

Číslicové systémy a jazyk VHDL

Pavol Galajda, KEMT, FEI, TUKE

Pavol.Galajda@tuke.sk

Základné vlastnosti, parametre, aplikácia pasívnych a polovodičových prvkov, modely prvkov a ich využitie pri analýze a syntéze jednoduchých elektronických obvodov.

Elektronikou rozumieme odvetvie fyziky, ktoré sa zaoberá vedením elektrického prúdu a príbuznými javmi v tuhých látkach - kovoch, polovodičoch, dielektrikách, v kvapalinách a v ionizovaných plynoch. Do elektroniky ďalej zahŕňame časť techniky, ktorá sa zaoberá využitím týchto javov pri návrhu a konštrukcii elektronických prvkov a obvodov.

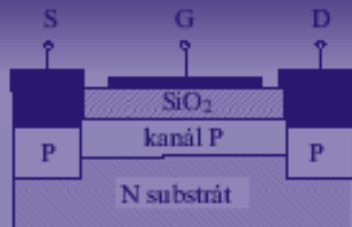
Rozvoj techniky sa začal v druhej polovici 19. storočia. Hlavná pozornosť sa vtedy zamerala na mechanizmy vedenia elektrického prúdu v zriedených plynoch. Toto štúdium viedlo k objavu katódových lúčov (1899). Ďalší výskum ukázal, že katódové lúče sú rýchlo sa pohybujúce záporne nabité častice, ktoré sa začali približne od roku 1900 nazývať *elektrónmi*.

I keď v prvej etape elektroniky dominovali vákuové elektróny, rozvíjali sa postupne aj elektronické prvky z tuhých látok.

Dôležitým medzníkom v polovodičovej elektronike bol objav tranzistora. Koncom roku 1947 to bol hrotový tranzistor, ktorý objavil J. Barden a W. H. Brattain. V priebehu ďalších mesiacov W. Shockley sformuloval koncepciu *plošného tranzistora* s využitím vlastností *přechodov PN* (pri jeho činnosti sa vyžadujú obidva druhy nosičov náboja - elektróny a diery a preto patrí medzi *bipolárne súčastky*). Nie je bez zaujímavosti, že hrotový a plošný tranzistor sa objavili ako dôsledok experimentov, cieľom ktorých bolo získať *unipolárne súčastky*. Keďže tieto súčastky využívajú efekt poľa, nazývajú sa *tranzistory ovládané elektrickým poľom* a označujú sa *FET* (Field Effect Transistor).

Spoločne s *integráciou* pokračovala aj *miniaturizácia* súčastok a polovodičová technika vyúsťila tak do svojej súčasnej etapy - *mikroelektroniky*. Polovodičové súčastky a obvody okrem špeciálnych aplikácií (napr. obrazovky) nahradili elektróny.

Ale o tom až neskôr...



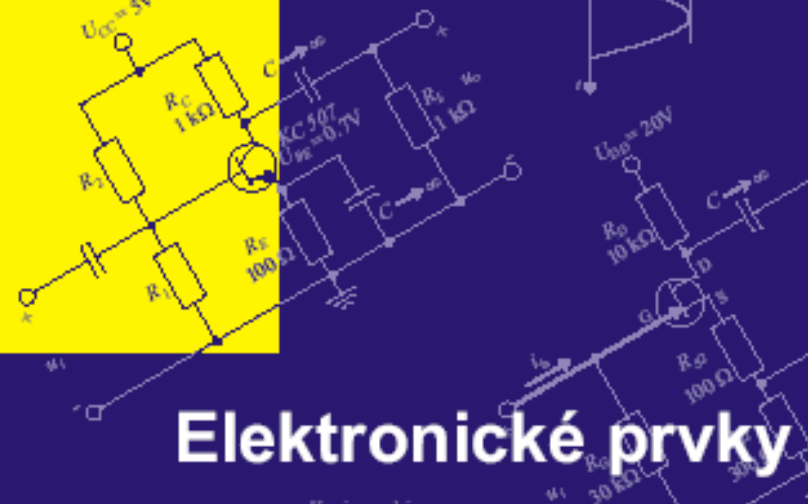
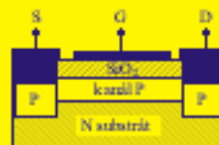
ISBN 80-89061-51-6



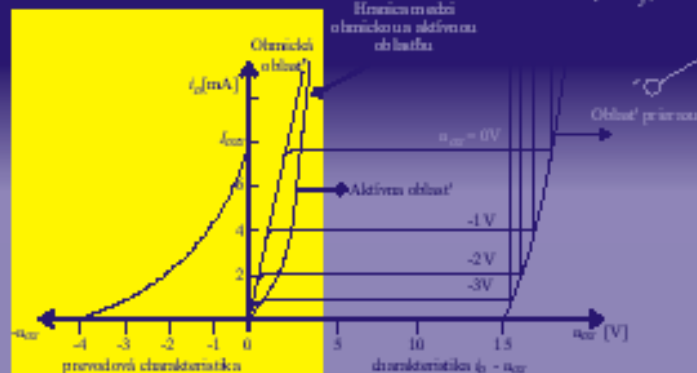
Elektronické prvky

Galajda - Lukáč

Pavol Galajda
Rastislav Lukáč



Elektronické prvky



Prečo práve S.O.S Electronic

Široký výber sortiments,
prispôbený Vašim požiadavkám...

Viac ako 7000 typov súčiastok
priamo na našom sklade...

Rýchle dodávky
tovaru...



Vždy máme pre Vás
prípravené niečo nové...

Technické poradenstvo,
vyskúšaný personál,
kvalitné služby...

Kvalitné služby pre Vás
- ISO certifikát...

ISBN 80-89061-59-1



9 788089 761594

OnLine Shop ...
- aktuálne
ceny a skladové
množstvá...

Novinky, zaujímavosti, akcie ...

Komplexné informácie ...

4 x ročne časopis ...

