a) je splnená.

Zavedieme operátor  $\mathcal{A}$ , ktorý špecifikuje prechod množiny  $U_f^{(k)}$  na množinu  $U_f^{(k+1)}$

$$U_f^{(k+1)} = \mathcal{A}(U_f^{(k)}) \quad (10.43)$$

Musí existovať také kladné celé číslo  $n$ , že tento proces tvorby nových množín je ukončený, t. j. platí  $U_f^{(n+1)} = \mathcal{A}(U_f^{(n)}) = \emptyset$ . Rekurentný proces (10.43) je inicializovaný množinou  $U_f^{(0)} = U_f$ . Môžeme hovoriť o etapách procesu tvorby nových klauzúl z pôvodnej (počiatočnej) množiny klauzúl. V 1. etape vytvoríme procesom sčítania dvoch klauzúl z množiny  $U_f^{(0)} = U_f$  klauzuly s jedným prázdny symbolom '#', v 2. etape vytvoríme z množiny  $U_f^{(1)}$  klauzuly s dvoma symbolmi '#'. Tento rekurentný proces je ukončený vtedy, ak operátor  $\mathcal{A}$  aplikovaný na množinu  $U_f^{(n)}$  produkuje prázdnu množinu, t. j.  $\mathcal{A}(U_f^{(n)}) = \emptyset$ .

Ako ilustračný príklad rekurentnej tvorby množín  $U_f^{(k)}$  použijeme množinu (A.42), výsledky je možné uviesť vo forme tabuľky

| 0. etapa |       |  | 1. etapa |       |       | 2. etapa |              |       |
|----------|-------|--|----------|-------|-------|----------|--------------|-------|
| 1        | (111) |  | 1        | (1,2) | (1#1) | 1        | (1,3), (2,4) | (##1) |
| 2        | (101) |  | 2        | (1,3) | (#11) |          |              |       |
| 3        | (011) |  | 3        | (2,4) | (#01) |          |              |       |
| 4        | (001) |  | 4        | (3,4) | (0#1) |          |              |       |
| 5        | (000) |  | 5        | (4,5) | (00#) |          |              |       |

V stĺpcoch pre prvú a druhú etapu sú uvedené aj dvojice indexov klauzúl z predchádzajúceho stĺpca, ktoré boli použité v sumačnom procese.

Stojíme pred úlohou, ako vybrať taký minimálny počet klauzúl zostrojených v prvej alebo v ďalších etapách, ktoré sú odvoditeľné zo všetkých pôvodných klauzúl z množiny

$U_f^{(0)} = U_f$ . K presnej špecifikácii tohto posledného kroku Quinovej a McCluskyeho metódy musíme zaviesť nový pojem „pokrytie“. Klauzula  $e'$  **pokrýva** klauzulu  $e$ , formálne  $e' \subseteq e$ , vtedy a len vtedy, ak platí pre každé  $i=1,2,\dots,n$  práve jedna z týchto podmienok

1.  $(e'_i = 1) \Rightarrow (e_i = 1)$
  2.  $(e'_i = 0) \Rightarrow (e_i = 0)$
- (10.44)
3.  $(e'_i = \#) \Rightarrow (e_i \in \{0,1,\#\})$

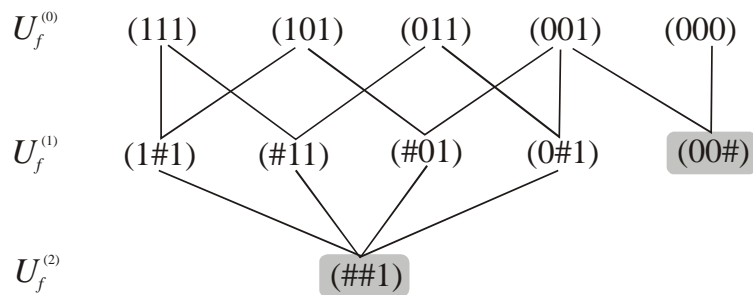
Pre ilustráciu tejto relácie uvedieme jednoduchý ilustračný príklad ktorý vyhovuje týmto podmienkam

|   |   |   |   |   |   |   |   |   |           |
|---|---|---|---|---|---|---|---|---|-----------|
| 0 | 1 | # | 1 | # | 0 | 1 | 0 | 1 | <b>E</b>  |
| 0 | 1 | # | 1 | # | # | 1 | 0 | # | <b>e'</b> |

