

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 päťica $A = (S, X, Z, \delta, \lambda)$, kde $S = \mathbf{B}^k$, $X = \mathbf{B}^n$, $Z = \mathbf{B}^m$, $\delta : S \times X \rightarrow S$, $\lambda : S \times X \rightarrow Z$. **Dvojkový Mooreov automat** definujeme analogicky, teda ako päťicu $A = (S, X, Z, \delta, \mu)$, kde $S = \mathbf{B}^k$, $X = \mathbf{B}^n$, $Z = \mathbf{B}^m$, $\delta : S \times X \rightarrow S$, $\mu : S \rightarrow Z$.

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

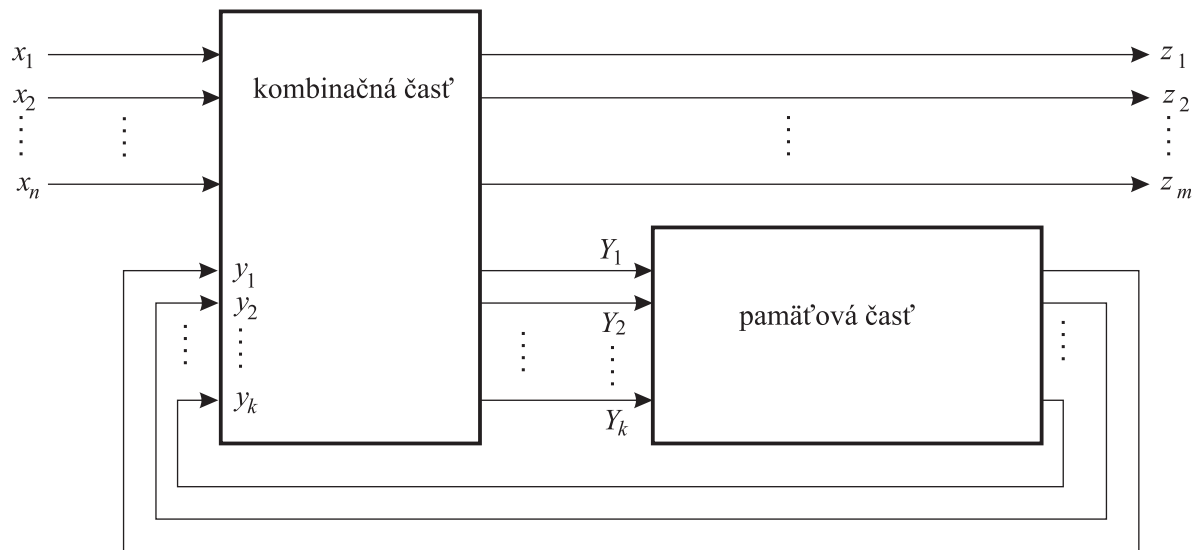
	0	1	0	1	
(0,0)	(0,0)	(0,1)	0	1	}
(1,0)	(1,1)	(1,0)	1	0	
(1,1)	(1,1)	(0,1)	1	1	
(0,1)	(1,0)	(1,0)	1	0	
$\delta((y_1, y_2), x) = (Y_1, Y_2)$					$\lambda((y_1, y_2), x) = z$

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 $\delta : \mathbf{B}^2 \times \mathbf{B} \rightarrow \mathbf{B}^2$ môžeme pokladať za funkciu $\delta : \mathbf{B}^3 \rightarrow \mathbf{B}^2$ troch premenných y_1, y_2, x , ktorá má dve zložky $\delta_1 : \mathbf{B}^3 \rightarrow \mathbf{B}$, $\delta_1(y_1, y_2, x) = Y_1$ a $\delta_2 : \mathbf{B}^3 \rightarrow \mathbf{B}$, $\delta_2(y_1, y_2, x) = Y_2$. Tak isto aj funkciu $\lambda : \mathbf{B}^2 \times \mathbf{B} \rightarrow \mathbf{B}$ môžeme považovať za funkciu troch premenných, ktorá má jednu zložku $\lambda : \mathbf{B}^3 \rightarrow \mathbf{B}$, $\lambda(y_1, y_2, x) = z$. 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:

$$\begin{aligned}\delta_1(y_1, y_2, x) &= Y_1 = y_1\bar{y}_2 + y_1\bar{x} + \bar{y}_1y_2, \\ \delta_2(y_1, y_2, x) &= Y_2 = \bar{y}_1\bar{y}_2x + y_1\bar{x} + y_1y_2, \\ \lambda(y_1, y_2, x) &= z = \bar{y}_1\bar{y}_2x + y_1\bar{x} + y_1y_2 + y_2\bar{x}.\end{aligned}$$

Vidíme, že budeme schopní generovať nový stav a výstup, ak budeme vedieť opísať zariadenie, ktoré uchová predošlý stav (y_1, y_2) . ■

Teraz budeme uvažovať všeobecne. Nech $A = (\mathbf{B}^k, \mathbf{B}^n, \mathbf{B}^m, \delta, \lambda)$ je dvojkový automat. Potom prechodovú funkciu $\delta : \mathbf{B}^k \times \mathbf{B}^n \rightarrow \mathbf{B}^k$ budeme považovať za funkciu $k + n$ premenných, pričom táto funkcia bude mať k zložiek. Píšeme



OBR. 2. Schéma sekvenčnej logickej siete (najjednoduchší prípad)

neskôr), z ktorých je zostavená. Samotné budiace funkcie sa potom generujú a do ďalšieho taktu uchovávajú práve v týchto preklápacích obvodoch.

Budovanie pamätevej 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 Mooreovom 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

		S		R		
		0	1	0	0	
Q		1	1	0	0	
		0	1			
		1				
		0	1			
		1				

a)

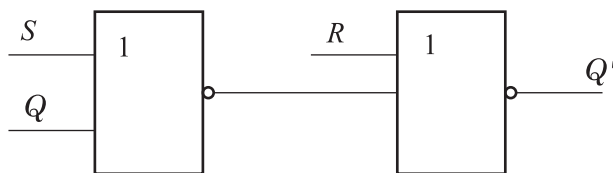
b)

OBR. 3. Tabuľky SR-preklápacieho obvodu

vstupné 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 $\delta(Q, S, R) = Q' = Q\bar{R} + S\bar{R} = (Q + S)\bar{R}$. Navrhne logickú sieť patriacu k tejto funkcii, ktorú zostavíme z členov NOR. Preto

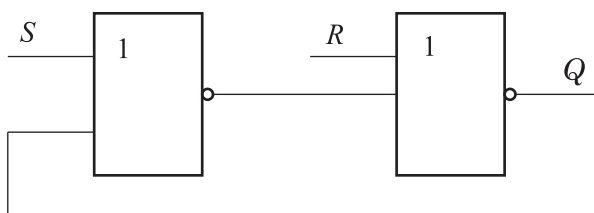
$$Q' = (Q + S)\bar{R} = \overline{\overline{(Q + S)\bar{R}}} = \overline{\overline{(Q + S)} + R}.$$

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 $P' = \overline{(Q' + S)}$.

- 1) $S = 0, R = 0, Q = 0 = Q', P = 1 = P'$,
- 2) $S = 1, R = 1, Q = 0 = Q', P = 0 = P'$,
- 3) $S = 0, R = 1, Q = 0 = Q', P = 1 = P'$,
- 4) $S = 0, R = 0, Q = 1 = Q', P = 0 = P'$,
- 5) $S = 1, R = 0, Q = 1 = Q', P = 0 = P'$.

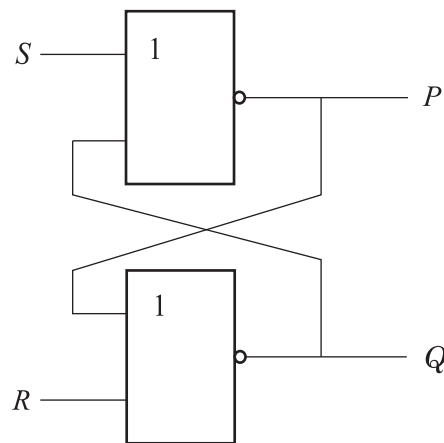
Teraz sa ešte budeme zaoberať zvyšnými troma situáciami, ktoré zodpovedajú jednej jednotke v prvom riadku tabuľky na obr. 3, časť a) a dvom nulám v druhom riadku tejto tabuľky.

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).

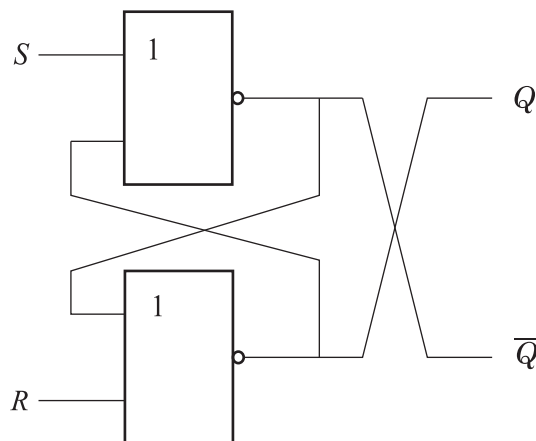
8) $S = 0, R = 1, Q = 1, P = 0$. Nový stav $Q' = (Q + S)\bar{R} = 0$ a $P' = \overline{(Q' + S)} = 1$. Nová situácia je $S = 0, R = 1, Q' = 0, P' = 1$, čo zodpovedá stabilnej situácii 3).



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 $P' = \overline{Q'}$. Prípady 2) a 7) zodpovedajú situácii, keď $S = 1$ a $R = 1$, čiže pri ustálenej situácii (v stabilnom stave) bude $P = \overline{Q}$ 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-



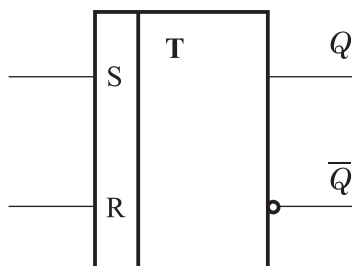
OBR. 7. Ilustrácia SR-preklápacieho obvodu

venčný logický obvod, ktorý je fyzikálnou realizáciou neúplne špecifikovaného Mooreovho 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 nazývať **SR-preklápací obvod**. Na jeho logickú sieť používame štandardné označenie, ktoré uvádzame na obr. 9.

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

		$\overline{S} \quad R$	
		$\overline{S}$	R
Q	0	1	$\times$
	1	1	$\times$

OBR. 8. Tabuľka prechodovej funkcie SR-preklápacieho obvodu



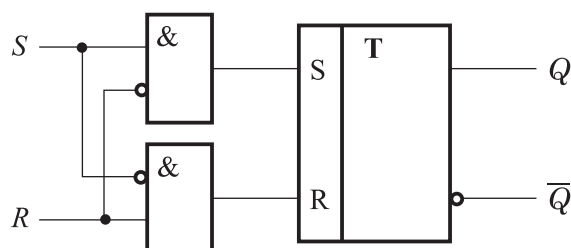
OBR. 9. SR-preklápací obvod

$S = 1$ 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.

		$\overline{S} \quad R$	
		$\overline{S}$	R
Pm	1	$\times$	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



OBR. 11. Ilustrácia SR-preklápacieho obvodu

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 $y = Q$ 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 $y \rightarrow Y$. 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 ($y \rightarrow Y$), ako už bolo spomenuté. Týmito zmenami je vlastne budiaca funkcia definovaná. Prehľad zmien $y \rightarrow Y$ 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

$y \rightarrow Y$	S	R
$0 \rightarrow 0$	0	$\times$
$0 \rightarrow 1$	1	0
$1 \rightarrow 0$	0	1
$1 \rightarrow 1$	$\times$	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 Mooreov 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

		D
	0	1
Q	0	1

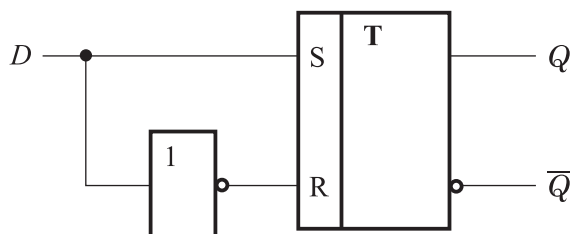
OBR. 12. Tabuľka prechodovej funkcie D-preklápacieho obvodu

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. 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:

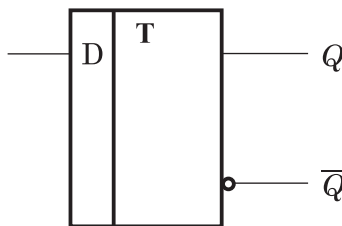
$$Q' = D = D + DQ = (D + Q)D = (D + Q)\bar{D}.$$

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 Mooreov automat $A = (\mathbf{B}, \mathbf{B}^2, \mathbf{B}, \delta, \mu)$, ktorého prechodová funkcia je daná pomocou tabuľky na obr. 15.