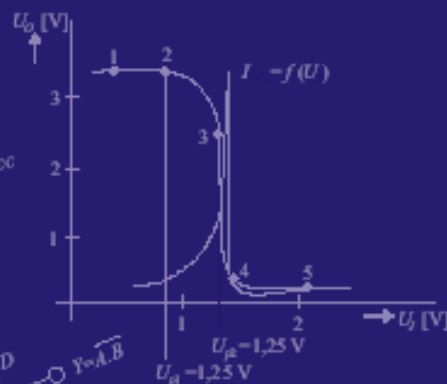
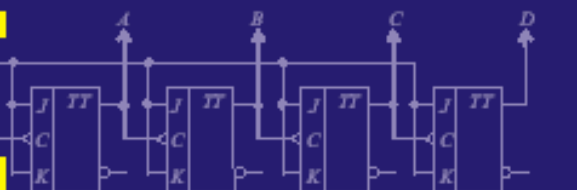
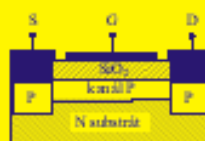
www.soselectronic.sk

S.O.S. electronic, Ždiarska 32, 04001 Košice, tel. 055/623 40 00-4, fax. 055/623 40 07, e-mail: info@sos.sk

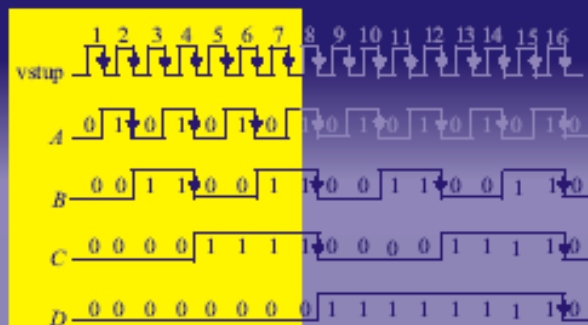


Elektronické obvody

Pavol Galajda¹⁾
Rastislav Lukáč²⁾
vstup



Elektronické obvody



Galajda - Lukáč

3 Logické systémy

- **3.1 Základné pojmy a definície**
- **3.2 Činnosť logických systémov v čase**
- **3.3 Kombinačné a sekvenčné obvody**
- **3.4 Základné úlohy pri práci s logickými systémami**

3.1 Základné pojmy a definície

Zavedieme niekoľko základných definícií, ktoré budeme v ďalšom používať.

•***System***

System je množina prvkov určitých vlastností a množina väzieb medzi nimi, ktoré spolu určujú vlastnosti, chovanie a funkciu systému ako celku.

•***Okolie systému***

Okolie systému je charakterizované účelovo definovanou množinou prvkov, ktoré nie sú prvkami daného systému, ale sú spojené väzbami s tzv. hraničnými prvkami systému.

•***Hraničný prvok***

Hraničný prvok umožňuje vytvoriť väzbu (interakciu) medzi systémom a okolím.

•***Logický systém***

Logický systém je systém, ktorého veličiny nadobúdajú hodnoty len v určitých diskretných časových intervaloch a majú konečný počet hodnôt.

3.1 Základné pojmy a definície

Zavedieme niekoľko základných definícií, ktoré budeme v ďalšom používať.

- ***Okolie logického systému***

Okolie logického systému charakterizuje iné logické systémy, ktoré pôsobia na uvažovaný logický systém prostredníctvom účelových väzieb realizovaných vstupno-výstupnými kanálmi.

- ***Logické signály (najčastejšie dvojhodnotové signály)***

Logické signály sú nositeľmi informácií medzi logickým systémom a okolím. Môžu nadobúdať konečný počet hodnôt. Najčastejšie sa používajú logické systémy, ktoré pracujú s *dvojhodnotovými signálmi*.

- ***Vstupný stav systému***

Vstupný stav systému je kombinácia hodnôt vstupných signálov.

- ***Výstupný stav systému***

Výstupný stav systému je kombinácia hodnôt výstupných signálov.

- ***Vnútorový stav systému (vnútorné signály)***

Vnútorový stav je kombinácia hodnôt vnútorných signálov.

3.1 Základné pojmy a definície

Zavedieme niekoľko základných definícií, ktoré budeme v ďalšom používať.

- ***Vstupný stav systému***

Vstupný stav systému je kombinácia hodnôt vstupných signálov.

- ***Výstupný stav systému***

Výstupný stav systému je kombinácia hodnôt výstupných signálov.

V danom časovom intervale nadobúdajú vstupné resp. výstupné signály určité hodnoty.

- ***Vnútorový stav systému (vnútorné signály)***

Vnútorový stav je kombinácia hodnôt vnútorných signálov.

V každom logickom systéme sa vyskytujú systémom determinované vzťahy medzi vstupnými a výstupnými signálmi. Tieto vzťahy sprostredkujú signály vo vnútornej štruktúre systému, ktoré nazývame *vnútorné signály*. V danom časovom intervale nadobúdajú vnútorné signály určité hodnoty.

3.2 Činnosť logických systémov v čase

Logické systémy pracujú obvykle v diskretnom čase, t.j. v plynule prebiehajúcim čase sa vyskytujú časové intervaly, v ktorých dvojhodnotové signály systému nadobúdajú určitú, v tomto intervale nemeniacu sa hodnotu. Tieto diskkrétne, po sebe idúce a navzájom neprekrývajúce sa časové intervaly nazývame *taktami systému*.

Z hľadiska vymedzenia taktov na časovej osi a vymedzenia časových intervalov, v ktorých sa realizujú zmeny hodnôt signálov, možno logické systémy rozdeliť na *synchronne* a *asynchronne*.

3.2 Činnosť logických systémov v čase

V ***synchronných systémoch*** sú takty určované osobitnými *synchronizačnými signálmi*, ktoré generuje osobitný synchronizačný systém. Často sa v systéme vyskytuje jeden dvojhodnotový synchronizačný signál. Takty potom zodpovedajú časovým intervalom, v ktorých tento synchronizačný signál nadobudne jednu z dvoch hodnôt.

V ***asynchronných systémoch*** sú takty určované vstupnými a vnútornými signálmi. Nový takt začne vždy vtedy, keď hociktorý zo vstupných alebo vnútorných signálov nadobúda novú hodnotu. Takt pritom trvá dovtedy, kým všetky vstupné a vnútorné signály majú určitú, resp. jednu hodnotu.

3.2 Činnosť logických systémov v čase

V jednotlivých taktoch teda obidva druhy systémov charakterizuje určitý vstupný, vnútorný a výstupný stav. Prechody medzi vnútornými stavmi daného logického systému určuje jeho vnútorná, resp. *prechodová funkcia*, ktorú možno zapísať v tvare

$$S(t+1) = \delta[S(t), X(t)]$$

kde:

$S(t)$ je vnútorný stav systému v takte t ,

$X(t)$ je vstupný stav v takte t ,

$S(t+1)$ je vnútorný stav systému v takte $(t+1)$ pre $t = 0, 1, 2, \dots$,

δ je zobrazenie $S \times X \rightarrow S$.

Uvedenú funkciu možno pre konkrétny logický systém zapísať do konečnej tabuľky. Táto tabuľka sa nazýva *tabuľka vnútornej funkcie* alebo *tabuľka prechodov* daného logického systému.

3.2 Činnosť logických systémov v čase

Napr. v Tab. 3.1 je zapísaná vnútorná funkcia logického systému, ktorý má štyri navzájom rôzne vnútorné stavy, teda $S = \{S_1, S_2, S_3, S_4\}$ a tri navzájom rôzne vstupné stavy, teda $X = \{X_1, X_2, X_3\}$.