Lahko sa presvedčíme, že sa jedná o **reláciu čiastočného usporiadania** nad množinou vektorov  $e \in \{0,1,\#\}^n$  (pozri knihu [xx], kapitolu 3.2). Potom množina klauzúl, ktorá vznikne zjednotením množín  $U_f^{(0)}, U_f^{(1)}, U_f^{(2)}, \dots$

$$\tilde{U}_f = U_f^{(0)} \cup U_f^{(1)} \cup U_f^{(2)} \cup \dots$$

je čiastočne usporiadaná a diagramaticky znázornená Hasseho diagramom



Z tohto diagramu vyplýva, že má 5 maximálnych klauzúl (klauzuly patriace do množiny  $U_f^{(0)}$ ) a dve minimálne klauzuly ( $\#\#\#$ ) a  $(00\#)$ , ktoré sú na Hasseho diagrame vysvietené.

Pre každú klauzulu  $e$  obsahujúcu aspoň jeden prázdny symbol,  $e \in U_f^{(1)} \cup U_f^{(2)} \cup \dots$ , zostrojíme množinu  $U(e) \subseteq U_f$ , ktorá obsahuje všetky pôvodné klauzuly (neobsahujúce prázdne symboly #), ktoré sú pokryté klauzulou  $e$

$$U(e) = \{e' ; (e' \in U_f) \wedge (e \subseteq e')\} \quad (10.45)$$

Pre lepšie pochopenie tejto množinovej koncepcie uvedieme príklady tejto množiny vychádzajúce z vyššie uvedeného Hassovho diagramu

$$U(1\#1) = \{(111), (101)\}$$

$$U(\#11) = \{(111), (011)\}$$

$$U(\#01) = \{(101), (001)\}$$

$$U(0\#1) = \{(011), (001)\}$$

$$U(00\#) = \{(001), (000)\}$$

$$U(\#\#\#) = \{(1,1,1), (101), (011), (001)\}$$

kde posledné dve klauzuly sú označené ako minimálne

Naším cieľom je vybrať také minimálne klauzuly, ktoré pokrývajú pôvodné klauzuly z množiny  $U_f^{(0)}$ . Množinu týchto minimálnych klauzúl označíme  $V$ , potom v rámci tejto množiny hľadáme takú podmnožinu  $\tilde{V} \subseteq V$ , ktorej klauzuly plne pokrývajú množinu  $U_f^{(0)}$

$$\bigcup_{e \in \tilde{V}} U(e) = U_f \quad (10.46)$$

Podmnožina je určená podmienkou minimálnosti počtu literálov,  $[\tilde{V}]$ , ktoré obsahuje

$$\tilde{V} = \arg \min_{V' \subseteq V} [V'] \quad (10.47)$$

Riešenie tohto optimalizačného problému je pre malý počet premenných (cca do päť) obvykle zvládnuteľný ručne tak, že preberieme všetky možnosti, ktoré pokrývajú množinu  $U_f$ . Pre väčšie problémy môže byť použitá metóda spätného prehľadávania, ktorá systematicky preskúma všetky možnosti. Žiaľ, tento prístup je nepoužiteľný pre niekoľko desiatok premenných, v dôsledku exponenciálneho rastu zložitosti. V praxi sa používajú rôzne heuristické metódy, ktoré poskytujú kvalitne suboptimálne riešenie, ktoré je často rovné optimálnemu riešeniu. V ďalšej časti tejto kapitoly budeme diskutovať veľmi jednoduchú metódu uskutočnenia optimálneho pokrytia pôvodných (maximálnych klauzúl bez prázdných symbolov #), ktorú zo súčasného pohľadu algoritmov môžeme nazvať "greedy" metóda [xx].

V tomto konkrétnom príklade sa jedná o extrémne jednoduchý problém, môžeme vybrať obe minimálne klauzuly (##1) a (00#), ktoré pokrývajú pôvodné klauzuly. Potom môžeme písať Boolovu funkciu (7.39) v ekvivalentnom tvare

$$f(x, y, z) = z + \bar{x} \bar{y}$$

čo reprezentuje podstatné zjednodušenie (optimalizáciu) pôvodnej Boolovej funkcie (7.39), ktorej počet literálov z 15 klesol na 3.

