Logické systémy

doc. RNDr. Jana Galanová, PhD.

RNDr. Peter Kaprálik, PhD.

Mgr. Marcel Polakovič, PhD.

KAPITOLA 1

Úvodné pojmy

V tejto časti uvádzame základné pojmy, prevažne z diskrétnej matematiky, ktoré sú potrebné pre štúdium logických systémov. V prvom rade sa dohodnime na tomto označení číselných množín:

$\mathbf{N} = \{0, 1, 2, \dots, n, \dots\}$,
$\mathbf{N}^+ = \{1, 2, \dots, n, \dots\}$,
$\mathbf{Z}$ – množina všetkých celých čísel,
$\mathbf{R}$ – množina všetkých reálnych čísel,
$\mathbf{R}^+$ – množina všetkých kladných reálnych čísel.

1. Zobrazenia a operácie

Definícia 1.1. Nech A, B sú množiny. Pravidlo (predpis) f, ktoré každému prvku $x \in A$ priradí jeden prvok $y \in B$, sa nazýva ***zobrazenie množiny*** A ***do množiny*** B a označuje sa $f : A \to B$. Prvok y priradený prvku x nazývame ***obrazom prvku*** x a píšeme $y = f(x)$ (tiež $x \mapsto y$). Prvok x nazývame ***vzorom prvku*** y. Množina A sa nazýva ***obor*** (tiež ***definičný obor***) a množina B ***koobor zobrazenia*** f.

Príklad 1.1. Nech $A = \{a, b, c\}, B = \{0, 1\}$. Potom $f : A \to B; a \mapsto 1, b \mapsto 0, c \mapsto 1$ je zobrazenie množiny A do množiny B, lebo každý prvok množiny A má len jeden obraz. ■

Definícia 1.2. Zobrazenie $f : A \to B$ sa nazýva

injektívne (tiež ***prosté***), ak pre každé $x_1, x_2 \in A$, platí: ak $x_1 \neq x_2$, potom $f(x_1) \neq f(x_2)$;
surjektívne, ak každý prvok $y \in B$ má vzor (t.j. ku každému $y \in B$ existuje také $x \in A$, že $y = f(x)$);
bijektívne, ak je injektívne aj surjektívne.

Príklad 1.2. Zobrazenie f z predchádzajúceho príkladu nie je injektívne, lebo $a \neq c$, ale $f(a) = 1 = f(c)$. Je však surjektívne, lebo každý prvok množiny B má vzor, konkrétne vzorom prvku 0 je b a vzorom prvku 1 je a. ■

Definícia 1.3. ***Karteziánskym súčinom množín*** $M_1, M_2, \dots, M_n$ nazývame množinu

$$M_1 \times M_2 \times \cdots \times M_n = \{(x_1, x_2, \dots, x_n); x_1 \in M_1, x_2 \in M_2, \dots, x_n \in M_n\}.$$

V prípade, že $M_1 = M_2 = \cdots = M_n = M$, namiesto $M \times M \times \cdots \times M$ píšeme M^n.
Prvok $(x_1, x_2, \dots, x_n)$ sa nazýva ***usporiadaná*** n***-tica***.
Usporiadané n-tice $(x_1, x_2, \dots, x_n)$, $(y_1, y_2, \dots, y_n)$ sú rovnaké práve vtedy, keď $x_1 = y_1$, $x_2 = y_2, \dots, x_n = y_n$.

Príklad 1.3. Karteziánskym súčinom množín $A = \{a, b, c\}, B = \{0, 1\}$ je množina $A \times B = \{(a, 0), (a, 1), (b, 0), (b, 1), (c, 0), (c, 1)\}$. ■

DEFINÍCIA 1.4. Nech A je množina, n je prirodzené číslo. Každé zobrazenie $h : A^n \to A$ nazývame ***n-árnou operáciou na množine*** A. Špeciálne pre $n = 1$ sa operácia h nazýva ***unárna*** a pre $n = 2$ ***binárna***.

PRÍKLAD 1.4. Funkcia $f(x, y) = x + y$ je binárna operácia na množine $\mathbf{R}$, pretože je definovaná pre každé $x, y \in \mathbf{R}$ a tiež $f(x, y) \in \mathbf{R}$.

Funkcia $g(x, y) = x \cdot y$ je tiež binárnou operáciou na $\mathbf{R}$, pretože jej definičný obor je $\mathbf{R} \times \mathbf{R}$ a pre každé $x, y \in \mathbf{R}$ platí $g(x, y) \in \mathbf{R}$.

Funkcia $h(x, y) = x - y$ nie je binárna operácia na $\mathbf{R}^+$, pretože rozdiel každých dvoch kladných reálnych čísel nemusí byť kladné reálne číslo, napr. $2 - 3 = -1 \notin \mathbf{R}^+$, pričom $2, 3 \in \mathbf{R}^+$.

Funkcia $k(x) = x+3$ je unárna operácia na $\mathbf{R}$, pretože pre každé $x \in \mathbf{R}$ platí $x+3 \in \mathbf{R}$. ■

POZNÁMKA 1.1. Nech $\Box : A^2 \to A$ je binárna operácia na množine A. Potom obraz dvojice $(a, b) \in A^2$ namiesto $\Box(a, b)$ často označujeme symbolom $a \Box b$.

DEFINÍCIA 1.5. Binárna operácia $\Box : A^2 \to A$ sa nazýva
komutatívna, ak pre každé $a, b \in A$ platí $a \Box b = b \Box a$,
asociatívna, ak pre každé $a, b, c \in A$ platí $(a \Box b) \Box c = a \Box (b \Box c)$.
Prvok $e \in A$ sa nazýva ***neutrálny prvok*** binárnej operácie $\Box$ na množine A, ak pre všetky $x \in A$ platí $x \Box e = e \Box x = x$.

POZNÁMKA 1.2. Binárnu operáciu, ktorú označujeme $\cdot$, obvykle nazývame ***súčin*** a jej neutrálny prvok (pokiaľ existuje) nazývame ***jednotkový prvok***.
Binárnu operáciu, ktorú označujeme $+$, obvykle nazývame ***súčet*** a jej neutrálny prvok (pokiaľ existuje) nazývame ***nulový prvok***.

PRÍKLAD 1.5. Binárna operácia $\circ : \mathbf{R}^2 \to \mathbf{R}, x \circ y = x^3 y^3$ je komutatívna, lebo pre všetky $x, y \in \mathbf{R}$

$$x \circ y = x^3 y^3 = y^3 x^3 = y \circ x,$$

ale nie je asociatívna, lebo napr. pre $x = \sqrt[3]{2}, y = 1, z = \sqrt[9]{2}$

$$(x \circ y) \circ z = (\sqrt[3]{2} \circ 1) \circ \sqrt[9]{2} = (2 \cdot 1) \circ \sqrt[9]{2} = 8\sqrt[3]{2}$$

$$x \circ (y \circ z) = \sqrt[3]{2} \circ (1 \circ \sqrt[9]{2}) = \sqrt[3]{2} \circ (1 \cdot \sqrt[3]{2}) = 2 \cdot 2 = 4$$

Pre neutrálny prvok $e \in \mathbf{R}$ by malo platiť $x \circ e = x$, teda $x^3 e^3 = x$ a po úprave (za predpokladu, že $x \neq 0$) $e = \sqrt[3]{x^{-2}}$. Prvok e musí byť pre všetky x rovnaký, čo v našom prípade nenastáva. To znamená, že táto operácia nemá neutrálny prvok. ■

DEFINÍCIA 1.6. Nech $\Box, \triangle$ sú binárne operácie na množine A. Hovoríme, že ***binárna operácia*** $\Box$ ***je distributívna vzhľadom k binárnej operácii*** $\triangle$, ak pre všetky $x, y, z \in A$

$$x \Box (y \triangle z) = (x \Box y) \triangle (x \Box z),$$
$$(y \triangle z) \Box x = (y \Box x) \triangle (z \Box x).$$

PRÍKLAD 1.6. Na množine $\mathbf{N}^+$ uvažujme binárne operácie:
$+ : (\mathbf{N}^+)^2 \to \mathbf{N}^+$ – štandardné sčitovanie,
$\Box : (\mathbf{N}^+)^2 \to \mathbf{N}^+, x \Box y = x$,

$\triangle : (\mathbf{N}^+)^2 \to \mathbf{N}^+, x \triangle y = y.$
Binárna operácia $\square$ je distributívna vzhľadom k operácii $\triangle$, lebo pre všetky $x, y, z \in \mathbf{N}^+$

$$\begin{aligned} x \square (y \triangle z) &= x \square z = x, \\ (x \square y) \triangle (x \square z) &= x \triangle x = x = x \square (y \triangle z), \end{aligned}$$

$$\begin{aligned} (y \triangle z) \square x &= z \square x = z, \\ (y \square x) \triangle (z \square x) &= y \triangle z = z = (y \triangle z) \square x. \end{aligned}$$

Analogicky sa dá ukázať, že operácia $\triangle$ je distributívna vzhľadom k operácii $\square$. Naproti tomu operácie $\square$, $\triangle$ nie sú distributívne vzhľadom k operácii $+$, čo vyplýva z faktu

$$\begin{aligned} 1 \square (2+3) &= 1 \square 5 = 1, \\ (1 \square 2) + (1 \square 3) &= 1 + 1 = 2 \neq 1 \square (2+3), \end{aligned}$$

$$\begin{aligned} (2+3) \triangle 1 &= 5 \triangle 1 = 1 \\ (2 \triangle 1) + (3 \triangle 1) &= 1 + 1 = 2 \neq (2+3) \triangle 1. \end{aligned}$$

■

2. Jazyk nad abecedou

Jedným z prostriedkov výmeny informácií medzi ľuďmi je ľudská reč. Jej významové jednotky sú vety. Vety sa skladajú zo slov a slová z písmen. V tejto časti zovšeobecníme tieto pojmy z prirodzených jazykov.

Definícia 1.7. Neprázdnu množinu X nazývame ***abecedou***. Prvky množiny X nazývame ***písmenami*** (abecedy X). Konečnú postupnosť písmen abecedy X nazývame ***slovom*** nad X. Počet písmen v slove w nad abecedou X nazývame ***dĺžkou slova*** w a označujeme ju $|w|$. Množinu všetkých slov nad abecedou X označujeme X^+.
Nech $\nabla \notin X^+$. Symbol ∇ nazývame ***prázdnym slovom*** nad X a priraďujeme mu dĺžku nula. Množinu všetkých slov nad abecedou X spolu s prázdnym slovom označujeme X^*.

Príklad 1.7. Nech $X = \{a, b, c, d, e\}$. X je abeceda, jej písmenami sú a, b, c, d, e. Slovami sú napr. $aa, baba, cda, e, dedaeccb$. Slovo aa má dĺžku 2, slovo $baba$ má dĺžku 4 ($|baba| = 4$), $|cda| = 3$, $|e| = 1$, $|dedaeccb| = 8$. ■

Príklad 1.8. Nech $X = \{0, 1\}$. Napíšeme postupne všetky slová nad X dĺžok 1, 2, 3.
Riešenie.
Dĺžku 1 majú dve slová: 0, 1.
Dĺžku 2 majú štyri slová: 00, 01, 10, 11.
Dĺžku 3 má osem slov: 000, 001, 010, 100, 110, 101, 011, 111. ■

Poznamenajme, že všetkých slov s dĺžkou n nad dvojprvkovou abecedou je 2^n.

Definícia 1.8. Binárnu operáciu $\bullet$ na množine X^* definovanú:

$$\begin{aligned} x_1x_2\dots x_n \bullet y_1y_2\dots y_m &= x_1x_2\dots x_ny_1y_2\dots y_m \\ x_1x_2\dots x_n \bullet \nabla &= \nabla \bullet x_1x_2\dots x_n = x_1x_2\dots x_n \\ \nabla \bullet \nabla &= \nabla \end{aligned}$$

kde $x_1x_2\dots x_n \in X^+, y_1y_2\dots y_m \in X^+$ a ∇ je prázdne slovo, nazývame ***zreťazením slov*** z X^*. Množinu X^+ spolu s binárnou operáciou $\bullet$ zreťazenia nazývame ***voľná pologrupa nad množinou*** X (množina X^* spolu s operáciou zreťazenia slov sa nazýva ***voľná pologrupa s jednotkou*** alebo tiež ***monoid***). Každú podmnožinu množiny X^* nazývame ***jazykom*** nad abecedou X.

Poznamenajme, že tak ako pri násobení reálnych čísel namiesto $x.y$ píšeme xy, tak aj pre binárnu operáciu $\bullet$ namiesto $x \bullet y$ píšeme xy. Potom môžeme definíciu zreťazenia slov z X^* prepísať takto:

$$\begin{aligned}(x_1x_2\dots x_n)\,(y_1y_2\dots y_m) &= x_1x_2\dots x_ny_1y_2\dots y_m \\ x_1x_2\dots x_n\nabla &= \nabla x_1x_2\dots x_n = x_1x_2\dots x_n \\ \nabla\nabla &= \nabla\end{aligned}$$

Príklad 1.9. Slovo 01100 nad abecedou $X = \{0,1\}$ je zreťazením slov 0 a 1100, tiež zreťazením slov 01 a 100, ale nie je zreťazením slov 000 a 11. ■

Príklad 1.10. Ak $X = \{0,1\}$, tak zreťazením slov $1101, \nabla$ je slovo $1101\nabla = 1101$. Slovo 1101 je zreťazením slov $11, 01$, ako aj zreťazením slov $1, \nabla, 101$, pretože $(1\nabla)\,101 = 1101$. ■

Veta 1.1. Pre binárnu operáciu zreťazenia slov na množine X^* platí asociatívny zákon

$$(xy)\,z = x\,(yz)\,.$$

Dôkaz. Ak $x, y, z \in X^*$, tak podľa definície zreťazenia pre slová xy, z, x, yz platí $(xy)\,z = xyz$, $x\,(yz) = xyz$, teda $(xy)\,z = x\,(yz)$. □

Tak, ako je obvyklé v prípade platnosti asociatívneho zákona, budeme označovať $xyz = (xy)\,z = x\,(yz)$.

Príklad 1.11. Nech $X = \{0,1,2\}$. Potom zreťazením slov $01, 210, \nabla, 22$ je slovo $01\,(210\,(\nabla 22)) = 01\,(21022) = 0121022$. Slovo 120021 je zreťazením slov 12, 00, ∇, 21, ale aj zreťazením slov 12∇, $\nabla\nabla$, 0021. ■

3. Výroková logika

3.1. Výroky.

Mnohé vety, ktoré používame v prirodzených jazykoch (slovenčina, angličtina, atď.) alebo v jazykoch vedných odborov sú pravdivé, mnohé sú nepravdivé (i keď o ich pravdivosti či nepravdivosti nemusíme vedieť práve rozhodnúť) a sú vety, o pravdivosti ktorých nemá zmysel vôbec uvažovať (napr. „Neotravujte ma s matematikou!“). V tejto časti sa budeme zaoberať vetami, o pravdivosti ktorých má zmysel uvažovať.

Definícia 1.9. Vety, ktoré sú pravdivé a vety, ktoré sú nepravdivé, majú ***pravdivostnú hodnotu***. Vety, ktoré majú pravdivostnú hodnotu a tá sa nemení, sa nazývajú ***výroky***. Pravdivostná hodnota pravdivého výroku je 1, pravdivostná hodnota nepravdivého výroku je 0.

Na označenie pravdivostnej hodnoty výroku A budeme používať symbol $\mathrm{ph}(A)$.

Príklad 1.12. V teórii prirodzených čísel je veta „Súčet čísla dva a čísla tri je číslo päť.“ pravdivým výrokom. „Tri je párne číslo.“ je nepravdivý výrok. „Číslo x je nepárne číslo.“ nie je výrok, lebo táto veta nie je ani pravdivá, ani nepravdivá. Pravdivou alebo nepravdivou sa stane až po dosadení konkrétneho čísla za x a to už bude iná veta. Veta „Číslo tri je nepárne číslo.“ je pravdivý výrok. ■

Výroky môžeme spájať pomocou špeciálnych slovných spojení a vytvárať tak zložitejšie výroky, hovoríme im ***zložené výroky***.

DEFINÍCIA 1.10. Ak A, B sú výroky, tak
negáciou výroku A nazývame výrok „Nie je pravda, že A“ a označujeme ho $\overline{A}$ alebo $\neg A$,
konjunkciou výrokov A, B nazývame výrok „A a B“ a označujeme ho $A \wedge B$,
disjunkciou výrokov A, B nazývame výrok „A alebo B“ a označujeme ho $A \vee B$,
implikáciou výrokov A, B nazývame výrok „Ak A, potom B“ a označujeme ho $A \Rightarrow B$,
ekvivalenciou výrokov A, B nazývame výrok „A vtedy a len vtedy, keď B“ a označujeme ho $A \Leftrightarrow B$.

Okrem uvedených slovných spojení sa pri vytváraní zložených výrokov používajú aj iné. Napríklad, ak A je výrok „Číslo 2 je väčšie ako 3.“, tak jeho negáciu $\overline{A}$ môžeme vyjadriť: „Nie je pravda, že číslo 2 je väčšie ako 3.“ alebo „Číslo 2 nie je väčšie ako 3.“ alebo „Číslo 2 je menšie alebo sa rovná číslu 3.“ Tieto tri výroky samozrejme považujeme za rovnaké.
Konjunkciu dvoch výrokov A, B môžeme vyjadriť aj „A a súčasne B.“, ich implikáciu zas „Ak A, B.“ alebo „Ak A, tak B.“.

Aký je súvis medzi pravdivostnými hodnotami výrokov A, B a pravdivostnými hodnotami výrokov $\overline{A}$, $A \wedge B$, $A \vee B$, $A \Rightarrow B$, $A \Leftrightarrow B$? Pri ich určovaní sa vychádzalo zo zaužívaných spôsobov hodnotenia myslenia. Napríklad, ak výrok A je pravdivý, tak na otázku „Je pravdivý výrok A alebo B?“ odpovieme „Áno, je pravdivý.“. Teda $\mathrm{ph}(A \vee B) = 1$, keď $\mathrm{ph}(A) = 1$. Pravdivostné hodnoty negácie, konjunkcie, disjunkcie, implikácie a ekvivalencie uvádzame v tab. 1.

TABUĽKA 1. Pravdivostné hodnoty zložených výrokov

A	$\overline{A}$
0	1
1	0

A	B	$A \wedge B$
0	0	0
0	1	0
1	0	0
1	1	1

A	B	$A \vee B$
0	0	0
0	1	1
1	0	1
1	1	1

A	B	$A \Rightarrow B$
0	0	1
0	1	1
1	0	0
1	1	1

A	B	$A \Leftrightarrow B$
0	0	1
0	1	0
1	0	0
1	1	1

DEFINÍCIA 1.11. Tieto tabuľky sa nazývajú ***pravdivostné tabuľky*** postupne negácie, konjunkcie, disjunkcie, implikácie, ekvivalencie.

Znaky negácie, konjunkcie, disjunkcie, implikácie, ekvivalencie, t.j. znaky $^{-}, \wedge, \vee, \Rightarrow, \Leftrightarrow$ a im zodpovedajúce slovné spojenia nazývame ***logické spojky*** alebo ***funktory***.

Výroky A, B sa nazývajú ***ekvivalentné***, ak $ph(A \Leftrightarrow B) = 1$.

Uvažujme teraz o výroku „Ak zajtra bude u nás pekne, tak pôjdeme k jazeru alebo pôjdeme do hory.“ Skladá sa z výrokov „Zajtra bude u nás pekne.“, „Pôjdeme k jazeru alebo pôjdeme do hory.“. Keby sme prvý výrok označili A, druhý B, tak pôvodný výrok je $A \Rightarrow B$. Ale druhý výrok sa zasa skladá s dvoch výrokov, a to „Pôjdeme k jazeru.“ a „Pôjdeme do hory.“. Keď ich označíme postupne C, D, tak pôvodný výrok môžeme zapísať v tvare $A \Rightarrow (C \vee D)$. Výroky A, C, D sa už ďalej nedajú rozkladať na jednoduchšie výroky.

DEFINÍCIA 1.12. Výroky, ktoré sa nedajú rozložiť na jednoduchšie výroky, alebo výroky ktoré vždy vystupujú ako celok, sa nazývajú ***atomické*** (alebo ***atomárne***).

PRÍKLAD 1.13. „Číslo 12 delí číslo 24 a číslo 4 číslo 12, a preto číslo 4 delí číslo 24“ je výrok v teórii prirodzených čísel. Skladá sa z atomických výrokov „Číslo 12 delí číslo 24“, „Číslo 4 delí číslo 12“, „Číslo 4 delí číslo 24“. Ak by sme si označili tieto atomické výroky postupne A, B, C, môžeme pôvodný výrok zapísať $(A \wedge B) \Rightarrow C$. ∎

3.2. Kvantifikované výroky.

Už sme spomenuli, že veta (výraz) „Číslo x je nepárne číslo." nie je výrok. Ale ak za x dosadíme konkrétne celé číslo, dostaneme výrok.
Výraz, ktorý obsahuje jednu alebo viac premenných, z ktorého po dosadení prípustných hodnôt za premenné vznikne výrok, sa nazýva ***výroková forma***. Ak $A(x_1,\ldots,x_k)$ je výroková forma k premenných $x_1,\ldots,x_k$, tak množina k-tic prípustných hodnôt sa nazýva ***definičný obor výrokovej formy*** $A(x_1,\ldots,x_k)$. Množina všetkých $(q_1,\ldots,q_k)$ z definičného oboru D výrokovej formy $A(x_1,\ldots,x_k)$, pre ktoré je výrok $A(q_1,\ldots,q_k)$ pravdivý, sa nazýva ***obor pravdivosti výrokovej formy*** $A(x_1,\ldots,x_k)$ a označuje sa $\{(x_1,\ldots,x_k)\in D; A(x_1,\ldots,x_k)\}$.

Príklad 1.14. $x^2+y^2\le 0$ je výroková forma dvoch premenných x,y. Jej definičný obor je $\mathbf{R}^2$. Do oboru pravdivosti patrí len dvojica $(0,0)$, teda
$\{(x,y)\in\mathbf{R}^2; x^2+y^2\le 0\}=\{(0,0)\}$. ■

Výrokové formy môžeme spájať pomocou logických spojok, a tak vytvárať nové výrokové formy.

Príklad 1.15. Negáciou výrokovej formy $x\ge 2$ s definičným oborom $\mathbf{R}$ je výroková forma $x<2$ s rovnakým definičným oborom.
Konjunkciou výrokových foriem $x^2-x-6=0$, $x>0$ je výroková forma
$x^2-x-6=0\wedge x>0$. ■

Z výrokovej formy sa dá vytvoriť výrok aj iným spôsobom ako je dosadenie hodnôt za premenné. Môžeme vytvoriť výrok, v ktorom sa hovorí o počte prvkov, ktoré keď dosadíme za premenné do výrokovej formy, dostaneme pravdivý výrok. Takéto výroky nazývame ***kvantifikované výroky***. Nech $A(x)$ je výroková forma jednej premennej x s definičným oborom D. Potom môžeme vytvoriť takéto kvantifikované výroky:

Existuje (existuje aspoň jedno) x, že $A(x)$.
Existujú aspoň štyri x, že $A(x)$.
Existuje najviac päť x, že $A(x)$.
Existujú práve tri x, že $A(x)$.
Pre všetky x (platí) $A(x)$.

Výrazy vytlačené polotučnou kurzívou sa nazývajú ***kvantifikátory***. V matematike sa najčastejšie používajú prvý a posledný. Ten prvý sa nazýva ***existenčný*** (alebo ***malý***) ***kvantifikátor*** a posledný ***všeobecný*** (alebo ***veľký***) ***kvantifikátor***.

Existenčný kvantifikátor sa označuje symbolom $\exists$ (otočené písmeno E). Kvantifikovaný výrok

„Existuje x, že $A(x)$."

môžeme stručne zapísať

$$\exists x\in D\ A(x) \qquad \text{alebo} \qquad \mathop{\exists}_{x\in D} A(x)$$

Ak je známy obor premennej x, môžeme použiť aj zápis

$$\exists x\ A(x) \qquad \text{alebo} \qquad \mathop{\exists}_{x} A(x).$$

Všeobecný kvantifikátor sa označuje symbolom $\forall$ (otočené písmeno A). Kvantifikovaný výrok

„Pre všetky x $A(x)$."

stručne zapisujeme

$$\forall x \in D\ A(x) \qquad \text{alebo} \qquad \underset{x\in D}{\forall} A(x)$$

alebo, ak je známy obor premennej x,

$$\forall x\ A(x) \qquad \text{alebo} \qquad \underset{x}{\forall} A(x).$$

Príklad 1.16. Rozhodnite o pravdivosti výrokov

(1) $\exists n \in \mathbf{Z}\ n^2 = 9$,
(2) $\forall n \in \mathbf{Z}\ n^2 = 9$,
(3) $\exists n \in \mathbf{Z}\ n^2 = 2$,
(4) $\forall n \in \mathbf{Z}\ n^2 = 2$.

Riešenie. Prvý výrok je pravdivý, lebo pre $n = -3$ je $(-3)^2 = 9$.

Druhý výrok je nepravdivý, lebo rovnosť $n^2 = 9$ neplatí pre každé celé číslo n, napr. pre $n = 0$ je $n^2 = 0^2 = 0 \neq 9$.

Tretí aj štvrtý výrok sú nepravdivé, lebo len pre dve reálne čísla platí, že ich druhá mocnina je 2. Sú to čísla $\sqrt{2}$ a $-\sqrt{2}$, no ani jedno z nich nie je celé číslo. ■

Kvantifikované výroky môžeme vytvoriť aj z výrokových foriem dvoch, troch a viac premenných. Napr. výraz

$$\exists x \in \mathbf{Z}\ \forall y \in \mathbf{R}\ x + y^2 \leq -4$$

je výrok, ktorý čítame:

„Existuje celé číslo x tak, že pre všetky reálne čísla y je $x + y^2 \leq -4$.“

Výrok

$$\forall y \in \mathbf{R}\ \exists x \in \mathbf{Z}\ \forall z \in \mathbf{R}\ x + y^2 - z \leq -4$$

čítame

„Pre každé reálne číslo y existuje celé číslo x tak, že pre všetky reálne čísla z je $x + y^2 - z \leq -4$.“

Nech $A(x, y)$ je výroková forma definovaná pre $x \in D, y \in E$. Výraz $\forall x \in D\ A(x, y)$ resp. $\exists x \in D\ A(x, y)$ nie je výrok. Ak však za premennú y dosadíme konkrétny prvok množiny E, dostaneme výrok. Uvedené výrazy sú teda výrokové formy premennej y.

Príklad 1.17. Rozhodnite o pravdivosti výrokov

(1) $\forall x \in \mathbf{R}\ \exists y \in \mathbf{R}\ x - y = 1$,
(2) $\exists y \in \mathbf{R}\ \forall x \in \mathbf{R}\ x - y = 1$.

Riešenie.

(1) Nech x je ľubovoľné reálne číslo. K splneniu rovnosti $x - y = 1$ stačí zvoliť $y = x - 1$. Ku každému $x \in \mathbf{R}$ sme našli (teda existuje) $y \in \mathbf{R}$ $(y = x - 1)$ tak, že platí $x - y = 1$. To znamená, že výrok je pravdivý.

(2) Malo by existovať y také, že pre všetky reálne čísla x je $x - y = 1$. Označme to číslo y_0. Pokiaľ existuje, má preň platiť $x - y_0 = 1$ pre všetky $x \in \mathbf{R}$. Lenže pre $x = y_0$ platí $x - y_0 = y_0 - y_0 = 0 \neq 1$. Znamená to, že také y_0 neexistuje, a teda výrok je nepravdivý. ■

Poznámka 1.3. Výroky z predchádzajúceho príkladu sa od seba líšia len poradím kvantifikátorov. Aby bol prvý výrok pravdivý, je potrebné, aby pre každé x existovalo y tak, že $x - y = 1$. Uvedomme si, že pre rôzne čísla x aj zodpovedajúce čísla y môžu byť rôzne. Naproti tomu, aby bol pravdivý druhý výrok, je potrebné, aby existovalo jedno y tak, že pre všetky x je $x - y = 1$. Teda to y je pre všetky x stále to isté. Vidíme, že zmenou poradia malého a veľkého kvantifikátora dostávame odlišné výroky, ktoré nemusia byť ekvivalentné.

Poradie za sebou idúcich kvantifikátorov rovnakého typu (oba sú všeobecné alebo oba sú existenčné) môžeme meniť a výrok sa pritom nezmení. Preto aj výrazy $\forall x \in D\, \forall y \in D$ resp. $\exists x \in D\, \exists y \in D$ skracujeme na $\forall x, y \in D$ resp. $\exists x, y \in D$.

Venujme sa teraz tomu, ako tvoriť negácie kvantifikovaných výrokov. Nech $A(x)$ je výroková forma definovaná na množine D. Negáciou výroku $\exists x \in D\ A(x)$ je výrok „Nie je pravda, že existuje $x \in D$, pre ktoré platí $A(x)$.“ To je to isté ako „Neexistuje $x \in D$, pre ktoré platí $A(x)$.“ a tiež ako „Pre všetky $x \in D$ platí negácia $A(x)$,“, čo môžeme zapísať takto: $\forall x \in D\ \overline{A(x)}$. Podobnými úvahami by sme utvorili aj negácie ďalších typov kvantifikovaných výrokov. Uvádzame ich v tabuľke 2. Číslo k je tu jedno konkrétne ale ináč ľubovoľné prirodzené číslo väčšie ako 1.

Tabuľka 2. Negácie kvantifikovaných výrokov

Výrok	Negácia výroku
$\exists x \in D\ A(x)$	$\forall x \in D\ \overline{A(x)}$
Existuje aspoň k prvkov x, že $A(x)$.	Existuje najviac $k-1$ prvkov x, že $A(x)$.
Existuje najviac k prvkov x, že $A(x)$.	Existuje aspoň $k+1$ prvkov x, že $A(x)$.
Existuje práve k prvkov x, že $A(x)$.	Existuje najviac $k-1$ alebo aspoň $k+1$ prvkov x, že $A(x)$.
$\forall x \in D\ A(x)$	$\exists x \in D\ \overline{A(x)}$

Negáciu výroku, ktorý obsahuje niekoľko malých a veľkých kvantifikátorov získame tak, že každý malý kvantifikátor zmeníme na veľký, veľký kvantifikátor zmeníme na malý a výrokovú formu negujeme.

Príklad 1.18. Napíšte negácie výrokov

(1) $\exists y \in \mathbf{R}\, \forall x \in \mathbf{R}\ xy \leq y$,
(2) $\forall x \in \mathbf{N}^+\, \exists y \in \mathbf{N}^+\, \exists z \in \mathbf{N}^+\ x^2 + y^2 = z^2$.

Riešenie.

(1) $\forall y \in \mathbf{R}\, \exists x \in \mathbf{R}\ xy > y$,
(2) $\exists x \in \mathbf{N}^+\, \forall y \in \mathbf{N}^+\, \forall z \in \mathbf{N}^+\ x^2 + y^2 \neq z^2$.

■

3.3. Výrokové formuly.

Výroková logika pri skúmaní pravdivostných hodnôt zložených výrokov si nevšíma obsah jednotlivých atomických výrokov ale ich pravdivostnú hodnotu a tiež tvar (štruktúru) zloženého výroku. Preto je vhodné zaviesť ***výrokové premenné*** – premenné, za ktoré je možné dosadzovať výroky – a pomocou nich, logických spojok a zátvoriek vytvárať výrazy, ktoré majú vlastnosť, že keď do nich za výrokové premenné dosadíme výroky, vzniknú z týchto výrazov výroky. Ako výrokové premenné budeme používať písmená p, q, r, s, t, poprípade tieto písmena s indexami napr. p_1, p_2, q_4. Výrazy, o ktorých sme teraz hovorili, nazývame výrokové formuly. Ich presná definícia je takáto:

DEFINÍCIA 1.13. ***Formula výrokového počtu*** (skrátene ***výroková formula*** alebo ***formula***) je každé slovo nad abecedou $\{^{-}, \wedge, \vee, \Rightarrow, \Leftrightarrow\ (,), p, q, r, s, t, p_1, q_1, r_1, s_1, t_1, \ldots\}$, ktoré vzniklo podľa pravidiel:

(1) Každá výroková premenná je formula.
(2) Ak a, b sú formuly, tak

$$\overline{a}, (a \wedge b), (a \vee b), (a \Rightarrow b), (a \Leftrightarrow b)$$

sú formuly.
(3) Žiadne iné slová nie sú formuly.

Formulu $\overline{a}$ nazývame ***negáciou formuly*** a, formulu $(a \wedge b)$ nazývame ***konjunkciou formúl*** a, b, formulu $(a \vee b)$ nazývame ***disjunkciou formúl*** a, b, formulu $(a \Rightarrow b)$ nazývame ***implikáciou formúl*** a, b a formulu $(a \Leftrightarrow b)$ ***ekvivalenciou formúl*** a, b.

PRÍKLAD 1.19. p, q, r sú výrokové premenné, a teda aj formuly, preto $(p \Rightarrow q), (p \vee r)$ sú tiež formuly. Keď použijeme opakovane postup z bodu 2 definície formuly, tak dostaneme aj tieto formuly: $(p \Rightarrow (q \vee r))$, $((p \wedge \overline{q}) \Rightarrow (r \Rightarrow (r \Rightarrow q)))$. ■

Keby sme v definícii formuly nepoužili zátvorky, tak by formuly z predchádzajúceho príkladu mali tvar $p \Rightarrow q$, $p \vee r$, ale už pri posledných dvoch formulách $p \Rightarrow q \vee r$, $p \wedge \overline{q} \Rightarrow r \Rightarrow r \Rightarrow q$ by sme po dosadení výrokov za výrokové premenné mohli dostať vetu, ktorá nie je po obsahovej stránke jednoznačná a teda nie je výrok. Napríklad veta „Keď Vlado hovorí, vtedy Jano mlčí alebo Ivan plače“ vznikne dosadením do $p \Rightarrow (q \vee r)$, ale tiež do $(p \Rightarrow q) \vee r$. V hovorovej reči sa obvykle používajú nepísané dohody a väčšina by túto vetu priradila k $p \Rightarrow (q \vee r)$ a nie k $(p \Rightarrow q) \vee r$. V matematike, ale aj iných odboroch, je takáto nejednoznačnosť neprípustná.

Situácia, aby veľa zátvoriek nebolo na ujmu prehľadnosti, ale pritom nevznikala nejednoznačnosť, sa rieši buď

- dohodou, že zátvorky vynechávame, keď nemôže prísť k nejednoznačnosti

alebo

- dohodou o vynechávaní zátvoriek a priorite logických spojok.

Dohoda o vonkajších zátvorkách: V samostatne stojacich formulách budeme vonkajšie zátvorky vynechávať.

Princíp priority logických spojok: Ak logické spojky usporiadame do postupnosti $^{-}, \wedge, \vee, \Rightarrow, \Leftrightarrow$, tak každá logická spojka, stojaca vľavo od uvažovanej, viaže silnejšie. Nech p, q, r sú výrokové premenné. Logická spojka L ***viaže silnejšie*** ako logická spojka K znamená, že $p\,L\,q\,K\,r$ je formula $(p\,L\,q)\ K\,r$.

Poznamenajme, že tento princíp poznáme z reálnych čísel, kde „krát“ viaže silnejšie ako „plus“, čiže napr. $2.3 + 1$ znamená $(2.3) + 1$ a nie $2.(3 + 1)$.

PRÍKLAD 1.20.
$p \wedge q \vee \overline{r}$ znamená formulu $(p \wedge q) \vee \overline{r}$,
$p \Rightarrow q \vee r \Leftrightarrow p$ znamená formulu $(p \Rightarrow (q \vee r)) \Leftrightarrow p$,
$p \vee r \Leftrightarrow p \wedge q \vee r$ znamená formulu $(p \vee r) \Leftrightarrow ((p \wedge q) \vee r)$. ■

DEFINÍCIA 1.14. Postupnosť formúl $a_1, a_2, \ldots, a_n$ sa nazýva ***vytvárajúca postupnosť formuly*** a, ak $a = a_n$ a pre každé $k \in \{1, 2, \ldots, n\}$ je a_k buď výroková premenná alebo existujú $i, j \in \{1, 2, \ldots, k-1\}$ tak, že a_k je jedna z formúl $\overline{a}_i, a_i \wedge a_j, a_i \vee a_j$, $a_i \Rightarrow a_j$, $a_i \Leftrightarrow a_j$.

PRÍKLAD 1.21. Vytvárajúcou postupnosťou formuly

$$a = ((p \vee \overline{q}) \wedge \overline{\overline{p} \vee r}) \Rightarrow (p \Rightarrow (q \Rightarrow r))$$

je postupnosť p, q, r, $\overline{p}$, $\overline{q}$, $p \vee \overline{q}$, $\overline{p} \vee r$, $\overline{\overline{p} \vee r}$, $q \Rightarrow r$, $(p \vee \overline{q}) \wedge \overline{\overline{p} \vee r}$, $p \Rightarrow (q \Rightarrow r)$, $((p \vee \overline{q}) \wedge \overline{\overline{p} \vee r}) \Rightarrow (p \Rightarrow (q \Rightarrow r))$. ■

DEFINÍCIA 1.15. Ak formula b obsahuje výrokové premenné $p_1, p_2, \dots, p_n$ a žiadne iné, tak formulu b označíme $b\,(p_1, p_2, \dots, p_n)$ a hovoríme, že b je formula n premenných $p_1, p_2, \dots, p_n$.

DEFINÍCIA 1.16. Výrok, ktorý vznikne z formuly, keď za všetky jej výrokové premenné dosadíme výroky, sa nazýva ***interpretácia formuly***.

POZNÁMKA 1.4. Ak $A_1, \dots, A_n$ sú výroky a $b\,(p_1, p_2, \dots, p_n)$ je formula, tak dosadením výroku A_1 za premennú p_1, A_2 za p_2 až A_n za p_n dostaneme výrok $b\,(A_1, A_2, \dots, A_n)$, ktorý je interpretáciou formuly b.

PRÍKLAD 1.22. Vezmime formulu $r \wedge s \Rightarrow t$. Jej interpretácie v teórii prirodzených čísel sú napríklad:
„Keď 2 delí 12 aj 3 delí 12, tak aj 6 delí 12.“
„Ak číslo 3 je väčšie ako číslo 2 a číslo 7 je väčšie ako číslo 32, tak číslo 3 je väčšie ako číslo 32.“
„Ak 5 je menšie ako 6 a 15 je väčšie ako 13, tak 5 je menšie ako 3.“

Formula $r \wedge s \Rightarrow t$ je formulou troch premenných r, s, t, môžeme ju teda označiť napr. $a\,(r, s, t)$. ■

Výroková formula nie je výrok, nemá teda pravdivostnú hodnotu. Každej výrokovej formule $a(p_1, p_2, \dots, p_n)$ však môžeme priradiť funkciu, ktorá vyjadruje závislosť pravdivostných hodnôt interpretácií $a(A_1, A_2, \dots, A_n)$ tejto formuly od pravdivostných hodnôt výrokov A_1, A_2 až A_n dosadených za jednotlivé výrokové premenné.

DEFINÍCIA 1.17. Nech $a(p_1, p_2, \dots, p_n)$ je formula n premenných. Funkcia

$$\mathrm{ph}_a : \{0,1\}^n \to \{0,1\}, \mathrm{ph}_a(x_1, x_2, \dots, x_n) = \mathrm{ph}(a(A_1, A_2, \dots, A_n)),$$

kde $A_1, A_2, \dots, A_n$ sú výroky, pre ktoré $\mathrm{ph}(A_1) = x_1$, $\mathrm{ph}(A_2) = x_2$, $\dots$, $\mathrm{ph}(A_n) = x_n$, sa nazýva ***pravdivostné ohodnotenie formuly*** a.

Všetkých usporiadaných n-tíc prvkov 0, 1 je 2^n. Označme ich $\beta_1 = (0, 0, \dots, 0, 0)$, $\beta_2 = (0, 0, \dots, 0, 1), \beta_3 = (0, 0, \dots, 1, 0), \dots, \beta_{2^n} = (1, 1, \dots, 1)$. Potom môžeme pravdivostné ohodnotenie formuly $a(p_1, p_2, \dots, p_n)$ zapísať v tvare tabuľky (pozri tab. 3), ktorú nazývame ***pravdivostnou tabuľkou formuly*** $a(p_1, p_2, \dots, p_n)$.

TABUĽKA 3. Pravdivostná tabuľka formuly a

$(p_1, p_2, \dots, p_n)$	$a\,(p_1, p_2, \dots, p_n)$
β_1	$\mathrm{ph}_a\,(\beta_1)$
β_2	$\mathrm{ph}_a\,(\beta_2)$
$\vdots$	$\vdots$
β_{2^n}	$\mathrm{ph}_a\,(\beta_{2^n})$

PRÍKLAD 1.23. Pravdivostné tabuľky formúl

$$r \wedge \overline{s} \Leftrightarrow s \vee \overline{r}, \ (r \wedge s) \vee t \Rightarrow (s \Rightarrow t)$$

sú v tab. 4.

TABUĽKA 4

(r,s)	$r \wedge \overline{s} \Leftrightarrow s \vee \overline{r}$
(0,0)	0
(0,1)	0
(1,0)	0
(1,1)	0

(r,s,t)	$(r \wedge s) \vee t \Rightarrow (s \Rightarrow t)$
(0,0,0)	1
(0,0,1)	1
(0,1,0)	1
(0,1,1)	1
(1,0,0)	1
(1,0,1)	1
(1,1,0)	0
(1,1,1)	1

Pri ich tvorbe je vhodné postupovať tak, že do hlavičky tabuľky zapíšeme vytvárajúcu postupnosť danej formuly (pozri tab. 5 a 6), pod premenné dáme všetky možné zoskupenia pravdivostných hodnôt a postupne po stĺpcoch vypĺňujeme pravdivostné ohodnotenia jednotlivých formúl vytvárajúcej postupnosti na základe pravdivostných tabuliek zložených výrokov (tab. 1). ■

TABUĽKA 5

r,s	$\overline{s}$	$r \wedge \overline{s}$	$\overline{r}$	$s \vee \overline{r}$	$(r \wedge \overline{s}) \Leftrightarrow (s \vee \overline{r})$
0,0	1	0	1	1	0
0,1	0	0	1	1	0
1,0	1	1	0	0	0
1,1	0	0	0	1	0

TABUĽKA 6

r,s,t	$r \wedge s$	$(r \wedge s) \vee t$	$s \Rightarrow t$	$(r \wedge s) \vee t \Rightarrow (s \Rightarrow t)$
0,0,0	0	0	1	1
0,0,1	0	1	1	1
0,1,0	0	0	0	1
0,1,1	0	1	1	1
1,0,0	0	0	1	1
1,0,1	0	1	1	1
1,1,0	1	1	0	0
1,1,1	1	1	1	1

DEFINÍCIA 1.18. Formula sa nazýva
tautológia, ak jej pravdivostným ohodnotením je konštantná funkcia s hodnotou 1;
kontradikcia, ak jej pravdivostným ohodnotením je konštantná funkcia s hodnotou 0;
splniteľná, ak nie je kontradikciou.

PRÍKLAD 1.24.
Formula $p \vee \overline{p}$ je tautológia.
Formula $p \wedge \overline{p}$ je kontradikcia.
Formula $p \Rightarrow q$ je splniteľná.
Formula $(p \wedge (p \Rightarrow q)) \Rightarrow q$ je tautológia.
Výsledok vidíme z tab. 7. ■

TABUĽKA 7

p	$\overline{p}$	$p \vee \overline{p}$	$p \wedge \overline{p}$
0	1	1	0
1	0	1	0

p q	$p \Rightarrow q$	$p \wedge (p \Rightarrow q)$	$(p \wedge (p \Rightarrow q)) \Rightarrow q$
0 0	1	0	1
0 1	1	0	1
1 0	0	0	1
1 1	1	1	1

Tautológia $(p \wedge (p \Rightarrow q)) \Rightarrow q$ je základom pre správne usudzovanie. Ukazuje nám, ako z pravdivosti jedného výroku usúdiť, že je pravdivý druhý výrok. Z pravdivostnej tabuľky tejto formuly vidieť, že keď A, B sú výroky, pričom výroky A, $A \Rightarrow B$ sú pravdivé, potom je pravdivý aj výrok B. Na usúdenie pravdivosti výroku B nepostačuje len pravdivosť implikácie $A \Rightarrow B$. Vidieť to z prvého riadku pravdivostnej tabuľky formuly $p \Rightarrow q$. Výrok $A \Rightarrow B$ môže byť pravdivý a pritom výrok B je nepravdivý.

DEFINÍCIA 1.19. Dve formuly a, b s rovnakými premennými sa nazývajú ***tautologicky ekvivalentné***, ak majú rovnaké pravdivostné ohodnotenie. Tautologicky ekvivalentné formuly označujeme $a \sim b$.

PRÍKLAD 1.25. Dokážte, že formuly $\overline{p \wedge q}$ a $\overline{p} \vee \overline{q}$ sú tautologicky ekvivalentné.

Riešenie. Napíšeme pravdivostné tabuľky (tab. 8) týchto formúl. Obidve formuly majú dve premenné, a to p, q. Tabuľkami sú definované rovnaké funkcie, teda formuly sú tautologicky ekvivalentné. ■

TABUĽKA 8

p q	$\overline{p \wedge q}$
0 0	1
0 1	1
1 0	1
1 1	0

p q	$\overline{p} \vee \overline{q}$
0 0	1
0 1	1
1 0	1
1 1	0

PRÍKLAD 1.26. Zistite, či platí $(p \vee q) \vee r \sim p \vee (q \vee r)$.

Riešenie. Urobíme pomocnú tabuľku (tab. 9) a z nej už bude vidieť, keď si pozrieme prvý, predposledný a posledný stĺpec, či formuly majú rovnaké pravdivostné ohodnotenie. ■

TABUĽKA 9

p, q, r	$p \vee q$	$q \vee r$	$(p \vee q) \vee r$	$p \vee (q \vee r)$
0, 0, 0	0	0	0	0
0, 0, 1	0	1	1	1
0, 1, 0	1	1	1	1
0, 1, 1	1	1	1	1
1, 0, 0	1	0	1	1
1, 0, 1	1	1	1	1
1, 1, 0	1	1	1	1
1, 1, 1	1	1	1	1

V nasledujúcej vete uvedieme základné tautologické ekvivalencie.

VETA 1.2. Nech a, b, c sú formuly s rovnakými premennými. Potom platia tieto tautologické ekvivalencie:

E1.	$a \vee b \sim b \vee a,$	$a \wedge b \sim b \wedge a,$
E2.	$(a \vee b) \vee c \sim a \vee (b \vee c),$	$(a \wedge b) \wedge c \sim a \wedge (b \wedge c),$
E3.	$(a \vee b) \wedge c \sim (a \wedge c) \vee (b \wedge c),$	$(a \wedge b) \vee c \sim (a \vee c) \wedge (b \vee c),$
E4.	$\overline{a \vee b} \sim \overline{a} \wedge \overline{b},$	$\overline{a \wedge b} \sim \overline{a} \vee \overline{b},$
E5.	$a \vee a \sim a,$	$a \wedge a \sim a,$
E6.	$a \vee (b \wedge \overline{b}) \sim a,$	$a \wedge (b \vee \overline{b}) \sim a,$
E7.	$\overline{\overline{a}} \sim a,$	
E8.	$a \vee (a \wedge b) \sim a,$	$a \wedge (a \vee b) \sim a,$
E9.	$a \Rightarrow b \sim \overline{a} \vee b,$	$a \Rightarrow b \sim \overline{a \wedge \overline{b}},$
E10.	$a \Leftrightarrow b \sim (\overline{a} \vee b) \wedge (a \vee \overline{b}),$	$a \Leftrightarrow b \sim (a \wedge b) \vee (\overline{a} \wedge \overline{b}).$

Dôkaz tejto vety neurobíme, necháme na čitateľa, aby urobil a porovnal príslušné pravdivostné tabuľky. □

PRÍKLAD 1.27. Dokážte $p \Rightarrow q \sim \overline{p \wedge \overline{q}}$, druhú z ekvivalencií E9 z predchádzajúcej vety.

Riešenie. Uvedený vzťah môžeme dokázať buď pomocou pravdivostných tabuliek oboch formúl, alebo využitím tabuľky tautologických ekvivalencií. Použijeme tautologické ekvivalencie. Nad znak tautologickej ekvivalencie napíšeme ktoré pravidlo z tabuľky tautologických ekvivalencií sme použili. Z E9 sme použili prvú ekvivalenciu.

$$p \Rightarrow q \overset{E9}{\sim} \overline{p} \vee q \overset{E7}{\sim} \overline{\overline{\overline{p} \vee q}} \overset{E4}{\sim} \overline{p \wedge \overline{q}}$$

■

DEFINÍCIA 1.20. Hovoríme, že množina logických spojok S je ***úplný systém logických spojok*** (skrátene USLS), ak pre každú formulu a existuje formula b, ktorá obsahuje iba logické spojky z množiny S a platí $a \sim b$. Vtedy tiež hovoríme, že formula a ***sa dá vyjadriť pomocou*** S.

PRÍKLAD 1.28. Ukážte, že množiny $\{\vee, \wedge, ^-\}$, $\{^-, \Rightarrow\}$ sú úplné systémy logických spojok.

Riešenie. Pre každé formuly c, d platí

$$c \Rightarrow d \sim \overline{c} \vee d,$$
$$c \Leftrightarrow d \sim (\overline{c} \vee d) \wedge (c \vee \overline{d}).$$

Pomocou týchto ekvivalencií môžeme každú formulu a upraviť na ekvivalentnú formulu b, ktorá neobsahuje logické spojky $\Rightarrow, \Leftrightarrow$, teda obsahuje iba logické spojky z množiny $\{\vee, \wedge, ^-\}$. Tým sme dokázali, že $\{\vee, \wedge, ^-\}$ je USLS.

Aby sme dokázali, že $\{^-, \Rightarrow\}$ je USLS, stačí dokázať, že každá formula a, ktorá obsahuje logické spojky len z množiny $\{\vee, \wedge, ^-\}$ je ekvivalentná niektorej formule b, ktorá obsahuje logické spojky len z množiny $\{^-, \Rightarrow\}$. Pravdivosť tohto tvrdenia vyplýva z toho, že pre každé formuly c, d platí

$$c \vee d \sim \overline{c} \Rightarrow d,$$
$$c \wedge d \sim \overline{\overline{c \wedge d}} \sim \overline{\overline{c} \vee \overline{d}} \sim \overline{c \Rightarrow \overline{d}}.$$

■

Príklad 1.29. Vyjadrite formulu $p \Leftrightarrow q$ pomocou

a) $\{\vee, \wedge, ^-\}$,
b) $\{^-, \Rightarrow\}$.

Riešenie.

a) $p \Leftrightarrow q \sim (p \Rightarrow q) \wedge (q \Rightarrow p) \sim (\overline{p} \vee q) \wedge (p \vee \overline{q})$.

b) Použijeme $c \wedge d \sim \overline{\overline{c} \vee \overline{d}} \sim \overline{c \Rightarrow \overline{d}}$. Potom platí

$$p \Leftrightarrow q \sim (p \Rightarrow q) \wedge (q \Rightarrow p) \sim \overline{(p \Rightarrow q) \Rightarrow \overline{q \Rightarrow p}}.$$

■

4. Relácie

Definícia 1.21. ***Binárnou reláciou*** (stručne len ***reláciou***) ***na množine*** A nazývame ľubovoľnú podmnožinu karteziánskeho súčinu A^2.

Poznámka 1.5. Ak $\sigma \subset A^2$ je binárna relácia na množine A a $(x, y) \in \sigma$, tak hovoríme, že ***prvok*** x ***je v relácii*** σ ***s prvkom*** y a namiesto $(x, y) \in \sigma$ píšeme aj $x \, \sigma \, y$.

Definícia 1.22. Relácia σ na množine A sa nazýva
reflexívna, ak $\forall x \in A \ (x, x) \in \sigma$,
symetrická, ak $\forall x, y \in A \ (x, y) \in \sigma \Rightarrow (y, x) \in \sigma$,
antisymetrická, ak $\forall x, y \in A \ (x, y) \in \sigma \wedge (y, x) \in \sigma \Rightarrow x = y$,
tranzitívna, ak $\forall x, y, z \in A \ (x, y) \in \sigma \wedge (y, z) \in \sigma \Rightarrow (x, z) \in \sigma$.

Príklad 1.30. Zistite, či relácia $\sigma \subset \mathbf{R}^2, \sigma = \{(x, y)\,; x \leq y\}$ je reflexívna, symetrická, antisymetrická, tranzitívna.

Riešenie. Relácia σ je reflexívna, lebo pre každé $x \in \mathbf{R}$ platí $x \leq x$, teda $(x, x) \in \sigma$.
Relácia nie je symetrická, pretože napríklad $(2, 3) \in \sigma$, ale $(3, 2) \notin \sigma$.
Relácia je antisymetrická: Nech $(x, y) \in \sigma$, $(y, x) \in \sigma$, teda $x \leq y$, $y \leq x$. Z vlastností reálnych čísel vieme, že potom platí $x = y$.
Relácia je tranzitívna: Ak $(x, y) \in \sigma$ a zároveň $(y, z) \in \sigma$, teda $x \leq y$ a zároveň $y \leq z$, potom z vlastností reálnych čísel vyplýva $x \leq z$, teda $(x, z) \in \sigma$. ■

Definícia 1.23. Binárna relácia na množine A, ktorá je reflexívna, symetrická a tranzitívna, sa nazýva ***ekvivalencia*** alebo ***relácia ekvivalencie*** na množine A.

Príklad 1.31. Je relácia $\varrho = \{(x,y) \in \mathbf{Z}^2; 3 \mid (y-x)\}$ ekvivalenciou na množine $\mathbf{Z}$ všetkých celých čísel? ($a \mid b$ znamená: číslo b je deliteľné číslom a.)

Riešenie. $x - x = 0$, čo je číslo deliteľné tromi, preto $(x,x) \in \varrho$. Relácia ϱ je teda reflexívna.

Nech $(x,y) \in \varrho$. To znamená, že 3 delí číslo $y - x$. V takom prípade $y - x = 3k$, kde $k \in \mathbf{Z}$. Potom $x - y = -(y-x) = 3(-k)$, čo znamená, že $3 \mid (x-y)$ a teda $(y,x) \in \varrho$. Z toho vyplýva, že relácia ϱ je symetrická.

Nech $(x,y),(y,z) \in \varrho$. Existujú teda také celé čísla m,n, že $y - x = 3m, z - y = 3n$. Potom $z - x = z - x - y + y = (z-y) + (y-x) = 3m + 3n = 3(m+n)$. Číslo $z - x$ je deliteľné číslom 3, to znamená $(x,z) \in \varrho$ a teda relácia ϱ je tranzitívna.

Relácia ϱ má požadované vlastnosti, je preto ekvivalenciou na množine $\mathbf{Z}$. ■

Poznámka 1.6. Ak ϱ je relácia ekvivalencie a $(x,y) \in \varrho$, tak budeme hovoriť, že x ***je ekvivalentné s*** y.

Definícia 1.24. Nech σ je relácia ekvivalencie na množine A a $a \in A$. Potom množinu $\sigma(a) = \{x \in A; a\,\sigma\,x\}$ nazývame ***triedou ekvivalencie prvku*** a.

Príklad 1.32. Pre reláciu ekvivalencie ϱ definovanú v predchádzajúcom príklade nájdime triedy ekvivalencie prvkov 0, 1, 2.

Riešenie.
$\varrho(0) = \{x \in \mathbf{Z}; 0\,\varrho\,x\} = \{x \in \mathbf{Z}; 3 \mid (x-0)\} = \{x; x = 3k, k \in \mathbf{Z}\} = \{3k; k \in \mathbf{Z}\}$,
$\varrho(1) = \{x \in \mathbf{Z}; 1\,\varrho\,x\} = \{x \in \mathbf{Z}; 3 \mid (x-1)\} = \{x; x = 3k+1, k \in \mathbf{Z}\} = \{3k+1; k \in \mathbf{Z}\}$,
$\varrho(2) = \{x \in \mathbf{Z}; 2\,\varrho\,x\} = \{x \in \mathbf{Z}; 3 \mid (x-2)\} = \{x; x = 3k+2, k \in \mathbf{Z}\} = \{3k+2; k \in \mathbf{Z}\}$. ■

Veta 1.3. Nech σ je relácia ekvivalencie na množine A. Potom pre každé $x, y \in A$

(1) $x \in \sigma(x)$,
(2) $x \in \sigma(y)$ práve vtedy, keď $\sigma(x) = \sigma(y)$,
(3) ak $\sigma(x) \neq \sigma(y)$, tak $\sigma(x) \cap \sigma(y) = \emptyset$.

Dôkaz.
(1) Relácia σ je reflexívna, preto pre každé $x \in A$ platí $(x,x) \in \sigma$, čo znamená, že $x \in \sigma(x)$.

(2) Nech $x \in \sigma(y)$ t.j. $(y,x) \in \sigma$. Ukážeme, že každý prvok $z \in \sigma(x)$ je prvkom aj množiny $\sigma(y)$. Nech teda $z \in \sigma(x)$, čo znamená, že $(x,z) \in \sigma$. Keďže $(y,x) \in \sigma$ a súčasne $(x,z) \in \sigma$, z tranzitívnosti relácie σ vyplýva $(y,z) \in \sigma$, a teda $z \in \sigma(y)$. Ukázali sme, že $\sigma(x) \subset \sigma(y)$. Analogicky sa ukáže, že $\sigma(y) \subset \sigma(x)$. Potom však platí $\sigma(x) = \sigma(y)$.

(3) (nepriamo) Nech $\sigma(x) \cap \sigma(y) \neq \emptyset$. Potom existuje $z \in \sigma(x) \cap \sigma(y)$ a pre tento prvok platí $(x,z) \in \sigma, (y,z) \in \sigma$. Keďže relácia σ je symetrická a tranzitívna, tak $(x,y) \in \sigma$ čiže $x \in \sigma(y)$ a podľa (2) $\sigma(x) = \sigma(y)$, čo je spor. □

Príklad 1.33. V relácii ekvivalencie $\varrho = \{(x,y) \in \mathbf{Z}^2; 3 \mid (y-x)\}$ jedinými triedami ekvivalencie sú $\varrho(0) = \{3k; k \in \mathbf{Z}\}$, $\varrho(1) = \{3k+1; k \in \mathbf{Z}\}$, $\varrho(2) = \{3k+2; k \in \mathbf{Z}\}$, lebo ľubovoľné celé číslo má tvar $3m$, $3m+1$ alebo $3m+2$ ($m \in \mathbf{Z}$) a potom $\varrho(3m) = \varrho(0)$, lebo $3m \in \varrho(0)$; $\varrho(3m+1) = \varrho(1)$, lebo $3m+1 \in \varrho(1)$; $\varrho(3m+2) = \varrho(2)$, lebo $3m+2 \in \varrho(2)$. ■

Definícia 1.25. Nech A je neprázdna množina a T je systém podmnožín množiny A, pre ktorý platí

(1) $\forall X \in T \;\; X \neq \emptyset$,
(2) $\forall X, Y \in T \;\; X \neq Y \Rightarrow X \cap Y = \emptyset$,

(3) $\bigcup\limits_{X \in T} X = A$.

Potom T sa nazýva ***rozklad množiny*** A a prvky množiny T sa nazývajú ***triedy rozkladu množiny*** A.

Príklad 1.34. Nech $A = \{a, b, c, d, e\}$, $A_1 = \{a\}$, $A_2 = \{b, d, e\}$, $A_3 = \{c\}$, $B_1 = \{a, b, c\}$, $B_2 = \{b, d, e\}$. Potom $T = \{A_1, A_2, A_3\}$ je rozklad množiny A, lebo množiny A_1, A_2, A_3 sú neprázdne, prienik každých dvoch rôznych z nich je prázdna množina a zjednotenie všetkých troch množín je množina A. Naproti tomu $S = \{B_1, B_2\}$ nie je rozklad množiny A, lebo $B_1 \neq B_2$ ale $B_1 \cap B_2 = \{b\} \neq \emptyset$. ■

Príklad 1.35. Triedy ekvivalencie $\varrho(0)$, $\varrho(1)$, $\varrho(2)$ relácie ekvivalencie $\varrho = \{(x, y) \in \mathbf{Z}^2; 3 \mid (y - x)\}$ tvoria rozklad množiny $\mathbf{Z}$, pretože $\varrho(0) = \{0, \pm 3, \pm 6, \dots\}$, $\varrho(1) = \{1, -2, 4, -5, 7, -8, 10, \dots\}$, $\varrho(2) = \{2, -1, 5, -4, 8, -7, 11, \dots\}$, čo sú neprázdne množiny, $\varrho(0) \cap \varrho(1) = \emptyset$, $\varrho(0) \cap \varrho(2) = \emptyset$, $\varrho(1) \cap \varrho(2) = \emptyset$ a $\varrho(0) \cup \varrho(1) \cup \varrho(2) = \mathbf{Z}$. ■

Veta 1.4. Nech σ je ekvivalencia na množine A. Potom triedy ekvivalencie $\sigma(a)$, pre $a \in A$ tvoria rozklad množiny A. Triedami rozkladu sú triedy ekvivalencie relácie σ.

Dôkaz. Z vety 1.3 vyplýva, že každá trieda ekvivalencie $\sigma(a)$ prvku $a \in A$ je neprázdna, lebo $a \in A$. V tretej vlastnosti tejto vety sa priamo hovorí, že prienik dvoch rôznych tried ekvivalencie je prázdna množina. Každá trieda ekvivalencie obsahuje len prvky množiny A, preto $\bigcup\limits_{a \in A} \sigma(a) \subset A$. Na druhej strane pre ľubovoľný prvok $a \in A$ platí $a \in \sigma(a)$, a teda aj $a \in \bigcup\limits_{a \in A} \sigma(a)$, preto $A \subset \bigcup\limits_{a \in A} \sigma(a)$. V spojení s predchádzajúcou inklúziou dostávame $\bigcup\limits_{a \in A} \sigma(a) = A$. □

Poznámka 1.7. O rozklade, ktorý je tvorený triedami ekvivalencie, hovoríme, že je to ***rozklad indukovaný*** danou ***ekvivalenciou***.

Veta 1.5. Ku každému rozkladu množiny A existuje jednoznačne určená ekvivalencia na množine A, ktorá tento rozklad indukuje.

Dôkaz. 1. Nech T je rozklad množiny A. Definujme na množine A reláciu σ takto:

$$a \, \sigma \, b \text{ práve vtedy, keď existuje } X \in T \text{ také, že } a, b \in X.$$

Je zrejmé, že táto relácia je reflexívna, symetrická a tranzitívna. Teda je to ekvivalencia. Ďalej vidíme, že prvky patriace do jednej triedy rozkladu sú navzájom ekvivalentné. Preto každá trieda rozkladu X je podmnožinou jednej triedy ekvivalencie relácie σ. Táto trieda ekvivalencie už nemôže obsahovať ďalšiu triedu rozkladu Y ($Y \neq X$), lebo v opačnom prípade by pre prvky $x \in X, y \in Y$ na základe definície relácie σ platilo $(x, y) \notin \sigma$ a na druhej strane $(x, y) \in \sigma$, pretože x aj y patria do tej istej triedy ekvivalencie, a to je spor. Tým sme ukázali, že každá trieda rozkladu T je triedou ekvivalencie σ.

2. Z časti 1 vyplýva, že existuje aspoň jedna ekvivalencia, ktorá indukuje rozklad množiny A, ktorý je zhodný s T. Teraz ukážeme, že táto ekvivalencia je jediná.

Predpokladajme, že existuje ešte jedna ekvivalencia $\tau \subset A \times A$ na množine A, ktorou indukovaný rozklad množiny A na triedy ekvivalencie je opäť T.
Nech $a \, \tau \, b$. Teda existuje $X \in T$ také, že $X = \tau(a) = \tau(b)$. To znamená, že $a, b \in X$, a teda $a \, \sigma \, b$, kde σ je ekvivalencia z prvej časti dôkazu. Z toho vyplýva, že podmienka $(a, b) \in \tau$ implikuje $(a, b) \in \sigma$, a teda $\tau \subset \sigma$.
Nech naopak $(a, b) \in \sigma$. Teda existuje $X \in T$ také, že $a, b \in X$. Ale pretože T je rozklad indukovaný aj ekvivalenciou τ, X je jednou z tried ekvivalencie τ. Preto $(a, b) \in \tau$. Z toho vyplýva, že $\sigma \subset \tau$, a teda spolu s predošlou časťou dostávame $\sigma = \tau$. Tým je veta dokázaná. □

Dokázali sme, že medzi ekvivalenciami na množine A a rozkladmi množiny A je vzájomne jedno-jednoznačné priradenie. Túto skutočnosť budeme využívať tak, že ekvivalenciu budeme definovať pomocou k nej patriaceho rozkladu.

Príklad 1.36. Nech systém množín $T = \{\{a, b, c\}, \{d, e\}, \{f\}\}$ je rozklad množiny $A = \{a, b, c, d, e, f\}$. Potom relácia ekvivalencie σ, ktorá tento rozklad indukuje, je $\sigma = \{(a, a), (b, b), (c, c), (a, b), (b, a), (a, c), (c, a), (b, c), (c, b), (d.d)(e, e), (d, e), (e, d), (f, f)\}$. Vidíme, že zápis ekvivalencie pomocou príslušného rozkladu je oveľa prehľadnejší. ■

5. Orientované grafy

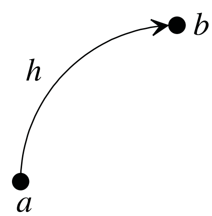
Teraz uvedieme základné pojmy z teórie grafov, ktoré využijeme pri štúdiu teórie konečných automatov.

Definícia 1.26. ***Orientovaný graf*** je usporiadaná trojica $G = (V, H, e)$, kde V, H sú konečné množiny, $V \neq \emptyset, V \cap H = \emptyset$ a e je zobrazenie $e : H \to V^2$. Prvky množiny V sa nazývajú ***vrcholy***, prvky množiny H sa nazývajú ***orientované hrany*** a zobrazenie e sa nazýva ***incidencia***. Ak u, v sú vrcholy, h je orientovaná hrana a $e(h) = (u, v)$, tak u, v sa nazývajú ***krajné vrcholy orientovanej hrany*** h, špeciálne: vrchol u sa nazýva ***začiatočný vrchol orientovanej hrany*** h a vrchol v ***koncový vrchol orientovanej hrany*** h.

Príklad 1.37. Nech $V = \{1, 2, 3, 4\}$, $H = \{h_1, h_2, h_3, h_4, h_5, h_6\}$ $e : e(h_1) = (1, 4)$, $e(h_2) = (4, 2)$, $e(h_3) = (3, 2)$, $e(h_4) = e(h_5) = (3, 4)$, $e(h_6) = (2, 2)$. Potom $G = (V, H, e)$ je orientovaný graf. ■

Definícia 1.27. Grafy $G = (V, H, e), G' = (V', H', e')$ sú rovnaké, ak $V = V', H = H', e = e'$.

Orientované grafy budeme znázorňovať aj graficky a to takýmto spôsobom: vrcholy orientovaného grafu G znázorníme ako body roviny, ktoré označíme rovnako ako samotné vrcholy. Nech h je orientovaná hrana s krajnými vrcholmi a, b. Potom orientovanú hranu



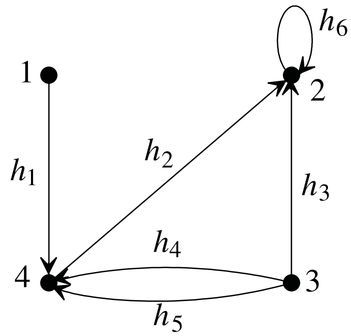
Obr. 1

h s počiatočným vrcholom a a koncovým b znázorníme ako čiaru spájajúcu body a, b so šípkou pri koncovom vrchole b (obr. 1). Takto vytvorený objekt budeme nazývať ***diagram orientovaného grafu*** G. Často však namiesto „diagram grafu G“ budeme hovoriť iba „graf G“.

Poznámka 1.8. Tu sa budeme zaoberať iba orientovanými grafmi, preto slovo orientovaný budeme často vynechávať.

Príklad 1.38. Diagram orientovaného grafu z príkladu 1.37 je na obr. 2. ■

Definícia 1.28. Ak hrana h orientovaného grafu G inciduje s dvoma rôznymi vrcholmi, nazýva sa ***orientovaná linka*** a ak inciduje len s jedným vrcholom, nazýva sa ***orientovaná slučka***.

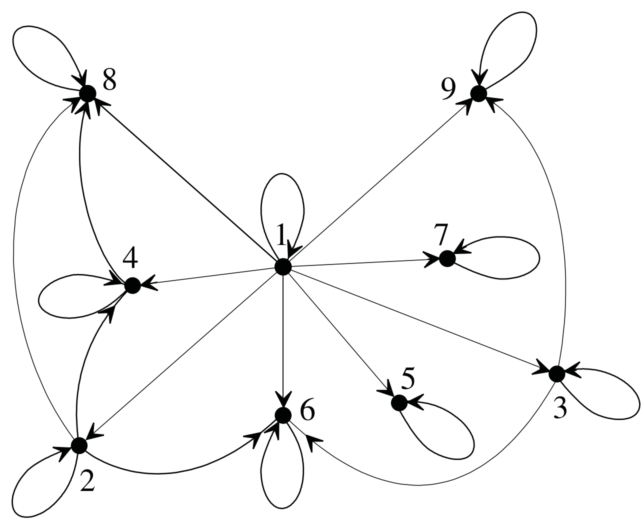


OBR. 2

PRÍKLAD 1.39. V grafe G (obr. 2) hrany h_1, h_2, h_3, h_4, h_5 sú linky a hrana h_6 je slučka.

DEFINÍCIA 1.29. ***Násobnosťou orientovanej hrany so začiatočným vrcholom*** u ***a koncovým vrcholom*** v v orientovanom grafe G nazývame počet orientovaných hrán, ktorých začiatočný vrchol je u a koncový je v. Toto číslo budeme označovať $m(u, v)$.

Orientovaný graf nazývame ***jednoduchý*** (tiež ***prostý***) ***orientovaný graf***, ak pre každé dva jeho vrcholy u, v je $m(u, v) \leq 1$. Orientovaný graf, ktorý nie je jednoduchý sa nazýva ***orientovaný multigraf***.



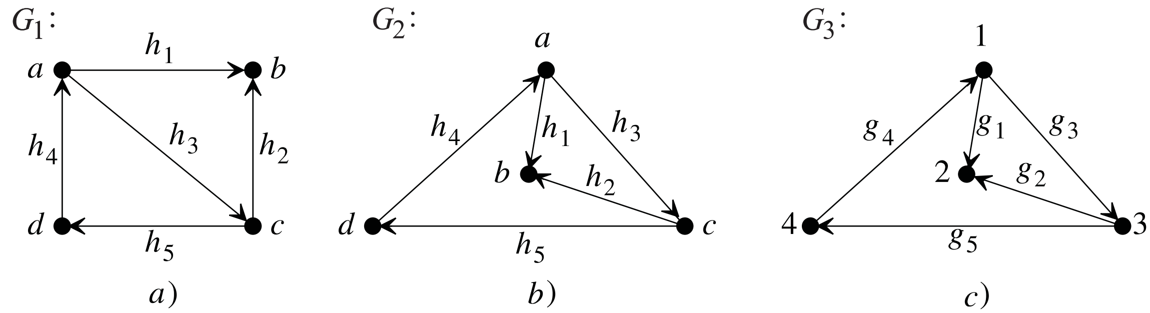
OBR. 3

Jednoduché grafy, presnejšie ich diagramy, nám umožňujú grafické znázornenie binárnych relácií na množine. Deje sa to takto:

Nech A je množina a $\sigma \subset A \times A$ je binárna relácia na množine A. Definujme graf $G = (A, \sigma, e)$, ktorého vrcholmi sú prvky množiny A, hranami sú usporiadané dvojice patriace do relácie σ a incidenciou je zobrazenie $e : \sigma \to A \times A$, $e(x, y) = (x, y)$. Graf $G = (A, \sigma, e)$ sa nazýva ***orientovaný graf binárnej relácie*** σ.

PRÍKLAD 1.40. Na množine $A = \{1, 2, 3, 4, 5, 6, 7, 8, 9\}$ je definovaná relácia $\varrho = \{(x, y) \in A^2; x \mid y\}$. Orientovaný graf relácie ϱ (presnejšie jeho diagram) je na obr. 3. ■

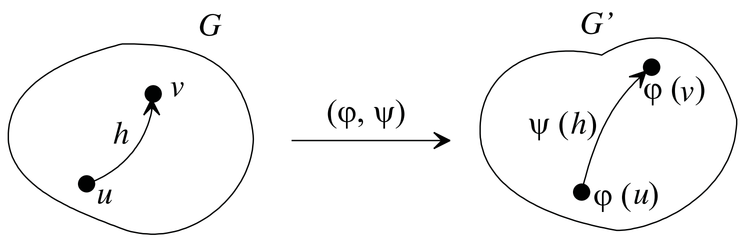
Všimnime si diagramy grafov G_1, G_2 a G_3 na obr. 4. Grafy G_1 a G_2 sú rovnaké (majú rovnaké množiny vrcholov, množiny hrán a tiež incidencie). Grafy G_1 a G_3 sú rôzne. Ale keď porovnáme ich diagramy (G_1 na obr. 4a resp. 4b, lebo $G_1 = G_2$ a G_3 na obr. 4c),



Obr. 4

vidíme, že medzi týmito grafmi nie je podstatný rozdiel, líšia sa iba v pomenovaní vrcholov a hrán. Takéto grafy budeme nazývať izomorfné.
Presná definícia je takáto:

Definícia 1.30. Orientované grafy $G = (V, H, e), G' = (V', H', e')$ sa nazývajú ***izomorfné***, ak existujú bijektívne zobrazenia $\varphi : V \to V',\ \psi : H \to H'$ také, že pre každú hranu $h \in H$ platí: ak $e(h) = (u, v)$, tak $e'(\psi(h)) = (\varphi(u), \varphi(v))$ (obr. 5).



Obr. 5

Veta 1.6. Nech $G = (V, H, e), G' = (V', H', e')$ sú jednoduché orientované grafy. Potom nasledujúce tvrdenia sú ekvivalentné:

(a) G a G' sú izomorfné.
(b) Existuje bijekcia $\varphi : V \to V'$ tak, že pre všetky $u, v \in V$ platí: u je začiatočný a v koncový vrchol jednej hrany práve vtedy, keď $\varphi(u)$ je začiatočný a $\varphi(v)$ koncový vrchol jednej hrany.

Dôkaz. Priamo z definície izomorfizmu grafov je zrejmé, že z (a) vyplýva (b). Dokážeme, že z (b) vyplýva (a). Nech teda $\varphi : V \to V'$ je bijekcia splňujúca podmienku: u je začiatočný a v koncový vrchol jednej hrany v grafe G práve vtedy, keď $\varphi(u)$ je začiatočný a $\varphi(v)$ koncový vrchol jednej hrany v grafe G'. Uvedená vlastnosť zobrazenia φ nám umožňuje definovať zobrazenie ψ medzi množinami hrán grafov G a G' týmto spôsobom:

$$\psi : H \to H',\ \psi(h) = h',\ \text{pričom platí: ak } e(h) = (u, v),\ \text{tak } e'(h') = (\varphi(u), \varphi(v))$$

Keďže φ je bijekcia a h je hrana v G práve vtedy, keď h' je hrana v G', je ψ tiež bijekcia. $\square$

Predchádzajúcu vetu je možné zovšeobecniť aj pre multigrafy.

Veta 1.7. Orientované grafy $G = (V, H, e), G' = (V', H', e')$ sú izomorfné práve vtedy, keď existuje bijekcia $\varphi : V \to V'$ a pre každé vrcholy u, v orientovaného grafu G je $m(u, v) = m(\varphi(u), \varphi(v))$.

Dôkaz. Nech teda orientované grafy G a G' sú izomorfné. To znamená, že existujú také bijekcie $\varphi : V \to V'$, $\psi : H \to H'$, že pre každú hranu $h \in H$ platí: ak $e(h) = (x, y)$, tak $e'(\psi(h)) = (\varphi(x), \varphi(y))$. Nech $u, v \in V$ a $m(u, v) = k$. Ak $k = 0$, znamená to, že v grafe G neexistuje hrana so začiatočným vrcholom u a koncovým v. Potom ani v grafe G' neexistuje hrana so začiatočným vrcholom $\varphi(u)$ a koncovým $\varphi(v)$, v opačnom prípade by ψ nebola bijekcia. Ak $k \geq 1$, tak existuje k orientovaných hrán $h_1, \ldots, h_k$, pre ktoré je

$$e(h_i) = (u, v),\ i \in \{1, \ldots, k\}.$$

Vzhľadom na to, že

$$e'(\psi(h_i)) = (\varphi(u), \varphi(v)),\ \text{pre } i \in \{1, \ldots, k\}$$

a ψ je bijekcia, je $m(\varphi(u), \varphi(v)) = k$.

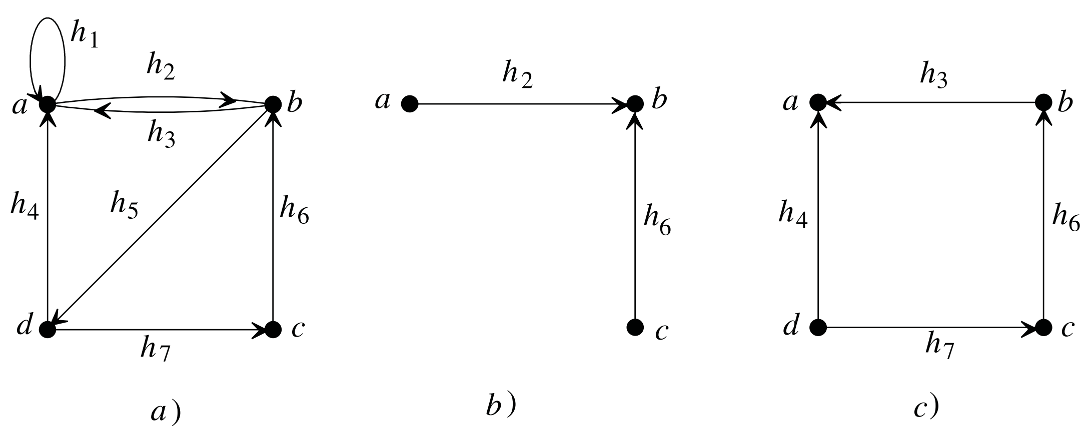
Dokážeme opačnú implikáciu. Nech teda existuje bijekcia $\varphi : V \to V'$ tak, že pre všetky $x, y \in V$ je $m(x, y) = m(\varphi(x), \varphi(y))$. Nech u je začiatočný a v koncový vrchol niektorej orientovanej hrany grafu G a nech $m(u, v) = k \geq 1$. Existuje teda práve k orientovaných hrán $h_1, \ldots, h_k$ so začiatočným vrcholom u a koncovým v. Keďže $m(\varphi(u), \varphi(v)) = m(u, v)$, existuje v grafe G' k orientovaných hrán $g_1, \ldots, g_k$ so začiatočným vrcholom $\varphi(u)$ a koncovým $\varphi(v)$. To nám umožňuje definovať zobrazenie ψ takto:

$$\psi : H \to H',\ \psi(h_i) = g_i \text{ pre } i \in \{1, \ldots, k\}.$$

Ľahko sa ukáže (prenechávame to čitateľovi), že ψ je bijektívne zobrazenie a že zobrazenia φ a ψ zachovávajú incidenciu. Grafy G a G' sú teda izomorfné. □

Definícia 1.31. Orientovaný graf $G' = (V', H', e')$ sa nazýva ***podgraf*** orientovaného grafu $G = (V, H, e)$, ak $V' \subset V$, $H' \subset H$, a e' je zúžením incidencie e na množinu hrán H', t.j. pre každú hranu $h \in H'$ je $e'(h) = e(h)$.