Tab. 3.1 Vnútorná funkcia logického systému s tromi navzájom rôznymi vstupnými stavmi a štyrmi navzájom rôznymi vnútornými stavmi.

$S(t)$	$X(t)$		
	X_1	X_2	X_3
S_1	S_1	S_2	S_3
S_2	S_1	S_3	S_2
S_3	S_4	S_3	S_2
S_4	S_1	S_2	S_4

Každému vnútornému stavu, v ktorom sa systém môže nachádzať, je priradený riadok tabuľky. Každému vstupnému stavu, ktorý sa môže v systéme nachádzať, je priradený stĺpec tabuľky. V priesečníku riadku zodpovedajúceho určitému vnútornému stavu S_j ($j = 1, 2, 3, 4$) so stĺpcom, ktorý zodpovedá určitému stavu na vstupe X_i ($i = 1, 2, 3$), je zapísaný vnútorný stav $S_k = \delta(S_j, X_i)$.

Ak napr. vnútorný stav v takte t sa rovná $S(t) = S_2$ a stav na vstupe v takte t je $X(t) = X_2$, potom vnútorný stav v takte $(t+1)$ bude $S(t+1) = \delta(S_2, X_2) = S_3$.

3.2 Činnosť logických systémov v čase

Stav na výstupe logického systému je charakterizovaný *výstupnou funkciou*, ktorú možno vyjadriť v tvare

$$Y(t) = \lambda[S(t), X(t)],$$

kde $Y(t)$ je výstupný stav logického systému v čase t ($t = 0, 1, 2, \dots$),
 $S(t)$, $X(t)$ majú rovnaký význam, ako v predchádzajúcom prípade,
 λ je zobrazenie $S \times X \rightarrow Y$

3.3 Kombinačné a sekvenčné obvody

Budeme predpokladať, že štruktúru systému tvoria jednoduchšie systémy, napr. logické členy, ktoré realizujú logické funkcie a vzájomné väzby medzi nimi. Tieto systémy tiež nazývame *logické obvody*.

Logické obvody delíme do dvoch základných skupín:

- kombinačné logické obvody
- sekvenčné logické obvody.

3.3 Kombinačné a sekvenčné obvody

Kombinačné logické obvody sú také obvody, v ktorých každému vstupnému stavu X_i z konečnej množiny navzájom rôznych stavov $\{X_1, X_2, X_3, \dots, X_N\}$ prislúcha určitý výstupný stav Y_i z konečnej množiny navzájom rôznych stavov $\{Y_1, Y_2, Y_3, \dots, Y_M\}$, pre $M \leq N$, bez ohľadu na to, aké stavy boli na vstupe obvodu v minulosti. Pre tieto obvody platí, že hodnoty výstupných signálov v takte t sú jednoznačne určené kombináciou hodnôt vstupných signálov v tom istom takte t a nezávisia od hodnôt vstupných signálov v predchádzajúcich taktoch. Kombinačné obvody preto neobsahujú pamäťové prvky na uchovávanie informácie z predchádzajúcich taktov.

Závislosť medzi vstupnými a výstupnými stavmi kombinačného logického obvodu nazývame *funkciou kombinačného obvodu*, resp. *výstupnou funkciou*.

3.3 Kombinačné a sekvenčné obvody

Logické obvody delíme do dvoch základných skupín:

- kombinačné logické obvody- *funkciu kombinačného obvodu možno vyjadriť v tvare*

$$Y(t) = \delta [X(t)]$$

kde Y je výstupný stav obvodu,

X je vstupný stav obvodu,

δ je zobrazenie $X \rightarrow Y$

Funkciu kombinačného obvodu treba chápať ako sústavu vzťahov, ktoré udávajú hodnoty jednotlivých výstupných signálov v závislosti od kombinácií hodnôt vstupných signálov.

3.3 Kombinačné a sekvenčné obvody

Logické obvody delíme do dvoch základných skupín:

- kombinačné logické obvody- *pravdivostná tabuľka logickej funkcie*

Výstupné funkcie kombinačného obvodu možno zapísať v tvare tabuľky. Napr. v Tab. 3.2 sú zapísané výstupné funkcie, ktoré predstavujú model kombinačného obvodu, ktorý realizuje funkciu AND pre tri vstupné signály x_1, x_2, x_3 . Tab. 3.2 predstavuje pravdivostnú tabuľku logickej funkcie.

Tab. 3.2 Pravdivostná tabuľka logickej funkcie AND

X/\bar{X}	x_1	x_2	x_3	y_1	\bar{Y}
X_0	0	0	0	0	Y_0
X_1	0	0	1	0	Y_0
X_2	0	1	0	0	Y_0
X_3	0	1	1	0	Y_0
X_4	1	0	0	0	Y_0
X_5	1	0	1	0	Y_0
X_6	1	1	0	0	Y_0
X_7	1	1	1	1	Y_1

3.3 Kombinačné a sekvenčné obvody

Logické obvody delíme do dvoch základných skupín:

- sekvenčné logické obvody.

V sekvenčných logických obvodoch výstupný stav v danom časovom okamžiku nie je jednoznačne určený okamžitým stavom na vstupe obvodu, ale je závislý od postupnosti (sekvencie) vstupných signálov v minulosti. To znamená, že sekvenčný obvod musí obsahovať pamäťové podsystemy, ktoré sú schopné pamätať si informáciu z predchádzajúcich taktov. Táto informácia je vyjadrená vnútorným stavom sekvenčného obvodu, ktorý je reprezentovaný určitou kombináciou hodnôt vnútorných premenných. Hodnoty vnútorných premenných sú uchované v pamäťovej časti obvodu.

3.4 Základné úlohy pri práci s logickými systémami

Pri práci s logickými systémami existujú dve hlavné úlohy, ktoré je potrebné riešiť:

- štruktúrna analýza logických systémov, skrátene *analýza* je určenie chovania sa systému pre jeho zadanú vnútornú štruktúru a pre jeho počiatočný vnútorný stav.

Pritom pod chovaním systému rozumieme súhrn všetkých vzťahov medzi časovými postupnosťami vstupných signálov a im zodpovedajúcich výstupných signálov.

3.4 Základné úlohy pri práci s logickými systémami

Pri práci s logickými systémami existujú dve hlavné úlohy, ktoré je potrebné riešiť:

- štruktúrna syntéza logických systémov. resp. *syntézou* rozumieme určenie vnútornej štruktúry logického systému, ktorý má predpísané chovanie a obsahuje len podsystémy zadaného typu.