**Príklad 10.17.** Quinova a McCluskeyho metóda pôvodne nevyužívala algoritmus spätného prehľadávania, bola založená na heuristickom prístupe naformulovanom v podobe tabuliek a ktorý v súčasnosti nazývame „greedy“ metóda. Uvažujme Boolovu funkciu

$$f(w, x, y, z) = wxy\bar{z} + w\bar{x}yz + \bar{w}xyz + w\bar{x}\bar{y}z + \bar{w}x\bar{y}z + \bar{w}\bar{x}yz + \bar{w}\bar{x}\bar{y}z$$

V nasledujúcej tabuľke je znázornený postup vytvárania všetkých možných sumácií medzi klauzulami (v binárnej reprezentácii) k tejto Boolovej funkcie

| 0. etapa |        |  | 1. etapa |       |        | 2. etapa |       |        |
|----------|--------|--|----------|-------|--------|----------|-------|--------|
| 1        | (1110) |  | 1        | (1,4) | (1#10) | 1        | (4,7) | (0##1) |
| 2        | (1011) |  | 2        | (2,4) | (101#) | 23       | (5,6) | (0##1) |
| 3        | (0111) |  | 3        | (2,6) | (#011) |          |       |        |
| 4        | (1010) |  | 4        | (3,5) | (01#1) |          |       |        |
| 5        | (0101) |  | 5        | (3,6) | (0#11) |          |       |        |
| 6        | (0011) |  | 6        | (5,7) | (0#01) |          |       |        |
| 7        | (0001) |  | 7        | (6,7) | (00#1) |          |       |        |

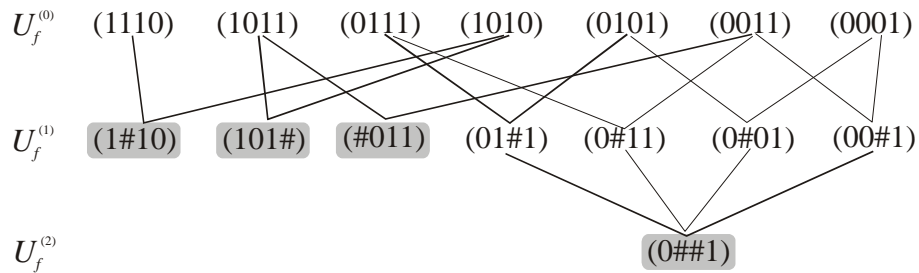
V 0. etape máme zoznam všetkých klauzúl z Boolovej funkcie, množina  $U_f^{(0)}$  má tvar

$$U_f^{(0)} = \{(1110), (1011), (0111), (1010), (0101), (0011), (0001)\}$$

V 1. etape vytvoríme sumáciami nové klauzuly, ktoré obsahujú jeden prázdny znak # (tieto klauzuly sú uvedené v šiestom stĺpci vyššie uvedenej tabuľky

$$U_f^{(1)} = \mathcal{A}(U_f^{(0)}) = \{(1\#10), (101\#), (\#011), (01\#1), (0\#11), (0\#01), (00\#1)\}$$

V 2. etape vytvoríme množinu kroku množina  $U_f^{(2)} = \mathcal{A}(U_f^{(1)}) = \{(0\#\#1)\}$  obsahuje už len jednu klauzulu  $(0\#\#1)$ , ktorá vznikla dvoma alternatívnymi spôsobmi, súčtom klauzúl 5 a 6 resp. 4 a 7 zo 6. stĺpca. Hasseho diagram má tvar