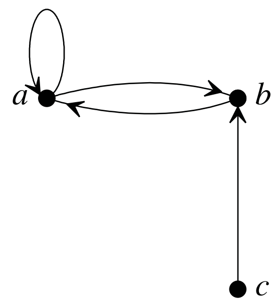
Príklad 1.41. Na obr. 6b, 6c sú podgrafy grafu G, ktorého diagram je na obr. 6a. ■



Obr. 6

Definícia 1.32. Nech $G = (V, H, e)$ je orientovaný graf, $V' \subset V$, $V' \neq \emptyset$. Podgraf $G(V')$ orientovaného grafu G, ktorého množina vrcholov je V' a množina hrán H' je určená vlastnosťou: ak $u, v \in V'$, tak H' obsahuje všetky hrany z H, ktoré incidujú s oboma vrcholmi u, v, sa nazýva ***indukovaný podgraf***.

Príklad 1.42. Nech G je graf z obr. 6a, $V' = \{a, b, c\}$. Indukovaný podgraf $G(V')$ grafu G je na obr. 7. ■



Obr. 7

Definícia 1.33. Nech u, v sú vrcholy orientovaného grafu G, $k \in \mathbf{N}$. ***Orientovaným sledom dĺžky*** k ***z vrchola*** u ***do vrchola*** v nazývame postupnosť

$$v_0, h_1, v_1, h_2, v_2, \ldots, v_{k-1}, h_k, v_k,$$

kde

1. $v_0, v_1, \ldots, v_k$ sú vrcholy orientovaného grafu G,
2. $h_1, \ldots, h_k$ sú hrany orientovaného grafu G,
3. pre $i \in \{1, \ldots, k\}$ je v_{i-1} začiatočný a v_i koncový vrchol hrany h_i,
4. $v_0 = u,\ v_n = v$.

Vrchol u sa nazýva ***začiatočný*** a vrchol v ***koncový vrchol*** tohto sledu.

Príklad 1.43. V grafe z obr. 6a je
$a, h_1, a, h_2, b, h_5, d$ orientovaným sledom z vrcholu a do vrcholu d dĺžky 3,
a, h_2, b, h_5, d orientovaným sledom z vrcholu a do vrcholu d dĺžky 2,
a, h_1, a orientovaným sledom z vrcholu a do vrcholu a dĺžky 1,
a orientovaným sledom z vrcholu a do vrcholu a dĺžky 0.

Definícia 1.34. Orientovaný sled, v ktorom sa každá hrana grafu vyskytuje najviac raz, sa nazýva ***orientovaný ťah***.

Orientovaný sled, v ktorom sa každý vrchol grafu vyskytuje najviac raz, sa nazýva ***orientovaná cesta***.

Definícia 1.35. Graf $G = (V, H, e)$ sa nazýva ***silne súvislý*** graf, ak pre každé dva vrcholy $u, v \in V$ existuje orientovaný sled z vrchola u do vrchola v.

Definícia 1.36. Podgraf F grafu G sa nazýva ***silne súvislý komponent*** grafu G, ak platí:

1. F je indukovaný podgraf grafu G.
2. F je silne súvislý graf.
3. Ak $F' \neq F$ je taký podgraf grafu G, že F je jeho podgrafom, tak F' už nie je silne súvislým grafom.

V aplikáciách často k adekvátnemu opisu študovaného systému nestačí orientovaný graf, ale je potrebné k hranám a vrcholom pripísať nejaké údaje (najčastejšie číselné hodnoty). Graf, ktorého hrany a (alebo) vrcholy sú označené číselnými (alebo inými) hodnotami, sa nazýva ***ohodnotený graf***.

KAPITOLA 2

Booleovské funkcie

V úvode sme sa stretli s pravdivostnými ohodnoteniami výrokových foriem, čo sú funkcie, ktoré zobrazujú prvky množiny $\{0,1\}^n$ do množiny $\{0,1\}$. Takýmito funkciami sa budeme v tejto kapitole podrobnejšie zaoberať. Dohodnime sa, že ďalej budeme používať označenie $\mathbf{B} = \{0,1\}$.

1. Booleovské funkcie a booleovské výrazy

Definícia 2.1. Každá funkcia $f: \mathbf{B}^n \to \mathbf{B}$, $n \in \mathbf{N}^+$ sa nazýva ***booleovská*** (tiež ***logická***) ***funkcia*** n premenných.

Dohoda. Ako premenné v booleovských funkciách budeme používať x, y, z, u, v, resp. tieto premenné s indexami.

Funkcie jednej premennej sú uvedené v tab. 1.

Tabuľka 1. Booleovské funkcie jednej premennej

x	$f_1(x)$	$f_2(x)$	$f_3(x)$	$f_4(x)$
0	0	0	1	1
1	0	1	0	1

Existujú aj ďalšie booleovské funkcie jednej premennej? Ak by sme chceli do tabuľky 1 pridať ďalšiu funkciu, t.j. vložiť do tabuľky ďalší stĺpec núl a jednotiek, zistíme, že je rovný niektorému z predchádzajúcich štyroch stĺpcov. Znamená to, že booleovské funkcie jednej premennej sú iba štyri.

Koľko existuje booleovských funkcií dvoch premenných? Zapíšme tieto funkcie do tabuľky. Keďže definičný obor booleovskej funkcie dvoch premenných má štyri prvky, bude mať tabuľka (okrem hlavičky) štyri riadky. Funkcií bude toľko, koľko existuje rôznych stĺpcov núl a jednotiek. V každom riadku je jedna z dvoch hodnôt 0, 1, preto môžeme vytvoriť $2 \cdot 2 \cdot 2 \cdot 2 = 2^4 = 2^{2^2} = 16$ rôznych booleovských funkcií dvoch premenných. Uvádzame ich v tab. 2.

Veta 2.1. Booleovských funkcií n premenných je 2^{2^n}.

Dôkaz. Počet všetkých n-tíc prvkov z množiny $\mathbf{B}$ je 2^n. Aby sme definovali booleovskú funkciu n premenných, musíme pre každú z týchto 2^n n-tíc zvoliť jednu funkčnú hodnotu z dvoch prvkov množiny $\mathbf{B}$. Každá takáto funkcia je teda určená 2^n-ticou núl a jednotiek a tých je možné vytvoriť $\underbrace{2 \cdot 2 \cdot \ldots \cdot 2}_{2^n} = 2^{2^n}$. □

Poznámka 2.1. Takéto sú počty booleovských funkcií pre niektoré hodnoty n:

$n = 1$, $2^{2^1} = 2^2 = 4$ | $n = 4$, $2^{2^4} = 2^{16} = 65\,536$

$n = 2$, $2^{2^2} = 2^4 = 16$ | $n = 5$, $2^{2^5} = 2^{32} \doteq 4,29497 \times 10^9$

$n = 3$, $2^{2^3} = 2^8 = 256$ | $n = 10$, $2^{2^{10}} = 2^{1024} \doteq 1,7976931348 \times 10^{308}$

TABUĽKA 2. Booleovské funkcie dvoch premenných

(x,y)	$f_5(x,y)$	$f_6(x,y)$	$f_7(x,y)$	$f_8(x,y)$	$f_9(x,y)$	$f_{10}(x,y)$	$f_{11}(x,y)$	$f_{12}(x,y)$
$(0,0)$	0	0	0	0	1	0	0	0
$(0,1)$	0	0	0	1	0	0	1	1
$(1,0)$	0	0	1	0	0	1	0	1
$(1,1)$	0	1	0	0	0	1	1	0

(x,y)	$f_{13}(x,y)$	$f_{14}(x,y)$	$f_{15}(x,y)$	$f_{16}(x,y)$	$f_{17}(x,y)$	$f_{18}(x,y)$	$f_{18}(x,y)$	$f_{20}(x,y)$
$(0,0)$	1	1	1	0	1	1	1	1
$(0,1)$	0	0	1	1	0	1	1	1
$(1,0)$	0	1	0	1	1	0	1	1
$(1,1)$	1	0	0	1	1	1	0	1

Každá booleovská funkcia n premenných je n-árnou operáciou na množine $\mathbf{B}$. Všimnime si špeciálne jednu unárnu – f_3 (tab. 1) a dve binárne – f_6, f_{16} (tab. 2). Ak porovnáme tabuľky týchto funkcií a pravdivostné tabuľky výrokových formúl $\bar{p}, p \wedge g, p \vee q$, zistíme, že pravdivostnými ohodnoteniami týchto formúl v danom poradí sú funkcie f_3, f_6, f_{16}, čiže

$$\mathrm{ph}_{\overline{p}}(x) = f_3(x),\ \mathrm{ph}_{p\wedge q}(x,y) = f_6(x,y),\ \mathrm{ph}_{p\vee q}(x,y) = f_{16}(x,y).$$

DEFINÍCIA 2.2. ***Logickým súčtom*** nazývame binárnu operáciu na množine $\mathbf{B}$, ktorú označujeme $+$ a definujeme takto:

$$x + y = f_{16}(x,y).$$

Logickým súčinom nazývame binárnu operáciu na množine $\mathbf{B}$, ktorú označujeme $\cdot$ a definujeme takto:

$$x \cdot y = f_6(x,y).$$

Negáciou (alebo ***komplementom***) nazývame unárnu operáciu na množine $\mathbf{B}$, ktorú označujeme $\bar{\ }$ a definujeme takto:

$$\overline{x} = f_3(x).$$

Z tabuliek funkcií f_6 a f_{16} vidieť, že

$$0+0=0,\ 0+1=1,\ 1+0=1,\ 1+1=1,$$

$$0\cdot 0=0,\ 0\cdot 1=0,\ 1\cdot 0=0,\ 1\cdot 1=1,$$

čo môžeme zapísať aj pomocou Cayleyho multiplikačných tabuliek, ktoré v tomto prípade majú tvar:

$+$	0	1
0	0	1
1	1	1

$\cdot$	0	1
0	0	0
1	0	1

Z tabuľky funkcie f_3 zase zistíme, že

$$\overline{0} = 1, \quad \overline{1} = 0.$$

Poznámka 2.2. Množinu $\mathbf{B}$ spolu s binárnymi operáciami logický súčet, logický súčin a unárnou operáciou negácia nazývame ***Booleova algebra***.

Práve definované operácie nám umožňujú zapísať mnohé booleovské funkcie šikovnejším spôsobom než sú tabuľky, napr. $f_{17}(x,y) = x + (\overline{x} \cdot \overline{y})$. O platnosti tejto rovnosti sa ľahko presvedčíme:

$0 + (\overline{0} \cdot \overline{0}) = 0 + (1 \cdot 1) = 0 + 1 = 1 = f_{17}(0,0)$,
$0 + (\overline{0} \cdot \overline{1}) = 0 + (1 \cdot 0) = 0 + 0 = 0 = f_{17}(0,1)$,
$1 + (\overline{1} \cdot \overline{0}) = 1 + (0 \cdot 1) = 1 + 0 = 1 = f_{17}(1,0)$,
$1 + (\overline{1} \cdot \overline{1}) = 1 + (0 \cdot 0) = 1 + 0 = 1 = f_{17}(1,1)$,

Veta 2.2. Pre každé $x, y, z \in \mathbf{B}$ platí

1.	$x + y = y + x$	$x \cdot y = y \cdot x$	komutatívne zákony
2.	$(x + y) + z = x + (y + z)$	$(x \cdot y) \cdot z = x \cdot (y \cdot z)$	asociatívne zákony
3.	$(x + y) \cdot z = (x \cdot z) + (y \cdot z)$	$(x \cdot y) + z = (x + z) \cdot (y + z)$	distributívne zákony
4.	$\overline{x + y} = \overline{x} \cdot \overline{y}$	$\overline{x \cdot y} = \overline{x} + \overline{y}$	de Morganove zákony
5.	$x + x = x$	$x \cdot x = x$	zákony idempotentnosti
6.	$x + \overline{x} = 1$	$x \cdot \overline{x} = 0$	zákony komplementárnosti
7.	$\overline{\overline{x}} = x$		zákon involúcie
8.	$x + (x \cdot y) = x$	$x \cdot (x + y) = x$	zákony absorpcie
9.	$x + 0 = x$	$x \cdot 1 = x$	zákony identity
10.	$x + 1 = 1$	$x \cdot 0 = 0$	zákon jednotkového sčitovania a nulového násobenia

Dôkaz. Dokážeme napr. vlastnosť $x + (x \cdot y) = x$. Ostatné vlastnosti sa dajú dokázať analogicky. Označme $\mathrm{L}(x,y) = x + (x \cdot y)$ a $\mathrm{P}(x,y) = x$. Potom

$\mathrm{L}(0,0) = 0 + (0 \cdot 0) = 0 + 0 = 0 = \mathrm{P}(0,0)$,
$\mathrm{L}(0,1) = 0 + (0 \cdot 1) = 0 + 0 = 0 = \mathrm{P}(0,1)$,
$\mathrm{L}(1,0) = 1 + (1 \cdot 0) = 1 + 0 = 1 = \mathrm{P}(1,0)$,
$\mathrm{L}(1,1) = 1 + (1 \cdot 1) = 1 + 1 = 1 = \mathrm{P}(1,1)$, □

Z tabuliek operácií logického súčtu a súčinu priamo vidieť, že platí nasledujúce tvrdenie.

Veta 2.3. Pre každé $x, y \in \mathbf{B}$ platí

$$\begin{aligned} x + y = 1 &\Leftrightarrow x = 1 \vee y = 1 \\ x + y = 0 &\Leftrightarrow x = 0 \wedge y = 0 \\ x \cdot y = 1 &\Leftrightarrow x = 1 \wedge y = 1 \\ x \cdot y = 0 &\Leftrightarrow x = 0 \vee y = 0 \end{aligned}$$

□

Označme F_n množinu všetkých booleovských funkcií n premenných. Na množine F_n definujme operácie súčet, súčin a negáciu, ktoré budeme označovať v poradí $+, \cdot, ^-$.

Definícia 2.3. Nech $f, g : \mathbf{B}^n \to \mathbf{B}$ sú booleovské funkcie, potom ***súčtom booleovských funkcií*** f, g nazývame funkciu

$f + g : \mathbf{B}^n \to \mathbf{B}$, $(f + g)(x_1, \ldots, x_n) = f(x_1, \ldots, x_n) + g(x_1, \ldots, x_n)$,

súčinom booleovských funkcií f, g nazývame funkciu

$f \cdot g : \mathbf{B}^n \to \mathbf{B}$, $(f \cdot g)(x_1, \ldots, x_n) = f(x_1, \ldots, x_n) \cdot g(x_1, \ldots, x_n)$,

negáciou funkcie f nazývame funkciu

$\overline{f} : \mathbf{B}^n \to \mathbf{B}$, $\overline{f}(x_1, \ldots, x_n) = \overline{f(x_1, \ldots, x_n)}$.

Príklad 2.1. $f(x, y, z) = x + (y \cdot z)$, $g(x, y, z) = y + z$ sú booleovské funkcie troch premenných. Aký je predpis funkcie $h = \overline{f} \cdot g$?

Riešenie. $h(x, y, z) = (\overline{f} \cdot g)(x, y, z) = \overline{f}(x, y, z) \cdot g(x, y, z) = \overline{f(x, y, z)} \cdot g(x, y, z) = \overline{x + (y \cdot z)} \cdot (y + z)$.

Poznámka 2.3.

1. Vlastnosti z vety 2.2 zostanú v platnosti aj v prípade, keby x, y, z boli booleovské funkcie n premenných. Množina F_n spolu s práve definovanými operáciami súčtu, súčinu a negácie je tiež Booleova algebra. (Booleovu algebru by sme mohli definovať ako ľubovoľnú množinu $\mathbf{B}$, na ktorej sú definované operácie $+$, $\cdot$, $^{-}$ tak, že sú splnené vlastnosti vety 2.2.)

2. V každej Booleovej algebre platí ***princíp duality***: Ak v pravdivom tvrdení zmeníme $+$ na $\cdot$, 0 na 1 a naopak, dostaneme opäť pravdivé tvrdenie. Overte si, že je tento princíp splnený vo všetkých desiatich vlastnostiach vety 2.2.

Podľa zákona absorpcie sú funkcie dvoch premenných $f(x, y) = x$, $g(x, y) = x + (x \cdot y)$ rovnaké, aj keď výrazy x, $x + (x \cdot y)$, pokiaľ sa ne pozeráme ako na postupnosti znakov, sú rôzne. Vidíme, že niektoré funkcie sa dajú vyjadriť viacerými výrazmi vytvorenými z premenných, znakov operácií $+$, $\cdot, ^{-}$ a zátvoriek. Na druhej strane sa vynára otázka, či je možné pomocou takýchto výrazov vyjadriť každú booleovskú funkciu. Poďme sa týmito výrazmi zaoberať podrobnejšie.

Definícia 2.4. ***Booleovský výraz*** (krátko ***B-výraz***) n premenných $x_1, \dots, x_n$ je slovo nad abecedou $\{+, \cdot, ^{-}, (,), 0, 1, x_1, \dots, x_n\}$ definované takto:

(1) $x_1, \dots, x_n, 0, 1$ sú booleovské výrazy.
(2) Ak U, V sú booleovské výrazy, tak aj $\overline{U}, (U + V), (U \cdot V)$ sú booleovské výrazy.
(3) Iné slová nie sú booleovské výrazy.

Dohoda.

(1) V samostatne stojacom B-výraze vynechávame vonkajšie zátvorky.
(2) Princíp priority „$\cdot$ pred $+$“: $x \cdot y + z$ znamená $(x \cdot y) + z$.
(3) Znak logického súčinu budeme často vynechávať, teda namiesto $x \cdot y$ budeme písať xy.
(4) B-výraz U n premenných $x_1, \dots, x_n$ označujeme aj $U(x_1, \dots, x_n)$.

Poznamenajme, že body 1 a 2 tejto dohody sú analogické ako v dohode o formulách.

Príklad 2.2.

(1) Premenné x, y, z sú B-výrazy.
(2) $x + y, xy, yz, \overline{zx}$ sú B-výrazy.
(3) $(x + y)z, xyz, yz + x$ sú B-výrazy.
(4) $x + y + +z$ nie je B-výraz. ■

Poznámka 2.4. Keďže B-výrazy sú slová, tak rovnosť B-výrazov je rovnosť slov, čiže dva B-výrazy sú rovnaké, ak majú rovnaký počet písmen a rovnaké prvé písmená, atď., rovnaké posledné písmená. Napríklad nerovnajú sa B-výrazy $(x + y)z$, $xz + yz$. Pritom, ako ľahko zistíme z tabuliek, ako booleovské funkcie sú rovnaké. Preto je vždy potrebné vedieť, či ide o B-výraz, alebo booleovskú funkciu.

Každý B-výraz $U(x_1, \dots, x_n)$ určuje jednoznačne booleovskú funkciu. Jej hodnotu v bode $(\beta_1, \dots, \beta_n) \in \mathbf{B}^n$ určíme tak, že za premenné $x_1, \dots, x_n$ dosadíme prvky $\beta_1, \dots, \beta_n$ a vykonáme uvedené operácie s týmito prvkami. Túto funkciu môžeme označiť takisto $U(x_1, \dots, x_n)$. Ak však chceme, aby bolo priamo z označenia vidieť, že je to funkcia určená

B-výrazom U, použijeme označenie f_U resp. $f_U(x_1, \ldots, x_n)$. Tiež hovoríme, že ***B-výraz*** $U(x_1, \ldots, x_n)$ ***reprezentuje funkciu*** $f_U(x_1, \ldots, x_n)$.

Príklad 2.3. B-výraz $U(x, y, z) = (x + \overline{y})z$ určuje funkciu $f_U(x, y, z) = (x + \overline{y})z$. Jej hodnota v bode $(0, 0, 1)$ je $f_U(0, 0, 1) = (0 + \overline{0}) \cdot 1 = (0 + 1) \cdot 1 = 1$. ■

Definícia 2.5. B-výrazy $U(x_1, x_2, \ldots, x_n)$, $V(x_1, x_2, \ldots, x_n)$ sa nazývajú ***ekvivalentné***, ak určujú rovnakú booleovskú funkciu. Označujeme to

$$U \cong V, \text{ resp. } U(x_1, x_2, \ldots, x_n) \cong V(x_1, x_2, \ldots, x_n).$$

Poznámka 2.5. Predchádzajúcu definíciu môžeme stručne zapísať takto:

$$U \cong V \Leftrightarrow f_U = f_V.$$

Príklad 2.4. Dokážte, že $xz + yz \cong (x + y)\, z$.

Riešenie. Vytvorme tabuľky funkcií určených týmito B-výrazmi (tab. 3). ■

Tabuľka 3. Tabuľka funkcií z príkladu 2.4

x, y, z	xz	yz	$xz + yz$	$x + y$	$(x + y)z$
0, 0, 0	0	0	0	0	0
0, 0, 1	0	0	0	0	0
0, 1, 0	0	0	0	1	0
0, 1, 1	0	1	1	1	1
1, 0, 0	0	0	0	1	0
1, 0, 1	1	0	1	1	1
1, 1, 0	0	0	0	1	0
1, 1, 1	1	1	1	1	1

Príklad 2.5. Dokážte, že pre B-výrazy platí

(1) $x + y \cong \overline{\overline{x}\,.\overline{y}}$,
(2) $x.y \cong \overline{\overline{x} + \overline{y}}$.

Riešenie. Stačí ukázať, že B-výrazy $x + y$, $\overline{\overline{x}\,.\overline{y}}$ a B-výrazy y, $\overline{\overline{x} + \overline{y}}$ určujú rovnaké booleovské funkcie. Vidíme to z nasledujúcej spoločnej tab. 4. ■

Tabuľka 4. Tabuľka funkcií z príkladu 2.5

x, y	$\overline{x}, \overline{y}$	$\overline{x} + \overline{y}$	$\overline{x}.\overline{y}$	$\overline{\overline{x} + \overline{y}}$	$x.y$	$\overline{\overline{x}.\overline{y}}$	$x + y$
0, 0	1, 1	1	1	0	0	0	0
0, 1	1, 0	1	0	0	0	1	1
1, 0	0, 1	1	0	0	0	1	1
1, 1	0, 0	0	0	1	1	1	1

Veta 2.4. (Tabuľka ekvivalencií B-výrazov.) Nech U, V, W sú B-výrazy s premennými $x_1, x_2, \ldots, x_n$. Potom platí:

E1.	$U+V \cong V+U$	$UV \cong VU$	komutatívne zákony
E2.	$(U+V)+W \cong U+(V+W)$	$(UV)W \cong U(VW)$	asociatívne zákony
E3.	$(U+V)W \cong UW+VW$	$UV+W \cong (U+W)(V+W)$	distributívne zákony
E4.	$\overline{U+V} \cong \overline{U}\,\overline{V}$	$\overline{UV} \cong \overline{U}+\overline{V}$	de Morganove zákony
E5.	$U+U \cong U$	$U\,U \cong U$	zákony idempotentnosti
E6.	$U+\overline{U} \cong 1$	$U\overline{U} \cong 0$	zákony komplementárnosti
E7.	$U \cong \overline{\overline{U}}$		zákon involúcie
E8.	$U+(UV) \cong U$	$U(U+V) \cong U$	zákon absorpcie (pohltenia)
E9.	$U+0 \cong U$	$U \cdot 1 \cong U$	zákony identity
E10.	$U+1 \cong 1$	$U \cdot 0 \cong 0$	zákony jednotkového sčitovania a nulového násobenia

Dôkaz. $f_{U+V} = f_U + f_V = f_V + f_U = f_{V+U}$. Z toho vyplýva $U+V \cong V+U$. Ostatné vlastnosti sa dokážu podobne. □

VETA 2.5. (zákony stability) Pre každé B-výrazy U, V, W platí: ak $U \cong V$, tak

$$
\begin{aligned}
U+W &\cong V+W\\
UW &\cong VW\\
\overline{U} &\cong \overline{V}
\end{aligned}
$$

Dôkaz. B-výrazy U, V sú ekvivalentné, preto nimi určené booleovské funkcie sú rovnaké, t.j. $f_U = f_V$. Potom

$$
\begin{aligned}
f_{U+W} &= f_U + f_W = f_V + f_W = f_{V+W}\\
f_{UW} &= f_U f_W = f_V f_W = f_{VW}\\
f_{\overline{U}} &= \overline{f_U} = \overline{f_V} = f_{\overline{V}}
\end{aligned}
$$

odkiaľ už vyplýva tvrdenie vety. □

DOHODA. Dohodneme sa na ďalších zjednodušeniach pri zápise B-výrazov.

1. Nebudeme rozlišovať medzi B-výrazmi $U+V$, $V+U$ a tiež medzi UV, VU.
2. B-výrazy $(U+V)+W$, $U+(V+W)$ budeme považovať za rovnaké a miesto nich budeme písať $U+V+W$. Podobne namiesto $(UV)W$ či $U(VW)$ budeme písať UVW.
3. Za rovnaké budeme považovať aj B-výrazy $1 \cdot U$, $0+U$, U, tiež $1+U$, 1 a tiež $0 \cdot U$, 0.

PRÍKLAD 2.6. Napíšte B-výraz $xyzu + xyz\overline{u} + x\overline{y}zu$ pomocou najviac 5 výskytov písmen x, y, z, u.

Riešenie.

$$
xyzu + xyz\overline{u} + x\overline{y}zu \overset{E5}{\cong} xyzu + xyzu + xyz\overline{u} + x\overline{y}zu \overset{E1}{\cong} xyzu + xyz\overline{u} + xyzu + x\overline{y}zu \overset{E3,E1}{\cong}
$$

$$
\cong xyz(u+\overline{u}) + xzu(y+\overline{y}) \overset{E6}{\cong} xyz + xzu \overset{E3,E1}{\cong} xz(y+u).
$$ ■

DEFINÍCIA 2.6. Nech $f : \mathbf{B}^n \to \mathbf{B}$. Prvok $(\alpha_1, \dots, \alpha_n) \in \mathbf{B}^n$ sa nazýva

jednotkový bod funkcie f, ak $f(\alpha_1, \dots, \alpha_n) = 1$,
nulový bod funkcie f, ak $f(\alpha_1, \dots, \alpha_n) = 0$.

Množinu všetkých jednotkových resp. nulových bodov funkcie f budeme označovať $J(f)$ resp. $N(f)$. Tieto pojmy budeme vzťahovať aj na B-výrazy, ktorými je funkcia f určená.

VETA 2.6. Nech f, g sú booleovské funkcie n premenných. Potom

(1)	$f = g \;\Leftrightarrow\; J(f) = J(g)$,	$f = g \;\Leftrightarrow\; N(f) = N(g)$,
(2)	$J(f+g) = J(f) \cup J(g)$,	$N(f+g) = N(f) \cap N(g)$,
(3)	$J(f \cdot g) = J(f) \cap J(g)$,	$N(f \cdot g) = N(f) \cup N(g)$,
(4)	$J(\overline{f}\,) = N(f)$,	$N(\overline{f}\,) = J(f)$.

Dôkaz.

(1) Tvrdenie vyplýva z toho, že každý bod definičného oboru booleovskej funkcie je buď jednotkový alebo nulový.

(2) Z vlastnosti

$$f(x_1,\dots,x_n) + g(x_1,\dots,x_n) = 1 \Leftrightarrow f(x_1,\dots,x_n) = 1 \vee g(x_1,\dots,x_n) = 1$$

vyplýva $J(f+g) = J(f) \cup J(g)$.

(3) Z vlastnosti

$$f(x_1,\dots,x_n) \cdot g(x_1,\dots,x_n) = 1 \Leftrightarrow f(x_1,\dots,x_n) = 1 \wedge g(x_1,\dots,x_n) = 1$$

vyplýva $J(f \cdot g) = J(f) \cap J(g)$.

(4) Z vlastnosti

$$\overline{f(x_1,\dots,x_n)} = 1 \Leftrightarrow f(x_1,\dots,x_n) = 0$$

vyplýva $J(\overline{f}\,) = N(f)$.

Tvrdenia o nulových bodoch sa dokazujú analogicky. □

Príklad 2.7. Určte množinu všetkých jednotkových bodov funkcie

$$f(x,y,z,u) = xy + \overline{x}\,\overline{z}u + yz\overline{u}.$$

Riešenie. Funkcia f je súčtom troch funkcií

$$f_1(x,y,z,u) = xy,\;\; f_2(x,y,z,u) = \overline{x}\,\overline{z}u,\;\; f_3(x,y,z,u) = yz\overline{u},$$

preto $J(f) = J(f_1) \cup J(f_2) \cup J(f_3)$. Z vlastnosti: súčin prvkov z $\mathbf{B}$ sa rovná 1 práve vtedy, keď každý prvok sa rovná 1, vyplýva

$$J(f_1) = \{(1,1,0,0),(1,1,0,1),(1,1,1,0),(1,1,1,1)\},$$
$$J(f_2) = \{(0,0,0,1),(0,1,0,1)\},$$
$$J(f_3) = \{(0,1,1,0),(1,1,1,0)\}.$$

Potom

$$J(f) = \{(0,0,0,1),(0,1,0,1),(0,1,1,0),(1,1,1,1),(1,1,0,0),(1,1,0,1),(1,1,1,0)\}.$$

■

1.1. Úplná normálna disjunktívna a konjunktívna forma.

Definícia 2.7. ***Elementárnym súčinovým členom*** (***elementárnym súčinom***) n premenných $x_1, x_2, \dots, x_n$ nazývame B-výraz

$$S(x_1, x_2, \dots, x_n) = y_1 y_2 \dots y_n,$$

kde $y_i \in \{x_i, \overline{x}_i\}$, pre $i \in \{1, 2, \dots, n\}$.

Príklad 2.8. Elementárne súčinové členy 2 premenných sú 4. Sú to: $xy, \overline{x}y$, $x\overline{y}$, $\overline{x}\,\overline{y}$. Elementárnym súčinovým členom nie je $\overline{xy}$. Elementárnych súčinových členov 3 premenných je 8. Sú to: xyz, $\overline{x}yz$, $x\overline{y}z$, $xy\overline{z}$, $\overline{x}\,\overline{y}z$, $\overline{x}y\overline{z}$, $x\overline{y}\,\overline{z}$, $\overline{x}\,\overline{y}\,\overline{z}$. ■

Veta 2.7. Jednotkové body každých dvoch rôznych elementárnych súčinových členov sú rôzne.

Dôkaz. Nech

$$\begin{aligned} U(x_1, x_2, \dots, x_n) &= u_1 u_2 \dots u_n, \qquad u_i \in \{x_i, \overline{x}_i\}, \\ V(x_1, x_2, \dots, x_n) &= v_1 v_2 \dots v_n, \qquad v_i \in \{x_i, \overline{x}_i\} \end{aligned}$$

sú rôzne elementárne súčinové členy. Keďže sú rôzne, tak pre niektoré $k \in \{1, 2, \dots, n\}$ $u_k = x_k$, $v_k = \overline{x}_k$ alebo $u_k = \overline{x}_k$, $v_i = x_k$. Bez ujmy na všeobecnosti môžeme predpokladať, že nastal prvý prípad. Potom pre k-té zložky β_k, γ_k jednotkových bodov členov v poradí U, V platí $\beta_k = 1, \gamma_k = 0$. To však znamená, že tieto jednotkové body sú rôzne. □

Je zrejmé, že počet rôznych elementárnych súčinových členov s n premennými je 2^n. Každý elementárny súčinový člen má práve jeden jednotkový bod, ktorý ho teda jednoznačne charakterizuje.

Každej n-tici $(j_1, j_2, \dots, j_n) \in \{0,1\}^n$ môžeme jednoznačne priradiť nezáporné celé číslo $j = j_1.2^{n-1} + j_2.2^{n-2} + \cdots + j_{n-1}.2^1 + j_n.2^0$, $0 \leq j \leq 2^n - 1$. Ale aj naopak každému celému číslu j, $0 \leq j \leq 2^n - 1$ je jednoznačne priradená n-tica $(j_1, j_2, \dots, j_n)$ taká, že $j = j_1.2^{n-1} + j_2.2^{n-2} + \cdots + j_{n-1}.2^1 + j_n.2^0$.

Definícia 2.8. Nech $(j_1, j_2, \dots, j_n) \in \mathbf{B}^n$ je jednotkový bod elementárneho súčinového člena a nech $j = j_1.2^{n-1} + j_2.2^{n-2} + \cdots + j_{n-1}.2^1 + j_n.2^0$. Potom tento súčinový člen budeme označovať $S_{(j_1, j_2, \dots, j_n)}(x_1, x_2, \dots, x_n)$ alebo $S_j(x_1, x_2, \dots, x_n)$ a nazývať ho ***elementárny súčinový člen priradený k** n**-tici*** $(j_1, j_2, \dots, j_n)$.

Príklad 2.9. Ako označujeme elementárny súčinový člen $x\overline{y}z$? Ktoré elementárne súčinové členy majú označenie $S_6(x,y,z,u)$ a $S_{15}(x,y,z,u)$?

Riešenie. Jednotkový bod elementárneho súčinového člena $x\overline{y}z$ je $(1,0,1)$. Potom $j = 1.2^2 + 0.2 + 1.2^0 = 5$, a teda $x\overline{y}z = S_{(1,0,1)}(x,y,z) = S_5(x,y,z)$.

$6 = 0.2^3 + 1.2^2 + 1.2 + 0.2^0$, preto jednotkovým bodom elementárneho súčinového člena $S_6(x,y,z,u)$ je $(0,1,1,0)$ a teda $S_6(x,y,z,u) = \overline{x}yz\overline{u}$.

$15 = 1.2^3 + 1.2^2 + 1.2 + 1.2^0$, preto $(1,1,1,1)$ je jednotkový bod člena $S_{15}(x,y,z,u)$, a teda $S_{15}(x,y,z,u) = xyzu$.

Definícia 2.9. Každý B-výraz

$$U\left(x_1, x_2, \dots, x_n\right) = \sum_{j=0}^{2^n-1} c_j S_j\left(x_1, x_2, \dots, x_n\right), \text{ kde } c_j \in \{0,1\}$$

nazývame ***úplná normálna disjunktívna forma***, v skratke UNDF.

V zápise UNDF (na základe dohody o zjednodušení písania B-výrazov) budeme súčinové členy, pri ktorých je nulový koeficient vynechávať, a nebudeme písať koeficienty $c_k = 1$. Teda napríklad $1.xy + 0.\overline{x}y + 1.x\overline{y} + 0.\overline{x}\,\overline{y}$ budeme písať v tvare $xy + x\overline{y}$.

Príklad 2.10. UNDF dvoch premenných sú napr. xy, $xy + \overline{x}y$, $\overline{x}y + x\overline{y} + \overline{x}\,\overline{y}$. UNDF troch premenných sú napr. $x\overline{y}z$, $xyz + \overline{x}y\overline{z} + x\overline{y}z$, $\overline{x}\,\overline{y}z + \overline{x}\,\overline{y}\,\overline{z}, 0$. ■

Veta 2.8. Každé dve rôzne UNDF určujú dve rôzne booleovské funkcie.

Dôkaz. Nech

$$\begin{aligned} U(x_1, x_2, \dots, x_n) &= \sum_{j=0}^{2^n-1} c_j S_j(x_1, x_2, \dots, x_n), \ c_j \in \mathbf{B}, \\ V(x_1, x_2, \dots, x_n) &= \sum_{j=0}^{2^n-1} d_j S_j(x_1, x_2, \dots, x_n), \ d_j \in \mathbf{B} \end{aligned}$$

sú dve rôzne UNDF. Potom existuje $k \in \{0, 1, \ldots, 2^n - 1\}$, že $c_k \neq d_k$. Nech $(\alpha_1, \alpha_2, \ldots, \alpha_n)$ je jednotkový bod elementárneho súčinového člena $S_k(x_1, x_2, \ldots, x_n)$. Potom pre funkcie určené B-výrazmi U, V platí

$$f_U(\alpha_1, \alpha_2, \ldots, \alpha_n) = \sum_{j=0}^{2^n-1} c_j S_j(\alpha_1, \alpha_2, \ldots, \alpha_n) = c_k S_k(\alpha_1, \alpha_2, \ldots, \alpha_n) = c_k,$$
$$f_V(\alpha_1, \alpha_2, \ldots, \alpha_n) = \sum_{j=0}^{2^n-1} d_j S_j(\alpha_1, \alpha_2, \ldots, \alpha_n) = d_k S_k(\alpha_1, \alpha_2, \ldots, \alpha_n) = d_k.$$

Keďže $c_k \neq d_k$, tak z toho vyplýva $f_U \neq f_V$. □

Veta 2.9. Každá booleovská funkcia je určená B-výrazom.

Dôkaz. Úplná normálna disjunktívna forma n premenných je súčtom 2^n elementárnych súčinových členov násobených konštantami, ktoré môžu mať hodnotu 0 alebo 1. Z toho vyplýva, že UNDF n premenných je 2^{2^n}. Presne toľko je aj booleovských funkcií n premenných. Preto každá booleovská funkcia je určená niektorou UNDF. □

Definícia 2.10. Nech $U(x_1, x_2, \ldots, x_n)$ je UNDF a $W(x_1, x_2, \ldots, x_n)$ je B-výraz, taký že $W \cong U$. Potom U sa nazýva ***úplná normálna disjunktívna forma B-výrazu*** W, tiež ***úplná normálna disjunktívna forma funkcie*** f_W a píšeme

$$\mathrm{UNDF}(W) = U, \quad \mathrm{UNDF}(f_W) = U.$$

Veta 2.10. Nech $g : \mathbf{B}^n \to \mathbf{B}$ je booleovská funkcia n premenných. Potom

$$\mathrm{UNDF}(g) = \sum_{(a_1,a_2,\ldots,a_n)\in\mathbf{B}^n} g(a_1, a_2, \ldots, a_n) S_{(a_1,a_2,\ldots,a_n)}(x_1, x_2, \ldots, x_n) =$$
$$= \begin{cases} 0, & \text{ak } g \text{ nemá jednotkový bod} \\ \displaystyle\sum_{(a_1,a_2,\ldots,a_n)\in J(g)} S_{(a_1,a_2,\ldots,a_n)}(x_1, x_2, \ldots, x_n), & \text{ak } g \text{ má jednotkový bod} \end{cases}$$

Dôkaz. Stačí ukázať, že hodnota funkcie určenej uvedeným B-výrazom v ľubovoľnom bode $(b_1, b_2, \ldots, b_n)$ sa rovná $g(b_1, b_2, \ldots, b_n)$. Ak si uvedomíme, že pre elementárne súčinové členy platí

$$S_{(a_1,a_2,\ldots,a_n)}(b_1, b_2, \ldots, b_n) = 1 \Leftrightarrow (a_1, a_2, \ldots, a_n) = (b_1, b_2, \ldots, b_n),$$

tak

$$\sum_{(a_1,\ldots,a_n)\in\mathbf{B}^n} g(a_1, \ldots, a_n) S_{(a_1,\ldots,a_n)}(b_1, \ldots, b_n) =$$
$$= g(b_1, \ldots, b_n) S_{(b_1,\ldots,b_n)}(b_1, \ldots, b_n) + \sum_{\substack{(a_1,\ldots,a_n)\in\mathbf{B}^n \\ (a_1,\ldots,a_n)\neq(b_1,\ldots,b_n)}} g(a_1, \ldots, a_n) S_{(a_1,\ldots,a_n)}(b_1, \ldots, b_n) =$$
$$= g(b_1, \ldots, b_n) \cdot 1 + \sum_{\substack{(a_1,\ldots,a_n)\in\mathbf{B}^n \\ (a_1,\ldots,a_n)\neq(b_1,\ldots,b_n)}} g(a_1, \ldots, a_n) \cdot 0 = g(b_1, \ldots, b_n).$$
□

Príklad 2.11. Nájdite úplnú normálnu disjunktívnu formu B-výrazu

$$W(x, y, z) = \overline{\overline{x}y + \overline{z}} + \overline{x}y\overline{z}.$$

Riešenie. Keďže $\mathrm{UNDF}(W) = \mathrm{UNDF}(f_W)$, zostavíme tabuľku funkcie f_W, ktorá je určená B-výrazom W a ku každému jednotkovému bodu funkcie f_W napíšeme k nemu priradený elementárny súčinový člen.

$x\ y\ z$	$f_W(x,y,z)$	S_j
0 0 0	0	
0 0 1	1	$\overline{x}\,\overline{y}z$
0 1 0	1	$\overline{x}y\overline{z}$
0 1 1	0	
1 0 0	0	
1 0 1	1	$x\overline{y}z$
1 1 0	0	
1 1 1	1	xyz

Hľadaná UNDF je súčtom týchto elementárnych súčinových členov:

$$\mathrm{UNDF}(W) = \mathrm{UNDF}(f_W) = \overline{x}\,\overline{y}z + \overline{x}y\overline{z} + x\overline{y}z + xyz.$$

■

Uvedieme ešte algoritmus, ktorým nájdeme UNDF B-výrazu $W(x_1, x_2, \ldots, x_n)$ bez toho, aby sme konštruovali tabuľku funkcie f_W. Použijeme pritom ekvivalencie z vety 2.4.

1. Pomocou de Morganových zákonov zostrojíme k B-výrazu $W(x_1, x_2, \ldots, x_n)$ ekvivalentný B-výraz $W_1(x_1, x_2, \ldots, x_n)$, v ktorom nie je znak negácie (pruh) nad znakom operácií $+$ a $\cdot$.

2. Pomocou distributívnych zákonov odstránime v B-výraze $W_1(x_1, x_2, \ldots, x_n)$ zátvorky. Dostaneme B-výraz $W_2(x_1, x_2, \ldots, x_n)$ ekvivalentný s $W_1(x_1, x_2, \ldots, x_n)$, pričom $W_2(x_1, x_2, \ldots, x_n)$ bude v tvare „súčtu súčinov“.

3. Pomocou zákonov involúcie, komutatívnosti, asociatívnosti, idempotentnosti, komplementárnosti, identity, nulového násobenia a jednotkového sčitovania každý sčítanec upravíme na tvar $y_1, y_2, \ldots, y_n$, kde $y_i \in \{x_i, \overline{x}_i, 1\}$ pre $i = 1, 2, \ldots, n$. Pri úpravách môžeme, ale nemusíme používať zákony absorpcie. Získame tak B-výraz $W_3(x_1, x_2, \ldots, x_n) \cong W_2(x_1, x_2, \ldots, x_n)$.

4. V každom sčítanci B-výrazu $W_3(x_1, x_2, \ldots, x_n)$ nahradíme $y_i = 1$ ekvivalentným B-výrazom $x_i + \overline{x}_i$. Dostaneme B-výraz $W_4(x_1, x_2, \ldots, x_n) \cong W_3(x_1, x_2, \ldots, x_n)$.

5. Pomocou distributívneho zákona odstránime v B-výraze $W_4(x_1, x_2, \ldots, x_n)$ zátvorky. Dostaneme B-výraz $W_5(x_1, x_2, \ldots, x_n)$, v ktorom každý sčítanec je elementárny súčinový člen.

6. Pomocou zákona idempotentnosti zlúčime rovnaké elementárne súčinové členy. Získaný B-výraz $W_6(x_1, x_2, \ldots, x_n)$ je už $\mathrm{UNDF}(W)$.

Príklad 2.12. Pomocou ekvivalentných úprav nájdite UNDF B-výrazu

$$W(x,y,z) = \overline{\overline{x}y + \overline{z}} + \overline{x}y\overline{z}.$$

Riešenie. $W(x,y,z) = \overline{\overline{x}y + \overline{z}} + \overline{x}y\overline{z} \cong \overline{\overline{x}y}\,\overline{\overline{z}} + \overline{x}y\overline{z} \cong (\overline{\overline{x}} + \overline{y})\overline{\overline{z}} + \overline{x}y\overline{z} = W_1(x,y,z)$

$W_1(x,y,z) \cong \overline{\overline{x}}\,\overline{\overline{z}} + \overline{y}\,\overline{\overline{z}} + \overline{x}y\overline{z} = W_2(x,y,z)$

$W_2(x,y,z) \cong xz + \overline{y}z + \overline{x}y\overline{z} \cong x \cdot 1 \cdot z + 1 \cdot \overline{y}z + \overline{x}y\overline{z} = W_3(x,y,z)$

$W_3(x,y,z) \cong x(y + \overline{y})z + (x + \overline{x})\overline{y}z + \overline{x}y\overline{z} = W_4(x,y,z)$

$W_4(x,y,z) \cong xyz + x\overline{y}z + x\overline{y}z + \overline{x}\,\overline{y}z + \overline{x}y\overline{z} = W_5(x,y,z)$

$W_5(x,y,z) \cong xyz + x\overline{y}z + \overline{x}\,\overline{y}z + \overline{x}y\overline{z} = \mathrm{UNDF}(W)$ ■

Duálnymi pojmami k pojmom elementárny súčinový člen a úplná normálna disjunktívna forma sú elementárny súčtový člen a úplná normálna konjunktívna forma. Stručne uvedieme ich definície a vlastnosti.

Definícia 2.11. ***Elementárnym súčtovým členom*** (***elementárnym súčtom***) n premenných $x_1, x_2, \ldots, x_n$, nazývame B-výraz

$$T(x_1, x_2, \ldots, x_n) = y_1 + y_2 + \cdots + y_n,$$

kde $y_i \in \{x_i, \overline{x}_i\}$, pre $i \in \{1, \ldots, n\}$.

Príklad 2.13. Elementárne súčtové členy 2 premenných sú 4. Sú to: $x + y$, $\overline{x} + y$, $x+\overline{y}$, $\overline{x}+\overline{y}$. Elementárne súčtové členy troch premenných sú $x+y+z$, $\overline{x}+y+z$, $x+\overline{y}+z$, $x + y + \overline{z}$, $\overline{x} + \overline{y} + z$, $\overline{x} + y + \overline{z}$, $x + \overline{y} + \overline{z}$, $\overline{x} + \overline{y} + \overline{z}$. ■

Počet elementárnych súčtových členov n premenných je 2^n.

Definícia 2.12. Nech $(\nu_1, \nu_2, \ldots, \nu_n)$ je nulový bod elementárneho súčtového člena a nech $\nu = \nu_1.2^{n-1} + \nu_2.2^{n-2} + \ldots + \nu_n.2^0$. Potom tento súčtový člen budeme označovať $T_{(\nu_1,\nu_2,\ldots,\nu_n)}(x_1, x_2, \ldots, x_n)$ alebo $T_\nu(x_1, x_2, \ldots, x_n)$ a nazývať ho ***elementárny súčtový člen priradený k n-tici*** $(\nu_1, \nu_2, \ldots, \nu_n)$.

Definícia 2.13. Každý B-výraz

$$V\,(x_1, x_2, \ldots, x_n) = \prod_{\nu=0}^{2^n-1} (c_\nu + T_\nu\,(x_1, x_2, \ldots, x_n)),\ \text{kde } c_\nu \in \{0, 1\}$$

nazývame ***úplnou normálnou konjunktívnou formou***, v skratke UNKF.

Nech $W(x_1, x_2, \ldots, x_n)$ je B-výraz, taký že $W \cong V$. Potom V sa nazýva ***úplná normálna konjunktívna forma B-výrazu*** W, tiež ***úplná normálna konjunktívna forma funkcie*** f_W a píšeme

$$\mathrm{UNKF}(W) = V, \quad \mathrm{UNKF}(f_W) = V.$$

Príklad 2.14. UNKF dvoch premenných sú napr.

$$x + y,\ (x + y)\,(\overline{x} + y)\,(\overline{x} + \overline{y})\,,\ (x + y)\,(x + \overline{y})\,, 1.$$

UNKF troch premenných sú napr.

$$(x + y + z)\,(x + \overline{y} + z)\,,\ (\overline{x} + y + z)\,(x + \overline{y} + \overline{z})\,(x + y + \overline{z})\,(\overline{x} + \overline{y} + z)\,(\overline{x} + y + \overline{z})\,.$$ ■

Veta 2.11. Nech $g : \mathbf{B}^n \to \mathbf{B}$ je booleovská funkcia n premenných. Potom

$$\mathrm{UNKF}(g) = \prod_{(a_1,a_2,\ldots,a_n)\in\mathbf{B}^n} \big(g(a_1, a_2, \ldots, a_n) + T_{(a_1,a_2,\ldots,a_n)}(x_1, x_2, \ldots, x_n)\big) =$$

$$= \begin{cases} 1, & \text{ak } g \text{ nemá nulový bod} \\ \displaystyle\prod_{(a_1,a_2,\ldots,a_n)\in N(g)} T_{(a_1,a_2,\ldots,a_n)}(x_1, x_2, \ldots, x_n), & \text{ak } g \text{ má nulový bod} \end{cases}$$

□

Príklad 2.15. Nájdite UNKF funkcie $g(x, y, z) = \overline{\overline{x}y + \overline{z}} + \overline{x}y\overline{z}$.

Riešenie. Zostavíme tabuľku hodnôt funkcie g. Do ďalšieho stĺpca tabuľky zapíšeme priradené elementárne súčtové členy k nulovým bodom funkcie.

$x\ y\ z$	$g(x,y,z)$	T_ν
0 0 0	0	$x+y+z$
0 0 1	1	
0 1 0	1	
0 1 1	0	$x+\overline{y}+\overline{z}$
1 0 0	0	$\overline{x}+y+z$
1 0 1	1	
1 1 0	0	$\overline{x}+\overline{y}+z$
1 1 1	1	

UNKF funkcie g je súčinom týchto členov:

$$\mathrm{UNKF}(g) = (x+y+z)(x+\overline{y}+\overline{z})(\overline{x}+y+z)(\overline{x}+\overline{y}+z).$$

Uvedieme ešte druhý spôsob – ekvivalentné úpravy B-výrazu, ktorým je funkcia g určená. Algoritmus je podobný ako pre UNDF, akurát používame duálne postupy:

$$\overline{\overline{x}y+\overline{z}}+\overline{x}y\overline{z} \cong \overline{\overline{x}y}\,\overline{\overline{z}}+\overline{x}y\overline{z} \cong (\overline{\overline{x}}+\overline{y})\overline{\overline{z}}+\overline{x}y\overline{z} \cong (x+\overline{y})z+\overline{x}y\overline{z} \cong$$
$$\cong ((x+\overline{y})z+\overline{x})((x+\overline{y})z+y)((x+\overline{y})z+\overline{z}) \cong$$
$$\cong (x+\overline{y}+\overline{x})(z+\overline{x})(x+\overline{y}+y)(z+y)(x+\overline{y}+\overline{z})(z+\overline{z}) \cong 1\cdot(\overline{x}+z)\cdot 1\cdot(y+z)(x+\overline{y}+\overline{z}) \cong$$
$$\cong (\overline{x}+z)(y+z)(x+\overline{y}+\overline{z}) \cong (\overline{x}+0+z)(0+y+z)(x+\overline{y}+\overline{z}) \cong$$
$$\cong (\overline{x}+y\overline{y}+z)(x\overline{x}+y+z)(x+\overline{y}+\overline{z}) \cong$$
$$\cong (\overline{x}+y+z)(\overline{x}+\overline{y}+z)(x+y+z)(\overline{x}+y+z)(x+\overline{y}+\overline{z}) \cong$$
$$\cong (\overline{x}+y+z)(\overline{x}+\overline{y}+z)(x+y+z)(x+\overline{y}+\overline{z}) = \mathrm{UNKF}(g)$$ ■

1.2. Normálna disjunktívna a konjunktívna forma.

V tejto časti sa budeme zaoberať zovšeobecnením elementárnych súčinových (súčtových) členov a úplnej normálnej disjunktívnej (konjunktívnej) formy. Elementárny súčinový člen sme definovali ako B-výraz

$$M(x_1,x_2,\ldots,x_n) = y_1y_2\ldots y_n, \text{ kde } y_i \in \{x_i,\overline{x}_i\}.$$

Nás budú teraz zaujímať podobné B-výrazy, len v nich niektoré y_i môžeme nahradiť aj hodnotou 1 (túto jednotku na základe zákona o jednotkovom násobení spravidla do výrazu nebudeme písať; takýto výraz vznikne z elementárneho súčinového člena vynechaním niektorých premenných).

Definícia 2.14. ***Súčinovým členom*** n premenných $x_1,\ldots,x_n$ nazývame B-výraz $S(x_1,x_2,\ldots,x_n) = y_1y_2\ldots y_n$, kde $y_i \in \{x_i,\overline{x}_i,1\}$ (ak $y_i = 1$ môžeme túto jednotku v zápise súčinového člena vynechať).

Príklad 2.16. Súčinových členov 2 premenných x,y je 9. Sú to tieto súčinové členy: 1, x, $\overline{x}$, y, $\overline{y}$, xy, $\overline{x}y$, $x\overline{y}$, $\overline{x}\,\overline{y}$. ■

Je zrejmé, že počet rôznych súčinových členov n premenných $x_1,x_2,\ldots,x_n$ je práve 3^n. Môžeme ich teda očíslovať číslami od 1 po 3^n. Predpokladajme, že sme jedno pevné očíslovanie $S_1(x_1,x_2,\ldots,x_n),\ldots,S_{3^n}(x_1,x_2,\ldots,x_n)$ už zvolili.

DEFINÍCIA 2.15. Každý B-výraz

$$U(x_1, x_2, \dots, x_n) = \sum_{k=1}^{3^n} c_k S_k(x_1, x_2, \dots, x_n), \text{ kde } c_k \in \mathbf{B},$$

nazývame ***normálna disjunktívna forma***, v skratke NDF.

Ak $V(x_1, x_2, \dots, x_n)$ je B-výraz a $U(x_1, x_2, \dots, x_n)$ je NDF, ktorá je ekvivalentná s $V(x_1, x_2, \dots, x_n)$, budeme hovoriť, že $U(x_1, x_2, \dots, x_n)$ je ***normálna disjunktívna forma B-výrazu*** $V(x_1, x_2, \dots, x_n)$ a písať $\mathrm{NDF}(V) = U$.

Ak booleovská funkcia $f : \mathbf{B}^n \to \mathbf{B}$ je určená normálnou disjunktívnou formou $U(x_1, x_2, \dots, x_n)$, tak $U(x_1, x_2, \dots, x_n)$ budeme nazývať ***normálna disjunktívna forma booleovskej funkcie*** $f : \mathbf{B}^n \to \mathbf{B}$ a budeme písať $\mathrm{NDF}(f) = U$.

V zápise normálnej disjunktívnej formy budeme súčinové členy, pri ktorých stojí nulový koeficient, vynechávať a nebudeme písať jednotkové koeficienty. Teda napríklad $0 \,.\, 1 + 0 \,.\, x + 1 \,.\, \overline{x} + 0 \,.\, y + 0 \,.\, \overline{y} + 1 \,.\, xy + 0 \,.\, \overline{x}y + 0 \,.\, x\overline{y} + 0 \,.\, \overline{x}\,\overline{y}$ budeme stručnejšie písať v tvare $\overline{x} + xy$. Všeobecne, ak

$$U(x_1, x_2, \dots, x_n) = \sum_{k=1}^{3^n} c_k S_k(x_1, x_2, \dots, x_n)$$

je NDF a m je počet tých koeficientov c_k, ktoré sa rovnajú 1, tak tie súčinové členy, pri ktorých stoja jednotkové koeficienty, teraz nanovo očíslujeme od 1 až po m a danú NDF budeme písať v tvare

$$U(x_1, x_2, \dots, x_n) = \sum_{k=1}^{m} S_k(x_1, x_2, \dots, x_n)$$

Pretože každý elementárny súčinový člen je aj súčinovým členom, je UNDF B-výrazu aj NDF tohto B-výrazu. Je preto pravdivé tvrdenie, že každú booleovskú funkciu môžeme určiť pomocou NDF. Táto reprezentácia, prirodzene, už nie je jednoznačná, lebo napríklad pre každé $(x, y) \in \mathbf{B}^2$ je $f(x, y) = \overline{x}y + x = y + x$.

DEFINÍCIA 2.16. ***Súčtovým členom*** n premenných $x_1, x_2, \dots, x_n$ nazývame B-výraz $T(x_1, x_2, \dots, x_n) = y_1 + y_2 + \ldots + y_n$, kde $y_i \in \{x_i, \overline{x}_i, 0\}$ pre $i \in \{1, \dots, n\}$.

PRÍKLAD 2.17. Súčtových členov 2 premenných x, y je 9. Sú to tieto súčtové členy: 0, x, $\overline{x}$, y, $\overline{y}$, $x + y$, $\overline{x} + y$, $x + \overline{y}$, $\overline{x} + \overline{y}$. ■

Podobne, ako súčinových, tak aj súčtových členov n premenných je práve 3^n. Predpokladajme, že sme tieto súčtové členy pevne očíslovali:

$$T_1(x_1, x_2, \dots x_n), \dots, T_{3^n}(x_1, x_2, \dots, x_n).$$

Potom, podobne ako sme definovali NDF, môžeme definovať aj normálnu konjunktívnu formu.

DEFINÍCIA 2.17. Každý B-výraz

$$U(x_1, x_2, \dots, x_n) = \prod_{k=1}^{3^n} (c_k + T_k(x_1, x_2, \dots, x_n)), \text{ kde } c_k \in \mathbf{B},$$

nazývame ***normálna konjunktívna forma***, skrátene NKF.

Analogicky, ako pre NDF, sa definujú pojmy ***normálna konjunktívna forma B-výrazu*** a ***normálna konjunktívna forma booleovskej funkcie***. Opäť platí, že každá booleovská funkcia sa dá určiť pomocou normálnej konjunktívnej formy.

Tak ako pre NDF aj pre NKF budeme používať skrátený zápis: Nech m je počet koeficientov c_k, ktoré sa rovnajú 0. Tie súčtové členy, pri ktorých sú tieto nulové koeficienty, nanovo očíslujeme od 1 až po m a danú NKF budeme písať v tvare

$$U(x_1, x_2, \ldots, x_n) = \prod_{k=1}^{m} T_k(x_1, x_2, \ldots, x_n).$$

PRÍKLAD 2.18. Ukážte, že $U(x,y,z) = \overline{x}\,\overline{z} + xy + \overline{y}z$, $V(x,y,z) = \overline{x}\,\overline{y} + y\overline{z} + xz$ sú NDF tej istej booleovskej funkcie.

Riešenie. Stačí ukázať, že B-výrazy U a V sú ekvivalentné:

$$U(x,y,z) = \overline{x}\,\overline{z} + xy + \overline{y}z \cong \overline{x}(y+\overline{y})\overline{z} + xy(z+\overline{z}) + (x+\overline{x})\overline{y}z \cong$$

$$\cong \overset{3.}{\overline{x}y\overline{z}} + \overset{1.}{\overline{x}\,\overline{y}\,\overline{z}} + \overset{5.}{xyz} + \overset{4.}{xy\overline{z}} + \overset{6.}{x\overline{y}z} + \overset{2.}{\overline{x}\,\overline{y}z} \cong (\overline{x}\,\overline{y}\,\overline{z} + \overline{x}\,\overline{y}z) + (\overline{x}y\overline{z} + xy\overline{z}) + (xyz + x\overline{y}z) \cong$$

$$\cong \overline{x}\,\overline{y}(\overline{z}+z) + (\overline{x}+x)y\overline{z} + x(y+\overline{y})z \cong \overline{x}\,\overline{y} + y\overline{z} + xz = V(x,y,z)$$

■

1.3. Normálne disjunktívne a konjunktívne formy výrokových formúl.

Pravdivostné ohodnotenie každej formuly výrokového počtu je booleovská funkcia. Špeciálne vieme, že

$$\mathrm{ph}_{p\vee q}(x,y) = x+y,\ \ \mathrm{ph}_{p\wedge q}(x,y) = xy,\ \ \mathrm{ph}_{\overline{p}}(x) = \overline{x}.$$

Keďže množina $\{\vee, \wedge, {}^{-}\}$ je úplným systémom logických spojok, môžeme ľubovoľnú výrokovú formulu n premenných $p_1, p_2, \ldots, p_n$ vyjadriť len pomocou týchto logických spojok. Ak v tejto formule prepíšeme $\vee$ na $+$, $\wedge$ na $\cdot$ a p_i na x_i pre $i \in \{1, 2, \ldots, n\}$, zrejme dostaneme B-výraz, ktorý reprezentuje pravdivostné ohodnotenie tejto formuly. Z druhej strany, každú booleovskú funkciu $f(x_1, x_2, \ldots, x_n)$ je možné reprezentovať B-výrazom, ktorý neobsahuje 0 a 1. Ak v tomto B-výraze prepíšeme $+$ na $\vee$, $\cdot$ na $\wedge$ a x_i na p_i pre $i \in \{1, 2, \ldots, n\}$, dostaneme výrokovú formulu, ktorej pravdivostným ohodnotením je funkcia f. Z týchto úvah tiež vyplýva:

VETA 2.12. Každá booleovská funkcia je pravdivostným ohodnotením niektorej výrokovej formuly. □

PRÍKLAD 2.19. Určte pravdivostné ohodnotenie formuly $a(p,q,r) = ((p \Rightarrow r) \wedge q) \Rightarrow r$ pomocou B-výrazu.

Riešenie. Najprv nájdeme k formule $a(p,q,r)$ ekvivalentnú formulu, ktorá z logických spojok obsahuje len $\vee, \wedge, {}^{-}$.

$$a(p,q,r) \sim ((\overline{p} \vee r) \wedge q) \Rightarrow r \sim \overline{((\overline{p} \vee r) \wedge q)} \vee r$$

Teraz môžeme pravdivostné ohodnotenie formuly a vyjadriť takto (výrokové premenné p, q, r nahradíme v uvedenom poradí premennými x, y, z):

$$\mathrm{ph}_{a(p,q,r)}(x,y,z) = \overline{((\overline{x}+z)y)} + z = x\overline{z} + \overline{y} + z$$

■

PRÍKLAD 2.20. Napíšte formulu $a(p,q,r,s)$, ktorej pravdivostné ohodnotenie je reprezentované B-výrazom $U(x,y,z,u) = 1\cdot x\overline{y}u + 0\cdot yzu + 1\cdot xy\overline{z}u$.

Riešenie. Keďže $U(x,y,z,u) \cong (x\overline{y}u) + (xy\overline{z}u)$, hľadaná formula je

$$a(p,q,r,s) = (p \wedge \overline{q} \wedge s) \vee (p \wedge q \wedge \overline{r} \wedge s)$$

a tiež ktorákoľvek s ňou ekvivalentná formula. ■

Normálna disjunktívna a konjunktívna forma výrokových formúl je analógiou k pojmu normálna disjunktívna a konjunktívna forma B-výrazov. Vzhľadom na vyššie spomenutú príbuznosť B-výrazov a výrokových formúl môžeme normálnu disjunktívnu a konjunktívnu formu výrokových formúl definovať pomocou B-výrazov.

Najprv sa dohodnime na nasledujúcom označení. Nech $U(x_1,x_2,\dots,x_n)$ je B-výraz, v ktorom sa nevyskytujú znaky 0, 1. Výrokovú formulu, ktorú z neho dostaneme prepisom znaku + na $\vee$, znaku $\cdot$ na $\wedge$ a zámenou booleovských premenných $x_1,x_2,\dots,x_n$ na výrokové premenné $p_1,p_2,\dots,p_n$, budeme označovať $a_U(p_1,p_2,\dots,p_n)$.

DEFINÍCIA 2.18. Nech B-výraz $U(x_1,x_2,\dots,x_n)$ je NDF (resp. NKF), pričom sa v nej nevyskytujú znaky 0 a 1. Potom výrokovú formulu $a_U(p_1,p_2,\dots,p_n)$ nazývame ***normálna disjunktívna*** (resp. ***konjunktívna***) ***forma*** každej výrokovej formuly b ekvivalentnej s formulou $a_U(p_1,p_2,\dots,p_n)$ a označujeme ju NDF(b) (resp. NKF(b)).

DEFINÍCIA 2.19. Nech $U(x_1,x_2,\dots,x_n)$ je elementárny súčet a $V(x_1,x_2,\dots,x_n)$ je elementárny súčin premenných $x_1,x_2,\dots,x_n$. Potom formulu $a_U(p_1,p_2,\dots,p_n)$ nazývame ***elementárnou disjunkciou*** premenných $p_1,p_2,\dots,p_n$ a formulu $a_V(p_1,p_2,\dots,p_n)$ ***elementárnou konjunkciou*** premenných $p_1,p_2,\dots,p_n$.

Nech B-výraz W je UNKF, $W \neq 1$. Potom formulu a_W nazývame ***úplnou normálnou konjunktívnou formou formuly*** b tautologicky ekvivalentnej s formulou a_W a označujeme ju UNKF(b).

Nech B-výraz W je UNDF, $W \neq 0$. Potom formulu a_W nazývame ***úplnou normálnou disjunktívnou formou formuly*** b tautologicky ekvivalentnej s formulou a_W a označujeme ju UNDF(b).

PRÍKLAD 2.21. Nájdite UNKF a UNDF formuly $p \Leftrightarrow q$.

Riešenie. Urobíme pravdivostnú tabuľku danej formuly (tab. 5). Pomocou nej nájdeme UNKF a UNDF B-výrazu. K nim prepisom priradíme UNKF resp. UNDF formuly.

Môžeme postupovať aj tak, že priamo z tabuľky priradíme elementárnym súčtom elementárne disjunkcie, resp. elementárnym súčinom elementárne konjunkcie. Tie potom spojíme konjunkciou, resp disjunkciou.

TABUĽKA 5. Pravdivostná tabuľka formuly $p \Leftrightarrow q$

p q	$p \Leftrightarrow q$		
0 0	1	$\overline{p} \wedge \overline{q}$ $(\overline{x}\,\overline{y})$	
0 1	0		$p \vee \overline{q}$ $(x + \overline{y})$
1 0	0		$\overline{p} \vee q$ $(\overline{x} + y)$
1 1	1	$p \wedge q$ (xy)	

UNDF$(p \Leftrightarrow q) = (\overline{p} \wedge \overline{q}) \vee (p \wedge q)$,
UNKF$(p \Leftrightarrow q) = (p \vee \overline{q}) \wedge (\overline{p} \vee q)$. ■

VETA 2.13. Nech b je formula. Ak existuje UNDF (b), tak $b \sim$ UNDF (b). Ak existuje UNKF (b), tak $b \sim$ UNKF (b).

Dôkaz. Formula b, aj UNDF (b) a UNKF (b), pokiaľ existujú, majú rovnaké pravdivostné ohodnotenie, teda sú tautologicky ekvivalentné.

Poznamenajme, že UNDF (b) neexistuje práve vtedy, keď b je kontradikcia a UNKF(b) neexistuje len vtedy, keď b je tautológia. □

2. Úplný systém booleovských funkcií

Z časti o UNDF a UNKF vieme, že každá booleovská funkcia $f : \mathbf{B}^n \to \mathbf{B}$ sa dá vyjadriť pomocou logického súčtu $g_1 : B^2 \to B,\ g_1(x,y) = x + y$, logického súčinu $g_2 : B^2 \to B,\ g_2(x,y) = xy$ a negácie $g_3 : B \to B,\ g_3(x) = \overline{x}$. Presnejšie, funkcia f sa dá vyjadriť ako zložená funkcia z funkcií g_1, g_2, g_3.

PRÍKLAD 2.22. Vyjadrite funkciu $f(x,y,z) = \overline{x + \overline{y} + yz}$ ako zloženú funkciu z logického súčtu $g_1(x,y)$, logického súčinu $g_2(x,y)$ a negácie $g_3(x)$.

Riešenie. $f(x,y,z) = \overline{(x+\overline{y}) + (yz)} = \overline{g_1(x,\overline{y}) + g_2(y,z)} = \overline{g_1(g_1(x, g_3(y)), g_2(y,z))} =$
$= g_3(g_1(g_1(x, g_3(y)), g_2(y,z)))$ ■

Predošlé úvahy nás vedú k nasledujúcej definícii.

DEFINÍCIA 2.20. Množina $\{f_1, f_2, \ldots, f_k\}$ booleovských funkcií sa nazýva ***úplný systém booleovských funkcií*** (stručne USBF), ak každá booleovská funkcia f sa dá vyjadriť ako zložená funkcia z funkcií $f_1, f_2, \ldots, f_k$.

VETA 2.14. Nech $\mathcal{Q}$ je USBF a $\mathcal{R}$ je ľubovoľná množina booleovských funkcií. Ak sa dá každá funkcia $f \in \mathcal{Q}$ vyjadriť ako zložená funkcia z funkcií množiny $\mathcal{R}$, potom aj množina $\mathcal{R}$ je USBF.

Dôkaz. Ľubovoľnú booleovskú funkciu vieme napísať ako zloženú funkciu z funkcií množiny $\mathcal{Q}$, lebo táto množina je USBF. Ak sem za funkcie množiny $\mathcal{Q}$ dosadíme ich vyjadrenia pomocou funkcií množiny $\mathcal{R}$, dostaneme funkciu f zapísanú ako zloženú funkciu len z funkcií množiny $\mathcal{R}$. □

PRÍKLAD 2.23. Dokážte, že množina $\mathcal{P}_2 = \{h_1\}$, kde $h_1 : \mathbf{B}^2 \to \mathbf{B},\ h_1(x,y) = \overline{x+y}$ je USBF.

Riešenie. Podľa predchádzajúcej vety stačí dokázať, že logický súčet, súčin a negácia sa dajú vyjadriť len pomocou funkcie h_1.

$$\begin{aligned}
\overline{x} &= \overline{x+x} = h_1(x,x),\\
x+y &= \overline{\overline{x+y}} = \overline{\overline{x+y} + \overline{x+y}} = h_1(h_1(x,y), h_1(x,y)),\\
xy &= \overline{\overline{xy}} = \overline{\overline{x} + \overline{y}} = \overline{\overline{x+x} + \overline{y+y}} = h_1(h_1(x,x), h_1(y,y)).
\end{aligned}$$

■

DEFINÍCIA 2.21. Booleovskú funkciu $h_1(x,y) = \overline{x+y}$ nazývame ***Pierceovou funkciou***, a označujeme $h_1(x,y) = (x \downarrow y)$.
Booleovskú funkciu $h_2(x,y) = \overline{xy}$ nazývame ***Shefferovou funkciou***, a označujeme $h_2(x,y) = (x \uparrow y)$.

Ak použijeme označenia $\downarrow$, $\uparrow$ binárnych operácií definovaných Pierceovou a Shefferovou funkciou, tak

$$
\begin{aligned}
\overline{x} &= (x \downarrow x),\\
x+y &= ((x \downarrow y) \downarrow (x \downarrow y)),\\
xy &= ((x \downarrow x) \downarrow (y \downarrow y)).
\end{aligned}
$$

$$
\begin{aligned}
\overline{x} &= \overline{xx} = (x \uparrow x),\\
x+y &= \overline{\overline{x+y}} = \overline{\overline{x}\,\overline{y}} = \overline{\overline{xx}\,\overline{yy}} = ((x \uparrow x) \uparrow (y \uparrow y)),\\
xy &= \overline{\overline{xy}} = \overline{\overline{xy}\,\overline{xy}} = ((x \uparrow y) \uparrow (x \uparrow y)).
\end{aligned}
$$

Vidíme, že aj množina $\mathcal{S}_2 = \{h_2\}$ je USBF.

Každá booleovská funkcia $f : \mathbf{B}^n \to \mathbf{B}$ je n-árnou operáciou na množine $\mathbf{B}$. Pre zjednodušenie zápisu tejto operácie a zlepšenie jeho prehľadnosti používame namiesto $f(x_1, x_2, \ldots, x_n)$ iné, vhodnejšie označenie. Napríklad pre binárnu operáciu určenú Shefferovou funkciou používame znak $\uparrow$ a namiesto $h_2(x, y)$ píšeme $(x \uparrow y)$. Vo všeobecnosti budeme prvok $f(x_1, x_2, \ldots, x_n)$, ktorý je zapísaný pomocou znaku príslušnej operácie, označovať $(x_1, x_2, \ldots, x_n)_{\omega_f}$.

Nech $\mathcal{Q}$ je množina (navzájom rôznych) booleovských funkcií. Analogicky ako sme definovali B-výrazy budeme definovať $\mathcal{Q}$-výrazy.

Definícia 2.22. Nech $\mathcal{Q}$ je množina booleovských funkcií. Potom ***$\mathcal{Q}$-výraz*** n ***premenných*** $x_1, x_2, \ldots, x_n$ definujeme takto:

1. $x_1, x_2, \ldots, x_n, 0, 1$ sú $\mathcal{Q}$-výrazy n premenných.
2. Ak $U_1, U_2, \ldots, U_k$ sú $\mathcal{Q}$-výrazy n premenných a $(f : \mathbf{B}^k \to \mathbf{B}) \in \mathcal{Q}$, potom aj $(U_1, U_2, \ldots, U_k)_{\omega_f}$ je $\mathcal{Q}$-výraz n premenných.
3. Každý $\mathcal{Q}$-výraz n premenných $x_1, x_2, \ldots, x_n$ vznikne pomocou bodov 1 a 2.

Poznámka 2.6. Ak $\mathcal{Q}$ je USBF, tak každá booleovská funkcia sa dá vyjadriť pomocou $\mathcal{Q}$-výrazu.

Príklad 2.24. Nájdite $\mathcal{P}_2$ a $\mathcal{S}_2$-výrazy, ktoré reprezentujú funkciu $f(x, y, z) = \overline{x + yz} + z$.

Riešenie.

$$
\begin{aligned}
f(x,y,z) &= \overline{\overline{\overline{x + \overline{\overline{\overline{y}\,\overline{z}}}} + z}} = \overline{\overline{\overline{x + \overline{\overline{y} + \overline{z}}} + z}} = \overline{\overline{\overline{x + (\overline{y} \downarrow \overline{z})} + z}} = \overline{\overline{(x \downarrow (\overline{y} \downarrow \overline{z})) + z}} =\\
&= \overline{((x \downarrow (\overline{y} \downarrow \overline{z})) \downarrow z)} = (((x \downarrow (\overline{y} \downarrow \overline{z})) \downarrow z) \downarrow ((x \downarrow (\overline{y} \downarrow \overline{z})) \downarrow z)) =\\
&= (((x \downarrow ((y \downarrow y) \downarrow (z \downarrow z)) \downarrow z) \downarrow ((x \downarrow ((y \downarrow y) \downarrow (z \downarrow z))) \downarrow z)),
\end{aligned}
$$

$$
\begin{aligned}
f(x,y,z) &= \overline{x}\,\overline{yz} + z = \overline{\overline{\overline{x}\,\overline{yz} + z}} = \overline{\overline{\overline{x}\,\overline{yz}}\,\overline{z}} = \overline{\overline{(x \uparrow x)(y \uparrow z)}\,(z \uparrow z)} =\\
&= \overline{((x \uparrow x) \uparrow (y \uparrow z))\,(z \uparrow z)} = (((x \uparrow x) \uparrow (y \uparrow z)) \uparrow (z \uparrow z)).
\end{aligned}
$$

■

Dohoda.
1. V $\mathcal{P}_2$ a $\mathcal{S}_2$-výrazoch budeme výrazy $(U \downarrow U)$, $(U \uparrow U)$ skracovať na $(U\downarrow)$, $(U\uparrow)$.
2. V samostatne stojacich $\mathcal{P}_2$ a $\mathcal{S}_2$-výrazoch budeme vynechávať vonkajšie zátvorky.

Na základe tejto dohody, môžeme $\mathcal{P}_2$ a $\mathcal{S}_2$-výrazy určujúce funkciu f z predchádzajúceho príkladu napísať takto:

$$((x \downarrow ((y\downarrow) \downarrow (z\downarrow))) \downarrow z) \downarrow, \qquad ((x\uparrow) \uparrow (y \uparrow z)) \uparrow (z\uparrow).$$

Definícia 2.23. ***Pierceeovou funkciou*** n ***premenných***, $n \geq 2$, nazývame booleovskú funkciu

$$h_1^{(n)}(x_1, x_2, \ldots, x_n) = \overline{x_1 + x_2 + \ldots + x_n}$$

a budeme používať označenie $(x_1 \downarrow x_2 \downarrow \ldots \downarrow x_n)$.

Shefferovou funkciou n ***premenných***, $n \geq 2$, nazývame booleovskú funkciu

$$h_2^{(n)}(x_1, x_2, \ldots, x_n) = \overline{x_1 x_2 \ldots x_n}$$

a budeme používať označenie $(x_1 \uparrow x_2 \uparrow \ldots \uparrow x_n)$.

Veta 2.15. Každá z množín $\mathcal{P}_n = \{h_1^{(n)}\}$, $\mathcal{S}_n = \{h_2^{(n)}\}$ pre $n \geq 2$ je USBF.

Dôkaz. Stačí ukázať, že Pierceova funkcia $h_1(x, y) = x \downarrow y$ sa dá vyjadriť pomocou funkcie $h_1^{(n)}$ a Shefferova funkcia $h_2(x, y) = x \uparrow y$ zase pomocou funkcie $h_2^{(n)}$.

$$h_1(x, y) = \overline{x + y} = \overline{x + y + 0 + \ldots + 0} = (x \downarrow y \downarrow 0 \downarrow \ldots \downarrow 0)$$
$$h_2(x, y) = \overline{xy} = \overline{xy \cdot 1 \cdot \ldots \cdot 1} = (x \uparrow y \uparrow 1 \uparrow \ldots \uparrow 1)$$

□

Poznámka 2.7. Pierceovu a Shefferovu funkciu n premenných sme mohli definovať vďaka tomu, že binárne operácie $+$ a $\cdot$ sú asociatívne. Keby sme uvažovali o ternárnych operáciách definovaných funkciami $f_1 : \mathbf{B}^3 \to \mathbf{B}$, $f_1(x_1, x_2, x_3) = x_1 + (x_2 + x_3)$ a $f_2 : \mathbf{B}^3 \to \mathbf{B}$, $f_2(x_1, x_2, x_3) = (x_1 + x_2) + x_3$, zistili by sme, že sú rovnaké ako operácia, ktorej hodnoty označujeme $x_1 + x_2 + x_3$. To pravdaže neplatí v prípade Pierceovej funkcie. Uvažujme o funkcii $h_1^{(3)} : \mathbf{B}^3 \to \mathbf{B}$, $h_1^{(3)}(x_1, x_2, x_3) = (x_1 \downarrow x_2 \downarrow x_3) = \overline{x_1 + x_2 + x_3} = \overline{x}_1 \overline{x}_2 \overline{x}_3$. Porovnajme túto funkciu s funkciou $g_1 : \mathbf{B}^3 \to \mathbf{B}$, $g_1(x_1, x_2, x_3) = ((x_1 \downarrow x_2) \downarrow x_3) = \overline{\overline{\overline{x_1 + x_2}} + x_3} = \overline{\overline{x_1 + x_2}}\,\overline{x}_3 = (x_1 + x_2)\overline{x}_3$ a s funkciou $g_2 : \mathbf{B}^3 \to \mathbf{B}$, $g_2(x_1, x_2, x_3) = (x_1 \downarrow (x_2 \downarrow x_3)) = \overline{x_1 + \overline{\overline{x_2 + x_3}}} = \overline{x}_1 \overline{\overline{x_2 + x_3}} = \overline{x}_1(x_2 + x_3)$. Teraz je už veľmi jednoduché sa presvedčiť, že tieto tri funkcie sú rôzne. To znamená, že binárna Pierceova operácia nie je asociatívna a ternárna Pierceova operácia nie je definovaná ani jednou z funkcií g_1, g_2. Rovnaká situácia nastáva aj pre Shefferove funkcie.

Príklad 2.25. Nájdite $\mathcal{S}_3$-výraz, ktorý reprezentuje funkciu

$$f(x, y, z, u) = yz + \overline{x}yu + x\overline{y}zu.$$

Riešenie.

$$f(x, y, z, u) = yz + \overline{x}yu + x\overline{y}zu = \overline{\overline{yz + \overline{x}yu + x\overline{y}zu}} = \overline{\overline{(yzz)}\,\overline{(\overline{x}yu)}\,\overline{(x\overline{y}zu)}} =$$
$$= \overline{\overline{(yzz)}\,\overline{(\overline{(xxx)}yu)}\,\overline{(xzu)\overline{y}}} = \overline{\overline{(yzz)}\,\overline{(\overline{(xxx)}yu)}\,\overline{\overline{\overline{(xzu)}}\,\overline{yyy}}} =$$
$$= \overline{\overline{(yzz)}\,\overline{(\overline{(xxx)}yu)}\,\overline{\overline{(\overline{(\overline{(xzu)}\cdot 1\cdot 1)}\,\overline{yyy}\cdot 1)}}} =$$
$$= \overline{(y\uparrow z\uparrow z)((x\uparrow x\uparrow x)\uparrow y\uparrow u)\overline{(((x\uparrow z\uparrow u)\uparrow 1\uparrow 1)(y\uparrow y\uparrow y)\cdot 1)}} =$$
$$= \overline{(y\uparrow z\uparrow z)((x\uparrow x\uparrow x)\uparrow y\uparrow u)(((x\uparrow z\uparrow u)\uparrow 1\uparrow 1)\uparrow(y\uparrow y\uparrow y)\uparrow 1)} =$$
$$= ((y\uparrow z\uparrow z)\uparrow((x\uparrow x\uparrow x)\uparrow y\uparrow u)\uparrow(((x\uparrow z\uparrow u)\uparrow 1\uparrow 1)\uparrow(y\uparrow y\uparrow y)\uparrow 1))$$

■

Úplným systémom booleovských funkcií sú aj množiny $\mathcal{P} = \{h_1^{(2)}, h_1^{(3)}, \ldots, h_1^{(n)}, \ldots\}$ a $\mathcal{S} = \{h_2^{(2)}, h_2^{(3)}, \ldots, h_2^{(n)}, \ldots\}$. Preto každú booleovskú funkciu môžeme reprezentovať $\mathcal{P}$-výrazmi aj $\mathcal{S}$-výrazmi.