Ak je riešenie syntézy možné, potom je vždy viacznačné. Preto sú obvykle predpísané aj ďalšie podmienky, ktoré má štruktúra spĺňať, napr. požiadavka na minimálne materiálové náklady, rýchlosť, spoľahlivosť, atď.

4 Kombinačné obvody

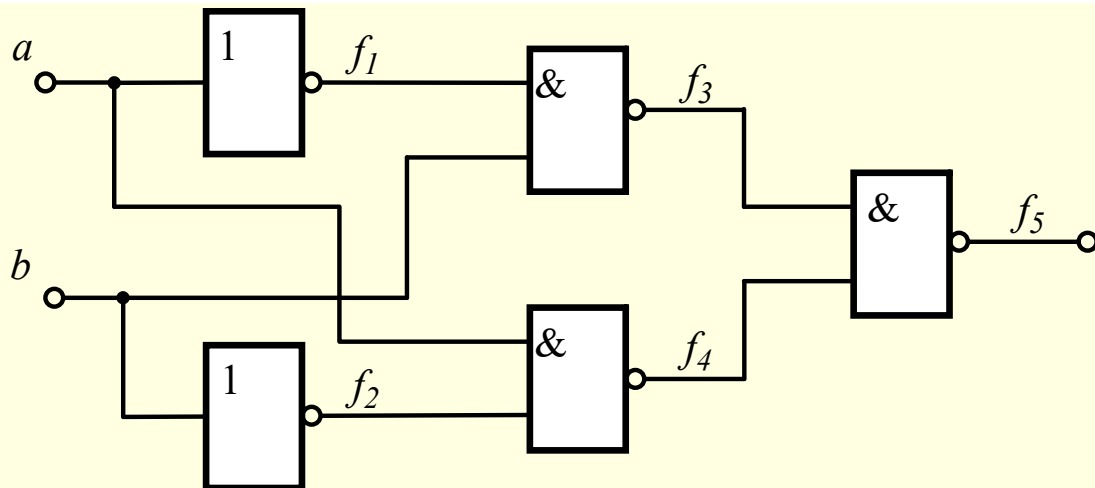
- **4.1** **Analýza kombinačných obvodov**
- **4.2** **Syntéza kombinačných obvodov**
- **4.3** **Niektoré kombinačné funkcie**
- **4.4** **Niektoré kombinačné obvody**
 - 4.4.1 *Dekodér*
 - 4.4.2 *Multiplexor*
 - 4.4.3 *Demultiplexor*
 - 4.4.4 *Kodéry a prevodníky kódov*
 - 4.4.5 *Obvody na realizáciu aritmetických operácií*

4.1 Analýza kombinačných obvodov

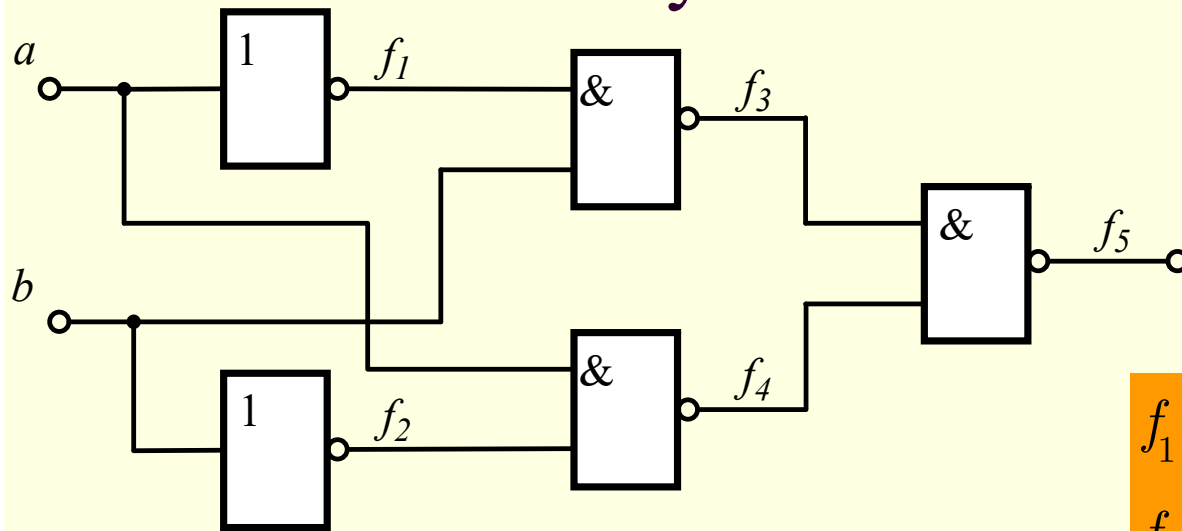
Analýzou kombinačných obvodov rozumieme určenie výstupnej funkcie resp. systému výstupných funkcií, ktoré realizuje daná štruktúra kombinačného obvodu.

Prvým krokom analýzy je určenie algebraického tvaru výstupnej funkcie resp. *algebraického modelu* kombinačného obvodu. Algebraický model kombinačného obvodu zostavíme pomocou dielčích funkcií vnútornej štruktúry, ktoré postupne upravíme tak, aby sme získali výstupnú funkciu v závislosti od vstupných premenných.

Takto získanú výstupnú funkciu daného kombinačného obvodu upravíme, resp. prepíšeme do tabuľkového tvaru, aby sme získali informáciu o činnosti kombinačného obvodu.