		$\overline{J} \quad K$	
		0	1
Q	0	1	1
	1	1	0

OBR. 15. Tabuľka JK-preklápacieho obvodu

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	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

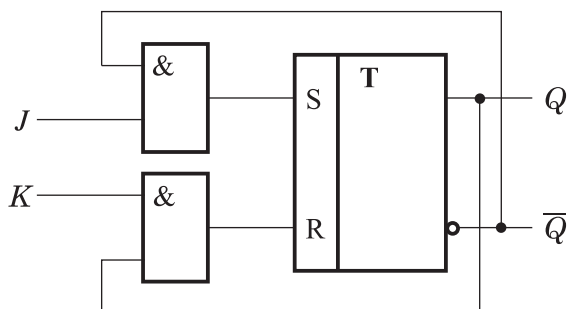
$y \rightarrow Y$	J	K
$0 \rightarrow 1$	1	$\times$
$0 \rightarrow 0$	0	$\times$
$1 \rightarrow 1$	$\times$	0
$1 \rightarrow 0$	$\times$	1

Zjednodušenú formu tejto tabuľky uvádzame na obr. 17.

	J	K
y	Y	$\times$
	$\times$	$\bar{Y}$

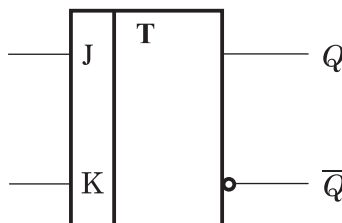
OBR. 17. Zjednodušená tabuľka budiacej funkcie JK-preklápacieho obvodu

Budiaca funkcia JK-preklápacieho obvodu má tvar $Q' = Q\bar{K} + \bar{Q}J$. Jednoduchými úpravami sa možno presvedčiť, že platí aj $Q' = (Q + \bar{Q}J)\bar{K}\bar{Q}$. 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-



OBR. 18. Sekvenčná logická sieť patriaca k JK-preklápaciemu obvodu

preklápacieho obvodu, ak položíme $S = \bar{Q}J$ a $R = KQ$. Všimnime si, že je pritom vylúčená možnosť $(S, R) = (1, 1)$. Teda jednu zo sekvenčných sietí patriacich k JK-preklápaciemu obvodu môžeme navrhnuť 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 Mooreov automat $A = (\mathbf{B}, \mathbf{B}, \mathbf{B}, \delta, \mu)$, ktorého prechodová funkcia je daná pomocou tabuľky na obr. 20.

		T
		$\overline{\hspace{1cm}}$
	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.

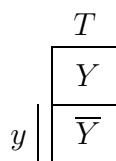
	T
	$\overline{\hspace{1cm}}$
Pm	Pk

OBR. 21. Symbolický zápis tabuľky T-preklápacieho obvodu

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 $y \rightarrow Y$ nastane voľbou $T = Y$, pre $y = 1$ zmena $y \rightarrow Y$ nastane voľbou $T = \bar{Y}$. Toto generovanie budiacej funkcie Y s ohľadom na zmeny $y \rightarrow Y$ v závislosti na vstupe T môžeme znázorniť pomocou obr. 22.

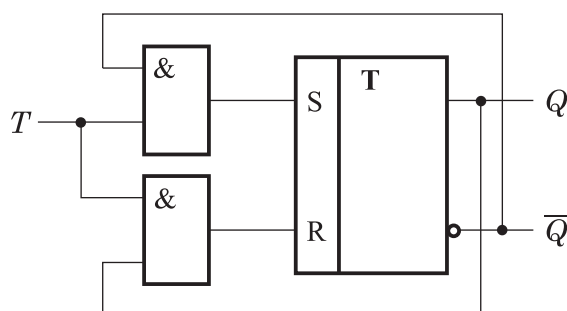
Z UNDF budiacej funkcie Q' odvodíme tvar, ktorý nám pomôže navrhnuť jednu z možných sekvenčných logických sietí patriacich k danému automatu. Z tabuľky prechodovej funkcie čítame:

$$\begin{aligned} Q' &= \bar{Q}T + Q\bar{T} = (\bar{Q}T + Q)(\bar{Q}T + \bar{T}) = (\bar{Q}T + Q)((\bar{Q} + \bar{T})(T + \bar{T})) = \\ &= (\bar{Q}T + Q)(\bar{Q} + \bar{T}) = (\bar{Q}T + Q)\bar{Q}\bar{T}. \end{aligned}$$



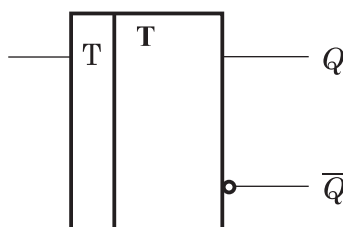
OBR. 22. Zjednodušená tabuľka budiacej funkcie T-preklápacieho obvodu

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

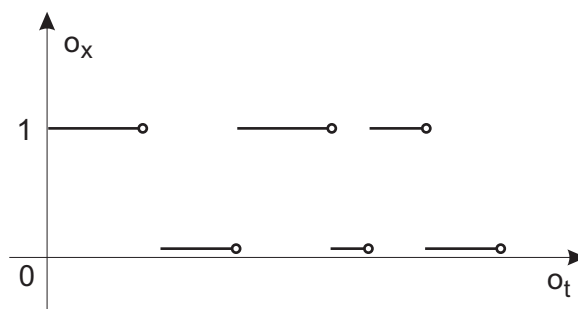
siete patriacej k T-preklápaciemu obvodu, ktorý uvádzame na obr. 23. Všimnime si, že opäť je vylúčená možnosť $(S, R) = (1, 1)$. Pre ľubovoľnú sekvenčnú logickú sieť patriacu k T-preklápaciemu obvodu sa používa schematické označenie z obr. 24.



OBR. 24. T-preklápací obvod

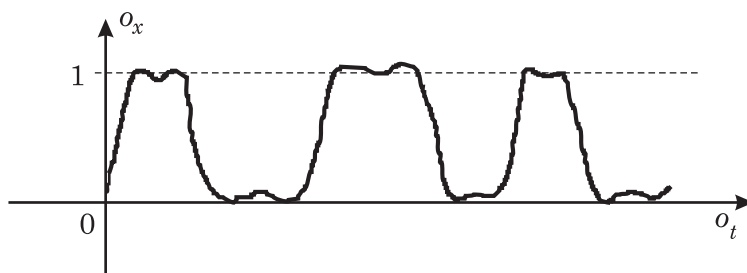
3. Synchronné a asynchronné logické obvody

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.



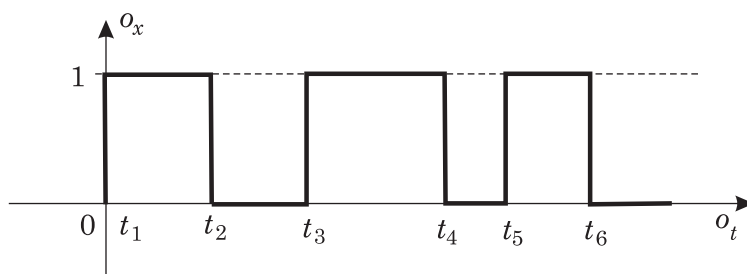
OBR. 25. Graf impulznej funkcie

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

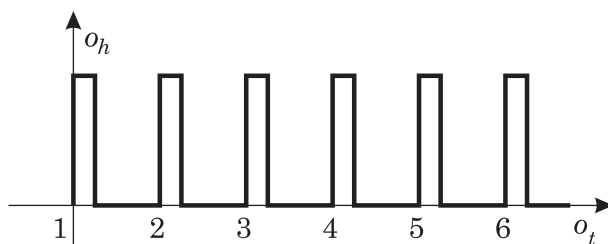
Tu chceme opäť pripomenúť, že vstupy, stavy a výstupy logických obvodov sa menia v časových taktach. Pri takto definovanej impulznej funkcii sa nám núka diskretný čas (takty) určiť v bodoch, kde sa daná funkcia mení buď z 0 na 1 alebo z hodnoty 1 na hodnotu 0. Teda impulzná funkcia z obr. 27 predstavuje postupnosť 101010..., pričom tieto



OBR. 27. Graf idealizovanej impulznej funkcie

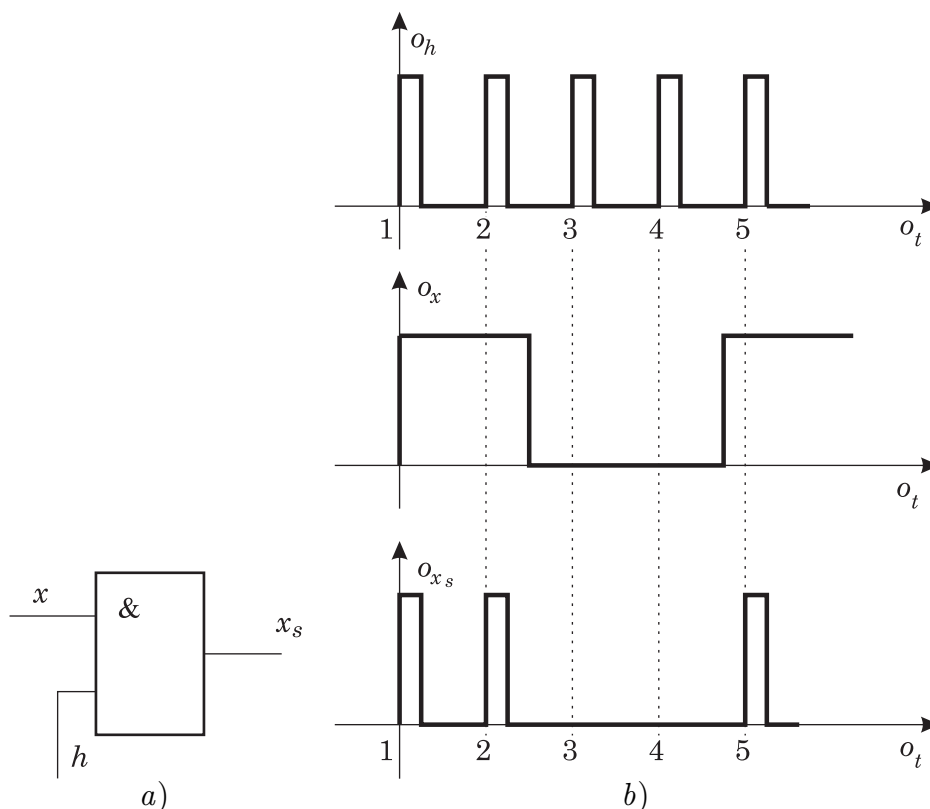
hodnoty sme namerali v bodoch $t_1, t_2, t_3, t_4, t_5, t_6, \dots$ a využili sme predpoklad, že impulzná funkcia je v každom bode spojitá sprava. V budúcnosti budeme body diskretného času $t_1, t_2, t_3, \dots$ jednoducho označovať iba pomocou postupnosti $1, 2, 3, \dots$. Ako vidíme, pri takomto nazeraní na impulznú funkciu, tá môže predstavovať iba postupnosť 101010... 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 diskrétného 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-



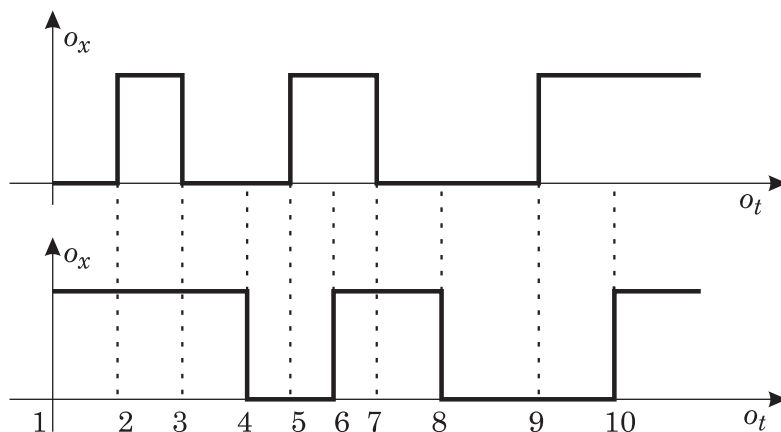
OBR. 28. Generátor časových impulzov

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 $t = 1, 2, 3, \dots$ 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**.