Príklad 2.26. Vyjadrite funkciu $f(x,y,z,u) = yz + \overline{x}yu + x\overline{y}zu$ pomocou $\mathcal{P}$-výrazov aj $\mathcal{S}$-výrazov.

Riešenie.

$$f(x,y,z,u) = yz + \overline{x}yu + x\overline{y}zu = \overline{\overline{yz}} + \overline{\overline{\overline{x}yu}} + \overline{\overline{x\overline{y}zu}} = \overline{\overline{y} + \overline{z}} + \overline{x + \overline{y} + \overline{u}} + \overline{\overline{x} + y + \overline{z} + \overline{u}} =$$
$$= \overline{\overline{\overline{\overline{y} + \overline{z}} + \overline{x + \overline{y} + \overline{u}} + \overline{\overline{x} + y + \overline{z} + \overline{u}}}} = \overline{\overline{(\overline{y} \downarrow \overline{z}) + (x \downarrow \overline{y} \downarrow \overline{u}) + (\overline{x} \downarrow y \downarrow \overline{z} \downarrow \overline{u})}} =$$
$$= \overline{((\overline{y} \downarrow \overline{z}) \downarrow (x \downarrow \overline{y} \downarrow \overline{u}) \downarrow (\overline{x} \downarrow y \downarrow \overline{z} \downarrow \overline{u}))} = ((\overline{y} \downarrow \overline{z}) \downarrow (x \downarrow \overline{y} \downarrow \overline{u}) \downarrow (\overline{x} \downarrow y \downarrow \overline{z} \downarrow \overline{u}))\downarrow =$$
$$= (((y\downarrow) \downarrow (z\downarrow)) \downarrow (x \downarrow (y\downarrow) \downarrow (u\downarrow)) \downarrow ((x\downarrow) \downarrow y \downarrow (z\downarrow) \downarrow (u\downarrow)))\downarrow$$

$$f(x,y,z,u) = yz + \overline{x}yu + x\overline{y}zu = \overline{\overline{yz + \overline{x}yu + x\overline{y}zu}} = \overline{\overline{yz}\;\overline{\overline{x}yu}\;\overline{x\overline{y}zu}} =$$
$$= \overline{(y \uparrow z)(\overline{x} \uparrow y \uparrow u)(x \uparrow \overline{y} \uparrow z \uparrow u)} = ((y \uparrow z) \uparrow ((x\uparrow) \uparrow y \uparrow u) \uparrow (x \uparrow (y\uparrow) \uparrow z \uparrow u)) \quad \blacksquare$$

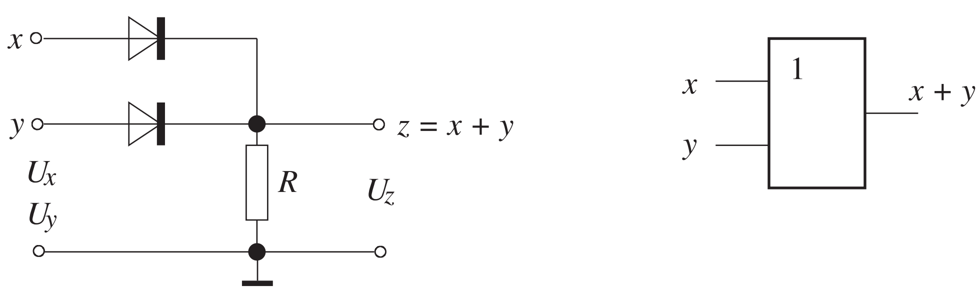
3. Kombinačné logické siete

Booleovské funkcie je možné realizovať (modelovať) pomocou rôznych technických prostriedkov, napr. mechanických, hydraulických, pneumatických alebo elektrických. Ich základom je realizácia tých booleovských funkcií, ktoré tvoria úplný systém. Obvykle sú to logický súčet, logický súčin a negácia (tvoria úplný systém booleovských funkcií). Fyzikálne systémy, ktoré realizujú tieto základné booleovské funkcie budeme nazývať (***základné***) ***logické členy***. Pri realizácii pomocou elektrických prostriedkov sa najčastejšie používajú tieto dva spôsoby:

1. Využíva sa prítomnosť elektrického prúdu (napätia) na priradenie hodnoty 1 a jeho neprítomnosť na priradenie hodnoty 0. (V tomto prípade ide o tzv. pozitívnu logiku. Je možné uvažovať aj o opačnom priradení, vtedy hovoríme o negatívnej logike. My budeme používať iba pozitívnu logiku.)

2. Využívajú sa technické zariadenia, ktoré sú schopné prevádzky pri dvoch (značne) odlišných hladinách elektrického napätia. Pre naše úvahy je výhodné jednu z nich označiť ako „vysokú“ hladinu napätia (v praxi spravidla 5 V) a priradiť jej hodnotu 1 a druhú označiť ako „nízku“ hladinu napätia (v praxi spravidla 0,5 V) a priradiť jej logickú hodnotu 0 (pozitívna logika).

V zariadeniach prvej skupiny základnou zložkou logických členov sú obvykle kontaktné relé. V zariadeniach druhej skupiny sa na konštrukciu logických členov používajú polovodičové diódy a tranzistory.



a)   b)

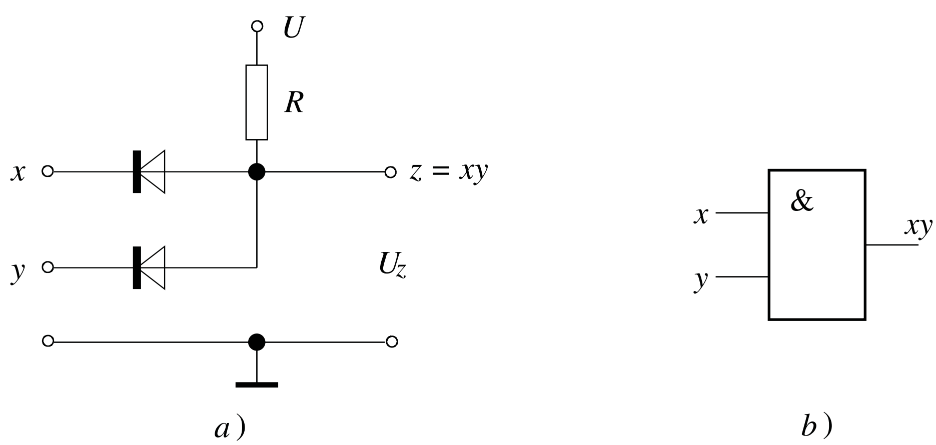
Obr. 1. Súčtový logický člen.

Každý logický člen je charakterizovaný vzťahom medzi jeho vstupnými a výstupnými logickými premennými. ***Vstupné premenné*** sú nezávislé a ***výstupné premenné*** sú závislé premenné tej booleovskej funkcie, ktorú daný logický člen realizuje. Každej vstupnej premennej x (výstupnej premennej z) je priradené napätie $U_x \in \{U_0, U_I\}$ $(U_z \in \{U_0, U_I\})$.

U_I nazývame ***jednotkovou úrovňou***, U_0 ***nulovou úrovňou*** napätia (signálu). Napätie U_x nazývame ***vstupným***, U_z ***výstupným napätím*** logického člena. Pripomeňme, že pre stručnosť vyjadrovania obvykle nerozlišujeme medzi x a U_x, resp. z a U_z.

Súčtový logický člen (člen OR) je realizáciou booleovskej funkcie $f(x, y) = x + y$. Jedna takáto realizácia je znázornená na obr. 1a. Ideálne berme do úvahy, že napr. $U_I = 5\,\mathrm{V}$, $U_0 = 0\,\mathrm{V}$. Ak potom pripojíme na ktorúkoľvek z diód jednotkovú úroveň napätia, dióda sa otvorí a $U_z = U_I$ (zmenšené o úbytok na dióde, čo zanedbávame). Diódy pripojené na úroveň U_0 sa uzatvoria. Ak sa zanedbajú úbytky napätí na diódach, pre výstupné napätie platí $U_z = \max\{U_x, U_y\}$, čo skutočne zodpovedá funkčným hodnotám logického súčtu. Pre súčtový logický člen budeme používať symbolickú značku uvedenú na obr. 1b.

Súčinový logický člen (člen AND) realizuje booleovskú funkciu $f(x, y) = xy$. Na obr. 2a je jeho diódová realizácia. Nech je $U > U_I$. Ak je na ktorúkoľvek diódu pripojené vstupné napätie U_0, dióda je otvorená, a ak zanedbáme jej vnútorný odpor v priamom smere, úroveň U_0 (zväčšená o úbytok na dióde, ktorý môžeme zanedbať) je aj na výstupe člena. Úroveň napätia U_I sa objaví na výstupe člena práve vtedy, keď sú obidve diódy

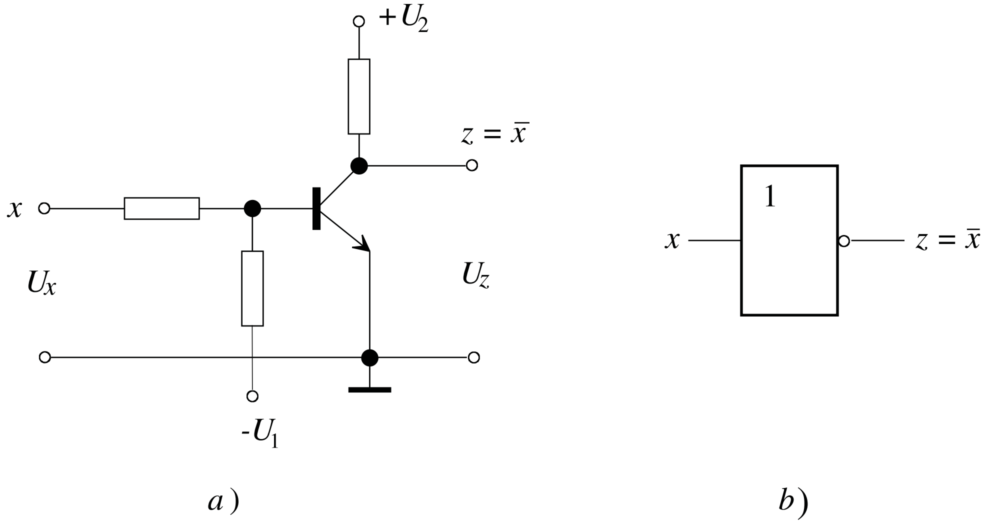


Obr. 2. Súčinový logický člen.

uzavreté, t.j. keď je na ne pripojené vstupné napätie U_I. Ak zanedbáme úbytky napätia na diódach, tak pre výstupné napätie platí $U_z = \min\{U_x, U_y\}$. Obvod teda skutočne realizuje logický súčin. Symbolická značka súčinového logického člena je na obr. 2b.

Logický člen negátor (***invertor***) (člen NOT) realizuje funkciu $f(x) = \overline{x}$. Na obr. 3a je príklad jednoduchého tranzistorového negátora, kde tranzistor pracuje ako bezkontaktový spínač. Pri $U_x = U_I$ je tranzistor otvorený a ak zanedbáme jeho odpor v priamom smere, $U_z = U_0$. Naopak pri $U_x = U_0$ je tranzistor zatvorený, takže $U_z = U_I$. Schematická značka negátora je na obr. 3b.

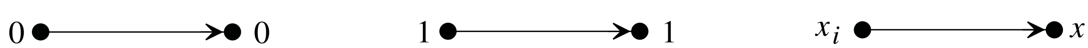
Fyzikálnu realizáciu ľubovoľnej booleovskej funkcie nazývame ***kombinačný logický obvod***. Nás teraz nebude zaujímať skutočná fyzikálna realizácia booleovskej funkcie ale iba jej schéma (kombinačná logická sieť), ktorá opisuje, ako sa kombinačný logický obvod zostaví zo základných logických členov. Takéto schémy sme už definovali pre logické členy, sú to ich schematické značky. Všimnime si, že ich môžeme považovať za orientované grafy, ktoré v prípade súčtového a súčinového logického člena majú dva vstupné vrcholy x, y a jeden výstupný vrchol a v prípade negátora jeden vstupný vrchol x a jeden výstupný vrchol $z = x + y$, resp. $z = xy$. Takúto kombinačnú logickú sieť môžeme priradiť ku každému B-výrazu.



Obr. 3. Logický člen negátor.

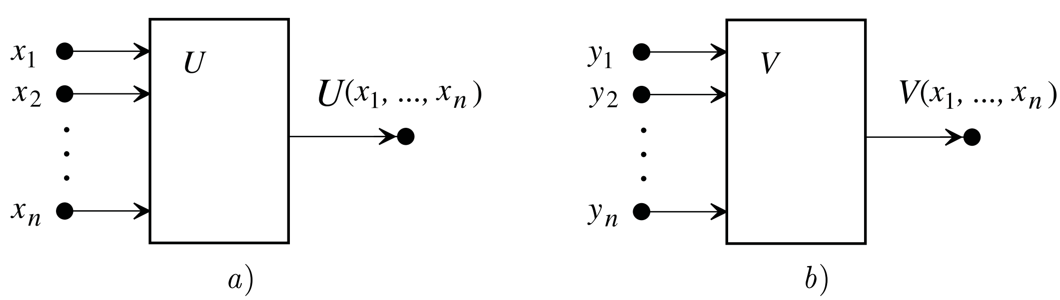
Definícia 2.24. Kombinačná logická sieť prislúchajúca k B-výrazu $U(x_1, x_2, \dots, x_n)$ je jednoduchý orientovaný graf, ktorý obsahuje n vstupných vrcholov $x_1, x_2, \dots, x_n$, jeden výstupný vrchol $U(x_1, x_2, \dots, x_n)$ a niekoľko funkčných vrcholov, ktoré predstavujú použité logické členy. Schéma takéhoto grafu je na obr. 5a. Vnútorná štruktúra (ďalšie hrany a funkčné vrcholy) tohto grafu je reprezentovaná pomocou logických členov v závislosti od induktívnej definície B-výrazu takto:

1. Kombinačné logické siete patriace k B-výrazom $U(x_1, x_2, \dots, x_n) = 0$, $V(x_1, x_2, \dots, x_n) = 1$ a $W(x_1, x_2, \dots, x_n) = x_i$ sú na obr. 4.



Obr. 4. Kombinačné logické siete patriace k B-výrazom 0, 1 a x_i

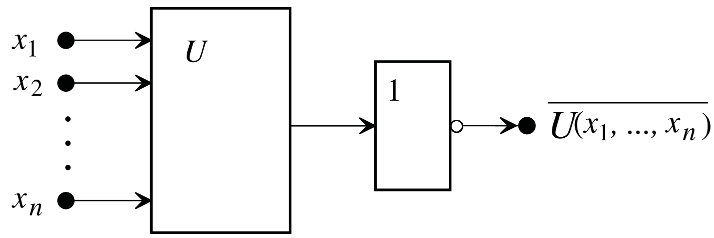
2. Nech kombinačná logická sieť patriaca k B-výrazu $U(x_1, x_2, \dots, x_n)$ je na obr. 5a. Potom kombinačná logická sieť patriaca k B-výrazu $\overline{U(x_1, x_2, \dots, x_n)}$ je na obr. 6.



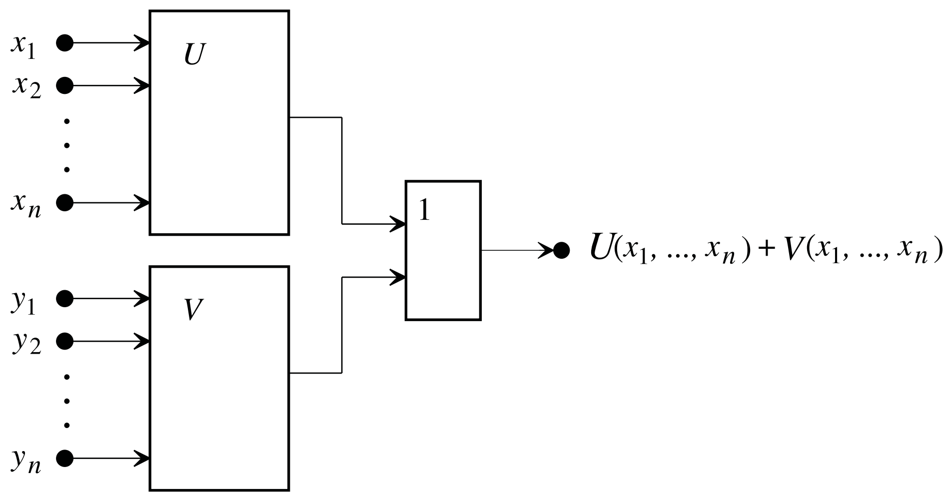
Obr. 5. Kombinačné logické siete patriace k B-výrazom U, V

3. Nech kombinačná logická sieť patriaca k B-výrazu $U(x_1, x_2, \dots, x_n)$ je na obr. 5a a kombinačná logická sieť patriaca k B-výrazu $V(y_1, y_2, \dots, y_n)$ je na obr. 5b. Potom kombinačné logické siete patriace k B-výrazom $U(x_1, x_2, \dots, x_n) + V(y_1, y_2, \dots, y_n)$ a $U(x_1, x_2, \dots, x_n) \cdot V(y_1, y_2, \dots, y_n)$ sú na obr. 7 a 8.

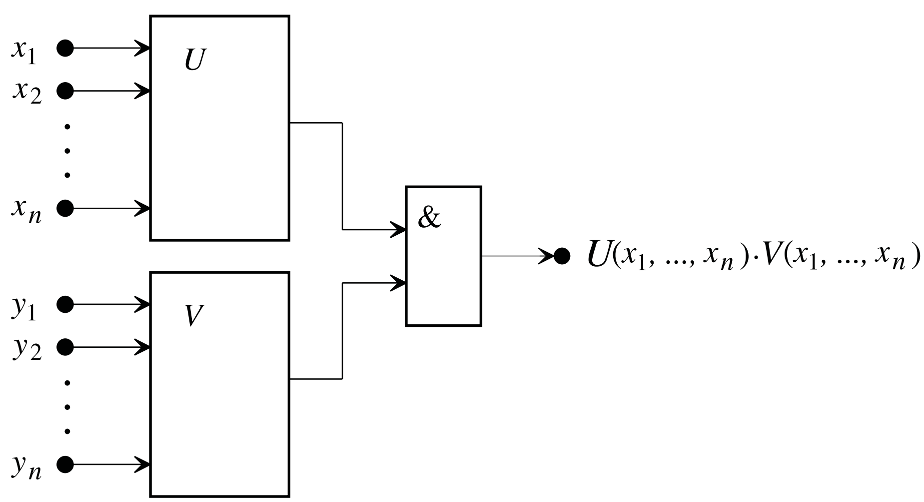
Príklad 2.27. Na obr. 9 je kombinačná logická sieť prislúchajúca k B-výrazu $U(x_1, x_2, x_3) = (\overline{x_1 + x_2})x_3 + (x_1 + x_2)\overline{x}_3$. ■



OBR. 6. Kombinačná logická sieť patriaca k B-výrazu $\overline{U}$



OBR. 7. Kombinačná logická sieť patriaca k B-výrazu $U + V$

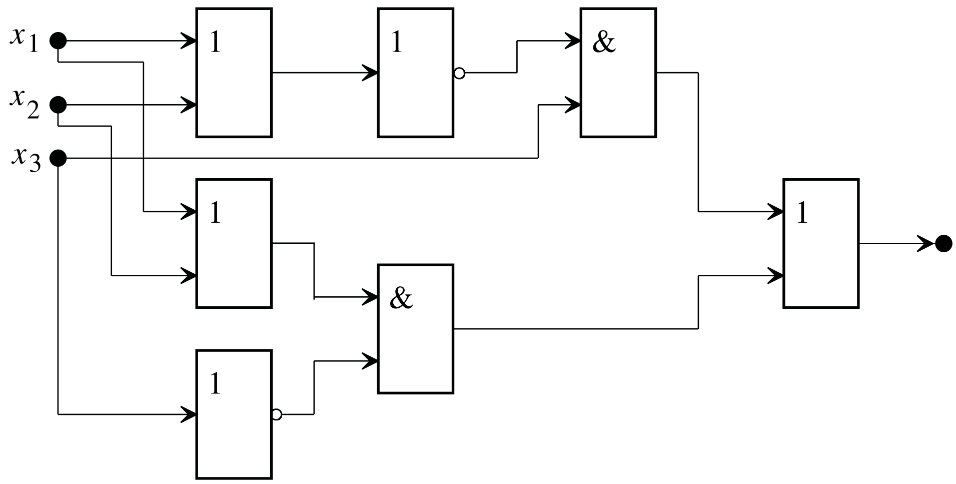


OBR. 8. Kombinačná logická sieť patriaca k B-výrazu $U \cdot V$

Dohoda o zjednodušení kreslenia kombinačných sietí

Pri kreslení kombinačných logických sietí budeme používať tieto zjednodušenia:

1. Orientované hrany budú smerovať zľava doprava, preto na nich prestaneme kresliť šípky.

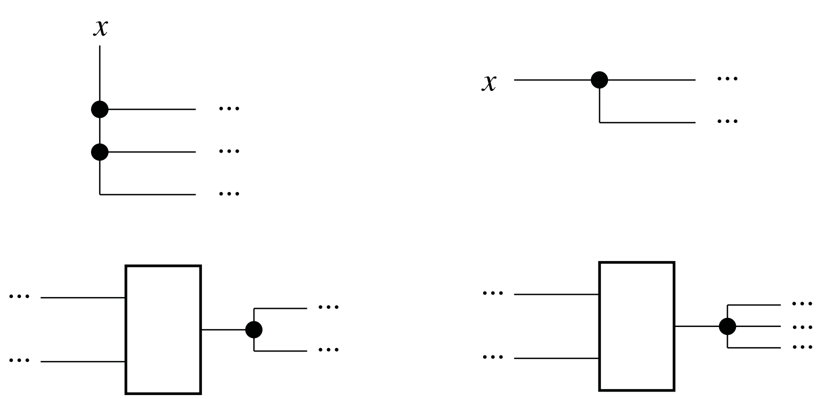


OBR. 9. Kombinačná logická sieť patriaca k B-výrazu $(\overline{x_1 + x_2})x_3 + (x_1 + x_2)\overline{x}_3$

2. Vstupné vrcholy budeme vyznačovať iba vtedy, keď z nich vystupujú najmenej dve hrany. Vstupné premenné zodpovedajúce nevyznačeným vstupným vrcholom pripíšeme k hranám, ktoré s týmito vrcholmi incidujú.

3. Výstupné vrcholy vynecháme a im priradené výstupné premenné pripíšeme k príslušným hranám.

4. Ak z jedného vstupného alebo funkčného vrchola vychádza viac hrán, pridávame do kombinačnej logickej siete nové vrcholy - ***vnútorné vrcholy***, ako to vidieť na obr. 10. Vnútorný vrchol kreslíme iba vtedy, keď z neho vychádzajú aspoň dve hrany.



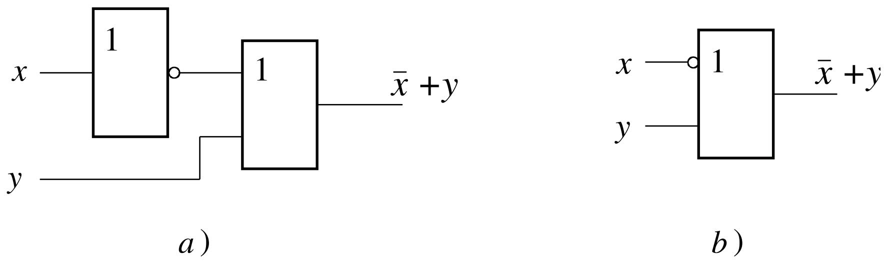
OBR. 10. Vnútorné vrcholy

5. V prípade, že na vstup niektorého logického člena privádzame negáciu niektorej premennej, vyznačíme to krúžkom na príslušnom vstupe. Teda napr. logickú sieť z obr. 11a nahradíme logickou sieťou z obr. 11b.

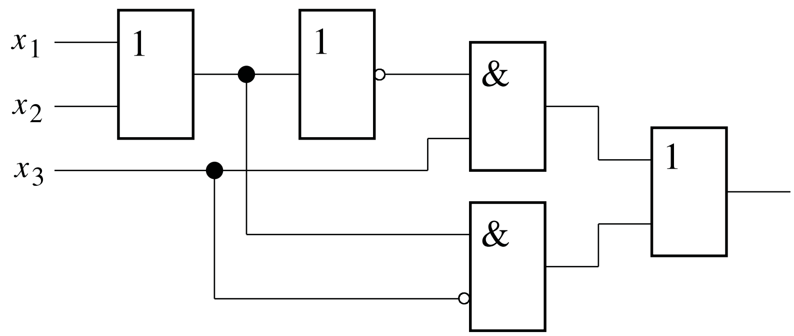
Kombinačná logická sieť z predchádzajúceho príkladu nakreslená podľa dohodnutých zjednodušení je na obr. 12.

Pri zostavovaní kombinačných logických sietí booleovských funkcií budeme ako funkčné vrcholy používať aj iné logické členy, než sme doteraz uviedli, napr.:

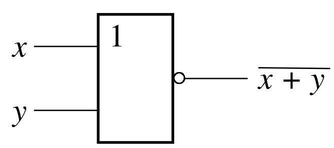
Logický člen NOR je realizáciou booleovskej funkcie $f(x, y) = \overline{x + y}$. Jeho označenie je na obr. 13.



Obr. 11. Zjednodušenie kreslenia logickej siete, ak na vstup logického člena je privádzaná negácia premennej

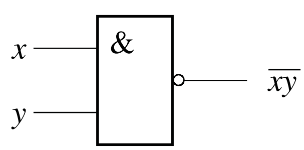


Obr. 12. Kombinačná logická sieť z príkladu 2.27



Obr. 13. Logický člen NOR

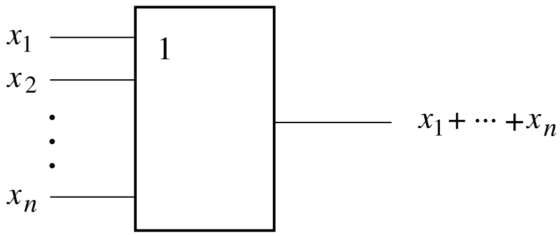
Logický člen NAND je realizáciou booleovskej funkcie $f(x,y) = \overline{xy}$. Jeho označenie je na obr. 14.



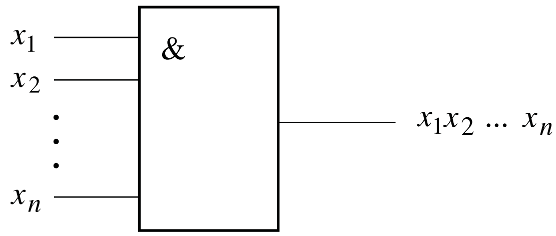
Obr. 14. Logický člen NAND

n***-vstupový logický člen OR*** (resp. ***AND***, ***NOR***, ***NAND***) $n \geq 2$, je realizáciou funkcie $f(x_1, x_2, \dots, x_n) = x_1 + x_2 + \cdots + x_n$ (resp. $f(x_1, x_2, \dots, x_n) = x_1 x_2 \dots x_n$, $f(x_1, x_2, \dots, x_n) = \overline{x_1 + x_2 + \cdots + x_n}$, $f(x_1, x_2, \dots, x_n) = \overline{x_1 x_2 \dots x_n}$). Označenia týchto logických členov sú v poradí na obr. 15 až 18. V technickej praxi je počet vstupov týchto logických členov zhora ohraničený spravidla číslom 8. My o hornom ohraničení vstupov logických členov nebudeme uvažovať.

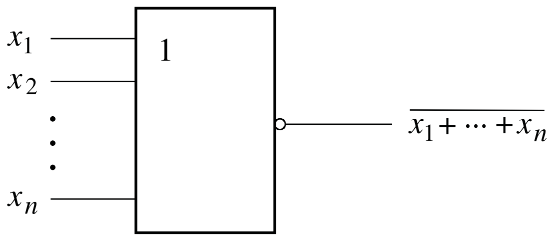
Logický člen XOR je realizáciou booleovskej funkcie $f(x,y) = \overline{x}y + x\overline{y} = x \oplus y$. Jeho kombinačná logická sieť a značka sú na obr. 19.



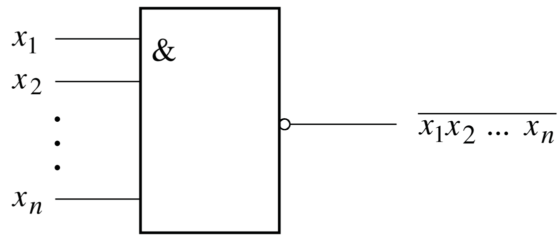
Obr. 15. n-vstupový logický člen OR



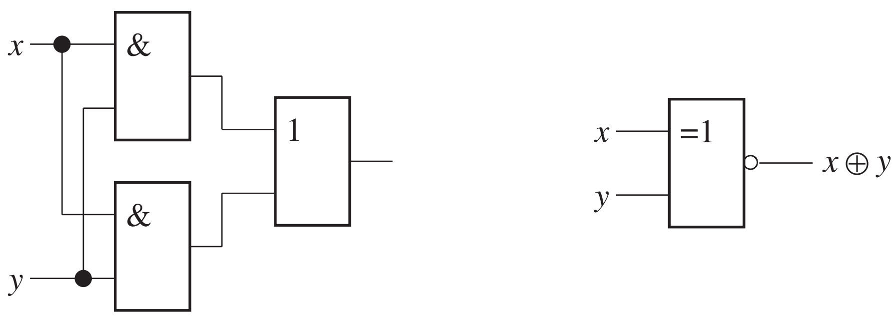
Obr. 16. n-vstupový logický člen AND



Obr. 17. n-vstupový logický člen NOR

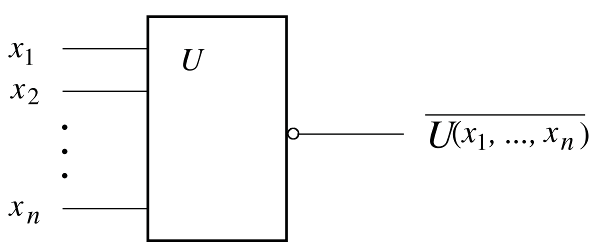


Obr. 18. n-vstupový logický člen NAND



Obr. 19. Logický člen XOR

Na záver sa ešte dohodnime, že pri kreslení kombinačnej siete patriacej ku komplementu výrazu budeme logickú sieť z obr. 6 nahrádzať logickou sieťou z obr. 20.



Obr. 20. Kombinačná logická sieť patriaca k $\overline{U(x_1, x_2, \dots, x_n)}$

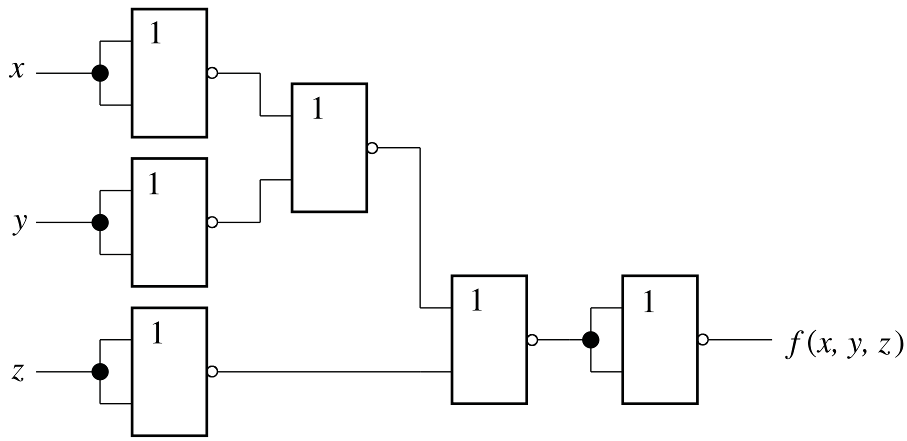
Príklad 2.28. Nakreslite kombinačnú logickú sieť zostavenú len z dvojvstupových logických členov NOR, patriacu booleovskej funkcii $f(x,y,z) = xy + \overline{z}$.

Riešenie. Vyjadrime funkciu f pomocou Pierceho funkcie dvoch premenných, teda pomocou $\mathcal{P}_2$-výrazov:

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

$$= (((x \downarrow x) \downarrow (y \downarrow y)) \downarrow (z \downarrow z)) \downarrow (((x \downarrow x) \downarrow (y \downarrow y)) \downarrow (z \downarrow z))$$

Na základe tohto vyjadrenia už ľahko zostavíme kombinačnú logickú sieť funkcie f. Je na obr. 21. ■



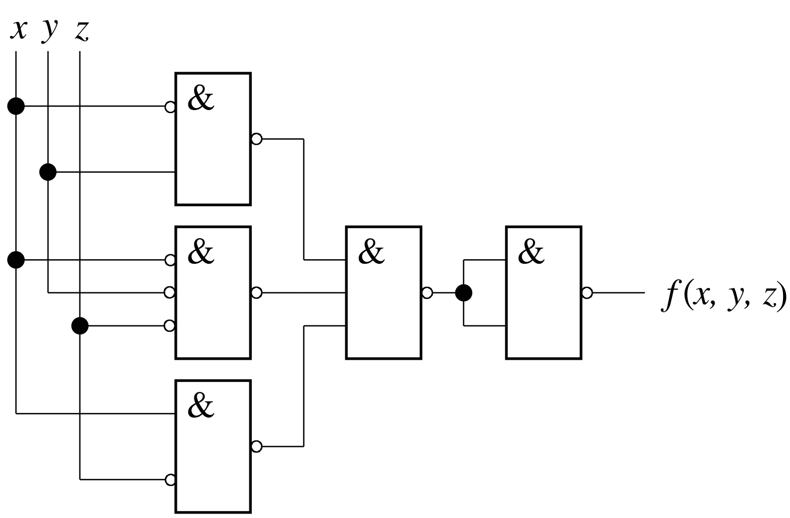
Obr. 21. Kombinačná logická sieť funkcie f z príkladu 2.28

Príklad 2.29. Nakreslite kombinačnú logickú sieť zostavenú len z logických členov NAND (resp. NOR) patriacu booleovskej funkcii $f(x,y,z) = (x+\overline{y})(x+y+z)(\overline{x}+z)$ bez obmedzenia počtu vstupov logických členov.

Riešenie. Vyjadríme funkciu f pomocou $\mathcal{S}$-výrazov resp. $\mathcal{P}$-výrazov. Pritom pripustíme, že vo výrazoch určujúcich funkciu f môžu vystupovať aj negácie premenných.

a) Kombinačná logická sieť zostavená pomocou logických členov NAND (obr.22).

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



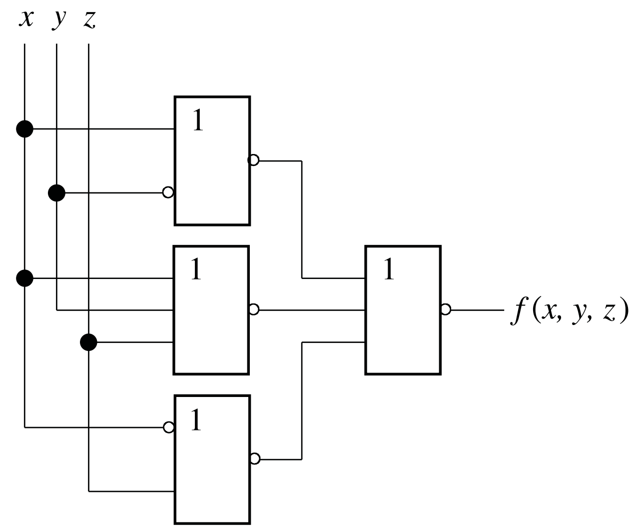
Obr. 22. Kombinačná logická sieť zostavená z logických členov NAND

b) Kombinačná logická sieť zostavená pomocou logických členov NOR (obr.23).

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

$$= ((x \downarrow \overline{y}) \downarrow (x \downarrow y \downarrow z) \downarrow (\overline{x} \downarrow z))$$

■



OBR. 23. Kombinačná logická sieť zostavená z logických členov NOR

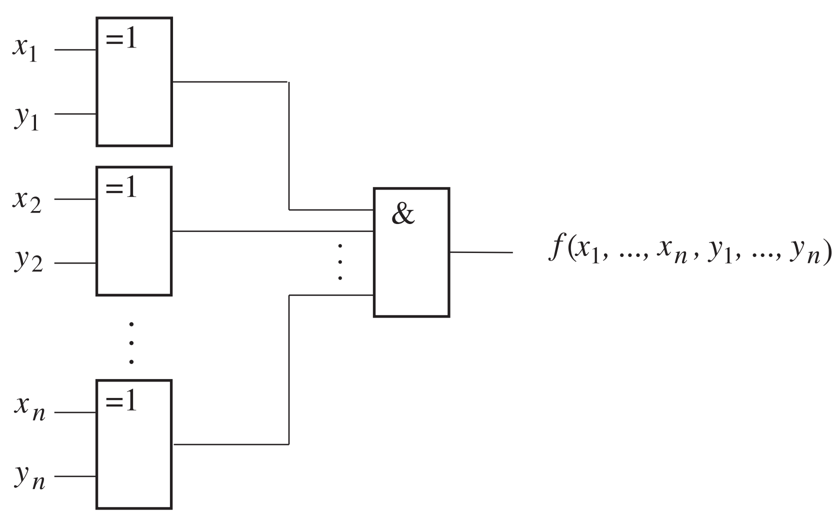
PRÍKLAD 2.30. (Porovnávací obvod - komparátor) Nech $c, d \in \mathbf{B}^n$, $c = (c_1, c_2, \ldots, c_n)$, $d = (d_1, d_2, \ldots, d_n)$. Chceme navrhnúť logickú sieť, pomocou ktorej by sme automaticky rozhodli, či $c = d$, alebo $c \neq d$. Preto uvažujeme o funkcii $f : \mathbf{B}^{2n} \to \mathbf{B}$, pre ktorú platí:

$$f(c, d) = f(c_1, c_2, \ldots, c_n, d_1, d_2, \ldots, d_n) = \begin{cases} 1, & \text{pre } c = d \\ 0, & \text{pre } c \neq d \end{cases}$$

Ľahko sa presvedčíme, že tejto požiadavke vyhovuje funkcia

$$f(x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n) = \overline{(x_1 \oplus y_1)}\,\overline{(x_2 \oplus y_2)} \ldots \overline{(x_n \oplus y_n)}.$$

Logickú sieť patriacu k danej funkcii uvádzame na obr. 24. ■



OBR. 24. Kombinačná logická sieť komparátora

4. Booleovské funkcie $f : \mathbf{B}^n \to \mathbf{B}^m$

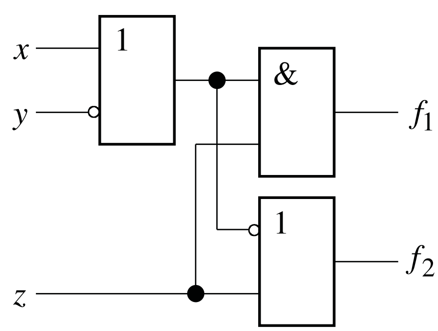
Funkcia $f : \mathbf{B}^n \to \mathbf{B}^m$, $f = (f_1, f_2, \dots, f_m)$ má m zložiek. Každá z týchto zložiek je zrejme booleovská funkcia, teda $f_i : \mathbf{B}^n \to \mathbf{B}$, pre $i \in \{1, 2, \dots, m\}$. Preto aj funkciu $f : \mathbf{B}^n \to \mathbf{B}^m$ budeme nazývať booleovská funkcia.

Každá zložka $f_i : \mathbf{B}^n \to \mathbf{B}$ booleovskej funkcie $f : \mathbf{B}^n \to \mathbf{B}^m$ je určená B-výrazom. Ku každému B-výrazu patrí kombinačná logická sieť. Dostávame m-ticu logických sietí, ktorá reprezentuje fyzikálnu realizáciu danej booleovskej funkcie. Túto m-ticu logických sietí budeme nazývať kombinačná logická sieť patriaca k funkcii $f : \mathbf{B}^n \to \mathbf{B}^m$. Táto logická sieť je daná pomocou B-výrazov určujúcich jednotlivé zložky danej funkcie. Jedným zo špecifík tejto siete je tá vlastnosť, že všetky logické siete patriace k jednotlivým zložkám majú spoločné vstupné vrcholy. Preto môžeme uvažovať aj o spoločných logických členoch, ktoré sa môžu nachádzať súčasne v niekoľkých logických sieťach, patriacich k jednotlivým zložkám danej funkcie. Tento problém už súvisí s otázkou minimalizácie kombinačných logických sietí. Podrobnejšie sa o ňom môžete dočítať v práci [**3**].

Príklad 2.31. Logická sieť patriaca k booleovskej funkcii

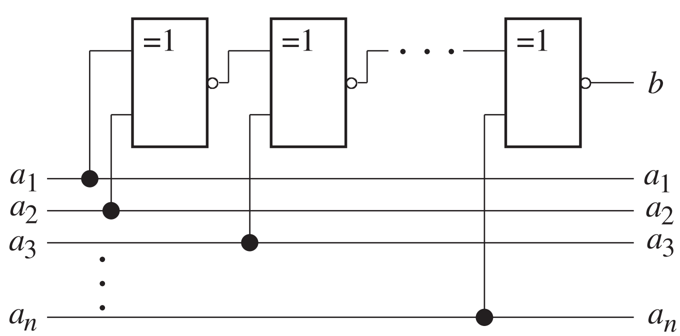
$$f : \mathbf{B}^3 \to \mathbf{B}^2, f(x,y,z) = (f_1(x,y,z), f_2(x,y,z)) = ((x+\overline{y})z, z + \overline{(x+\overline{y})})$$

je na obr. 25. ■



Obr. 25. Kombinačná logická sieť funkcie $f : \mathbf{B}^3 \to \mathbf{B}^2$ z príkladu 2.31

Príklad 2.32. (Kontrola parity) Pri prenose informácií sa často k prenášanej n-tici pridáva ešte jedno miesto (jeden bit), aby sme mali možnosť kontroly prenesenej užitočnej informácie (na vstupe máme n-ticu $(a_1, a_2, \dots, a_n)$ a na výstupe $(n+1)$-ticu). Na obr. 26 uvádzame logickú sieť, pomocou ktorej generujeme prídavný bit b tak, aby počet všetkých



Obr. 26. Generovanie prídavného bitu na kontrolu parity

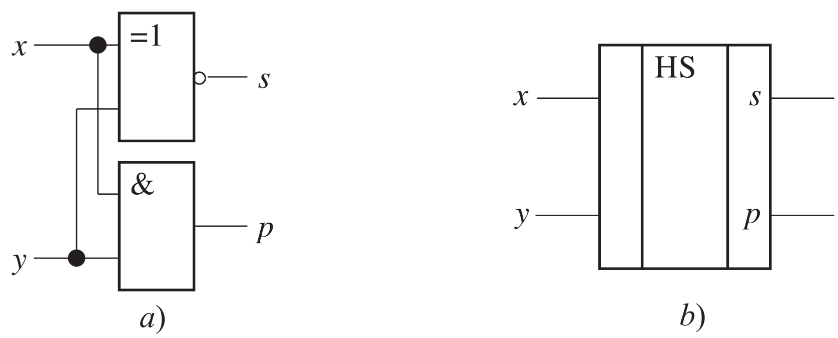
jednotiek na výstupe bol párny. To znamená, že ak na výstupe je nepárny počet jednotiek, ide o nesprávny prenos (šum). ■

PRÍKLAD 2.33. Uvažujme o funkcii $f : \mathbf{B}^2 \to \mathbf{B}^2$, $f(x,y) = (s,p)$, ktorej hodnoty uvádzame v tab. 6. Hodnoty tejto funkcie sa vyskytujú pri sčítaní dvoch čísel x, y v dvojkovej sústave. Hodnota s určuje zvyšok po delení (obyčajného) súčtu číslom 2 (teda je

TABUĽKA 6. Tabuľka funkcie $f(x,y) = (s,p)$

x, y	s	p
0, 0	0	0
0, 1	1	0
1, 0	1	0
1, 1	0	1

to sčítanie podľa modulu 2). Hodnota p nám udáva prenos, ktorý je nenulový iba pri sčítaní $1 + 1$. Z tabuľky priamo vyplýva, že $s = x \oplus y$ a $p = xy$. Logická sieť patriaca k tejto funkcii je nakreslená na obr. 27a. Na obr. 27b je nakreslená značka logického člena,



OBR. 27. Logický člen polosčítačka

ktorý je realizáciou booleovskej funkcie $f : \mathbf{B}^2 \to \mathbf{B}^2$, $f(x,y) = (s,p)$. Tento logický člen nazývame ***polosčítačka***. ■

PRÍKLAD 2.34. Pri sčítovaní čísel v dvojkovej sústave je potrebná aj funkcia $f : \mathbf{B}^3 \to \mathbf{B}^2$, $f(x,y,z) = (s,p)$, ktorá udáva súčet s podľa modulu 2 čísel x, y, z a prenos p. Jej hodnoty uvádzame v tab. 7. Ľahko sa dá overiť, že $s = x \oplus y \oplus z$ a priamo z tabuľky dostávame

$$p = \overline{x}yz + x\overline{y}z + xy\overline{z} + xyz = (\overline{x}yz + xyz) + (x\overline{y}z + xyz) + (xy\overline{z} + xyz) = yz + xz + xy$$

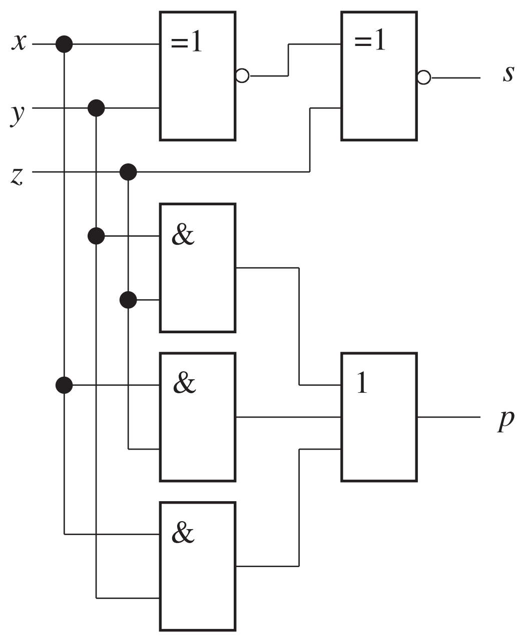
Príslušnú logickú sieť funkcie f uvádzame na obr. 28. Na obr. 29 je značka logického člena, ktorý je realizáciou funkcie f. Tento logický člen sa nazýva ***úplná sčítačka***. ■

Logické členy polosčítačka a úplná sčítačka sa dajú využiť na generovanie logickej siete, pomocou ktorej je dvom číslam $x_n\, x_{n-1} \ldots x_0$ a $y_n\, y_{n-1} \ldots y_0$, zapísaným v dvojkovej sústave, priradený ich súčet $s_{n+1}\, s_n\, s_{n-1} \ldots s_0$. Túto logickú sieť uvádzame na obr. 30.

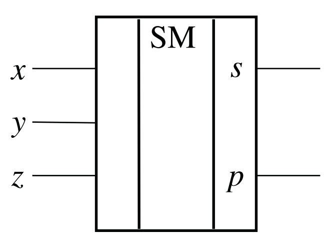
Všimnime si ešte, že keby sme chceli zostrojiť booleovskú funkciu vyjadrujúcu sčítanie dvoch 8-miestnych čísel v dvojkovej sústave, potrebovali by sme vytvoriť tabuľku funkcie $f : \mathbf{B}^{16} \to \mathbf{B}^9$. Tabuľka tejto funkcie by mala $2^{16} = 2^6 \cdot 2^{10} \doteq 64 \cdot 10^3 = 64000$ riadkov, čo predstavuje asi 2000 tlačených strán.

Tabuľka 7. Tabuľka funkcie $f(x, y, z) = (s, p)$

$x\ y\ z$	s	p
0 0 0	0	0
0 0 1	1	0
0 1 0	1	0
0 1 1	0	1
1 0 0	1	0
1 0 1	0	1
1 1 0	0	1
1 1 1	1	1

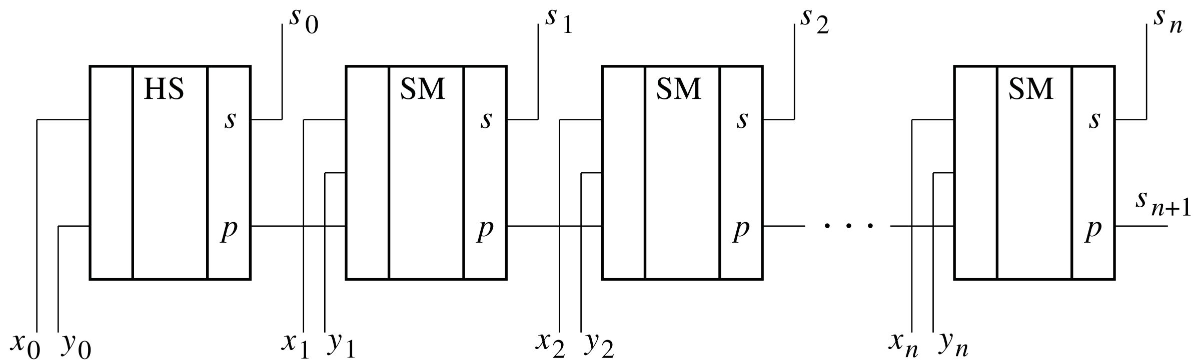


Obr. 28. Kombinačná logická sieť úplnej sčítačky



Obr. 29. Značka pre úplnú sčítačku

Príklad 2.35. (Dvojkový dekóder) Budeme uvažovať o logickom obvode, ktorý má n adresových vstupov a 2^n výstupov. Adresové vstupy sa zvyknú označovať postupne $A, B, C, \dots$ a výstupy $s_0, s_1, s_2, \dots$. Na vstup privádzame n-ticu $j_1 j_2 \dots j_n \in \mathbf{B}^n$, ktorú považujeme za dvojkový zápis čísla $j = j_1 2^{n-1} + j_2 2^{n-2} + \dots + j_n 2^0$, pričom číslicu j_n s najnižšou váhou privádzame na vstup A, číslicu j_{n-1} na vstup B atď. Ak na vstup privedieme n-ticu $j_1 j_2 \dots j_n$, aktivuje sa práve výstup s_j, kde $j = j_1 2^{n-1} + j_2 2^{n-2} + \dots +$



Obr. 30. Logická sieť pre sčitovanie dvoch čísel v dvojkovej sústave

Tabuľka 8. Booleovská funkcia z príkladu 2.35

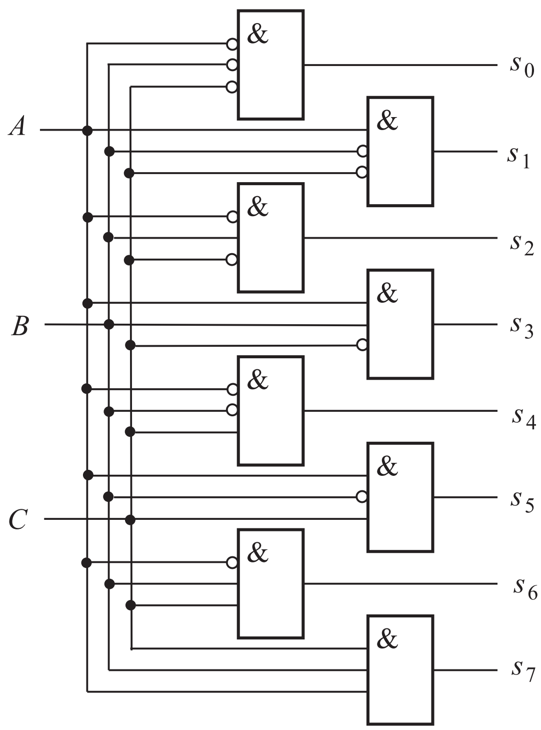
C	B	A	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

$j_n 2^0$, ktorý nadobúda hodnotu 1, ostatné výstupy majú hodnotu 0. Booleovskú funkciu $f : \mathbf{B}^3 \to \mathbf{B}^8, f = (s_0, s_1, \dots, s_7)$, ktorá opisuje daný logický obvod pre $n = 3$, uvádzame v tabuľke 8. Pre výstupy daného logického obvodu dostávame: $s_0 = \overline{A}\,\overline{B}\,\overline{C}$, $s_1 = A\overline{B}\,\overline{C}$, $s_2 = \overline{A}B\overline{C}$, $s_3 = \overline{A}BC$, $s_4 = \overline{A}\,\overline{B}C$, $s_5 = A\overline{B}C$, $s_6 = \overline{A}BC$, $s_7 = ABC$. Príslušnú logickú sieť uvádzame na obr. 31. Na obr. 32 je logický člen, ktorý je symbolom (ľubovoľnej) logickej siete patriacej k uvedenej funkcii $f : \mathbf{B}^3 \to \mathbf{B}^8$. Tento logický člen nazývame ***dvojkový dekóder*** (s troma adresovými vstupmi). ■

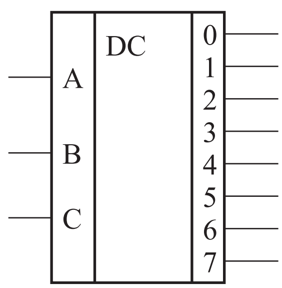
Príklad 2.36. (Multiplexor) Pomocou dvojkového dekódera zostavíme schému logického obvodu, ktorý má n adresových vstupov $A, B, C, \dots$ a 2^n tzv. informačných vstupov $E_0, E_1, E_2, \dots$. Tento logický obvod má iba jeden výstup S. Na adresové vstupy privedieme n-ticu $j_1 j_2 \dots j_n \in \mathbf{B}^n$, ktorú považujeme za dvojkový zápis čísla $j = j_1 2^{n-1} + j_2 2^{n-2} + \dots + j_n 2^0$, pričom na vstup A privádzame hodnotu j_n s najmenšou váhou, na vstup B hodnotu j_{n-1} atď. Na informačné vstupy privádzame ľubovoľné (logické) hodnoty. Vtedy na výstupe S dostávame tú hodnotu, ktorá je privedená na informačný vstup E_j. Symbolicky uvádzame hodnoty príslušnej funkcie S pre $n = 3$ v tabuľke 9. Všimnime si, že v skutočnosti ide o funkciu $3 + 2^3 = 11$ premenných. Na vyplnenie tabuľky takejto funkcie by sme potrebovali 2^{11} riadkov $= 2048$ riadkov $= 65$ strán. Ďalej je zrejmé, že

$$S = \overline{A}\,\overline{B}\,\overline{C}E_0 + A\overline{B}\,\overline{C}E_1 + \overline{A}B\overline{C}E_2 + AB\overline{C}E_3 + \overline{A}\,\overline{B}CE_4 + A\overline{B}CE_5 + \overline{A}BCE_6 + ABCE_7.$$

Logickú sieť patriacu k danej funkcii sme získali sme získali pomocou dvojkového dekódera. Táto sieť je nakreslená na obr. 33. Označenie príslušného logického člena, ktorý nazývame ***multiplexor***, uvádzame na obr. 34. ■



Obr. 31. Logická sieť z príkladu 2.35

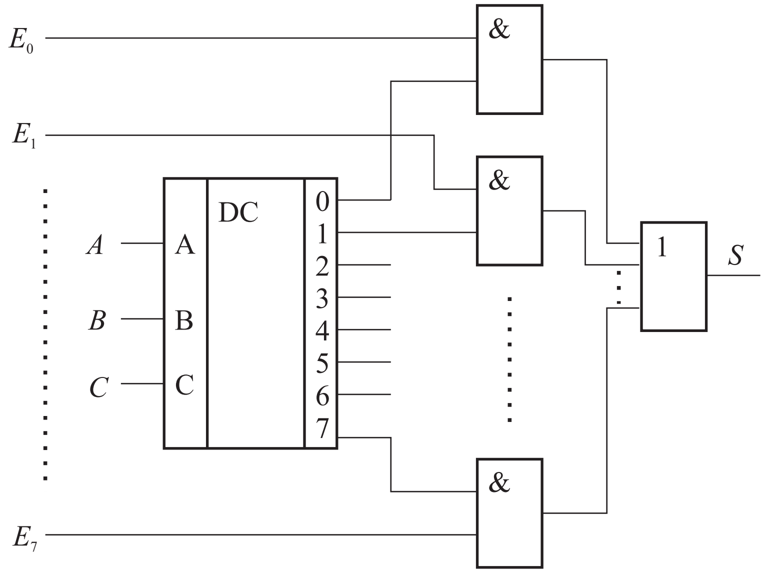


Obr. 32. Logický člen dvojkový dekóder

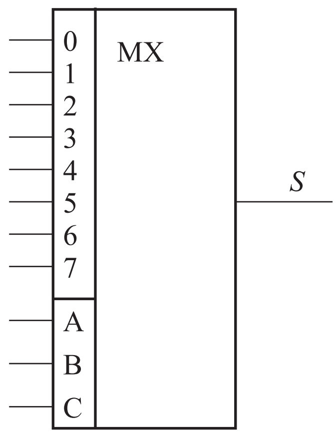
Tabuľka 9. Funkcia z príkladu 2.36

C	B	A	S
0	0	0	E_0
0	0	1	E_1
0	1	0	E_2
0	1	1	E_3
1	0	0	E_4
1	0	1	E_5
1	1	0	E_6
1	1	1	E_7

Príklad 2.37. (Použitie multiplexora na generovanie logických funkcií) Uvažujme o ľubovoľnej (ale pevne zvolenej) funkcii $f : \mathbf{B}^3 \to \mathbf{B}$. Jednu funkciu sme zvolili pomocou tabuľky 10. Ak chceme, aby na výstupe S multiplexora bola funkcia f, porovnaním tabu-



Obr. 33. Logická sieť z príkladu 2.36

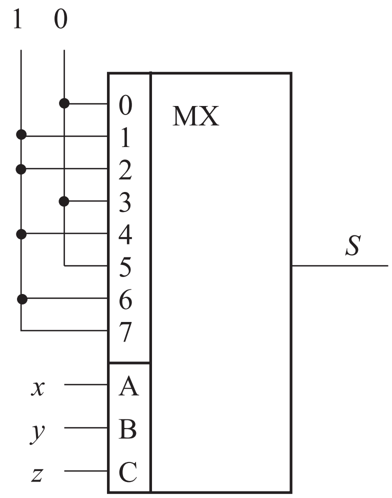


Obr. 34. Logický člen multiplexor

Tabuľka 10. Funkcia z príkladu 2.37

x	y	z	f
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	1

liek 10 a 9, zistíme, že stačí túto funkciu považovať za funkciu premenných $C = x$, $B = y$, $A = z$ a pre $j = j_1 2^2 + j_2 2^1 + j_3 2^0$ na informačný vstup E_j pripojiť konštantu 0 práve vtedy, keď $f(j_1, j_2, j_3) = 0$, a konštantu 1 práve vtedy, keď $f(j_1, j_2, j_3) = 1$. Logickú sieť patriacu k funkcii f z tabuľky 10 sme zostavili pomocou multiplexora s troma adresovými vstupmi. Je nakreslená na obr. 35. ■



Obr. 35. Logická sieť z príkladu 2.37

Príklad 2.38. Keď pozornejšie študujeme tabuľku 9, vidíme, že v každom riadku máme dve možnosti pre voľbu E_j, a teda na výstupe môžeme dostať 16 rôznych možností. To znamená, že pomocou multiplexora s troma adresovými vstupmi môžeme generovať aj funkciu štyroch premenných. Postupujeme pritom takto. Uvažujme o funkcii $f : \mathbf{B}^4 \to \mathbf{B}$, ktorej premenné označme postupne x, y, z, u. Tak ako v predošlom príklade položme $x = C$, $y = B$, $z = A$. Ďalší postup vyvsvetlíme na konkrétnej funkcii $f : \mathbf{B}^4 \to \mathbf{B}$, ktorej hodnoty uvádzame v tabuľke 11. Pri zostavovaní tejto schémy sme postupovali takto. Štvorice $(x, y, z, u) = (C, B, A, u)$ usporiadame do dvojíc tak, aby sa prvky v jednotlivých dvojiciach nelíšili na prvých troch miestach. Dostávame dvojice $(j_1, j_2, j_3, 0)$, $(j_1, j_2, j_3, 1)$. Každej takej dvojici je priradený informačný vstup E_j, kde $j = j_1 2^2 + j_2 2^1 + j_3 2^0$. Potom vidíme, že pre voľbu pripojenia na vstupe E_j máme štyri možnosti:

(1) Ak $f(j_1, j_2, j_3, 0) = f(j_1, j_2, j_3, 1) = 0$, položíme $E_j = 0$.
(2) Ak $f(j_1, j_2, j_3, 0) = f(j_1, j_2, j_3, 1) = 1$, položíme $E_j = 1$.
(3) Ak $f(j_1, j_2, j_3, 0) = 0, f(j_1, j_2, j_3, 1) = 1$, položíme $E_j = u$.
(4) Ak $f(j_1, j_2, j_3, 0) = 1, f(j_1, j_2, j_3, 1) = 0$, položíme $E_j = \overline{u}$.

Na základe týchto pravidiel sme zostrojili aj logickú sieť, ktorá patrí k funkcii z tabuľky 11. Túto logickú sieť uvádzame na obr. 36. ■

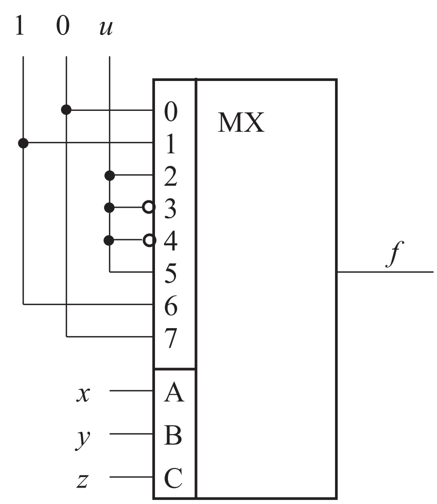
5. Karnaughove mapy

Karnaughova mapa je špecifická tabuľka, slúžiaca na zápis booleovskej funkcie, ktorá nám umožní jednoduché grafické vyznačenie NDF tejto funkcie. Budeme ju definovať rekurentne vzhľadom na počet premenných booleovskej funkcie.

Definícia 2.25. ***Karnaughova mapa*** pre booleovské funkcie n premenných $x_1, \ldots, x_n$ je tabuľka pozostávajúca z 2^n štvorčekov, pričom každému štvorčeku je priradený práve jeden bod z $\mathbf{B}^n$.
Pre $n = 1$ je Karnaughova mapa znázornená na obr. 37a. Štvorčeku, ktorý je zľava vyznačený zvislou čiarou, je priradený bod 1 a štvorčeku nad ním bod 0.
Nech M_n je Karnaughova mapa pre funkcie n premenných $x_1, \ldots, x_n$. Mapa M_{n+1} sa skladá z mapy M_n a jej súmerne združeného obrazu M_n' podľa osi a (obr. 37b) resp. podľa

Tabuľka 11. Funkcia z príkladu 2.38

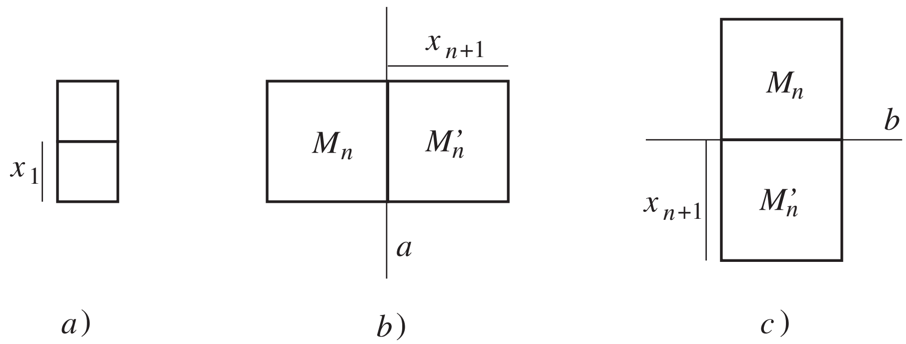
C	B	A	u	f	
0	0	0	0	0	$E_0 = 0$
0	0	0	1	0	
0	0	1	0	1	$E_1 = 1$
0	0	1	1	1	
0	1	0	0	0	$E_2 = u$
0	1	0	1	1	
0	1	1	0	1	$E_3 = \overline{u}$
0	1	1	1	0	
1	0	0	0	1	$E_4 = \overline{u}$
1	0	0	1	0	
1	0	1	0	0	$E_5 = u$
1	0	1	1	1	
1	1	0	0	1	$E_6 = 1$
1	1	0	1	1	
1	1	1	0	0	$E_7 = 0$
1	1	1	1	0	



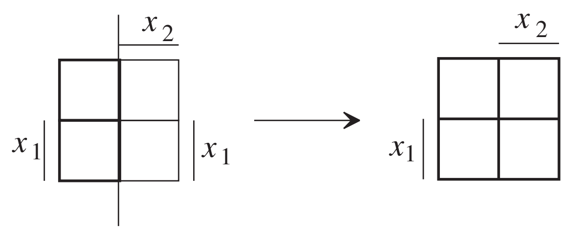
Obr. 36. Logická sieť z príkladu 2.38

osi b (obr. 37c). Vodorovná resp. zvislá kódovacia čiara pri M'_n označená premennou x_{n+1} vyjadruje, že každému štvorčeku z M'_n je priradený bod $(x_1, \dots, x_n, x_{n+1}) \in \mathbf{B}^{n+1}$, pre ktorý $x_{n+1} = 1$, a každému štvorčeku z M_n zas bod, pre ktorý $x_{n+1} = 0$. Kódovacie čiary pri M'_n, rovnobežné s osou, ktoré vznikli súmernosťou podľa tejto osi, sa nevykresľujú.

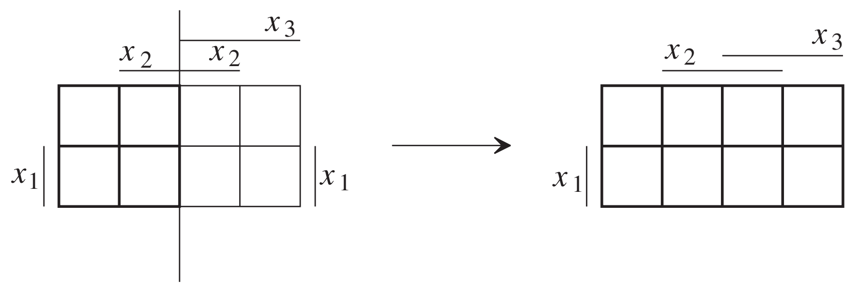
Na obr. 38, 39, 40, 41 a 42 je znázornený spôsob zostavenia Karnaughových máp pre 2, 3, 4, 5 a 6 premenných. Použitie Karnaughových máp pre viac ako 6 premenných je už problematické. Priradenie bodov z $\mathbf{B}^n$ jednotlivým štvorčekom Karnaughovej mapy pre $n = 4$ vidieť na obr. 43.



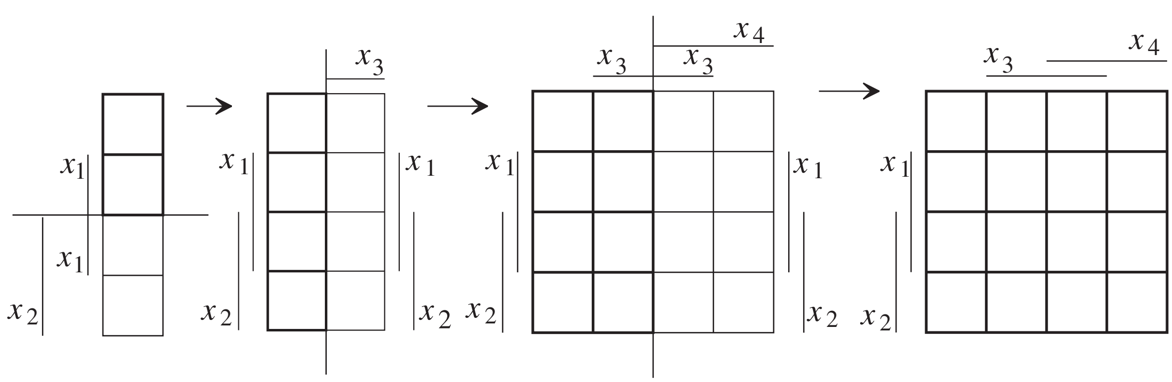
OBR. 37. Karnaughova mapa pre a) jednu premennú, b) $n+1$ premenných



OBR. 38. Karnaughova mapa pre 2 premenné



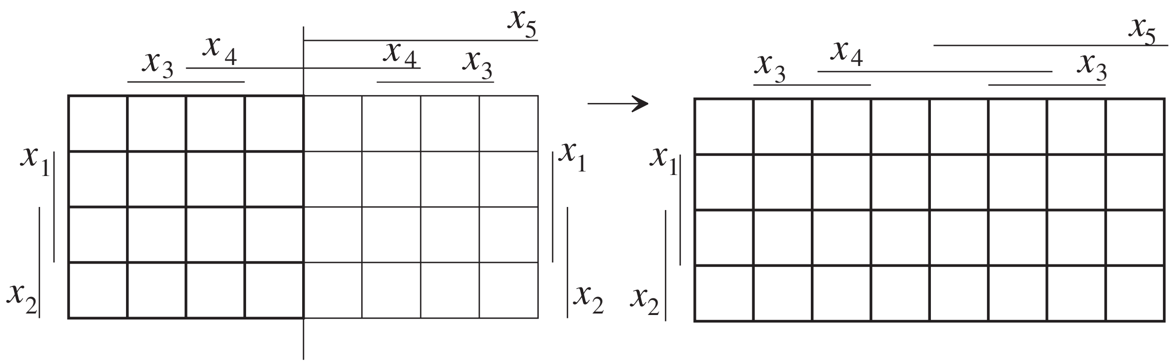
OBR. 39. Karnaughova mapa pre 3 premenné



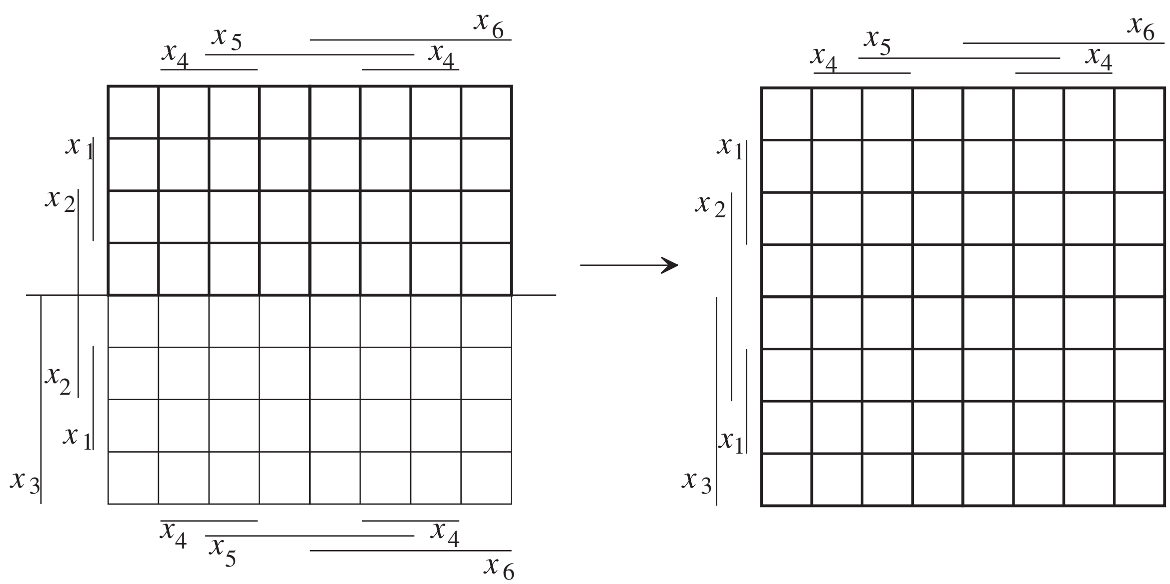
OBR. 40. Karnaughova mapa pre 4 premenné

DEFINÍCIA 2.26. Ak v Karnaughovej mape pre n premenných do každého štvorčeka, ktorý zodpovedá bodu $(a_1, \ldots, a_n) \in \mathbf{B}^n$, vpíšeme hodnotu funkcie $f : \mathbf{B^n} \to \mathbf{B}$ v tomto bode, t.j. $f(a_1, \ldots, a_n)$, získame ***Karnaughovu mapu booleovskej funkcie*** f.

PRÍKLAD 2.39. Na obr. 44 je uvedená tabuľka a tiež Karnaughova mapa booleovskej funkcie f štyroch premenných. ■



Obr. 41. Karnaughova mapa pre 5 premenných



Obr. 42. Karnaughova mapa pre 6 premenných

	x_3		x_4	
	(0,0,0,0)	(0,0,1,0)	(0,0,1,1)	(0,0,0,1)
x_1	(1,0,0,0)	(1,0,1,0)	(1,0,1,1)	(1,0,0,1)
	(1,1,0,0)	(1,1,1,0)	(1,1,1,1)	(1,1,0,1)
x_2	(0,1,0,0)	(0,1,1,0)	(0,1,1,1)	(0,1,0,1)

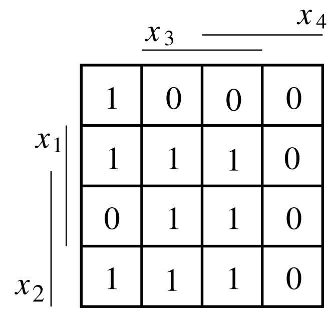
Obr. 43. Body priradené k jednotlivým štvorčekom Karnaughovej mapy

Ukážeme si, ako je možné na Karnaughovej mape funkcie f znázorniť jej NDF. Aby sme to mohli urobiť, bude vhodné zaviesť nasledujúce pojmy.

Definícia 2.27. V normálnej disjunktívnej resp. konjunktívnej forme n premenných $x_1, x_2, \ldots, x_n$ budeme každý výskyt x_i alebo $\overline{x}_i$ pre $i \in \{1, 2, \ldots, n\}$ nazývať ***písmenom***. Počet písmen v súčinovom, resp. súčtovom člene budeme nazývať jeho ***dĺžkou***.

Všimnime si teraz, aké sú množiny jednotkových bodov funkcií určených niektorým súčinovým členom. V tejto súvislosti budeme používať nasledujúce označenie: nech x je

x_1	x_2	x_3	x_4	$f(x_1, x_2, x_3, x_4)$
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	0
0	1	1	0	1
0	1	1	1	1
1	0	0	0	1
1	0	0	1	0
1	0	1	0	1
1	0	1	1	1
1	1	0	0	0
1	1	0	1	0
1	1	1	0	1
1	1	1	1	1



Obr. 44. Tabuľka a Karnaughova mapa funkcie f

premenná a $j \in \{0, 1, *\}$, potom

$$x^j = \begin{cases} x\,, \text{ ak } j = 1 \\ \overline{x}, \text{ ak } j = 0 \\ 1\,, \text{ ak } j = * \end{cases} .$$

Pri tomto označení ľubovoľný súčinový člen n premenných $x_1, \ldots, x_n$ môžeme písať v tvare

$$x_1^{j_1} x_2^{j_2} \ldots x_n^{j_n}.$$

Elementárny súčinový člen $M(x_1, x_2, \ldots, x_n) = x_1^{j_1} x_2^{j_2} \ldots x_n^{j_n}$ (v tomto prípade $j_i \in \{0, 1\}$) má len jeden jednotkový bod, je to n-tica $(j_1, j_2, \ldots, j_n)$. Ak v tomto elementárnom súčinovom člene nahradíme napríklad j_r hodnotou $*$, dostaneme súčinový člen $S(x_1, x_2, \ldots, x_n)$, ktorý vlastne vznikne z $M(x_1, x_2, \ldots, x_n)$ vynechaním písmena x_r alebo $\overline{x}_r$. Tento súčinový člen má dva jednotkové body, sú to n-tice $(j_1, \ldots, j_{r-1}, 0, j_{r+1}, \ldots, j_n)$ a $(j_1, \ldots, j_{r-1}, 1, j_{r+1}, \ldots, j_n)$. Množinu týchto dvoch n-tíc budeme označovať

$$(j_1, \ldots, j_{r-1}, *, j_{r+1}, \ldots, j_n).$$

Ak budeme v týchto úvahách pokračovať ďalej a vynecháme ďalšie písmeno v súčinovom člene $S(x_1, x_2, \ldots, x_n)$, dostaneme súčinový člen dĺžky $n-2$, ktorý má 2^2 jednotkových bodov. Zrejme súčinový člen $x_1^{j_1} x_2^{j_2} \ldots x_n^{j_n}$ n premenných dĺžky $n-i$ má 2^i jednotkových bodov. Množinu týchto bodov budeme označovať symbolom $(j_1, j_2, \ldots, j_n)$, v ktorom na miestach vynechaných písmen bude figurovať znak $*$.

Príklad 2.40. Súčinový člen $\overline{x}_1 x_3 \overline{x}_4 x_6$ šiestich premenných x_1, x_2, x_3, x_4, x_5, x_6 môžeme zapísať v tvare $x_1^0 x_3^1 x_4^0 x_6^1$. Tento súčinový člen má štvorprvkovú množinu jednotkových bodov

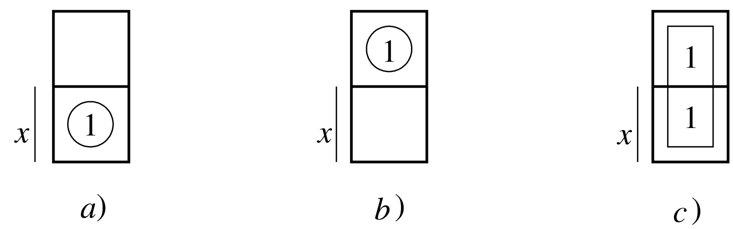
$$(0, *, 1, 0, *, 1) = \{(0,0,1,0,0,1), (0,0,1,0,1,1), (0,1,1,0,0,1), (0,1,1,0,1,1)\}.$$

■

DEFINÍCIA 2.28. Útvar pozostávajúci zo všetkých štvorčekov Karnaughovej mapy zodpovedajúcich jednotkovým bodom súčinového člena n premenných, ktorý má dĺžku k, nazývame 2^{n-k}***-tica jednotkových bodov***. Špeciálne pre $k = n$ ho nazývame ***izolovaný bod***. Pre izolované body, dvojice, štvorice, osmice, ... budeme používať spoločný názov ***konfigurácie jednotkových bodov***. Izolované body budeme na mape vyznačovať krúžkom, ostatné konfigurácie rámčekom.
Voľne budeme hovoriť, že ***súčinový člen pokrýva konfiguráciu jednotkových bodov*** tohto súčinového člena a tiež, že ***súčinový člen pokrýva množinu svojich jednotkových bodov***.