4.1 Analýza kombinačných obvodov

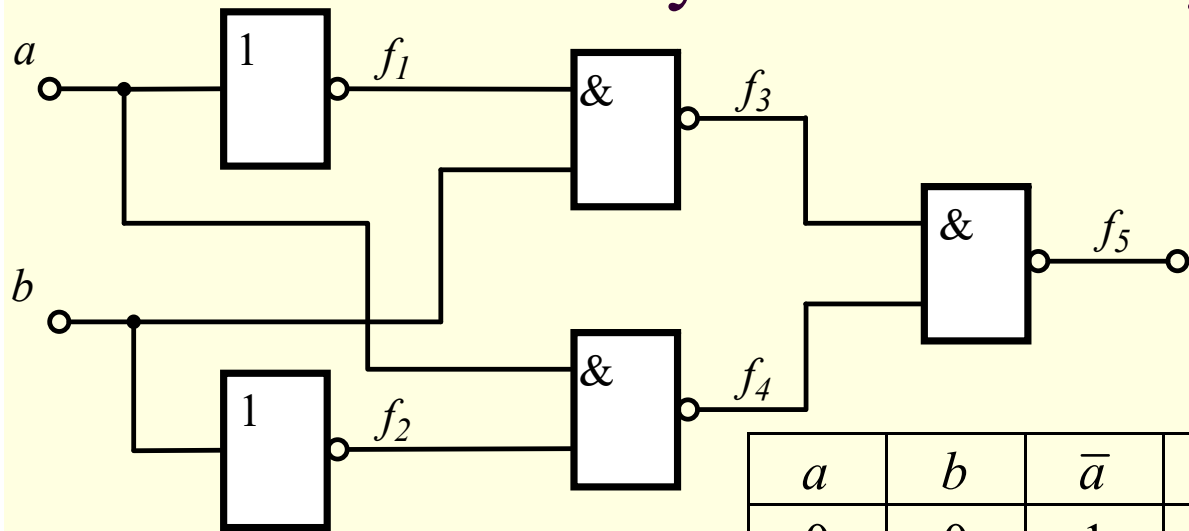


$$f_1 = \bar{a} \quad f_2 = \bar{b} \quad f_3 = \overline{f_1 \cdot b}$$
$$f_4 = \overline{f_2 \cdot a} \quad f_5 = \overline{f_3 \cdot f_4}$$

$$f = f_5 = \overline{\overline{\bar{a}b} \cdot \overline{a\bar{b}}} = \bar{a}b + a\bar{b} = a \oplus b$$

Prvým krokom analýzy je určenie algebraického tvaru výstupnej funkcie resp. *algebraického modelu* kombinačného obvodu. Algebraický model kombinačného obvodu zostavíme pomocou dielčích funkcií vnútornej štruktúry, ktoré postupne upravíme tak, aby sme získali výstupnú funkciu v závislosti od vstupných premenných.

4.1 Analýza kombinačných obvodov



a	b	\bar{a}	\bar{b}	$\bar{a}b$	$a\bar{b}$	$f = \bar{a}b + a\bar{b}$
0	0	1	1	0	0	0
0	1	1	0	1	0	1
1	0	0	1	0	1	1
1	1	0	0	0	0	0

Takto získanú výstupnú funkciu daného kombinačného obvodu upravíme, resp. prepíšeme do tabuľkového tvaru, aby sme získali informáciu o činnosti kombinačného obvodu.

Analogickým spôsobom postupujeme, ak má kombinačný obvod viacero výstupných funkcií. Činnosť kombinačného obvodu je potom popísaná systémom viacerých výstupných funkcií a tiež zodpovedajúcou pravdivostnou tabuľkou.

4 Kombinačné obvody

- 4.1 **Analýza kombinačných obvodov**
- 4.2 **Syntéza kombinačných obvodov**
- 4.3 **Niektoré kombinačné funkcie**
- 4.4 **Niektoré kombinačné obvody**
 - 4.4.1 *Dekodér*
 - 4.4.2 *Multiplexor*
 - 4.4.3 *Demultiplexor*
 - 4.4.4 *Kodéry a prevodníky kódov*
 - 4.4.5 *Obvody na realizáciu aritmetických operácií*

4.2 Syntéza kombinačných obvodov

Úlohou syntézy kombinačného obvodu je nájsť jeho štruktúru, ak je známa výstupná funkcia. Vzhľadom k viacznačnosti riešenia sa stanovujú dodatočné podmienky napr. typ logických členov, minimálne materiálové náklady.

Postup pri syntéze kombinačných obvodov možno zhrnúť do nasledovných krokov:

- podľa zadanej činnosti definujeme vstupné premenné a výstupné funkcie,
- zapíšeme funkciu obvodu pomocou *pravdivostnej tabuľky*, alebo *Karnaughovej mapy*,
- hľadáme algebraické vyjadrenie podľa zadaného kritéria metódami minimalizácie,
- pre výsledné algebraické vyjadrenie nakreslíme štruktúrnu schému, podľa ktorej odvod realizujeme.

4 Kombinačné obvody

- 4.1 **Analýza kombinačných obvodov**
- 4.2 **Syntéza kombinačných obvodov**
- 4.3 **Niektoré kombinačné funkcie**
- 4.4 **Niektoré kombinačné obvody**
 - 4.4.1 *Dekodér*
 - 4.4.2 *Multiplexor*
 - 4.4.3 *Demultiplexor*
 - 4.4.4 *Kodéry a prevodníky kódov*
 - 4.4.5 *Obvody na realizáciu aritmetických operácií*

4.3 Niektoré kombinačné funkcie

- **Funkcia prepínania** sa používa, ak je potrebné spojiť do spoločných bodov viaceré číslicové veličiny. Prepínanie sa realizuje riadiacimi signálmi.
- **Funkcia spájania**- Informácia, ktorá sa spracováva v číslicových systémoch sa obvykle organizuje do viacbitových údajov - *slov*. Pri práci s takto vyjadrenou informáciou často potrebujeme dva údaje o dĺžke $N/2$ bitov spojiť do jedného slova o dĺžke N bitov. Uvedenú operáciu realizujeme *funkciou spájania*.
- **Funkcia maskovania**- V číslicových systémoch sa často vyskytuje inverzná funkcia k funkcii spájania, tzv. *funkcia maskovania*. Pomocou tejto funkcie možno extrahovať z daného slova určitú časť, pričom ostávajúce bity budú nulové. Pri realizácii uvedenej funkcie sa definuje *extraktor* - maska, ktorá obsahuje hodnotu 1 v tých bitoch, ktoré chceme extrahovať a v ostatných bitoch je hodnota 0.
- **Funkcia zhodnosti**- Medzi obvyklé kombinačné funkcie patrí tiež *funkcia komparovania* dvoch veličín.

4 Kombinačné obvody

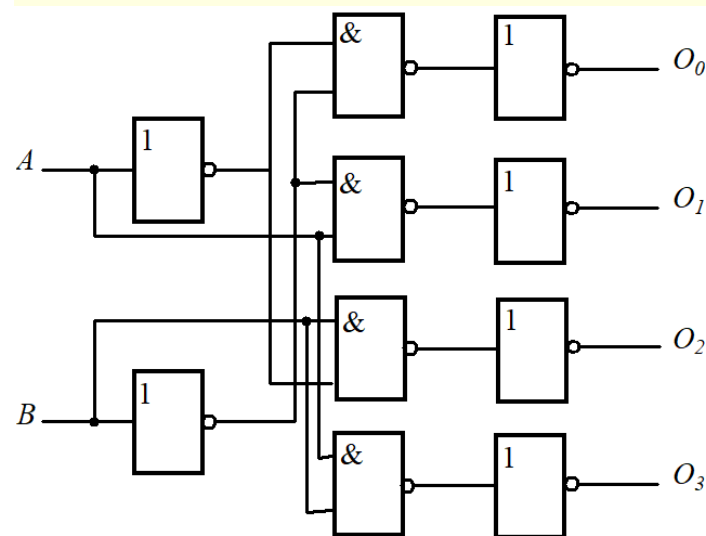
- 4.1 **Analýza kombinačných obvodov**
- 4.2 **Syntéza kombinačných obvodov**
- 4.3 **Niektoré kombinačné funkcie**
- **4.4 Niektoré kombinačné obvody**
 - 4.4.1 *Dekodér*
 - 4.4.2 *Multiplexor*
 - 4.4.3 *Demultiplexor*
 - 4.4.4 *Kodéry a prevodníky kódov*
 - 4.4.5 *Obvody na realizáciu aritmetických operácií*