OBR. 29. Synchronizácia signálov

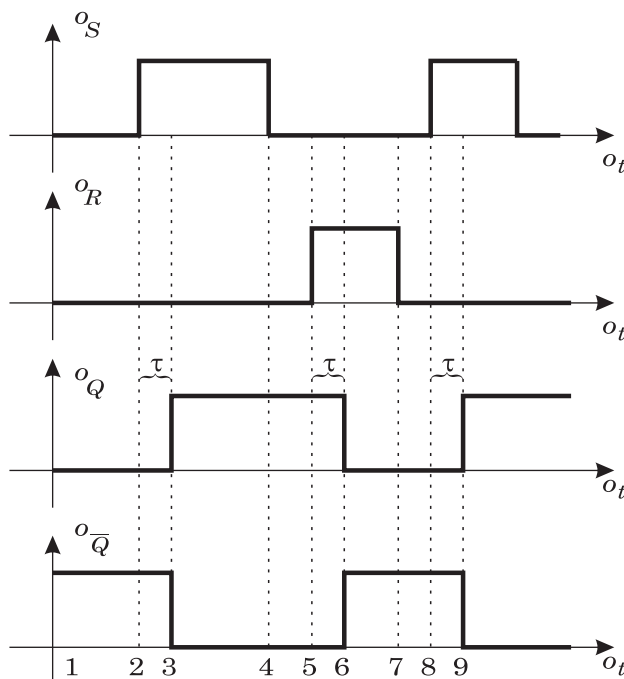
- (2) V prípade, že v logickom obvode uvažujeme o dvoch (a viacerých) impulzných funkciách vstupných a stavových premenných, môžeme za body diskrétného času zobrať všetky body, v ktorých sa mení hociktorá z uvažovaných impulzných funkcií. (pozri obr. 30). Potom impulzná funkcia x z obr. 30 reprezentuje dvojkovú postupnosť 0100110011... a impulzná funkcia y z toho istého obrázka dvojkovú



OBR. 30. Diskrétny čas v asynchrónnych logických obvodoch

postupnosť 1110011001... . Logické obvody, v ktorých je diskretný čas určený zmenami vstupných alebo stavových veličín, nazývame **asynchrónne logické obvody**.

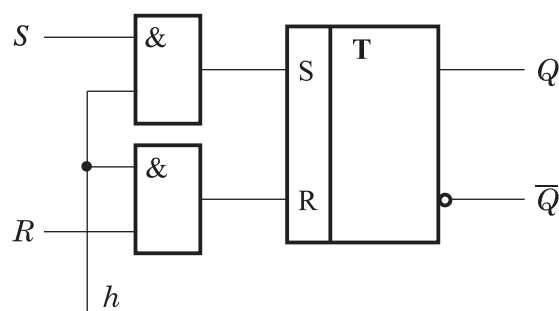
Pri preklápacích obvodoch sme doteraz neuvažovali o synchronizácii ani vstupných a ani stavových veličín. Pri nich sme teda uvažovali o asynchrónnej verzii. Spôsob práce SR-preklápacieho asynchrónneho logického obvodu potom môžeme znázorniť pomocou obr. 31. Pritom predpokladáme, že nový stav je nastavený s istým oneskorením τ , ktoré



OBR. 31. Režim práce asynchrónneho SR-preklápacieho obvodu

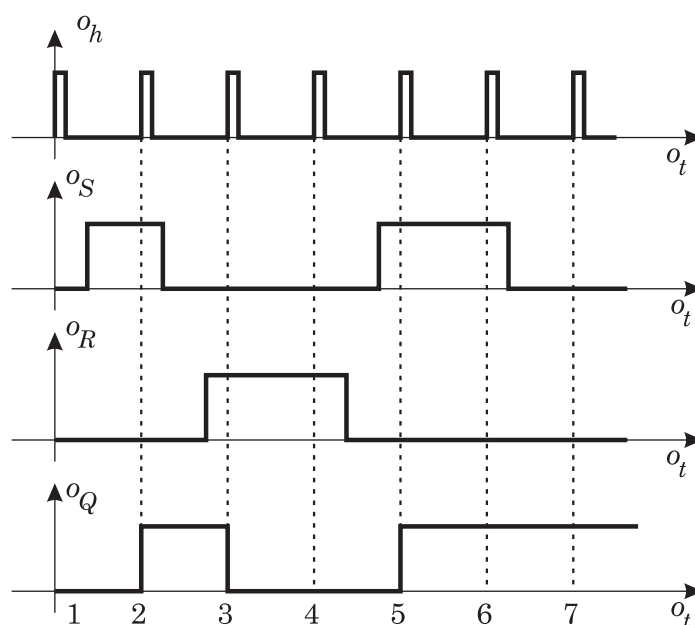
je spôsobené fyzikálnou podstatou SR-preklápacieho obvodu. Ak by sme chceli uvažovať

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

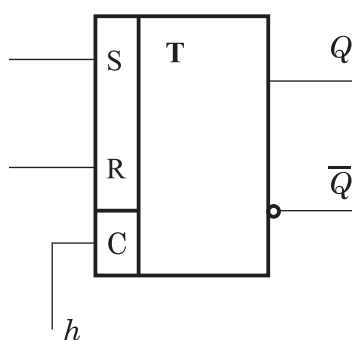


OBR. 32. Synchronizácia vstupných premenných v SR-preklápacom obvode

obr. 33. Pre synchronné SR-preklápacie obvody (ktorých zapojenie je znázornené napríklad na obr. 32) používame štandardné označenie z obr. 34.



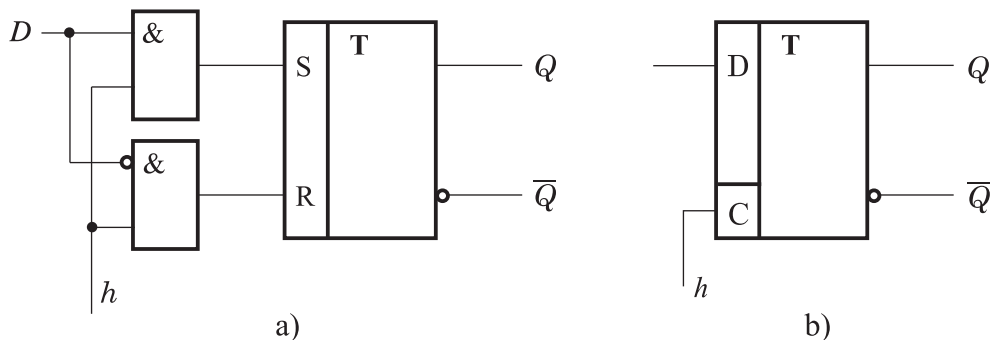
OBR. 33. Režim práce synchronného SR-preklápacieho obvodu



OBR. 34. Synchronný SR-preklápací obvod

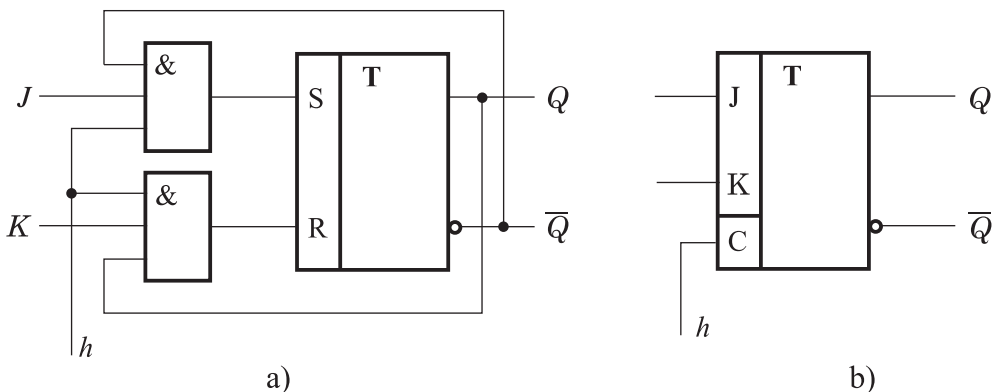
Podobne (ale v značne idealizovanej podobe) môžeme vytvoriť synchronné D-, JK- a T-preklápacie obvody.

Návrh sekvenčnej logickej siete a schematické označenie synchronného D-preklápacieho obvodu uvádzame na obr. 35, časť a), b).



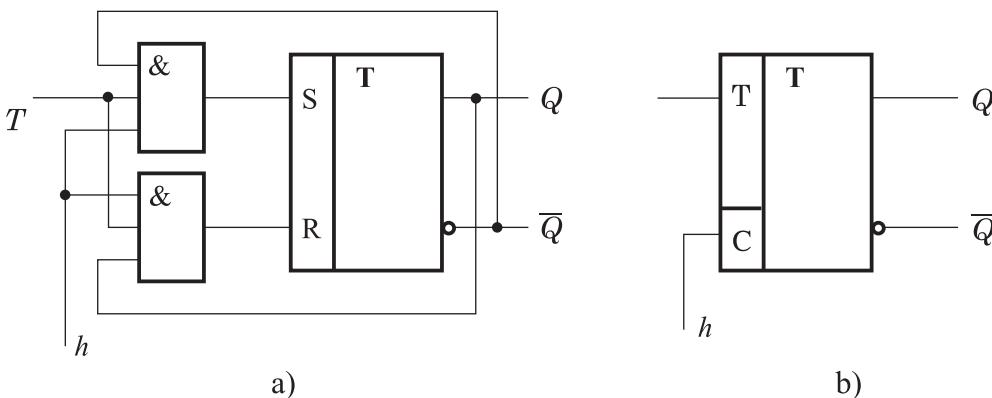
OBR. 35. Synchronný D-preklápací obvod

Návrh sekvenčnej logickej siete a schematické označenie synchronného JK-preklápacieho obvodu je na obr. 36, časť a), b).



OBR. 36. Synchronný JK-preklápací obvod

Návrh sekvenčnej logickej siete a schematické označenie synchronného T-preklápacieho obvodu je na obr. 37, časť a), b).



OBR. 37. Synchronný T-preklápací obvod

Pri našich návrhoch synchronných 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 asynchronné obvody majú iba teoretický význam. Existujú len ich synchronné 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.

Nech teraz $A = (S, X, Z, \delta, \lambda)$ je automat, kde

$$S = \{s_1, \dots, s_k\}, \quad X = \{x_1, \dots, x_n\}, \quad Z = \{z_1, \dots, z_m\}.$$

Prvky množín S, X, Z zakódujeme pomocou príslušných n -tíc núl a jednotiek. Preto definujeme injekcie (kódovacie funkcie) $f : S \rightarrow \mathbf{B}^\alpha, g : X \rightarrow \mathbf{B}^\beta, h : Z \rightarrow \mathbf{B}^\gamma$, kde α, β a γ sú vhodné prirodzené čísla. Aby sme zaručili injektívnosť týchto funkcií, musíme žiadať, aby bolo splnené $k \leq 2^\alpha, n \leq 2^\beta, m \leq 2^\gamma$. 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:

$$2^{\alpha-1} < k \leq 2^\alpha, \quad 2^{\beta-1} < n \leq 2^\beta, \quad 2^{\gamma-1} < m \leq 2^\gamma.$$

Potom definujeme automat $A_{\mathbf{B}} = (f(S), g(X), h(Z), \delta_{\mathbf{B}}, \lambda_{\mathbf{B}})$, kde prechodovú funkciu $\delta_{\mathbf{B}} : f(S) \times g(X) \rightarrow f(S)$ definujeme takto:

$$\delta_{\mathbf{B}}(f(s), g(x)) = f(t) \quad \text{práve vtedy, keď} \quad \delta(s, x) = t.$$

Pre výstupnú funkciu $\lambda_{\mathbf{B}} : f(S) \times g(X) \rightarrow h(Z)$ 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 prvých dvoch 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 $\tilde{\delta}_{\mathbf{B}} : \mathbf{B}^\alpha \times \mathbf{B}^\beta \rightarrow \mathbf{B}^\alpha$ a $\tilde{\lambda}_{\mathbf{B}} : \mathbf{B}^\alpha \times \mathbf{B}^\beta \rightarrow \mathbf{B}^\gamma$ 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 definujeme funkciu $f : \{p, q, r, s\} \rightarrow \mathbf{B}^2$ 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 $g : \{a, b, c, d\} \rightarrow \mathbf{B}^2$ 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 $h : \{z_1, z_2\} \rightarrow \mathbf{B}$ je definovaná predpisom $h(z_1) = 0, h(z_2) = 1$.

	y_2		x_2		
	p	s	a	b	z_1
y_1	q	r	c	d	z_2
	a)		b)		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 $A_{\mathbf{B}} = \bar{A}_{\mathbf{B}}$. Príslušná prechodová a výstupná funkcia sú v tabuľke 5. Táto

TABUĽKA 5. Tabuľka kódového ekvivalentu z príkladu 4.2

$(y_1, y_2) \backslash (x_1, x_2)$	(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

	x_1	x_2		x_1	x_2		x_1	x_2	
	0	0	1	1	0	0	1	1	0
y_1	1	0	0	0	0	1	0	0	1
y_2	1	1	1	0	1	1	1	0	1
	0	1	0	1	0	1	0	1	1
	a) Y_1		b) Y_2		c) z				

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 $A_{\mathbf{B}}$, ktorý pokrýva automat A .