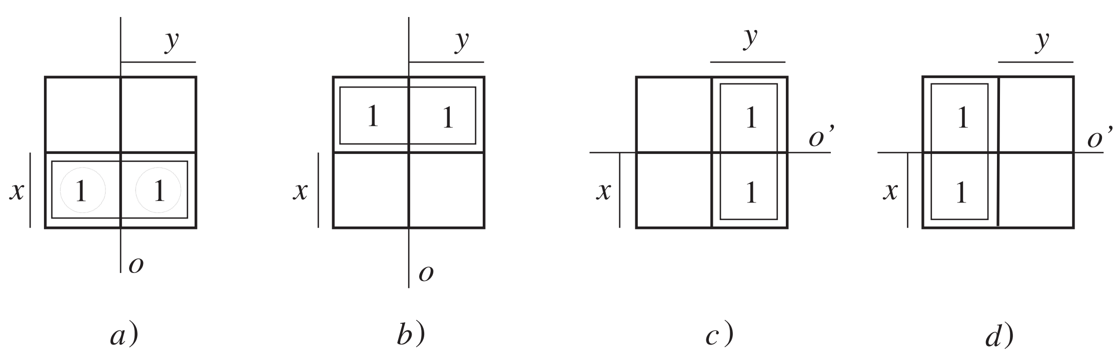
Teraz ukážeme, ako vyzerajú konfigurácie v jednotlivých Karnaughových mapách.

V prípade booleovských funkcií jednej premennej je situácia jednoduchá. Do úvahy prichádzajú iba tri súčinové členy: $x, \overline{x}, 1$. Prvý pokrýva jednoprvkovú množinu (1), druhý (0) a tretí pokrýva dvojprvkovú množinu $(*) = \{1, 0\}$. Konfigurácie zodpovedajúce týmto množinám jednotkových bodov sú na obr. 45 a, b, c.



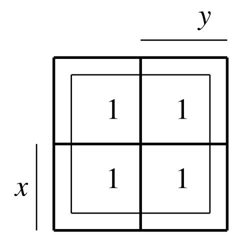
OBR. 45. Konfigurácie jednotkových bodov
a) Izolovaný bod pokrytý súčinovým členom x.
b) Izolovaný bod pokrytý súčinovým členom $\overline{x}$.
c) Dvojica pokrytá súčinovým členom 1.

Pri funkciách dvoch premenných x, y už musíme uvažovať o deviatich súčinových členoch xy, $\overline{x}y$, $x\overline{y}$, $\overline{x}\,\overline{y}$, $x, \overline{x}$, y, $\overline{y}$, 1. Prvé štyri pokrývajú izolované body $(1, 1), (0, 1)$, $(1, 0), (0, 0)$. Súčinový člen x pokrýva množinu jednotkových bodov $(1, *) = \{(1, 0), (1, 1)\}$. Zodpovedajúca dvojica jednotkových bodov je na obr. 46a. Táto dvojica tvorí celý riadok mapy dvoch premenných, v ktorom $x = 1$. Všimnime si ešte, že táto dvojica vznikne spojením dvoch izolovaných bodov (na obr. 46a sú vyznačené čiarkovaným krúžkom), ktoré sú súmerné podľa zvislej osi súmernosti o tejto mapy. Podobne súčinový člen $\overline{x}$ pokrýva dvojicu, ktorá v mape dvoch premenných tvorí riadok, v ktorom $x = 0$ (obr. 46b).



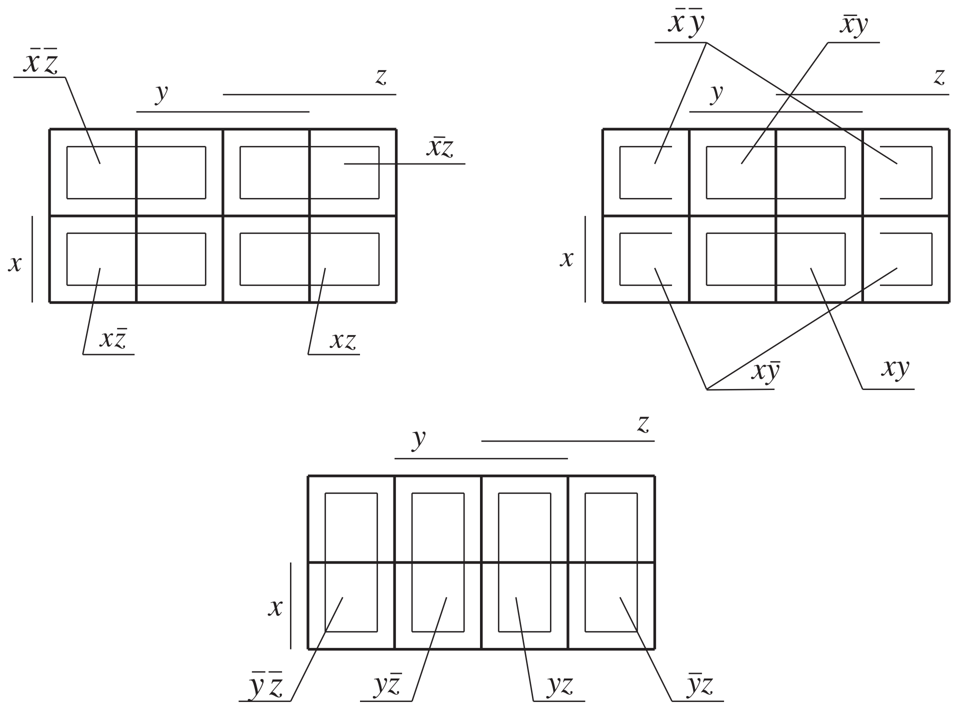
OBR. 46. Dvojice na Karnaughovej mape pre 2 premenné

Konfigurácie jednotkových bodov pokryté súčinovými členmi y a $\overline{y}$ sú nakreslené na obr. 46c, d. Súčinový člen 1 pokrýva množinu $(*, *) = \{(1, 0), (1, 1), (0, 0), (0, 1)\}$. Zodpovedajúca štvorica v Karnaughovej mape dvoch premenných je tvorená všetkými jej štvorčekmi (obr. 47). Všimnime si, že táto štvorica sa skladá z dvoch dvojíc, ktoré sú súmerné podľa jednej z osí súmernosti mapy, pokrytých súčinovými členmi x a $\overline{x}$, resp. y a $\bar{y}$.



Obr. 47. Štvorica na Karnaughovej mape 2 premenných

V prípade funkcií troch premenných x, y, z prichádza do úvahy 27 súčinových členov. Osem z nich sú elementárne súčinové členy, ktoré pokrývajú izolované body. Dvanásť súčinových členov má dĺžku 2. Dvojice, ktoré tieto členy pokrývajú sú na obr. 48. Pri každej dvojici je tam uvedený aj súčinový člen, ktorý túto dvojicu pokrýva.

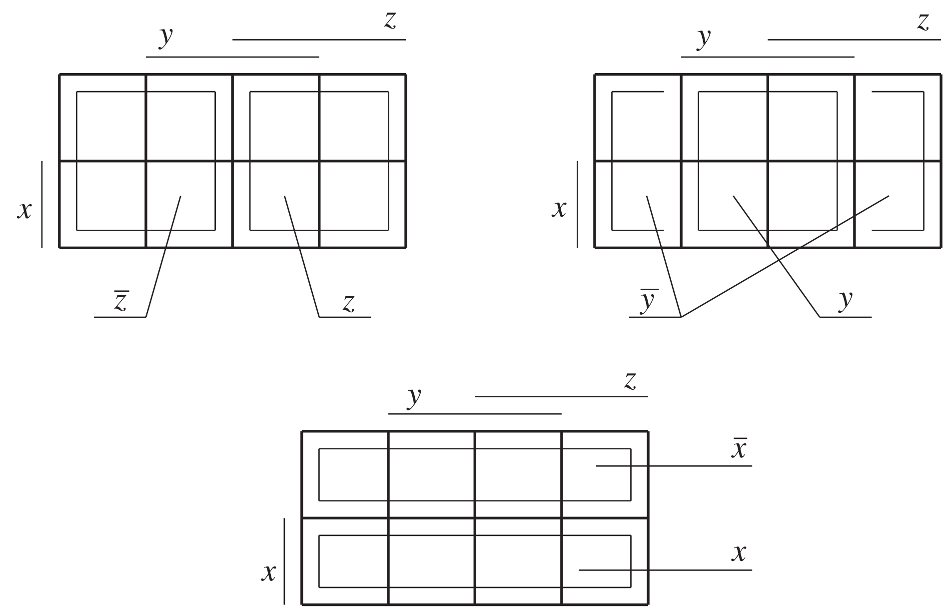


Obr. 48. Dvojice na Karnaughovej mape 3 premenných

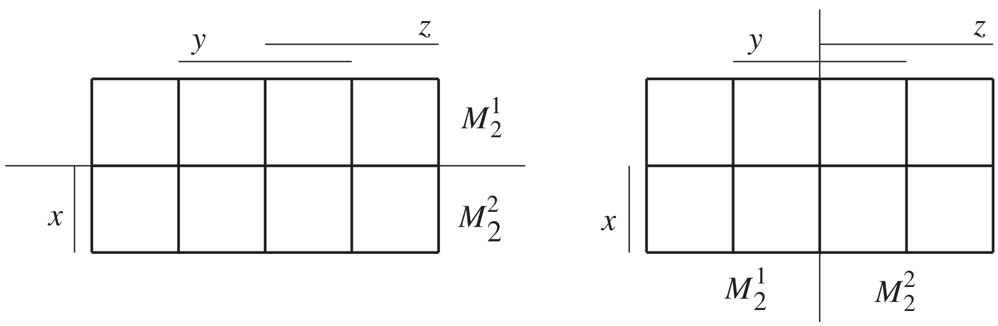
Štvorice pokryté šiestimi súčinovými členmi dĺžky 1 sú zobrazené na obr. 49. Súčinový člen 1 pokrýva osmicu, ktorá je tvorená všetkými štvorčekmi mapy.

Všimnime si aj tu zaujímavú vlastnosť: Karnaughova mapa M_3 je vodorovnou resp. zvislou osou súmernosti rozdelená na dve časti M_2^1 a M_2^2 (obr. 50), ktoré sú vlastne Karnaughovými mapami dvoch premenných. Potom každá 2^k-tica v M_3 je buď 2^k-ticou v jednej z máp M_2^1, M_2^2, alebo je spojením 2^{k-1}-tice v M_2^1 s jej osovo súmerným obrazom v M_2^2.

Dá sa dokázať, že táto vlastnosť platí aj pre ostatné Karnaughove mapy, tak ako je to uvedené v nasledujúcej vete.



Obr. 49. Štvorice na Karnaughovej mape 3 premenných



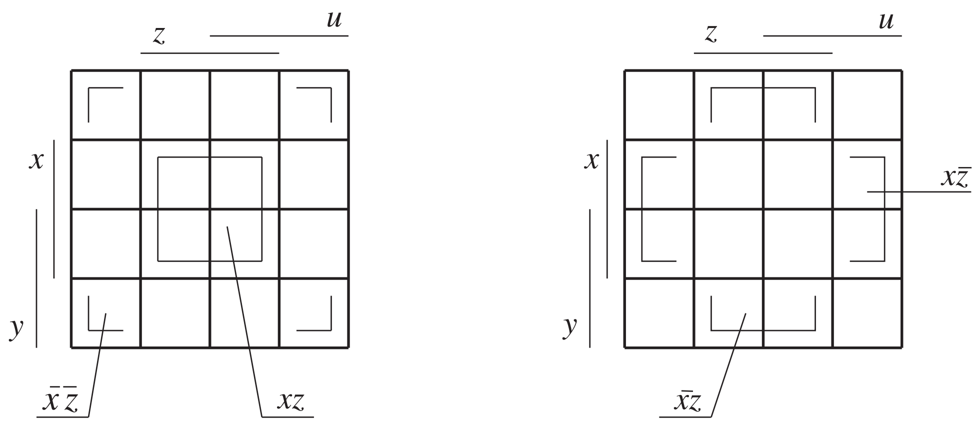
Obr. 50. Podmapy M_2^1 a M_2^2 na Karnaughovej mape M_3

Veta 2.16. (Konštrukcia 2^k-tíc) Nech M_n, $n \geq 2$ je Karnaughova mapa n premenných, ktorá je jej vodorovnou resp. zvislou osou súmernosti rozdelená na dve časti M_{n-1}^1 a M_{n-1}^2 (sú to Karnaughove mapy $n-1$ premenných). Potom každá 2^k-tica, $k \in \{1, \ldots, n\}$ je buď 2^k-ticou v jednej z máp M_{n-1}^1, M_{n-1}^2, alebo je spojením 2^{k-1}-tice v M_{n-1}^1 s jej osovo súmerným obrazom v M_{n-1}^2. □

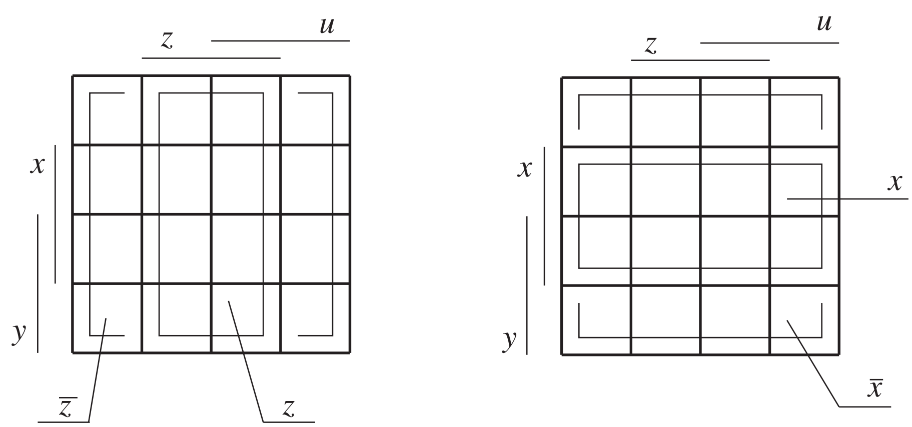
Súčinových členov štyroch premenných je 81. Z toho 16 má dĺžku 4, 32 má dĺžku 3, 24 má dĺžku 2 a 8 má dĺžku 1. Každá dvojica v Karnaughovej mape M_4 je dvojicou v M_3^1 alebo M_3^2 (berieme do úvahy obidve osi súmernosti). Na obr. 51 sú zobrazené práve tie štvorice, ktoré nie sú štvoricami ani v M_3^1 ani v M_3^2. Podobne na obr. 52 sú zobrazené práve tie osmice, ktoré nie sú osmicami ani v jednej z M_3^1, M_3^2.

Pri zobrazovaní konfigurácií na Karnaughových mapách pre 5, resp. 6 premenných využívame predchádzajúcu vetu. Na obr. 53 a 54 sú znázornené niektoré konfigurácie na mapách M_5 a M_6.

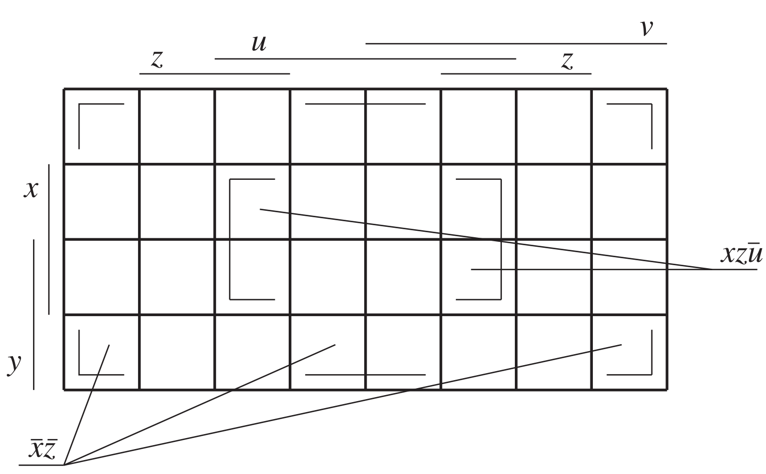
Definícia 2.29. Nech $f(x_1, \ldots, x_n) = \sum_{k=1}^{m} S_k(x_1, \ldots, x_n)$ je booleovská funkcia, pričom $\sum_{k=1}^{m} S_k(x_1, \ldots, x_n)$ je NDF. Túto NDF budeme na Karnaughovej mape funkcie f znázorňovať tak, že na nej vyznačíme konfigurácie jednotkových bodov, ktoré sú pokryté súčinovými členmi $S_k(x_1, \ldots, x_n)$ pre $k \in \{1, \ldots, m\}$.



Obr. 51. Niektoré štvorice na Karnaughovej mape 4 premenných



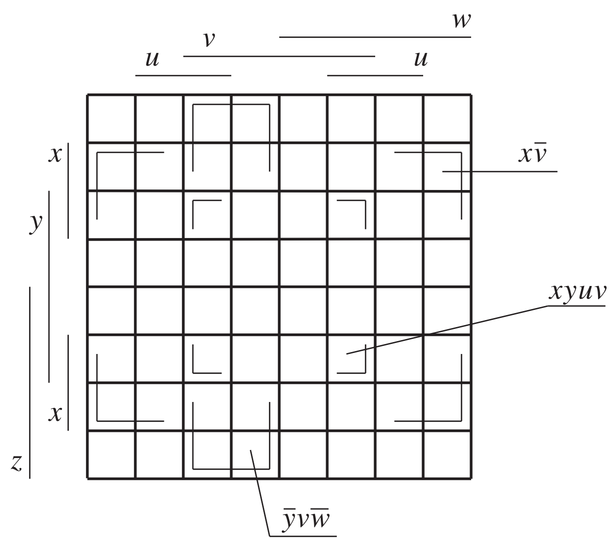
Obr. 52. Niektoré osmice na Karnaughovej mape 4 premenných



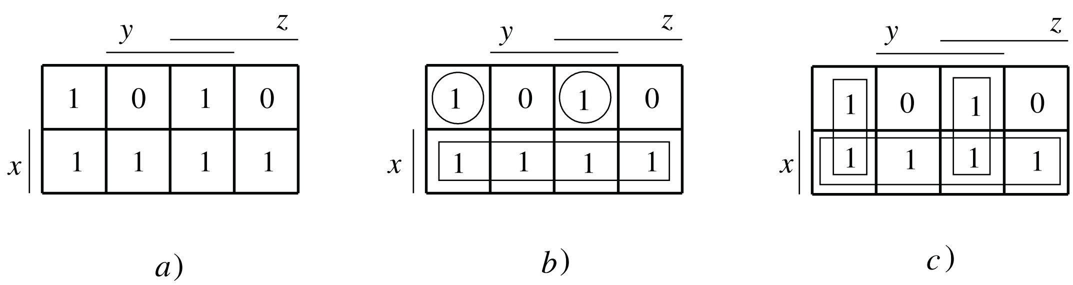
Obr. 53. Niektoré konfigurácie na Karnaughovej mape 5 premenných

Príklad 2.41. Nájdime dve rôzne NDF funkcie $f(x, y, z)$ určenej mapou na obr. 55a.

Riešenie. Vyznačíme na mape tejto funkcie konfigurácie tak, aby každý jednotkový bod funkcie f patril aspoň jednej konfigurácii. Tým je na mape vyznačená práve jedna NDF funkcie f. Na obr. 55b, c sú takto vyznačené konfigurácie, ktoré znázorňujú v poradí tieto NDF funkcie f: $\mathrm{NDF}_1(f) = \overline{x}\,\overline{y}\,\overline{z} + \overline{x}yz + x$, $\mathrm{NDF}_2(f) = \overline{y}\,\overline{z} + yz + x$. ∎



Obr. 54. Niektoré konfigurácie na Karnaughovej mape 6 premenných



Obr. 55. Karnaughova mapa funkcie a dvoch jej NDF

Príklad 2.42. Napíšme NDF funkcie $f : \mathbf{B}^6 \to \mathbf{B}$, ktorá je znázornená na Karnaughovej mape na obr. 56.

Riešenie. Jediný izolovaný bod je pokrytý súčinovým členom $xyz\overline{u}v\overline{w}$. Jedinú dvojicu pokrýva súčinový člen $\overline{x}y\overline{u}vw$. Štvorica v druhom stĺpci je pokrytá súčinovým členom $yu\overline{v}\,\overline{w}$. Štvorica rozložená v prvom a štvrtom riadku a v prvom a štvrtom stĺpci je pokrytá súčinovým členom $\overline{x}\,\overline{z}\,\overline{u}\,\overline{w}$. Jedinú osmicu pokrýva súčinový člen xuw a jedinú šestnásticu pokrýva súčinový člen $\overline{y}\,\overline{v}$. Teda NDF funkcie f danej Karnaughovou mapou na obr. 56 je

$$\mathrm{NDF}\,(f) = xyz\overline{u}v\overline{w} + \overline{x}y\overline{u}vw + yu\overline{v}\,\overline{w} + \overline{x}\,\overline{z}\,\overline{u}\,\overline{w} + xuw + \overline{y}\,\overline{v}.$$

■

Veta 2.17. Nech f je booleovská funkcia n premenných a nech

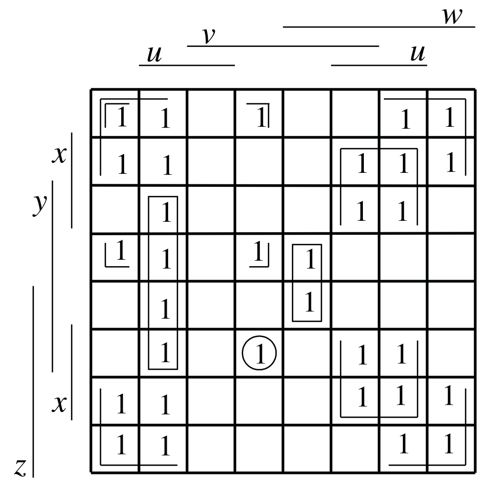
$$U(x_1, x_2, \dots, x_n) = \sum_{k=1}^{m} S_k(x_1, x_2, \dots, x_n)$$

je NDF funkcie $\overline{f}$. Potom

$$V(x_1, x_2, \dots, x_n) = \prod_{k=1}^{m} \widetilde{S}_k(x_1, x_2, \dots, x_n)$$

je NKF funkcie f, pričom ak

$$S_k(x_1, x_2, \dots, x_n) = y_1 y_2 \dots y_n, \quad y_j \in \{x_j, \overline{x}_j, 1\}$$



Obr. 56. NDF funkcie z príkladu 2.42

tak

$$\widetilde{S}_k(x_1, x_2, \dots, x_n) = \overline{y}_1 + \overline{y}_2 + \cdots + \overline{y}_n.$$

Dôkaz. Zrejme $V(x_1, x_2, \dots, x_n)$ je normálna konjunktívna forma. Stačí už len dokázať, že je to NKF funkcie f :

$$f(x_1, x_2, \dots, x_n) = \overline{\overline{f\,(x_1, x_2, \dots, x_n)}} = \overline{\sum_{k=1}^{m} S_k(x_1, x_2, \dots, x_n)} = \prod_{k=1}^{m} \overline{S_k(x_1, x_2, \dots, x_n)} = $$
$$= \prod_{k=1}^{m} \widetilde{S}_k(x_1, x_2, \dots, x_n).$$

□

Príklad 2.43. Booleovská funkcia $g : \mathbf{B}^3 \to \mathbf{B}$ je daná množinou jednotkových bodov $J(g) = \{(0,0,1), (0,1,0)\}$. Určte NKF (nie však UNKF) funkcie g.

Riešenie. Množina jednotkových bodov funkcie $\overline{g}$ je

$$J(\overline{g}) = \mathbf{B}^3 \setminus J(g) = \{(0,0,0), (0,1,1), (1,0,0), (1,0,1), ((1,1,0), (1,1,1)\}.$$

Karnaughova mapa funkcie $\overline{g}$ je na obr. 55a a jedna NDF tejto funkcie je znázornená v Karnaughovej mape na obr. 55c, odkiaľ vyplýva, že

$$g(x, y, z) = \overline{\overline{y}\,\overline{z} + yz + x} = (y + z)(\overline{y} + \overline{z})\overline{x}.$$

Posledný B-výraz je NKF funkcie g. ■

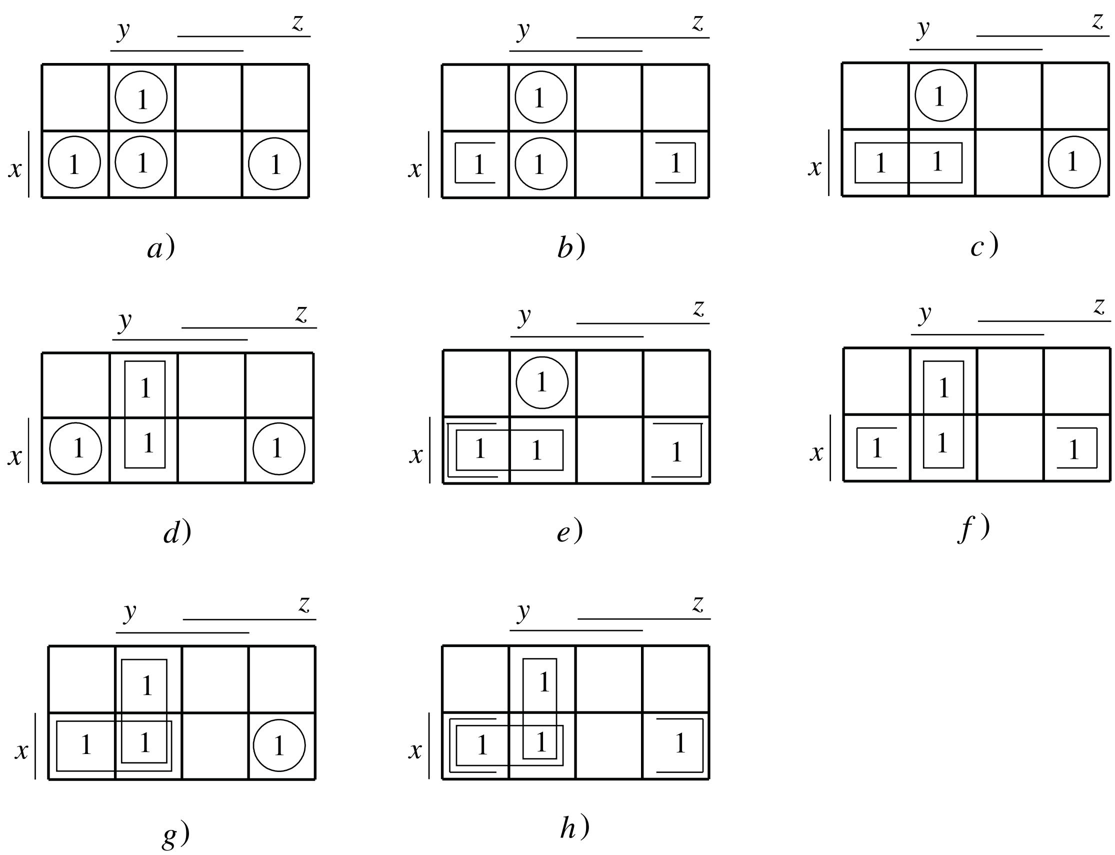
6. Minimalizácia B-výrazov

V tejto časti bude naším cieľom vyjadriť booleovskú funkciu pomocou NDF resp. NKF, ktoré majú najmenší počet písmen. Pretože normálnych disjunktívnych (konjunktívnych) foriem n premenných je iba konečný počet, je zrejmé, že táto úloha má vždy riešenie (aj keď nemusí byť jediné). Stačí „len“ zobrať všetky NDF (NKF) n premenných, z nich vybrať tie, ktoré určujú danú funkciu a spomedzi nich vybrať normálne disjunktívne (konjunktívne) formy s najmenším počtom písmen. Pravda, už pre $n = 4$ dostávame $2^{3^4} = 2^{81} = (2^{10})^8 \cdot 2 \doteq 2 \cdot (10^3)^8 = 2 \cdot 10^{24}$ rozličných NDF, a teda vidíme, že tento postup je prakticky nepoužiteľný. Preto budeme postupovať inak.

Definícia 2.30. Normálnu disjunktívnu (konjunktívnu) formu booleovskej funkcie f, ktorá má spomedzi všetkých normálnych disjunktívnych (konjunktívnych) foriem funkcie f najmenší počet písmen, budeme nazývať ***minimálna normálna disjunktívna (konjunktívna) forma booleovskej funkcie*** f, skrátene MNDF (MNKF).

Príklad 2.44. Nájdime MNDF funkcie $f(x,y,z) = (x+y)\overline{z} + x\overline{y}$.

Riešenie. Pomocou Karnaughovej mapy funkcie f vytvoríme všetky tie NDF tejto funkcie, ktoré majú túto vlastnosť: pre každé dva rôzne súčinové členy S_1, S_2 z tej istej NDF platí $J(S_1) \not\subset J(S_2)$ a $J(S_2) \not\subset J(S_1)$. Táto vlastnosť nám zabezpečí, že v NDF sa nevyskytnú mnohé nadbytočné členy. Ak by totiž platilo napr. $J(S_1) \subset J(S_2)$, tak v NDF môžeme S_1 vynechať a dostaneme opäť NDF funkcie f, ktorá však má menej písmen. Z takto vytvorených NDF vyberieme minimálnu. Na obr. 57a až h sú na Karnaughových mapách znázornené všetky NDF funkcie f majúce uvedenú vlastnosť. Napíšme jednotlivé



Obr. 57. NDF funkcie f z príkladu 2.44

NDF a vyhľadajme z nich tú, ktorá má najmenší počet písmen:

$a)$ $f(x,y,z) = \overline{x}y\overline{z} + x\overline{y}\,\overline{z} + xy\overline{z} + x\overline{y}z$, 12 písmen;
$b)$ $f(x,y,z) = \overline{x}y\overline{z} + xy\overline{z} + x\overline{y}$, 8 písmen;
$c)$ $f(x,y,z) = \overline{x}y\overline{z} + x\overline{y}z + x\overline{z}$, 8 písmen;
$d)$ $f(x,y,z) = x\overline{y}\,\overline{z} + x\overline{y}z + y\overline{z}$, 8 písmen;
$e)$ $f(x,y,z) = \overline{x}y\overline{z} + x\overline{y} + x\overline{z}$, 7 písmen;
$f)$ $f(x,y,z) = y\overline{z} + x\overline{y}$, 4 písmená;
$g)$ $f(x,y,z) = y\overline{z} + x\overline{z} + x\overline{y}z$, 7 písmen;
$g)$ $f(x,y,z) = y\overline{z} + x\overline{y} + x\overline{z}$, 6 písmen.

Vidíme, že MNDF $(f) = y\overline{z} + x\overline{y}$. ■

Takýto postup vyhľadávania MNDF je dosť nepraktický a často nepoužiteľný. Preto budeme postupovať inak.

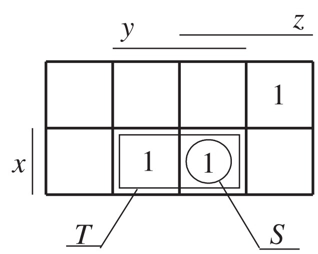
Definícia 2.31. Nech f, g sú dve booleovské funkcie n premenných. Budeme hovoriť, že funkcia f je ***implikant funkcie*** g, ak $J(f) \subset J(g)$. V súlade s touto terminológiou, ak je funkcia f daná B-výrazom $U(x_1, x_2, \ldots, x_n)$ a funkcia g B-výrazom $V(x_1, x_2, \ldots, x_n)$, budeme hovoriť, že B-výraz $U(x_1, x_2, \ldots, x_n)$ je ***implikant B-výrazu*** $V(x_1, x_2, \ldots, x_n)$. V tomto prípade budeme tiež hovoriť, že B-výraz $U(x_1, x_2, \ldots, x_n)$ je ***implikant funkcie*** g.

Z definície implikanta vyplýva, že každý súčinový člen z NDF funkcie f je implikantom funkcie f.

Definícia 2.32. Súčinový člen $S(x_1, x_2, \ldots, x_n) = x_1^{j_1} x_2^{j_2} \ldots x_n^{j_n}$, ktorý je implikantom booleovskej funkcie $f : \mathbf{B}^n \to \mathbf{B}$, sa nazýva ***prostý implikant*** (***prvoimplikant***) ***funkcie*** f, ak po nahradení hociktorého písmena vyskytujúceho sa v S jednotkou, dostaneme súčinový člen, ktorý už nie je implikantom funkcie f.

Príklad 2.45. Nech $f(x, y, z) = xyz + xy\overline{z} + \overline{x}\,\overline{y}z$. Je zrejmé, že $S(x, y, z) = xyz$ je implikant tejto funkcie. Budeme vyšetrovať, či je to aj prostý implikant.

Riešenie. Súčinový člen xyz pokrýva izolovaný bod $(1, 1, 1)$. Ak v S nahradíme písmeno z jednotkou, dostaneme súčinový člen $T(x, y, z) = xy$. Ako vidieť aj z Karnaughovej mapy funkcie f (obr.58), pre množiny jednotkových bodov platí $J(S) \subset J(T) \subset J(f)$. Preto podľa definície súčinový člen T je implikant funkcie f a S nie je prostý implikant tejto funkcie. Funkciu f zrejme môžeme vyjadriť v tvare $f\,(x, y, z) = xy + xy\overline{z} + \overline{xy}z$. ∎



Obr. 58. Implikanty T a S funkcie f z príkladu 2.45

Príklad 2.46. Zistime, či súčinový člen xy je prostý implikant funkcie $f(x, y, z) = xy + xy\overline{z} + \overline{x}\,\overline{y}z$.

Riešenie. Otázka je, či môžeme z člena xy vynechať buď x alebo y a či pritom zvyšok zostane implikantom funkcie f. Odpoveď je jednoduchá, ak si uvedomíme, aké sú množiny jednotkových bodov funkcie f a súčinových členov $S(x, y, z) = y,\ T(x, y, z) = x$. Zrejme

$$J(S) = (*, 1, *) = \{(0,1,0), (0,1,1), (1,1,0), (1,1,1)\},,$$
$$J(T) = (1, *, *) = \{(1,0,0), (1,0,1), (1,1,0), (1,1,1)\},$$
$$J(f) = \{(1,1,1), (1,1,0), (0,0,1)\}.$$

Vidíme, že $J(S) \not\subset J(f), J(T) \not\subset J(f)$. To znamená, že S, T nie sú implikanty funkcie f, a preto súčinový člen xy je prostý implikant funkcie f. ∎

Vyhľadávať prosté implikanty spôsobom uvedeným v predchádzajúcich dvoch príkladoch je nepraktické. Pritom prosté implikanty majú pri hľadaní MNDF funkcie dôležitú úlohu.

VETA 2.18. Nech $U(x_1, x_2, \dots, x_n) = \sum_{i=1}^{m} S_i(x_1, x_2, \dots, x_n)$ je MNDF funkcie $f : \mathbf{B}^n \to \mathbf{B}$. Potom každý súčinový člen $S_i(x_1, x_2, \dots, x_n)$ pre $i \in \{1, 2, \dots, m\}$ je prostým implikantom funkcie f.

Dôkaz. Pre každé $x = (x_1, x_2, \dots, x_n) \in \mathbf{B}^n$ je $f(x) = U(x) = S_1(x) + S_2(x) + \cdots + +S_m(x)$. Nech pre $i \in \{1, 2, \dots, m\}$ súčinový člen $S_i(x)$ obsahuje n_i písmen. Potom MNDF $U(x)$ obsahuje $k = n_1 + n_2 + \cdots + n_m$ písmen, čo je najmenší počet písmen zo všetkých NDF funkcie f. Predpokladajme teraz, že niektorý súčinový člen $S_i(x)$ nachádzajúci sa v MNDF $U(x)$ nie je prostým implikantom funkcie f. Teda v súčinovom člene $S_i(x)$ existuje také písmeno, že po jeho nahradení jednotkou dostaneme súčinový člen $T_i(x)$, ktorý je opäť implikantom funkcie f. To však znamená, že platí

$$J(S_i) \subset J(T_i) \subset J(\,f\,).$$

Z tejto vlastnosti vyplýva

$$J(S_i + T_i) = J(S_i) \cup J(T_i) = J(T_i),$$
$$J(T_i + f) = J(T_i) \cup J(f) = J(f),$$

a preto platí

$$S_i(x) + T_i(x) = T_i(x),$$
$$T_i(x) + f(x) = f(x).$$

Na základe týchto vlastností dostávame

$$f(x) = T_i(x) + f(x) = T_i(x) + S_1(x) + \cdots + S_{i-1}(x) + S_i(x) + S_{i+1}(x) + \cdots + S_m(x) =$$
$$= S_1(x) + \cdots + S_{i-1}(x) + T_i(x) + S_{i+1}(x) + \cdots + S_m(x).$$

Získali sme NDF funkcie f, ktorá má

$$n_1 + \cdots + n_{i-1} + (n_i - 1) + n_{i+1} + \cdots + n_m = k - 1$$

písmen. To je spor s minimálnosťou čísla k. Preto každý súčinový člen v MNDF danej funkcie musí byť prostým implikantom. □

Z predchádzajúcej vety vyplýva, že pri hľadaní MNDF danej booleovskej funkcie môžeme postupovať takto:

1. Nájdeme všetky prosté implikanty danej funkcie.
2. Z nájdených prostých implikantov zostavíme všetky NDF danej funkcie.
3. Spomedzi zostavených NDF vyberieme všetky MNDF.

Pri riešení tejto úlohy je možné použiť Quine - McCluskeyov algoritmus (je uvedený napr. v [**1**]), v ktorom sa vychádza z UNDF danej funkcie. My si ukážeme, ako je možné riešiť túto úlohu pomocou Karnaughových máp. Tu si treba uvedomiť, že *prostý implikant funkcie f je súčinový člen, ktorý pokrýva 2^k-ticu jednotkových bodov funkcie f, pričom táto 2^k-tica nie je časťou žiadnej 2^{k+1}-tice jednotkových bodov funkcie f.*

PRÍKLAD 2.47. Všetky prosté implikanty funkcie $f(x, y, z) = (x + y)\overline{z} + x\overline{y}$ sú vyznačené na Karnaughovej mape na obr. 57h. Všimnime si, že NDF $y\overline{z} + x\overline{y} + x\overline{z}$, ktorá obsahuje všetky prosté implikanty funkcie f, nie je v tomto prípade MNDF tejto funkcie. ■

DEFINÍCIA 2.33. Normálna disjunktívna forma booleovskej funkcie f, ktorá je súčtom všetkých prostých implikantov tejto funkcie, sa nazýva ***skrátená normálna disjunktívna forma funkcie*** f, skrátene SNDF.

Je zrejmé, že SNDF vždy existuje a je až na poradie prostých implikantov jednoznačne určená. Ako sme videli v predošlom príklade, SNDF nemusí byť MNDF.

Definícia 2.34. Nech $U(x_1, x_2, \dots, x_n)$ je NDF booleovskej funkcie $f : \mathbf{B}^n \to \mathbf{B}$, pre ktorú platí:

1. Každý súčinový člen, ktorý sa nachádza v tejto NDF (s koeficientom rovnajúcim sa 1), je prostý implikant funkcie f.
2. Ak v danej NDF vynecháme hociktorý súčinový člen, dostaneme B-výraz, ktorý už neurčuje funkciu f.

V takomto prípade budeme hovoriť, že $U(x_1, x_2, \dots, x_n)$ je ***iredundantná normálna disjunktívna forma funkcie*** f, skrátene INDF.

Uvedomme si, že pre každý prostý implikant nachádzajúci sa v INDF, musí existovať jednotkový bod funkcie, ktorý je pokrytý len týmto jediným prostým implikantom spomedzi všetkých implikantov nachádzajúcich sa v INDF (v opačnom prípade by sme tento prostý implikant mohli z INDF vynechať). Z definície INDF a MNDF ďalej vyplýva, že MNDF danej funkcie je zároveň INDF tejto funkcie.

Nájsť INDF danej funkcie pomocou Karnaughovej mapy, znamená vybrať také konfigurácie jednotkových bodov funkcie f, že

1. *každý jednotkový bod funkcie sa musí nachádzať v aspoň jednej vybranej konfigurácii,*
2. *žiadna vybraná konfigurácia nie je časťou ľubovoľnej inej konfigurácie jednotkových bodov funkcie f,*
3. *pre každú vybranú konfiguráciu existuje jednotkový bod funkcie, ktorý sa nachádza iba v tejto konfigurácii.*

MNDF funkcie f je jej INDF s najmenším počtom písmen.

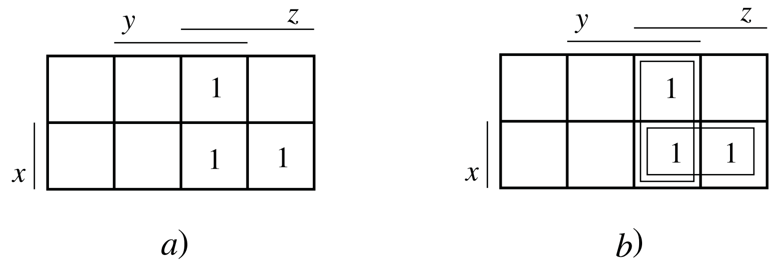
Pri vyhľadávaní INDF danej funkcie, je vhodné v prvom rade uvažovať o prostých implikantoch, ktoré sa musia nachádzať v každej INDF tejto funkcie. Sú to takéto prosté implikanty:

Definícia 2.35. Prostý implikant $P(x_1, x_2, \dots, x_n)$ funkcie $f : \mathbf{B}^n \to \mathbf{B}$ sa nazýva ***nevyhnutný prostý implikant funkcie*** f, ak existuje jednotkový bod funkcie f, ktorý je jednotkovým bodom jedine prostého implikanta $P(x_1, x_2, \dots, x_n)$ spomedzi všetkých prostých implikantov tejto funkcie.
NDF, ktorá je súčtom všetkých nevyhnutných prostých implikantov funkcie f, nazývame ***jadro funkcie*** f.

Príklad 2.48. Nájdime jadro funkcie

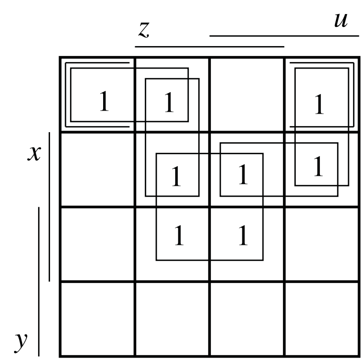
(1) $f(x, y, z) = \overline{x}yz + xyz + x\overline{y}z$,
(2) $f\,(x, y, z, u) = \overline{x}\,\overline{y}\,\overline{u} + \overline{y}z\overline{u} + x\overline{y}u + \overline{y}\,\overline{z}u + xyz$.

Riešenie. 1. Karnaughova mapa funkcie f je na obr. 59a. Jednotka v prvom riadku a treťom stĺpci sa nachádza v dvojici (na obr. 59b je v treťom stĺpci), ktorá nie je časťou žiadnej štvorice jednotkových bodov funkcie f. Táto dvojica je preto pokrytá prostým implikantom a je ním súčinový člen yz. Uvedený jednotkový bod sa nenachádza už v žiadnej inej konfigurácii pokrytej prostým implikantom, preto yz je nevyhnutný prostý implikant. Podobne jednotkový bod v druhom riadku a štvrtom stĺpci sa nachádza v jedinej konfigurácii, ktorá je pokrytá prostým implikantom. Je to dvojica v druhom riadku (obr. 59b), ktorá je pokrytá nevyhnutným prostým implikantom xz. B-výraz $U(x, y, z) = yz + xz$ je teda jadro funkcie f. Keďže vyznačené dvojice obsahujú všetky jednotkové body funkcie f, je $U(x, y, z)$ jedinou INDF tejto funkcie, a teda aj MNDF.



OBR. 59. a) Karnaughova mapa funkcie f
b) prosté implikanty funkcie f

2. Jednotkový bod v 1. riadku a 1. stĺpci Karnaughovej mapy funkcie f na obr. 60 sa nachádza v dvoch rôznych dvojiciach pokrytých prostými implikantmi $\overline{x}\,\overline{y}\,\overline{z}$ a $\overline{x}\,\overline{y}\,\overline{u}$. Podobne každý jednotkový bod v 1. a 2. riadku je aspoň v dvoch konfiguráciách pokrytých prostými implikantmi. Až jednotkové body v treťom riadku sú v jedinej konfigurácii (je to štvorica) pokrytej prostým implikantom xz. Toto je teda jediný nevyhnutný prostý



OBR. 60. Prosté implikanty funkcie f

implikant. Jadrom funkcie f je preto B-výraz $V(x,y,z,u) = xz$. V tomto prípade jadro nie je NDF funkcie f, lebo množiny jednotkových bodov $J(V)$ a $J(f)$ sú rôzne. ■

Nevyhnutné prosté implikanty funkcie nemusia existovať. Preto jadro funkcie sa môže rovnať nule. Ďalej je zrejmé, že každá INDF (a teda aj MNDF) musí obsahovať jadro funkcie. Ďalšie dopĺňanie jadra na INDF robíme pomocou voľby jednotlivých alternatív výberu prostých implikantov, pričom vo všeobecnosti treba preskúmať všetky možné alternatívy.

PRÍKLAD 2.49. Nájdime MNDF funkcie $f(x,y,z,u) = \overline{x}\,\overline{y}\,\overline{u} + \overline{y}z\overline{u} + x\overline{y}u + \overline{y}\,\overline{z}u + xyz$ (funkcia z druhej časti predošlého príkladu).

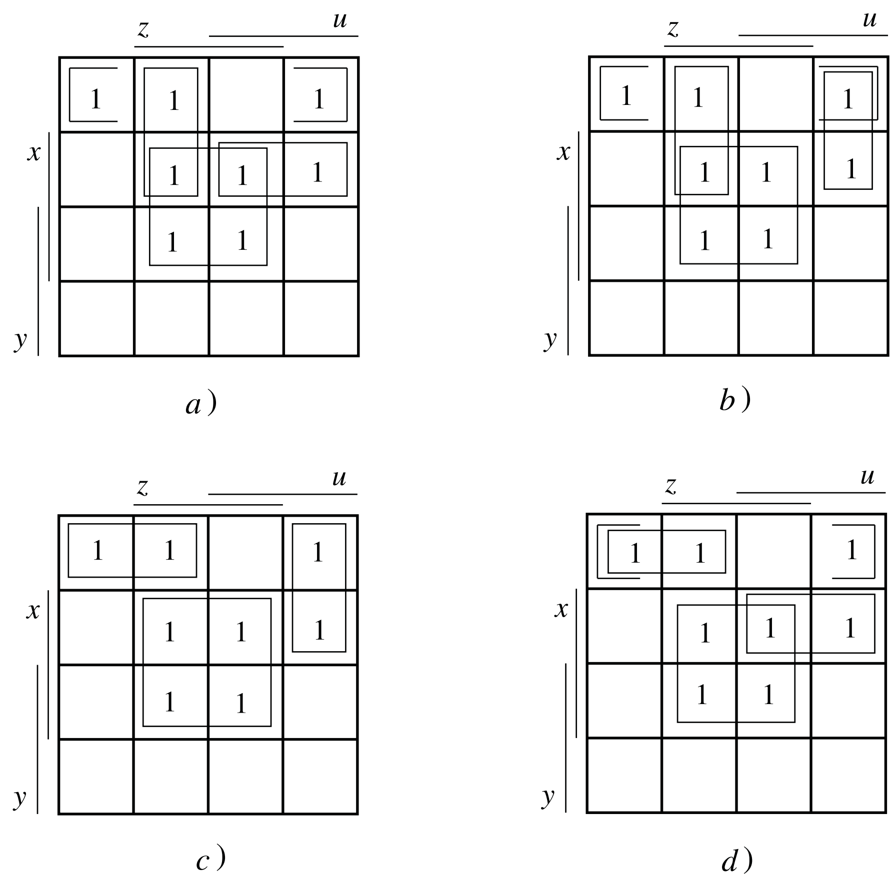
Riešenie. Na obr. 60 sú znázornené všetky prosté implikanty funkcie f. Odtiaľ vyplýva, že

$$U(x,y,z,u) = \overline{x}\,\overline{y}\,\overline{z} + \overline{x}\,\overline{y}\,\overline{u} + \overline{y}z\overline{u} + x\overline{y}u + \overline{y}\,\overline{z}u + xz$$

je SNDF tejto funkcie. Nie je to však INDF, pretože napr. prostý implikant $x\overline{y}u$ pokrýva množinu jednotkových bodov, z ktorých každý sa nachádza aj v inej konfigurácii jednotkových bodov funkcie f. Už vieme, že štvorica je nevyhnutná, preto sa bude vyskytovať v každej Karnaughovej mape, na ktorej je vyznačená INDF funkcie f. Jednotkový bod v prvom riadku a prvom stĺpci mapy, t.j. $(0,0,0,0)$, je pokrytý dvoma rôznymi prostými implikantmi.

1. Do INDF vyberme z nich najprv $\overline{x}\,\overline{y}\,\overline{z}$ a nevyberme $\overline{x}\,\overline{y}\,\overline{u}$ (obr. 61a, b). Potom jednotkový bod $(0,0,1,0)$ (v prvom riadku a druhom stĺpci) musíme pokryť prostým implikantom $\overline{y}z\overline{u}$. Ostáva nám pokryť už len jednotkový bod $(1,0,0,1)$ (druhý riadok, štvrtý stĺpec), čo môžeme urobiť prostým implikantom $x\overline{y}u$ (obr. 61a) alebo $\overline{y}\,\overline{z}u$ (obr. 61b). Dostávame tak dve INDF funkcie f:

$$\mathrm{INDF}_1(f) = xz + \overline{x}\,\overline{y}\,\overline{z} + \overline{y}z\overline{u} + x\overline{y}u,$$
$$\mathrm{INDF}_2(f) = xz + \overline{x}\,\overline{y}\,\overline{z} + \overline{y}z\overline{u} + \overline{y}\,\overline{z}u.$$



Obr. 61. INDF funkcie f

2. Vyberme teraz $\overline{x}\,\overline{y}\,\overline{u}$ a nevyberme $\overline{x}\,\overline{y}\,\overline{z}$ (obr. 61c). V takom prípade jednotkový bod $(0,0,0,1)$ (prvý riadok, štvrtý stĺpec) môžeme pokryť iba prostým implikantom $\overline{y}\,\overline{z}u$. A keďže každý jednotkový bod je pokrytý niektorým vybraným prostým implikantom, získali sme ďalšiu INDF:

$$\mathrm{INDF}_3(f) = xz + \overline{x}\,\overline{y}\,\overline{u} + \overline{y}\,\overline{z}u.$$

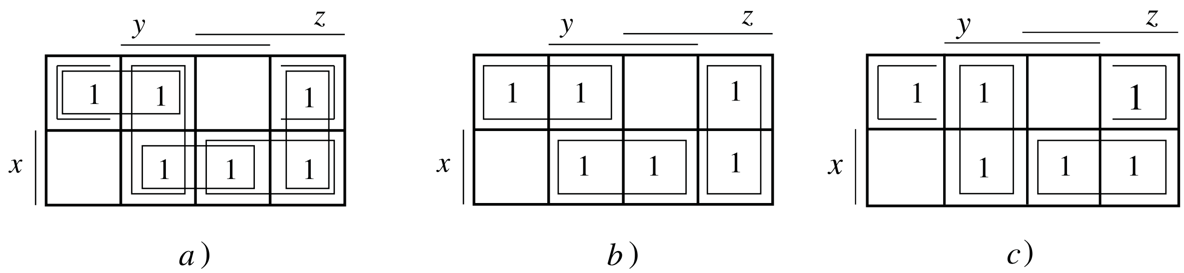
3. Do INDF vyberme teraz obidva prvoimplikanty $\overline{x}\,\overline{y}\,\overline{z}$ aj $\overline{x}\,\overline{y}\,\overline{u}$ (obr. 61d). Potrebujeme ešte pokryť jednotkový bod $(1,0,0,1)$, čo je možné urobiť prvoimplikantom $\overline{y}\,\overline{z}u$ alebo $x\overline{y}u$. Ak by sme vybrali prvý z nich, stal by sa prostý implikant $\overline{x}\,\overline{y}\,\overline{z}$ nadbytočný, lebo jeho jednotkové body by boli pokryté inými vybranými prostými implikantmi $(\overline{x}\,\overline{y}\,\overline{u}, \overline{y}\,\overline{z}u)$. Vyberieme teda $x\overline{y}u$ a dostávame štvrtú INDF:

$$\mathrm{INDF}_4(f) = xz + \overline{x}\,\overline{y}\,\overline{u} + \overline{x}\,\overline{y}\,\overline{z} + x\overline{y}u.$$

MNDF funkcie f je zrejme $\mathrm{INDF}_3(f) = xz + \overline{x}\,\overline{y}\,\overline{u} + \overline{y}\,\overline{z}u$, čo sa dalo v tomto prípade odhadnúť z Karnaughovej mapy priamo, bez hľadania všetkých INDF. ■

Príklad 2.50. Nájdime MNDF funkcie $f(x, y, z) = \overline{x}\,\overline{z} + y\overline{z} + \overline{x}\,\overline{y}z + xz + xyz$.

Riešenie. SNDF tejto funkcie je zobrazená na obr. 62a. Zároveň je tu vidieť, že jadro funkcie je nulové, lebo každý jednotkový bod sa nachádza v dvoch dvojiciach, ktoré sú pokryté prostými implikantmi. Preto, ak chceme nájsť INDF, musíme aspoň jednu dvojicu vynechať.



Obr. 62. SNDF a dve INDF funkcie f z príkladu 2.50

1. Vynechajme dvojicu pokrytú členom $\overline{x}\,\overline{y}$. Potom musíme vybrať dvojicu pokrytú členom $\overline{x}\,\overline{z}$ a $\overline{y}z$ (obr. 62b). Aby sme získali INDF, stačí už len k dvom vybraným dvojiciam pridať dvojicu pokrytú prostým implikantom xy alebo dve dvojice pokryté prvoimplikantmi $y\overline{z}$, xz. Nájdené INDF sú

$$\begin{aligned} \mathrm{INDF}_1(f) &= \overline{x}\,\overline{z} + \overline{y}z + xy, \\ \mathrm{INDF}_2(f) &= \overline{x}\,\overline{z} + \overline{y}z + y\overline{z} + xz. \end{aligned}$$

Keď na začiatku vynecháme prostý implikant xz získame opäť $\mathrm{INDF}_1(f)$ a

$$\mathrm{INDF}_3(f) = \overline{x}\,\overline{y} + y\overline{z} + xy + \overline{y}z.$$

Podobne, keď na začiatku vynecháme prostý implikant $y\overline{z}$, dostaneme opäť $\mathrm{INDF}_1(f)$ a

$$\mathrm{INDF}_4(f) = \overline{x}\,\overline{z} + xy + xz + \overline{x}\,\overline{y}.$$

2. Vynechajme dvojicu pokrytú členom $\overline{x}\,\overline{z}$. Potom musíme do Karnaughovej mapy zaradiť dvojice pokryté implikantmi $\overline{x}\,\overline{y}$ a $y\overline{z}$ (obr. 62c). Teraz už stačí pridať dvojicu pokrytú prostým implikantom xz a dostaneme

$$\mathrm{INDF}_5(f) = \overline{x}\,\overline{y} + y\overline{z} + xz,$$

alebo dve dvojice pokryté prvoimplikantmi xy, $\overline{y}z$ a dostaneme $\mathrm{INDF}_3(f)$.
Keď na začiatku vynecháme prostý implikant xy, dostaneme $\mathrm{INDF}_5(f)$ a $\mathrm{INDF}_2(f)$.
Podobne, keď na začiatku vynecháme prvoimplikant $\overline{y}z$, dostaneme $\mathrm{INDF}_5(f)$ a $\mathrm{INDF}_4(f)$.

Najmenší počet písmen (6) majú $\mathrm{INDF}_1(f)$ a $\mathrm{INDF}_5(f)$, sú to teda MNDF danej funkcie. ■

Na hľadanie minimálnej normálnej konjunktívnej formy by sa dala použiť podobná metóda ako na hľadanie MNDF. Urobíme to však inakšie. Využijeme vzťah medzi MNKF funkcie f a MNDF funkcie $\overline{f}$. Všimnime si najprv fakt: Ak

$$S(x_1, x_2, \ldots, x_n) = y_1 y_2 \ldots y_n, \text{ kde } y_i \in \{x_i, \overline{x}_i, 1\},$$

je súčinový člen, tak jeho komplement

$$\overline{S(x_1, x_2, \ldots, x_n)} = \overline{y_1 y_2 \ldots y_n}$$

je ekvivalentný s

$$\widetilde{S}(x_1, x_2, \ldots, x_n) = \overline{y}_1 + \overline{y}_2 + \cdots + \overline{y}_n,$$

čo je súčtový člen tej istej dĺžky ako $S(x_1, x_2, \dots, x_n)$.
Nech f je booleovská funkcia n premenných a

$$U(x_1, x_2, \dots, x_n) = \sum_{i=1}^{m} S_i(x_1, x_2, \dots, x_n)$$

je MNDF booleovskej funkcie $\overline{f}(x_1, x_2, \dots, x_n) = \overline{f(x_1, x_2, \dots, x_n)}$. Počet písmen v MNDF nech je p. Pre funkciu f potom platí

$$f(x_1, x_2, \dots, x_n) = \overline{\overline{f(x_1, x_2, \dots, x_n)}} = \overline{\sum_{i=1}^{m} S_i(x_1, x_2, \dots, x_n)} = \prod_{i=1}^{m} \widetilde{S}_i(x_1, x_2, \dots, x_n).$$

Vzhľadom k tomu, že $\widetilde{S}_i(x_1, x_2, \dots, x_n)$ sú súčtové členy, je B-výraz

$$V(x_1, x_2, \dots, x_n) = \prod_{i=1}^{m} \widetilde{S}_i(x_1, x_2, \dots, x_n)$$

NKF funkcie f. Navyše počet písmen v B-výraze $V(x_1, x_2, \dots, x_n)$ je ten istý ako v B-výraze $U(x_1, x_2, \dots, x_n)$, teda p. Ukážeme teraz, že $V(x_1, x_2, \dots, x_n)$ je MNKF funkcie f. Predpokladajme, že tomu tak nie je a že MNKF funkcie f je $\prod_{i=1}^{r} T_i(x_1, x_2, \dots, x_n)$. Počet písmen v nej nech je q. Potom $q < p$ a

$$\overline{f}(x_1, x_2, \dots, x_n) = \overline{\prod_{i=1}^{r} T_i(x_1, x_2, \dots, x_n)} = \sum_{i=1}^{r} \widetilde{T}_i(x_1, x_2, \dots, x_n),$$

pričom $\widetilde{T}_i(x_1, x_2, \dots, x_n) = \overline{y}_1 \overline{y}_2 \dots \overline{y}_n$, ak $T_i(x_1, x_2, \dots, x_n) = y_1 + y_2 + \dots + y_n$, $y_j \in \{x_j, \overline{x}_j, 0\}$. To však znamená, že $\sum_{i=1}^{r} \widetilde{T}_i(x_1, x_2, \dots, x_n)$ je NDF funkcie s počtom písmen q, čo je v spore s tým, že MNDF funkcie $\overline{f}$ je $U(x_1, x_2, \dots, x_n)$ s počtom písmen p. Tým sme dokázali tvrdenie:

VETA 2.19. Nech f je booleovská funkcia n premenných a

$$U(x_1, x_2, \dots, x_n) = \sum_{i=1}^{m} S_i(x_1, x_2, \dots, x_n)$$

je MNDF funkcie $\overline{f}$. Potom

$$V(x_1, x_2, \dots, x_n) = \prod_{i=1}^{m} \widetilde{S}_i(x_1, x_2, \dots, x_n)$$

je MNKF funkcie f, pričom ak

$$S_i(x_1, x_2, \dots, x_n) = y_1 y_2 \dots y_n, \ y_j \in \{x_j, \overline{x}_j, 1\},$$

tak

$$\widetilde{S}_i(x_1, x_2, \dots, x_n) = \overline{y}_1 + \overline{y}_2 + \dots + \overline{y}_n.$$

□

PRÍKLAD 2.51. Nájdime MNKF funkcie $g(x, y, z) = x\overline{y}\,\overline{z} + \overline{x}yz$.

Riešenie. Zostrojme najprv Karnaughove mapy funkcie g a $\overline{g}$ (obr. 63). Ak porovnáme Karnaughovu mapu funkcie $\overline{g}$ s Karnaughovou mapou na obr. 62a, vidíme, že funkcia $\overline{g}$ a funkcia f z predchádzajúceho príkladu sú rovnaké. Preto

$$U_1(x, y, z) = \overline{x}\,\overline{z} + \overline{y}z + xy \text{ a } U_2(x, y, z) = \overline{x}\,\overline{y} + y\overline{z} + xz$$

		y		z
	0	0	1	0
x	1	0	0	0

a)

		y		z
	1	1	0	1
x	0	1	1	1

b)

OBR. 63. Karnaughova mapa a) funkcie g, b) funkcie $\overline{g}$.

sú MNDF funkcie $\overline{g}$. Potom podľa predchádzajúcej vety sú B-výrazy

$$V_1(x, y, z) = (x + z)(y + \overline{z})(\overline{x} + \overline{y}) \text{ a } V_2(x, y, z) = (x + y)(\overline{y} + z)(\overline{x} + \overline{z})$$

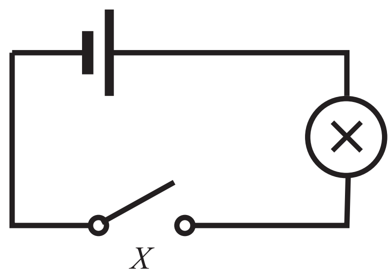
MNKF funkcie g. ∎

KAPITOLA 3

Konečné automaty

1. Definícia konečného automatu

Príklad 3.1. Uvažujme o obvode z obr. 1. Predpokladajme, že v tomto obvode je zaradený zdroj elektrického prúdu, žiarovka a tlačidlový kľúč X, ktorý pri stlačení obvod otvorí, ak bol predtým uzavretý, a naopak, pri stlačení obvod uzavrie, ak bol pred stlačením otvorený.



Obr. 1. Ilustrácia príkladu 3.1

Tento obvod ovládame teda prostredníctvom tlačidlového kľúča, ktorý môžeme považovať za vstup do zariadenia, pritom týmto vstupom je dávaný jediný riadiaci povel t – stlač tlačidlový kľúč.

Reakcia tohto obvodu na riadiaci povel sa prejaví na žiarovke, ktorá buď svieti, alebo nesvieti. Teda túto žiarovku môžeme pokladať za výstup uvedeného obvodu, pritom hodnoty tohto výstupu sú s – žiarovka svieti, n – žiarovka nesvieti. Je zrejmé, že výstup tohto zariadenia pri každom vstupnom povele t nebude rovnaký. Závisí to od toho, či bol obvod pred stlačením kľúča otvorený, alebo uzavretý. Teda má zmysel uvažovať o dvoch stavoch tohto zariadenia. Stav o – obvod je otvorený (žiarovka nesvieti), stav u – obvod je uzavretý (žiarovka svieti).

Je zrejmé, že po každom vstupe t sa mení stav. Ak bol obvod v stave o, po vstupe t sa stav zmení na u. Ak bol v stave u, po vstupe t sa stav zmení na o. Celú túto situáciu vidíme prehľadne v tabuľke 1, časť a).

a)

	t
o	u
u	o

b)

	t
o	s
u	n

c)

	t	t
o	u	s
u	o	n

d)

o	n
u	s

Tabuľka 1. Prechodová a výstupná funkcia z príkladu 3.1

O výstupe môžeme uvažovať dvoma spôsobmi. Buď uvažujeme o výstupe ako reakcii na vstup, a potom v závislosti od stavu pri každom vstupe môžeme priradiť nasledujúci výstup. Ak bol obvod uzavretý a stlačíme kľúč, obvod otvoríme a žiarovka nesvieti. Ak bol obvod otvorený a stlačíme kľúč, obvod uzavrieme a žiarovka bude svietiť. Túto celkovú situáciu sme opísali pomocou tabuľky 1, časť b). Tabuľky z časti a) a b) v tabuľke 1 zvykneme zapisovať spolu, tak ako sme to urobili v tabuľke 1, časť c).

V tomto príklade môžeme však o výstupe uvažovať aj ako o funkcii, ktorá závisí len od stavu zariadenia. Ak je obvod otvorený, žiarovka nesvieti a ak je obvod uzavretý, žiarovka svieti. Túto situáciu zapisujeme v tabuľke 1, časť d). ■

PRÍKLAD 3.2. Uvažujme o situácii, keď máme k dispozícii generátor signálov a, b, ktorý náhodným spôsobom v určených časových okamihoch generuje jeden z uvedených signálov. Teraz chceme opísať zariadenie, ktoré tieto signály číta a na tieto signály reaguje takto:
Ak v čase t je vyslaný ten istý signál ako v čase $t-2$, vyšle signál 1, v opačnom prípade vyšle signál 0.

Riešenie. Túto situáciu môžeme ilustrovať takto:

Vstup čítacieho zariadenia: $\ldots a\,a\,b\,b\,a\,b\,a\,b\,b\,b\,b\,a\,a\,a\,a\,a\,b\,a\,b\,b\,a\,b\ldots$

Výstup čítacieho zariadenia: $\ldots\ldots 0\,0\,0\,1\,1\,1\,0\,1\,1\,0\,0\,1\,1\,1\,0\,1\,1\,0\,0\,1\ldots$

Je zrejmé, že pri zariadení, ktoré chceme opísať, má zmysel uvažovať o vstupe X, ktorý prijíma signály a, b a o výstupe Z, ktorým sa vysielajú signály 0 a 1 (pozri obr. 2).

Aby sme vedeli opísať činnosť čítacieho zariadenia v čase t, potrebujeme poznať jeho históriu v čase $t-2$. Za týchto okolností budeme vedieť, ako reaguje zariadenie v čase $t-2$

OBR. 2. Ilustrácia príkladu 3.2

a t. V čase $t+1$ však budeme musieť poznať, aká bola situácia v čase $t-1$. Potom budeme vedieť opísať činnosť čítacieho zariadenia v čase $t-2$, $t-1$, t, $t+1$. Ale je zrejmé, že teraz už budeme vedieť povedať, aký je výstup aj v čase $t+2$, a teda aj v čase $t+3$ atď. Slovom, aby sme vedeli opísať činnosť tohto zariadenia od istého časového okamihu, musíme poznať jeho históriu v predošlých dvoch taktoch. V dvoch taktoch, ktoré nasledujú za sebou, je na vstupe čítacieho zariadenia možný len výskyt týchto dvojíc: aa, ab, ba, bb. Týmto dvojiciam môžeme priradiť štyri stavy čítacieho zariadenia. Stav s_{ab} bude zodpovedať situácii, keď v čase $t-2$ bol na vstupe signál a a v čase $t-1$ signál b. Podobný význam budú mať aj stavy s_{aa}, s_{ba}, a s_{bb}. Teda napríklad, keď v čase t je zariadenie v stave s_{ab} a

TABUĽKA 2. Ilustrácia príkladu 1.2

	a	b	a	b
s_{aa}	s_{aa}	s_{ab}	1	0
s_{ab}	s_{ba}	s_{bb}	1	0
s_{ba}	s_{aa}	s_{ab}	0	1
s_{bb}	s_{ba}	s_{bb}	0	1

na vstupe sa objaví signál a, zariadenie na výstupe generuje signál 1 a prechádza do stavu s_{ba}. Ak v stave s_{ab} vstúpi signál b, tak na výstupe sa objaví 0 a zariadenie prejde do stavu

s_{bb}. Celú túto situáciu opisujeme v tabuľke 2, kde v prvej časti opisujeme zmenu stavov a v druhej časti ukazujeme reakciu výstupu na jednotlivé vstupy pri daných stavoch. ■

V predošlých dvoch príkladoch sme sa zaoberali zariadeniami, ktorých opis je možné zhrnúť do jedného modelu, ktorý budeme nazývať konečný automat.

DEFINÍCIA 3.1. Nech $X = \{x_1, \dots, x_k\}$, $S = \{s_1, \dots, s_n\}$ a $Z = \{z_1, \dots, z_m\}$ sú konečné množiny. Nech $\delta : S \times X \to S$ a $\lambda : S \times X \to Z$ sú dané funkcie. Potom päticu $A = (S, X, Z, \delta, \lambda)$ budeme nazývať ***konečný automat***.

Množinu X nazývame ***vstupná abeceda*** a jej prvky nazývame ***písmená*** vstupnej abecedy (vstupy). Množinu S nazývame ***množina stavov*** automatu A a jej prvky nazývame ***stavy*** automatu A. Množinu Z nazývame ***výstupná abeceda*** a jej prvky sú ***písmená*** výstupnej abecedy (výstupy). Funkciu $\delta : S \times X \to S$, pomocou ktorej je každému stavu S a vstupnému písmenu priradený (nový) stav, nazývame ***prechodová funkcia*** automatu A. Funkciu $\lambda : S \times X \to Z$, pomocou ktorej je každému stavu a vstupnému písmenu priradené výstupné písmeno, sa nazýva ***výstupná funkcia*** automatu A.

V príklade 3.1 je prechodová a výstupná funkcia daná pomocou tabuľky 1, časť c). Prechodová a výstupná funkcia z príkladu 3.2 sú dané v tabuľke 2.

POZNÁMKA 3.1. V tejto učebnici sa budeme zaoberať iba konečnými automatmi. Preto slovo „konečný“ budeme v názve automatu vynechávať.

Ako sme videli v príklade 3.1, môžeme uvažovať o dvoch druhoch výstupných funkcií. V prvom prípade sme uvažovali o výstupnej funkcii, ktorej hodnoty závisia od stavu a vstupu. V druhom prípade sme uvažovali o funkcii, ktorej hodnoty závisia iba od stavu. To je aj dôvod, prečo sa robia rozdiely v definícii automatu. Automat z definície 3.1 sa presnejšie nazýva ***Mealyho automat***. Okrem Mealyho automatov sa budeme zaoberať aj tzv. Moorovými automatmi.

DEFINÍCIA 3.2. ***Moorov automat*** je pätica $A = (S, X, Z, \delta, \mu)$, kde S, X, Z a δ sú zhodné s im zodpovedajúcimi objektami v definícii Mealyho automatu. Výstupná funkcia μ priraďuje stavu výstup, teda $\mu : S \to Z$.

Pri automatoch predpokladáme činnosť v diskrétnom čase. Teda uvažujeme o časových okamihoch (taktoch) $t_1, t_2, \dots$, v ktorých sa vzhľadom na vstup a stav v danom okamihu generuje výstup (patriaci ešte k danému časovému okamihu) a nastavuje sa nový stav pre nasledujúci časový takt. Pretože pri práci automatu nie sú dôležité hodnoty t_i, ale iba ich poradie, budeme v budúcnosti uvažovať iba o číslach taktov 1, 2, 3, … Pritom intervaly medzi jednotlivými taktami nemusia byť rovnaké. Pri tejto časovej interpretácii môžeme písať:

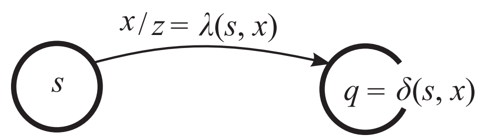
$$\begin{aligned} &\delta(s(t), x(t)) = s(t+1)\ , && \\ &\lambda(s(t), x(t)) = z(t) && \text{pri Mealyho automate,} \\ &\mu(s(t)) = z(t) && \text{pri Moorovom automate.} \end{aligned}$$

Pritom predpokladáme, že priradenie výstupu sa uskutočňuje počas jedného taktu (nie v jedinom časovom okamihu).

Z definície automatu vyplýva, že automat je úplne špecifikovaný pomocou tabuliek prechodovej a výstupnej funkcie. V týchto tabuľkách sa totiž nachádzajú aj množiny X, S, Z. Ku každému (konečnému) automatu je zvykom priradiť aj graf.

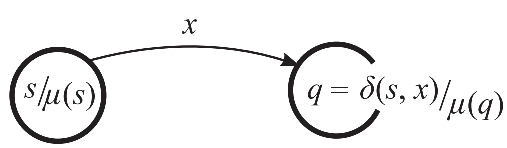
Definícia 3.3. Nech je daný automat $A = (S, X, Z, \delta, \lambda)$. ***Grafom automatu*** A budeme nazývať orientovaný graf $G = (V, H, f)$, ktorý je definovaný takto: Množina vrcholov $V = S$. Množina hrán H je zhodná s množinou „prechodov“ medzi stavmi, pričom prechod medzi stavmi s a q je definovaný práve vtedy, keď existuje písmeno $x \in X$ také, že $\delta(s, x) = q$. Potom vzťah incidencie je funkcia $f : H \to S \times S$, ktorá je definovaná takto: $f(h) = (s, q)$ práve vtedy, keď h je prechod medzi stavmi s a q (t.j. existuje $x \in X$ také, že $\delta(s, x) = q$).

Graf automatu budeme kresliť ako ohodnotený graf. Ak h je hrana (prechod) spájajúca vrcholy s a q, na základe podmienky $\delta(s, x) = q$ budeme k tejto hrane pripisovať označenie x a k tomuto označeniu ešte aj hodnotu výstupu $z = \lambda(s, x)$ (pozri obr. 3).



Obr. 3. Ilustrácia definície grafu automatu

V prípade Moorovho automatu definujeme graf automatu rovnako ako v prípade Mealyho automatu. Pri ohodnocovaní hrán však k hrane pripisujeme len hodnotu vstupu, ktorý spôsobuje príslušný prechod z jedného stavu do druhého. Výstup zapisujeme k jednotlivým stavom (vrcholom). Teda ak $\delta(s, x) = q$, potom dvojica vrcholov s, q bude spojená hranou tak, ako to vidno na obr. 4.



Obr. 4. Ilustrácia definície grafu Moorovho automatu

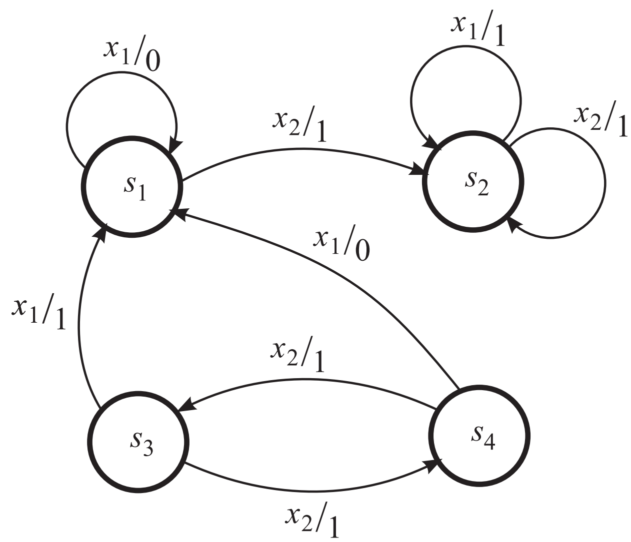
Príklad 3.3. Nakreslíme graf automatu A, ktorý je daný pomocou tabuľky 3.

Tabuľka 3. Tabuľka automatu z príkladu 3.3

	δ		λ	
	x_1	x_2	x_1	x_2
s_1	s_1	s_2	0	1
s_2	s_2	s_2	1	1
s_3	s_1	s_4	1	1
s_4	s_1	s_1	0	1

Riešenie. Je zrejmé, že ide o automat $A = (S, X, Z, \delta, \lambda)$, kde $S = \{s_1, s_2, s_3, s_4\}$, $X = \{x_1, x_2\}$, $Z = \{0, 1\}$. Funkcie δ a λ sú dané pomocou tabuľky 3.

Graf tohto automatu má štyri vrcholy s_1, s_2, s_3, s_4, pričom z každého vrchola musia vychádzať práve dve hrany, ktoré sú ohodnotené vstupmi x_1, a x_2. Graf tohto automatu je na obr. 5. ■



Obr. 5. Graf automatu z príkladu 3.3

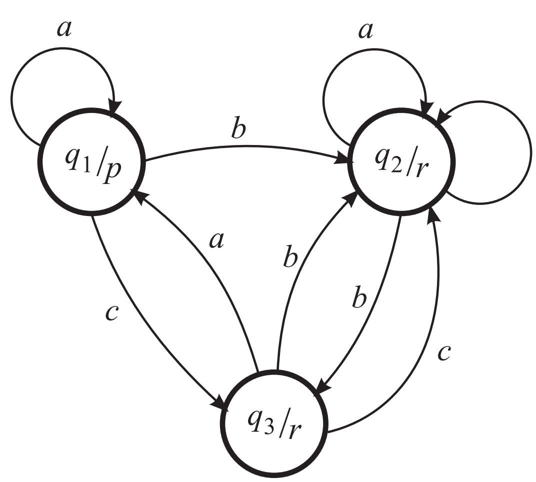
Tabuľka 4. Tabuľka automatu z príkladu 3.4

	δ			
	a	b	c	μ
q_1	q_1	q_2	q_3	p
q_2	q_2	q_3	q_2	r
q_3	q_1	q_2	q_2	r

Príklad 3.4. Nech Moorov automat A je daný pomocou tabuľky 4. Nakreslíme graf tohto automatu.

Riešenie. Graf bude mať tri vrcholy q_1, q_2, q_3, pritom z každého vrchola budú vychádzať práve tri hrany, ktoré budú ohodnotené vstupnými písmenami a, b, c (pozri obr. 6).

Uvažujme teraz o vstupnej abecede $X = \{x_1, \ldots, x_k\}$ automatu A a o tom, že daný automat pracuje v taktoch, ktoré sú priradené časovým okamihom očíslovaným ■



Obr. 6. Graf automatu z príkladu 3.4

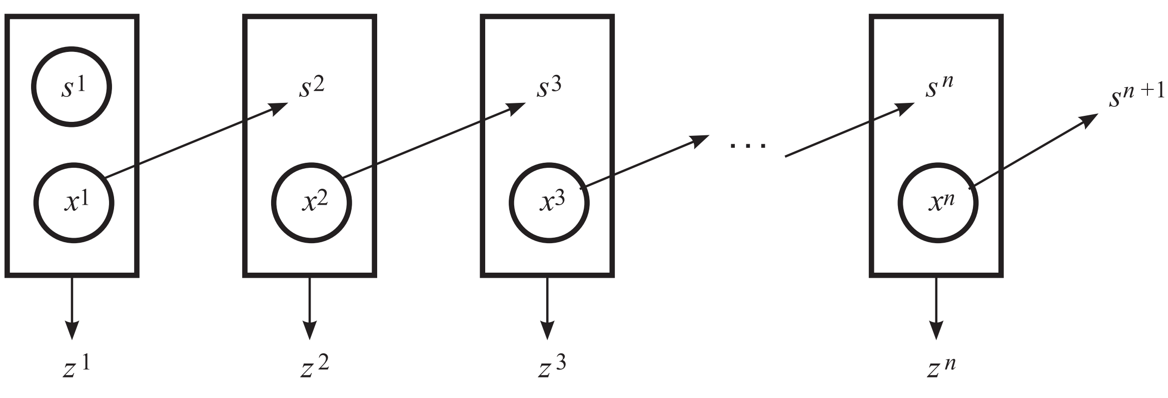
v poradí $1, 2, 3, \ldots$ Predpokladajme, že v čase $t = 1$ bude na vstupe automatu napríklad písmeno x_7. Môžeme písať $x(1) = x_7$. Nech v čase $t = 2$ je na vstupe písmeno x_4 a v čase $t = 3$ je to opäť x_4. Potom môžeme písať $x(2) = x_4$ a $x(3) = x_4$. Teda počas taktov $t = 1$, $t = 2$, $t = 3$ bola na vstupe automatu postupnosť $x(1)x(2)x(3) = x_7x_4x_4$. Je zrejmé, že má zmysel uvažovať o tom, že počas týchto troch taktov sa na vstupe mohla objaviť ľubovoľná trojica prvkov množiny X (pričom je možné aj opakovanie prvkov v tejto trojici). Ďalej je zrejmé, že môžeme uvažovať o ľubovoľnej konečnej postupnosti taktov $t = 1$, $t = 2, \ldots, t = n$, počas ktorej sa na vstupe objaví konečná postupnosť vstupných písmen $x(1)x(2)\ldots x(n)$. Tieto postupnosti budeme v budúcnosti zapisovať v tvare $x(1)x(2)\ldots x(n) = x^1x^2\ldots x^n$. Napríklad postupnosť $x^1x^2x^3x^4x^5$ môže mať tvar $x_3x_4x_4x_1x_2$. Je to postupnosť vytvorená z piatich vstupných písmen.