4.4.1 Dekodér

Kombinačný obvod, ktorý má n vstupov a 2^n výstupov, pričom je aktívny vždy iba jeden z týchto výstupov, označujeme ako *dekodér typu 1 z 2^n* . Ak výstupy dekodéra označíme číslami 0 až $2^n - 1$, potom číslo aktívneho výstupu zodpovedá dvojkovej hodnote kombinácie privedenej na vstup. Princíp dekodéra vysvetlíme na príklade pre $n = 2$ (Tab. 4.3).

Tab. 4.3 Pravdivostná tabuľka dekodéra 1 z 2^2 .

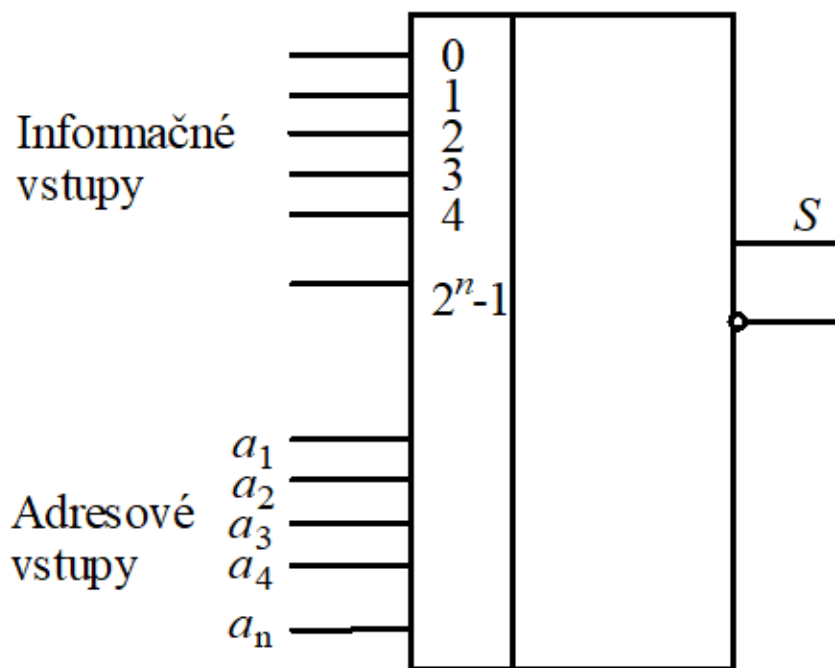
Vstupy		Výstupy			
A	B	O_0	O_1	O_2	O_3
0	0	1	0	0	0
0	1	0	1	0	0
1	0	0	0	1	0
1	1	0	0	0	1



Obr. 4.3 Realizácia dekodéra 1 z 2^2 na báze hradiel NAND

4.4.2 Multiplexor

Je to kombinačný obvod s 2^n informačnými vstupmi, s n adresovými vstupmi a jedným výstupom. Na výstup *multiplexora* sa prenesie údaj z informačného vstupu, ktorého číslo (v rozmedzí 0 až $2^n - 1$) zodpovedá kombinácii na adresových vstupoch.



Obr. 4.4 Bloková schéma multiplexora

Bloková schéma multiplexora je na Obr. 4.4, ktorý ukazuje, že výstup multiplexora je riešený obvykle ako komplementárny.

4.4.3 Demultiplexor

Je to kombinačný obvod, ktorý realizuje inverznú funkciu ako multiplexor. *Demultiplexor* má n adresových vstupov a 2^n výstupov, pričom určitá hodnota (napr. 0) sa vysiela do jedného z 2^n výstupov. Číslo tohto výstupu zodpovedá kombinácii na adresových vstupoch. Z tohto pohľadu ide o analogickú funkciu, ako má dekodér typu 1 z 2^n .

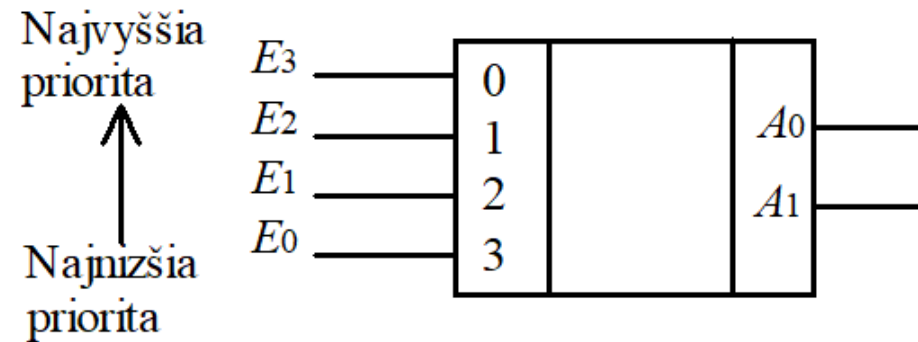
Ako príklad uvidíme demultiplexor pre $n = 2$, ktorého funkcia je vyjadrená v Tab. 4.5

Tab. 4.5 Pravdivostná tabuľka demultiplexora pre $n = 2$.

Vstupy		Výstupy			
B	A	O_0	O_1	O_2	O_3
0	0	0	1	1	1
0	1	1	0	1	1
1	0	1	1	0	1
1	1	1	1	1	0

4.4.4 Kodéry a prevodníky kódov

Kodéry sú kombinačné obvody, ktoré realizujú prevod číslicovej informácie na niektorý z binárnych kódov. Napr. kodér pre klávesnice realizuje prevod na kód ASCII, ktorým sa kóduje každý znak na klávesnici.



Obr. 4.5 Bloková schéma prioritného kodéra

Špeciálnym prípadom kodéra je *prioritný kodér*, ktorý má 2^n adresových vstupov a n adresových výstupov. Jednotlivým adresovým vstupom je priradená priorita v rozmedzí 0 až $2^n - 1$. Princíp činnosti prioritného kodéra spočíva v tom, že na výstupe sa objaví kódová kombinácia, ktorá zodpovedá číslu adresového vstupu, na ktorom je v danom časovom okamžiku aktívna úroveň. Pri súčasnom výskyte aktívnych úrovní na viacerých vstupoch sa na výstupe objaví kódová kombinácia adresového vstupu s najvyššou prioritou.