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

RIEŠENIE. Stavý, vstupy a výstupy zakódujeme pomocou kódovacích tabuliek na obr. 40, časť a), b), c). Potom kódový ekvivalent $A_{\mathbf{B}}$ automatu A je daný pomocou tabuľky 7.

	$\overline{y_2}$		$\overline{x_2}$		$\overline{z_2}$												
	<table> <tr><td>p</td><td>—</td></tr> <tr><td>r</td><td>s</td></tr> </table>	p	—	r	s		<table> <tr><td>a</td><td>—</td></tr> <tr><td>b</td><td>c</td></tr> </table>	a	—	b	c		<table> <tr><td>u</td><td>—</td></tr> <tr><td>v</td><td>w</td></tr> </table>	u	—	v	w
p	—																
r	s																
a	—																
b	c																
u	—																
v	w																
y_1	a)	x_1	b)	z_1	c)												

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

TABUĽKA 7. Tabuľka kódového ekvivalentu $A_{\mathbf{B}}$ z príkladu 4.3

	$(0, 0)$	$(1, 0)$	$(1, 1)$
$(0, 0)$	$(0, 0)/(0, 0)$	$(1, 0)/(0, 0)$	$(1, 1)/(1, 0)$
$(1, 0)$	$(0, 0)/(0, 0)$	$(1, 1)/(0, 0)$	$(1, 1)/(1, 1)$
$(1, 1)$	$(1, 1)/(0, 0)$	$(0, 0)/(1, 0)$	$(1, 0)/(1, 1)$

$\check{A}_{\mathbf{B}} = (\mathbf{B}^2, \mathbf{B}^2, \mathbf{B}^2, \check{\delta}_{\mathbf{B}}, \check{\lambda}_{\mathbf{B}})$, ktorý obsahuje automat $A_{\mathbf{B}}$ ako svoj podautomat. Tento neúplne špecifikovaný automat je daný pomocou tabuľky 8.

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, ktorý pokrýva daný automat A . Toto doplnenie urobíme tak, aby výsledné funkcie mali normálne disjunktívne formy s najmenším možným počtom písmen. Teda tie pomlčky,

TABUĽKA 8. Tabuľka dvojkového automatu $\check{A}_B$ z príkladu 4.3

$(y_1, y_2) \backslash (x_1, x_2)$	(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)	(-, -)/(-, -)	(-, -)/(-, -)	(-, -)/(-, -)	(-, -)/(-, -)

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:

$$Y_1 = x_2 + \bar{y}_2 x_1 + y_2 \bar{x}_1,$$

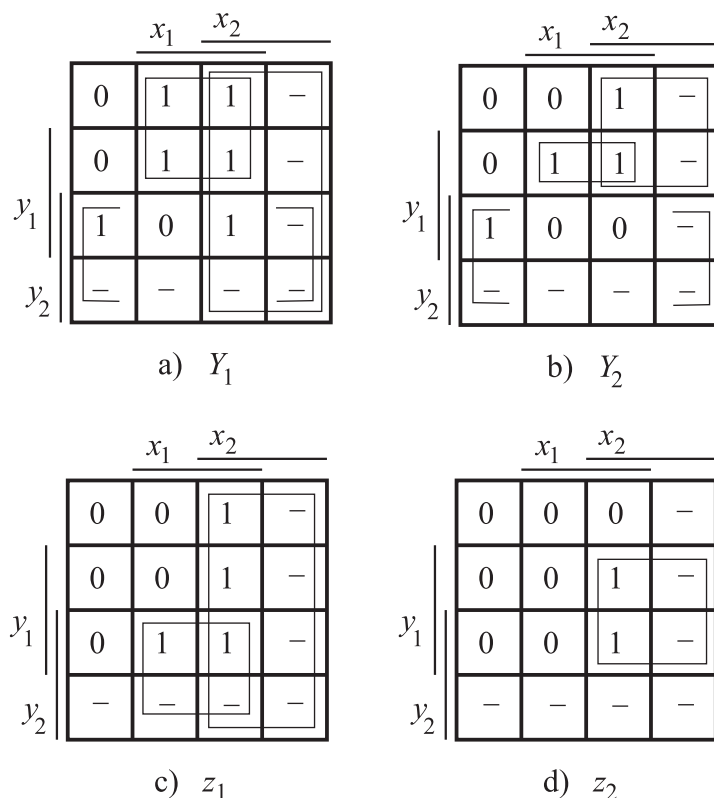
$$Y_2 = \bar{y}_2 x_2 + y_1 \bar{y}_2 x_1 + y_2 \bar{x}_1,$$

$$z_1 = x_2 + y_2 x_1,$$

$$z_2 = y_1 x_2.$$

Tým je však určený dvojkový automat $\check{A}_B$ 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



OBR. 41. Karnaughove mapy prechodových a výstupných funkcií z príkladu 4.3

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 synchronných logických obvodov, body diskrétného 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. Navrhujeme sekvenčnú logickú sieť, ktorá reprezentuje fyzikálnu realizáciu automatu $\tilde{A}_B = A_B$ z príkladu 4.2, pričom budiacu funkciu Y_1 budeme generovať pomocou synchronného SR-preklápacieho obvodu a funkciu Y_2 pomocou synchronného D-preklápacieho obvodu. Rozmiestnenie vrcholov a štruktúru pamätevej 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.

	x_1	x_2		
	0	0	1	1
y_1	1	0	0	0
	1	1	1	0
y_2	0	1	0	1
a) Y_1				
	x_1	x_2		
	0	0	1	1
y_1	x	0	0	0
	x	x	x	0
y_2	0	1	0	1
b) S				
	x_1	x_2		
	x	x	0	0
y_1	0	1	1	1
	0	0	0	1
y_2	x	0	x	0
c) R				

OBR. 42. Funkcia Y_1 a vstupné funkcie SR-preklápacieho obvodu z príkladu 4.4

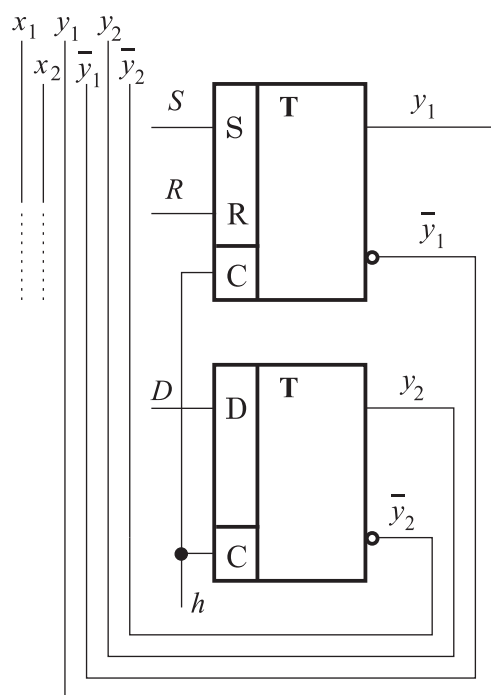
Uvažujme najprv o budiacej funkcii Y_1 . Jej Karnaughova mapa (v ďalšom kvôli stručnosti len tabuľka) je na obr. 39, časť a), na obr. 42, časť a) ju opakujeme. V tejto tabuľke je pre nás dôležitá zmena starého stavu y_1 na nový stav Y_1 v závislosti od ďalších premenných y_2, x_1, x_2 . Ako už bolo povedané, táto zmena sa má realizovať v SR-preklápacom 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 $0 \rightarrow Y_1$. 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 $1 \rightarrow Y_1$ 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 .

- Pri zmene $1 \rightarrow Y_1$ (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.
- Pri zmene $0 \rightarrow Y_1$ (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 $\times$.



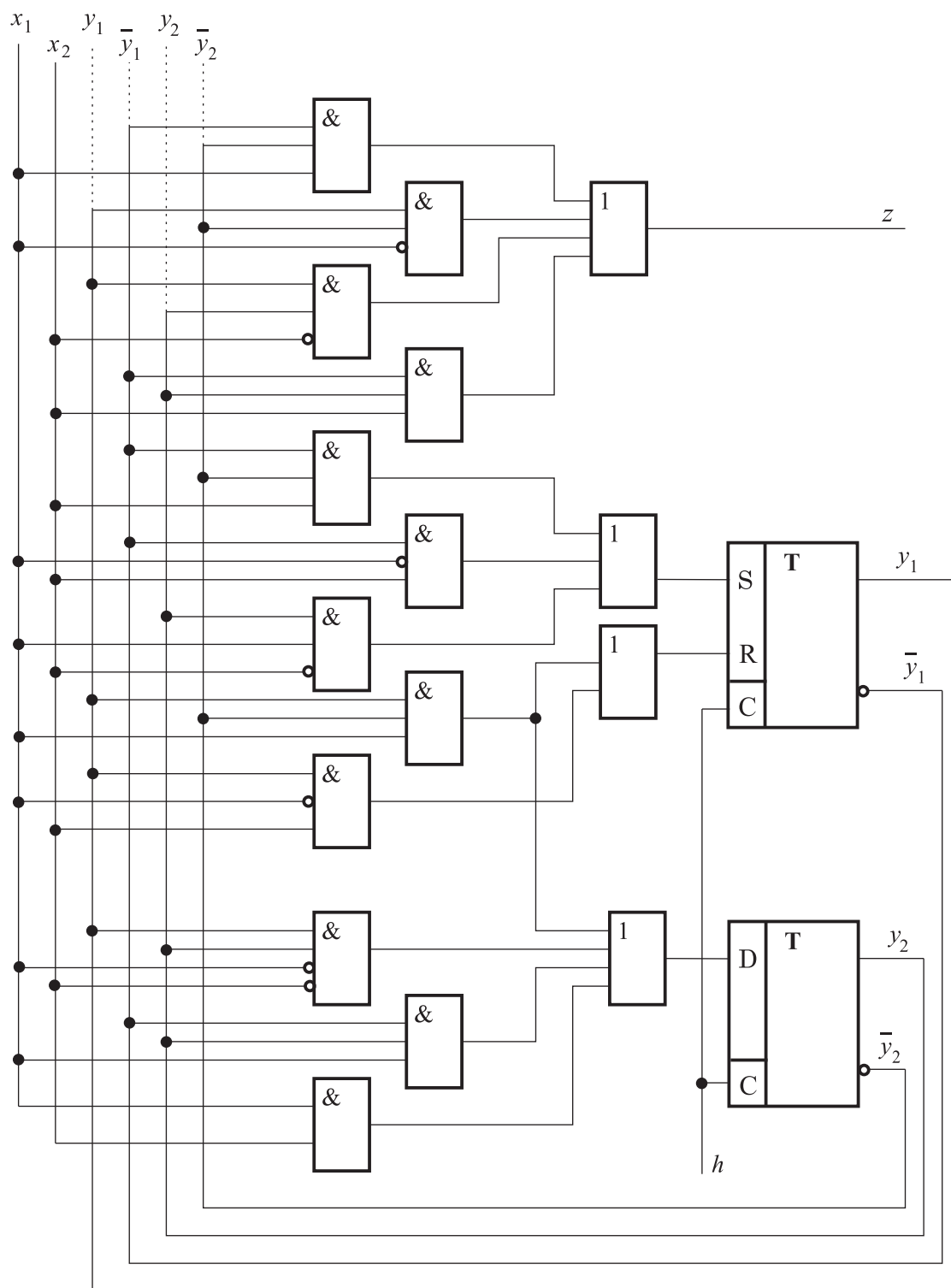
OBR. 43. Základné vrcholy sekvenčnej logickej siete z príkladu 4.4