Uvažujme o automate $A = (S, X, Z, \delta, \lambda)$, ktorý v čase $t = 1$ je v stave s^1 a na vstup je privedené vstupné písmeno x^1. Uvažujme o tom, že v nasledujúcich okamihoch postupne privedieme na vstup písmená $x^2, x^3, \ldots, x^n$. Teda automat je na začiatku v stave s^1 a v nasledujúcich n taktoch privedieme na vstup vstupné slovo $x^1, x^2, \ldots, x^n$ (pozri obr. 7, na ktorom sme dané hodnoty zakrúžkovali). Potom ako odozvu na stav s^1 a vstupné písmeno x^1 dostávame výstupné písmeno $z^1 = \lambda(s^1, x^1)$ a pomocou prechodovej funkcie nasledujúci stav $s^2 = \delta(s^1, x^1)$. Teraz pomocou stavu s^2 a vstupného písmena x^2 dostávame výstupné písmeno z^2 a nový stav s^3. Takto pokračujeme až po n-tý takt, keď dostávame $z^n = \lambda(s^n, x^n)$ a $s^{n+1} = \delta(s^n, x^n)$. Ako reakciu automatu na vstupné slovo x^1, x^2, $\ldots, x^n$ pri začiatočnom stave s^1 dostávame:

1. Postupnosť $s^1, s^2, \ldots, s^{n+1}$. Z týchto stavov nás zaujíma iba s^{n+1}, ktorý nám určuje, v akom stave začne automat pracovať, ak sa rozhodneme pokračovať so vstupnými povelmi.

2. Výstupné slovo $z^1, z^2, \ldots, z^n$ vo výstupnej abecede Z. Toto slovo nás zaujíma celé, lebo je to vonkajšia reakcia automatu na vstupné slovo. Výstupné slovo nám predstavuje postupnosť riadiacich príkazov automatu, ktoré vydá ako reakciu na vstupné slovo $x^1, x^2, \ldots, x^n$.

Má teda zmysel uvažovať o rozšírení definície prechodovej a výstupnej funkcie aj pre prípad, keď vstupné písmeno nahradíme celým slovom zloženým zo vstupných písmen.



Obr. 7. Ilustrácia definície rozšírenej prechodovej a výstupnej funkcie Mealyho automatu

Definícia 3.4. Nech $A = (S, X, Z, \delta, \lambda)$ je Mealyho automat. ***Rozšírenou prechodovou funkciou*** budeme nazývať funkciu $\widehat{\delta} : S \times X^+ \to S$, ktorá je definovaná takto:

a) $\widehat{\delta}(s, x) = \delta(s, x)$ pre každé $s \in S$ a každé $x \in X$.

b) $\widehat{\delta}(s, wx) = \delta(\widehat{\delta}(s, w), x)$ pre každé $s \in S$, $w \in X^+$ a $x \in X$.

Rozšírenou výstupnou funkciou budeme nazývať funkciu $\widehat{\lambda} : S \times X^+ \to Z^+$, ktorá je definovaná takto:

a) $\widehat{\lambda}(s, x) = \lambda(s, x)$ pre každé $s \in S$ a každé $x \in X$.
b) $\widehat{\lambda}(s, wx) = \widehat{\lambda}(s, w)\lambda(\widehat{\delta}(s, w), x)$ pre každé $s \in S$, $w \in X^+$ a $x \in X$.

Rozšírenú prechodovú a výstupnú funkciu sme definovali induktívnym spôsobom. V prvom kroku sme tieto funkcie definovali pre vstupné slová dĺžky 1. Potom za predpokladu, že tieto hodnoty poznáme pre slová dĺžky n, sme ich definovali pre slová dĺžky $n + 1$. Napríklad

$$\widehat{\delta}(s^1, x^1x^2) = \delta(\widehat{\delta}(s^1, x^1), x^2) = \delta(\delta(s^1, x^1), x^2) = \delta(s^2, x^2) = s^3,$$

$$\widehat{\lambda}(s^1, x^1x^2) = \widehat{\lambda}(s^1, x^1)\lambda(\widehat{\delta}(s^1, x^1), x^2) = \lambda(s^1, x^1)\lambda(\delta(s^1, x^1), x^2) = z^1\lambda(s^2, x^2) = z^1z^2$$

(pozri obr. 7).

Z definície rozšírených funkcií tiež dostávame:

$$\begin{aligned}
&\widehat{\delta}(s, x^1x^2 \dots x^n) = \delta(\widehat{\delta}(s, x^1 \dots x^{n-1}), x^n) = \delta(\delta(\widehat{\delta}(s, x^1 \dots x^{n-2}), x^{n-1}), x^n) = \dots = \\
&= \delta(\delta(\delta(\dots \delta(\delta(s, x^1), x^2) \dots), x^{n-2}), x^{n-1}), x^n) = \widehat{\delta}(\delta(s, x^1), x^2 \dots x^n) = \\
&= \widehat{\delta}(\widehat{\delta}(s, x^1x^2), x^3 \dots x^n) = \dots .
\end{aligned}$$

čiže pre každé $s \in S$, $x \in X$, $w \in X^+$, je $\widehat{\delta}(s, xw) = \widehat{\delta}(\delta(s, x), w)$. Takýmto spôsobom sa dá dokázať, že dokonca pre každé $u, v \in X^+$ a každé $s \in S$ je $\widehat{\delta}(s, uv) = \widehat{\delta}(\widehat{\delta}(s, u), v)$.

Podobne pre rozšírenú výstupnú funkciu dostávame:

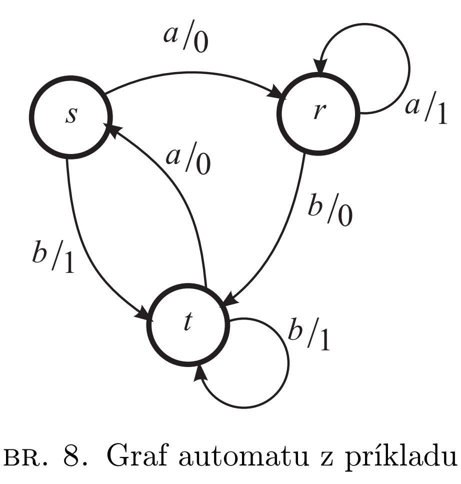
$$\begin{aligned}
&\widehat{\lambda}(s, x^1x^2 \dots x^n) = \widehat{\lambda}(s, x^1 \dots x^{n-1})\lambda(\widehat{\delta}(s, x^1 \dots x^{n-1}), x^n) = \\
&= \widehat{\lambda}(s, x^1 \dots x^{n-2})\lambda(\widehat{\delta}(s, x^1 \dots x^{n-2}), x^{n-1})\lambda(\widehat{\delta}(s, x^1 \dots x^{n-1}), x^n) = \dots = \\
&= \lambda(s, x^1)\lambda(\widehat{\delta}(s, x^1), x^2)\lambda(\widehat{\delta}(s, x^1x^2), x^3) \dots \lambda(\widehat{\delta}(s, x^1 \dots x^{n-1}), x^n).
\end{aligned}$$

Z toho už vyplýva, že pre každé $x \in X$ a každé $w \in X^+$ je

$$\widehat{\lambda}(s, xw) = \lambda(s, x)\widehat{\lambda}(\delta(s, x), w).$$

Pri hľadaní hodnôt rozšírenej prechodovej a výstupnej funkcie s výhodou využijeme graf automatu.

Príklad 3.5. Nech automat A je daný pomocou grafu na obr. 8. Uvažujme o vstupnom slove $w = abba$. Nájdeme $\widehat{\delta}(s, abba)$ a $\widehat{\lambda}(s, abba)$.



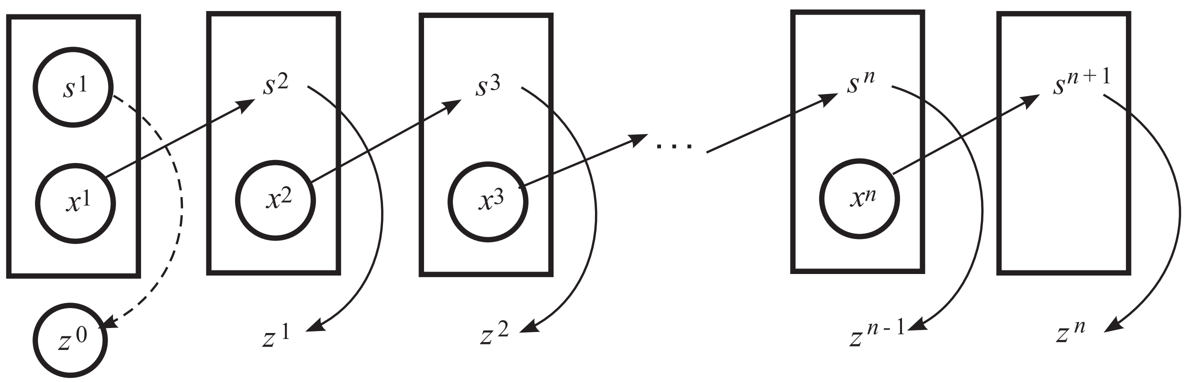
Obr. 8. Graf automatu z príkladu 3.5

Riešenie. $\widehat{\delta}(s, abba)$ je stav daného automatu, v ktorom sa bude nachádzať, keď v stave s postupne na vstup privedieme písmená a, b, b, a. Túto postupnosť, začnúc od stavu s, sledujeme na hranách grafu daného automatu. Zo stavu s pri vstupe a prechádzame do stavu r (t.j. $\delta(s, a) = r$), zo stavu r pri vstupe b do stavu t (t. j. $\delta(r, b) = t$), zo stavu t pri vstupe b do stavu t (t.j. $\delta(t, b) = t$) a zo stavu t pri vstupe a do stavu s (t.j. $\delta(t, a) = s$). Preto $\delta(s, abba) = s$.
Priamy výpočet vyzerá takto:

$$\begin{aligned}&\widehat{\delta}(s, abba) = \widehat{\delta}(\delta(s, a), bba) = \widehat{\delta}(r, bba) = \widehat{\delta}(\delta(r, b), ba) = \\ &= \widehat{\delta}(t, ba) = \widehat{\delta}(\delta(t, b), a) = \delta(t, a) = s.\end{aligned}$$

Zodpovedajúce výstupné signály sa nachádzali na hranách, po ktorých sme prechádzali zo stavu s do stavu r, zo stavu r do stavu t, zo stavu t do stavu t a zo stavu t do stavu s. Preto $\widehat{\lambda}(s, abba) = 0010$. ■

Rozšírenie prechodovej funkcie pri Moorovom automate definujeme tak isto ako pri Mealyho automate. S rozšírením výstupnej funkcie je to pri Moorovom automate trochu zložitejšie. Uvažujme podobne ako v prípade Mealyho automatu o situácii, keď v stave s^1 na vstup automatu privedieme vstupné slovo $x^1x^2\ldots x^n$ (obr. 9). V tomto prípade prvý výstup $z^0 = \mu(s^1)$ nezávisí od vstupného písmena x^1. Reakcia výstupu na vstup x^1 sa prejaví až pri výstupnom písmene $z^1 = \mu(s^2) = \mu(\delta(s^1, x^1))$. Ďalej z^2 je reakciou na x^2, lebo $z^2 = \mu(s^3) = \mu(\delta(s^2, x^2))$ atď. Z týchto úvah sa nám ponúka táto definícia rozšírenej výstupnej funkcie Moorovho automatu.



Obr. 9. Ilustrácia definície rozšírenej výstupnej funkcie Moorovho automatu

Definícia 3.5. Nech $A = (S, X, Z, \delta, \mu)$ je Moorov automat. ***Rozšírenou výstupnou funkciou Moorovho automatu*** budeme nazývať funkciu $\widehat{\mu} : S \times X^+ \to Z^+$, ktorá je definovaná takto:

a) $\widehat{\mu}(s, x) = \mu(\delta(s, x))$ pre každé $s \in S$ a každé $x \in X$.
b) $\widehat{\mu}(s, wx) = \widehat{\mu}(s, w)\mu(\widehat{\delta}(s, wx))$ pre každé $s \in S$, každé $x \in X$ a každé $w \in X^+$.

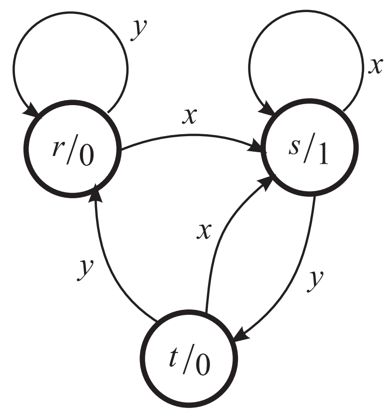
Z definície rozšírenej výstupnej funkcie Moorovho automatu vyplýva, že táto funkcia každému slovu $x^1x^2\ldots x^n$ dĺžky n opäť priradí slovo $z^1z^2\ldots z^n$ dĺžky n. Neberieme totiž do úvahy prvé výstupné písmeno z^0, ktoré závisí od stavu, a nie od prvého písmena v slove $x^1x^2\ldots x^n$.

Pre ľubovoľný stav $s \in S$ a ľubovoľné slovo $x^1x^2\ldots x^n \in X^+$ dostávame:

$$\begin{aligned}
&\widehat{\mu}(s, x^1x^2 \dots x^n) = \widehat{\mu}(s, x^1x^2 \dots x^{n-1})\mu(\widehat{\delta}(s, x^1x^2 \dots x^n)) = \\
&= \widehat{\mu}(s, x^1x^2 \dots x^{n-2})\mu(\widehat{\delta}(s, x^1x^2 \dots x^{n-1}))\mu(\widehat{\delta}(s, x^1x^2 \dots x^n)) = \dots = \\
&= \mu(\widehat{\delta}(s, x^1))\mu(\widehat{\delta}(s, x^1x^2)) \dots \mu(\widehat{\delta}(s, x^1x^2 \dots x^n)).
\end{aligned}$$

Opäť je výhodné hodnoty rozšírenej výstupnej funkcie Moorovho automatu získavať pomocou grafu automatu.

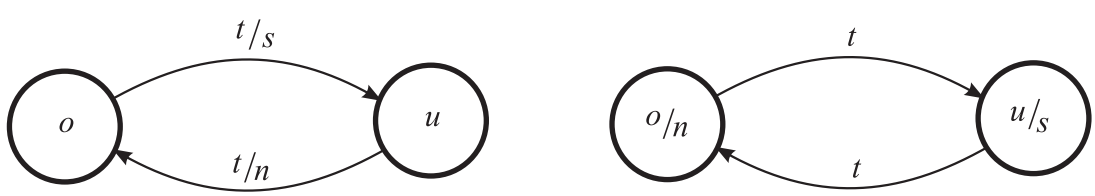
Príklad 3.6. Uvažujme o Moorovom automate, ktorý je daný grafom na obr. 10. Počítajme napríklad $\widehat{\mu}(r, xxyx)$.



Obr. 10. Graf automatu z príkladu 3.6

Riešenie.
$\widehat{\mu}(r, xxyx) = \mu(\widehat{\delta}(r, x))\mu(\widehat{\delta}(r, xx))\mu(\widehat{\delta}(r, xxy))\mu(\widehat{\delta}(r, xxxyx)) = \mu(s)\mu(s)\mu(t)\mu(s) = 1101.$
Vidíme, že tento výsledok získame, keď v grafe automatu začneme v stave r sledovať hrany, ktoré sú postupne ohodnotené vstupnými písmenami x, x, y, x a tak isto postupne vypisujeme výstupné písmená, ktoré sú priradené stavom, do ktorých jednotlivé hrany na našom postupe vstupujú. ■

Vráťme sa ešte k príkladu 3.1. Teraz vidíme, že na zariadenie z tohto príkladu sme mohli nazerať buď ako na Mealyho automat, ktorý je daný pomocou tabuľky 1, časť c), alebo ako na Moorov automat, ktorého prechodová funkcia je v tabuľke 1, časť a) a výstupná funkcia je daná pomocou tabuľky 1, časť d). Graf uvedeného Mealyho automatu je na obr. 11, časť a), graf Moorovho automatu je na obr. 11, časť b). Vidíme, že tieto



a) b)

Obr. 11. Grafy automatov z príkladu 3.1

grafy a aj samotné automaty sú do istej miery vysoko viazané tým, že majú spoločnú prechodovú funkciu.

Teraz budeme medzi Moorovými a Mealyho automatmi, ktoré majú podobné vlastnosti, definovať ekvivalenciu.

DEFINÍCIA 3.6. Nech je daný Mealyho automat $A = (S, X, Z, \delta, \lambda)$ a Moorov automat $\widetilde{A} = (S, X, Z, \delta, \mu)$ (tieto automaty majú rovnaké množiny vstupov, stavov, výstupov a prechodovú funkciu). Nech pre každé $x \in X$ a každé $s \in S$ je $\lambda(s, x) = \widehat{\mu}(s, x)$. Potom budeme hovoriť, že tieto automaty sú ***silno ekvivalentné***.

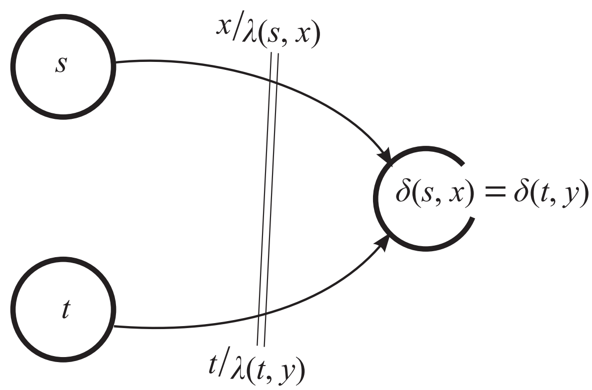
Nech $A = (S, X, Z, \delta, \lambda)$ je Mealyho a $\widetilde{A} = (S, X, Z, \delta, \mu)$ je Moorov automat a tieto automaty sú silno ekvivalentné. Nech $w = x^1x^2 \dots x^n$ je ľubovoľné vstupné slovo a $s \in S$ je ľubovoľný stav. Potom

$$\begin{aligned}
&\widehat{\mu}(s, x^1x^2 \dots x^n) = \mu(\widehat{\delta}(s, x^1))\mu(\widehat{\delta}(s, x^1x^2)) \dots \mu(\widehat{\delta}(s, x^1x^2 \dots x^n)) = \\
&= \widehat{\mu}(s, x^1)\widehat{\mu}(\widehat{\delta}(s, x^1), x^2) \dots \widehat{\mu}(\widehat{\delta}(s, x^1x^2 \dots x^{n-1}), x^n) = \\
&= \lambda(s, x^1)\lambda(\widehat{\delta}(s, x^1), x^2) \dots \lambda(\widehat{\delta}(s, x^1x^2 \dots x^{n-1}), x^n) = \\
&= \widehat{\lambda}(s, x^1x^2 \dots x^n)
\end{aligned}$$

To znamená, ak A a $\widetilde{A}$ sú silno ekvivalentné automaty, pre každé $w \in X^+$ a pre každé $s \in S$ je $\widehat{\mu}(s, w) = \widehat{\lambda}(s, w)$.

Ďalej je zrejmé, ak $\widetilde{A} = (S, X, Z, \delta, \mu)$ je Moorov automat, potom k nemu existuje silno ekvivalentný Mealyho automat $A = (S, X, Z, \delta, \lambda)$. Stačí totiž definovať funkciu $\lambda : S \times X \to S$ pomocou podmienky $\widehat{\mu}(s, x) = \lambda(s, x)$ pre každé $s \in S$ a každé $x \in X$. Teda vidíme, že podmienka existencie silno ekvivalentného Mealyho automatu k danému Moorovmu automatu nekladie na tento Moorov automat žiadne ohraničenie.

Naopak, pre Mealyho automat je to dosť silná podmienka. V prípade, že A a $\widetilde{A}$ sú silno ekvivalentné automaty, z podmienky $\delta(s, x) = \delta(t, y)$ vyplýva $\mu(\delta(s, x)) = \mu(\delta(t, y))$, a teda aj $\widehat{\mu}(s, x) = \widehat{\mu}(t, y)$. Z toho už dostávame $\lambda(s, x) = \lambda(t, y)$. To znamená, že v Mealyho automate podmienka $\delta(s, x) = \delta(t, y)$ implikuje $\lambda(s, x) = \lambda(t, y)$ (pozri obr. 12). Z toho vyplýva: Ak k Mealyho automatu A existuje silno ekvivalentný Moorov automat $\widetilde{A}$, v grafe Mealyho automatu všetky hrany vstupujúce do toho istého stavu sú ohodnotené tým istým výstupným písmenom.



OBR. 12. Ilustrácia silnej ekvivalencie automatov

Ak, naopak, Mealyho automat $A = (S, X, Z, \delta, \lambda)$ má tú istú vlastnosť, že všetky hrany vstupujúce do toho istého stavu sú ohodnotené tým istým výstupným písmenom, t.j. podmienka $\delta(s, x) = \delta(t, y)$ implikuje $\lambda(s, x) = \lambda(t, y)$, môžeme výstupnú funkciu $\mu : S \to Z$ silno ekvivalentného Moorovho automatu definovať takto:

a) Ak pre $s' \in S$ existuje $s \in S$ a $x \in X$ také, že $\delta(s, x) = s'$, definujeme $\mu(s') = \mu(\delta(s, x)) = \lambda(s, x)$. Z podmienky $\delta(s, x) = \delta(t, y)$ implikuje $\lambda(s, x) = \lambda(t, y)$ vyplýva, že funkcia μ je v bode s' dobre definovaná.

b) Ak pre $s' \in S$ neexistuje $s \in S$ a také $x \in X$, že $\delta(s,x) = s'$, (do stavu s' nevstupuje žiadna hrana grafu automatu), definujeme $\mu(s')$ ľubovoľne.

Z prvej podmienky vyplýva, že automaty $A = (S, X, Z, \delta, \lambda)$ a $\widetilde{A} = (S, X, Z, \delta, \mu)$ sú silno ekvivalentné, lebo pre každé $s \in S$ a každé $x \in X$ je $\mu(\delta(s,x)) = \lambda(s,x)$ a to znamená, že $\widehat{\mu}(s,x) = \lambda(s,x)$.

Príklad 3.7. Nech Moorov automat je daný pomocou tabuľky 5, časť a). V časti b) tejto tabuľky je daný jediný možný silno ekvivalentný Mealyho automat k danému Moorovmu automatu. Túto tabuľku často píšeme tak, ako je to v tabuľke 5, časti c). Z tohto zápisu lepšie vidieť, že $\delta(s,x) = \delta(t,y)$ implikuje $\lambda(s,x) = \lambda(t,y)$, lebo každý stav vnútri tabuľky je lomený tým istým výstupným písmenom, nech sa nachádza na ktoromkoľvek mieste tabuľky. Tento príznak nám umožňuje aj rozhodnúť, kedy naopak k Mealyho automatu môžeme zostrojiť silno ekvivalentný Moorov automat. ■

Tabuľka 5. Tabuľky automatov z príkladu 3.7

	δ		
	a	b	μ
s_1	s_1	s_2	0
s_2	s_3	s_4	1
s_3	s_1	s_4	1
s_4	s_3	s_2	0

a)

	δ		λ	
	a	b	a	b
s_1	s_1	s_2	0	1
s_2	s_3	s_4	1	0
s_3	s_1	s_4	0	0
s_4	s_3	s_2	1	1

b)

	δ/λ	
	a	b
s_1	$s_1/0$	$s_2/1$
s_2	$s_3/1$	$s_4/0$
s_3	$s_1/0$	$s_4/0$
s_4	$s_3/1$	$s_2/1$

c)

K ekvivalencii medzi Moorovými a Mealyho automatmi sa ešte vrátime v ďalšej časti tejto učebnice.

2. Konečné akceptory

Nech $X = \{x_1, x_2, \dots, x_n\}$ je ľubovoľná konečná množina. V teórii formálnych jazykov definujeme jazyk (v abecede X) ako ľubovoľnú podmnožinu voľnej pologrupy X^+.

Napríklad nech $X = \{a, b\}$. Potom jazykmi v abecede X sú napríklad tieto množiny. $L_1 = \emptyset$, $L_2 = \{a, ab, aba\}$, $L_3 = \{wa; w \in X^+\}$ = množina všetkých slov v abecede X, ktoré končia písmenom a a majú dĺžku aspoň dva, $L_4 = \{a^n b^n; n = 1, 2, \dots\}$. Vidíme, že môžeme vymenovať nekonečne veľa takýchto jazykov, Jedným zo spôsobov, ako sa vo voľnej pologrupe dajú jazyky špecifikovať, je ich akceptovanie pomocou konečného akceptora.

Konečný akceptor je možné definovať ako Moorov automat, ktorý oproti všeobecne definovaným Moorovým automatom sa vyznačuje istými špecifickými črtami.

(1) V konečnom akceptore je vyznačený jeden stav - začiatočný stav, v ktorom automat vždy začína svoju činnosť (každý automat s vyznačeným začiatočným stavom sa nazýva iniciálny automat).
(2) Výstupná abeceda $Z = \{0, 1\}$. Pritom sa dohodneme, že 1 použijeme ako indikátor akceptovania a 0 ako indikátor neakceptovania slova $w \in X^+$ akceptorom. To znamená, že ak v začiatočnom stave privedieme na vstup akceptora slovo $w \in X^+$, potom v prípade, že automat skončí svoju činnosť v stave s, pre ktorý je $\mu(s) = 1$, akceptor slovo w akceptuje, ak $\mu(s) = 0$, akceptor slovo w neakceptuje.

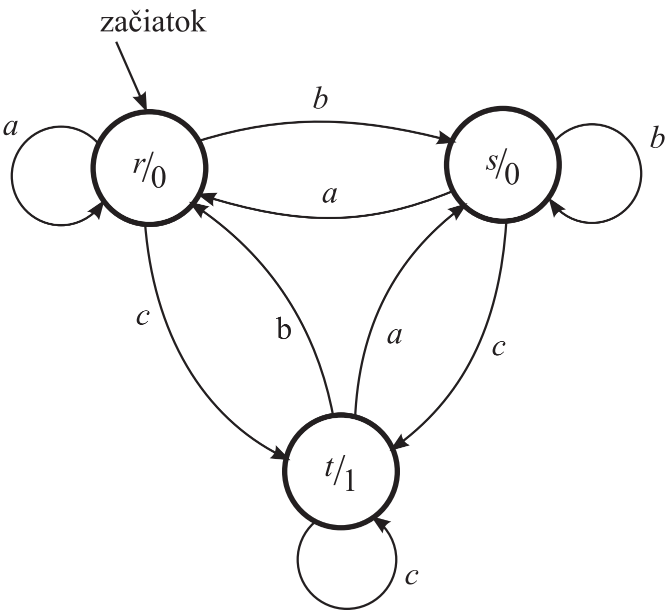
Teraz už môžeme vysloviť definíciu konečného akceptora.

DEFINÍCIA 3.7. ***Konečný akceptor*** je Moorov automat $A = (S, X, Z, \delta, \mu, s_0)$, kde $s_0 \in S$ je začiatočný stav a $Z = \{0, 1\}$.

Budeme hovoriť, že ***akceptor*** A ***akceptuje slovo*** $w \in X^+$, ak slovo $\widehat{\mu}(s_0, w)$ končí jednotkou. To znamená, že $\widehat{\mu}(\widehat{\delta}(s_0, w)) = 1$. Akceptor A slovo $w \in X^+$ neakceptuje práve vtedy, keď $\widehat{\mu}(\widehat{\delta}(s_0, w)) = 0$.

Jazyk $L(A)$ ***akceptovaný akceptorom*** A je množina všetkých slov z pologrupy X^+, ktoré daný akceptor akceptuje. Teda $L(A) = \{w \in X^+;\ \mu(\widehat{\delta}(s_0, w)) = 1\}$.

PRÍKLAD 3.8. Pokúsime sa opísať jazyk, ktorý akceptuje akceptor na obr. 13.



OBR. 13. Konečný akceptor z príkladu 3.8

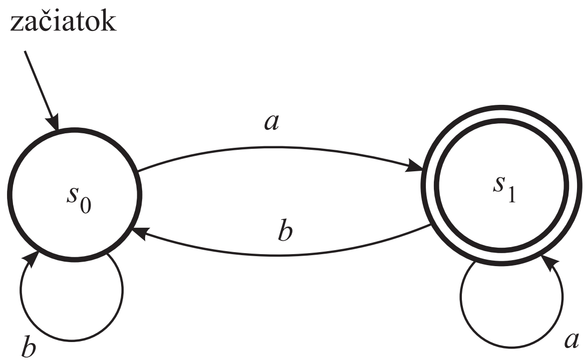
Riešenie. Z grafu je zrejmé, že $X = \{a, b, c\}$. Ak chceme rozhodnúť o akceptovaní hociktorého slova $w \in X^+$, musíme rozhodnúť o hodnote $\mu(\widehat{\delta}(r, w))$. Vidíme, že $\mu(\widehat{\delta}(r, w)) = 1$ práve vtedy, keď $\widehat{\delta}(r, w) = t$. Jedine v stave t je totiž $\mu(t) = 1$. Vidíme, že do stavu t vstupujú len tie hrany, ktoré sú ohodnotené vstupným písmenom c. A pretože hrany ohodnotené vstupným písmenom c nevstupujú do iného vrchola, je zrejmé, že $\mu(\widehat{\delta}(r, w)) = 1$ práve vtedy, keď slovo w končí písmenom c. Preto $L(A) = \{wc;\ w \in X^+\} \cup \{c\}$. ■

POZNÁMKA 3.2. Pri štúdiu konečných akceptorov je zvykom stav s, v ktorom je $\mu(s) = 1$, vyhlásiť za finálny stav. Preto má zmysel uvažovať o množine $F \subset S$, kde $F = \{s \in S;\ \mu(s) = 1\}$. Stavy z množiny F sa v grafe konečného akceptora vyznačujú dvojitým kruhom. V takomto prípade už nemusíme definovať výstupnú funkciu $\mu : S \to Z$ a pripisovať hodnoty výstupnej funkcie k stavu.

Pri takomto prístupe je konečný akceptor pätica $A = (S, X, \delta, s_0, F)$, kde význam prvých štyroch zložiek zostáva nezmenený z pôvodnej definície a $F \subset S$. Množinu F nazývame ***množina finálnych stavov***.

Potom jazyk akceptovaný akceptorom A je množina $L(A) = \{w \in X^+;\ \widehat{\delta}(s_0, w) \in F\}$.

PRÍKLAD 3.9. Opíšeme jazyk L(A), ktorý akceptuje akceptor daný grafom na obr. 14.



OBR. 14. Graf akceptora z príkladu 3.9

Riešenie. Z grafu vyplýva, že $X = \{a, b\}$ a $F = \{s_1\}$. Teda $w \in X^+$ bude akceptované práve vtedy, keď $\widehat{\delta}(s_0, w) = s_1$. Z grafu je zrejmé, že $\widehat{\delta}(s_0, w) = s_1$, práve vtedy, keď slovo w končí písmenom a. Preto $L(A) = \{a\} \cup \{wa;\ x \in X^+\}$. ■

PRÍKLAD 3.10. Uvažujme o abecede $X = \{0, 1, \ldots, 9\}$. Slová zapísané v tejto abecede budeme pokladať za čísla vyjadrené v desiatkovej sústave. V X^+ sa budú vyskytovať aj slová, ktoré začínajú 0. Napríklad 008 je iné slovo ako 08, a to je iné slovo, ako je 8. Ale v desiatkovej sústave budú tieto slová reprezentovať to isté číslo.
Teraz zostrojíme akceptor, ktorý bude akceptovať práve tie slová (čísla), ktoré sú deliteľné troma.

Riešenie. Vieme, že číslo je deliteľné troma práve vtedy, keď súčet jeho cifier je deliteľný troma, resp. súčet cifier po delení troma dáva zvyšok 0. Preto má zmysel uvažovať o troch stavoch:

- s_0 zodpovedá situácii, keď súčet cifier po delení troma dáva zvyšok 0.
- s_1 zodpovedá situácii, keď súčet cifier po delení troma dáva zvyšok 1.
- s_2 zodpovedá situácii, keď súčet cifier po delení troma dáva zvyšok 2.

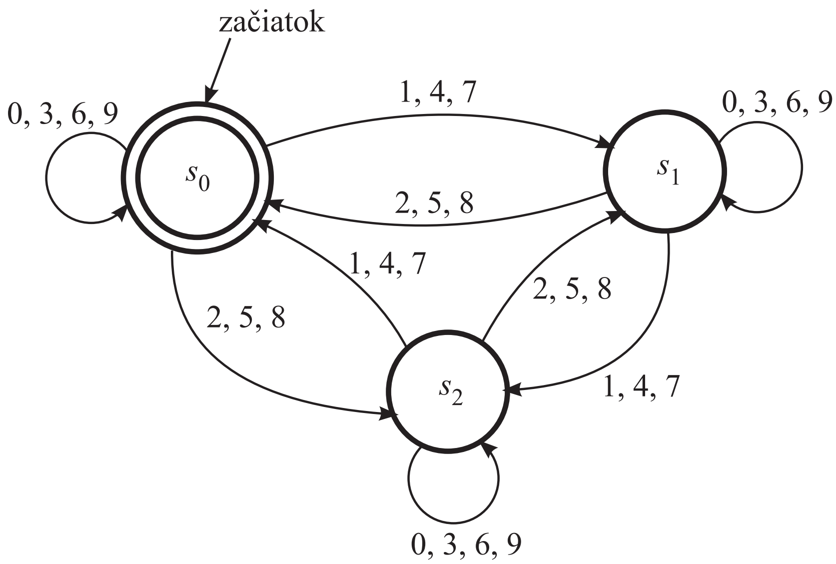
Pretože na začiatku môžeme pokladať súčet cifier za nulový, bude s_0 začiatočný stav. Slovo w bude akceptované iba vtedy, keď zvyšok po delení súčtu cifier číslom tri je nulový. Preto s_0 bude aj jediný finálny stav. Teda $F = s_0$. Graf hľadaného akceptora je na obr. 15. V tomto grafe sme pre prehľadnosť kreslili vždy iba jednu hranu, ktorá spája dva vrcholy, pritom sme k nej pripísali všetky vstupné písmená (číslice), ku ktorým patria príslušné hrany. ■

PRÍKLAD 3.11. V tomto príklade dokážeme, že neexistuje konečný akceptor, ktorý v abecede $Z = \{a, b\}$ akceptuje jazyk $L = \{a^n b^n;\ n = 1, 2, \ldots\} = \{ab, aabb, aaabbb, \ldots\}$.

Riešenie. Predpokladáme, že existuje akceptor $A = (S, X, \delta, s_0, F)$, ktorý akceptuje jazyk L. To znamená, že $\delta(s_0, w) \in F$ práve vtedy, keď existuje prirodzené číslo n také, že $w = a^n b^n$. Vo zvyšných prípadoch $\delta(s_0, w) \notin F$.

Nech množina stavov akceptora má m prvkov. Uvažujme o slove $w = a^i = aa \ldots a$, kde $i > m$. Potom $\delta(s_0, a) = \delta(s_0, a^1) = s^{(1)}$, $\widehat{\delta}(s_0, aa) = \widehat{\delta}(s_0, a^2) = \delta(s^{(1)}, a) = s^{(2)}$, $\widehat{\delta}(s_0, aaa) = \widehat{\delta}(s_0, a^3) = \delta(s^{(2)}, a) = s^{(3)}, \ldots, \widehat{\delta}(s_0, a^i) = s^{(i)}$. Pretože rôznych stavov je iba m a $i > m$, medzi stavmi $s^{(1)}, s^{(2)}, \ldots, s^{(i)}$ musia existovať aspoň dva rovnaké stavy. Teda existujú prirodzené čísla q, r, také, že $s^{(q)} = s^{(r)}$ a $q \neq r$. Pre jednoduchosť predpokladajme, že $r < q$. Potom platí $\widehat{\delta}(s_0, a^r b^r) \in F$ a $\widehat{\delta}(s_0, a^q b^r) \notin F$. Ale

$$\widehat{\delta}(s_0, a^r b^r) = \widehat{\delta}(\widehat{\delta}(s_0, a^r), b^r) = \widehat{\delta}(s^{(r)}, b^r) = \widehat{\delta}(s^{(q)}, b^r) = \widehat{\delta}(\widehat{\delta}(s_0, a^q), b^r) = \widehat{\delta}(s_0, a^q b^r).$$



Obr. 15. Graf akceptora z príkladu 3.10

To už dáva požadovaný spor. To znamená, že akceptor, ktorý by akceptoval jazyk L, neexistuje. ■

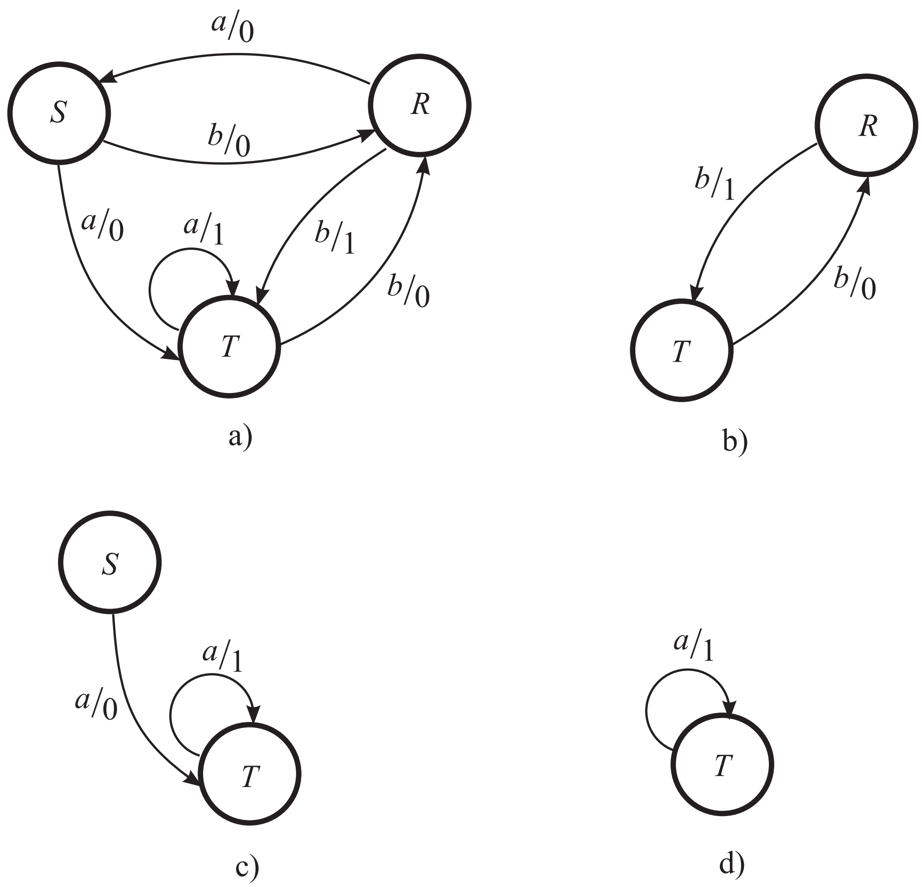
3. Základné pojmy v teórii konečných automatov

V nasledujúcich častiach sa budeme v prevažnej miere zaoberať iba Mealyho automatmi. Pretože ku každému Moorovmu automatu existuje silno ekvivalentný Mealyho automat, môžeme prostredníctvom tejto ekvivalencie preniesť znalosti z oblasti Mealyho automatov do oblasti Moorových automatov.

Definícia 3.8. Nech $A = (S, X, Z, \delta, \lambda)$ je automat. Nech $A' = (S', X', Z', \delta', \lambda')$ je taký automat, že $S' \subset S$, $X' \subset X$, $Z' \subset Z$ a pre funkcie $\delta' : S' \times X' \to S'$ a $\lambda' : S' \times X' \to Z'$ platí: pre každé $x \in X'$ a každé $s \in S'$ je $\delta'(s, x) = \delta(s, x) \in S'$ a $\lambda'(s, x) = \lambda(s, x) \in Z'$. Potom automat A' nazývame ***podautomat automatu*** A.

Príklad 3.12. Uvažujme o automate, ktorý je daný grafom na obr. 16, časť a). Je zrejmé, že jeden z podautomatov je aj daný automat. Každý podautomat, rôzny od daného automatu, budeme nazývať ***vlastný podautomat***. Ak vstupná abeceda obsahuje viac ako jedno písmeno, vlastný podautomat môžeme získať tak, že aspoň jedno písmeno zo vstupnej abecedy vynecháme. V našom príklade môžeme získať dva vlastné podautomaty tým, že vynecháme buď vstupné písmeno a, alebo vstupné písmeno b. Na grafe sa to prejaví vynechaním hrán, ktoré sú ohodnotené príslušným vynechaným písmenom.

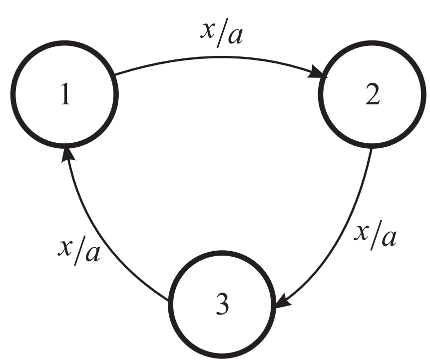
Ak v tomto príklade chceme získať podautomat, ktorý má menej stavov ako pôvodný automat, môžeme uvažovať len o vynechaní stavu, do ktorého nevchádzajú hrany ohodnotené všetkými vstupnými písmenami. V našom príklade z toho dôvodu nemôžeme vynechať stav T. V prípade, že vynecháme stav R, musíme vynechať aj vstupné písmeno b, lebo do stavu R vchádzajú hrany ohodnotené týmto písmenom. Graf tohto podautomatu je na obr. 16, časť c). Podobne ak chceme vynechať stav S, musíme vynechať vstupné písmeno a. Graf príslušného podautomatu je na obr. 16, časť b). Je zrejmé, že automat, ktorý je na obr. 16, časť c), má vlastný podautomat, ktorý získame vynechaním stavu S. Graf tohto podautomatu (ktorý je podautomatom aj pôvodného automatu) je na obr. 16, časť d). ■



Obr. 16. Graf automatu z príkladu 3.12

Teraz ukážeme príklad automatu, ktorý nemá vlastný podautomat. Z predošlého príkladu je zrejmé, že vstupná abeceda takéhoto automatu musí obsahovať jediné písmeno.

Príklad 3.13. Na obr. 17 je graf automatu, ktorý neobsahuje vlastný podautomat. Tento výsledok je dosť zrejmý, lebo vstupné písmeno je jediné, preto ho vynechať nemôžeme. Podobne je to aj s výstupným písmenom. Žiadny zo stavov vynechať nemôžeme, lebo do každého stavu vstupuje hrana ohodnotená (jediným) vstupným písmenom, ktoré nemôžeme vynechať. ■



Obr. 17. Graf automatu z príkladu 3.13

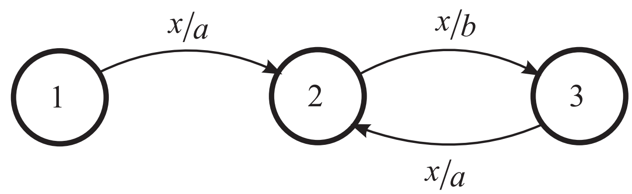
Ako sme si už mohli všimnúť, pojem podautomat, úzko súvisí s pojmom podgraf grafu. Teraz sa budeme zaoberať pojmami, ktoré súvisia s pojmom súvislý i silno súvislý orientovaný graf.

DEFINÍCIA 3.9. Nech $A = (S, X, Z, \delta, \lambda)$ je automat. Budeme hovoriť, že ***stav*** $s \in S$ ***je dosiahnuteľný zo stavu*** $t \in S$, ak buď $s = t$, alebo existuje $w \in X^+$ také, že $s = \widehat{\delta}(t, w)$.

Automat A sa nazýva ***súvislý automat*** ak existuje stav $s \in S$, z ktorého je každý stav automatu dosiahnuteľný.

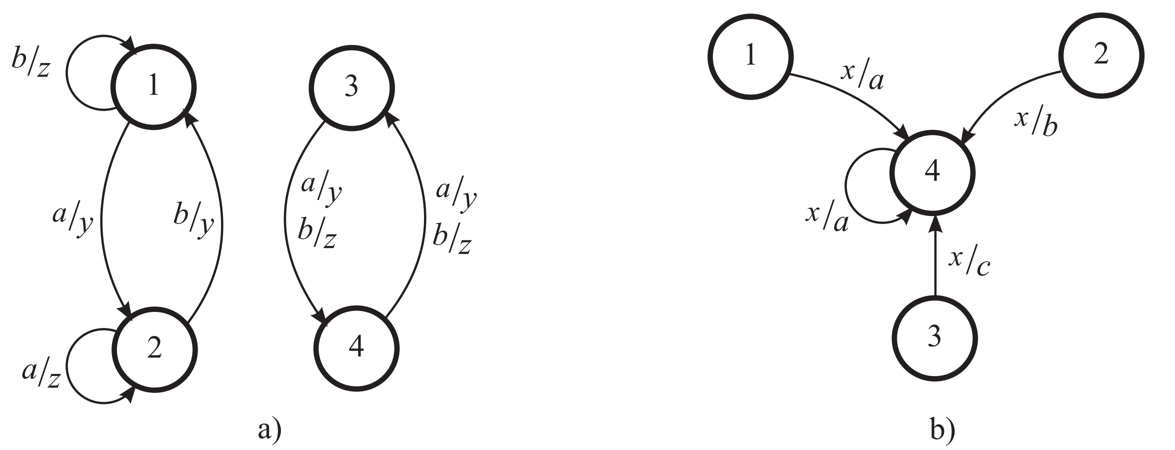
Automat A sa nazýva ***silno súvislý***, ak z každého stavu tohto automatu je každý iný stav dosiahnuteľný.

PRÍKLAD 3.14. Na obr. 18 je graf súvislého automatu, ktorý nie je silno súvislý (stav 1 nie je dosiahnuteľný zo stavu rôzneho od stavu 1).



OBR. 18. Graf súvislého automatu

Na obr. 19, časť a), b) uvádzame grafy automatov, ktoré nie sú súvislé. Všimnime si, že automat, ktorého graf je v časti b), nie je súvislý aj napriek tomu, že jeho graf sa skladá iba z jednej časti.



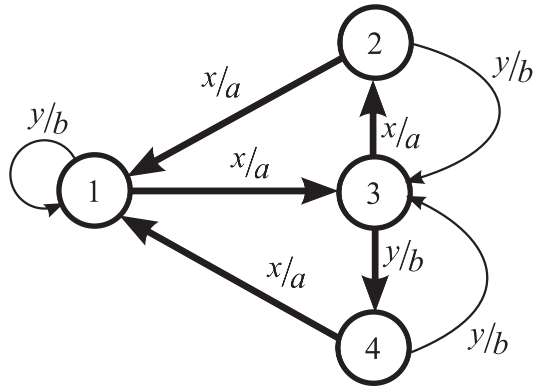
OBR. 19. Príklady grafov, ktoré nie sú súvislé

Na obr. 20 uvádzame graf silno súvislého automatu (silnejšie sú vyznačené hrany, ktoré ukazujú možnosť dosiahnuť každý stav z hociktorého iného stavu). ■

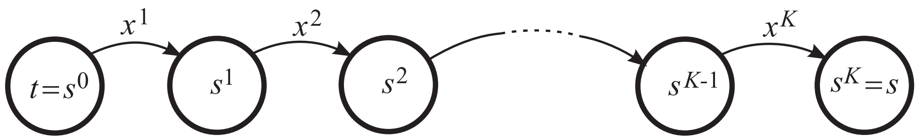
VETA 3.1. Nech je automat $A = (S, X, Z, \delta, \lambda)$, ktorého množina stavov má n prvkov. Ak stav s je dosiahnuteľný zo stavu t, pričom $s \neq t$, potom existuje slovo $w \in X^+$ dĺžky menšej ako n ($|w| \leq n - 1$), ktoré toto dosiahnutie sprostredkuje ($\widehat{\delta}(t, w) = s$).

Dôkaz. Nech $s \in S$, $s \neq t$. Nech existuje $w \in X^+$ také, že $\widehat{\delta}(t, w) = s$. To znamená, že množina $M = \{w \in X^+; \widehat{\delta}(t, w) = s\}$ nie je prázdna. Nech $K = min\{|w|;\ w \in M\}$ (je to dĺžka najkratšieho slova, ktoré sprostredkuje dosiahnuteľnosť stavu s zo stavu t). Predpokladajme, že $K \geq n$. Nech $w_0 \in M$ je také, že $|w_0| = K$ (w_0 je najkratšie možné slovo, ktoré sprostredkuje dosiahnuteľnosť stavu s zo stavu t). Nech $w_0 = x^1 \dots x^K$. Túto situáciu znázorňujeme na obr. 21.

V tejto schéme sa nachádza $K + 1$ stavov, pričom $K \geq n$. Keďže počet rôznych stavov je n, musia sa v postupnosti $t = s^0, s^1, s^2, \dots, s^{K-1}, s^K = s$ nachádzať aspoň dva stavy,

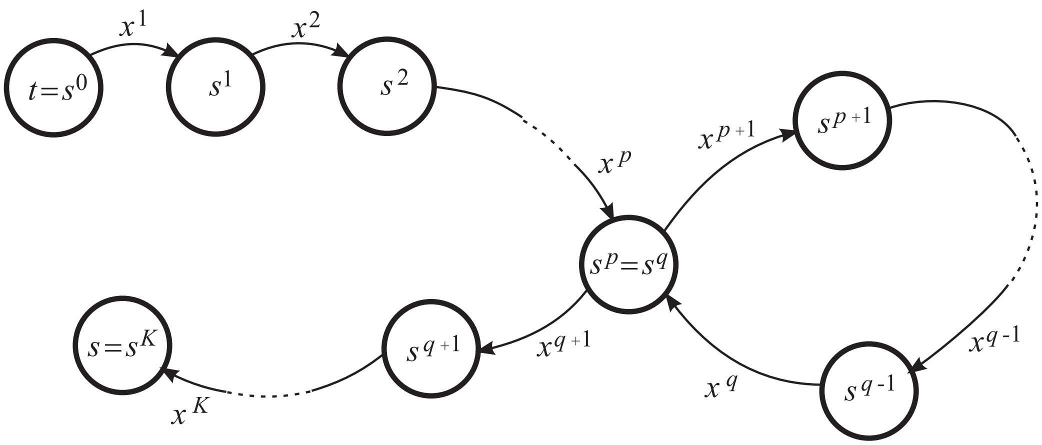


Obr. 20. Graf silno súvislého automatu



Obr. 21. Ilustrácia dôkazu vety 3.1

ktoré sa navzájom rovnajú. Preto musia existovať dve prirodzené čísla p, q také, že $s^p = s^q$. Pritom bez ujmy na všeobecnosti môžeme predpokladať, že $p < q$. Teda v grafe daného automatu musia existovať hrany spájajúce stavy t a s tak, ako je to znázornené na obr. 22.



Obr. 22. Ilustrácia dôkazu vety 3.1

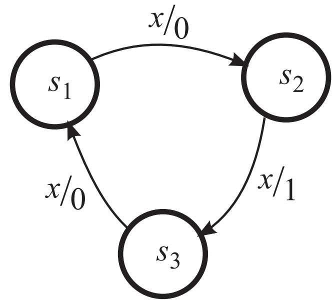
Ak $\widehat{\delta}(t, w) = s$ a $s^p = s^q$ potom

$$\begin{aligned} s &= \widehat{\delta}(t, w) = \widehat{\delta}(t, x^1x^2 \dots x^p \dots x^q \dots x^K) = \\ &= \widehat{\delta}(\widehat{\delta}(t, x^1x^2 \dots x^q), x^{q+1} \dots x^K) = \widehat{\delta}(s^q, x^{q+1} \dots x^K) = \widehat{\delta}(s^p, x^{p+1} \dots x^K) = \\ &= \widehat{\delta}(\widehat{\delta}(t, x^1x^2 \dots x^p), x^{q+1} \dots x^K) = \widehat{\delta}(t, x^1x^2 \dots x^px^{q+1} \dots x^K). \end{aligned}$$

Z toho vidíme, že na dosiahnutie stavu s zo stavu t stačí slovo $x^1x^2 \dots x^px^{q+1} \dots x^K$, ktoré má dĺžku $K - (q - p) < K$. To je spor s predpokladom, že K je najmenšia možná dĺžka slova pomocou ktorého je možné dosiahnuť stav s zo stavu t. Preto musí byť $K < n$. □

Na nasledujúcom (veľmi jednoduchom) príklade ukážeme, že odhad urobený v predošlej vete sa už nedá zlepšiť.

Príklad 3.15. Uvažujme o automate, ktorý je daný pomocou grafu na obr. 23. Tento automat je silno súvislý a obsahuje 3 stavy, teda $n = 3$. Vidíme, že na dosiahnutie stavu s_3, zo stavu s_1, treba slovo xx ($\widehat{\delta}(s_1, xx) = s_3$). Toto slovo má dĺžku $2 = n - 1$. ■



Obr. 23. Graf automatu z príkladu 3.15

4. Neúplne špecifikované (nedeterministické) automaty

Problematiku neúplne špecifikovaných automatov uvedieme na príklade.

Príklad 3.16. v informačnom kanáli prenášame dva symboly A a B, ktoré sú zakódované znakmi 1 a 0. Aby sme zvýšili bezpečnosť dekódovania, pridávame k týmto kódovacím znakom ešte dva znaky 0 a 1 tak, aby 1 značila začiatok prenášanej trojice a 0 koniec prenášanej trojice. Potom A je zakódované trojicou 110 a B trojicou 100.

Na výstupe tieto trojice dekódujeme pomocou blokového dekódera a jedine trojici 110 pri dekódovaní priradíme znak A a jedine trojici 100 priradíme znak B. Vo zvyšných prípadoch ide o chybný prenos.

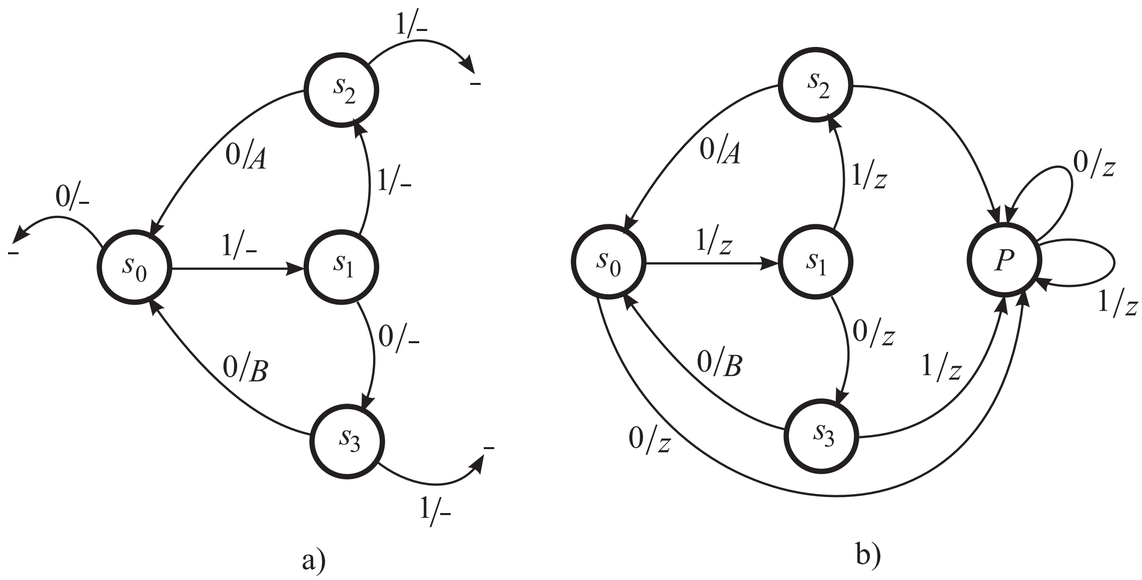
Teraz opíšeme toto dekódovacie zariadenie ako iniciálny automat.

Riešenie. Trojicu začneme dekódovať v začiatočnom stave s_0. Ak sa v tomto stave na vstupe dekódera objaví 0, vieme, že ide o chybný prenos. Vtedy môžeme buď dekódovanie trojice prerušiť tak, že v stave s_0 pri vstupe 0 nebudeme definovať ďalšiu činnosť, alebo hneď na začiatku definujeme stav P = pasca, do ktorého budú viesť hrany ohodnotené vstupným písmenom, keď budeme vedieť, že ide o chybný prenos. V takom prípade sa už zo stavu P pri nijakom vstupnom písmene nedostaneme.

Ak v stave s_0 sa na vstupe dekódera objaví písmeno 1, je možný bezchybný prenos, čo vyznačíme prechodom do nového stavu s_1. Výstup nás v tomto prípade ešte nezaujíma. Preto ho nemusíme definovať. V stave s_1 pri vstupe 1 je predpoklad, že bol vyslaný signál A. To zachytíme definovaním nového stavu s_2. Ak v stave s_1, je na vstupe 0, je predpoklad, že bol vyslaný signál B. To bude zaznamenané stavom s_3. V oboch prípadoch výstup ešte nie je zaujímavý. Prenos signálu A alebo B bude potvrdený, ak v stave s_2 alebo s_3 na vstupe objaví 0. V opačnom prípade pôjde o chybný prenos. Na obr. 24, časť a) graficky znázorňujeme proces dekódovania v prípade, že pri chybnom prenose zastavíme činnosť dekódera. V obr. 24, časť b) sme graf z obr. 24, časti a) doplnili na graf automatu tým, že v prípade chybného prenosu budú hrany smerovať do pomocného stavu P, z ktorého sa už nedostaneme.

Okrem prípadu, keď tretí znak v trojici na vstupe dekódera sa rovná 0, výstup nás v tomto príklade nezaujíma. Pri prvom prístupe ho nešpecifikujeme, Pri druhom prístupe pridáme pomocné výstupné písmeno z.

Aj pri prvom prístupe sme činnosť dekódera opísali podobne ako v prípade automatu, len prechodová a výstupná funkcia neboli definované v každom bode. V tomto prípade to boli iba parciálne funkcie. ■



Obr. 24. Graf dekódera z príkladu 3.16

Definícia 3.10. ***Neúplne špecifikovaným*** (nedeterministickým) ***automatom*** nazývame päticu $A = (S, X, Z, \delta, \lambda)$, kde S, X, Z sú konečné množiny stavov, vstupov, výstupov a $\delta : S \times X \to S$ a $\lambda : S \times X \to Z$ sú parciálne funkcie (nie sú definované pre každú dvojicu $(s, x) \in S \times X$). Funkcie δ a λ nazývame ***prechodová a výstupná funkcia neúplne špecifikovaného automatu***.

Pri prvom prístupe sme na opis dekódera z príkladu 3.16 už vlastne použili neúplne špecifikovaný automat. Jeho tabuľku uvádzame v tabuľke 6, časť a).

Tabuľka 6. Tabuľky automatu z príkladu 3.16

	δ		λ	
	0	1	0	1
s_0	-	s_1	-	-
s_1	s_3	s_2	-	-
s_2	s_0	-	A	-
s_3	s_0	-	B	-

a)

	δ		λ	
	0	1	0	1
s_0	-	s_1	-	-
s_1	s_3	s_2	-	-
s_2	s_0	-	A	-
s_3	s_0	-	B	-
-	-	-	-	-

b)

	δ		λ	
	0	1	0	1
s_0	P	s_1	z	z
s_1	s_3	s_2	z	z
s_2	s_0	P	A	z
s_3	s_0	P	B	z
P	P	P	z	z

c)

V neúplne špecifikovaných automatoch pri nedefinovaných stavoch a výstupoch sa prakticky vyskytujú len dve možnosti. Buď je prechod zakázaný pomocou technických prostriedkov pri fyzikálnej realizácii, alebo v týchto prípadoch stav alebo výstup nie sú zaujímavé a môžeme ich ľubovoľne dodefinovať.

V každom prípade môžeme o neúplne špecifikovaných automatoch uvažovať ako o konečnom automate, keď znak „-“ prijmeme za nový stav a aj nový výstupový symbol. V takom prípade hovoríme o ***rozšírenom automate***. Opis dekódera z príkladu 3.16 ako rozšíreného automatu uvádzame v tabuľke 6, časť b). Všimnime si, že tento prístup zodpovedá nášmu druhému prístupu k opisu dekódera v príklade 3.16 ako ku konečnému automatu, ktorý vznikol z nekompletne špecifikovaného automatu pomocou doplnenia množiny stavov stavom P a množiny výstupných písmen pomocným znakom z. Tento opis dekódera uvádzame v tabuľke 6, časť c).

Ak uvažujeme o neúplne špecifikovanom automate ako o automate, v ktorom môžeme nešpecifikované stavy a výstupy ľubovoľne doplniť, potom sa veľa problémov z teórie automatov dá pomocou neúplne špecifikovaných automatov značne zjednodušiť. Tento prístup k neúplne špecifikovaným automatom budeme dôsledne zachovávať. Neúplne špecifikovaný automat bude iba kostra konečného automatu, ktorej konštrukcia vyplynie zo zadania úlohy. Túto kostru vždy doplníme na konečný automat. Preto ak budeme používať pojmy, ktoré definujeme pre konečné automaty, pri práci s neúplne špecifikovanými automatmi, budeme mať vždy na mysli výsledný automat, na ktorý tento neúplne špecifikovaný automat doplníme.

Sme si vedomí toho, že tento prístup je značným (a nie vždy vhodným) zjednodušením problematiky neúplne špecifikovaných automatov. Pri neúplne špecifikovaných automatoch je zložitý postup pri redukcii automatu.

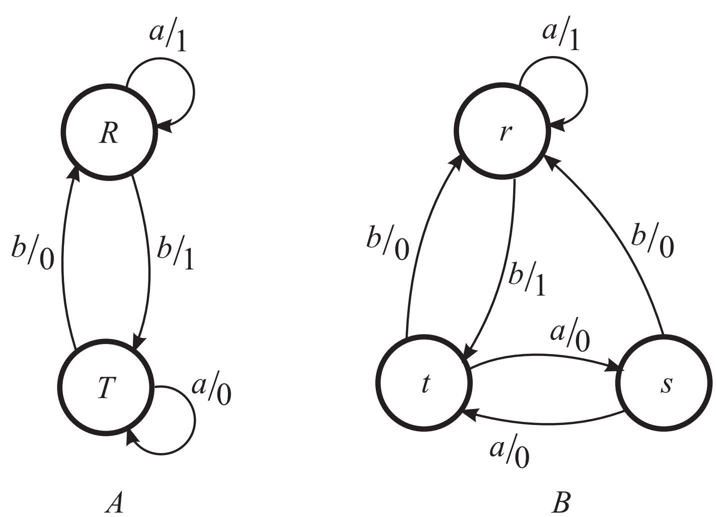
5. Ekvivalencia automatov

V tejto časti sa budeme zaoberať automatmi, ktoré ako reakciu na rovnaké vstupné slovo sú schopné generovať rovnakú výstupnú postupnosť. Preto budeme uvažovať o automatoch, ktoré majú spoločné vstupné a aj výstupné abecedy.

Definícia 3.11. Nech $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$. Budeme hovoriť, že ***stavy*** $s \in S$ a $t \in T$ ***sú ekvivalentné***, ak pre každé vstupné slovo $w \in X^+$ je $\widehat{\lambda}_A(s, w) = \widehat{\lambda}_B(t, w)$.

V prípade, že s a t sú ekvivalentné stavy, budeme písať $s \sim t$.

Príklad 3.17. Nech $A = \{\{R, T\}, \{a, b\}, \{0, 1\}, \delta_A, \lambda_A\}$ a $B = \{\{r, s, t\}, \{a, b\}, \{0, 1\}, \delta_B, \lambda_B\}$ sú automaty, ktorých grafy uvádzame na obr. 25.



Obr. 25. Grafy automatov z príkladu 3.17

Uvažujme o stavoch R a t. Z grafov automatov vidíme, že $\lambda_A(R, a) = 1 \neq 0 = \lambda_B(t, a)$. Preto je zrejmé, že R a t nie sú ekvivalentné stavy (píšeme $R \not\sim t$).

Teraz sa pokúsime dokázať, že R a r sú ekvivalentné.

Riešenie. Uvažujme o výstupných postupnostiach $\widehat{\lambda}_A(R, w)$ a $\widehat{\lambda}_B(r, w)$ Pri ľubovoľnom $w \in X^+$. Ak vstupné slovo w začína postupnosťou vstupných znakov a, potom obe výstupné postupnosti začínajú postupnosťou jednotiek a oba automaty zostávajú v pôvodných stavoch. Po prvom vstupnom písmene b zmenia oba automaty stav a výstup sa ešte stále rovná jednej. Pri ďalších znakoch b oba automaty menia stav z T na R, resp. z t na r a výstup z 1 na 0. Pri nasledujúcom a, ak sú v stave R, resp. r, sa situácia opakuje. Ak sú v stave T, resp. t, prvý ostane v stave T a druhý automat strieda stavy t a s. V oboch prípadoch sú však výstupy stále rovnaké a rovnajú sa 0. Nasledujúci znak b spôsobí prechod do stavu R v prvom automate a do stavu r v druhom automate. Pri tomto prechode sa výstup v oboch prípadoch rovná 0. Potom sa už situácia opakuje.

Tak sme dosť zložitým a neprehľadným spôsobom dokázali, že pre každé $w \in X^+$ je $\widehat{\lambda}_A(R,w)=\widehat{\lambda}_B(r,w)$ a teda $r \sim R$.

Uvedený dôkaz ekvivalencie stavov bol dosť neprehľadný a bol založený na náhode. V nasledujúcej časti opíšeme algoritmus, pomocou ktorého sa dajú systematicky vyhľadávať ekvivalentné stavy. ■

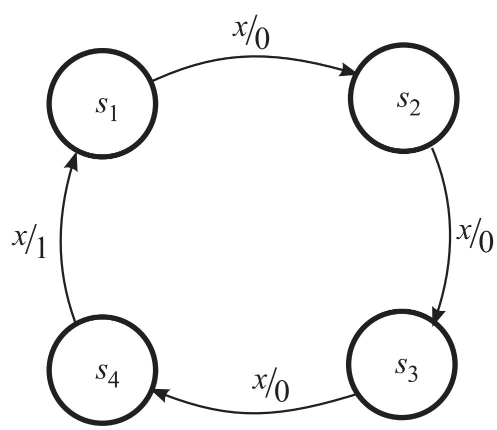
Poznámka 3.3. O ekvivalencii dvoch stavov má zmysel hovoriť aj v tom prípade, keď automat A je zhodný s automatom B. V prípade, že s a t sú stavy toho istého automatu a $s \sim t$, budeme písať sEt.

Všimnime si, že takto definovaná relácia E na množine stavov S automatu A je naozaj relácia ekvivalencie (je zrejmé, že je reflexívna, symetrická a tranzitívna). Táto ekvivalencia indukuje rozklad na množine stavov S. Tento rozklad budeme značiť rovnako ako ekvivalenciu, ktorá ho indukuje, teda znakom E.

Pri fyzikálnej realizácii automatov uvidíme, že zložitosť (a teda aj cena) fyzikálnej realizácie automatu závisí od počtu jeho stavov. Pretože dva ekvivalentné stavy na vstupné slovo reagujú rovnakou výstupnou postupnosťou riadiacich signálov, nie je ekonomické pracovať s automatmi, ktoré majú veľa ekvivalentných stavov.

Definícia 3.12. Automat A nazývame ***redukovaný automat***, ak ľubovoľné dva jeho rôzne stavy nie sú ekvivalentné.

Príklad 3.18. Ukážeme, že automat daný grafom na obr. 26 je redukovaný.



Obr. 26. Graf automatu z príkladu 3.18

Riešenie. V danom automate existuje jediné vstupné slovo dĺžky 1. Je to $w = x$. Pre toto slovo dostávame $\lambda(s_1,x) = \lambda(s_2,x) = \lambda(s_3,x) = 0 \neq 1 = \lambda(s_4,x)$, teda $s_4 \not\sim s_1, s_2, s_3$. Pre slovo dĺžky dva dostávame: $\widehat{\lambda}(s_1,xx) = \widehat{\lambda}(s_2,xx) = 00 \neq 01 = \widehat{\lambda}(s_3,xx)$, čiže $s_3 \not\sim s_1, s_2$. Ďalej $\widehat{\lambda}(s_1,xxx) = 000$ a $\widehat{\lambda}(s_2,xxx) = 001$. Preto aj $s_1 \not\sim s_2$. Z toho už vyplýva, že automat je redukovaný. ■

Definícia 3.13. Nech $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$ sú automaty. Nech ku každému stavu $s \in S$ existuje stav $t \in T$, ktorý je s ním ekvivalentný. Nech aj naopak pre každé $t \in T$ existuje $s \in S$ také, že $t \sim s$. Potom budeme hovoriť, že ***automaty*** A a B ***sú ekvivalentné***.

Z definície ekvivalencie automatov A a B vyplýva, že tieto automaty sú ekvivalentné práve vtedy, keď existujú funkcie $f_1 : S \to T$ a $f_2 : T \to S$ také, že pre každé $s \in S$ a každé $t \in T$ je $s \sim f_1(s)$ a $t \sim f_2(t)$.

Teraz chceme ku každému automatu A priradiť redukovaný automat A_R, ktorý bude ekvivalentný s automatom A. Pri plnení tohto cieľa nám v značnej miere pomôže táto úvaha.

Nech $A = (S, X, Z, \delta_A, \lambda_A) = B = (T, X, Z, \delta_B, \lambda_B)$ sú ekvivalentné automaty (budeme písať $A \sim B$). Na množine S je definovaná ekvivalencie E_A pomocou podmienky rE_As práve vtedy, keď pre každé $w \in X^+$ je $\widehat{\lambda}_A(r, w) = \widehat{\lambda}_A(s, w)$. Podobne na množine stavov T je definovaná ekvivalencia E_B. Tieto ekvivalencie indukujú na množinách S a T rozklady na triedy ekvivalencie $E_A = \{A_1, \dots, A_m\}$ a $E_B = \{B_1, \dots, B_n\}$.

Pretože $A \sim B$, musí ku každému $s \in S$ existovať $t \in T$ s vlastnosťou $s \sim t$. Predpokladajme, že pre dva prvky $r, s \in S$ je $r \sim t$ a $s \sim t$ $(t \in T)$. Potom pre každé $w \in X^+$ platí: $\widehat{\lambda}_A(r, w) = \widehat{\lambda}_B(t, w) = \widehat{\lambda}_A(s, w)$. Z toho už vyplýva, že rE_As, a teda r, s ležia v tej istej triede ekvivalencie E_A.

Naopak. nech rE_As a $r \sim t$. Potom z podmienky $\widehat{\lambda}_A(s, w) = \widehat{\lambda}_A(r, w) = \widehat{\lambda}_B(t, w)$ vyplýva, že $s \sim t$.

Tým sme dokázali, že pre každé $t \in T$ existuje taká trieda A_i ekvivalencie E_A, že pre každé $s \in A_i$ je $s \sim t$, teda $A_i = \{s \in S; s \sim t\}$ Ale k prvku t existuje aj trieda B_j ekvivalencie E_B, v ktorej sa tento prvok nachádza. Potom pre každé $u \in B_j$ a každé $s \in A_i$ dostávame uE_Bt a $t \sim s$. Z toho dostávame $u \sim s$.

To znamená, že pre každú triedu A_i ekvivalencie E_A existuje (práve) jedna trieda B_j ekvivalencie E_B, taká že pre každé $s \in A_i$ a každé $t \in B_j$ je $s \sim t$. Je zrejmé, že platí aj opačné tvrdenie. To znamená, že medzi rozkladmi $E_A = \{A_1, \dots, A_m\}$ a $E_B = \{B_1, \dots, B_n\}$ existuje bijekcia $f : E_A \to E_B$. Preto $m = n$ a bez ujmy na všeobecnosti môžeme označiť $f(A_i) = B_i$. (Prečíslujeme triedy ekvivalencie E_B tak, aby $f(A_1) = B_1$, $f(A_2) = B_2, \dots, f(A_n) = B_n$.) Toto označenie budeme používať aj neskoršie.

Pri definícii redukovaného automatu patriaceho k danému automatu využijeme vetu týkajúcu sa postupností výstupných písmen, t.j. slov vo voľnej pologrupe Z^+. Tu chceme poznamenať, že slová v každej voľnej pologrupe sú konečné postupnosti písmen a pre takéto dve postupnosti platí rovnosť práve vtedy, keď tieto postupnosti majú rovnaký počet prvkov a prvky na prvých, druhých, tretích, … miestach uvedených postupností sa navzájom rovnajú. Z toho vyplýva, ak X^+ je ľubovoľná voľná pologrupa a $u = x^1 \dots x^m$, $v = y^1 \dots y^m$ sú dve slová z tejto pologrupy, potom $u = v$ práve vtedy, keď $m = n$ a $x^i = y^i$ pre $i = 1, 2, \dots, m$.

V nasledujúcich častiach budeme pre ľubovoľný automat $A = (S, X, Z, \delta, \lambda)$ označovať znakom E (ak bude treba odlíšiť od iného automatu E_A) ekvivalenciu na množine stavov S, ktorá je daná podmienkou: aEt práve vtedy, keď pre každé $w \in X^+$ je $\widehat{\lambda}(s, w) = \widehat{\lambda}(t, w)$.

Veta 3.2. Nech $A = (S, X, Z, \delta, \lambda)$ je automat a sEt. Potom $\delta(s, x)E\delta(t, x)$ pre každé $x \in X$.

Dôkaz. Nech $w \in X^+$ je ľubovoľné slovo. Nech sEt. Chceme dokázať, že pre každé $x \in X$ je $\widehat{\lambda}(\delta(s, x), w) = \widehat{\lambda}(\delta(t, x), w)$.

Nech $x \in X$ je ľubovoľné vstupné písmeno. Potom xw je slovo z X^+. Pretože sEt, musí byť $\widehat{\lambda}(s, xw) = \widehat{\lambda}(t, xw)$. Z toho dostávame $\lambda(s,x)\widehat{\lambda}(\delta(s,x), w) = \lambda(t,x)\widehat{\lambda}(\delta(t,x), w)$. Pretože ide o rovnosť dvoch slov zo Z^+, musí byť $\lambda(s,x) = \lambda(t,x)$ a $\widehat{\lambda}(\delta(s,x), w) = \widehat{\lambda}(\delta(t,x), w)$. Pretože $w \in X^+$ bolo ľubovoľné slovo, z poslednej rovnosti vyplýva, že $\delta(s,x)E\delta(t,x)$ pre ľubovoľné $w \in X^+$. □

Z vety 3.2 dostávame takýto výsledok. Nech $A = (S, X, Z, \delta, \lambda)$ je automat a $E = \{A_1, \ldots, A_n\}$ je rozklad na množine jeho stavov, ktorý je indukovaný ekvivalenciou E. Ak $s, t \in A_i$, pre každé $x \in X$ existuje $j \in \{1, \ldots, n\}$ také, že $\delta(s,x), \delta(t,x) \in A_j$. Ak označíme $\delta(A_i, x) = \{\delta(s,x); s \in A_i\}$, z predošlej vety vyplýva, že pre každé A_i a každé $x \in X$ existuje také A_j, že $\delta(A_i, x) \subset A_j$.

Definícia 3.14. Nech $A = (S, X, Z, \delta, \lambda)$ je automat. Nech $A_R = (S_R, X, Z, \delta_R, \lambda_R)$ je taký automat, že $S_R = E = \{A_1, \ldots, A_n\}$ a funkcie $\delta_R : S_R \times X \to S_R$, $\lambda_R : S_R \times X \to Z$ sú definované takto:

- $\delta_R(A_i, x) = A_j$ práve vtedy, keď $\delta(A_i, x) \subset A_j$,
- $\lambda_R(A_i, x) = \lambda(s, x)$ pre ľubovoľné $s \in A_i$.

Potom automat A_R nazývame ***redukovaný automat*** (z) ***automatu*** A.

Napriek tomu, že v názve automatu A_R sa nachádza slovo „redukovaný“, redukovanosť tohto automatu budeme musieť ešte len dokázať.

Je zrejmé, že výstupné funkcie λ_R automatu A_R je dobre definovaná, lebo pre každé $s, t \in A_i$ podmienka sEt implikuje $\widehat{\lambda}(s, w) = \widehat{\lambda}(t, w)$ a táto rovnosť musí platiť aj pre $w = x \in X$.

Ďalej nech $w = x^1 \ldots x^n \in X^+$ je ľubovoľné slovo. Potom

$$\begin{aligned}
\widehat{\lambda}_R(A_i, w) &= \widehat{\lambda}_R(A_i, x^1 \ldots x^n) = \\
&= \lambda_R(A_i, x^1)\lambda_R(\delta_R(A_i, x^1), x^2)\lambda_R(\delta_R(\delta_R(A_i, x^1), x^2), x^3) \cdots = \\
&= \lambda(s, x^1)\lambda(\delta(s, x^1), x^2)\lambda(\delta(\delta(s, x^1), x^2), x^3) \cdots = \\
&= \lambda(s, x^1)\lambda(\delta(s, x^1), x^2)\lambda(\widehat{\delta}(s, x^1 x^2), x^3) \cdots = \\
&= \widehat{\lambda}(s, x^1 \ldots x^n) = \widehat{\lambda}(s, w)
\end{aligned}$$

pre každé $s \in A_i$.

Z toho vyplýva, že pre každé $s \in S$ existuje $A_i \in S_R$ (práve to A_i, pre ktoré je $s \in A_i$) také že, $s \sim A_i$. Je zrejmé, že aj pre každé $A_i \in S_R$ existuje $s \in S$ také, že $A_i \sim s$ (túto vlastnosť má každé $s \in A_i$). Preto A a A_R sú ekvivalentné automaty.

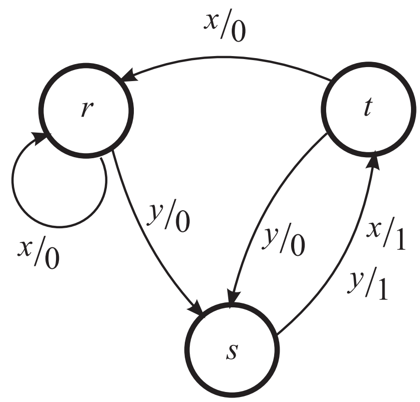
Teraz ukážeme, že automat A_R je naozaj redukovaný automat. Ak $i \neq j$ a $s \in A_i$,$t \in A_j$ potom $s\not{E}t$. Preto existuje $w \in X^+$ také, že $\widehat{\lambda}(s, w) \neq \widehat{\lambda}(t, w)$. Ale $\widehat{\lambda}(s, w) = \widehat{\lambda}_R(A_i, w)$ a $\widehat{\lambda}(t, w) = \widehat{\lambda}_R(A_j, w)$. Preto $\widehat{\lambda}_R(A_i, w) \neq \widehat{\lambda}_R(A_j, w)$. Podmienka $i \neq j$ implikuje $A_i \not\sim A_j$, čiže A_R je redukovaný automat.

Tieto výsledky zhrnieme v tejto vete.

Veta 3.3. Nech A je automat. Automat A_R je redukovaný a platí $A \sim A_R$. □

Príklad 3.19. Nech automat A je daný pomocou grafu na obr. 27. Nájdeme redukovaný automat A_R tohto automatu.

Riešenie. Aby sme túto úlohu rozriešili, potrebujeme nájsť triedy ekvivalencie. Priamo z grafu daného automatu dostávame $\lambda(r, x) = \lambda(t, x) = 0 \neq 1 = \lambda(s, x)$. Preto $r\not{E}s$. V jednej triede ekvivalencie E ešte môžu byť stavy r, t. Pre tieto stavy dostávame:



Obr. 27. Graf automatu z príkladu 3.19

$$\lambda(r,x)=0,\ \lambda(t,x)=0,\ \lambda(r,y)=0,\ \lambda(t,y)=0$$
$$\delta(r,x)=r,\ \delta(t,x)=r,\ \delta(r,y)=s,\ \delta(t,y)=s$$

Z toho vyplýva, že pre každé $w=x^1x^2\dots x^n\in X^+$ platí:

Ak $x^1=x$, tak $\widehat{\lambda}(r,w)=\widehat{\lambda}(t,w)=0\widehat{\lambda}(r,x^2\dots x^n)$.

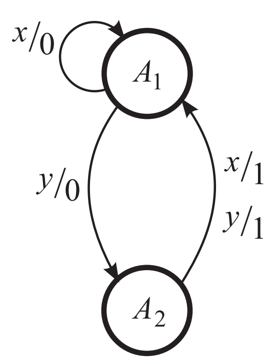
Ak $x^1=y$, tak $\widehat{\lambda}(r,w)=\widehat{\lambda}(t,w)=0\widehat{\lambda}(s,x^2\dots x^n)$.