Ako príklad uvidíme prioritný kodér pre $n = 2$, ktorý je na Obr. 4.5.

4.4.4 Kodéry a prevodníky kódov

Prevodníky kódov sú kombinačné obvody, ktoré realizujú prevod medzi binárnymi kódmi. Funkciu prevodníka kódov možno popísať pravdivostnou tabuľkou, v ktorej vstupné premenné tvoria jednotlivé bity vstupného kódu prevodníka a výstupné funkcie jednotlivé bity výstupného kódu. Pri syntéze prevodníka potom pravdivostnú tabuľku prepíšeme do Karnaughových máp a výstupné funkcie vyjadríme v algebraickom tvare.

4 Kombinačné obvody

- *4.4.5 Obvody na realizáciu aritmetických operácií*
 - 4.4.5.1 Neúplná jednobitová sčítačka
 - 4.4.5.2 Úplná jednobitová sčítačka
 - 4.4.5.3 Odčítačky
 - 4.4.5.4 Zrýchlenie prenosu pri sčítavaní dvojkových čísel
 - 4.4.5.5 Realizácia sčítačiek a odčítačiek
 - 4.4.5.6 Aritmeticko-logické jednotky

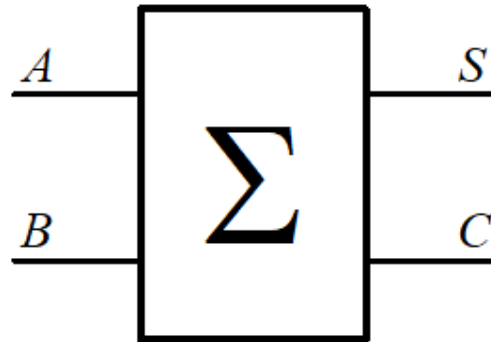
4.4.5 Obvody na realizáciu aritmetických operácií

Medzi základné operácie v číslicových obvodoch patria sčítanie a odčítanie. Sčítanie v dvojkovej sústave je najdôležitejšou aritmetickou operáciou, pretože je základom i pre ďalšie aritmetické operácie, napr. odčítanie, násobenie a delenie. Základnými obvodymi, ktoré realizujú elementárnu operáciu sčítania sú tzv. *jednobitové sčítačky*, ktoré môžu byť úplné alebo neúplné.

4.4.5 Obvody na realizáciu aritmetických operácií

4.4.5.1 Neúplná jednobitová sčítačka

Neúplná jednobitová sčítačka má dva vstupy a dva výstupy. Jeden výstup udáva výsledok súčtu dvoch bitov a druhý výstup udáva výsledok prenosu, ktorý vzniká pri súčte. Bloková schéma jednobitovej neúplnej sčítačky je na Obr. 4.6. Jej činnosť vyjadruje pravdivostná tabuľka Tab. 4.7.



Obr. 4.6 Bloková schéma jednobitovej neúplnej sčítačky

4.4.5 Obvody na realizáciu aritmetických operácií

Tab. 4.7 Pravdivostná tabuľka jednobitovej neúplnej sčítačky.

A	B	S	C
0	0	0	0
0	1	1	0
1	0	1	0
1	1	1	1

Po prepísaní pravdivostnej tabuľky do Karnaughovej mapy možno pre jednobitovú neúplnú sčítačku odvodiť vzťahy

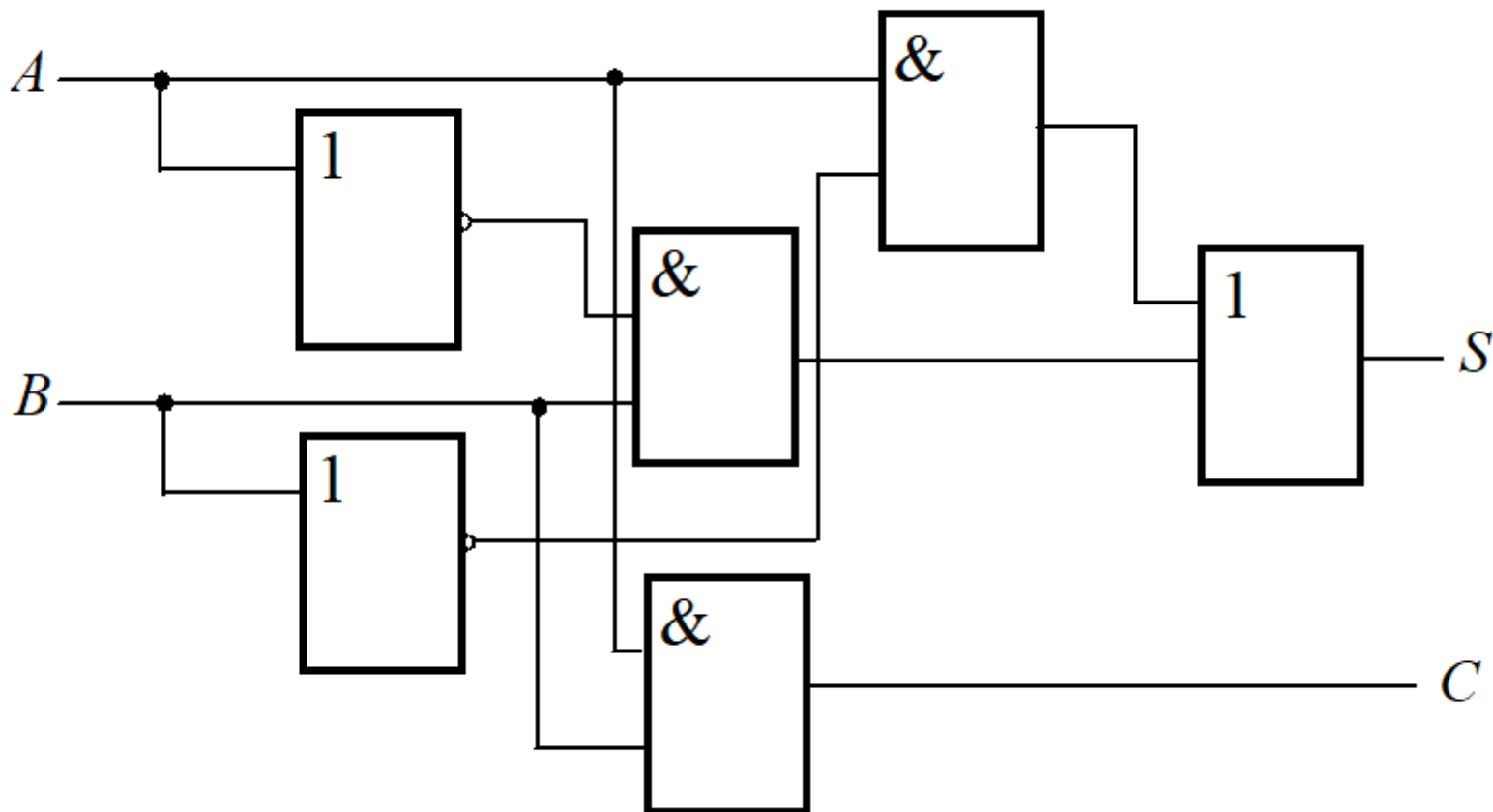
$$S = A\bar{B} + \bar{A}B = A \oplus B \quad (4.12)$$

$$C = AB \quad (4.13)$$

Realizácia jednobitovej neúplnej sčítačky je na Obr. 4.7.