Teraz stojíme pred problémom ako vybrať taký minimálny počet klauzúl, ktoré nám budú pokrývať celú pôvodnú množinu klauzúl. V prvom kroku vyberieme takú minimálnu klauzulu, ktorá pokrýva najväčší počet maximálnych klauzúl. V prípade, že existuje niekoľko rovnocenných minimálnych klauzúl, ktoré pokrývajú rovnaký počet maximálnych klauzúl, tak vyberieme takú minimálnu klauzulu, ktorá má minimálny počet literálov. V tomto prvom kroku vyberieme minimálnu klauzulu  $(0\#\#1)$ , ktorá pokrýva štyri maximálne klauzuly. V ďalšom kroku máme dve alternatívne možnosti, a to výber minimálnej klauzuly  $(1\#10)$  alebo výber minimálnej klauzuly  $(101\#)$ . V oboch prípadoch tieto minimálne klauzuly pokrývajú dve maximálne klauzuly, pretože majú rovnaký počet literálov, tak sú rovnocenné. Na záver máme dve rovnocenné možnosti výberu minimálnych klauzúl  $(101\#)$  resp.  $(\#011)$ , vybrali sme prvú možnosť  $(101\#)$ , tým sme dokázali, že sedem maximálnych klauzúl môže byť pokrytých pomocou troch minimálnych klauzúl  $(0\#\#1)$ ,  $(1\#10)$  a  $(101\#)$ , ktoré majú dohromady osem literálov. Ekvivalentná (minimálna) Boolova funkcia, ktorá je určená týmito klauzulami má tvar

$$f_1(w, x, y, z) = \bar{w}z + wy\bar{z} + w\bar{x}y$$

Ak by sme vybrali druhú alternatívnu možnosť  $(\#011)$ , potom alternatívny tvar minimálnej Boolovej funkcie je

$$f_2(w, x, y, z) = \bar{w}z + wy\bar{z} + \bar{x}yz$$

## Cvičenia

**Cvičenie 10.1.** Aká je hodnota Boolovej premennej, ktorá je určená podmienkou

- (a)  $x \cdot \mathbf{1} = \mathbf{0}$ ,
- (b)  $x + x = \mathbf{0}$ ,
- (c)  $x \cdot \mathbf{1} = x$ ,
- (d)  $x + \bar{x} = \mathbf{1}$ .
- (e)  $x \cdot \bar{x} = \mathbf{0}$

**Cvičenie 10.2.** Zostrojte tabuľku funkčných hodnôt Boolovej funkcie

- (a)  $f(x, y, z) = \bar{x}y$ ,
- (b)  $f(x, y, z) = x + yz$ ,
- (c)  $f(x, y, z) = \bar{x}\bar{y} + \overline{xyz}$ ,



**Cvičenie 10.3.** Znázornite Boolove funkcie  $f(x, y, z)$  z cvičenia A.2 na 3-rozmernej kocke tak, že hodnoty 1 (0) budú reprezentované na kocke čiernym (bielym) bodom.

**Cvičenie 10.4.** Pre ktoré hodnoty  $x$  a  $y$  platí  $xy = x + y$ .

**Cvičenie 10.5.** Zostrojte tabuľku všetkých možných binárnych Boolových funkcií a identifikujte v nej známe Boolove binárne operácie súčinu a súčtu. Vyjadrite ostatné binárne operácie pomocou súčtu, súčinu a komplementu.

**Cvičenie 10.6.** Riešte nasledujúce rovnice s exkluzívnou disjunkciou

- (a)  $x \oplus \mathbf{0}$ ,
- (b)  $x \oplus \mathbf{1}$ ,
- (c)  $x \oplus x$ ,
- (d)  $x \oplus \bar{x}$ .

**Cvičenie 10.7.** Dokážte, že platia rovnosti

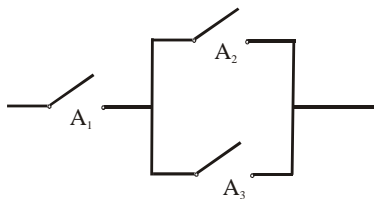
- (a)  $x \oplus y = (x + y)(\overline{xy})$ ,
- (b)  $x \oplus y = \bar{x}y + x\bar{y}$ .

**Cvičenie 10.8.** Zostrojte Boolovu funkciu  $f(x, y, z)$  vo forme sumy produktov klauzúl k premenným  $x, y$  a  $z$ , ktorá má hodnotu **1** vtedy a len vtedy, ak

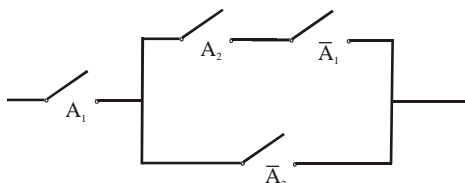
- (a)  $x = y = \mathbf{0}, z = \mathbf{1}$ ,
- (b)  $x = \mathbf{0}, y = \mathbf{1}, z = \mathbf{0}$ ,
- (c)  $y = z = \mathbf{1}$ .