Pretože už inej možnosti niet, je $r\sim t$. Preto $E=\{\{r,t\},\{s\}\}=\{\overline{r,t},\overline{s}\}=\{A_1,A_2\}$. Preto $A_R=(\{A_1,A_2\},\{x,y\},\{0,1\},\delta_R,\lambda_R)$ je automat, ktorý je daný pomocou tab. 7. Jeho graf vidíme na obr. 28. ■

Tabuľka 7. Tabuľka redukovaného automatu z príkladu 3.19

	x	y
A_1	$A_1/0$	$A_2/0$
A_2	$A_1/1$	$A_1/1$

Teraz už vidíme, že k danému automatu A vieme nájsť redukovaný automat A_R, ak dokážeme nájsť triedy ekvivalencie E. Zatiaľ sa ukazuje, že proces hľadania tried ekvivalencie E nie je finitný. Musíme totiž dokázať, že pre nekonečne veľa slov $w\in X^+$



Obr. 28. Graf redukovaného automatu z príkladu 3.19

platí istá rovnosť. Nie je to, pravdaže, až také komplikované, ako to vyzerá. Aby sme sa o tom presvedčili, budeme definovať nasledujúce označenie.

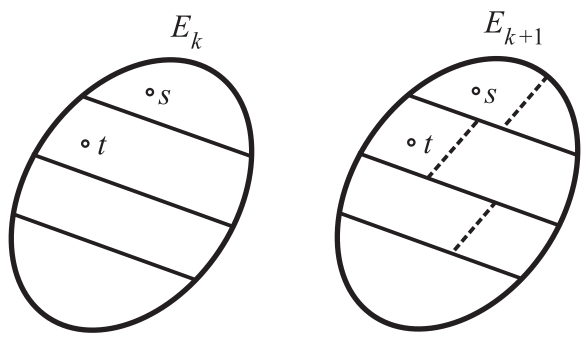
DEFINÍCIA 3.15. Nech je daný automat $A = (S, X, Z, \delta, \lambda)$. Budeme hovoriť, že dva ***stavy*** $s, t \in S$ ***sú k-ekvivalentné***, ak pre každé slovo $w = x^1 \dots x^k \in X^+$, dĺžky k platí:

$$\widehat{\lambda}(s, x^1 \dots x^k) = \widehat{\lambda}(t, x^1 \dots x^k).$$

V takomto prípade budeme písať sE_kt.

Pomocou definície 3.15 sme na množine stavov S pre každé $k = 1, 2, \dots$ definovali reláciu E_k. Je dosť zrejmé, že každá z týchto relácií je reflexívna, symetrická a tranzitívna. Preto je to ekvivalencia. Teda na množine S máme postupnosť ekvivalencií $E_1, E_2, \dots$ a ešte aj ekvivalenciu E. Aký je medzi týmito ekvivalenciami vzťah, to ukážeme v nasledujúcich vetách.

VETA 3.4. Nech $A = (S, X, Z, \delta, \lambda)$ je automat. Potom $E_{k+1} \subset E_k$. To znamená, že podmienka $sE_{k+1}t$ implikuje sE_kt pre $k = 1, 2, \dots$.



OBR. 29. Ilustrácia dôkazu vety 3.4

Dôkaz. Predpokladajme, že $s\not{E}_kt$. Teda existuje slovo $w \in X^+$ dĺžky k také, že $\widehat{\lambda}(s, w) \neq \widehat{\lambda}(t, w)$. Nech $x \in X$ je ľubovoľné vstupné písmeno. Potom wx je slovo dĺžky $k + 1$ a

$$\widehat{\lambda}(s, wx) = \widehat{\lambda}(s, w)\lambda(\widehat{\delta}(s, w), x) \neq \widehat{\lambda}(t, w)\lambda(\widehat{\delta}(t, w), x) = \widehat{\lambda}(t, wx).$$

Z toho vyplýva, že $s\not{E}_{k+1}t$, takže sE_kt je nutnou podmienkou $sE_{k+1}t$. Tým je veta 3.4 dokázaná. □

Z vety 3.4 vyplýva, že každá trieda ekvivalencie E_{k+1} leží v nejakej triede ekvivalencie E_k. V takomto prípade hovoríme, že príslušný rozklad E_{k+1} je zjemnením rozkladu E_k (pozri obr. 29).

Pre postupnosť $E_1, E_2, \dots$ sme dostali nekonečný počet inklúzií, ktoré môžeme vyjadriť takto: $E_1 \supset E_2 \supset E_3 \supset \dots$. Je mysliteľné, že v tejto postupnosti sa dve ekvivalencie rovnajú, teda že tieto inklúzie majú nasledujúci tvar: $E_1 \supset E_2 \supset \dots \supset E_{k-1} = E_k \supset E_{k+1} \supset \dots$. Dokážeme vetu, z ktorej vyplýva, že v takomto prípade už všetky ďalšie inklúzie, počnúc prvou rovnosťou, môžeme nahradiť rovnosťami.

VETA 3.5. Nech $A = (S, X, Z, \delta, \lambda)$ je automat sE_kt. Potom pre každé $x \in X$ je $\delta(s, x)E_{k-1}\delta(t, x)$.

Dôkaz. Nech $w \in X^+$ je slovo dĺžky $k - 1$ a $x \in X$ je ľubovoľné písmeno. Potom xw je slovo dĺžky k. Nech sE_kt. Potom $\widehat{\lambda}(s, xw) = \widehat{\lambda}(t, xw)$. Z toho už dostávame

$$\lambda(s, x)\widehat{\lambda}(\delta(s, x), w) = \lambda(t, x)\widehat{\lambda}(\delta(t, x), w).$$

Pretože ide o rovnosť dvoch slov vo výstupnej abecede, musí byť $\lambda(s, x) = \lambda(t, x)$ a $\widehat{\lambda}(\delta(s, x), w) = \widehat{\lambda}(\delta(t, x), w)$. Z poslednej rovnosti už dostávame $\delta(s, x)E_{k-1}\delta(t, x)$ pre každé $x \in X$. □

Teraz už môžeme vysloviť a dokázať vopred ohlásenú vetu.

VETA 3.6. Nech $A = (S, X, Z, \delta, \lambda)$ je automat. Nech $k \geq 2$ a $E_{k-1} = E_k$, potom $E_k = E_{k+1}$.

Dôkaz. Z vety 3.4 vyplýva, že za predpokladu $E_{k-1} = E_k$ stačí dokázať $E_k \subset E_{k+1}$. Potom z inklúzie $E_k \supset E_{k+1}$, ktorú nám dáva veta 3.4, už dostávame požadovanú rovnosť.

Nech $k \geq 2$ a $E_{k-1} = E_k$. Nech sE_kt. Potom z vety 3.5 vyplýva, že $\delta(s,x)E_{k-1}\delta(t,x)$ pre každé $x \in X$. Z rovnosti $E_{k-1} = E_k$ teraz dostávame, že $\delta(s,x)E_k\delta(t,x)$ pre každé $x \in X$. Nech $w \in X^+$ je ľubovoľné slovo dĺžky $k+1$. Toto slovo si môžeme napísať v tvare $w = xv$, kde $x \in X$ a v je slovo dĺžky k. Potom

$$\widehat{\lambda}(s,w) = \widehat{\lambda}(s,xv) = \lambda(s,x)\widehat{\lambda}(\delta(s,x),v).$$

Pretože je sE_kt, je aj sE_1t. Z toho vyplýva, že $\lambda(s,x) = \lambda(t,x)$. Z podmienky $\delta(s,x)E_k\delta(t,x)$ dostávame $\widehat{\lambda}(\delta(s,x),v) = \widehat{\lambda}(\delta(t,x),v)$. Preto

$$\widehat{\lambda}(s,w) = \lambda(s,x)\widehat{\lambda}(\delta(s,x),v) = \lambda(t,x)\widehat{\lambda}(\delta(t,x),v) = \widehat{\lambda}(t,xv) = \widehat{\lambda}(t,w).$$

Teda $sE_{k+1}t$. Tým je veta 3.6 dokázaná. □

Ak $E_{k-1} = E_k$ potom $E_k = E_{k+1}$. Z tejto rovnosti ďalej vyplýva, že $E_{k+1} = E_{k+2}$ atď.

Nech $E_{k-1} = E_k$, potom

$$E_1 \supset E_2 \supset \cdots \supset E_{k-1} = E_k = E_{k+1} = E_{k+2} = \ldots .$$

Ak je teraz $sE_{k-1}t$, stavy s, t dávajú rovnaké výstupné postupnosti na slová dĺžky $1, 2, \ldots, k-1, k, k+1, k+2, \ldots$. Tieto dva stavy dávajú rovnaké výstupné slovo pre ľubovoľné slovo $w \in X^+$. Preto v tomto prípade je $E_{k-1} = E_k = E_{k+1} = E$.

Teraz sa nám núkajú dve otázky:

1. Musí existovať také k, aby $E_{k-1} = E_k$?
2. Ak také k existuje, pre ako veľké k platí $E_{k-1} = E_k$?

Pokúsime sa dať odpoveď na tieto otázky. Nech $A = (S, X, Z, \delta, \lambda)$ je automat, ktorého množina stavov $S = \{s_1, \ldots, s_n\}$ má n prvkov. Vieme, že platí $E_1 \supset E_2 \supset E_3 \supset \cdots \supset$ $\supset E_n \supset \ldots$. Nech $E_1 \neq E_2$. Nech rozklad E_1 pozostáva z najhrubšieho možného delenia množiny stavov, ktoré pozostáva z jedinej triedy. Potom rozklad E_2 musí pozostávať aspoň z dvoch tried. Ak by všetky rozklady až po rozklad E_n boli navzájom rôzne, rozklad E_n musí pozostávať aspoň z n tried, čo už musia byť jednoprvkové triedy. Pretože rozklad E_n už nie je možné zjemňovať, nutne musí platiť $E_n = E_{n+1} = E_{n+2} = \ldots$. Ukážeme, že ešte aj tento odhad ide o trochu zlepšiť.

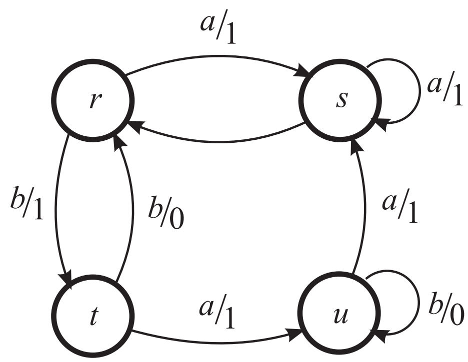
Predpokladajme, že E_1 pozostáva len z jednej triedy. Teda pre každé $s, t \in S$ a každé $x \in X$ je $\lambda(s,x) = \lambda(t,x)$. Uvažujme teraz o ľubovoľnom slove xy, ktoré má dĺžku 2. Pre každé $s, t \in S$ je

$$\widehat{\lambda}(s,xy) = \lambda(s,x)\lambda(\delta(s,x),y) = \lambda(t,x)\lambda(\delta(t,x),y) = \lambda(t,xy).$$

To znamená, že aj rozklad E_2 má iba jednu triedu, a tak, $E_1 = E_2 = E$. Z toho dostávame: Buď E_1 pozostáva iba z jednej triedy, a $E_1 = E$, alebo E_1 má aspoň dve triedy. Pre ekvivalenciu E_2 teraz dostávame: Buď $E_2 = E_1$, alebo E_2 má aspoň tri triedy. Podobne pre ekvivalenciu E_3 dostávame: Buď $E_3 = E_2$, alebo E_3 má aspoň štyri triedy. Tak postupujeme až po ekvivalenciu E_n, pre ktorú máme: Buď $E_n = E_{n-1}$, alebo E_n má aspoň $n+1$ tried. Pretože S má iba n stavov, nie je posledná možnosť reálna. Preto v každom automate, ktorý má n stavov, musí byť $E_{n-1} = E_n$. Tým sme dokázali nasledujúcu vetu.

VETA 3.7. Nech $A = (S, X, Z, \delta, \lambda)$ je automat, ktorého množina stavov má n prvkov. Potom sEt práve vtedy, keď pre každé vstupné slovo $w \in X^+$, ktoré má dĺžku $n-1$, je $\widehat{\lambda}(s, w) = \widehat{\lambda}(t, w)$. □

PRÍKLAD 3.20. Opíšeme ekvivalenciu E v automate, ktorý je daný grafom na obr. 30.



OBR. 30. Graf automatu z príkladu 3.20

Riešenie. Daný automat má štyri stavy. Preto v tomto príklade je $E = E_3$. Aby sme našli triedy ekvivalencie E_3, potrebujeme mať prehľad o všetkých výstupných slovách, ktoré sú reakciami na vstupné slová dĺžky 3. V tabuľke 8 uvádzame hodnoty $\widehat{\lambda}(q, w)$ pre každý stav $q \in \{r, s, t, u\}$ a pre slová w dĺžky 3.

TABUĽKA 8. Tabuľka výstupných slov dĺžky 3 z príkladu 3.20

	aaa	*aab*	*aba*	*baa*	*bba*	*bab*	*abb*	*bbb*
r	111	110	101	111	101	110	101	101
s	111	110	101	011	011	010	101	010
t	111	110	101	011	011	010	100	010
u	111	110	101	011	001	010	101	000

Zo stĺpca patriaceho k slovu *baa* vyplýva, že $r\not{E}s, r\not{E}t, r\not{E}u$. Zo stĺpca patriaceho k slovu *bba* vyplýva, že $s\not{E}u$, $t\not{E}u$. Stĺpec patriaci slovu *abb* implikuje $s\not{E}t$, takže $E = \{\overline{r}, \overline{s}, \overline{t}, \overline{u}\}$. To znamená, že daný automat je redukovaný a je zhodný so svojím automatom A_R. ■

Postup uvedený v príklade 3.20 by bol „trochu“ zdĺhavý už napríklad v prípade, keď množina stavov má 10 prvkov a vstupná abeceda tri písmená. V takomto prípade počet slov, ktoré majú dĺžku 9 je $3^9 = 19683$. To je počet stĺpcov v tabuľke, ktorá má 10 riadkov, čiže by bolo treba uvažovať o 196830 výstupných slovách.

Pri postupnom generovaní ekvivalencií E_k s výhodou využívame túto vetu, ktorá má veľký praktický význam pri riešení príkladov.

VETA 3.8. Nech $A = (S, X, Z, \delta, \lambda)$ je automat. Potom pre ľubovoľné $s, t \in S$ a každé $x \in X$ platí:

$$sE_{k+1}t \text{ práve vtedy, keď } sE_kt \text{ a súčasne } \delta(s, x)E_k\delta(t, x).$$

Dôkaz.

a) Z vety 3.4 a 3.5 vyplýva, že predpoklad $sE_{k+1}t$ implikuje sE_kt a $\delta(s,x)E_k\delta(t,x)$ pre každé $x \in X$

b) Nech sE_kt a súčasne $\delta(s,x)E_k\delta(t,x)$ pre každé $x \in X$. Pretože $E_1 \supset E_k$, je za tohto predpokladu sE_1t a súčasne $\delta(s,x)E_k\delta(t,x)$. Nech $w \in X^+$ ľubovoľné vstupné slovo dĺžky $k+1$. Toto slovo si môžeme napísať v tvare $w = xv$ kde $x \in X$ a $v \in X^+$ je slovo dĺžky k. Preto

$$\widehat{\lambda}(s,w) = \widehat{\lambda}(s,xv) = \lambda(s,x)\widehat{\lambda}(\delta(s,x),v) = \lambda(t,x)\widehat{\lambda}(\delta(t,x),v) = \widehat{\lambda}(t,xv) = \widehat{\lambda}(t,w).$$

Teda $sE_{k+}t$. Tým je veta 3.8 dokázaná. □

PRÍKLAD 3.21. Nech $A = (S, X, Z, \delta, \lambda)$ je automat, ktorý je daný pomocou tab. 9. Nájdeme redukovaný automat A_R tohto automatu.

TABUĽKA 9. Tabuľka automatu z príkladu 3.21

	δ		λ	
	x	y	x	y
1	2	1	0	1
2	1	3	0	0
3	3	4	0	1
4	4	4	0	1
5	2	2	0	0

Riešenie. Najprv nájdeme triedy ekvivalencie E. Triedy ekvivalencie E_1 môžeme popísať priamo z tabuľky daného automatu.

Vidíme, že $\lambda(1,x) = \lambda(3,x) = \lambda(4,x) = 0$, $\lambda(2,x) = \lambda(5,x) = 0$, $\lambda(1,y) = \lambda(3,y) = = \lambda(4,y) = 1$, $\lambda(2,y) = \lambda(5,y) = 0$. Preto $E_1 = \{\overline{1,3,4}, \overline{2,5}\}$.

Teraz opíšeme triedy ekvivalencie E_2. Využijeme podmienku $E_1 \supset E_2$. Preto musíme rozhodnúť len o týchto otázkach: 1) $1E_23$? 2) $1E_24$? 3) $3E_24$? 4) $2E_25$? Začneme prvou otázkou. Aby sme na túto otázku dali odpoveď, stačí vyšetriť, či: $\delta(1,x)E_1\delta(3,x)$ a súčasne $\delta(1,y)E_1\delta(3,y)$. Nemusíme totiž zisťovať, či $1E_13$, lebo všetky dvojice sme vybrali tak, aby práve túto podmienku spĺňali. Z tabuľky daného automatu máme: $\delta(1,x) = 2$ a $\delta(3,x) = 3$. $2\cancel{E_1}3$ preto $1\cancel{E_2}3$. Ďalej už budeme písať stručnejšie.

$1E_24$? Máme: $\delta(1,x) = 2$, $\delta(4,x) = 4$, $2\cancel{E_1}4$, preto $1\cancel{E_2}4$.
$3E_24$? Máme: $\delta(3,x) = 3$, $\delta(4,x) = 4$, $3E_14$, $\delta(3,y) = 4$, $\delta(4,y) = 4$, $4E_14$, preto $3E_24$.
$2E_25$? Máme: $\delta(2,x) = 1$, $\delta(5,x) = 2$, $1\cancel{E_1}2$, preto $2\cancel{E_2}5$.

Teda $E_2 = \{\overline{1}, \overline{3,4}, \overline{2}, \overline{5}\}$.

Aby sme našli triedy ekvivalencie E_3, stačí rozhodnúť iba o tom, či $3E_34$. Máme $\delta(3,x) = 3$, $\delta(4,x) = 4$, $3E_24$, $\delta(3,y) = 4$, $\delta(4,y) = 4$, $4E_24$, preto $3E_34$. Z toho vyplýva, že $E_3 = \{\overline{1}, \overline{3,4}, \overline{2}, \overline{5}\} = E_2 = E$. Označme triedy ekvivalencie E takto: $A_1 = \overline{1}$, $A_2 = \overline{2}$, $A_3 = \overline{3,4}$, $A_4 = \overline{5}$. Potom $A_R = (\{A_1, A_2, A_3, A_4\}, \{x, y\}, \{0, 1\}, \delta_R, \lambda_R)$, pričom tabuľka prechodovej funkcie δ_R a výstupnej funkcie λ_R je v tabuľke 10. ■

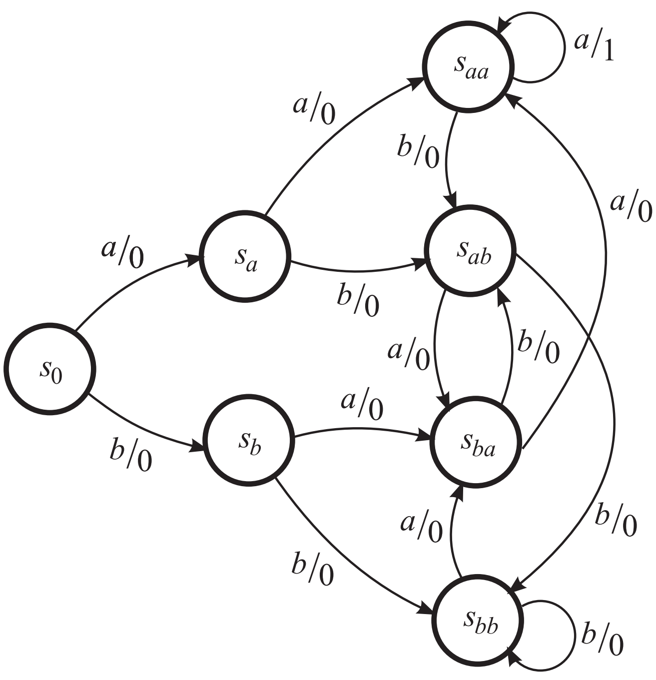
PRÍKLAD 3.22. Uvažujme o generátore postupnosti signálov, v ktorej sa vyskytujú iba dva signály a, b. Táto postupnosť prichádza na vstup počítadla, ktoré na výstupe reaguje hodnotou 1, ak sa na vstupe objavia tri signály a idúce za sebou. Vo zvyšných prípadoch je

TABUĽKA 10. Tabuľka redukovaného automatu z príkladu 3.21

	x	y	x	y
A_1	A_2	A_1	0	1
A_2	A_1	A_3	0	0
A_3	A_3	A_3	0	1
A_4	A_2	A_2	0	0

výstup 0 (ide o tzv. počítadlo blokov dĺžky 3). Teraz opíšeme toto počítadlo ako automat a k tomuto automatu nájdeme redukovaný automat.

Riešenie. Počítadlo budeme opisovať ako iniciálny automat so začiatočným stavom s_0. Stav s_a bude zodpovedať situácii, keď v stave s_0 sa na vstupe objavil signál a. Podobne stav s_b bude zodpovedať situácii, keď v stave s_0 sa na vstupe objavil signál b. Ďalej definujeme štyri stavy s_{aa},s_{ab}, s_{ba}, s_{bb}, pričom prvé dva zodpovedajú situácii, keď v stave s_a na vstup počítadla príde a, resp. b. Druhé dva stavy zodpovedajú situácii, keď v stave s_b na vstup počítadla príde a, resp. b. Ďalšia činnosť automatu je už zrejmá. Napríklad $\delta(s_{ab}, a) = s_{ba}$, $\delta(s_{ab}, b) = s_{bb}$. Je zrejmé, že $\lambda(s_{aa}, a) = 1$ a vo zvyšných prípadoch sa hodnoty výstupnej funkcie rovnajú 0. Graf tohto automatu uvádzame na obr. 31. Hodnoty prechodovej a výstupnej funkcie sú v tabuľke 11.



OBR. 31. Graf automatu z príkladu 3.22

Teraz nájdeme redukovaný automat automatu z tabuľky 11. Priamo z tabuľky čítame $E_1 = \{\overline{s_0, s_a, s_b, s_{ab}, s_{ba}, s_{bb}}, \overline{s_{aa}}\}$.

Aby sme získali triedy ekvivalencie E_2, postupne budeme testovať jednotlivé možnosti.

$s_0 E_2 s_a$? $\delta(s_0, a) = s_a$, $\delta(s_a, a) = s_{aa}$, $s_a \cancel{E_1} s_{aa}$, preto $s_0 \cancel{E_2} s_a$.

$s_0 E_2 s_b$? $\delta(s_0, a) = s_a$, $\delta(s_b, a) = s_{ba}$, $s_a E_1 s_{ba}$, $\delta(s_0, b) = s_b$, $\delta(s_b, b) = s_{bb}$, $s_b E_1 s_{bb}$, preto $s_0 E_2 s_b$.

$s_0E_2s_{ab}$? $\delta(s_0,a)=s_a$, $\delta(s_{ab},a)=s_{ba}$, $s_aE_1s_{ba}$, $\delta(s_0,b)=s_b$, $\delta(s_{ab},b)=s_{bb}$, $s_bE_1s_{bb}$, preto $s_0E_2s_{ab}$.

$s_0E_2s_{ba}$? $\delta(s_0,a)=s_a$, $\delta(s_{ba},a)=s_{aa}$, $s_a\not{E}_1s_{aa}$, preto $s_0\not{E}_2s_{ba}$.

$s_0E_2s_{bb}$? $\delta(s_0,a)=s_a$, $\delta(s_{bb},a)=s_{ba}$, $s_aE_1s_{ba}$, $\delta(s_0,b)=s_b$, $\delta(s_{bb},b)=s_{bb}$, $s_bE_1s_{bb}$, preto $s_0E_2s_{bb}$.

V jednej triede ekvivalencie E_2 ešte môžu byť stavy s_a, s_{ba}.

$s_aE_2s_{ba}$? $\delta(s_a,a)=s_{aa}$, $\delta(s_{ba},a)=s_{aa}$, $s_{aa}E_1s_{aa}$, $\delta(s_a,b)=s_{ab}$, $\delta(s_{ba},b)=s_{ab}$, $s_{ab}E_1s_{ab}$, preto $s_aE_2s_{ba}$.

Z doterajších výsledkov vyplýva, že $E_2=\{\overline{s_0,s_b,s_{ab},s_{bb}},\overline{s_a,s_{ba}},\overline{s_{aa}}\}$.

TABUĽKA 11. Tabuľka automatu z príkladu 3.22

	a	b	a	b
s_0	s_a	s_b	0	0
s_a	s_{aa}	s_{ab}	0	0
s_b	s_{ba}	s_{bb}	0	0
s_{aa}	s_{aa}	s_{ab}	1	0
s_{ab}	s_{ba}	s_{bb}	0	0
s_{ba}	s_{aa}	s_{ab}	0	0
s_{bb}	s_{ba}	s_{bb}	0	0

Teraz budeme hľadať triedy ekvivalencie E_3.

$s_0E_3s_b$? $\delta(s_0,a)=s_a$, $\delta(s_b,a)=s_{ba}$, $s_aE_2s_{ba}$, $\delta(s_0,b)=s_b$, $\delta(s_b,b)=s_{bb}$, $s_bE_2s_{bb}$, preto $s_0E_3s_b$.

$s_0E_3s_{ab}$? $\delta(s_0,a)=s_a$, $\delta(s_{ab},a)=s_{ba}$, $s_aE_2s_{ba}$, $\delta(s_0,b)=s_b$, $\delta(s_{ab},b)=s_{bb}$, $s_bE_2s_{bb}$, preto $s_0E_3s_{ab}$.

$s_0E_3s_{bb}$? $\delta(s_0,a)=s_a$, $\delta(s_{bb},a)=s_{ba}$, $s_aE_2s_{ba}$, $\delta(s_0,b)=s_b$, $\delta(s_{bb},b)=s_{bb}$, $s_bE_2s_{bb}$, preto $s_0E_3s_{bb}$.

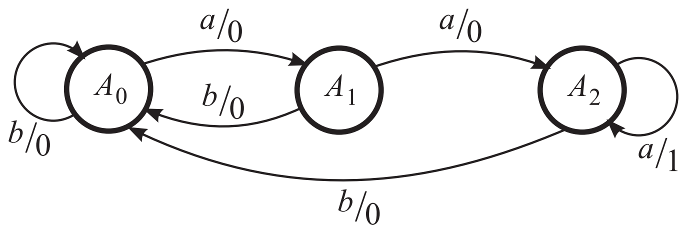
$s_aE_3s_{ba}$? $\delta(s_a,a)=s_{aa}$, $\delta(s_{ba},a)=s_{aa}$, $s_{aa}E_2s_{aa}$, $\delta(s_a,b)=s_{ab}$, $\delta(s_{ba},b)=s_{ab}$, $s_{ab}E_2s_{ab}$, preto $s_aE_3s_{ba}$.

Z tohto testu vyplýva, že $E_3=E_2=\{\overline{s_0,s_b,s_{ab},s_{bb}},\overline{s_a,s_{ba}},\overline{s_{aa}}\}=E$.
Označme $\overline{s_0,s_b,s_{ab},s_{bb}}=A_0$, $\overline{s_a,s_{ba}}=A_1$, $\overline{s_{aa}}=A_2$, potom
$A_R=(\{A_0,A_1,A_2\},\{a,b\},\{0,1\},\delta_R,\lambda_R)$. Hodnoty funkcií δ_R a λ_R uvádzame v tabuľke 12. Graf automatu A_R je nakreslený na obr. 32. Tento automat je tiež iniciálny automat so začiatočným stavom A_0. ■

TABUĽKA 12. Tabuľka redukovaného automatu z príkladu 3.22

	a	b	a	b
A_0	A_1	A_0	0	0
A_1	A_2	A_0	0	0
A_2	A_2	A_0	1	0

POZNÁMKA 3.4. Ak máme rozhodnúť o tom, či dva automaty sú ekvivalentné, môžeme uvažovať takto. Nech $A=(S,X,Z,\delta_A,\lambda_A)$ a $B=(T,X,Z,\delta_B,\lambda_B)$. Túto dvojicu



Obr. 32. Graf redukovaného automatu z príkladu 3.22

automatov môžeme považovať za jeden automat $A \cup B = (S \cup T, X, Z, \delta, \lambda)$, kde $\delta : (S \cup T) \times X \to S \cup T$ a $\lambda : (S \cup T) \times X \to Z$ sú definované tak, že

$$\delta(s,x) = \delta_A(s,x), \text{ ak } s \in S,$$
$$\delta(t,x) = \delta_B(t,x), \text{ ak } t \in T,$$
$$\lambda(s,x) = \lambda_A(s,x), \text{ ak } s \in S,$$
$$\lambda(t,x) = \lambda_B(t,x), \text{ ak } t \in T.$$

Potom v automate $A \cup B$ môžeme vyhľadať triedy ekvivalencie E. Ak v každej triede ekvivalencie sa nachádzajú aj prvky množiny S a aj prvky množiny T, automaty A a B sú ekvivalentné. Ak existuje trieda, v ktorej sú len prvky z jednej z uvedených množín, dané automaty nie sú ekvivalentné.

Príklad 3.23. Nech automat A je daný tabuľkou 13, časť a) a B je automat, ktorý je daný tabuľkou 13, časť b). Automat $A \cup B$ uvádzame v tabuľke 14.

Tabuľka 13. Automaty z príkladu 3.23

	x	x
s_1	s_2	0
s_2	s_2	0
s_3	s_1	1
s_4	s_4	1

časť a)

	x	x
q_1	q_1	0
q_2	q_1	1
q_3	q_3	1
q_4	q_3	1

časť b)

Tabuľka 14. Automat z príkladu 3.23

	x	x
s_1	s_2	0
s_2	s_2	0
s_3	s_1	1
s_4	s_4	1
q_1	q_1	0
q_2	q_1	1
q_3	q_3	1
q_4	q_3	1

Pre ekvivalencie E_k dostávame:

$$E_1 = \{\overline{s_1, s_2, q_1}, \overline{s_3, s_4, q_2, q_3, q_4}\}, E_2 = \{\overline{s_1, s_2, q_1}, \overline{s_3, q_2}, \overline{s_4, q_3, q_4}\} = E_3 = E.$$

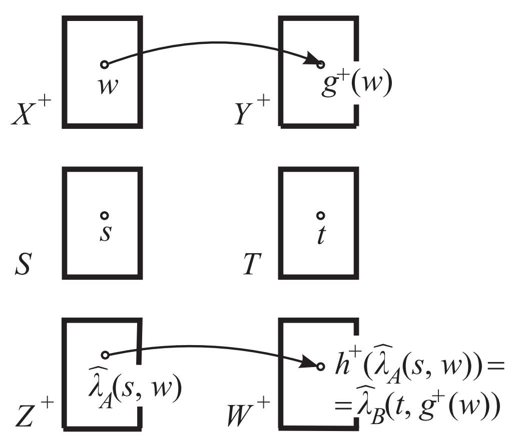
Pretože v každej triede ekvivalencie E sa nachádzajú stavy aj automatu A a aj automatu B, dané automaty sú ekvivalentné.

Poznámka 3.5. Definovať ekvivalenciu stavov a automatov je možné aj vo všeobecnejšom prípade. Najprv uvažujme takto: Nech $X = \{x_1, \ldots, x_n\}$ a $Y = \{y_1, \ldots, y_n\}$ sú dve abecedy s rovnakým počtom písmen. Nech $g : X \to Y$ je bijekcia. Uvažujme o zobrazení $g^+ : X^+ \to Y^+$, ktoré je definované predpisom: Pre každé $w = x^1 \ldots x^n$ je

$$g^+(w) = g^+(x^1 \ldots x^k) = g(x^1)g(x^2) \ldots g(x^k).$$

Nie je ťažké dokázať, že zobrazenie g^+ je tiež bijekcia.

Majme teraz dva automaty $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, Y, W, \delta_B, \lambda_B)$. Nech existujú bijekcie $g : X \to Y$ a $h : Z \to W$. Spolu s týmito bijekciami sú dané aj bijekcie $g^+ : X^+ \to Y^+$ a $h^+ : X^+ \to Y^+$, Potom budeme hovoriť, že stavy $s \in S$ a $t \in T$ sú ekvivalentné, ak pre každé $w \in X^+$ je $h^+(\widehat{\lambda}_A(s, w)) = \widehat{\lambda}_B(t, g^+(w))$ (pozri obr. 33). Túto definíciu ekvivalencie stavov môžeme využiť aj na zovšeobecnenie definície ekvivalencie automatov.



Obr. 33. Ilustrácia poznámky 3.5

6. Izomorfizmus automatov

V praxi sa často stretávame s prípadom, keď jeden automat vznikne z druhého preznačením (zakódovaním) stavov. V takom prípade budeme hovoriť, že tieto automaty sú izomorfné. Presnejšie sformulujeme tento pojem v tejto definícii.

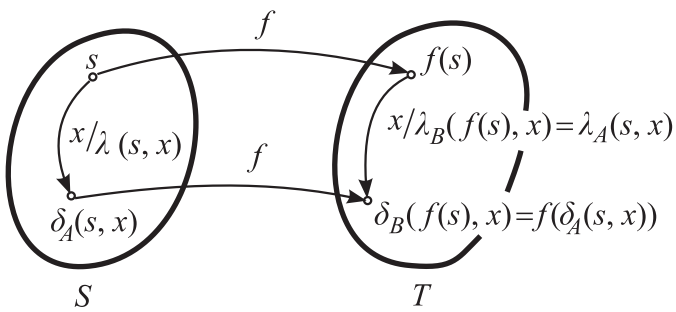
Definícia 3.16. Nech $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$ sú automaty. Nech existuje bijekcia $f : S \to T$ taká, že:

a) pre každé $s \in S$ a každé $x \in X$ sa $f(\delta_A(s, x)) = \delta_B(f(s), x)$,
b) pre každé $s \in S$ a každé $x \in X$ sa $\lambda_A(s, x) = \lambda_B(f(s), x)$.

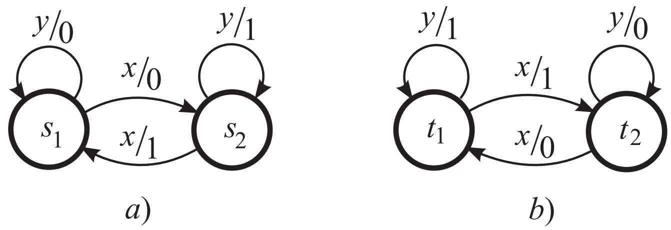
Potom budeme hovoriť, že ***automaty*** A a B sú ***izomorfné*** (pozri obr. 34).

Príklad 3.24. Nech automat A je daný grafom na obr. 35, časť a) a B je automat, ktorý je daný grafom, na obr. 35 časť b). Ukážeme, že tieto automaty sú izomorfné.

Riešenie. Množina stavov automatu A je $S = \{s_1, s_2\}$. Množina stavov automatu B



Obr. 34. Ilustrácia definície 3.16



Obr. 35. Ilustrácia automatov z príkladu 3.24

je $T = \{t_1, t_2\}$. Je zrejmé, že funkcia $f : S \to T$, ktorá je daná funkčnými hodnotami $f(s_1) = t_2$ a $f(s_2) = t_1$ je bijekcia. Ďalej dostávame:

$$\begin{aligned}
&f(\delta_A(s_1, x)) = f(s_2) = t_1 = \delta_B(t_2, x) = \delta_B(f(s_1), x),\\
&f(\delta_A(s_1, y)) = f(s_1) = t_2 = \delta_B(t_2, y) = \delta_B(f(s_1), y),\\
&f(\delta_A(s_2, x)) = f(s_1) = t_2 = \delta_B(t_1, x) = \delta_B(f(s_2), x),\\
&f(\delta_A(s_2, y)) = f(s_2) = t_1 = \delta_B(t_1, y) = \delta_B(f(s_2), y),\\
&\lambda_A(s_1, x) = 0 = \lambda_B(t_2, x) = \lambda_B(f(s_1), x),\\
&\lambda_A(s_1, y) = 0 = \lambda_B(t_2, y) = \lambda_B(f(s_1), y),\\
&\lambda_A(s_2, x) = 1 = \lambda_B(t_1, x) = \lambda_B(f(s_2), x),\\
&\lambda_A(s_2, y) = 1 = \lambda_B(t_1, y) = \lambda_B(f(s_2), y).
\end{aligned}$$

Z uvedených rovností vyplýva, že dané automaty sú izomorfné. Vidíme, že formálne overenie izomorfizmu je aj v tých najjednoduchších prípadoch dosť zdĺhavé. Preto izomorfizmus overujeme viac-menej vizuálnym spôsobom, keď z grafu dokážeme určiť bijekciu f, ktorá spĺňa podmienky izomorfizmu. ■

Keďže izomorfné automaty vykazujú vysoký stupeň príbuznosti, dá sa čakať, že tieto automaty budú aj ekvivalentné. Dôkaz tejto vety nie je až tak zrejmý.

Veta 3.9. Dva izomorfné automaty sú ekvivalentné.

Dôkaz. Nech $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$ sú izomorfné automaty, pričom $f : S \to \mathrm{T}$ je bijekcia, pomocou ktorej je tento izomorfizmus definovaný. Teraz dokážeme, že pre každé $s \in S$ sú stavy s a $f(s)$ ekvivalentné. Tým bude dokázaná ekvivalencia automatov A a B. Dôkaz urobíme indukciou vzhľadom na dĺžku slova $w \in X^+$. Na zjednodušenie použijeme označenie $sE_kf(s)$ na označenie skutočnosti, že stavy s a $f(s)$ dávajú rovnaké výstupné slovo na každé vstupné slovo dĺžky k.

1. Priamo z definície izomorfizmu vyplýva, že pre každé $s \in S$ je $sE_1f(s)$, lebo pre každé $x \in X$ platí: $\lambda_A(s, x) = \lambda_B(f(s), x)$.

2. Predpokladajme, že pre každé $s \in S$ je $sE_kf(s)$. Dokážeme, že za tohto predpokladu je aj $sE_{k+1}f(s)$. Nech $w \in X^+$ je ľubovoľné slovo dĺžky $k+1$. Preto ho môžeme napísať v tvare $w = xv$, kde $x \in X$ a v je slovo dĺžky k. Potom

$$\widehat{\lambda}_A(s,w) = \widehat{\lambda}_A(s,xv) = \lambda_A(s,x)\widehat{\lambda}_A(\delta_A(s,x),v).$$

Stavy $\delta_A(s,x)$ a $f(\delta_A(s,x))$ sú podľa indukčného predpokladu „E_k-ekvivalentné". Preto pre slovo v, ktoré má dĺžku k, platí:

$$\widehat{\lambda}_A(\delta_A(s,x),v) = \widehat{\lambda}_B(f(\delta_A(s,x)),v).$$

Teraz už máme:

$$\begin{aligned}\widehat{\lambda}_A(s,w) &= \lambda_A(s,x)\widehat{\lambda}_A(\delta_A(s,x),v) = \lambda_B(f(s),x)\widehat{\lambda}_B(f(\delta_A(s,x)),v) = \\ &= \lambda_B(f(s),x)\widehat{\lambda}_B(\delta_B(f(s),x),v) = \widehat{\lambda}_B(f(s),xv) = \widehat{\lambda}_B(f(s),w).\end{aligned}$$

Tým sme dokázali, že pre každé $k = 1, 2, \ldots$ a pre každé $s \in S$ je $sE_kf(s)$. Z toho už vyplýva, že pre každé $s \in S$ je stav s ekvivalentný so stavom $f(s)$. Tým je veta 3.9 dokázaná. □

Teraz sformulujeme hlavnú vetu tejto časti, ktorá ukazuje vzťah izomorfizmu a ekvivalencie automatov.

VETA 3.10. Dva konečné automaty $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$ sú ekvivalentné práve vtedy, keď ich redukované automaty A_R a B_R sú izomorfné.

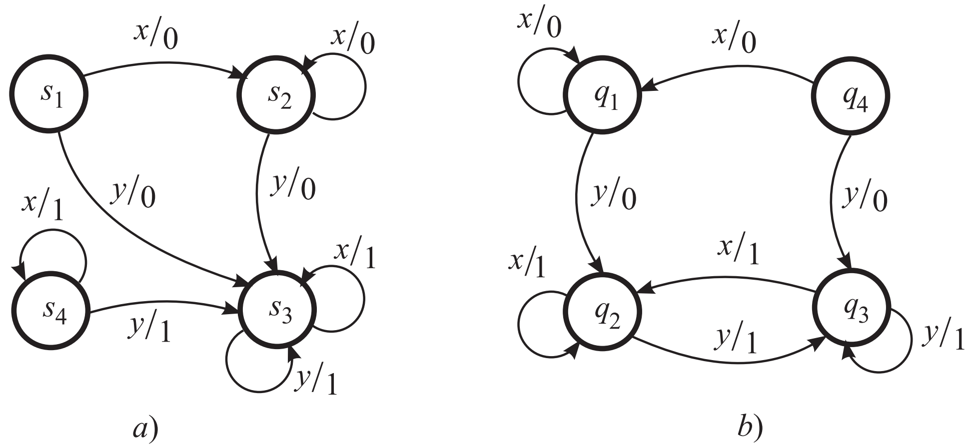
Dôkaz.

1. Nech automaty $A_R = (S_R, X, Z, \delta_{A_R}, \lambda_{A_R})$ a $B_R = (T_R, X, Z, \delta_{B_R}, \lambda_{B_R})$ sú izomorfné. Množina stavov S_R automatu A_R je množina tried ekvivalencie E_A na množine stavov S. Nech $S_R = \{A_1, \ldots, A_n\}$. Pretože automaty A_R a B_R sú izomorfné, musí existovať bijekcia, $f : S_R \to T_R$, pomocou ktorej je tento izomorfizmus definovaný. Z toho vyplýva, že $T_R = \{f(A_1), \ldots, f(A_n)\}$ je rozklad množiny T. Preto pre každé $t \in T$ existuje $f(A_1)$ také, že $t \in f(A_i)$. To znamená, že pre každé $s \in A_i$ a pre každé $t \in f(A_i)$ platí $s \sim A_i \sim f(A_i) \sim t$. Z toho už vyplýva, že automaty A a B sú ekvivalentné.

2. Nech automaty $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$ sú ekvivalentné. Nech $S_R = \{A_1, \ldots, A_n\}$ je rozklad množiny S, ktorý je indukovaný ekvivalenciou E_A a $T_R = \{B_1, \ldots, B_n\}$ je rozklad množiny T, ktorý je indukovaný ekvivalenciou E_B. Vieme, že tieto rozklady musia mať rovnaký počet tried a že existuje bijekcia $f : S_R \to T_R$, pomocou ktorej je každej triede A_i, priradená trieda $f(A_i)$ navzájom ekvivalentných stavov. Triedy v množine T_R očíslujeme tak, aby $f(A_i) = B_i$. pre $i = 1, 2, \ldots$. Potom v automatoch $A_R = (S_R, X, Z, \delta_{A_R}, \lambda_{A_R})$ a $B_R = (T_R, X, Z, \delta_{B_R}, \lambda_{B_R})$ sú stavy A_i a B_i ekvivalentné, a pretože automaty A_R a B_R sú redukované, sú to jediné možné dvojice ekvivalentných stavov. Z vety 1.34 teraz dostávame, že aj stavy $\delta_{A_R}(A_i, x)$ a $\delta_{B_R}(B_i, x)$ sú ekvivalentné pre každé $x \in X$. Z toho už vyplýva, že, $f(\delta_{A_R}(A_i, x)) = \delta_{B_R}(B_i, x) = \delta_{B_R}(f(A_i), x)$. Tým je dokázaná prvá rovnosť izomorfizmu automatov A_R a B_R. Druhá rovnosť je už jednoduchá. Pretože stavy A_i a $f(A_i)$ sú ekvivalentné, pre každé $x \in X$ musí platiť rovnosť $\lambda_{A_R}(A_i, x) = \lambda_{B_R}(f(A_i), x)$ Teda bijekcia $f : S_R \to T_R$ definuje izomorfizmus medzi automatmi A_R a B_R. Tým je veta 1.50 dokázaná. □

PRÍKLAD 3.25. Nech $A = (\{s_1, s_2, s_3, s_4\}, \{x, y\}, \{0, 1\}, \delta_A, \lambda_A)$ a $B = (\{q_1, q_2, q_3, q_4\}, \{x, y\}, \{0, 1\}, \delta_B, \lambda_B)$ sú automaty, ktoré sú dané grafmi na obr. 36 časť a), b). Ukážeme, že tieto automaty sú ekvivalentné ale nie izomorfné.

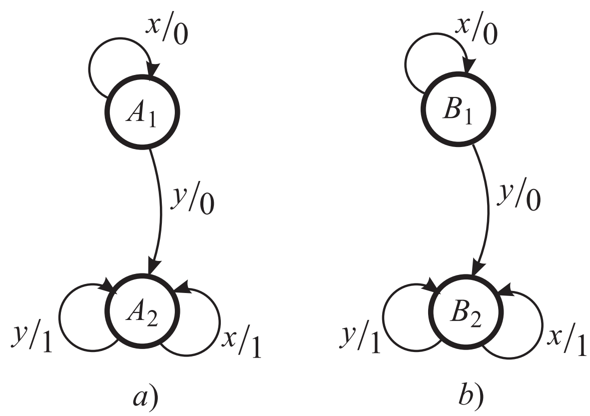
Riešenie. Najprv budeme dokazovať ekvivalenciu. Preto zostrojíme redukované automaty A_R a B_R.



OBR. 36. Grafy automatov z príkladu 3.25

Pre ekvivalenciu E_A na množine stavov automatu A dostávame $E_1 = \{\overline{s_1, s_2}, \overline{s_3, s_4}\} = E_2 = E_A$. Ak označíme $\overline{s_1, s_2} = A_1$, a $\overline{s_3, s_4} = A_2$, automat A_R je daný grafom na obr. 37, časť a).

Pre ekvivalenciu E_B na množine stavov automatu B dostávame $E_1 = \{\overline{q_1, q_4}, \overline{q_2, q_3}\} = E_2 = E_B$. Ak označíme $\overline{q_1, q_4} = B_1$, a $\overline{q_2, q_3} = B_2$, automat B_R je daný grafom na obr. 37, časť b).



OBR. 37. Grafy redukovaných automatov z príkladu 3.25

Je zrejmé, že automaty A_R a B_R sú izomorfné. Stačí zvoliť funkciu $f : \{A_1, A_2\} \to \{B_1, B_2\}$ tak, aby $f(A_1) = B_1$, $f(A_2) = B_2$. Je zrejmé, že ide o bijekciu, pomocou ktorej je uvedený izomorfizmus definovaný. Teda A a B sú ekvivalentné automaty.

Teraz ukážeme, že pôvodné automaty, aj keď majú rovnaký počet stavov a sú ekvivalentné, ešte nemusia byť izomorfné. Aby sme ukázali, že tieto automaty nie sú izomorfné, musíme vyšetriť všetky bijekcie $f : \{s_1, s_2, s_3, s_4\} \to \{q_1, q_2, q_3, q_4\}$ a ukázať, že ani jedna z nich nespĺňa podmienky, ktoré sú kladené na izomorfizmus automatov. V prvej časti ľahko vylúčime bijekcie, ktoré nespĺňajú podmienku $\lambda_A(s, x) = \lambda_B(f(s), x)$.

Uvažujme teda o bijekciách, ktoré podmienky pre výstupné funkcie izomorfných automatov spĺňajú. Nech f je jedna z nich. Potom $f(s_1)$ musí byť buď q_1 alebo q_4, lebo iba tieto dva stavy v automate B dávajú rovnaké výstupné hodnoty na vstupné písmená x, y ako dáva stav s_1.

a) Uvažujme teda o možnosti $f(s_1) = q_1$. Potom musí byť nutne $f(s_2) = q_4$, lebo stavy q_2, q_3, majú výstupy rôzne od stavu s_2. Pod podmienkou $f(s_1) = q_1$ a $f(s_2) = q_4$

poďme teraz overovať splnenie prvej podmienky pre izomorfizmus automatov a pritom sa pokúsime definovať zvyšné funkčné hodnoty. Počítajme: $f(\delta_A(s_1, x)) = f(s_2) = q_4$, $\delta_B(f(s_1), x) = \delta_B(q_1, x) = q_1$. Pretože $q_1 \neq q_4$, bijekcia f nespĺňa podmienky izomorfizmu automatov A a B. Musíme uvažovať o ďalších možnostiach. Tá však už je len jediná.

b) Ešte môžeme zvoliť $f(s_1) = q_4$, $f(s_2) = q_1$. Potom $f(\delta_A(s_1, x)) = f(s_2) = q_1$, $\delta_B(f(s_1), x) = \delta_B(q_4), x) = q_1$, $f(\delta_A(s_1, y)) = f(s_3)$, $\delta_B(f(s_1), y)) = \delta_B(q_4, y) = q_3$. Z toho vyplýva, že nutne musí byť $f(s_3) = q_3$, a teda $f(s_4) = q_2$. Inú možnosť pre voľbu bijekcie, ktorá spĺňa podmienky izomorfizmu automatov, už nemáme. Hodnoty bijekcie f boli volené tak, aby podmienky pre prechodovú funkciu boli splnené v stave s_1. Teraz musíme overiť, či sú splnené aj pre zvyšné stavy. Budeme ich overovať pre stav s_2:

$$f(\delta_A(s_2, x)) = f(s_2) = q_1, \delta_B(f(s_2), x) = \delta_B(q_1, x) = q_1,$$
$$f(\delta_A(s_2, y)) = f(s_3) = q_3,\ \delta_B(f(s_2), y) = \delta_B(q_1, y) = q_2.$$

Pretože $q_2 \neq q_3$, daná bijekcia nespĺňa podmienky izomorfizmu automatov A a B. Pretože inej možnosti už niet, dané automaty nie sú izomorfné. ∎

Poznámka 3.6. Tak ako v prípade ekvivalencie automatov môžeme aj pri izomorfizme automatov upustiť od požiadavky, aby automaty A a B mali spoločné vstupné a spoločné výstupné množiny. Túto podmienku treba nahradiť podmienkou existencie bijekcie medzi vstupnými a bijekcie medzi výstupnými množinami. Potom definíciu izomorfných automatov môžeme sformulovať takto:

Nech $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, Y, W, \delta_B, \lambda_B)$ sú automaty. Nech existujú také bijekcie $f : S \to T$, $g : X \to Y$, $h : Z \to W$, že pre každé $s \in S$ a každé $x \in X$ je:

a) $f(\delta_A(s, x)) = \delta_B(f(s), g(x))$,
b) $h(\lambda_A(s, x)) = \lambda_B(f(s), g(x))$.

Potom hovoríme, že automaty A a B sú izomorfné.

V tomto prípade okrem kódovania stavov sme kódovali aj vstupnú a aj výstupnú abecedu.

Dá sa dokázať, že pri takto chápanom izomorfizme automatov a pri zovšeobecnenej definícii ekvivalencie automatov, ktorú sme uviedli v poznámke 3.4, veta 3.10 platí v tom istom znení.

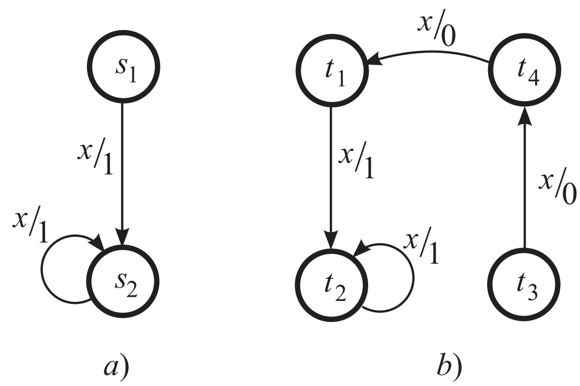
7. Pokrytie automatu automatom

Nech A je automat. V tejto časti sa budeme zaoberať automatom B, ktorý má tú vlastnosť, že pre každé vstupné slovo w dokáže generovať takú istú postupnosť (riadiacich) výstupných signálov ako automat A. Pritom automat B môže generovať aj výstupné postupnosti, ktoré automat A nebude schopný vydávať. V takom prípade budeme hovoriť, že automat B pokrýva automat A.

Definícia 3.17. Nech $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$ sú automaty. Ak existuje také zobrazenie $\varphi : S \to T$, že pre každé $w \in X^+$ je $\widehat{\lambda}_A(s, w) = \widehat{\lambda}_B(\varphi(s), w)$, budeme hovoriť, že ***automat*** B ***pokrýva automat*** A. Pritom budeme písať $B >> A$.

Príklad 3.26. Nech automaty $A = \{\{s_1, s_2\}, \{x\}, \{0, 1\}, \delta_A, \lambda_A\}$ a $B = \{\{t_1, t_2, t_3, t_4\}, \{x\}, \{0, 1\}, \delta_B, \lambda_B\}$ sú dané pomocou grafu na obr. 38, časť a), b). Nech $S = \{s_1, s_2\}$, $T = \{t_1, t_2, t_3, t_4\}$. V tomto príklade sa dá ľahko dokázať, že automat B pokrýva automat A. Pre voľbu funkcie $\varphi : S \to T$ sa nám v tomto prípade totiž núka aj viacero možností. Tieto možnosti sú takéto:

1. $\varphi(s_1) = \varphi(s_2) = t_1$,



Obr. 38. Grafy automatov z príkladu 3.26

2. $\varphi(s_1) = \varphi(s_2) = t_2$,
3. $\varphi(s_1) = t_1, \varphi(s_2) = t_2$,
4. $\varphi(s_1) = t_2, \varphi(s_2) = t_1$. ■

Veta 3.11. Automat $A = (S, X, Z, \delta, \lambda)$ pokrýva ľubovoľný svoj podautomat $A' = (S', X, Z, \delta', \lambda')$.

Dôkaz. Za funkciu $\varphi : S' \to S$ stačí voliť identické zobrazenie, ktoré každému $s \in S'$ priradí to isté $s \in S$. □

Je zrejmé, že automat B pokrýva automat A práve vtedy, keď pre každý stav $s \in S$ existuje $t \in T$, ktoré je ekvivalentné so stavom s. Toto tvrdenie nemusí platiť v opačnom smere. Z toho vyplýva, že v prípade ekvivalencie automatov A a B bude $A >> B$ a $B >> A$. Dokonca platí ešte silnejšie tvrdenie.

Veta 3.12. Dva automaty $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$ sú ekvivalentné práve vtedy, keď $A >> B$ a súčasne $B >> A$.

Dôkaz. Ak $A >> B$ a $B >> A$, priradenie ekvivalentných stavov sa deje dokonca pomocou zobrazení. Preto sú tieto automaty ekvivalentné.

Ak $A \sim B$, pre každé $s \in S$ existuje dokonca celá množina prvkov $t \in T$, ktoré sú s prvkom s ekvivalentné. Je to jedna trieda ekvivalencie E_B na množine T. Z každej takejto triedy vyberieme jedno t a priradíme ho ku stavu s. Tak dostaneme zobrazenie $\varphi : S \to T$, ktoré ku každému $s \in S$ priradí $\varphi(s) \sim s$. Teda $B >> A$. Podobne sa dokáže, že aj $A >> B$. □

Dôsledok 3.1.

1. Ak A a B sú izomorfné automaty, potom $A >> B$ a $B >> A$.
2. Ak A_R je redukovaný automat automatu A, potom $A_R >> A$ a $A >> A_R$. □

Ak si pozorne prezrieme obr. 36, časť b) a obr. 37, časť b), vidíme, že v automate B z príkladu 3.25 neexistuje podautomat, ktorý by bol izomorfný s redukovaným automatom B_R. To znamená, ak automat B pokrýva automat A, v automate B nemusí existovať podautomat, ktorý je izomorfný s automatom A. Dokážeme však dôležitú vetu, z ktorej vyplýva, že v automate B musí existovať podautomat, ktorý je s automatom A ekvivalentný. Pri dôkaze využijeme túto definíciu.

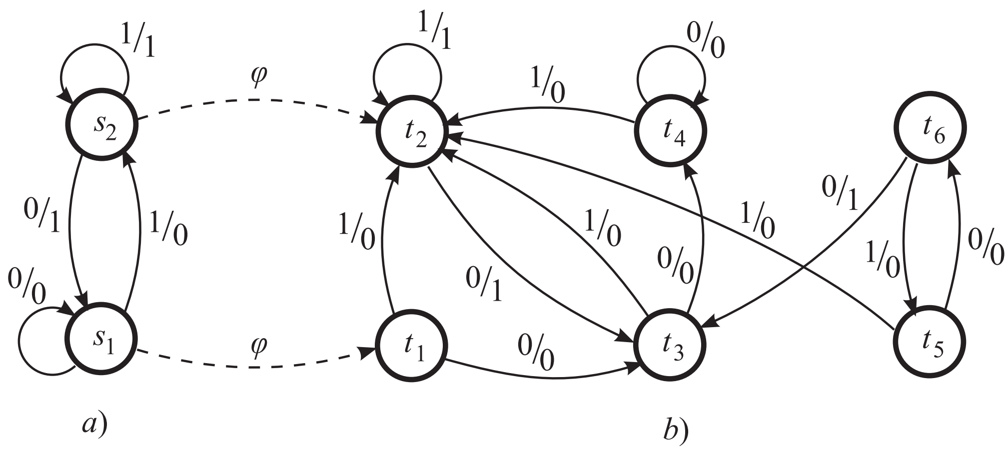
Nech $A = (S, X, Z, \delta, \lambda)$ je automat a $S' \subset S$. Nech T' je množina všetkých stavov automatu A, ktoré sú dosiahnuteľné z niektorého stavu množiny S'. Teda $T = S' \cup \{s \in S;$ existuje $s' \in S'$ a $w \in X^+$ tak, že $s = \widehat{\delta}(s', w)\}$. Potom je zrejmé, že pre každé $s \in T$ a $x \in X$ je $\delta(s, x) \in T$. Preto funkcie $\delta' : T \times X \to T$, $\delta'(s, x) = \delta(s, x)$ a $\lambda' : T \times X \to Z$,

$\lambda'(s,x) = \lambda(s,x)$ sú zúženia funkcií δ a λ. Z toho už vyplýva, že $A' = (T, X, Z, \delta', \lambda')$ je podautomat automatu A.

Definícia 3.18. Automat A' nazývame ***podautomat automatu*** A, ktorý je ***generovaný množinou stavov*** S'.

Je zrejmé, že pre každú neprázdnu podmnožinu $S' \subset S$ existuje podautomat, ktorý je touto množinou generovaný.

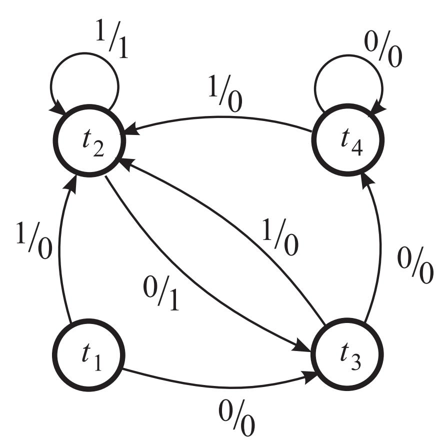
Príklad 3.27. Nech $A = (\{s_1, s_2\}, \{0,1\}, \{0,1\}, \delta_A, \lambda_A)$ a $B = (\{t_1, t_2, t_3, t_4, t_5, t_6\}, \{0,1\}, \{0,1\}, \delta_B, \lambda_B)$ sú automaty, ktoré sú dané grafmi na obr. 39, časť a), b).



Obr. 39. Grafy automatov z príkladu 3.27

Nech $S = \{s_1, s_2\}$ a $T = \{t_1, t_2, t_3, t_4, t_5, t_6\}$. Nech $\varphi : S \to T$ je také zobrazenie, že $\varphi(s_1) = t_1$, $\varphi(s_2) = t_2$. Označme $\varphi(S) = \{\varphi(s_1), \varphi(s_2)\} = \{t_1, t_2\} = T' \subset T$. Teraz opíšeme podautomat automatu B, ktorý je generovaný množinou T'.

Riešenie. Množina stavov tohto podautomatu $R = T' \cup \{t \in T;$ existuje $w \in X^+$ a $t' \in T'$ také, že $\delta_B(t', w) = t\} = \{t_1, t_2, t_3, t_4\}$. Vidíme, že stavy t_5 a t_6 nie sú zo stavov t_1 a t_2 dosiahnuteľné. Potom podautomat B', ktorý je generovaný množinou T, má graf na obr. 40.



Obr. 40. Ilustrácia príkladu 3.27

Na ďalšie účely bude pre nás výhodná ešte táto veta.

VETA 3.13. Nech $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, X, Z, \delta_B, \lambda_B)$. Nech stav $s \in S$ je ekvivalentný so stavom $t \in T$. Potom pre každé $w \in X^+$ je $\widehat{\delta}_A(s, w) \sim \widehat{\delta}_A(t, w)$.

Dôkaz. Nech $w \in X^+$ a $v \in X^+$ sú ľubovoľné slová. Potom aj $wv \in X^+$. Z predpokladu $s \sim t$ vyplýva, že $\widehat{\lambda}_A(s, wv) = \widehat{\lambda}_B(t, wv)$. Z toho už dostávame

$$\widehat{\lambda}_A(s, wv)\widehat{\lambda}_A(\widehat{\delta}_A(s, w), v) = \widehat{\lambda}_B(t, wv)\widehat{\lambda}_B(\widehat{\delta}_B(t, w), v).$$

Pretože ide o rovnosť dvoch slov zo Z^+, musí byť

$$\widehat{\lambda}_A(s, w) = \widehat{\lambda}_B(t, w) \text{ a } \widehat{\lambda}_A(\widehat{\delta}_A(s, w), v) = \widehat{\lambda}_B(\widehat{\delta}_B(t, w), v).$$

Z druhej rovnosti už dostávame požadovanú ekvivalenciu. □

Teraz už môžeme vysloviť a dokázať hlavnú vetu tejto časti.

VETA 3.14. Nech $A = (S, X, Z, \delta_A, \lambda_A)$ je redukovaný automat a automat $B = (T, X, Z, \delta_B, \lambda_B)$ pokrýva automat A. Potom automat B obsahuje podautomat B', ktorý je ekvivalentný s automatom A.

Dôkaz.

1. Nech teda $B = (S, X, Z, \delta_B, \lambda_B)$ pokrýva automat $A = (S, X, Z, \delta_A, \lambda_A)$. To znamená, že existuje zobrazenie $\varphi : S \to T$, ktoré pre každé $s \in S$ a každé $w \in X^+$ dáva rovnosť $\widehat{\lambda}_A(s, w) = \widehat{\lambda}_B(\varphi(s), w)$. Teraz ukážeme, že za podmienky redukovateľnosti automatu A je zobrazenie φ injekciou.

Predpokladajme, že $\varphi(s_1) = \varphi(s_2)$. Ale $s_1 \sim \varphi(s_1)$ a $s_2 \sim \varphi(s_2) = \varphi(s_1)$. Preto $s_1 \sim s_2$. Teraz už z redukovanosti automatu A vyplýva, že $s_1 = s_2$. Preto zobrazenie $\varphi : S \to T$ je injekciou.

2. Uvažujme o množine $\varphi(S) = \{\varphi(s) \in T; s \in S\}$. Označme túto množinu T'. Nech $B' = (R, X, Z, \delta'_B, \lambda'_B)$ je podautomat automatu B, ktorý je generovaný množinou T' (R je množina všetkých stavov, automatu B, ktoré sú dosiahnuteľné z niektorého stavu množiny T').

3. Pretože $\varphi(S) \subset R$ a pre každé $t \in R$ a každé $w \in X^+$ je $\widehat{\lambda}_{B'}(t, w) = \widehat{\lambda}_B(t, w)$, je zrejmé že automat B' pokrýva automat A.

4. Teraz dokážeme, že aj naopak automat A pokrýva automat B'. Definujeme funkciu $\psi : R \to S$ takto:

   a) $\psi(\varphi(s)) = s$ pre každé $s \in S$,

   b) $\psi(\widehat{\delta}_{B'}(\varphi(s), w)) = \widehat{\delta}(s, w)$ pre každé $s \in S$ a každé $w \in X^+$.

Treba ešte dokázať, že funkcia ψ je dobre definovaná. Je totiž dobre možné si predstaviť, že $\widehat{\delta}_{B'}(\varphi(s_i), w) = \widehat{\delta}_{B'}(\varphi(s_j), v)$ a pritom $\widehat{\delta}_A(s_i, w) \neq \widehat{\delta}_A(s_j, v)$. Ale $\varphi(s_i) \sim s_i$ a $\varphi(s_j) \sim s_j$. Preto je $\widehat{\delta}_A(s_i, w) \sim \widehat{\delta}_{B'}(\varphi(s_i), w) = \widehat{\delta}_{B'}(\varphi(s_j), v) \sim \widehat{\delta}_A(s_j, v)$, teda $\widehat{\delta}_A(s_i, w) \sim \widehat{\delta}_A(s_j, v)$. Z redukovanosti automatu A už dostávame $\widehat{\delta}_A(s_i, w) = \widehat{\delta}_A(s_j, v)$. To znamená, že predpoklad $\widehat{\delta}_{B'}(\varphi(s_i), w) = \widehat{\delta}_{B'}(\varphi(s_j), v)$ implikuje $\widehat{\delta}_A(s_i, w) = \widehat{\delta}_A(s_j, v)$ a teda zobrazenie ψ je dobre definované.

5. Stavy $\varphi(s)$ a s sú ekvivalentné. Preto aj $\widehat{\delta}_{B'}(\varphi(s), w)$ a $\widehat{\delta}_A(s, w)$ sú ekvivalentné stavy. To znamená, že funkcia ψ každému stavu priraďuje ekvivalentný stav. Z toho už vyplýva, že automat A pokrýva automat B'. Teda A a B' sú ekvivalentné automaty. Tým je veta 3.14 dokázaná. □

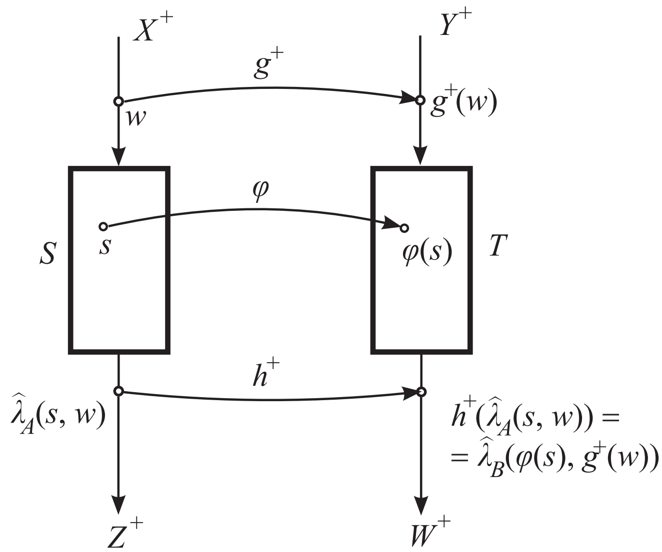
DÔSLEDOK 3.2.

1. Vetu 3.14 sme dokázali za predpokladu redukovanosti automatu A. V prípade, že A nie je redukovaný automat a $B >> A$, potom $B >> A_R$ (lebo $A >> A_R$) a v automate B sa nachádza podautomat B', ktorý je ekvivalentný s automatom

A_R. Potom máme $B' \sim A_R \sim A$. Teda B obsahuje podautomat B', ktorý je ekvivalentný s automatom A.

2. Ak k danému automatu A nájdeme automat B, ktorý ho pokrýva, potom máme automat, ktorý dokáže imitovať činnosť automatu A pomocou ekvivalentného automatu B'. Pochopiteľne, B dokáže okrem toho aj viac. □

Poznámka 3.7. Pojem pokrytia automatu automatom môžeme definovať aj v prípade automatov $A = (S, X, Z, \delta_A, \lambda_A)$ a $B = (T, Y, W, \delta_B, \lambda_B)$. V tomto prípade budeme žiadať aby existovali injekcie $g : X \to Y$ a $h : Z \to W$. Každú z týchto injekcií možno rozšíriť na injekciu $g^+ : X^+ \to Y^+$, resp. $h^+ : Z^+ \to W^+$, tak že pre každé $w = x^1 \dots x^n$, resp. $v = z^1 \dots z^n$ položíme $g^+(w) = g^+(x^1 \dots x^n) = g(x^1) \dots g(x^n)$, resp. $h^+(v) = h^+(z^1 \dots z^n) = h(z^1) \dots h(z^n)$. Potom budeme hovoriť, že automat $A = (S, X, Z, \delta_A, \lambda_A)$ pokrýva automat $B = (T, Y, W, \delta_B, \lambda_B)$ práve vtedy, ak existujú injekcie $g : X \to Y$, $h : Z \to W$ a funkcia $\varphi : S \to T$ také, že pre každé $s \in S$ a každé $w \in X^+$ je $h^+(\widehat{\delta}_A(s, w)) = \widehat{\delta}_B(\varphi(s), g^+(w))$ (pozri obr. 41).



Obr. 41. Ilustrácia definície pokrytia automatov

Pri takto definovanom pokrytí automatov, ak použijeme zovšeobecnenú definíciu ekvivalencie automatov, sa dá veta 3.14 dokázať v nezmenenom znení.

KAPITOLA 4

Fyzikálna realizácia automatov

1. Dvojkové automaty

V tejto časti sa budeme zaoberať automatmi, ktorých vstupná abeceda, množina stavov a výstupná abeceda sa skladajú z vektorov, ktorých zložky môžu nadobúdať iba dve hodnoty 0 a 1. Nech teda $\mathbf{B} = \{0,1\}$. Potom ***dvojkový Mealyho automat*** je pätica $A = (S, X, Z, \delta, \lambda)$, kde $S = \mathbf{B}^k, X = \mathbf{B}^n, Z = \mathbf{B}^m, \delta : S \times X \to S, \lambda : S \times X \to Z$. ***Dvojkový Moorov automat*** definujeme analogicky, teda ako päticu $A = (S, X, Z, \delta, \mu)$, kde $S = \mathbf{B}^k, X = \mathbf{B}^n, Z = \mathbf{B}^m, \delta : S \times X \to S, \mu : S \to Z$.

Príklad 4.1. Nech dvojkový automat $A = (\mathbf{B}^2, \mathbf{B}, \mathbf{B}, \delta, \lambda) = (\{(0,0),(0,1),(1,0),(1,1)\}, \{0,1\}, \{0,1\}, \delta, \lambda\}$ je daný pomocou tabuľky 4.1.

Tabuľka 1. Automat z príkladu 4.1

	0	1	0	1
(0,0)	(0,0)	(0,1)	0	1
(1,0)	(1,1)	(1,0)	1	0
(1,1)	(1,1)	(0,1)	1	1
(0,1)	(1,0)	(1,0)	1	0

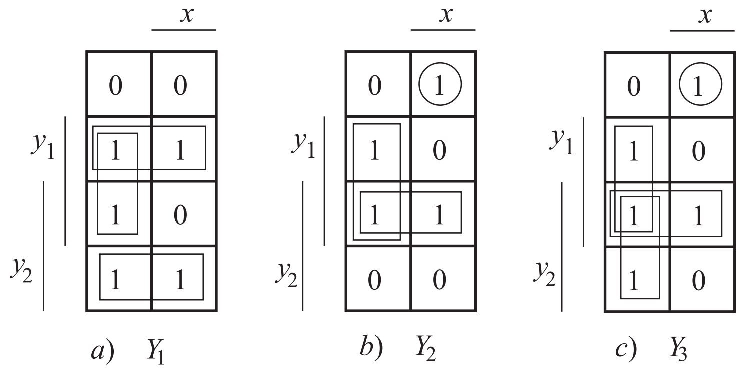
$\}\ \lambda((y_1, y_2), x) = z$

$\delta((y_1, y_2), x) = (Y_1, Y_2)$

V tomto automate je každej dvojici $(y_1, y_2) \in \mathbf{B}^2$ a každému $x \in \mathbf{B}$ priradený nový stav $(Y_1, Y_2) = \delta((y_1, y_2), x) \in \mathbf{B}^2$ a výstup $z = \lambda((y_1, y_2), x) \in \mathbf{B}$. Teda funkciu $\delta : \mathbf{B}^2 \times \mathbf{B} \to \mathbf{B}^2$ môžeme pokladať za funkciu $\delta : \mathbf{B}^3 \to \mathbf{B}^2$ troch premenných y_1, y_2, x, ktorá má dve zložky $\delta_1 : \mathbf{B}^3 \to \mathbf{B}$, $\delta_1(y_1, y_2, x) = Y_1$ a $\delta_2 : \mathbf{B}^3 \to \mathbf{B}$, $\delta_2(y_1, y_2, x) = Y_2$. Tak isto aj funkciu $\lambda : \mathbf{B}^2 \times \mathbf{B} \to \mathbf{B}$ môžeme považovať za funkciu troch premenných, ktorá má jednu zložku $\lambda : \mathbf{B}^3 \to \mathbf{B}$, $\lambda(y_1, y_2, x) = z$. Tieto funkcie budeme zapisovať aj pomocou Karnaughových máp. Tabuľky pre jednotlivé zložky funkcie δ a pre výstupnú funkciu λ uvádzame na obr. 1, časť a), b), c). Vidíme, že každá z týchto tabuliek predstavuje jednu logickú funkciu troch premenných y_1, y_2, x. Tieto funkcie budeme reprezentovať pomocou MNDF, ktoré sú priradené Karnaughovým mapám na obr. 1, časť a), b), c). Potom dostávame:

$$
\begin{aligned}
\delta_1(y_1, y_2, x) &= Y_1 = y_1\bar{y}_2 + y_1\bar{x} + \bar{y}_1 y_2,\\
\delta_2(y_1, y_2, x) &= Y_2 = \bar{y}_1\bar{y}_2 x + y_1\bar{x} + y_1 y_2,\\
\lambda(y_1, y_2, x) &= z = \bar{y}_1\bar{y}_2 x + y_1\bar{x} + y_1 y_2 + y_2\bar{x}.
\end{aligned}
$$

Vidíme, že budeme schopní generovať nový stav a výstup, ak budeme vedieť opísať zariadenie, ktoré uchová predošlý stav (y_1, y_2). ∎



OBR. 1. Tabuľky budiacich funkcií a výstupnej funkcie z príkladu 4.1

Teraz budeme uvažovať všeobecne. Nech $A = (\mathbf{B}^k, \mathbf{B}^n, \mathbf{B}^m, \delta, \lambda)$ je dvojkový automat. Potom prechodovú funkciu $\delta : \mathbf{B}^k \times \mathbf{B}^n \to \mathbf{B}^k$ budeme považovať za funkciu $k+n$ premenných, pričom táto funkcia bude mať k zložiek. Píšeme

$\delta : \mathbf{B}^{k+n} \to \mathbf{B}^k,\ \delta(y_1, \dots, y_k, x_1, \dots, x_n) = (\delta_1(y_1, \dots, y_k, x_1, \dots, x_n),$
$\delta_2(y_1, \dots, y_k, x_1, \dots, x_n), \dots, \delta_k(y_1, \dots, y_k, x_1, \dots, x_n)) = (Y_1, Y_2, \dots, Y_k).$

Zložky

$\delta_i : \mathbf{B}^{k+n} \to \mathbf{B}, \quad \delta_i(y_1, \dots, y_k, x_1 \dots, x_n) = Y_i$ pre $i = 1, 2, \dots, k$

nazývame ***budiace funkcie*** daného dvojkového automatu. Tieto funkcie pomocou súčasného stavu a vstupu generujú (budia) nový stav $(Y_1, Y_2, \dots, Y_k)$.

Funkciu $\lambda : \mathbf{B}^k \times \mathbf{B}^n \to \mathbf{B}^m$ tiež považujeme za funkciu $k+n$ premenných, ktorá má m zložiek. Preto

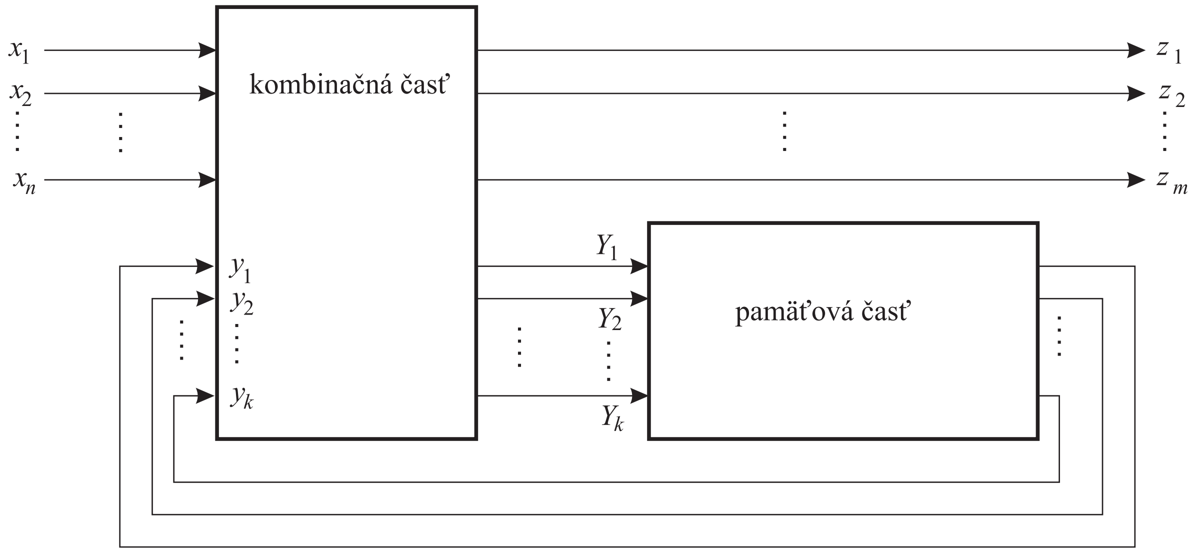
$\lambda : \mathbf{B}^{k+n} \to \mathbf{B}^m,$
$\lambda(y_1, \dots, y_k, x_1, \dots, x_n) = (\lambda_1(y_1, \dots, y_k, x_1, \dots, x_n),$
$\lambda_2(y_1, \dots, y_k, x_1, \dots, x_n), \dots, \lambda_m(y_1, \dots, y_k, x_1, \dots, x_n)) = (z_1, z_2, \dots, z_m).$

Zložky

$\lambda_j : \mathbf{B}^{k+n} \to \mathbf{B}, \quad \lambda_j(y_1, \dots, y_k, x_1, \dots, x_n)) = z_j$ pre $j = 1, 2, \dots, m$

nazývame ***výstupné funkcie*** daného dvojkového automatu. V prípade Moorovho automatu budú výstupné funkcie iba funkciami premenných $y_1, y_2, \dots, y_k$.

To znamená, že spolu s každým dvojkovým automatom môžeme uvažovať o k-tici budiacich funkcií a o m-tici výstupných funkcií. Tieto funkcie sú logické funkcie $k+n$ premenných. Je zrejmé, že fyzikálnou realizáciou dvojkového automatu bude logický obvod, alebo presnejšie ***logický sekvenčný obvod***. Tento obvod nebudeme môcť reprezentovať pomocou obyčajnej kombinačnej logickej siete, ale pomocou tzv. ***sekvenčnej logickej siete***. Kombinačné siete priradíme iba budiacim a výstupným funkciám, avšak vstupné premenné $y_1, y_2, \dots, y_k$ týchto logických sietí (teda stavové premenné daného dvojkového automatu) musia byť generované v inej časti, v ktorej sa budú uchovávať z predchádzajúceho taktu. Táto časť sekvenčnej logickej siete sa bude nazývať ***pamäťová časť***. V najjednoduchšom prípade, keď pamäťová časť vytvára len jednoduché oneskorenie o jeden takt (teda je napr. zostavená z tzv. D-preklápacích obvodov - pozri neskôr), môžeme schému sekvenčnej logickej siete znázorniť na obr. 2. Vstupné vrcholy pamäťovej časti budú priradené hodnotám $Y_1, Y_2, \dots, Y_k$ budiacich funkcií. Výstupné vrcholy pamäťovej časti budú ohodnotené hodnotami stavov $y_1, y_2, \dots, y_k$, ktoré sa v pamäťovej časti uchovávajú z predošlého taktu. Toto ohodnotenie môžeme v časovej interpretácii vyjadriť takto: $Y_i(t) = y_i(t+1)$ pre $i = 1, 2, \dots, k$.