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

× nahradíme nulami. Potom dostávame

$$S = \bar{y}_1 \bar{y}_2 x_2 + \bar{y}_1 \bar{x}_1 x_2 + y_2 x_1 \bar{x}_2$$

$$R = y_1 \bar{y}_2 x_1 + y_1 \bar{x}_1 x_2.$$



OBR. 44. Logická sieť z príkladu 4.4

Budiacu funkciu Y_2 budeme generovať pomocou D-preklápacieho obvodu. Nakoľko D-preklápací obvod je oneskorovací člen, situácia je veľmi jednoduchá, lebo stačí položiť

$$D = Y_2 = y_1 y_2 \bar{x}_1 \bar{x}_2 + y_1 \bar{y}_2 x_1 + \bar{y}_1 y_2 x_1 + x_1 x_2.$$

(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

$$z = \bar{y}_1 \bar{y}_2 x_1 + y_1 \bar{y}_2 \bar{x}_1 + y_1 y_2 \bar{x}_2 + \bar{y}_1 y_2 x_2.$$

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 navrhujeme sekvenčnú logickú sieť, ktorá reprezentuje fyzikálnu realizáciu dvojkového automatu $\tilde{A}_B$ z príkladu 4.3. Budiacu funkciu Y_1 budeme generovať pomocou synchronného JK-preklápacieho obvodu a funkciu Y_2 pomocou synchronného 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

	x_1	x_2		
	0	1	1	1
y_1	0	1	1	1
y_2	1	0	1	1
	1	0	1	1

a) Y_1

	x_1	x_2		
	0	1	1	1
y_1	x	x	x	x
y_2	x	x	x	x
	1	0	1	1

b) J

	x_1	x_2		
	x	x	x	x
y_1	1	0	0	0
y_2	0	1	0	0
	x	x	x	x

c) K

OBR. 45. Funkcia Y_1 a vstupné funkcie JK-preklápacieho obvodu z príkladu 4.5

doplnenú tabuľku funkcie Y_1 . Ďalej v našom postupe využijeme tabuľku 3.

Z tabuľky 3 vidieť, že pri zmene $0 \rightarrow Y_1$ 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 $0 \rightarrow Y_1$. Pri zmene $1 \rightarrow Y_1$ 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 $1 \rightarrow Y_1$ 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 $0 \rightarrow Y_1$ 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 možným počtom písmen. Z obr. 45, časť b), c) plynie, že riešením je

$$J = x_2 + y_2 \bar{x}_1 + \bar{y}_2 x_1$$

$$K = \bar{y}_2 \bar{x}_1 \bar{x}_2 + y_2 x_1 \bar{x}_2.$$

Funkciu Y_2 budeme generovať pomocou T-preklápacieho obvodu, teda musíme navrhnuť jeho vstupnú funkciu T . Pritom využijeme doplnenú tabuľku funkcie Y_2 z obr. 41, časť b) (uvádzame ju na obr. 46, časť a)) a zjednodušenú tabuľku pre vstupnú funkciu

	x_1	x_2		x_1	x_2	
	0	0	1	1		
	0	1	1	1		
y_1	1	0	0	1		
y_2	1	0	0	1		

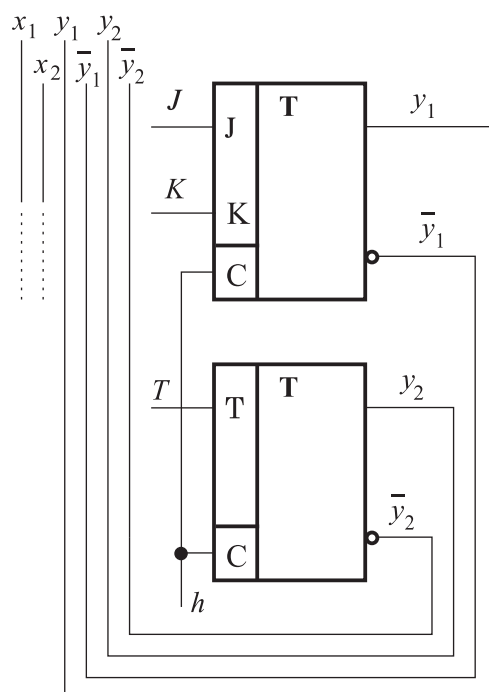
a) Y_2

	x_1	x_2	
	0	0	1
	0	1	1
y_1	0	1	1
y_2	0	1	1

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 $0 \rightarrow Y_2$ 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 $0 \rightarrow Y_2$). Pri zmene $1 \rightarrow Y_2$ 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).



OBR. 47. Základné vrcholy logickej siete z príkladu 4.5

Z tabuľky funkcie T potom dostávame

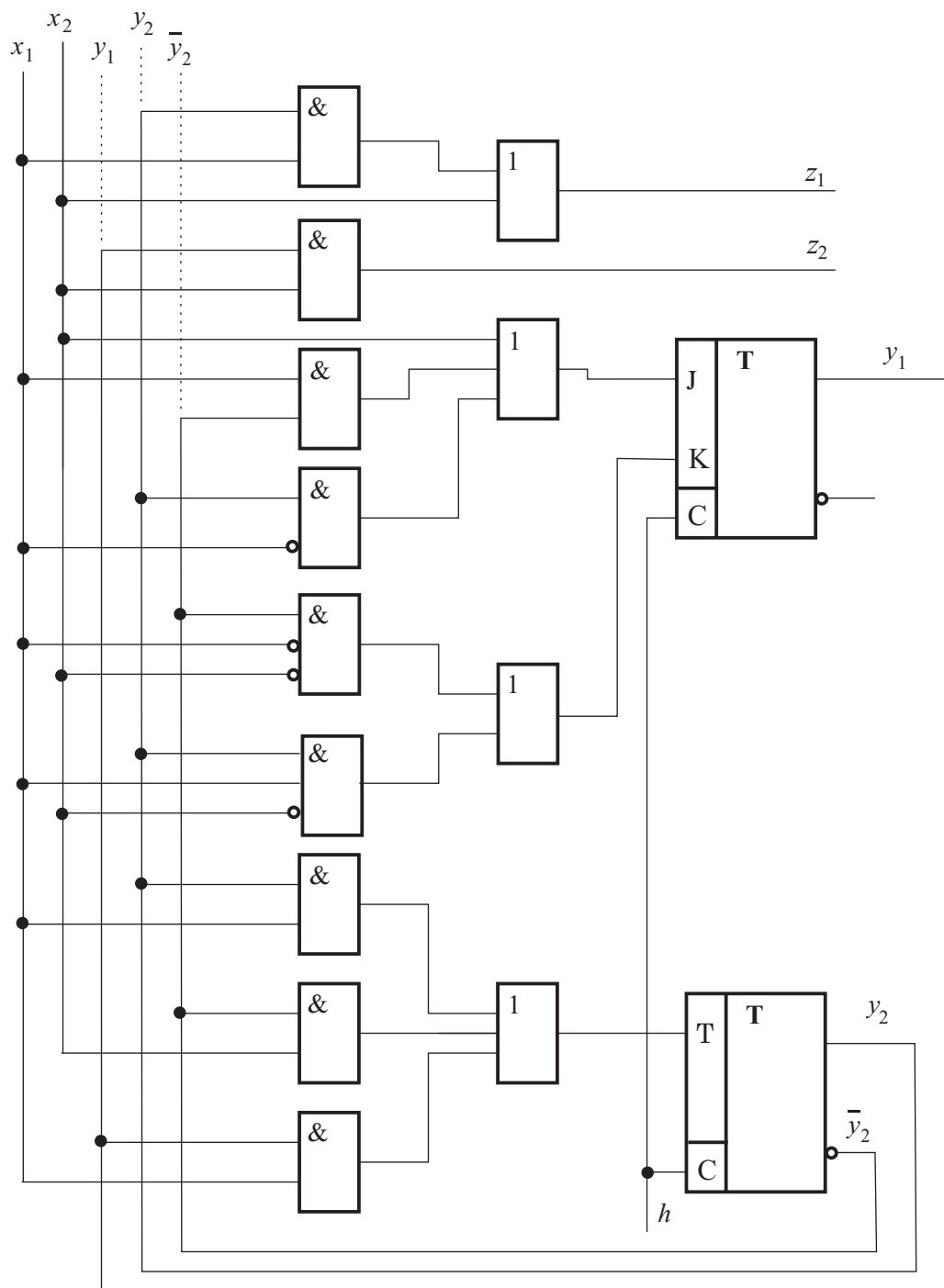
$$T = \bar{y}_2 x_2 + y_1 x_1 + y_2 x_1.$$

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.



OBR. 48. Logická sieť z príkladu 4.5

Ukážeme teraz výhodnejší spôsob riešenia tejto úlohy. V predošlom postupe sme si funkcie Y_1 a Y_2 najprv špecifikovali. Teraz ich necháme neúplne špecifikované a ich špecifikáciu vykonáme až nakoniec. Teda máme neúplne špecifikované funkcie Y_1 a Y_2 z obr. 41.

Uvádžeme ich na obr. 49, pričom pomlčky sme nahradili krížikmi. Funkcie J a K vytvá-

	x_1	x_2		
	0	1	1	×
y_1	0	1	1	×
	1	0	1	×
y_2	×	×	×	×
	Y_1			

	x_1	x_2		
	0	0	1	×
y_1	0	1	1	×
	1	0	0	×
y_2	×	×	×	×
	Y_2			

OB. 49. Prechodové funkcie z príkladu 4.5

rame z tabuľky funkcie Y_1 . Z tabuľky 3 vidno, že pri zmene $0 \rightarrow Y_1$ je $J = Y_1$. Teda do tabuľky funkcie J stačí preniesť celý prvý a štvrtý riadok tabuľky funkcie Y_1 (tu totiž dochádza k zmene $0 \rightarrow Y_1$). Pri zmene $1 \rightarrow Y_1$ môže byť vstup J ľubovoľný. Preto v druhom a treťom riadku tabuľky funkcie J bude $\times$ (pozri obr. 50, časť a)).

Teraz funkcia K . Pri zmene $1 \rightarrow Y_1$ je $K = \bar{Y}_1$. Teda hodnoty z druhého a tretieho

	x_1	x_2		
	0	1	1	×
y_1	×	×	×	×
	×	×	×	×
y_2	×	×	×	×
	a) J			

	x_1	x_2		
	×	×	×	×
y_1	1	0	0	×
	0	1	0	×
y_2	×	×	×	×
	b) K			

OB. 50. Vstupné funkcie JK-preklápacieho obvodu

riadku tabuľky funkcie Y_1 preniesieme do príslušného riadku tabuľky funkcie K , pričom zmeníme 0 na 1 a 1 na 0. Samozrejme krížik sa preniesie ako krížik. Pri zmene $0 \rightarrow Y_1$ je $K = \times$. To sa prejaví v prvom a štvrtom riadku tabuľky funkcie K - pozri obr. 50, časť b)).

Teraz všetky krížiky v tabuľkách funkcií J , K na obr. 50 nahradíme nulami a jednotkami tak, aby výsledné funkcie mali NDF s najmenším možným počtom písmen. Z obr. 50 plynie, že riešením je

$$J = x_1$$

$$K = \bar{y}_2 \bar{x}_1 + y_2 x_1 \bar{x}_2$$

Pre generovanie funkcie Y_2 pomocou T-preklápacieho obvodu navrhujeme jeho vstupnú funkciu T . Opäť budeme vychádzať z tabuľky neúplne špecifikovanej funkcie Y_2 na obr. 49. Využijeme tabuľku pre vstupnú funkciu T T-preklápacieho obvodu z obr. 22. Keďže zmene $0 \rightarrow Y_2$ zodpovedá prvý a druhý riadok tabuľky funkcie Y_2 , tieto dva riadky prepisujeme priamo do tabuľky funkcie T . Pri zmene $1 \rightarrow Y_2$ je $T = \bar{Y}_2$. Teda v treťom a štvrtom riadku tabuľky funkcie Y_2 zmeníme 0 na 1 a 1 na 0 (krížiky zostanú opäť nezmenené) a preniesieme do zodpovedajúcich riadkov tabuľky funkcie T (obr. 51).