**Cvičenie 10.9.** Zostrojte spínacie funkcie pre spínacie obvody

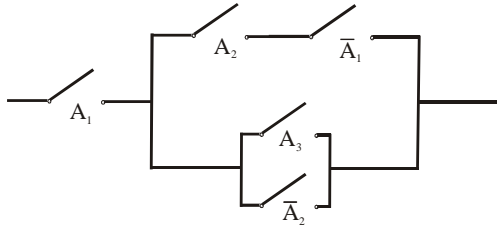
(a)



(b)

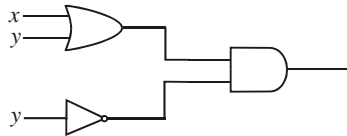


(c)

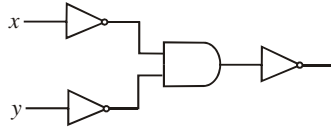


**Cvičenie 10.10.** Zostrojte tabuľku výstupov logických obvodov

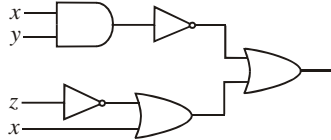
(a)



(b)



(c)



**Cvičenie 10.11.** Zostrojte logické obvody, ktoré simulujú Boolove funkcie

(a)  $\bar{x} + y$ ,

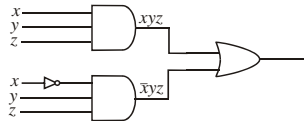
(b)  $\overline{(x + y)}x$ ,

(c)  $xyz + \bar{x} \bar{y} \bar{x}$ ,

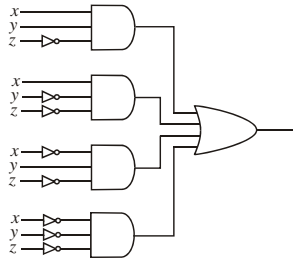
(d)  $\overline{(\bar{x} + z)(y + \bar{z})}$ .

**Cvičenie 10.12.** Zjednodušte logické obvody

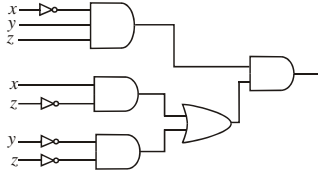
(a)



(b)



(c)



**Cvičenie 10.13.** Pomocou Quinovej a McCluskeyho metódy nájdite optimálne výrazy k Boolovým funkciám

(a)  $wxyz + wx\bar{y}z + wx\bar{y}\bar{z} + w\bar{x}y\bar{z} + w\bar{x}\bar{y}z$ ,

(b)  $wxy\bar{z} + wx\bar{y}z + w\bar{x}yz + \bar{w}x\bar{y}z + \bar{w}\bar{x}y\bar{z} + \bar{w}\bar{x}\bar{y}z$ ,

(c)  $wxyz + wxy\bar{z} + wx\bar{y}z + w\bar{x}\bar{y}z + w\bar{x}y\bar{z} + \bar{w}x\bar{y}z + \bar{w}\bar{x}y\bar{z} + \bar{w}\bar{x}\bar{y}z$ .

## Literatúra

- [1] Astola, J. T., Stankovic, R. S.: *Fundamentals of Switching Theory and Logic Design*. Springer, Berlin, 2006.
- [2] Frištacký, N., Kolesár, M., Kolenička, J., Hlavatý, J.: *Logické systémy*. Alfa, Bratislava, 1990.
- [3] Chen, W. K.: *Logic Desing*. CRC Press, Boca Raton, 2003.
- [4] Kvasnička V., Pospíchal, J.: *Algebra a diskrétna matematika*. Vydavateľstvo STU, Bratislava, 2008.
- [5] Kvasnička V., Pospíchal, J.: *Matematická logika*. Vydavateľstvo STU, Bratislava, 2006.
- [6] Shannon, C. E.: A Symbolic Analysis of Relay and Switching Circuit. *AIEE Transactions*, **57** (1938), 713- 723.