Obr. 2. Schéma sekvenčnej logickej siete (najjednoduchší prípad)

Pamäťová časť vo všeobecnosti môže mať zložitejšie správanie ako jednoduché oneskorovanie o jeden takt. Potom jej vstupné vrcholy nebudú priradené hodnotám budiacich funkcií $Y_1, \dots, Y_k$, ale vstupným funkciám jednotlivých ***preklápacích obvodov*** (pozri neskôr), z ktorých je zostavená. Samotné budiace funkcie sa potom generujú a do ďalšieho taktu uchovávajú práve v týchto preklápacích obvodoch.

Budovanie pamäťovej časti ukážeme postupne po definovaní jej logických členov (preklápacích obvodov), z ktorých je zostavená. Podotknime, že sekvenčná logická sieť sa od kombinačnej logickej siete odlišuje hlavne prítomnosťou spätných väzieb. Ďalšia odlišnosť je nasledovná: hodnoty výstupných premenných v sekvenčných logických sieťach závisia (na rozdiel od kombinačných logických sietí) nielen od vektora hodnôt vstupných premenných v danom čase, ale aj od postupnosti vstupných vektorov v predchádzajúcom čase. Preto sekvenčná logická sieť spracováva postupnosti (sekvencie) vstupných vektorov.

2. Preklápacie obvody

A) ***SR - preklápací obvod***

Uvažujme o dvojkovom Moorovom automate $A = (\mathbf{B}, \mathbf{B}^2, \mathbf{B}, \delta, \mu)$, ktorého prechodová funkcia je daná na obr. 3, časť a) a výstupná funkcia na obr. 3, časť b). Pritom vstupné

		S	S, R	R
	0	1	0	0
Q	1	1	0	0

a)

	0
Q	1

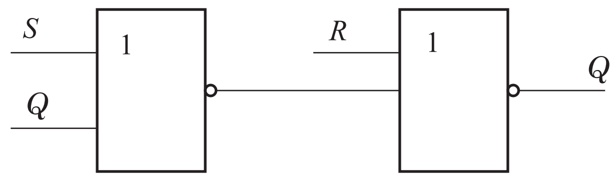
b)

Obr. 3. Tabuľky SR - preklápacieho obvodu

premenné označujeme S, R a stavovú premennú znakom Q. Vidíme, že v tomto prípade má výstupná funkcia hodnoty zhodné so stavmi, teda $z = \mu(Q) = Q$. Pre budiacu funkciu, ktorú v tomto prípade budeme označovať Q', z Karnaughovej mapy na obr. 3, časť a) dostávame $\delta(Q, S, R) = Q' = Q\bar{R} + S\bar{R} = (Q + S)\bar{R}$. Navrhneme logickú sieť patriacu k tejto funkcii, ktorú zostavíme z členov NOR. Preto

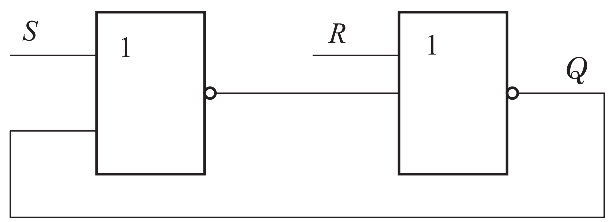
$$Q' = (Q + S)\bar{R} = \overline{\overline{(Q + S)\bar{R}}} = \overline{\overline{(Q + S)} + R}.$$

Teda má zmysel uvažovať o kombinačnej logickej sieti na obr. 4. Pri tomto zapojení sa budeme zaujímať o tie situácie, keď pri nezmenených vstupných hodnotách S, R sa od



Obr. 4. Ilustrácia SR - preklápacieho obvodu

seba nebudú odlišovať hodnoty stavových premenných Q a Q', čiže o situácie, keď $Q = Q'$. Preto má zmysel uvažovať o zapojení na obr. 5. V tomto prípade už nejde o kombinačnú



Obr. 5. Ilustrácia SR - preklápacieho obvodu

logickú sieť, ale o sekvenčnú logickú sieť. Vyznačuje sa prítomnosťou spätnej väzby. Ak označíme $P = \overline{(Q + S)}$, môžeme túto schému kresliť tak, ako to uvádzame na obr. 6. Pri tomto zapojení má už zmysel hovoriť o nasledujúcom stave, v ktorom sa sieť ustáli, ako bude zrejmé z ďalšieho.

Z tabuľky na obr. 3, časť a) vyplýva, že nový stav Q' sa rovná predošlému stavu Q (teda stav Q je stabilný) v troch prípadoch pre $Q = 0$ (tri nuly v prvom riadku tabuľky) a v dvoch prípadoch pre $Q = 1$ (dve jednotky v druhom riadku tabuľky). Teraz vymenujeme všetky tieto prípady, pričom ešte uvádzame aj hodnoty $P = \overline{(Q + S)}$ a $P' = \overline{(Q' + S)}$.

$$\begin{aligned}
&1)\quad S = 0, R = 0, Q = 0 = Q', P = 1 = P',\\
&2)\quad S = 1, R = 1, Q = 0 = Q', P = 0 = P',\\
&3)\quad S = 0, R = 1, Q = 0 = Q', P = 1 = P',\\
&4)\quad S = 0, R = 0, Q = 1 = Q', P = 0 = P',\\
&5)\quad S = 1, R = 0, Q = 1 = Q', P = 0 = P'.
\end{aligned}$$

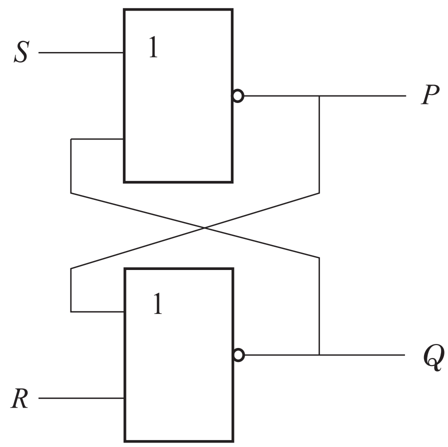
Teraz sa ešte budeme zaoberať zvyšnými troma situáciami , ktoré zodpovedajú jednej jednotke v prvom riadku tabuľky na obr. 3, časť a) a dvom nulám v druhom riadku tejto tabuľky.
V prvom prípade sa stav $Q = 0$ mení na stav $Q' = 1$ a vo zvyšných dvoch prípadoch sa stav $Q = 1$ mení na stav $Q' = 0$. Teda máme nasledujúce tri situácie:

6) $S = 1, R = 0, Q = 0, P = 0$. V tomto prípade pri pevnom $S = 1$ a $R = 0$ dostávame nový stav $Q' = (Q + S)\bar{R} = 1$ a $P' = \overline{(Q' + S)} = 0$. Pri nezmenenom vstupe (S, R) sa

zapojenie dostáva do situácie $S = 1, R = 0, Q' = 1, P' = 0$, čo zodpovedá stabilnej situácii 5).

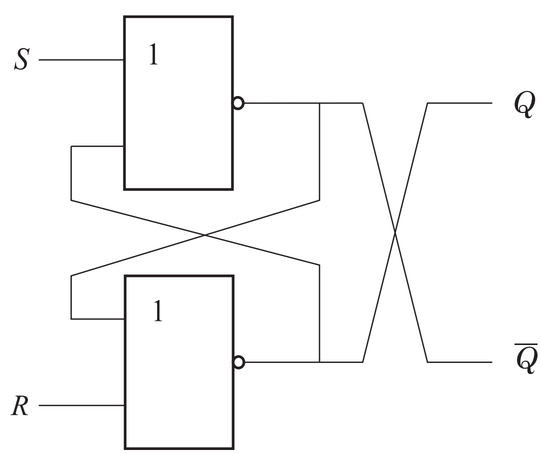
7) $S = 1, R = 1, Q = 1, P = 0$. V tomto prípade nový stav $Q' = (Q + S)\bar{R} = 0$ a $P' = \overline{(Q' + S)} = 0$. Pri nezmenenom vstupe sa zapojenie dostáva do situácie $S = 1$, $R = 1$, $Q' = 0$, $P' = 0$, čo zodpovedá stabilnej situácii 2).

8) $S = 0, R = 1, Q = 1, P = 0$. Nový stav $Q' = (Q + S)\bar{R} = 0$ a $P' = \overline{(Q' + S)} = 1$. Nová situácia je $S = 0, R = 1, Q' = 0, P' = 1$, čo zodpovedá stabilnej situácii 3).



Obr. 6. Ilustrácia SR - preklápacieho obvodu

Vidíme, že zapojenie na obr. 6 sa pri pevne zvolených vstupných hodnotách vždy dostáva do stabilného stavu, ktorý zodpovedá jednej z možných situácií 1) - 5). Vzhľadom k fyzikálnym charakteristikám SR - preklápacieho obvodu dochádza k tomuto prechodu (pri fyzikálnej realizácii) vždy s istým časovým oneskorením.
Všimnime si ešte jednu dôležitú vec. Okrem prípadov 2) a 7) vo všetkých zvyšných prípadoch je $P' = \bar{Q}$. Prípady 2) a 7) zodpovedajú situácii, keď $S = 1$ a $R = 1$, čiže pri ustálenej situácii (v stabilnom stave) bude $P = \bar{Q}$ okrem prípadu, keď $S = 1$ a súčasne $R = 1$. Podmienka komplementárnosti výstupov je jedným zo základných znakov tzv. ***preklápacích obvodov***. Ak v schéme na obr. 6 nedovolíme výskyt vstupného vektora $(S, R) = (1, 1)$, túto schému môžeme interpretovať takým spôsobom, ako to vidno na obr. 7. Túto schému možno považovať za sekvenčnú logickú sieť, ktorá reprezentuje sek-



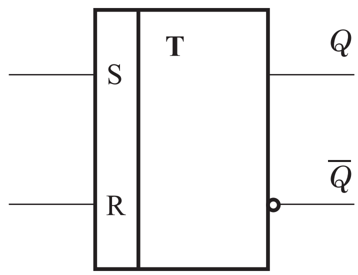
Obr. 7. Ilustrácia SR - preklápacieho obvodu

venčný logický obvod, ktorý je fyzikálnou realizáciou neúplne špecifikovaného Moorovho dvojkového automatu, ktorého prechodová funkcia je daná tabuľkou na obr. 8 (teda vstupy $R = S = 1$ sú zakázané). Výstupná funkcia tohto automatu je daná na obr. 3, časť b). Tento dvojkový automat, jeho príslušnú logickú sieť a aj jeho fyzikálnu realizáciu budeme

		S	S, R	R
	0	1	×	0
Q	1	1	×	0

Obr. 8. Tabuľka prechodovej funkcie SR - preklápacieho obvodu

nazývať ***SR - preklápací obvod***. Na jeho logickú sieť používame štandardné označenie, ktoré uvádzame na obr. 9.



Obr. 9. SR - preklápací obvod

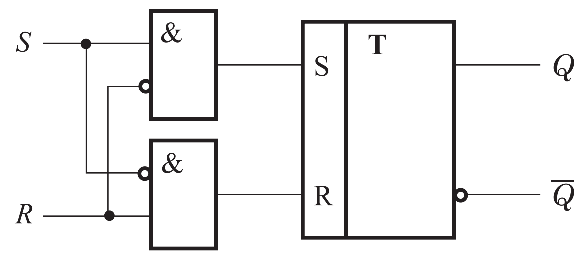
Všimnime si, že SR - preklápací obvod pri vstupe $(S, R) = (0, 0)$ zachováva stav (a aj výstup). Hovoríme, že má pamäťové správanie. Pri $(S, R) = (1, 0)$ tento automat nastavuje nový stav (a teda aj výstup) $Q = 1$. Pri $(S, R) = (0, 1)$ nastavuje nový stav $Q = 0$, čiže $S = 1$ nastavuje nový stav $Q = 1$ a $R = 1$ nastavuje nový stav $Q = 0$. Toto správanie SR - preklápacieho obvodu symbolicky zaznamenávame na obr. 10, kde Pm znamená pamäťové správanie.

	S	S, R	R
Pm	1	×	0

Obr. 10. Symbolický zápis tabuľky SR - preklápacieho obvodu

Zamedzenie vstupu $(S, R) = (1, 1)$ môžeme urobiť pomocou zapojenia, ktoré uvádzame na obr. 11. Pri takomto zapojení v prípade vstupných vektorov $(0, 0), (0, 1), (1, 0)$ sa naozaj tieto vektory dostávajú na vstup SR - preklápacieho obvodu. V prípade $(S, R) = (1, 1)$ sa na vstup preklápacieho obvodu dostáva vektor $(0, 0)$, a teda preklápací obvod zostáva v pôvodnom stave (má pamäťové správanie).

Zamedzenie prístupu vektora $(S, R) = (1, 1)$ na vstup SR - preklápacieho obvodu môžeme robiť aj pomocou programovacích prostriedkov pri generovaní budiacej funkcie SR - preklápacieho obvodu. Znamená to, že všetky možné zmeny starého stavu SR - preklápacieho obvodu Q na nový stav Q' vieme vyrobiť vhodnou voľbou vektora (S, R), pričom nikdy nepotrebujeme $(S, R) = (1, 1)$. Bude to jasné z nasledujúceho. Ak pre stavy SR - preklápacieho obvodu v zhode s označením budiacich funkcií použijeme označenie



Obr. 11. Ilustrácia SR - preklápacieho obvodu

$y = Q$ a $Y = Q'$, z podmienky $Q' = (Q + S)\bar{R}$ dostávame $Y = (y + S)\bar{R}$. Túto rovnosť budeme analyzovať. Pýtame sa, aké hodnoty vstupných premenných S, R treba priviesť na vstup, aby sme zabezpečili všetky možné prechody $y \to Y$. Uvažujme o dvoch možnostiach.

- $y = 0$, potom $Y = S\bar{R}$. V takomto prípade je $Y = 1$ práve vtedy, keď $S = 1$ a $R = 0$. Ďalej $Y = 0$ práve vtedy, keď buď $S = 0$ a R je ľubovoľné ($R = \times$), alebo $S = 1$ a $R = 1$ (táto možnosť je však zakázaná).
- $y = 1$, potom $Y = \bar{R}$. V tomto prípade je $Y = 0$, keď je $R = 1$ a S je ľubovoľné (ale možnosť S=R=1 je zakázaná, teda ostáva $S = 0$ a $R = 1$). $Y = 1$ práve vtedy, keď $R = 0$ a S je ľubovoľné ($S = \times$).

Ak teraz analyzujeme uvedené možnosti, zaujíma nás hlavne zmena starého stavu y na nový stav Y ($y \to Y$), ako už bolo spomenuté. Týmito zmenami je vlastne budiaca funkcia definovaná. Prehľad zmien $y \to Y$ v závislosti od hodnôt S, R uvádzame v tabuľke 2. Všimnime si, že v žiadnom riadku tabuľky sa nevyskytuje možnosť $(S, R) = (1, 1)$, teda náš predpoklad o zamedzení prístupu tohto vektora na vstup SR - preklápacieho obvodu je splnený.

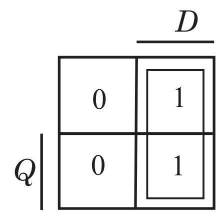
Tabuľka 2. Generovanie budiacej funkcie v SR - preklápacom obvode

$y \to Y$	S	R
$0 \to 0$	0	$\times$
$0 \to 1$	1	0
$1 \to 0$	0	1
$1 \to 1$	$\times$	0

Túto tabuľku budeme používať v konkrétnych príkladoch, keď pomocou SR - preklápacích obvodov zostavíme pamäťovú časť sekvenčného obvodu, ktorý bude reprezentovať fyzikálnu realizáciu daného dvojkového automatu.

B) ***D - preklápací obvod***

D - preklápací obvod nazývame Moorov automat $A = (\mathbf{B}, \mathbf{B}, \mathbf{B}, \delta, \mu)$, ktorého prechodová funkcia je daná tabuľkou na obr. 12. Vstupnú premennú označujeme znakom D, stavovú premennú znakom Q. Tento automat má opäť výstupy zhodné so stavmi. Preto tabuľka výstupnej funkcie je tá istá ako pri SR - preklápacom obvode na obr. 3, časť b). Tak isto ako daný automat nazývame aj príslušnú logickú sieť a jej fyzikálnu realizáciu. Pre budiacu funkciu tohto automatu dostávame $Y = D$. Tento výsledok čítame z Karnaughovej mapy na obr. 12, pričom $Y = Q'$. Tento automat uchováva (zapamätáva si) vstup D do nasledujúceho taktu vo forme stavu. Preto sa nazýva aj oneskorovací člen.



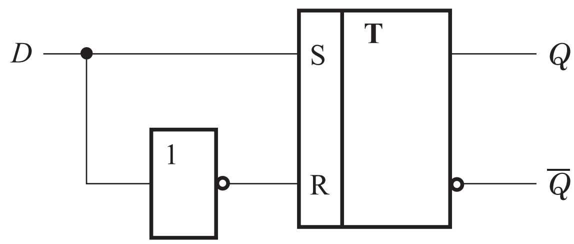
Obr. 12. Tabuľka prechodovej funkcie D - preklápacieho obvodu

Ak pamäťová časť sekvenčnej logickej siete je zostavená zo samých oneskorovacích členov, potom schéma sekvenčnej logickej siete na obr. 2 je presná, ako už bolo spomenuté skôr.

Fyzikálnu realizáciu D - preklápacieho obvodu získame takto:

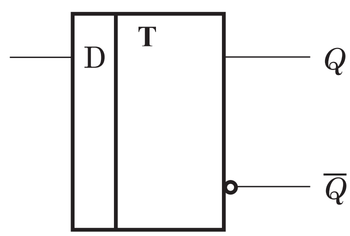
$$Q' = D = D + DQ = (D + Q)D = (D + Q)\bar{\bar{D}}.$$

Ak porovnáme tento výsledok s budiacou funkciou $Q' = (S + Q)\bar{R}$ v SR - preklápacom obvode, vidíme, že na vstup S v SR - preklápacom obvode stačí priviesť D a na vstup R hodnotu $\bar{D}$. Toto zapojenie uvádzame na obr. 13. Všimnime si pritom, že je vylúčená možnosť $(S, R) = (1, 1)$.



Obr. 13. Sekvenčná logická sieť patriaca k D - preklápaciemu obvodu

Pre sekvenčnú logickú sieť D - preklápacieho obvodu používame štandardné označenie, ktoré uvádzame na obr. 14



Obr. 14. D - preklápací obvod

C) ***JK - preklápací obvod***

JK - preklápací obvod nazývame dvojkový Moorov automat $A = (\mathbf{B}, \mathbf{B}^2, \mathbf{B}, \delta, \mu)$, ktorého prechodová funkcia je daná pomocou tabuľky na obr. 15.

V tomto automate označujeme vstupné premenné znakmi J,K a stavovú premennú znakom Q. Tento automat má opäť výstupy zhodné so stavmi, preto tabuľka výstupnej funkcie je zhodná s tabuľkou na obr. 3, časť b). Tak isto ako tento automat nazývame aj príslušnú sekvenčnú logickú sieť a jej fyzikálnu realizáciu. JK - preklápací obvod sa odlišuje od SR - preklápacieho obvodu tým, že je povolený aj vstupný vektor $(J, K) = (1, 1)$, pri ktorom sa mení (preklápa) stav Q z 0 na 1 a naopak. Hovoríme, že vtedy má obvod preklápacie správanie. Správanie JK - preklápacieho obvodu symbolicky uvádzame na obr. 16, kde Pm znamená pamäťové a Pk preklápacie správanie.

	J	J, K	K
0	1	1	0
Q: 1	1	0	0

Obr. 15. Tabuľka JK - preklápacieho obvodu

	J	K	K
Pm	1	Pk	0

Obr. 16. Symbolický zápis tabuľky JK - preklápacieho obvodu

Pre budiacu funkciu $Y = Q'$ dostávame $Y = y\bar{K} + \bar{y}J$, kde $y = Q$. Potom pre $y = 0$ je $Y = J$ a pre $y = 1$ je $Y = \bar{K}$. Z toho vyplýva, že:

- Ak $y = 0$, tak $Y = 1$ práve vtedy, keď $J = 1$ a K je ľubovoľné (píšeme $K = \times$) a $Y = 0$ práve vtedy, keď $J = 0$ a $K = \times$.
- Ak $y = 1$, tak $Y = 1$ práve vtedy, keď $K = 0$ a $J = \times$, a $Y = 0$ práve vtedy, keď $K = 1$ a $J = \times$.

Z toho už vyplýva konštrukcia tabuľky, ktorá opisuje zmeny hodnôt budiacej funkcie Y zo starého stavu y vzhľadom na možné hodnoty vstupného vektora (J, K). Tieto zmeny sú opísané v tabuľke 3.

Tabuľka 3. Budiaca funkcia JK - preklápacieho obvodu

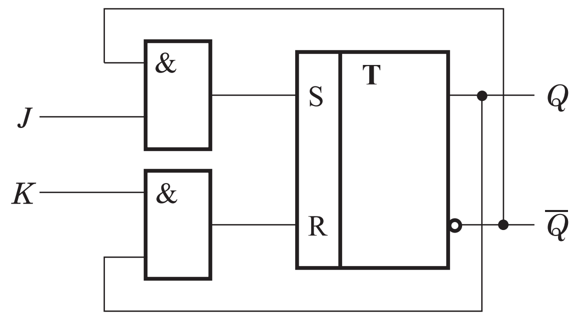
$y \to Y$	J	K
$0 \to 1$	$\left.\begin{matrix}1\\0\end{matrix}\right\} Y$	$\times$
$0 \to 0$		$\times$
$1 \to 1$	$\times$	$\left.\begin{matrix}0\\1\end{matrix}\right\} \overline{Y}$
$1 \to 0$	$\times$	

Zjednodušenú formu tejto tabuľky uvádzame na obr. 17.

	J	K
	Y	$\times$
y	$\times$	$\overline{Y}$

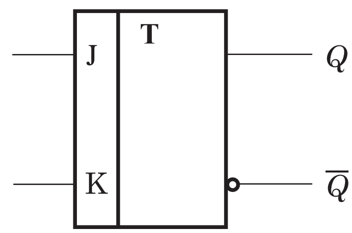
Obr. 17. Zjednodušená tabuľka budiacej funkcie JK - preklápacieho obvodu

Budiaca funkcia JK - preklápacieho obvodu má tvar $Q' = Q\bar{K} + \bar{Q}J$. Jednoduchými úpravami sa možno presvedčiť, že platí aj $Q' = (Q + \bar{Q}J)\overline{KQ}$. Ak porovnáme tento zápis budiacej funkcie s budiacou funkciou SR - preklápacieho obvodu, ktorá je $Q' = (Q + S)\bar{R}$, vidíme, že budiacu funkciu JK - preklápacieho obvodu môžeme realizovať pomocou SR - preklápacieho obvodu, ak položíme $S = \bar{Q}J$ a $R = KQ$. Všimnime si, že



Obr. 18. Sekvenčná logická sieť patriaca k JK - preklápaciemu obvodu

je pritom vylúčená možnosť $(S, R) = (1, 1)$. Teda jednu zo sekvenčných sietí patriacich k JK - preklápaciemu obvodu môžeme navrhnúť podľa obr. 18. Štandardné označenie pre ľubovoľnú logickú sieť patriacu k JK - preklápaciemu obvodu je na obr. 19.



Obr. 19. JK - preklápací obvod

D) ***T - preklápací obvod***

T - preklápací obvod je dvojkový Moorov automat $A = (\mathbf{B}, \mathbf{B}, \mathbf{B}, \delta, \mu)$, ktorého prechodová funkcia je daná pomocou tabuľky na obr. 20.

		T
	0	1
Q	1	0

Obr. 20. Prechodová funkcia T - preklápacieho obvodu

V tejto tabuľke je vstupná premenná označená znakom T a stavová premenná znakom Q. T - preklápacím obvodom nazývame aj príslušnú logickú sieť a sekvenčný logický obvod, ktorý je fyzikálnou realizáciou daného dvojkového automatu. Predpokladáme, že výstup aj v tomto automate je zhodný so stavom. Preto tabuľka výstupnej funkcie je daná pomocou obr. 3, časť b). Na obr. 20 možno vidieť, že T - preklápací obvod má preklápacie správanie pre $T = 1$ (preklápa stav Q z 0 na 1 a naopak) a pamäťové správanie pre $T = 0$ (pamätá si stav do ďalšieho taktu). Toto správanie T - preklápacieho obvodu je symbolicky znázornené na obr. 21, kde Pm označuje pamäťové a Pk preklápacie správanie.

Ak označíme $Q' = Y$ a $Q = y$, z tabuľky prechodovej funkcie dostávame $Y = \bar{y}T + y\bar{T}$. To znamená, že pri $y = 0$ dostávame $Y = T$ a pri $y = 1$ dostávame $Y = \bar{T}$ (a teda $T = \bar{Y}$). Teda ak $y = 0$, zmena $y \to Y$ nastane voľbou $T = Y$, pre $y = 1$ zmena $y \to Y$ nastane voľbou $T = \bar{Y}$. Toto generovanie budiacej funkcie Y s ohľadom na zmeny $y \to Y$ v závislosti na vstupe T môžeme znázorniť pomocou obr. 22.

	T
Pm	Pk

Obr. 21. Symbolický zápis tabuľky T - preklápacieho obvodu

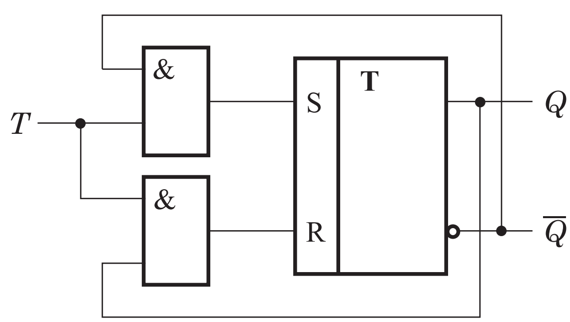
	T
	Y
y	$\overline{Y}$

Obr. 22. Zjednodušená tabuľka budiacej funkcie T - preklápacieho obvodu

Z UNDF budiacej funkcie Q' odvodíme tvar, ktorý nám pomôže navrhnúť jednu z možných sekvenčných logických sietí patriacich k danému automatu. Z tabuľky prechodovej funkcie čítame:

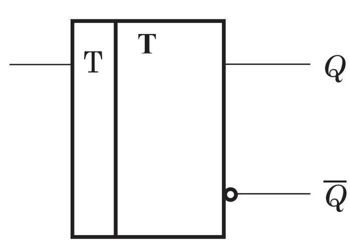
$$\begin{aligned} Q' &= \bar{Q}T + Q\bar{T} = (\bar{Q}T + Q)(\bar{Q}T + \bar{T}) = (\bar{Q}T + Q)((\bar{Q} + \bar{T})(T + \bar{T})) = \\ &= (\bar{Q}T + Q)(\bar{Q} + \bar{T}) = (\bar{Q}T + Q)\overline{QT}. \end{aligned}$$

Ak porovnáme túto budiacu funkciu s budiacou funkciou $Q' = (S + Q)\bar{R}$ SR - preklápacieho obvodu, vidíme, že stačí položiť $S = \bar{Q}T$ a $R = QT$, aby sme dostali realizáciu budiacej funkcie Q' pre T - preklápací obvod. Z toho už vyplýva návrh sekvenčnej logickej



Obr. 23. Sekvenčná logická sieť patriaca k T - preklápaciemu obvodu

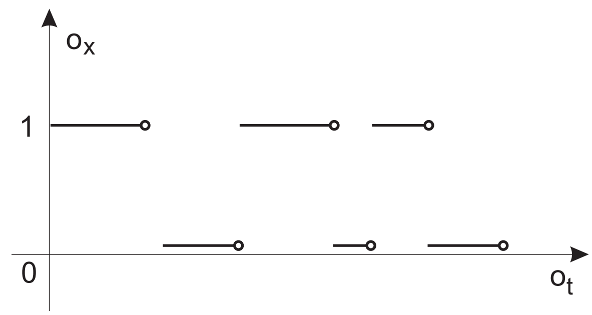
siete patriacej k T - preklápaciemu obvodu, ktorý uvádzame na obr. 23. Všimnime si, že opäť je vylúčená možnosť $(S, R) = (1, 1)$. Pre ľubovoľnú sekvenčnú logickú sieť patriacu k T - preklápaciemu obvodu sa používa schematické označenie z obr. 24.



Obr. 24. T - preklápací obvod

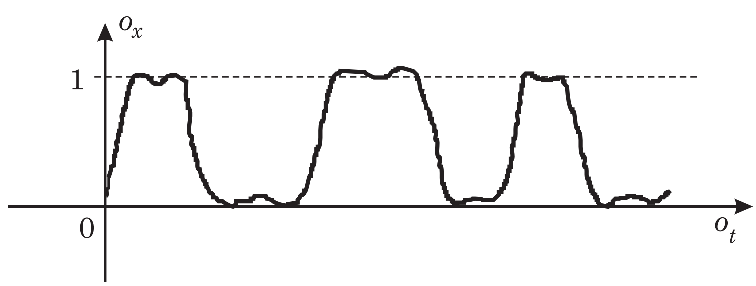
3. Synchrónne a asynchrónne logické obvody

Na vstupoch a výstupoch logických obvodov dostávame vždy iba dve hodnoty, logickú nulu alebo logickú jednotku. Či už pri kombinačných, alebo sekvenčných obvodoch, uvažujeme o jednotlivých vstupných a výstupných vektoroch, ktoré môžu postupne prichádzať na vstup alebo výstup tohto obvodu. Preto na každom vstupe a aj výstupe logického obvodu môžeme uvažovať o postupnosti logických núl a jednotiek, ktoré logický obvod spracúva alebo generuje. V reálnych logických obvodoch sa táto postupnosť prejavuje ako postupnosť vyšších a nižších hladín elektrického napätia alebo prúdu. Preto má zmysel uvažovať o funkcii, ktorá nadobúda iba dve možné hodnoty, 0 a 1. Pre jej jednoznačné určenie môžeme predpokladať, že v bodoch nespojitosti je spojitá sprava. Teda neuvažujeme o možnosti, aby funkcia iba v izolovanom bode mala hodnotu rovnajúcu sa jednej a na celom jeho okolí hodnotu rovnajúcu sa nule a naopak. Takéto funkcie budeme nazývať ***impulzné funkcie***. Graf takejto funkcie je na obr. 25.



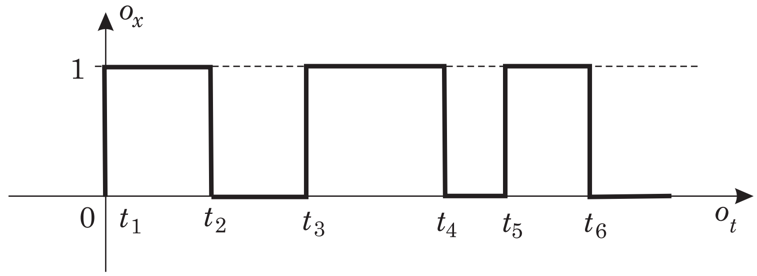
Obr. 25. Graf impulznej funkcie

V reálnych zariadeniach prakticky nikdy nedochádza k okamžitému poklesu (vzostupu) hladiny napätia z nižšej na vyššiu (z vyššej na nižšiu). Zariadenie spravidla potrebuje istý čas na spojitý prechod z jednej hladiny na druhú. Preto reálne „impulzné“ funkcie, ktoré môžeme sledovať na vstupných vrcholoch logického obvodu, majú tvar, ktorý je znázornený na obr. 26. Tento priebeh budeme idealizovane kresliť tak, ako je to na obr. 27, pričom stále platí dohoda o spojitosti impulznej funkcie sprava.



Obr. 26. Graf reálnej impulznej funkcie

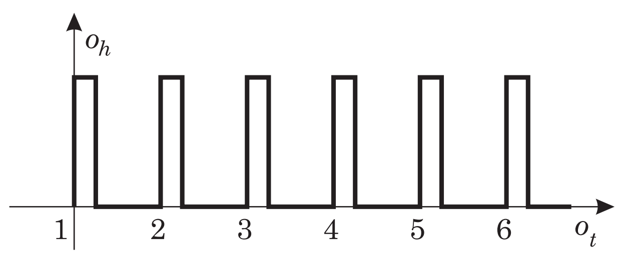
Tu chceme opäť pripomenúť, že vstupy, stavy a výstupy logických obvodov sa menia v časových taktoch. Pri takto definovanej impulznej funkcii sa nám núka diskrétny čas (takty) určiť v bodoch, kde sa daná funkcia mení buď z 0 na 1 alebo z hodnoty 1 na hodnotu 0. Teda impulzná funkcia z obr. 27 predstavuje postupnosť 101010…, pričom tieto hodnoty sme namerali v bodoch $t_1, t_2, t_3, t_4, t_5, t_6, \ldots$ a využili sme predpoklad, že impulzná funkcia je v každom bode spojitá sprava. V budúcnosti budeme body diskrétneho času $t_1, t_2, t_3, \ldots$ jednoducho označovať iba pomocou postupnosti $1, 2, 3, \ldots$. Ako vidíme, pri takomto nazeraní na impulznú funkciu, tá môže predstavovať iba postupnosť 101010…



Obr. 27. Graf idealizovanej impulznej funkcie

alebo postupnosť 010101.... V takomto prípade nám robí problém generovať dva rovnaké signály za sebou. Tento nedostatok sa dá odstrániť dvoma spôsobmi.

(1) Použijeme generátor časových impulzov, čo je v podstate opäť impulzná funkcia, ktorá je periodická a každá perióda sa skladá iba z jedného intervalu, na ktorom je daná funkcia rovnajúca sa jednej (impulz) a z jedného intervalu, na ktorom sa daná funkcia rovná nule (pauza). Budeme využívať jedno veľké zjednodušenie, pri ktorom predpokladáme, že interval, v ktorom daná funkcia nadobúda hodnotu rovnajúcu sa jednej, je veľmi krátky (ide o generátor časových impulzov s krátkym vzorkovaním). Preto o hodnotách diskrétneho času budeme uvažovať iba pri zmenách 0 na 1 (pozri obr. 28). Ak teraz z impulznej funkcie $x = x(t)$ chceme generovať postupnosť zloženú z 0 a 1 (dvojkovú postupnosť), stačí po-

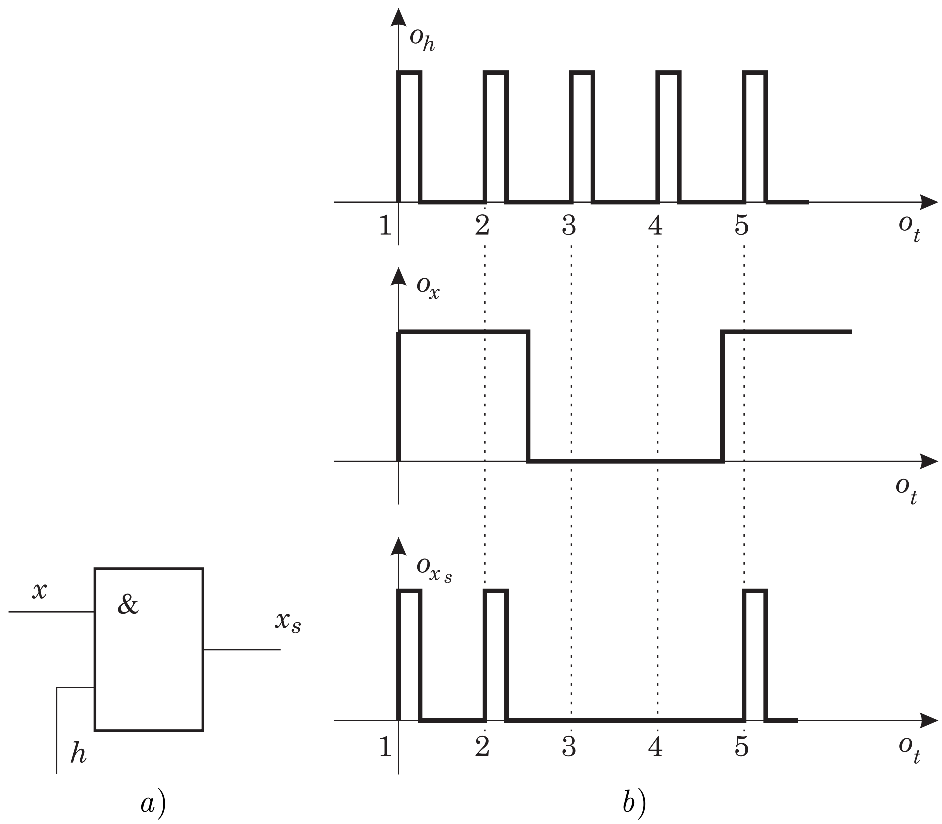


Obr. 28. Generátor časových impulzov

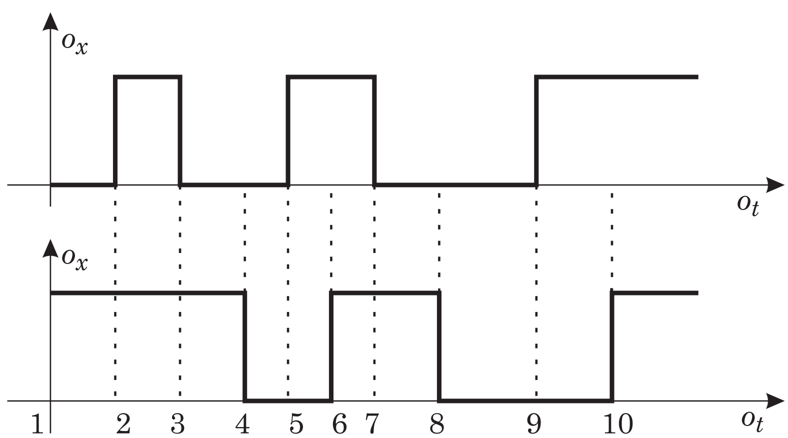
mocou súčinového člena spojiť vstup x a generátor časových impulzov $h = h(t)$ (obr. 29, časť a)). Potom pri takomto spojení generujeme na výstupe súčinového člena v čase $t = 1, 2, 3, \ldots$ príslušnú postupnosť synchronizovaných signálov x_s (obr. 29, časť b)). Logické obvody, v ktorých buď vstupné veličiny, alebo stavové veličiny, alebo oboje sú synchronizované pomocou generátora časových impulzov, nazývame ***synchrónne logické obvody***.

(2) V prípade, že v logickom obvode uvažujeme o dvoch (a viacerých) impulzných funkciách vstupných a stavových premenných, môžeme za body diskrétneho času zobrať všetky body, v ktorých sa mení hociktorá z uvažovaných impulzných funkcií. (pozri obr. 30). Potom impulzná funkcia x z obr. 30 reprezentuje dvojkovú postupnosť 0100110011... a impulzná funkcia y z toho istého obrázka dvojkovú postupnosť 1110011001... . Logické obvody, v ktorých je diskrétny čas určený zmenami vstupných alebo stavových veličín, nazývame ***asynchrónne logické obvody***.

Pri preklápacích obvodoch sme doteraz neuvažovali o synchronizácii ani vstupných a ani stavových veličín. Pri nich sme teda uvažovali o asynchrónnej verzii. Spôsob práce SR - preklápacieho asynchrónneho logického obvodu potom môžeme znázorniť pomocou obr. 31. Pritom predpokladáme, že nový stav je nastavený s istým oneskorením τ, ktoré



Obr. 29. Synchronizácia signálov

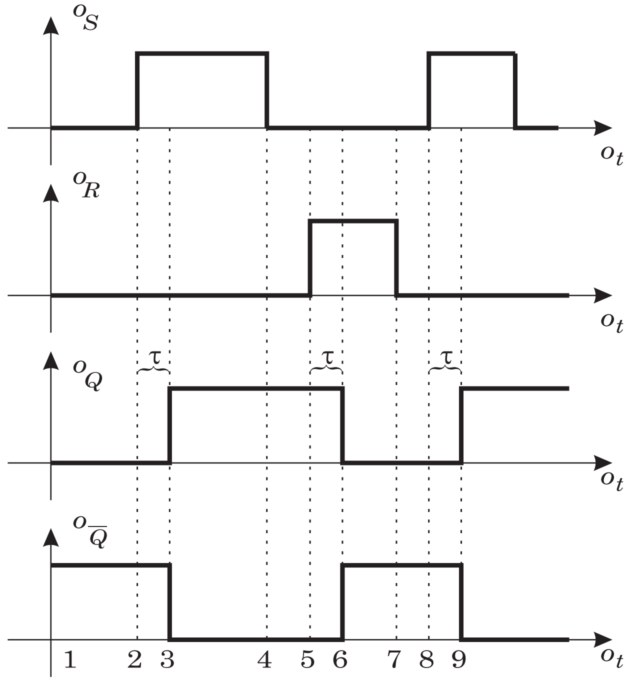


Obr. 30. Diskrétny čas v asynchrónnych logických obvodoch

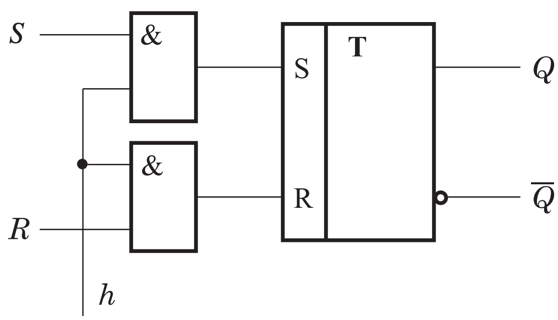
je spôsobené fyzikálnou podstatou SR - preklápacieho obvodu. Ak by sme chceli uvažovať o synchronizácii vstupných premenných S, R, tak by sme mohli uvažovať o zapojení na obr. 32. V takomto prípade je režim práce tohto SR - preklápacieho obvodu znázornený na obr. 33. Pre synchrónne SR - preklápacie obvody (ktorých zapojenie je znázornené napríklad na obr. 32) používame štandardné označenie z obr. 34.

Podobne (ale v značne idealizovanej podobe) môžeme vytvoriť synchrónne D -, JK - a T - preklápacie obvody.

Návrh sekvenčnej logickej siete a schematické označenie synchrónneho D - preklápacieho obvodu uvádzame na obr. 35, časť a), b).



Obr. 31. Režim práce asynchrónneho SR - preklápacieho obvodu



Obr. 32. Synchronizácia vstupných premenných v SR - preklápacom obvode

Návrh sekvenčnej logickej siete a schematické označenie synchrónneho JK - preklápacieho obvodu je na obr. 36, časť a), b).

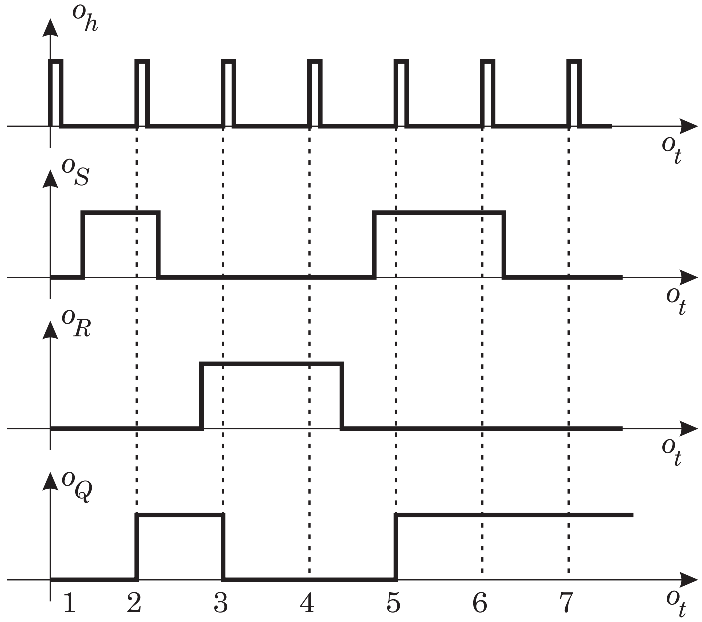
Návrh sekvenčnej logickej siete a schematické označenie synchrónneho T - preklápacieho obvodu je na obr. 37, časť a), b).

Pri našich návrhoch synchrónnych preklápacích obvodov sme buď uvažovali iba o najjednoduchších možných zapojeniach alebo o zjednodušených schémach. Oveľa podrobnejšie a presnejšie sa čitateľ môže o tejto problematike dozvedieť v učebnici [**3**]. Tu chceme ešte poznamenať, že D -, JK - a T - preklápacie asynchrónne obvody majú iba teoretický význam. Existujú len ich synchrónne realizácie.

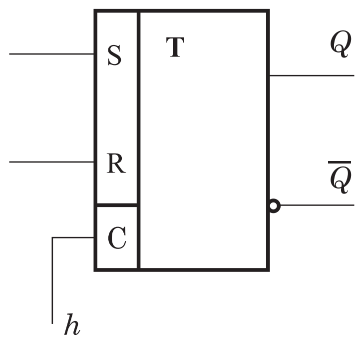
4. Fyzikálna realizácia automatov

Pod fyzikálnou realizáciou automatu A rozumieme logický obvod, ktorý na každú postupnosť vstupných signálov reaguje rovnako ako pôvodný automat A. Tento logický obvod budeme reprezentovať pomocou logickej siete. V ďalšom bude tento postup upresnený.

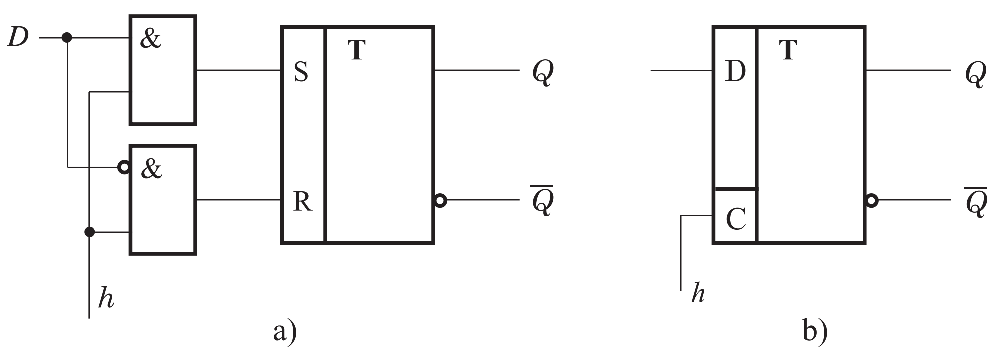
Kvôli zrozumiteľnosti ďalších úvah treba zopakovať nasledujúce. Nech $f : X \to Y$ je ľubovoľné zobrazenie. Hovoríme, že f je injektívne (injekcia) práve vtedy, keď je prosté (a to znamená, že rôznym vzorom sú priradené rôzne obrazy). Ďalej nech M je ľubovoľná



Obr. 33. Režim práce synchrónneho SR - preklápacieho obvodu



Obr. 34. Synchrónny SR - preklápací obvod



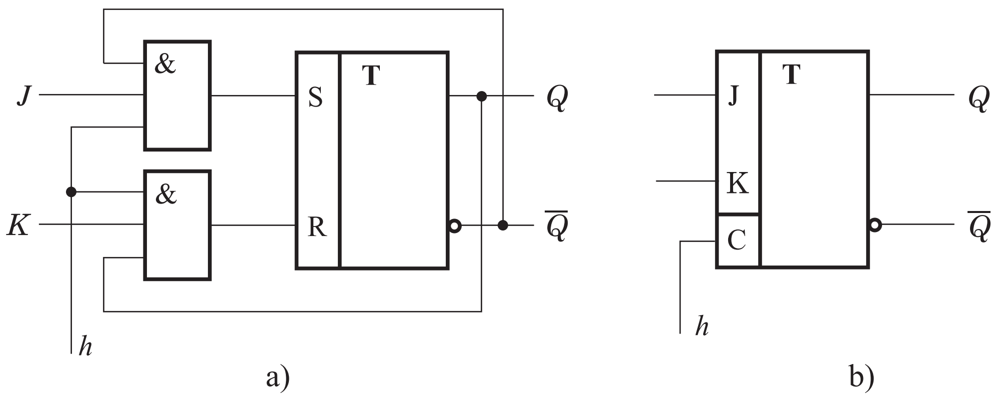
Obr. 35. Synchrónny D - preklápací obvod

množina a $n \in N$ je ľubovoľné prirodzené číslo. Potom M^n označuje množinu $M \times M \times \cdots \times M$ - karteziánsky súčin n kópií množiny M.

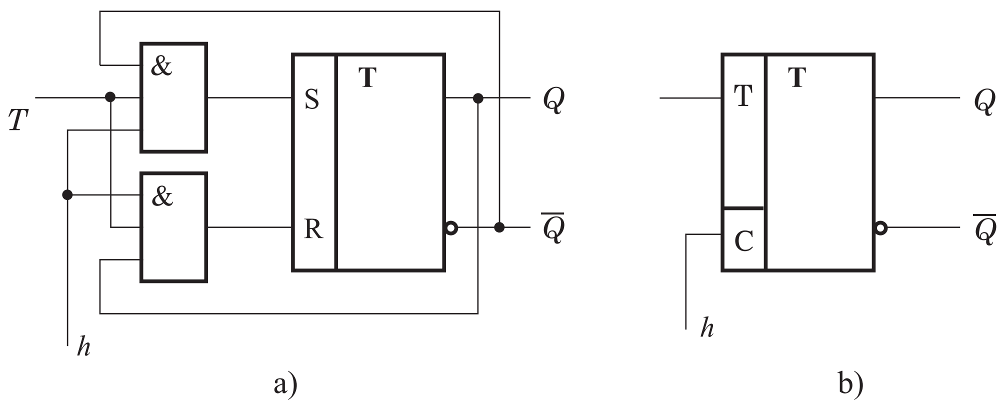
Nech teraz $A = (S, X, Z, \delta, \lambda)$ je automat, kde

$$S = \{s_1, \ldots, s_k\}, \quad X = \{x_1, \ldots, x_n\}, \quad Z = \{z_1, \ldots, z_m\}.$$

Prvky množín S, X, Z zakódujeme pomocou príslušných n-tíc núl a jednotiek. Preto definujeme injekcie (kódovacie funkcie) $f : S \to \mathbf{B}^\alpha, g : X \to \mathbf{B}^\beta$,



Obr. 36. Synchrónny JK - preklápací obvod



Obr. 37. Synchrónny T - preklápací obvod

$h : Z \to \mathbf{B}^\gamma$, kde α, β a γ sú vhodné prirodzené čísla. Aby sme zaručili injektívnosť týchto funkcií, musíme žiadať, aby bolo splnené $k \le 2^\alpha, n \le 2^\beta, m \le 2^\gamma$. Prvky množín $\mathbf{B}^\alpha, \mathbf{B}^\beta, \mathbf{B}^\gamma$ budeme nazývať kódové slová. Ak chceme zaručiť, aby kódové slová neboli zbytočne dlhé, uvedené nerovnosti môžeme doplniť takto:

$$2^{\alpha-1} < k \le 2^\alpha, \quad 2^{\beta-1} < n \le 2^\beta, \quad 2^{\gamma-1} < m \le 2^\gamma.$$

Potom definujeme automat $A_\mathbf{B} = (f(S), g(X), h(Z), \delta_\mathbf{B}, \lambda_\mathbf{B})$, kde prechodovú funkciu $\delta_\mathbf{B} : f(S) \times g(X) \to f(S)$ definujeme takto:

$$\delta_\mathbf{B}(f(s), g(x)) = f(t) \quad \text{práve vtedy, keď} \quad \delta(s, x) = t.$$

Pre výstupnú funkciu $\lambda_\mathbf{B} : f(S) \times g(X) \to h(Z)$ žiadame, aby

$$\lambda_\mathbf{B}(f(s), g(x)) = h(z) \quad \text{práve vtedy, keď} \quad \lambda(s, x) = z,$$

čiže pre každé $s \in S$ a každé $x \in X$ je

$$f(t) = f(\delta(s, x)) = \delta_\mathbf{B}(f(s), g(x))$$

a

$$h(z) = h(\lambda(s, x)) = \lambda_\mathbf{B}(f(s), g(x)).$$

Preto automaty A a $A_\mathbf{B}$ sú izomorfné. Automat $A_\mathbf{B}$ budeme nazývať ***kódový ekvivalent automatu A***.

Aj napriek tomu, že $f(S) \subset \mathbf{B}^\alpha, g(X) \subset \mathbf{B}^\beta, h(Z) \subset \mathbf{B}^\gamma$, automat $A_\mathbf{B}$ ešte nemusí byť dvojkový automat (niektoré z uvedených inklúzií môžu byť ostré), môžeme ho však pokryť dvojkovým automatom $\tilde{A}_\mathbf{B} = (\mathbf{B}^\alpha, \mathbf{B}^\beta, \mathbf{B}^\gamma, \tilde{\delta}_\mathbf{B}, \tilde{\lambda}_\mathbf{B})$, ktorý obsahuje automat $A_\mathbf{B}$ ako svoj podautomat, a funkcie $\tilde{\delta}_\mathbf{B} : \mathbf{B}^\alpha \times \mathbf{B}^\beta \to \mathbf{B}^\alpha$ a $\tilde{\lambda}_\mathbf{B} : \mathbf{B}^\alpha \times \mathbf{B}^\beta \to \mathbf{B}^\gamma$ sú ľubovoľné rozšírenia funkcií $\delta_{\mathbf{B}}$ a $\lambda_{\mathbf{B}}$. V tomto prípade je automat $A_{\mathbf{B}}$ podautomatom automatu $\tilde{A}_{\mathbf{B}}$ a automat $A_{\mathbf{B}}$ je izomorfný a automatom A, preto automat $\tilde{A}_{\mathbf{B}}$ pokrýva automat A.

Príklad 4.2. Nech automat $A = (\{p.q, r, s\}, \{a, b, c, d\}, \{z_1, z_2\}, \delta, \lambda)$ je daný pomocou tabuľky 4. Nájdeme dvojkový automat $\tilde{A}_{\mathbf{B}}$,ktorý daný automat pokrýva.

Tabuľka 4. Automat z príkladu 4.2

	a	b	c	d
p	p/z_1	q/z_1	p/z_2	r/z_2
q	q/z_2	p/z_2	s/z_1	s/z_1
r	r/z_2	p/z_1	q/z_2	r/z_1
s	p/z_1	q/z_2	r/z_1	s/z_2

Riešenie. Budeme definovať príslušné kódovacie funkcie. Najprv definujme funkciu $f : \{p, q, r, s\} \to \mathbf{B}^2$ takto:

$$f(p) = (0, 0), f(q) = (1, 0), f(r) = (1, 1), f(s) = (0, 1).$$

Toto zakódovanie zvykneme symbolicky zapisovať do tabuľky na obr. 38, časť a). Kódovaciu tabuľku funkcie $g : \{a, b, c, d\} \to \mathbf{B}^2$ uvádzame na obr. 38, časť b). Z nej vyplýva,že

$$g(a) = (0, 0), g(b) = (0, 1), g(c) = (1, 0), g(d) = (1, 1).$$

Podobne z obr. 38, časť c) vidíme, že kódovacia funkcia $h : \{z_1, z_2\} \to \mathbf{B}$ je definovaná predpisom $h(z_1) = 0, h(z_2) = 1$.

		y_2
	p	s
y_1	q	r

a)

		x_2
	a	b
x_1	c	d

b)

	z_1
z	z_2

c)

Obr. 38. Kódovacie tabuľky z príkladu 4.2

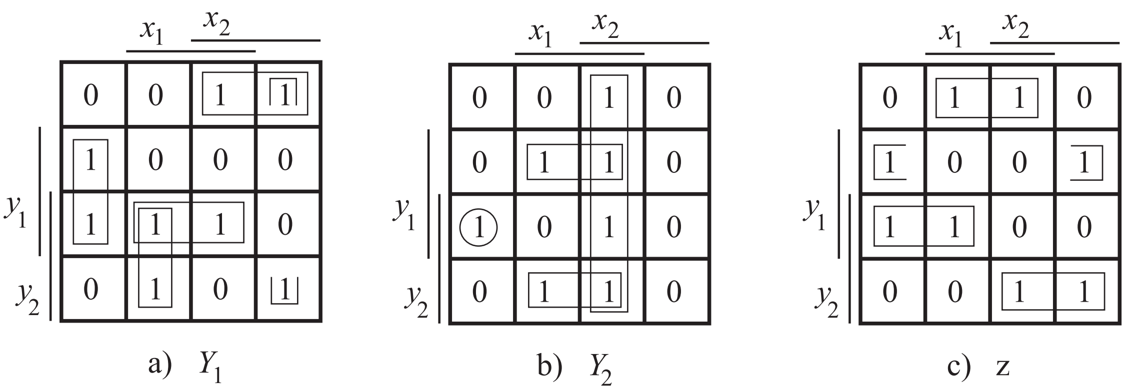
Po tomto zakódovaní dostaneme dvojkový automat $A_{\mathbf{B}} = (\mathbf{B}^2, \mathbf{B}^2, \mathbf{B}, \delta_{\mathbf{B}}, \lambda_{\mathbf{B}})$, ktorý je kódovým ekvivalentom automatu A, a zároveň je to dvojkový automat, ktorý pokrýva automat A, teda $A_{\mathbf{B}} = \tilde{A}_{\mathbf{B}}$. Príslušná prechodová a výstupná funkcia sú v tabuľke 5. Táto

Tabuľka 5. Tabuľka kódového ekvivalentu z príkladu 4.2

$(y_1, y_2) \setminus (x_1, x_2)$	$(0, 0)$	$(0, 1)$	$(1, 0)$	$(1, 1)$
$(0, 0)$	$(0, 0)/0$	$(1, 0)/0$	$(0, 0)/1$	$(1, 1)/1$
$(1, 0)$	$(1, 0)/1$	$(0, 0)/1$	$(0, 1)/0$	$(0, 1)/0$
$(1, 1)$	$(1, 1)/1$	$(0, 0)/0$	$(1, 0)/1$	$(1, 1)/0$
$(0, 1)$	$(0, 0)/0$	$(1, 0)/1$	$(1, 1)/0$	$(0, 1)/1$

tabuľka pozostáva z tabuliek dvoch budiacich funkcií Y_1, Y_2 a jednej výstupnej funkcie z.

Ich hodnoty sú v tvare $(Y_1, Y_2)/z$. Ak teraz v tejto tabuľke (tab. 5) presunieme druhý stĺpec na posledné miesto, budeme mať v záhlaví zakódované vstupné slová v takom poradí, ktoré je vhodné pre zápis do Karnaughovej mapy. Teraz bude jednoduché zostrojiť Karnaughove mapy pre funkcie Y_1, Y_2 a z. Uvádzame ich na obr. 39. Z nich potom dostávame



Obr. 39. Budiace a výstupné funkcie z príkladu 4.2

$$
\begin{aligned}
Y_1 &= y_1\bar{x}_1\bar{x}_2 + y_1y_2x_1 + y_2x_1\bar{x}_2 + \bar{y}_1\bar{y}_2x_2 + \bar{y}_1\bar{x}_1x_2,\\
Y_2 &= y_1y_2\bar{x}_1\bar{x}_2 + y_1\bar{y}_2x_1 + \bar{y}_1y_2x_1 + x_1x_2,\\
z &= \bar{y}_1\bar{y}_2x_1 + y_1\bar{y}_2\bar{x}_1 + y_1y_2\bar{x}_2 + \bar{y}_1y_2x_2.
\end{aligned}
$$

■

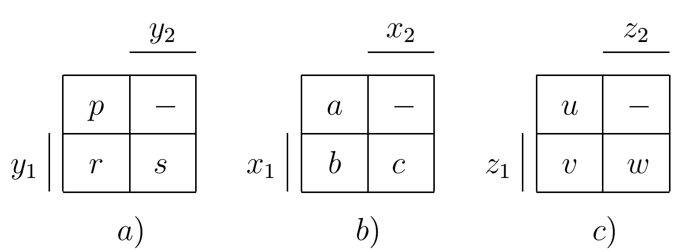
Príklad 4.3. Nech automat $A = (S, X, Z, \delta, \lambda) = (\{p, r, s\}, \{a, b, c\}, \{u, v, w\}, \delta, \lambda)$ je daný pomocou tabuľky 6. Nájdeme dvojkový automat $\breve{A}_{\mathbf{B}}$, ktorý pokrýva automat A.

Riešenie. Stavy, vstupy a výstupy zakódujeme pomocou kódovacích tabuliek na obr. 40,

Tabuľka 6. Automat z príkladu 4.3

	a	b	c
p	p/u	r/u	s/v
r	p/u	s/u	s/w
s	s/u	p/v	r/w

časť a), b), c). Potom kódový ekvivalent $A_{\mathbf{B}}$ automatu A je daný pomocou tabuľky 7.



Obr. 40. Kódovacie tabuľky z príkladu 4.3

Tento automat ešte nie je dvojkový automat, lebo množiny stavov, vstupov a výstupov sa nerovnajú celému $\mathbf{B}^2$. Preto budeme uvažovať o neúplne špecifikovanom automate $\breve{A}_{\mathbf{B}} = (\mathbf{B}^2, \mathbf{B}^2, \mathbf{B}^2, \breve{\delta}_{\mathbf{B}}, \breve{\lambda}_{\mathbf{B}})$, ktorý obsahuje automat $A_{\mathbf{B}}$ ako svoj podautomat. Tento neúplne špecifikovaný automat je daný pomocou tabuľky 8.

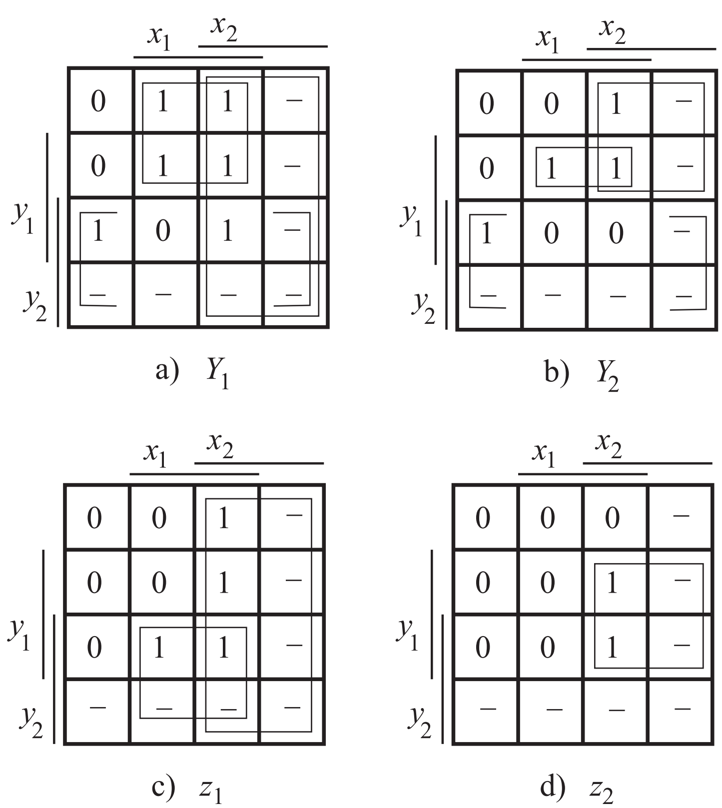
Tabuľka 7. Tabuľka kódového ekvivalentu $A_{\mathbf{B}}$ z príkladu 4.3

	$(0,0)$	$(1,0)$	$(1,1)$
$(0,0)$	$(0,0)/(0,0)$	$(1,0)/(0,0)$	$(1,1)/(1,0)$
$(1,0)$	$(0,0)/(0,0)$	$(1,1)/(0,0)$	$(1,1)/(1,1)$
$(1,1)$	$(1,1)/(0,0)$	$(0,0)/(1,0)$	$(1,0)/(1,1)$

Tabuľka 8. Tabuľka dvojkového automatu $\breve{A}_{\mathbf{B}}$ z príkladu 4.3

(y_1,y_2) \ (x_1,x_2)	$(0,0)$	$(1,0)$	$(1,1)$	$(0,1)$
$(0,0)$	$(0,0)/(0,0)$	$(1,0)/(0,0)$	$(1,1)/(1,0)$	$(-,-)/(-,-)$
$(1,0)$	$(0,0)/(0,0)$	$(1,1)/(0,0)$	$(1,1)/(1,1)$	$(-,-)/(-,-)$
$(1,1)$	$(1,1)/(0,0)$	$(0,0)/(1,0)$	$(1,0)/(1,1)$	$(-,-)/(-,-)$
$(0,1)$	$(-,-)/(-,-)$	$(-,-)/(-,-)$	$(-,-)/(-,-)$	$(-,-)/(-,-)$

V tabuľke tohto automatu sú dané hodnoty $(Y_1,Y_2)/(z_1,z_2)$, kde Y_1,Y_2 sú dve budiace funkcie a z_1,z_2 predstavujú dvojicu výstupných funkcií. Karnaughove mapy týchto funkcií uvádzame na obr. 41. V týchto mapách tie miesta, ktoré sú označené znakom „-“ (pomlčka), môžeme doplniť ľubovoľným spôsobom a vždy dostaneme dvojkový automat,



Obr. 41. Karnaughove mapy prechodových a výstupných funkcií z príkladu 4.3

ktorý pokrýva daný automat A. Toto doplnenie urobíme tak, aby výsledné funkcie mali normálne disjunktívne formy s najmenším možným počtom písmen. Teda tie pomlčky, ktoré budú zahrnuté v nejakej konfigurácii, budú nahradené jednotkami a zvyšné budú nahradené nulami. Ktoré to konkrétne sú, vidno z obr. 41. Pri takejto dohode dostávame:

$$\begin{aligned}
Y_1 &= x_2 + \bar{y}_2 x_1 + y_2 \bar{x}_1,\\
Y_2 &= \bar{y}_2 x_2 + y_1 \bar{y}_2 x_1 + y_2 \bar{x}_1,\\
z_1 &= x_2 + y_2 x_1,\\
z_2 &= y_1 x_2.
\end{aligned}$$

Tým je však určený dvojkový automat $\tilde{A}_{\mathbf{B}}$ pokrývajúci automat A, ktorý sme podľa zadania hľadali. ■

Teraz každý automat vieme pokryť (podľa vzoru predchádzajúceho príkladu) neúplne špecifikovaným dvojkovým automatom. Tento neúplne špecifikovaný dvojkový automat môžeme rôznymi spôsobmi doplniť na dvojkový automat. Kvôli úspornosti je vhodné vybrať ten z nich, ktorého budiace a výstupné funkcie sú zapísané normálnymi disjunktívnymi formami v minimálnom možnom tvare. Samozrejme, v zásade je možné vybrať ľubovoľný z nich. Ako fyzikálnu realizáciu priradíme tomuto vybranému automatu logický obvod, ktorý budeme reprezentovať pomocou sekvenčnej logickej siete. Jej základnú schému (zdôraznime, že pre najjednoduchší prípad), sme uviedli na obr. 2. Kombinačnou časťou tejto siete je kombinačná logická sieť, ktorú už vieme zostrojovať. Pamäťovú časť budeme zostrojovať pomocou preklápacích obvodov, pričom každej budiacej funkcii Y_i odpovedá jeden preklápací obvod, ktorý túto funkciu generuje.

V prípade, že fyzikálnu realizáciu budeme robiť pomocou synchrónnych logických obvodov, body diskrétneho času nám určuje generátor časových impulzov. Hlavným problémom je len určiť šírku impulzu a dĺžku periódy, ktorú generuje generátor časových impulzov. Tieto problémy sú podrobne riešené v práci [**3**].

V nasledujúcich dvoch príkladoch ukážeme návrh logických sietí, ktoré reprezentujú fyzikálne realizácie automatov z príkladov 4.2 a 4.3. Hoci to nie je zvykom, budeme každú z budiacich funkcií generovať pomocou iného preklápacieho obvodu, aby sme na týchto dvoch príkladoch mohli demonštrovať použitie všetkých štyroch typov preklápacích obvodov.

Príklad 4.4. Navrhneme sekvenčnú logickú sieť, ktorá reprezentuje fyzikálnu realizáciu automatu $\tilde{A}_{\mathbf{B}} = A_{\mathbf{B}}$ z príkladu 4.2, pričom budiacu funkciu Y_1 budeme generovať pomocou synchrónneho SR - preklápacieho obvodu a funkciu Y_2 pomocou synchrónneho D - preklápacieho obvodu. Rozmiestnenie vrcholov a štruktúru pamäťovej časti hľadanej sekvenčnej logickej siete uvádzame na obr. 43.

Riešenie. Všimnime si, že táto logická sieť má komplikovanejšiu štruktúru ako na obr. 2. Napríklad SR - preklápací obvod generuje budiacu funkciu Y_1, ale tá na rozdiel od obr. 2 nie je vstupnou funkciou tohoto obvodu. Obvod má dve vstupné funkcie S a R, ktoré budeme musieť určiť. Samotná funkcia Y_1 sa generuje až vo vnútri SR - preklápacieho obvodu a do ďalšieho taktu sa prenáša ako y_1.

Teraz teda opíšeme vstupné funkcie (teda S, R a D) jednotlivých preklápacích obvodov vzhľadom na budiace funkcie, ktoré tieto preklápacie obvody majú generovať.

Uvažujme najprv o budiacej funkcii Y_1. Jej Karnaughova mapa (v ďalšom kvôli stručnosti len tabuľka) je na obr. 39, časť a), na obr. 42, časť a) ju opakujeme. V tejto tabuľke je pre nás dôležitá zmena starého stavu y_1 na nový stav Y_1 v závislosti od ďalších premenných y_2, x_1, x_2. Ako už bolo povedané, táto zmena sa má realizovať v SR - preklápacom

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

a) Y_1

0	0	1	1
x	0	0	0
x	x	x	0
0	1	0	1

b) S

x	x	0	0
0	1	1	1
0	0	0	1
x	0	x	0

c) R

Obr. 42. Funkcia Y_1 a vstupné funkcie SR - preklápacieho obvodu z príkladu 4.4

obvode. Aby táto zmena prebiehala presne podľa tabuľky na obr. 42, časť a), potrebujeme vhodne určiť vstupné funkcie S a R SR - preklápacieho obvodu v závislosti od vstupných premenných y_1, y_2, x_1, x_2. Hodnoty týchto funkcií uvádzame na obr. 42, časť b), c). Tieto tabuľky sú zatiaľ neúplné, namiesto niektorých hodnôt je znak $\times$ a tieto hodnoty doplníme neskôr. Pri zostavovaní týchto tabuliek sme vychádzali jednak z tabuľky funkcie Y_1 (obr. 42, časť a)) a jednak z tabuľky 2, ktorú sme uviedli pri definícii SR - preklápacieho obvodu. Teraz tento postup opíšeme.

Začnime zostavením tabuľky pre funkciu S. Najprv uvažujme o zmene $0 \to Y_1$. Z tabuľky 2 čítame, že v tom prípade je $S = Y_1$. Preto prvý a štvrtý riadok z tabuľky funkcie Y_1 prepíšeme do tabuľky funkcie S. Pri zmene $1 \to Y_1$ platí nasledujúce. Ak $Y_1 = 0$, potom $S = 0$. Ak $Y_1 = 1$, potom S je ľubovoľné (píšeme $S = \times$). Toto všetko sme opäť dostali z tabuľky 2. Preto v druhom a treťom riadku tabuľky funkcie S ponecháme všetky nuly z druhého a tretieho riadku tabuľky Y_1 a jednotky nahradíme znakom $\times$, čo znamená, že tieto miesta môžeme ľubovoľne doplniť.

Podobne postupujeme pri písaní tabuľky funkcie R.

a) Pri zmene $1 \to Y_1$ (druhý a tretí riadok tabuľky funkcie Y_1) je $R = \bar{Y}_1$. Teda v druhom a treťom riadku tabuľky meníme 0 na 1 a 1 na 0.

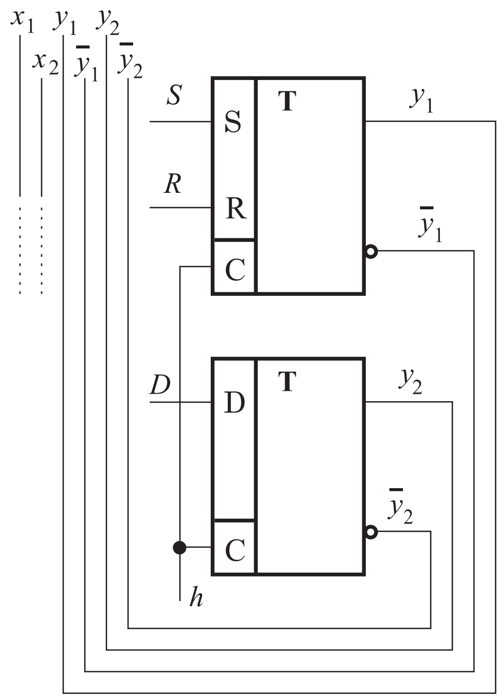
b) Pri zmene $0 \to Y_1$ (prvý a štvrtý riadok tabuľky funkcie Y_1) ak $Y_1 = 1$, potom $R = 0$. Ak $Y_1 = 0$, R je ľubovoľné. To znamená, že v prvom a štvrtom riadku tabuľky funkcie Y_1 všetky jednotky zameníme na hodnotu 0 a na miestach, kde bola 0, môžeme voliť ľubovoľnú hodnotu, čiže píšeme $\times$.

Z Karnaughových máp funkcií S, R čítame normálne disjunktívne formy týchto funkcií, pričom sa usilujeme o ich minimalizáciu. Rozhodneme teda, ktoré znaky $\times$ nahradíme jednotkami a ktoré nulami, aby výsledná normálna disjunktívna forma mala najmenší možný počet písmen. Z obr. 42, časť b), c) vidno, ako túto voľbu urobiť. V tabuľke funkcie S nahradíme jediný znak $\times$ jednotkou a ostatné nulami, v tabuľke funkcie R všetky znaky $\times$ nahradíme nulami. Potom dostávame

$$
\begin{aligned}
S &= \bar{y}_1\bar{y}_2x_2 + \bar{y}_1\bar{x}_1x_2 + y_2x_1\bar{x}_2 \\
R &= y_1\bar{y}_2x_1 + y_1\bar{x}_1x_2.
\end{aligned}
$$

Budiacu funkciu Y_2 budeme generovať pomocou D - preklápacieho obvodu. Nakoľko D - preklápací obvod je oneskorovací člen, situácia je veľmi jednoduchá, lebo stačí položiť

$$D = Y_2 = y_1y_2\bar{x}_1\bar{x}_2 + y_1\bar{y}_2x_1 + \bar{y}_1y_2x_1 + x_1x_2.$$



Obr. 43. Základné vrcholy sekvenčnej logickej siete z príkladu 4.4

(Pripomeňme na tomto mieste, že ak by celá pamäťová časť logickej siete bola zostavená len z D - preklápacích obvodov, potom by realizovala jednoduché oneskorenie o jeden takt a v plnej miere by platila schéma 2.)

Napokon spomeňme, že v tomto príklade je výstupná funkcia

$$z = \bar{y}_1\bar{y}_2x_1 + y_1\bar{y}_2\bar{x}_1 + y_1y_2\bar{x}_2 + \bar{y}_1y_2x_2.$$

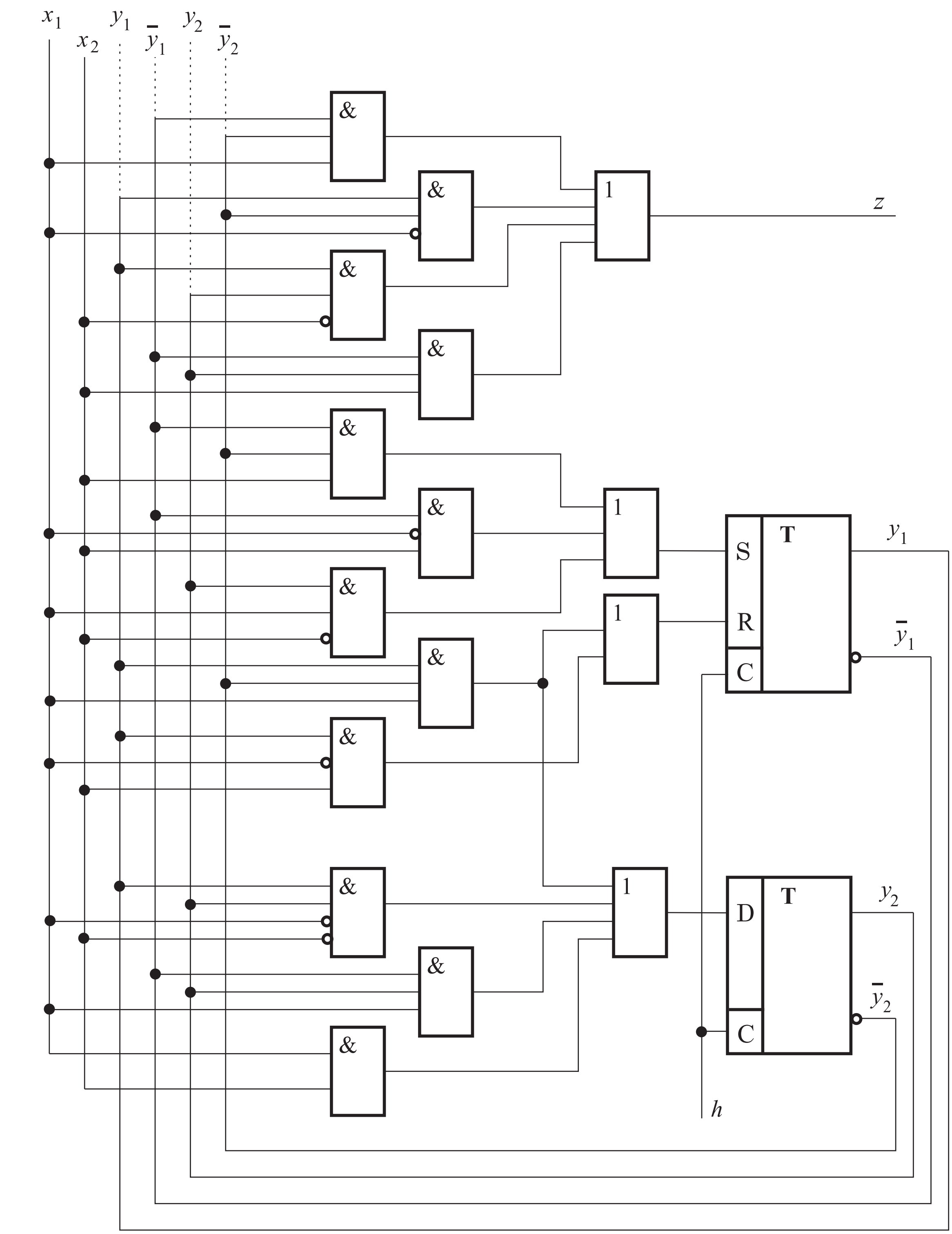
Teraz už len treba doplniť graf na obr. 43 pomocou funkcií S, R, D, z, ktorých normálne disjunktívne formy sme práve uviedli. Príslušnú sekvenčnú logickú sieť uvádzame na obr. 44. ■

Príklad 4.5. Teraz navrhneme sekvenčnú logickú sieť, ktorá reprezentuje fyzikálnu realizáciu dvojkového automatu $\tilde{A}_{\mathbf{B}}$ z príkladu 4.3. Budiacu funkciu Y_1 budeme generovať pomocou synchrónneho JK - preklápacieho obvodu a funkciu Y_2 pomocou synchrónneho T - preklápacieho obvodu. Základné vrcholy a štruktúru pamäťovej časti tejto siete uvádzame na obr. 47.

Riešenie. Pri navrhovaní vstupných funkcií J a K JK - preklápacieho obvodu využijeme Karnaughovu mapu (v ďalšom len tabuľku) funkcie Y_1 z obr. 41, časť a). Pravda, tam je uvedená ešte pred záverečným doplnením. Teraz na obr. 45, časť a) uvádzame už doplnenú tabuľku funkcie Y_1. Ďalej v našom postupe využijeme tabuľku 3.