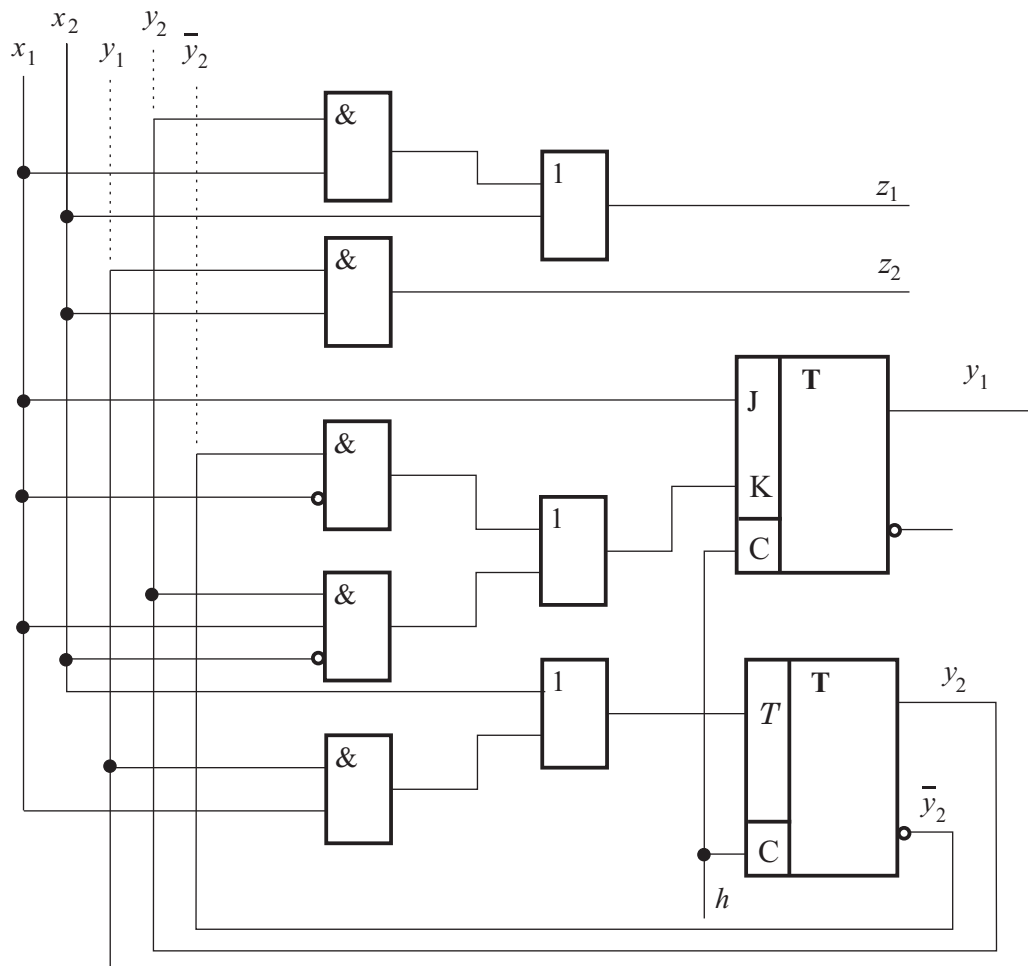
		x_1	x_2	
		0	1	$\times$
y_1	0	1	1	$\times$
	0	1	1	$\times$
y_2	$\times$	$\times$	$\times$	$\times$
		T		

OBR. 51. Vstupná funkcia T-preklápacieho obvodu

Z tejto tabuľky plynie

$$T = y_1 x_1 + x_2$$

Výstupné funkcie ostávajú nezmenené. Príslušná logická sieť je na obr. 52.



OBR. 52. Logická sieť z príkladu 4.5

Všimnime si, že tento postup dal jednoduchšie riešenie ako v predchádzajúcej časti, kde sme najprv špecifikovali funkcie Y_1 a Y_2 a až potom sme pokračovali v riešení. Ukazuje sa, že je výhodnejšie pri neúplne špecifikovaných budiacich funkciách nechať ich neúplne

špecifikované, vyrobiť príslušné (neúplne špecifikované) vstupné funkcie preklápacích obvodov a až nakoniec všetko došpecifikovať tak, aby príslušné NDF boli minimálne. ■

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.

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 rámečkom. 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 ***Huffmanov automat***, ak pre každé $s \in S$ a každé $x \in X$ existuje $n > 0$ také, že $\hat{\delta}(s, x^n) = \hat{\delta}(s, x^{n+1})$ (v tomto prípade x^n označuje vstupné slovo $xx \dots x$, v ktorom sa nachádza n písmen x).

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. 53. Teraz ukážeme, že daný automat je fundamentálny.

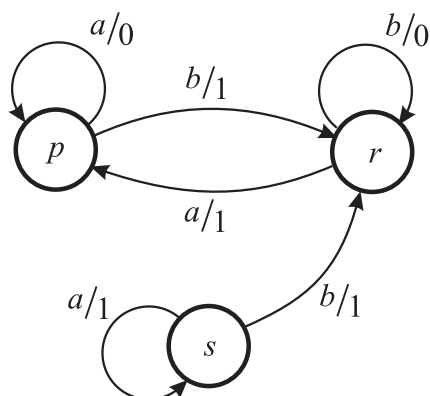
RIEŠENIE. Dosť ľahko overíme, že

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

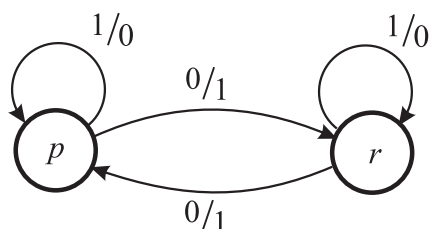
$$\begin{aligned}
\hat{\delta}(p, a) &= \hat{\delta}(p, a^2) = p, \\
\hat{\delta}(p, b) &= \hat{\delta}(p, b^2) = r, \\
\hat{\delta}(r, a) &= \hat{\delta}(r, a^2) = p, \\
\hat{\delta}(r, b) &= \hat{\delta}(r, b^2) = r, \\
\hat{\delta}(s, a) &= \hat{\delta}(s, a^2) = s, \\
\hat{\delta}(s, b) &= \hat{\delta}(s, b^2) = r.
\end{aligned}$$

Z toho už vyplýva, že daný automat je fundamentálny. ■



OBR. 53. Graf automatu z príkladu 4.6

Príklad 4.7. Automat A je daný grafom na obr. 54. 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. 54. 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. Stav 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 $y_i \rightarrow \tilde{y}_i$ a $y_j \rightarrow \tilde{y}_j$ 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

	$(0,0)$	$(1,0)$	$(1,1)$	$(0,1)$
$(0,0)$	$(1,1)$	$(0,0)$	$(0,0)$	$(0,1)$
$(1,0)$	$(1,1)$	$(1,0)$	—	$(1,0)$
$(1,1)$	$(1,1)$	$(1,0)$	$(0,0)$	$(1,0)$
$(0,1)$	$(0,1)$	$(1,0)$	$(0,0)$	$(0,1)$

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 najprv zmenila druhá zložka stavu) alebo do stavu $(1,0)$ (ak sa najprv 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 $(0,1) \rightarrow (1,0)$. 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. Zás ide o kritický súbeh. Pokiaľ by šlo o Mooreov 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 $\hat{\delta}((0, 1), (1, 0)^2) = \delta(\delta((0, 1), (1, 0)), (1, 0)) = \delta((1, 1), (1, 0)) = (1, 0)$. Takýmto postupom sa hovorí **organizácia postupného prechodu medzi stavmi**.

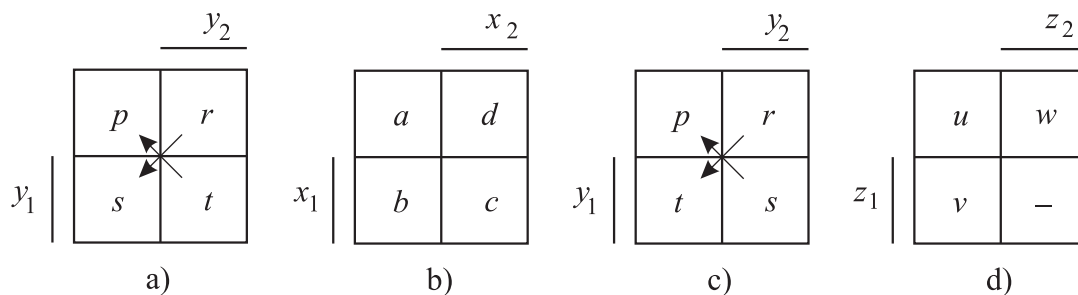
3. Pri vstupe $(1, 1)$ a stave $(1, 1)$ tiež nastáva súbeh, lebo $\delta((1, 1), (1, 1)) = (0, 0)$. V tomto prípade však stačí nedefinovanú hodnotu $\delta((1, 0), (1, 1))$ nahradiť hodnotou $(0, 0)$. Potom pri vstupe $(1, 1)$ sa všetky stavy dostávajú do stabilného stavu $(0, 0)$. V takomto prípade hovoríme o **nekritickom súbehu**. Nekritické súbehy spravidla nemusíme odstraňovať. ■

Príklad 4.9. Navrhujeme fyzikálnu realizáciu neúplne špecifikovaného Mooreovho automatu, ktorý je daný pomocou tabuľky 11.

TABUĽKA 11. Automat z príkladu 4.9

	a	b	c	d	μ
p	$\boxed{p}$	r	$-$	$\boxed{p}$	u
r	s	$\boxed{r}$	t	$-$	u
s	$\boxed{s}$	$-$	$-$	p	v
t	$-$	$-$	$\boxed{t}$	p	w

RIEŠENIE. Stavy zakódujeme pomocou kódovacej tabuľky na obr. 55 časť a). Touto tabuľkou je daná kódovacia funkcia $f : \{p, r, s, t\} \rightarrow \mathbf{B}^2$, kde $f(p) = (0, 0)$, $f(r) = (0, 1)$, $f(s) = (1, 0)$, $f(t) = (1, 1)$. Na obr. 55 časť b) je kódovacia tabuľka vstupnej abecedy. Tabuľku prechodovej funkcie dvojkového automatu $A_{\mathbf{B}}$, ktorý pokrýva daný automat,



OBR. 55. Kódovacie tabuľky z príkladu 4.9

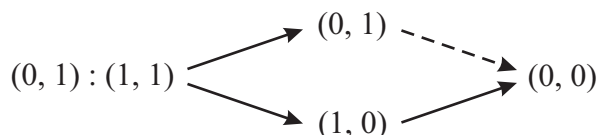
uvádzame v tabuľke 12. Z tejto tabuľky vidíme, že k súbehu dochádza pri vstupe $(0, 0)$ pri prechode zo stavu $(0, 1)$ do stavu $(1, 0)$, lebo $\delta_{\mathbf{B}}((0, 1), (0, 0)) = (1, 0)$. V tomto prípade budeme písať $(0, 0) : (0, 1) \rightarrow (1, 0)$. Ďalší súbeh má tvar $(0, 1) : (1, 1) \rightarrow (0, 0)$. Teraz sa budeme zaoberať otázkou, či tieto súbehy sú kritické.

Zaoberajme sa najprv súbehom $(0, 1) : (1, 1) \rightarrow (0, 0)$, 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

TABUĽKA 12. Tabuľka prechodovej funkcie automatu z príkladu 4.9

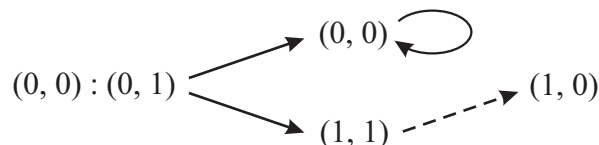
	(0, 0)	(1, 0)	(1, 1)	(0, 1)
(0, 0)	(0,0)	(0, 1)	–	(0,0)
(0, 1)	(1, 0)	(0,1)	(1, 1)	–
(1, 0)	(1,0)	–	–	(0, 0)
(1, 1)	–	–	(1,1)	(0, 0)

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 požadovaného stavu. Tento súbeh nie je kritický. Uvedený postup budeme znázorňovať takto:



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:

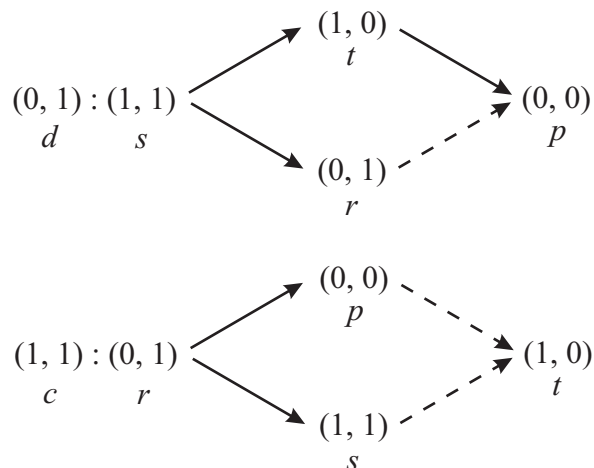


Vidíme, že zo stavu (1, 1) môžeme pri vstupe (0, 0) dodefinovať prechod do požadovaného stavu (1, 0). Ale stav (0, 0) je pri vstupe (0, 0) stabilný. Pri pevnom vstupe (0, 0) sa automat zo stavu (0, 0) už nedostane, čiže nemáme zabezbečený požadovaný prechod do stavu (1, 0). Preto je tento súbeh kritický.