Z tabuľky 3 vidieť, že pri zmene $0 \to Y_1$ je $J = Y_1$. Teda pri tvorbe tabuľky funkcie J treba z tabuľky funkcie Y_1 prevziať celý prvý a štvrtý riadok, pri ktorých nastáva zmena $0 \to Y_1$. Pri zmene $1 \to Y_1$ môže byť vstup J ľubovoľný. Preto v druhom a treťom riadku tabuľky funkcie J píšeme $\times$ (pozri obr. 45, časť b)). Teraz zostrojíme tabuľku funkcie K. Pri zmene $1 \to Y_1$ je $K = \bar{Y}_1$. Preto pri navrhovaní tabuľky funkcie K v druhom a treťom riadku tabuľky funkcie Y_1 preklápame 0 na 1 a 1 na 0. Pri zmene $0 \to Y_1$ je $K = \times$. To sa prejaví v prvom a v štvrtom riadku tabuľky funkcie K (pozri obr. 45, časť c)).

Teraz krížiky v tabuľkách funkcií J, K na obr. 45, časť b), c) nahradíme nulami a jednotkami tak, aby výsledné funkcie mali normálne disjunktívne formy s najmenším

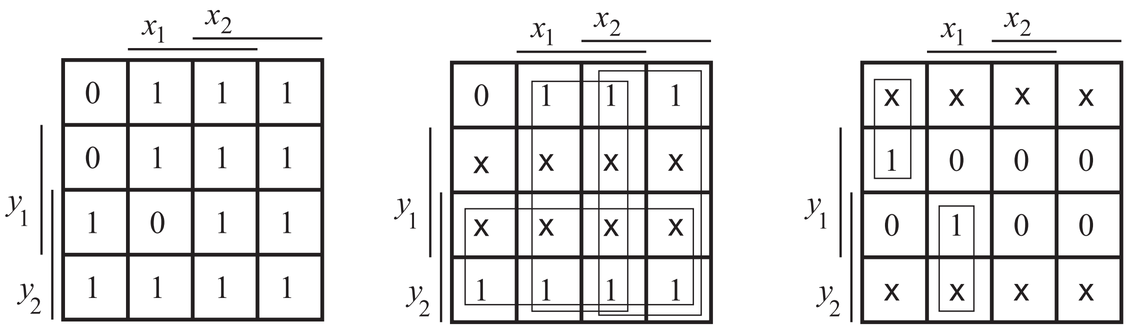


Obr. 44. Logická sieť z príkladu 4.4

možným počtom písmen. Z obr. 45, časť b), c) plynie, že riešením je

$$J = x_1 + x_2 + y_2$$
$$K = \bar{y}_2\bar{x}_1\bar{x}_2 + y_2x_1\bar{x}_2.$$

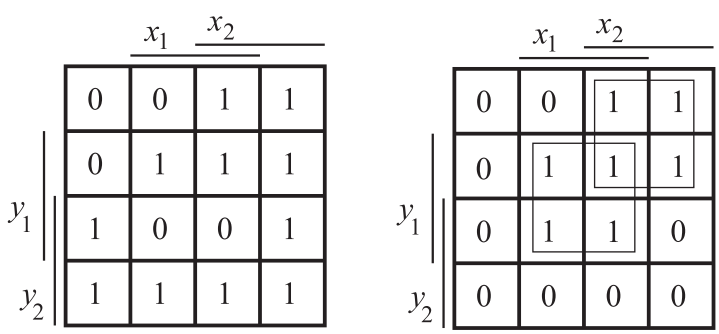
Funkciu Y_2 budeme generovať pomocou T - preklápacieho obvodu, teda musíme navrhnúť jeho vstupnú funkciu T. Pritom využijeme doplnenú tabuľku funkcie Y_2 z obr. 41,



a) Y_1 b) J c) K

Obr. 45. Funkcia Y_1 a vstupné funkcie JK - preklápacieho obvodu z príkladu 4.5

časť b) (uvádzame ju na obr. 46, časť a)) a zjednodušenú tabuľku pre vstupnú funkciu



a) Y_2 b) T

Obr. 46. Funkcia Y_2 a vstupná funkcia T - preklápacieho obvodu z príkladu 4.5

T T - preklápacieho obvodu z obr. 22. Z tejto tabuľky vyplýva, že pri zmene $0 \to Y_2$ je $T = Y_2$. To znamená, že pri tvorbe tabuľky funkcie T prepisujeme prvý a druhý riadok z tabuľky funkcie Y_2 (v tom prípade totiž tieto dva riadky zodpovedajú zmene $0 \to Y_2$). Pri zmene $1 \to Y_2$ je $T = \bar{Y}_2$. Preto tretí a štvrtý riadok tabuľky funkcie T získame tak, že v treťom a štvrtom riadku tabuľky funkcie Y_2 preklopíme 0 na 1 a 1 na 0. Tabuľku funkcie T uvádzame na obr. 46, časť b).

Z tabuľky funkcie T potom dostávame

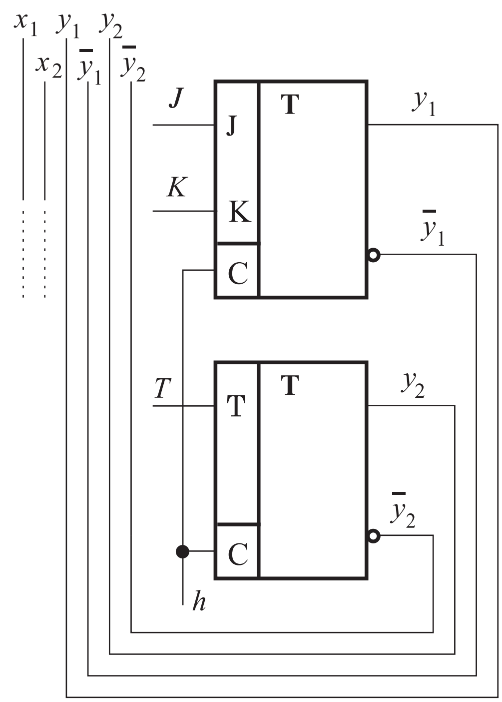
$$T = \bar{y}_2 x_2 + y_1 x_1.$$

Spolu s výstupnými funkciami

$$z_1 = x_2 + y_2 x_1$$
$$z_2 = y_1 x_2$$

už dostávame všetky hodnoty, aby sme mohli navrhnúť príslušnú sekvenčnú logickú sieť. Tá je zostrojená na obr. 48. ■

Pri fyzikálnej realizácii automatu pomocou asynchrónnych logických obvodov narážame na pomerne veľké ťažkosti. Tie spočívajú v tom, že reakcia asynchrónneho logického obvodu nezáleží len na vstupnom vektore logického obvodu, ale aj na režime práce tohto obvodu. Podrobnejšie sa môže čitateľ s touto problematikou zoznámiť v práci [**3**]. My sa budeme zaoberať iba dvomi spôsobmi režimu práce asynchrónneho logického obvodu, pričom problémy, ktoré sú s tým spojené, vyložíme predovšetkým na príkladoch.



Obr. 47. Základné vrcholy logickej siete z príkladu 4.5

A) Fundamentálny režim práce asynchrónneho logického obvodu – Fundamentálny automat

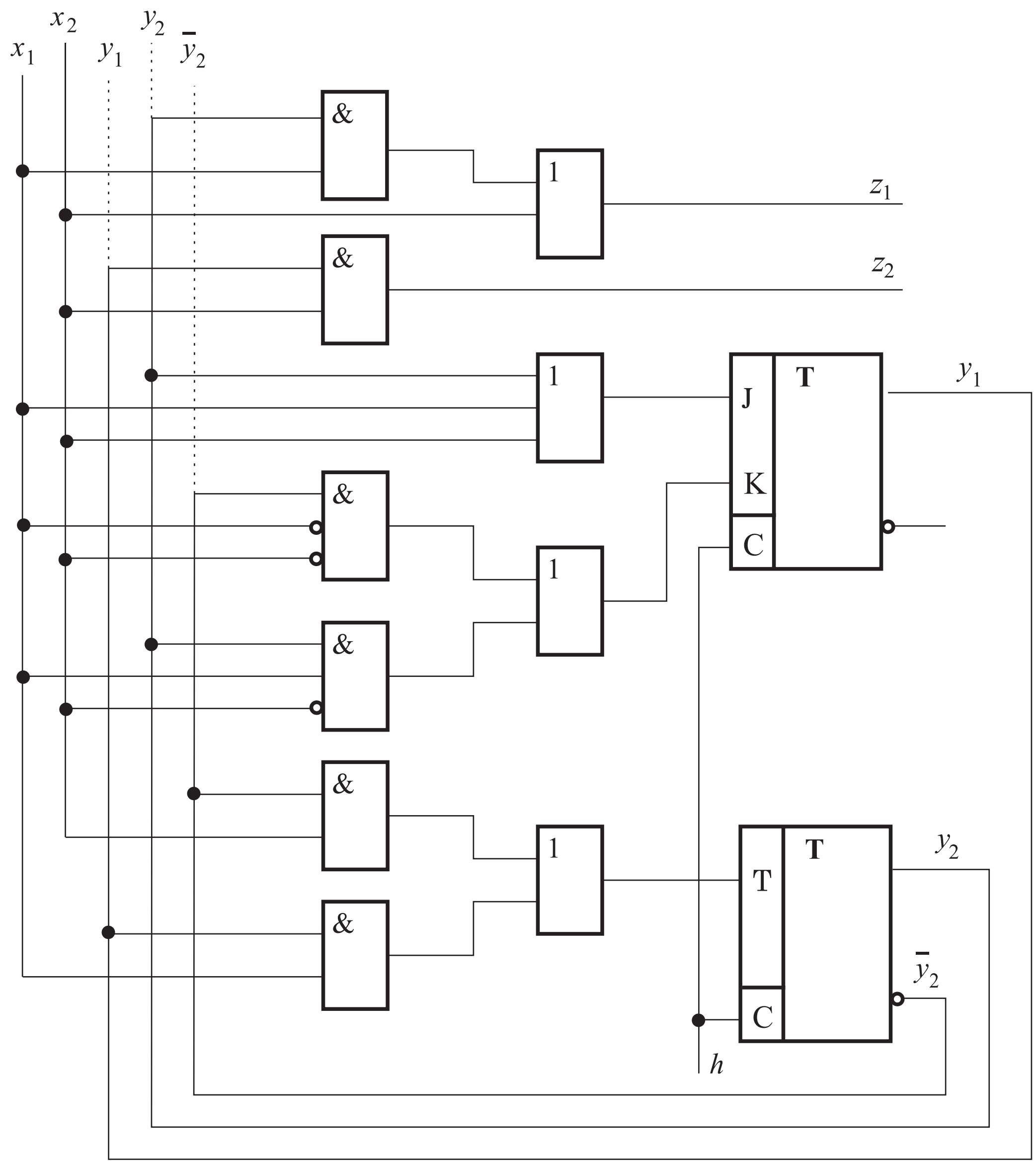
Nejednoznačnosti pri činnosti asynchrónneho logického obvodu vznikajú najmä vtedy, keď pri tomto obvode meníme vstupné veličiny bez toho, aby stav tohto obvodu bol ustálený (stav sa nemenil). Asynchrónny logický obvod budeme nazývať ***fundamentálny***, ak pri každom pevnom rozložení logických veličín na vstupoch tohto obvodu sa stav obvodu po istom konečnom čase ustáli a viac sa nemení. Ďalej budeme hovoriť, že tento ***fundamentálny logický obvod pracuje vo fundamentálnom režime***, ak vstupný vektor zmeníme iba v tom prípade, keď sa už obvod nachádza v ustálenom stave (stav sa už nemení).

Nie všetky automaty budeme môcť fyzikálne realizovať pomocou fundamentálneho asynchrónneho logického obvodu. Tie automaty, ktoré sa dajú pomocou týchto obvodov fyzikálne realizovať, teraz bližšie opíšeme.

Nech $A = (S, X, Z, \delta, \lambda)$ je automat. Nech $s \in S$ a $x \in X$. Budeme hovoriť, že ***stav*** s ***je stabilný pri vstupe*** x, ak $\delta(s, x) = s$. Táto skutočnosť sa v tabuľke prechodovej funkcie prejaví tak, že v riadku patriacom stavu s a v stĺpci patriacom vstupu x sa nachádza stav s. Tento stav budeme v tabuľke automatu vyznačovať rámčekom. Pre jednoduchšie vyjadrovanie sa ešte dohodneme, že dvojicu (s, x) budeme nazývať ***celkový stav automatu*** a budeme hovoriť, že tento celkový stav je stabilný, ak $\delta(s, x) = s$.

Automat $A = (S, X, Z, \delta, \lambda)$ sa bude dať realizovať pomocou fundamentálneho asynchrónneho logického obvodu, ak bude mať takú vlastnosť, že pri opakovanom nasadzovaní hociktorého vstupného písmena na vstup automatu sa v konečnom čase dostane do stabilného stavu. To je obsahom nasledujúcej definície.

Definícia 4.1. Automat $A = (S, X, Z, \delta, \lambda)$ nazývame ***fundamentálny*** alebo ***Huffmannov automat***, ak pre každé $s \in S$ a každé $x \in X$ existuje $n > 0$ také, že



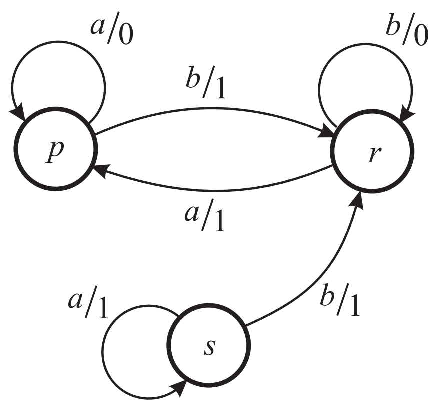
Obr. 48. logická sieť z príkladu 4.5

$\hat{\delta}(s, x^n) = \hat{\delta}(s, x^{n+1})$ (v tomto prípade x^n označuje vstupné slovo $xx \ldots x$, v ktorom sa nachádza n písmen x).

Tabuľka 9. Tabuľka automatu z príkladu 4.6

	a	b	a	b
p	$\boxed{p}$	r	0	1
r	p	$\boxed{r}$	1	0
s	$\boxed{s}$	r	1	1

Príklad 4.6. Nech $A = (S, X, Z, \delta, \lambda) = (\{p, r, s\}, \{a, b\}, \{0, 1\}, \delta, \lambda)$ je automat, ktorý je daný pomocou tabuľky 9. Graf tohto automatu uvádzame na obr. 49. Teraz ukážeme, že daný automat je fundamentálny.



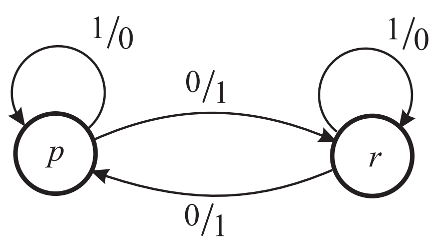
Obr. 49. Graf automatu z príkladu 4.6

Riešenie. Dosť ľahko overíme, že

$$\begin{aligned}
&\hat{\delta}(p, a) = \hat{\delta}(p, a^2) = p,\\
&\hat{\delta}(p, b) = \hat{\delta}(p, b^2) = r,\\
&\hat{\delta}(r, a) = \hat{\delta}(r, a^2) = p,\\
&\hat{\delta}(r, b) = \hat{\delta}(r, b^2) = r,\\
&\hat{\delta}(s, a) = \hat{\delta}(s, a^2) = s,\\
&\hat{\delta}(s, b) = \hat{\delta}(s, b^2) = r.
\end{aligned}$$

Z toho už vyplýva, že daný automat je fundamentálny. ■

Príklad 4.7. Automat A je daný grafom na obr. 50. Tento automat nie je fundamentálny, lebo $\delta(p, 0) = r$ a $\delta(r, 0) = p$. Je zrejmé, že v tomto prípade je vždy $\hat{\delta}(p, 0^n) \neq \hat{\delta}(p, 0^{n+1})$. ■



Obr. 50. Graf automatu z príkladu 4.7

Ak teda fundamentálny automat budeme fyzikálne realizovať pomocou fundamentálneho asynchrónneho logického obvodu pracujúceho vo fundamentálnom režime, tento obvod pri pevnom vstupnom vektore môže prejsť do stabilného stavu buď nepriamo cez nestabilné stavy, alebo priamo.

Budeme sa zaoberať fyzikálnou realizáciou fundamentálnych dvojkových automatov. Pri asynchrónnej realizácii dvojkového automatu sa stretávame s týmto problémom. Stavy tohto automatu sú k-tice $(y_1, \dots, y_k) \in \mathbf{B}^k$. Nech pri danom vstupnom vektore $x = (x_1, \dots, x_n)$ je $\delta((y_1, \dots, y_k), x) = (\tilde{y}_1, \dots, \tilde{y}_k)$ a vektory $(y_1, \dots, y_k)$, $(\tilde{y}_1, \dots, \tilde{y}_k)$ sa líšia viac ako na jednom mieste. Nech napríklad $y_i \neq \tilde{y}_i$ a $y_j \neq \tilde{y}_j$. Prechody $y_i \longrightarrow \tilde{y}_i$ a

$y_j \longrightarrow \tilde{y}_j$ sa vo fyzikálnej realizácii tohto automatu sa nikdy neuskutočnia súčasne, lebo fyzikálne elementy, ktoré tieto prechody uskutočňujú, nemajú úplne identické vlastnosti. Z toho dôvodu sa logický obvod na prechodný čas dostáva do stavu, v ktorom už je y_i nahradené hodnotou $\tilde{y}_i$ a y_j ešte nie je alebo naopak. V tomto prípade hovoríme, že dochádza k ***súbehovému prechodu v množine stavov*** (k ***súbehu***). Ak ani jeden z uvažovaných (prechodových) stavov nie je stabilný, môžu sa ešte zvyšné zložky dostať na požadované hodnoty. Ak však stav, v ktorom iba niektoré zložky nadobudli vyžadované hodnoty, je pri vstupe x stabilný, zvyšné zložky sa už nedostanú na požadované hodnoty a stav logického obvodu nebude zodpovedať stavu automatu, ktorý je týmto obvodom realizovaný. V takomto prípade hovoríme o ***kritickom súbehu***. Kritické súbehy je možné odstrániť buď prechodom k inému dvojkovému automatu pomocou zmeny kódovacej funkcie množiny stavov, alebo tzv. organizovaním nekritických prechodov medzi stavmi. Pri organizovaní nekritických prechodov však zaplatíme zväčšenou dĺžkou kódovacích slov, a teda pridaním budiacich funkcií. To má za následok zvýšenie počtu preklápacích obvodov, a teda zdraženie fyzikálnej realizácie. Problém súbehov a organizácie nekritických prechodov medzi stavmi ukážeme na príklade. Podrobnejšia zmienka o tejto problematike je v práci [**3**].

Príklad 4.8. Uvažujme o prechodovej funkcii neúplne špecifikovaného dvojkového automatu, ktorá je daná v tabuľke 10.

Z tabuľky prechodovej funkcie vyplýva, že pri vstupe $(0,0)$ nastáva súbeh v stave $(0,0)$. Pri vstupe $(1,0)$ k súbehu dochádza v stave $(0,1)$. Pri vstupe $(1,1)$ súbeh nastáva

Tabuľka 10. Prechodová funkcia automatu z príkladu 4.8

	$(0,0)$	$(1,0)$	$(1,1)$	$(0,1)$
$(0,0)$	$(1,1)$	$\boxed{(0,0)}$	$\boxed{(0,0)}$	$(0,1)$
$(1,0)$	$(1,1)$	$\boxed{(1,0)}$	–	$\boxed{(1,0)}$
$(1,1)$	$\boxed{(1,1)}$	$(1,0)$	$(0,0)$	$(1,0)$
$(0,1)$	$\boxed{(0,1)}$	$(1,0)$	$(0,0)$	$\boxed{(0,1)}$

v stave $(1,1)$. Pri vstupe $(0,1)$ k súbehom nedochádza. Teraz jednotlivé súbehy budeme analyzovať.

1. Pri vstupe $(0,0)$ máme $\delta((0,0),(0,0)) = (1,1)$. Takže pri vstupe $(0,0)$, ak počítame s nekoordinovaným preklopením stavu $(0,0)$ na stav $(1,1)$, automat sa môže (na prechodný čas) dostať buď do stavu $(0,1)$ (ak sa skorej zmenila druhá zložka stavu) alebo do stavu $(1,0)$ (ak sa skôr zmenila prvá zložka stavu). Pri pevnom vstupe $(0,0)$ sa stav $(1,0)$ zmení na požadovaný stav, lebo $\delta((1,0),(0,0)) = (1,1)$, avšak $\delta((0,1),(0,0)) = (0,1)$. Teda stav $(0,1)$ je pri vstupe $(0,0)$ stabilný, a tak v prípade, že sa najskôr zmenila druhá zložka stavu $(0,0)$, pri pevnom vstupe $(0,0)$ zostane automat v stabilnom stave $(0,1)$, ktorý sa už nezmení na požadovaný stav. Teda tento súbeh je kritický.

2. Keďže $\delta((0,1),(1,0)) = (1,0)$, dochádza pri vstupe $(1,0)$ k súbehu $(0,1) \longrightarrow (1,0)$. Stav $(0,1)$ môžeme zmeniť na stav $(1,0)$ buď cez stav $(1,1)$, ktorý pri pevnom vstupe $(1,0)$ sa zmení na požadovaný stav $(1,0)$, alebo cez stav $(0,0)$, ktorý je pri vstupe $(1,0)$ stabilný. Zas ide o kritický súbeh. Pokiaľ by šlo o Moorov automat a nás by zaujímal iba výsledný riadiaci povel patriaci k stabilnému stavu $(1,0)$, mohli by sme využiť fakt, že automat pri danom vstupe $(1,0)$ prechádza do stavu $(1,0)$ nielen zo stavu $(0,1)$ ale aj zo stavu $(1,1)$. Preto ak v definícii automatu zmeníme hodnotu $\delta((0,1),(1,0)) = (1,0)$ na

hodnotu $\delta((0,1),(1,0)) = (1,1)$, už k súbehu nedochádza a pri pevnom vstupe $(1,0)$ sa nakoniec zo stavu $(0,1)$ dostaneme do stavu $(1,0)$. V tomto prípade by sme totiž mali $\hat{\delta}((0,1),(1,0)^2) = \delta(\delta((0,1),(1,0)),(1,0)) = \delta((1,1),(1,0)) = (1,0)$. Takémuto postupu sa hovorí ***organizácia postupného prechodu medzi stavmi***.

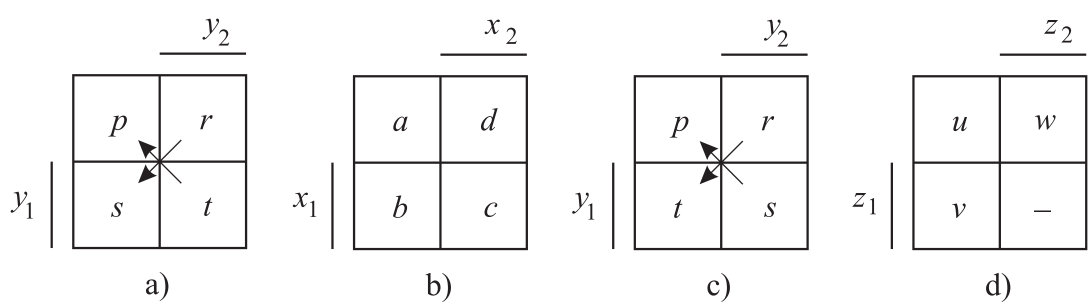
3. Pri vstupe $(1,1)$ a stave $(1,1)$ tiež nastáva súbeh, lebo $\delta((1,1),(1,1)) = (0,0)$. V tomto prípade však stačí nedefinovanú hodnotu $\delta((1,0),(1,1))$ nahradiť hodnotou $(0,0)$. Potom pri vstupe $(1,1)$ sa všetky stavy dostávajú do stabilného stavu $(0,0)$. V takomto prípade hovoríme o ***nekritickom súbehu***. Nekritické súbehy spravidla nemusíme odstraňovať. ■

Príklad 4.9. Navrhneme fyzikálnu realizáciu neúplne špecifikovaného Moorovho automatu, ktorý je daný pomocou tabuľky 11.

Tabuľka 11. Automat z príkladu 4.9

	a	b	c	d	μ
p	$\boxed{\text{p}}$	r	–	$\boxed{\text{p}}$	u
r	s	$\boxed{\text{r}}$	t	–	u
s	$\boxed{\text{s}}$	–	–	p	v
t	–	–	$\boxed{\text{t}}$	p	w

Riešenie. Stavy zakódujeme pomocou kódovacej tabuľky na obr. 51 časť a). Touto tabuľkou je daná kódovacia funkcia $f : \{p,r,s,t\} \to \mathbf{B}^2$, kde $f(p) = (0,0)$, $f(r) = (0,1)$, $f(s) = (1,0)$, $f(t) = (1,1)$. Na obr. 51 časť b) je kódovacia tabuľka vstupnej abecedy. Tabuľku prechodovej funkcie dvojkového automatu $A_{\mathbf{B}}$, ktorý pokrýva daný automat, uvádzame v tabuľke 12. Z tejto tabuľky vidíme, že k súbehu dochádza pri vstupe $(0,0)$ pri prechode zo stavu $(0,1)$ do stavu $(1,0)$, lebo $\delta_{\mathbf{B}}((0,1),(0,0)) = (1,0)$. V tomto prípade budeme písať $(0,0) : (0,1) \longrightarrow (1,0)$. Ďalší súbeh má tvar $(0,1) : (1,1) \longrightarrow (0,0)$. Teraz sa budeme zaoberať otázkou, či tieto súbehy sú kritické.



Obr. 51. Kódovacie tabuľky z príkladu 4.9

Zaoberajme sa najprv súbehom $(0,1) : (1,1) \longrightarrow (0,0)$, ktorý má nasledujúci priebeh. Pri pevnom vstupe $(0,1)$ sa v stave $(1,1)$ buď najskôr zmení druhá zložka a automat sa dostane do stavu $(1,0)$, ktorý pri pevnom vstupe $(0,1)$ prechádza do požadovaného stavu $(0,0)$, alebo sa v stave $(1,1)$ najskôr zmení prvá zložka a automat sa dostane do stavu $(0,1)$. V tomto stave pri vstupe $(0,1)$ nie je definovaná ďalšia činnosť automatu. Preto dodefinujeme hodnotu $\delta_{\mathbf{B}}((0,1),(0,1)) = (0,0)$ a tým zabezpečíme prechod do

Tabuľka 12. Tabuľka prechodovej funkcie automatu z príkladu 4.9

	$(0,0)$	$(1,0)$	$(1,1)$	$(0,1)$
$(0,0)$	$\boxed{(0{,}0)}$	$(0,1)$	$-$	$\boxed{(0{,}0)}$
$(0,1)$	$(1,0)$	$\boxed{(0{,}1)}$	$(1,1)$	$-$
$(1,0)$	$\boxed{(1{,}0)}$	$-$	$-$	$(0,0)$
$(1,1)$	$-$	$-$	$\boxed{(1{,}1)}$	$(0,0)$

požadovaného stavu. Tento súbeh nie je kritický. Uvedený postup budeme znázorňovať takto:

$$\begin{tikzcd} & (0,1) \arrow[dr, dashed] & \\ (0,1):(1,1) \arrow[ur] \arrow[dr] & & (0,0) \\ & (1,0) \arrow[ur] & \end{tikzcd}$$

V tomto diagrame čiarkovaná šípka označuje, že v dvojkovom automate sme dodefinovali pri vstupe $(0,1)$ prechod zo stavu $(0,1)$ do stavu $(0,0)$.

Teraz sledujme druhý súbeh $(0,0):(0,1) \longrightarrow (1,0)$, ktorý má tento priebeh:

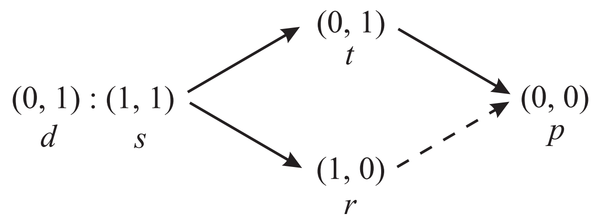
$$\begin{tikzcd} & (0,0) \arrow[loop right] & \\ (0,0):(0,1) \arrow[ur] \arrow[dr] & & (1,0) \\ & (1,1) \arrow[ur, dashed] & \end{tikzcd}$$

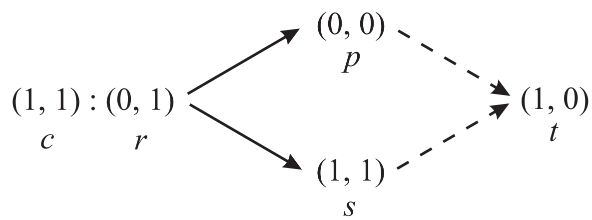
Vidíme, že zo stavu $(1,1)$ môžeme pri vstupe $(0,0)$ dodefinovať prechod do požadovaného stavu $(1,0)$. Ale stav $(0,0)$ je pri vstupe $(0,0)$ stabilný. Pri pevnom vstupe $(0,0)$ sa automat zo stavu $(0,0)$ už nedostane, čiže nemáme zabezbečený požadovaný prechod do stavu $(1,0)$. Preto je tento súbeh kritický.

Tento kritický súbeh môžeme ešte odstrániť tým, že zmeníme kódovanie množiny stavov. Treba si uvedomiť, že v dvojkovom automate, ktorý pokrýva daný automat, súbehy nastávajú práve vtedy, keď v pôvodnom automate existuje prechod medzi stavmi, ktoré sa nachádzajú na uhlopriečke kódovacej tabuľky. Tieto existujúce prechody sme v kódovacej tabuľke na obr. 51, časť a) vyznačili šípkami. Pritom prechod $r \longrightarrow s$ viedol pri danom kóde ku kritickému súbehu $(0,1) \longrightarrow (1,0)$. Aby sme tento súbeh odstránili, musia sa stavy r, s v kódovacej tabuľke nachádzať vedľa seba. Použijeme kódovanie množiny stavov pomocou kódovacej tabuľky na obr. 51, časť c) (systematický postup pre voľbu jednotlivých kódovacích funkcií množiny stavov je uvedený v práci [**3**]). Tejto kódovacej tabuľke bude zodpovedať nový dvojkový automat, ktorý pokrýva pôvodný automat. V tomto dvojkovom automate ponecháme pôvodné zakódovanie vstupnej abecedy. Vidíme, že súbehy, ktoré sa vyskytovali v predchádzajúcom dvojkovom automate, sme odstránili, ale objavili sa nové, pretože $\delta(s,d) = p$, $\delta(r,c) = t$ a dvojice stavov s, p a r, t sa nachádzajú na uhlopriečkach novej kódovacej tabuľky. Preto sme prechody $s \longrightarrow p$ a $r \longrightarrow t$ vyznačili v kódovacej tabuľke na obr. 51, časť c). Týmto prechodom v kódovom ekvivalente zodpovedajú súbehy:

$$(0,1):(1,1) \longrightarrow (0,0) \qquad (d : s \longrightarrow p \text{ v pôvodnom automate}),$$
$$(1,1):(0,1) \longrightarrow (1,0) \qquad (c : r \longrightarrow t \text{ v pôvodnom automate}).$$

Tieto súbehy majú tento priebeh:





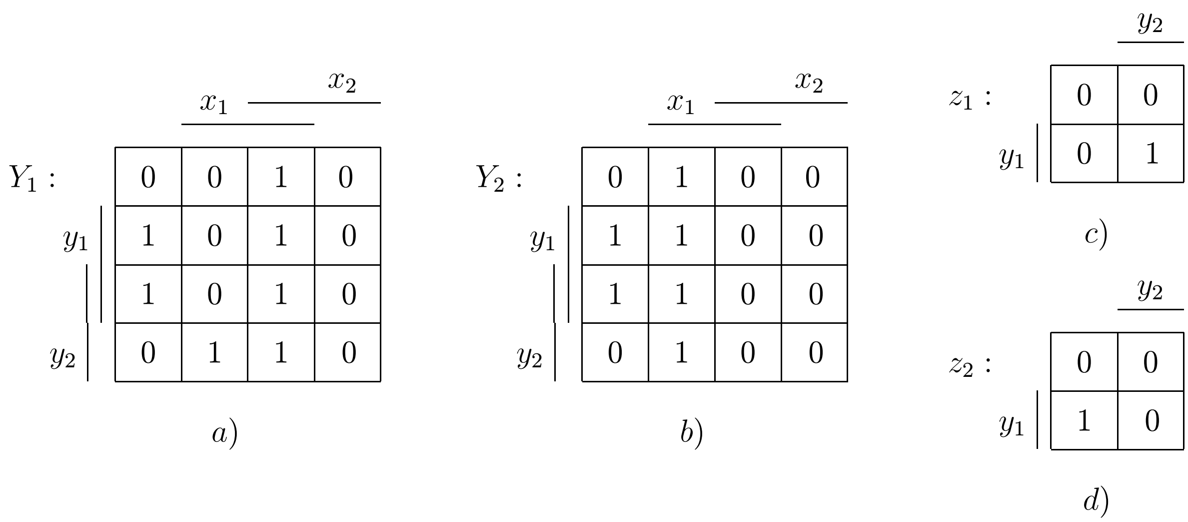
Vidíme, že máme možnosť dodefinovať tri prechody medzi stavmi tak, aby ani jeden zo súbehov nebol kritický. V tabuľke 13 uvádzame hodnoty prechodovej a výstupnej funkcie dvojkového automatu, ktorý sme získali z pôvodného automatu pomocou kódovacích

TABUĽKA 13. Tabuľka automatu z príkladu 4.9

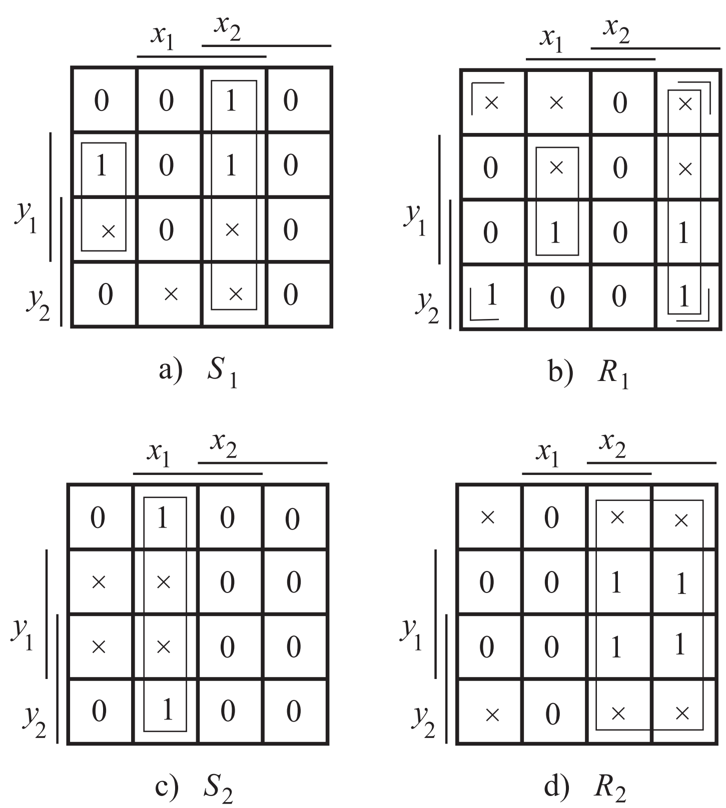
$(y_1, y_2) \backslash (x_1, x_2)$	$(0,0)$	$(1,0)$	$(1,1)$	$(0,1)$	(z_1, z_2)
$p \ldots (0,0)$	$(0,0)$	$(0,1)$	$\boxed{\boxed{(1,0)}}$	$(0,0)$	$(0,0)$
$r \ldots (0,1)$	$(1,1)$	$(0,1)$	$(1,0)$	$\boxed{\boxed{(0,0)}}$	$(0,0)$
$s \ldots (1,1)$	$(1,1)$	$(-,-)$	$\boxed{\boxed{(1,0)}}$	$(0,0)$	$(1,0)$
$t \ldots (1,0)$	$(-,-)$	$(-,-)$	$(1,0)$	$(0,0)$	$(0,1)$

funkcií, ktoré sú dané kódovacími tabuľkami na obr. 51, časť b), c), d). V tabuľke prechodovej funkcie sme dvojitým rámčekom vyznačili hodnoty, ktoré sme dodefinovali tak, aby súbehy neboli kritické. Táto tabuľka predstavuje tabuľku dvoch budiacich funkcií Y_1, Y_2 a dvoch výstupných funkcií z_1, z_2. Na miestach s pomlčkami máme teraz možnosť dodefinovať tabuľky ľubovoľným spôsobom a vždy dostaneme automat, ktorý obsahuje podautomat imitujúci činnosť pôvodného neúplne špecifikovaného automatu. Toto dodefinovanie urobíme len náhodným spôsobom, pričom dávame pozor, aby sa pri tomto dodefinovaní nevyskytli nové súbehy. Teda definujme $\delta_{\mathbf{B}}((1,1),(1,0)) = (0,1)$, $\delta_{\mathbf{B}}((1,0),(0,0)) = (0,0)$, $\delta_{\mathbf{B}}((1,0),(1,0)) = (1,1)$. Tabuľky funkcií Y_1, Y_2 už s uvedeným dodefinovaním sú na obr. 52, časť a), b). Tabuľky výstupných funkcií z_1, z_2 uvádzame na obr. 52, časť c), d). Dvojkový automat budeme realizovať pomocou asynchrónnych preklápacích obvodov. Pre dve budiace funkcie Y_1, Y_2 budeme potrebovať dva SR - preklápacie obvody. Vstupné funkcie SR - preklápacieho obvodu, ktorý bude generovať budiacu funkciu Y_1, označíme S_1 a R_1, a toho, ktorý bude generovať budiacu funkciu Y_2, označíme S_2 a R_2. Tabuľky funkcií S_1, R_1 a S_2, R_2 dostaneme z tabuliek funkcií Y_1 a Y_2 už známym spôsobom.

Uvádzame ich na obr. 53. Z Karnaughových máp na tomto obrázku čítame:



Obr. 52. Tabuľky budiacich a výstupných funkcií z príkladu 4.9



Obr. 53. Tabuľky vstupných funkcií SR - preklápacích obvodov z príkladu 4.9

$$S_1 = y_2\overline{x}_1\overline{x}_2 + x_1x_2, \qquad R_1 = \overline{y}_2\overline{x}_1 + y_2x_1\overline{x}_2 + \overline{x}_1x_2,$$

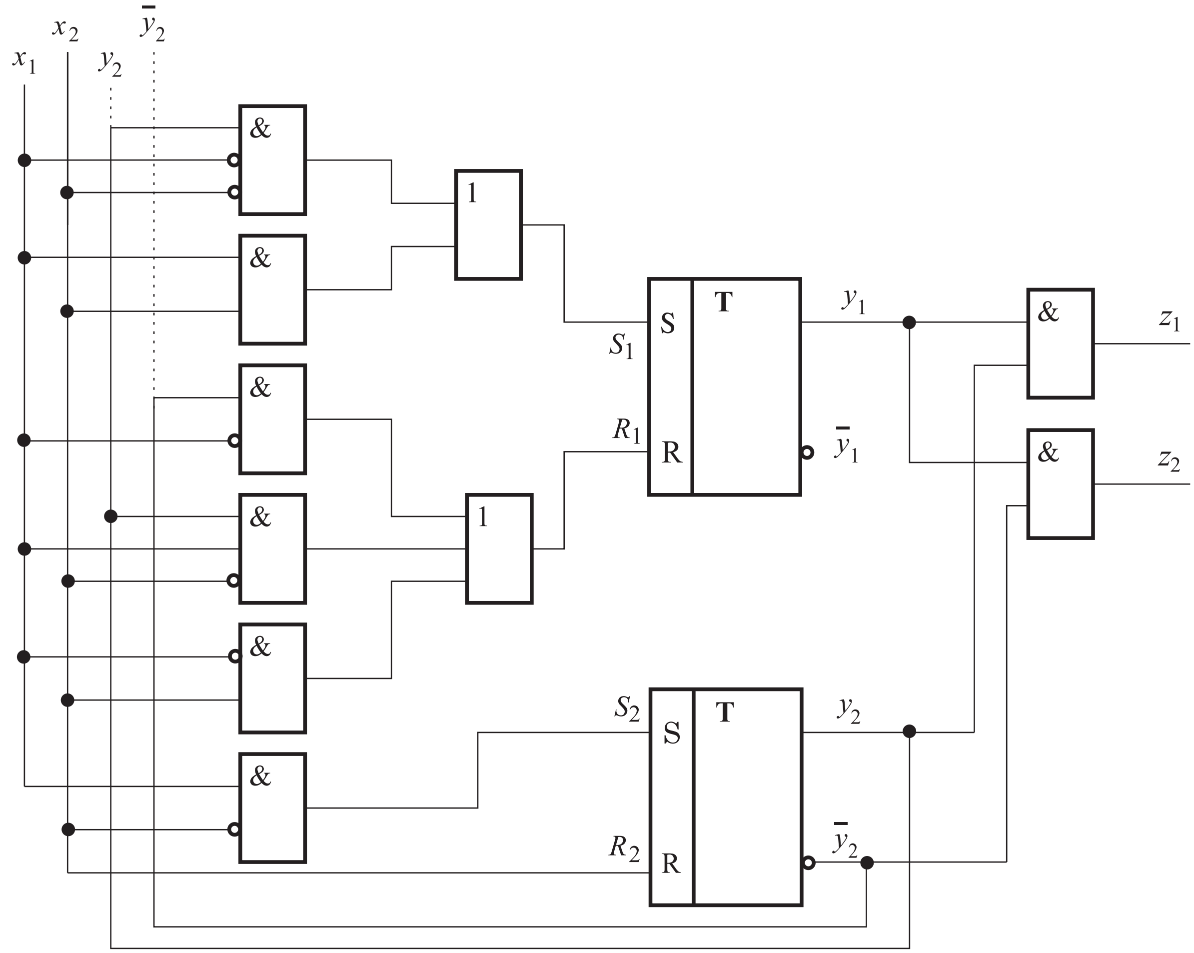
$$S_2 = x_1\overline{x}_2, \qquad R_2 = x_2.$$

Pre výstupné funkcie z_1, z_2 z obr. 53 čítame:

$$z_1 = y_1y_2,$$

$$z_2 = y_1\overline{y}_2.$$

Príslušnú logickú sieť uvádzame na obr. 54. ■



Obr. 54. Logická sieť patriaca k automatu z príkladu 4.9

Na záver tohto príkladu chceme podotknúť, že pri fyzikálnej realizácii fundamentálneho automatu neuvažujeme o dvojkovom automate, ktorý pokrýva daný automat, ale len o automate, ktorý imituje činnosť daného automatu v tom zmysle, že pri pevnom vstupe a danom stave sa dvojkový automat dostáva do stabilného stavu,ktorý zodpovedá stabilnému stavu pôvodného automatu, a v tomto stabilnom stave vydáva dvojkový automat stabilný výstup, ktorý zodpovedá stabilnému výstupu pôvodného automatu. Nestabilné medzistavy a k nim patriace výstupy nás nezaujímajú. V takomto zmysle pracuje aj zodpovedajúci logický obvod, ktorý je fyzikálnou realizáciou daného dvojkového automatu. Ďalej si treba uvedomiť, že konečný automat je dosť nedokonalým modelom správania sa asynchrónneho sekvenčného logického obvodu. Navrhnutý model treba ešte podrobiť dôkladnej analýze a navrhnúť spôsob jeho činnosti. To však nie je obsahom nášho predmetu.

Príklad 4.10. Teraz navrhneme fyzikálnu realizáciu Moorovho automatu, ktorý je daný tabuľkou 14.

Riešenie. Najprv si všimnime, aké sú možnosti pre kódovanie stavov tohto automatu pomocou kódovacích slov z $\mathbf{B}^2$. Treba si uvedomiť, že kódy dvoch stavov, ktoré ležia v tom istom riadku alebo stĺpci kódovacej tabuľky sa líšia iba na jednom mieste. Preto medzi týmito stavmi nedochádza k súbehu. Pri skúmaní súbehov, preto stačí uvažovať o takýchto rozloženiach stavov v kódovacích tabuľkách:

Tabuľka 14. Automat z príkladu 4.10

	a	b	c	d	μ
A	$\boxed{A}$	B	$\boxed{A}$	$\boxed{A}$	1
B	A	$\boxed{B}$	D	$\boxed{B}$	0
C	$\boxed{C}$	D	A	B	1
D	C	$\boxed{D}$	$\boxed{D}$	A	0

(1) A je v uhlopriečke s B, a teda C je v uhlopriečke s D, napr. $$\begin{array}{c|c|c|} \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{\overline{y_2}} \\ \cline{2-3} & A & D \\ \cline{2-3} y_1 & C & B \\ \cline{2-3} \end{array},$$

(2) A je v uhlopriečke s C, a teda B je v uhlopriečke s D, napr. $$\begin{array}{c|c|c|} \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{\overline{y_2}} \\ \cline{2-3} & A & D \\ \cline{2-3} y_1 & B & C \\ \cline{2-3} \end{array},$$

(3) A je v uhlopriečke s D, a teda B je v uhlopriečke s C, napr. $$\begin{array}{c|c|c|} \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{\overline{y_2}} \\ \cline{2-3} & A & C \\ \cline{2-3} y_1 & B & D \\ \cline{2-3} \end{array}.$$

V prvom prípade v kódovom ekvivalente dochádza k súbehu pri prechode medzi stavmi, ktorý zodpovedá prechodu $b : A \longrightarrow B$. Tento súbeh prebieha takto:

$$\begin{tikzcd}
& \begin{matrix} C \\ (1,0) \end{matrix} \arrow[r] & \begin{matrix} D \\ (0,1) \end{matrix} \arrow[loop right] \\
b:\ \begin{matrix} A \\ (0,0) \end{matrix} \arrow[ur] \arrow[dr] & & \\
& \begin{matrix} D \\ (0,1) \end{matrix} \arrow[r] & \begin{matrix} D \\ (0,1) \end{matrix} \arrow[loop right]
\end{tikzcd}$$

Stav D je pri vstupe b stabilný

V druhom prípade dochádza v dvojkovom ekvivalente k súbehu pri prechode medzi stavmi, ktorý zodpovedá prechodu $c : B \longrightarrow D$. Tento súbeh prebieha takto:

$$\begin{tikzcd}
& \begin{matrix} A \\ (0,0) \end{matrix} \arrow[r] & \begin{matrix} A \\ (0,0) \end{matrix} \arrow[loop right] \\
c:\ \begin{matrix} B \\ (1,0) \end{matrix} \arrow[ur] \arrow[dr] & & \\
& \begin{matrix} C \\ (1,1) \end{matrix} \arrow[r] & \begin{matrix} A \\ (0,0) \end{matrix} \arrow[loop right]
\end{tikzcd}$$

Stav A je pri vstupe c stabilný. Preto sa automat opäť nedostane do požadovaného stavu. Opäť dostávame kritický súbeh a zvyšné súbehy už nemusíme vyšetrovať.

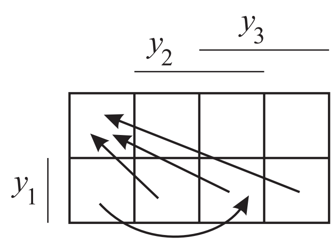
V treťom prípade v dvojkovom automate dostávame súbeh napríklad pri prechode medzi stavmi, ktorý zodpovedá prechodu $d : C \longrightarrow B$. Tento prechod má takýto priebeh:

$$\begin{tikzcd} & \underset{(0,0)}{A} \arrow[r] & \underset{(0,0)}{A} \arrow[loop right] \\ d:\ \underset{(0,1)}{C} \arrow[ur] \arrow[dr] & & \\ & \underset{(1,1)}{D} \arrow[r] & \underset{(0,0)}{A} \arrow[loop right] \end{tikzcd}$$

Z toho vidíme, že opäť dostávame kritický súbeh.

Treba si uvedomiť, že uvedené tri možnosti vzhľadom na vyšetrovanie súbehov predstavujú v podstate všetky možnosti pre kódovanie množiny stavov pomocou dvojíc núl a jednotiek. Zvyšné možnosti by viedli len k výmene riadkov v schémach, ktoré opisujú súbehy. To znamená, že kódovanie stavov pomocou dvojíc vedie v tomto príklade vždy ku kritickým súbehom. Preto budeme stavy kódovať pomocou trojíc, pričom budeme organizovať bezsúbehové prechody medzi stavmi. To znamená, že zostrojíme dvojkový automat, ktorý bude mať 8 stavov. Tento automat bude neúplne špecifikovaný. Preto budeme môcť organizovať bezsúbehové prechody do stabilného stavu cez nestabilné stavy a potom navrhneme fyzikálnu realizáciu takéhoto automatu.

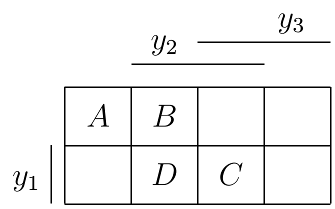
Keď kódujeme stavy pomocou trojíc, súbehy nastávajú pri prechodoch zo stavu do stavu, pri ktorých sa menia aspoň dve kódovacie miesta. Vzájomnú polohu niektorých takých stavov, medzi ktorými dochádza k súbehu, v kódovacej tabuľke sme symbolicky vyznačili na obr. 55. V našom príklade sú všetky možné prechody medzi stavmi tieto:



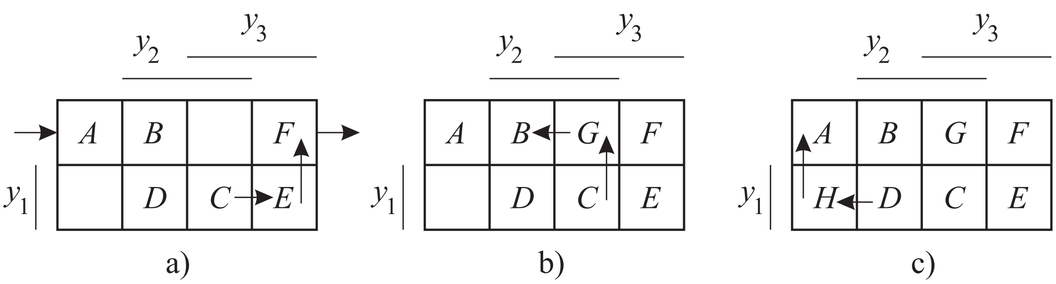
Obr. 55. Súbehy pri prechodoch medzi stavmi

$A \longrightarrow B$, $B \longrightarrow A$, $B \longrightarrow D$, $C \longrightarrow D$, $C \longrightarrow A$, $C \longrightarrow B$, $D \longrightarrow C$, $D \longrightarrow A$. Pretože medzi týmito prechodmi sa vyskytujú aj dvojice prechodov $A \longrightarrow B$, $B \longrightarrow A$ a $C \longrightarrow D$, $D \longrightarrow C$, je vhodné zakódovať stavy tak, aby sa stavy A, B a C, D nachádzali v kódovacej tabuľke vedľa seba. V našom príklade začneme s návrhom kódovania stavov pomocou kódovacej tabuľky na obr. 56. Vidíme, že pri tomto kódovaní budú prechody $b : A \longrightarrow B$, $a : B \longrightarrow A$, $b : C \longrightarrow D$, $a : D \longrightarrow C$, $c : B \longrightarrow D$, bezsúbehové. Avšak pri tomto kóde v kódovom ekvivalente k súbehom vedú prechody: $c : C \longrightarrow A$, $d : C \longrightarrow B$, $d : D \longrightarrow A$. Preto treba tieto prechody zorganizovať cez nové stavy, ktoré pridáme k stavom pôvodného automatu.

Najprv budeme organizovať prechod $c : C \longrightarrow A$. K danému automatu pridáme stavy E, F, ktoré zakódujeme pomocou tabuľky na obr. 57, časť a). Potom musíme žiadať, aby sa pri pevnom vstupe c uskutočnil prechod $c : C \longrightarrow E \longrightarrow F \longrightarrow A$. V rozšírenom automate tento prechod pre prechodovú funkciu predstavuje podmienku: $\delta(C, c) = E$, $\delta(E, c) = F$, $\delta(F, c) = A$.



OBR. 56. Kódovacia tabuľka množiny stavov z príkladu 4.10



OBR. 57. Organizácia bezsúbehových prechodov medzi stavmi v príklade 4.10

Ďalej budeme organizovať prechod $d : C \longrightarrow B$. Pre tento prechod stačí pridať jeden stav G, ktorý zakódujeme pomocou kódovacej tabuľky na obr. 57, časť b). Potom budeme žiadať, aby sa pri pevnom vstupe d uskutočnil prechod $d : C \longrightarrow G \longrightarrow B$. V rozšírenom automate žiadame, aby $\delta(C, d) = G$, $\delta(G, d) = B$.

Prechod $d : D \longrightarrow A$ zorganizujeme pridaním stavu H, ktorý zakódujeme pomocou tabuľky na obr. 57, časť c). Žiadame teda, aby sa pri pevnom vstupe d uskutočnil prechod $d : D \longrightarrow H \longrightarrow A$. To znamená, že žiadame, aby $\delta(D, d) = H$, $\delta(H, d) = A$.

Pre výstupnú funkciu budeme žiadať, aby pridané stavy dávali ten istý výstup, ako dáva stav, do ktorého sprostredkujú prechod. V našom prípade žiadame, aby $\mu(E) = \mu(F) = \mu(A) = 1$, $\mu(G) = \mu(B) = 0$ a $\mu(H) = \mu(A) = 1$. Takto doplnený neúplný špecifikovaný automat uvádzame v tabuľke 15.

TABUĽKA 15. Neúplne špecifikovaný automat z príkladu 4.10

	a	b	c	d	μ
A	A	B	A	A	1
B	A	B	D	B	0
C	C	D	E	G	1
D	C	D	D	H	0
E	–	–	F	–	1
F	–	–	A	–	1
G	–	–	–	B	0
H	–	–	–	A	1

Stavy tohto automatu už máme zakódované pomocou tabuľky na obr. 57, časť c). Výstupy kódovať nemusíme a vstupnú abecedu zakódujeme pomocou kódovacej tabuľky na obr. 58. Teraz už môžeme opísať kódový ekvivalent tohto automatu. Tento opis je daný pomocou tabuľky 16. Na obr. 59 uvádzame tabuľku budiacich funkcií Y_1, Y_2, Y_3 úplne

		x_2
	a	d
x_1	b	c

Obr. 58. Kódovacia tabuľka vstupnej abecedy z príkladu 4.10

špecifikovaného dvojkového automatu, na ktorý sme doplnili automat z tabuľky 16. Pri dopĺňaní sme prihliadali iba na ten fakt, aby pri doplnených prechodoch medzi stavmi ne-

Tabuľka 16. Kódový ekvivalent z príkladu 4.10

	$g(a) = (0,0)$	$g(b) = (1,0)$	$g(c) = (1,1)$	$g(d) = (0,1)$	$\mu_{\mathbf{B}}$
$f(A) = (0,0,0)$	$(0,0,0)$	$(0,1,0)$	$(0,0,0)$	$(0,0,0)$	1
$f(H) = (1,0,0)$	$(-,-,-)$	$(0,1,0)$	$(0,0,0)$	$(0,0,0)$	1
$f(D) = (1,1,0)$	$(1,1,1)$	$(1,1,0)$	$(1,1,0)$	$(1,0,0)$	0
$f(B) = (0,1,1)$	$(0,1,0)$	$(0,0,0)$	$(0,1,0)$	$(0,1,0)$	0
$f(G) = (1,1,0)$	$(-,-,-)$	$(-,-,-)$	$(-,-,-)$	$(0,1,0)$	0
$f(C) = (1,1,1)$	$(1,1,1)$	$(1,1,0)$	$(1,0,1)$	$(0,1,1)$	1
$f(E) = (1,0,1)$	$(-,-,-)$	$(-,-,-)$	$(0,0,1)$	$(-,-,-)$	1
$f(F) = (0,0,1)$	$(-,-,-)$	$(-,-,-)$	$(0,0,0)$	$(-,-,-)$	1

dochádzalo k súbehom. Tabuľku výstupnej funkcie $z = \mu(y_1, y_2, y_3)$ uvádzame na obr. 60.

Budiace funkcie daného dvojkového automatu budeme generovať pomocou SR - preklápacích obvodov. Vstupné funkcie SR - preklápacieho obvodu, ktorý generuje budiacu funkciu Y_i, sme označili S_i a R_i pre $i = 1, 2, 3$. Ich tabuľky, ktoré sme získali už známym spôsobom, uvádzame na obr. 61. Z Karnaughových máp pre vstupné funkcie S_1, R_1, S_2, R_2, S_3, R_3 a pre výstupnú funkciu z čítame:

$$\begin{aligned}
S_1 &= y_2\overline{y}_3x_1x_2, & R_1 &= y_2 + y_3\overline{x}_1x_2,\\
S_2 &= \overline{y}_1\overline{y}_2\overline{y}_3x_1\overline{x}_2, & R_2 &= \overline{y}_1y_2\overline{y}_3\overline{x}_1\overline{x}_2 + y_1\overline{y}_3\overline{x}_1x_2 + y_1y_3x_1x_2,\\
S_3 &= y_1y_2\overline{y}_3\overline{x}_1\overline{x}_2, & R_3 &= \overline{y}_1 + y_2x_1\overline{x}_2,\\
z &= \overline{y}_2 + y_1y_3.
\end{aligned}$$

Príslušnú logickú sieť reprezentujúcu fyzikálnu realizáciu daného dvojkového automatu uvádzame na obr. 62. Ak logický obvod patriaci k nakreslenej logickej sieti bude pracovať vo fundamentálnom režime, bude imitovať činnosť pôvodného automatu. ■

	$(0,0,0)$	$(0,1,0)$	$(0,0,0)$	$(0,0,0)$
y_1	$(0,0,0)$	$(0,0,0)$	$(0,0,0)$	$(0,0,0)$
	$(1,1,1)$	$(1,1,0)$	$(1,1,0)$	$(1,0,0)$
y_2	$(0,0,0)$	$(0,1,0)$	$(1,1,0)$	$(0,1,0)$
	$(0,1,0)$	$(0,1,0)$	$(0,1,0)$	$(0,1,0)$
	$(1,1,1)$	$(1,1,0)$	$(1,0,1)$	$(0,1,1)$
y_1	$(0,0,1)$	$(0,0,1)$	$(0,0,1)$	$(0,0,1)$
y_3	$(0,0,0)$	$(0,0,0)$	$(0,0,0)$	$(0,0,0)$

(Y_1, Y_2, Y_3)

Obr. 59. Budiace funkcie z príkladu 4.10

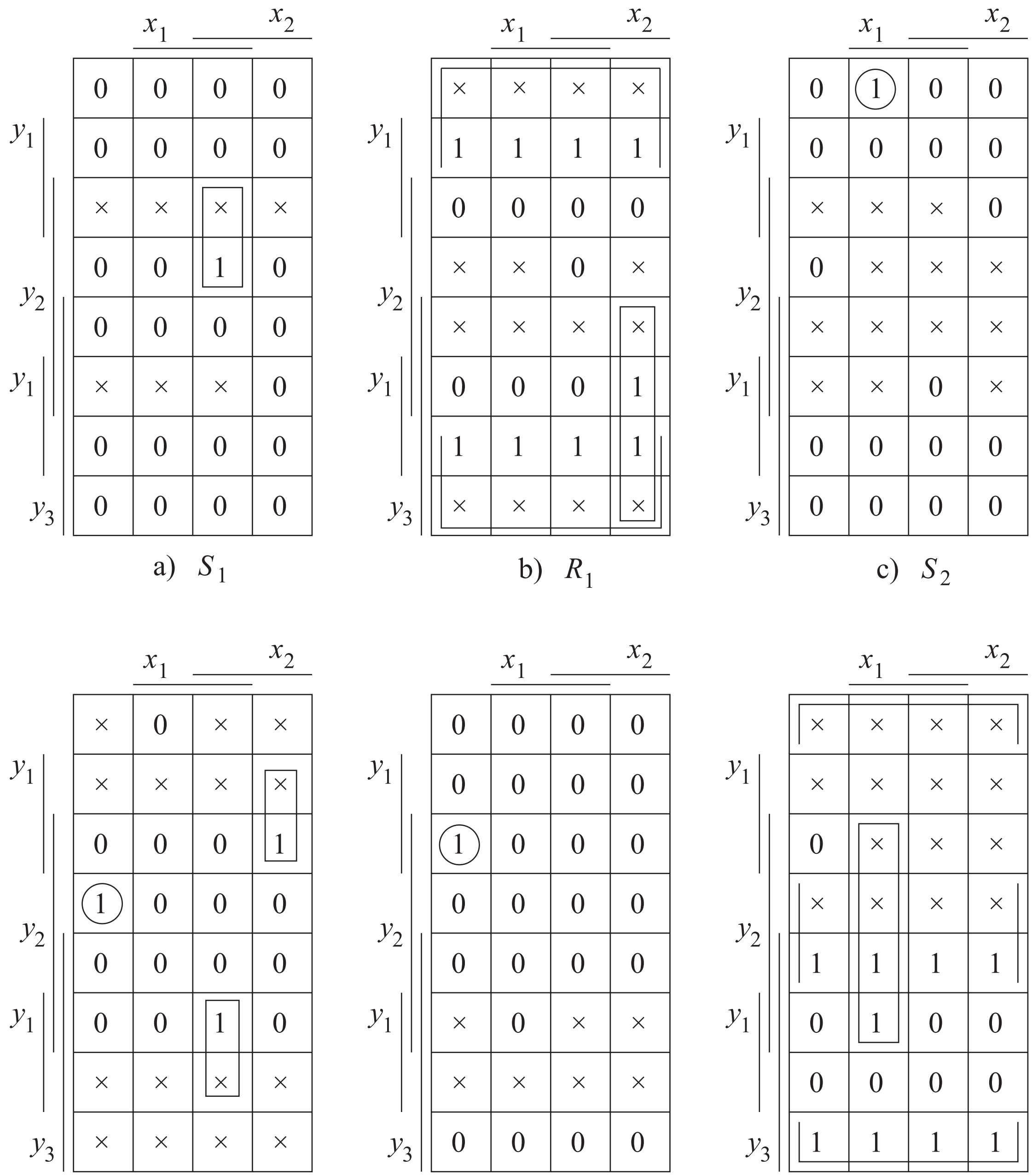
	1	0	0	1
y_1	1	0	1	1

z

Obr. 60. Tabuľka výstupnej funkcie z príkladu 4.10

B) Impulzové asynchrónne sekvenčné logické obvody

V časti **A)** sme ukázali triedu fundamentálnych automatov, ktoré je možné fyzikálne realizovať pomocou fundamentálnych asynchrónnych logických obvodov pracujúcich vo fundamentálnom režime. V tejto časti, najmä na príkladoch, ukážeme, že pri vhodne zvolenom režime práce niektorých sekvenčných logických obvodov sa činnosť týchto obvodov dá opísať pomocou konečného automatu. Ide o logické obvody, pri ktorých prechody medzi stavmi môžu byť vyvolané významnou zmenou hodnoty (buď z 0 na 1 alebo naopak, poprípade obe zmeny môžu byť významné) pri špecifikovaných vstupných, prípadne aj stavových premenných logického obvodu. Tieto špecifikované premenné sa nazývajú ***impulzové premenné***. Prechody medzi stavmi však vo všeobecnosti závisia aj od momentálnych hodnôt ostatných „neimpulzových" premenných. Tieto vstupné a stavové premenné nazývame ***hladinové premenné***. Zmeny týchto premenných a aj nevýznamné zmeny impulzových premenných nemôžu vyvolať prechody medzi stavmi a neurčujú body diskrétneho času. Body diskrétneho času sú určované iba významnými zmenami impulzových premenných. Tieto významné zmeny pri niektorých impulzových premenných budú zmeny z 0 na 1, vtedy hovoríme, že bod diskrétneho času je určený čelom impulzu. Keď významnou zmenou impulzovej premennej je zmena z 1 na 0, vtedy hovoríme, že bod diskrétneho času je určený tylom impulzu. Spravidla, pri každej impulzovej premennej je iba jedna z týchto zmien významnou. Je zrejmé, že môžeme uvažovať aj o impulzových premenných, pri ktorých obe tieto zmeny sú významné.



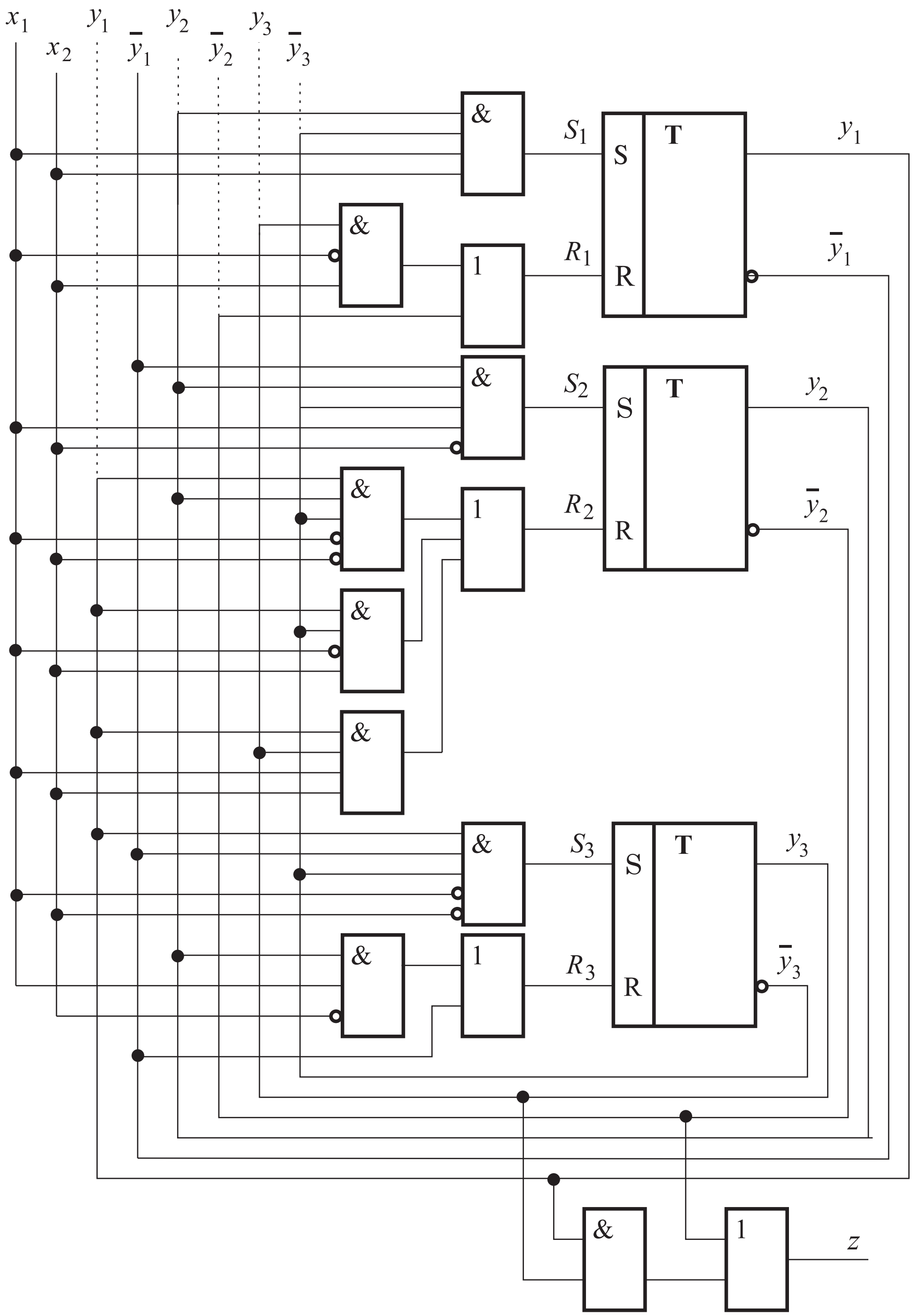
a) S_1 b) R_1 c) S_2

d) R_2 e) S_3 f) R_3

Obr. 61. Vstupné funkcie SR - preklápacieho obvodu z príkladu 4.10

Pre činnosť sekvenčných obvodov z uvedenej triedy je dôležité, aby okolie, ktoré generuje vstupné signály a aj samotný logický obvod, dodržalo v priebehu činnosti istý režim práce. Nazývame ho ***impulzový režim*** a dá sa charakterizovať takto:

1) V každom bode diskrétneho času nastáva významná zmena hodnoty iba pri jednej impulzovej premennej.
2) Pri významnej zmene impulzovej hodnoty sú hladinové premenné konštantné.
3) Významná zmena hodnoty niektorej vstupnej impulzovej premennej sa objaví iba vtedy, keď sa obvod nachádza v stabilnom stave (t.j. v takom stave, v ktorom by obvod zotrval, keby nenastala žiadna významná zmena hodnoty impulzovej premennej).



Obr. 62. Logická sieť patriaca k automatu z príkladu 4.10

Aby sa netriviálnym spôsobom mohla splniť tretia podmienka impulzového režimu, musí mať sekvenčný obvod túto vlastnosť: Po indikovaní významnej zmeny niektorej impulzovej premennej alebo zostane v rovnakom stave ako bol, alebo s určitým oneskorením, ktoré je dané fyzikálnymi vlastnosťami obvodu, prejde do niektorého iného stavu. V tomto stave zotrvá najmenej do ďalšej významnej zmeny niektorej z impulzových premenných.

Asynchrónne sekvenčné obvody, ktoré majú vyššie opísané vlastnosti, nazývame ***impulzové*** (alebo ***hladinovo impulzové***) ***logické obvody***.

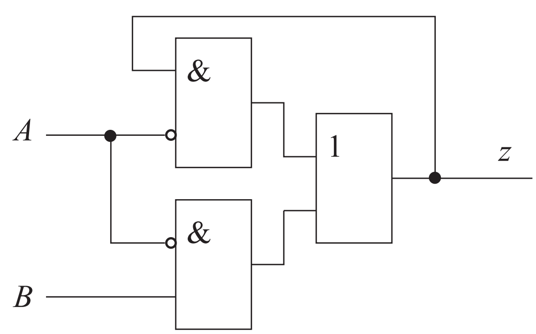
Pri impulzových sekvenčných obvodoch sú najčastejšie za impulzové premenné zvolené niektoré zo vstupných premenných a všetky stavové premenné majú hladinový charakter.

Špeciálnym prípadom takéhoto obvodu je synchrónny logický obvod, v ktorom diskrétny čas určuje jedna impulzová vstupná premenná.

V prípade, že každej impulzovej premennej je priradená iba jedna význačná zmena (buď čelo alebo tyl impulzu), dá sa aj pri n vstupných impulzových premenných daný impulzový obvod opísať ako Mealyho alebo Moorov automat. V prípade, že daný obvod má ešte r hladinových vstupných premenných, bude treba pre každú z n impulzových premenných uvažovať o 2^r možnostiach pre r hladinových premenných (z impulzových premenných môže mať významnú hodnotu vždy iba jedna). Preto v takomto obvode môžeme uvažovať o vstupnej abecede, ktorá sa skladá z $n2^n$ vstupných vektorov, čiže treba opísať činnosť daného obvodu pri n možných rozdeleniach významných hodnôt pri impulzových premenných. V tomto prípade vlastne ide o zovšeobecnenie synchrónnych logických obvodov. Rozloženie hodnôt impulzových premenných, pri významnej zmene jednej z nich, je zachytené v stavoch automatu.

V nasledujúcom príklade sa budeme zaoberať impulzovým logickým obvodom, v ktorom všetky vstupné premenné sú impulzové a pri všetkých impulzových premenných je každá zmena (aj z 0 na 1 a aj z 1 na 0) významná.

Príklad 4.11. Uvažujme o logickom obvode, ktorý je reprezentovaný sekvenčnou logickou sieťou na obr. 63. Budeme predpokladať, že tento obvod pracuje v impulzovom režime, pričom impulzové premenné sú iba vstupné premenné a každá zmena vstupnej premennej je významná. Pretože hladinové premenné sa v tomto prípade na vstupoch nevyskytujú, a každá zo vstupných impulzových premenných nadobúda dve významné



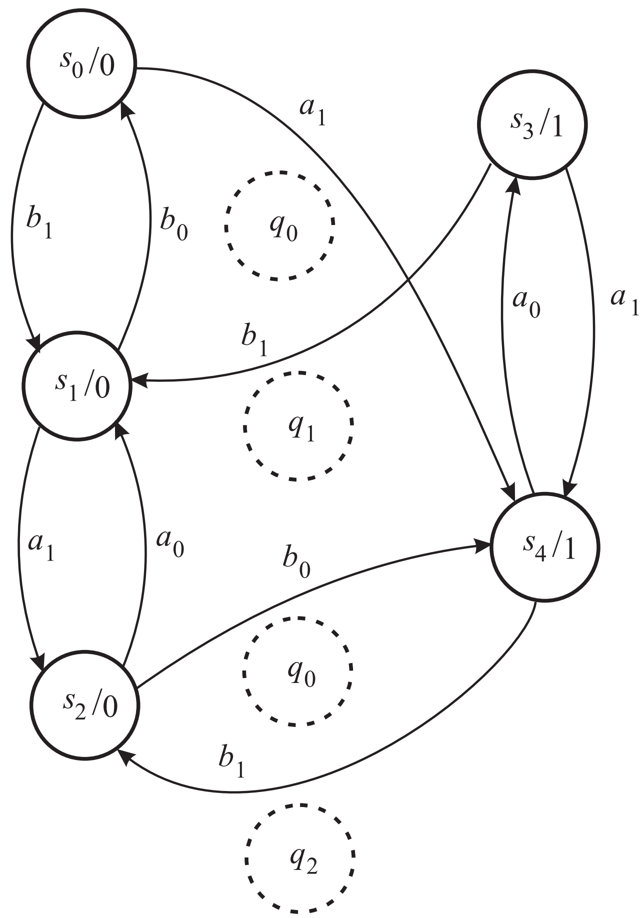
Obr. 63. Logická sieť z príkladu 4.11

zmeny hodnôt, automat, ktorý bude opisovať činnosť tohto obvodu, bude mať vstupnú abecedu pozostávajúcu zo štyroch písmen, ktoré zodpovedajú možnostiam: A sa zmení na 1, A sa zmení na 0, B sa zmení na 1, B sa zmení na 0. Tieto možnosti postupne označíme takto:

$$a_1 = (A \to 1), a_0 = (A \to 0), b_1 = (B \to 1), b_0 = (B \to 0.)$$

Aby činnosť toho obvodu mohla byť opísaná pomocou automatu, musia existovať jeho stabilné stavy, čiže stavy, ktoré sa pri nezmenenom vstupe nemenia. Hodnoty výstupnej funkcie tohto obvodu môžeme reprezentovať pomocou NDF v tvare $\overline{B}z + A\overline{B}$. Je zrejmé, že daný obvod bude v stabilnom stave, práve vtedy, keď $z = \overline{B}z + A\overline{B}$. Uvažujme teraz o dvoch možnostiach.

a) Nech $z = 0$, potom nutne $A\overline{B} = 0$. To je splnené práve v týchto prípadoch $(A = 0, B = 0)$, $(A = 0, B = 1)$, $(A = 1, B = 1)$.
b) Nech $z = 1$, potom $1 = \overline{B} + A\overline{B} = \overline{B}$. Táto podmienka je splnená práve v prípadoch $(A = 0, B = 0)$, $(A = 1, B = 0)$.



Obr. 64. Graf automatu z príkladu 4.11

Vidíme, že daný obvod má päť stabilných stavov, ktoré zodpovedajú trojiciam (A, B, z). Tieto stavy označíme takto: $s_0 = (A = 0, B = 0, z = 0) = (0, 0, 0)$,$s_1 = (0, 1, 0)$, $s_2 = (1, 1, 0)$, $s_3 = (0, 0, 1)$, $s_4 = (1, 0, 1)$.

Zvyšné tri stavy $q_0 = (1, 0, 0)$, $q_1 = (0, 1, 1)$, $q_2 = (1, 1, 1)$ sú nestabilné. V prvom z nich pri $A = 1, B = 0, z = 0$ dostávame výstupnú hodnotu $\overline{0} \cdot 0 + 1 \cdot \overline{0} = 1$, a teda pri pevnom vstupe $A = 1, B = 0$ logický obvod prejde do stabilného stavu $s_4 = (1, 0, 1)$. Píšeme $q_0 = (1, 0, 0) \to s_4 = (1, 0, 1)$. Podobne dostaneme $q_1 = (0, 1, 1) \to s_1 = (0, 1, 0)$ a $q_2 = (1, 1, 1) \to s_2 = (1, 1, 0)$.

Ak za body diskrétneho času budeme považovať iba body, v ktorých sa zmení jedna z impulzových premenných, v stave $s_0 = (0, 0, 0)$ môže prísť iba ku zmene premennej A z 0 na 1 alebo premennej B z 0 na 1. Podobná situácia je vo zvyšných stabilných stavoch. Vidíme, že tento logický obvod môžeme opísať ako neúplne špecifikovaný Moorov automat, ktorého vstupnú abecedu sme už opísali na začiatku, a výstup v každom stave sa bude rovnať tretej zložke tohto stavu. Graf tohto automatu je na obr. 64. V prípade prechodu cez nestabilný stav sme tento stav v grafe na obr. 64 vyznačili pod príslušnou hranou. Tabuľku tohto neúplne špecifikovaného automatu uvádzame v tabuľke 17, časť a). V tabuľke 17, časť b) sme tento automat úplne špecifikovali na základe toho, že daný logický obvod pri nezmenenom vstupe zostáva v stabilnom stave. ■

Tabuľka 17. Tabuľky automatov z príkladu 4.11

	a_0	a_1	b_0	b_1	
s_0	$-$	s_4	$-$	s_1	0
s_1	$-$	s_2	s_0	$-$	0
s_2	s_1	$-$	s_4	$-$	0
s_3	$-$	s_4	$-$	s_1	1
s_4	s_3	$-$	$-$	s_2	1

a)

	a_0	a_1	b_0	b_1	
s_0	$\boxed{s_0}$	s_4	$\boxed{s_0}$	s_1	0
s_1	$\boxed{s_1}$	s_2	s_0	$\boxed{s_1}$	0
s_2	s_1	$\boxed{s_2}$	s_4	$\boxed{s_2}$	0
s_3	$\boxed{s_3}$	s_4	$\boxed{s_3}$	s_1	1
s_4	s_3	$\boxed{s_4}$	$\boxed{s_4}$	s_2	1

b)

Literatúra

[1] Satko, L.: Teória automatov. EF SVŠT, Bratislava 1988.
[2] Galanová, J., Kaprálik, P.: Diskrétna matematika. FEI STU, Bratislava 1997
[3] Frištacký, N., Kolesár, M.,Kolenička, J.,Hlavatý, J.: Logické systémy. Alfa, Bratislava 1986.
[4] Pütz, J. a kol.: Úvod do číslicovej techniky. SNTL, Praha 1983.
[5] Aho, A.V., Ullman, J.D.: Foundation of Computer Science. W.H. Freeman and Company, New York 1992.

Obsah

Kapitola 1. Úvodné pojmy 3
1. Zobrazenia a operácie 3
2. Jazyk nad abecedou 5
3. Výroková logika 6
4. Relácie 16
5. Orientované grafy 19

Kapitola 2. Booleovské funkcie 25
1. Booleovské funkcie a booleovské výrazy 25
2. Úplný systém booleovských funkcií 40
3. Kombinačné logické siete 43
4. Booleovské funkcie $f : \mathbf{B}^n \to \mathbf{B}^m$ 52
5. Karnaughove mapy 58
6. Minimalizácia B-výrazov 68

Kapitola 3. Konečné automaty 79
1. Definícia konečného automatu 79
2. Konečné akceptory 89
3. Základné pojmy v teórii konečných automatov 92
4. Neúplne špecifikované (nedeterministické) automaty 96
5. Ekvivalencia automatov 98
6. Izomorfizmus automatov 110
7. Pokrytie automatu automatom 114

Kapitola 4. Fyzikálna realizácia automatov 119
1. Dvojkové automaty 119
2. Preklápacie obvody 121
3. Synchrónne a asynchrónne logické obvody 130
4. Fyzikálna realizácia automatov 133

Literatúra 163