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. 55, 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. 55, 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. 55, časť c). Týmto prechodom v kódovom ekvivalente zodpovedajú súbehy:

$(0, 1) : (1, 1) \longrightarrow (0, 0) \quad (d : s \longrightarrow p \text{ v pôvodnom automate}),$
 $(1, 1) : (0, 1) \longrightarrow (1, 0) \quad (c : r \longrightarrow t \text{ v pôvodnom automate}).$

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

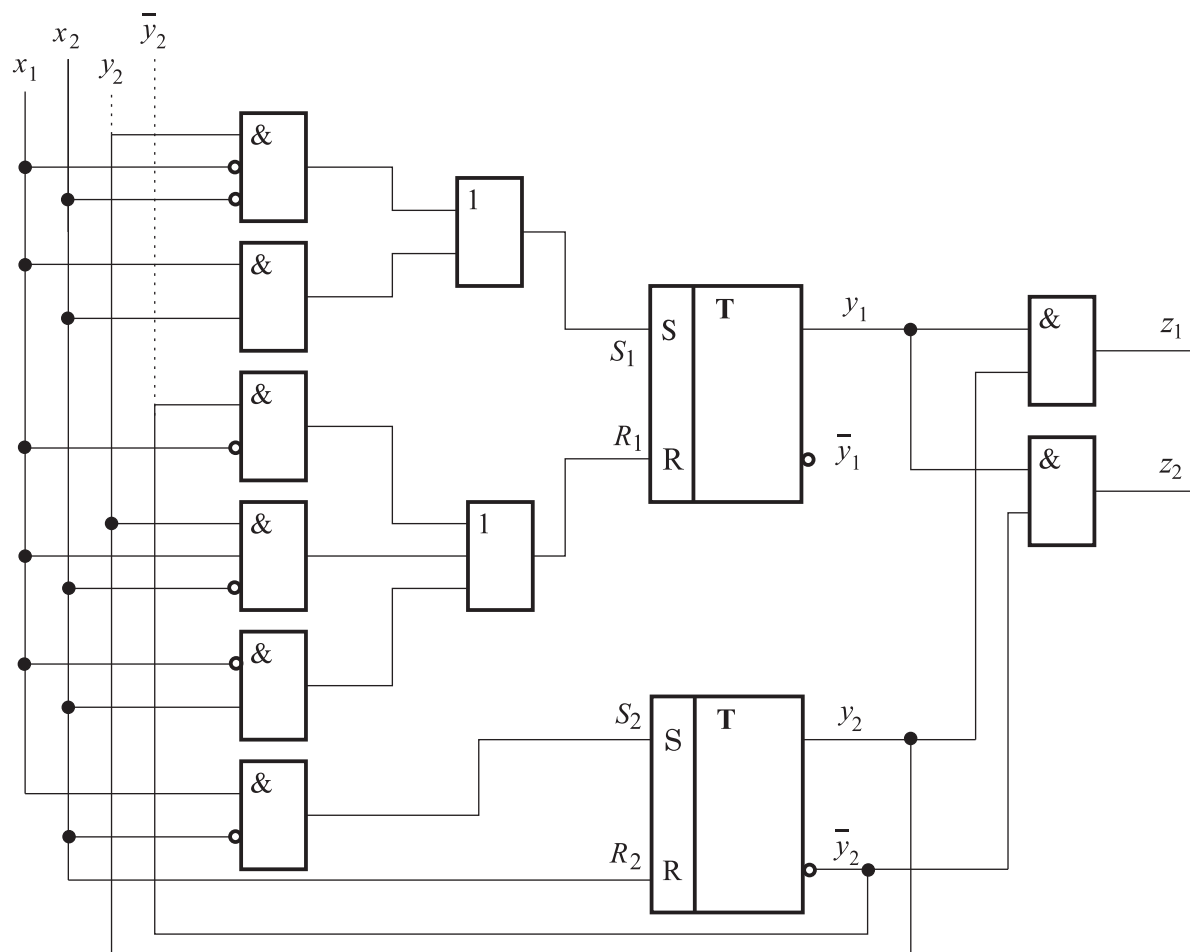
TABUĽKA 13. Tabuľka automatu z príkladu 4.9

$(y_1, y_2) \backslash (x_1, x_2)$	$(0, 0)$	$(1, 0)$	$(1, 1)$	$(0, 1)$	(z_1, z_2)
$p \dots (0, 0)$	$(0, 0)$	$(0, 1)$	$\boxed{(1, 0)}$	$(0, 0)$	$(0, 0)$
$r \dots (0, 1)$	$(1, 1)$	$(0, 1)$	$(1, 0)$	$\boxed{(0, 0)}$	$(0, 0)$
$s \dots (1, 1)$	$(1, 1)$	$(-, -)$	$\boxed{(1, 0)}$	$(0, 0)$	$(1, 0)$
$t \dots (1, 0)$	$(-, -)$	$(-, -)$	$(1, 0)$	$(0, 0)$	$(0, 1)$

funkcií, ktoré sú dané kódovacími tabuľkami na obr. 55, 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 definujeme $\delta_{\mathbf{B}}((1, 1), (1, 0)) = (0, 1)$, $\delta_{\mathbf{B}}((1, 0), (0, 0)) = (0, 0)$, $\delta_{\mathbf{B}}((1, 0), (1, 0)) = (1, 1)$. Tabuľky funkcií Y_1, Y_2 už s uvedeným dodefinovaním sú na obr. 56, časť a), b). Tabuľky výstupných funkcií z_1, z_2 uvádzame na obr. 56, 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

Príslušnú logickú sieť uvádzame na obr. 58. ■



OBR. 58. 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 navrhujeme fyzikálnu realizáciu Mooreovho 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.

$$y_1 \left| \begin{array}{c|c} \hline & y_2 \\ \hline A & D \\ C & B \\ \hline \end{array} \right.,$$

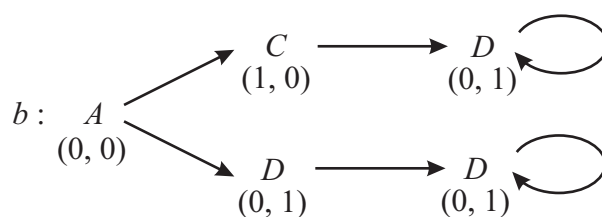
(2) A je v uhlopriečke s C , a teda B je v uhlopriečke s D , napr.

$$y_1 \left| \begin{array}{c|c} \hline & y_2 \\ \hline A & D \\ B & C \\ \hline \end{array} \right.,$$

(3) A je v uhlopriečke s D , a teda B je v uhlopriečke s C , napr.

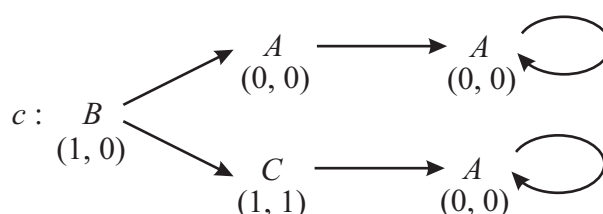
$$y_1 \left| \begin{array}{c|c} \hline & y_2 \\ \hline A & C \\ B & D \\ \hline \end{array} \right..$$

V prvom prípade v kódovom ekvivalente dochádza k súbehu pri prechode medzi stavmi, ktorý zodpovedá prechodu $b : A \rightarrow B$. Tento súbeh prebieha takto:



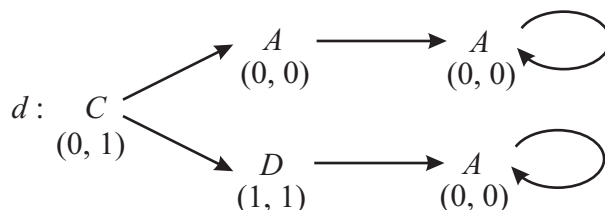
Stav D je pri vstupe b stabilný

V druhom prípade dochádza v dvojkovom ekvivalente k súbehu pri prechode medzi stavmi, ktorý zodpovedá prechodu $c : B \rightarrow D$. Tento súbeh prebieha takto:



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.

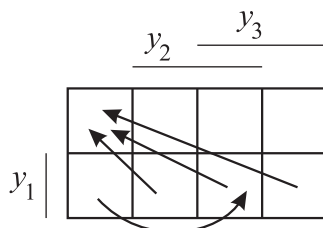
V treťom prípade v dvojkovom automate dostávame súbeh napríklad pri prechode medzi stavmi, ktorý zodpovedá prechodu $d : C \longrightarrow B$. Tento prechod má takýto priebeh:



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 prípade 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 navrhne 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. 59. V našom príklade sú všetky možné prechody medzi stavmi tieto:



OBR. 59. 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. 60. Vidíme, že pri tomto kódovaní budú prechody

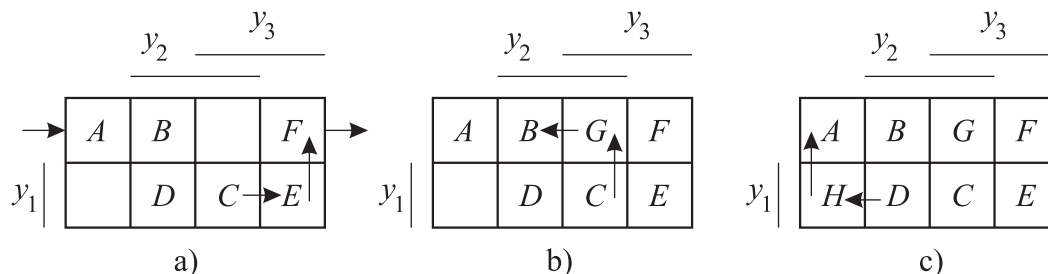
		y_2 y_3	
y_1		A	B
		D	C

OBR. 60. Kódovacia tabuľka množiny stavov z príkladu 4.10

$b : A \longrightarrow B$, $a : B \longrightarrow A$, $b : C \longrightarrow D$, $a : D \longrightarrow C$, $c : B \longrightarrow D$, 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. 61, časť a). Potom musíme žiadať, aby



OBR. 61. Organizácia bezsúbehových prechodov medzi stavmi v príklade 4.10

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: $\delta(C, c) = E$, $\delta(E, c) = F$, $\delta(F, c) = A$.

Ďalej budeme organizovať prechod $d : C \longrightarrow B$. Pre tento prechod stačí pridať jeden stav G , ktorý zakódujeme pomocou kódovacej tabuľky na obr. 61, 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. 61, 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. 61, časť c). Výstupy kódovať nemusíme a vstupnú abecedu zakódujeme pomocou kódovacej tabuľky na obr. 62. Teraz už môžeme opísať kódový ekvivalent tohto automatu. Tento opis je daný pomocou tabuľky 16.

$$\begin{array}{c|cc} & \overbrace{x_2} & \\ \hline x_1 & \begin{array}{|c|c|} \hline a & d \\ \hline b & c \\ \hline \end{array} & \end{array}$$

OBR. 62. Kódovacia tabuľka vstupnej abecedy z príkladu 4.10

Na obr. 63 uvádzame tabuľku budiacich funkcií Y_1, Y_2, Y_3 úplne špecifikovaného dvoj-
kového automatu, na ktorý sme doplnili automat z tabuľky 16.

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
$f(H) = (1, 0, 0)$	$(-, -, -)$	$(-, -, -)$	$(-, -, -)$	$(0, 0, 0)$	1
$f(D) = (1, 1, 0)$	$(1, 1, 1)$	$(1, 1, 0)$	$(1, 1, 0)$	$(1, 0, 0)$	0
$f(B) = (0, 1, 0)$	$(0, 0, 0)$	$(0, 1, 0)$	$(1, 1, 0)$	$(0, 1, 0)$	0
$f(G) = (0, 1, 1)$	$(-, -, -)$	$(-, -, -)$	$(-, -, -)$	$(0, 1, 0)$	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

Pri dopĺňaní sme prihliadali iba na ten fakt, aby pri doplnených prechodoch medzi stavmi nedochádzalo k súbehom. Tabuľku výstupnej funkcie $z = \mu(y_1, y_2, y_3)$ uvádzame na obr. 64.

		x_1		x_2	
y_1		(0, 0, 0)	(0, 1, 0)	(0, 0, 0)	(0, 0, 0)
		(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)
		(0, 0, 0)	(0, 0, 0)	(0, 0, 0)	(0, 0, 0)
y_3					

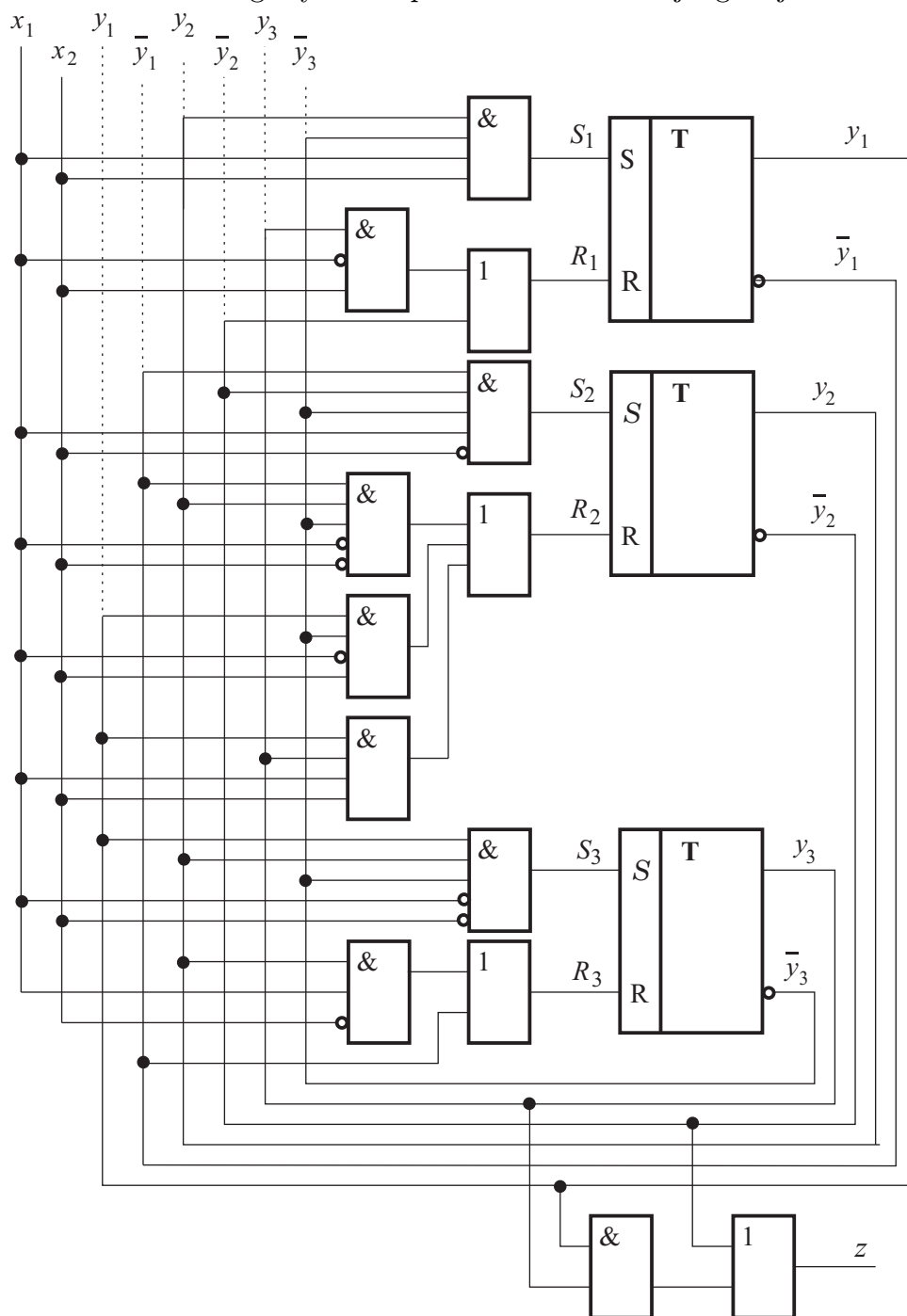
(Y_1, Y_2, Y_3)

OBR. 63. Budiace funkcie z príkladu 4.10

Z Karnaughových máp pre vstupné funkcie $S_1, R_1, S_2, R_2, S_3, R_3$ a pre výstupnú funkciu z čítame:

$$\begin{aligned} S_1 &= y_2 \bar{y}_3 x_1 x_2, & R_1 &= \bar{y}_2 + y_3 \bar{x}_1 x_2, \\ S_2 &= \bar{y}_1 \bar{y}_2 \bar{y}_3 x_1 \bar{x}_2, & R_2 &= \bar{y}_1 y_2 \bar{y}_3 \bar{x}_1 \bar{x}_2 + y_1 \bar{y}_3 \bar{x}_1 x_2 + y_1 y_3 x_1 x_2, \\ S_3 &= y_1 y_2 \bar{y}_3 \bar{x}_1 \bar{x}_2, & R_3 &= \bar{y}_1 + y_2 x_1 \bar{x}_2, \\ z &= \bar{y}_2 + y_1 y_3. \end{aligned}$$

Príslušnú logickú sieť reprezentujúcu fyzikálnu realizáciu daného dvojkového automatu uvádzame na obr. 66. Ak logický obvod patriaci k nakreslenej logickej sieti bude pracovať



OBR. 66. Logická sieť patriaca k automatu z príkladu 4.10

vo fundamentálnom režime, bude imitovať činnosť pôvodného automatu. ■

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 diskrétného času. Body diskrétného 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 diskrétného času je určený čelom impulzu. Keď významnou zmenou impulzovej premennej je zmena z 1 na 0, vtedy hovoríme, že bod diskrétného 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.

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 diskrétného 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).

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**.

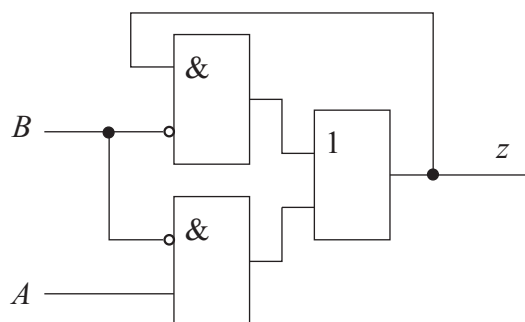
Pri impulzových sekvenčných obvodoch sú najčastejšie za impulzové premenné zvolené niektoré zo vstupných premenných a všetky stavové premenné majú hladinový charakter. Špeciálnym prípadom takéhoto obvodu je synchronný logický obvod, v ktorom diskretný čas určuje jedna impulzová vstupná premenná.

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 Mooreov 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 $n2^r$ vstupných vektorov, čiže

treba opísať činnosť daného obvodu pri n možných rozdeleniach významných hodnôt pri impulzových premenných. V tomto prípade vlastne ide o zovšeobecnenie synchronných 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. 67. 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. 67. 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:

$$a_1 = (A \rightarrow 1), a_0 = (A \rightarrow 0), b_1 = (B \rightarrow 1), b_0 = (B \rightarrow 0).$$

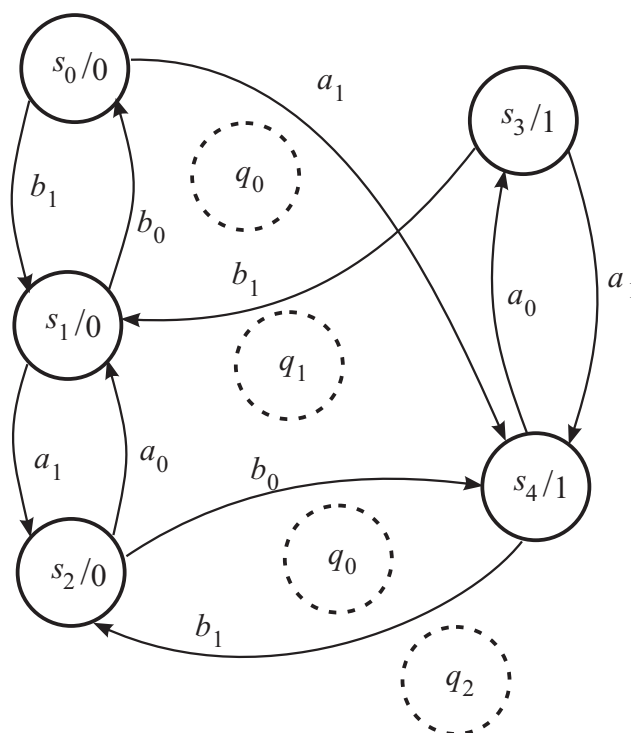
Aby činnosť toho obvodu mohla byť opísaná pomocou automatu, musia existovať jeho stabilné stavy, čiže stavy, ktoré sa pri nezmenenom vstupe nemenia. Hodnoty výstupnej funkcie tohto obvodu môžeme reprezentovať pomocou NDF v tvare $\overline{B}z + A\overline{B}$. Je zrejmé, že daný obvod bude v stabilnom stave, práve vtedy, keď $z = \overline{B}z + A\overline{B}$. Uvažujme teraz o dvoch možnostiach.

- Nech $z = 0$, potom nutne $A\overline{B} = 0$. To je splnené práve v týchto prípadoch $(A = 0, B = 0)$, $(A = 0, B = 1)$, $(A = 1, B = 1)$.
- Nech $z = 1$, potom $1 = \overline{B}z + A\overline{B} = \overline{B}$. Táto podmienka je splnená práve v prípadoch $(A = 0, B = 0)$, $(A = 1, B = 0)$.

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 $q_0 = (1, 0, 0) \rightarrow s_4 = (1, 0, 1)$. Podobne dostaneme $q_1 = (0, 1, 1) \rightarrow s_1 = (0, 1, 0)$ a $q_2 = (1, 1, 1) \rightarrow s_2 = (1, 1, 0)$.

Ak za body diskrétného 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



OBR. 68. Graf automatu z príkladu 4.11

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 Mooreov 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. 68. V prípade prechodu cez nestabilný stav sme tento stav v grafe na obr. 68 vyznačili pod príslušnou hranou. Tabuľku tohto neúplne špecifikovaného automatu uvádzame v tabuľke 17, časť a).

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	s_0	s_4	s_0	s_1	0
s_1	s_1	s_2	s_0	s_1	0
s_2	s_1	s_2	s_4	s_2	0
s_3	s_3	s_4	s_3	s_1	1
s_4	s_3	s_4	s_4	s_2	1

b)

V tabuľke 17, časť b) sme tento automat úplne špecifikovali na základe toho, že daný logický obvod pri nezmenenom vstupe zostáva v stabilnom stave. ■

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

Predslov	3
Kapitola 1. Úvodné pojmy	5
1. Zobrazenia a operácie	5
2. Jazyk nad abecedou	7
3. Výroková logika	8
4. Relácie	18
5. Orientované grafy	21
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 $f : \mathbf{B}^n \rightarrow \mathbf{B}^m$	51
5. Karnaughove mapy	59
6. Minimalizácia B-výrazov	68
Kapitola 3. Konečné automaty	77
1. Definícia konečného automatu	77
2. Konečné akceptory	87
3. Základné pojmy v teórii konečných automatov	90
4. Neúplne špecifikované (nedeterministické) automaty	94
5. Ekvivalencia automatov	96
6. Izomorfizmus automatov	108
7. Pokrytie automatu automatom	112
Kapitola 4. Fyzikálna realizácia automatov	117
1. Dvojkové automaty	117
2. Preklápacie obvody	119
3. Synchronne a asynchronne logické obvody	127
4. Fyzikálna realizácia automatov	133
Literatúra	163