4.4.5 Obvody na realizáciu aritmetických operácií

Realizácia jednobitovej neúplnej sčítačky je na Obr. 4.7.



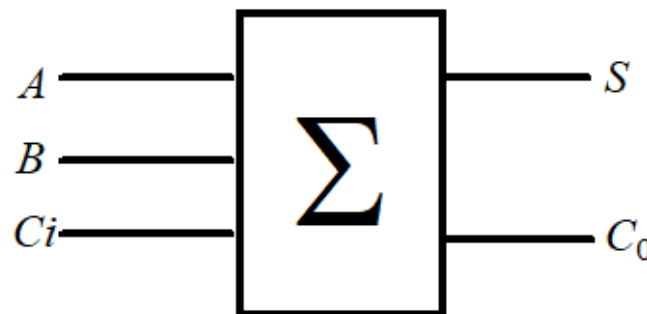
Obr. 4.7 Realizácia jednobitovej neúplnej sčítačky

4.4.5 Obvody na realizáciu aritmetických operácií

4.4.5.2 Úplná jednobitová sčítačka

Úplná jednobitová sčítačka sa líši od neúplnej tým, že má tri vstupy. Na výstupe S je výsledok súčtu dvoch jednobitových čísiel A, B a prenosu C_i z predchádzajúcej sčítačky, a na výstupe C_o je prenos k nasledujúcej sčítačke.

Bloková schéma úplnej jednobitovej sčítačky je na Obr.4.8 a jej činnosť je vyjadrená pravdivostnou tabuľkou Tab. 4.8.



Obr.4.8 Bloková schéma jednobitovej úplnej sčítačky

4.4.5 Obvody na realizáciu aritmetických operácií

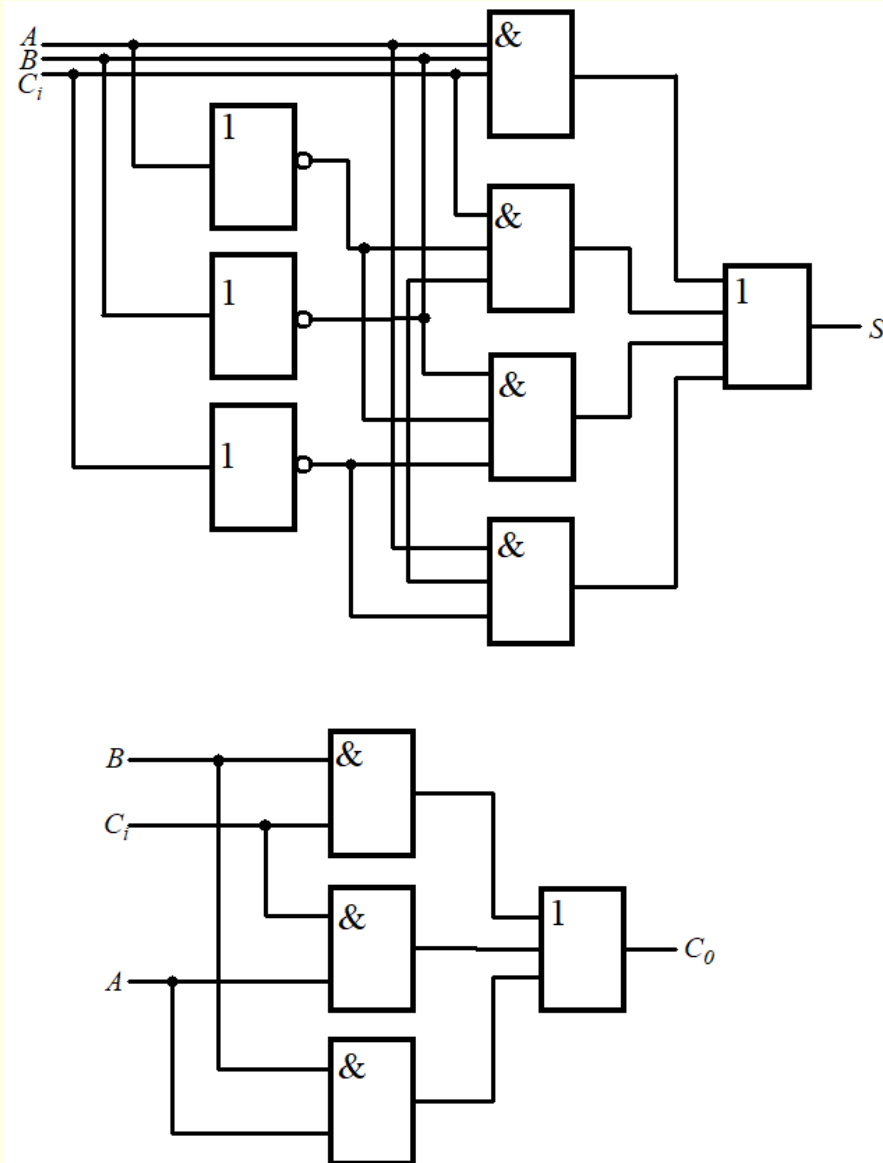
Tab. 4.8 Pravdivostná tabuľka jednobitovej úplnej sčítačky.

A	B	C_i	S	C_o
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

Pravdivostnú tabuľku Tab. 4.7 prepíšeme do Karnaughových máp a odvodíme algebraické výrazy pre úplnú jednobitovú sčítačku v tvare

$$\begin{aligned}
 S &= A\bar{B}\bar{C}_i + \bar{A}B\bar{C}_i + \bar{A}\bar{B}C_i + ABC_i = & C_o &= AB + AC_i + BC_i = \\
 &= A(\bar{B}\bar{C}_i + BC_i) + \bar{A}(\bar{B}C_i + B\bar{C}_i) = & &= AB + C_i(A + B) \\
 &= A(\overline{B \oplus C_i}) + \bar{A}(B \oplus C_i) = A \oplus B \oplus C_i
 \end{aligned}$$

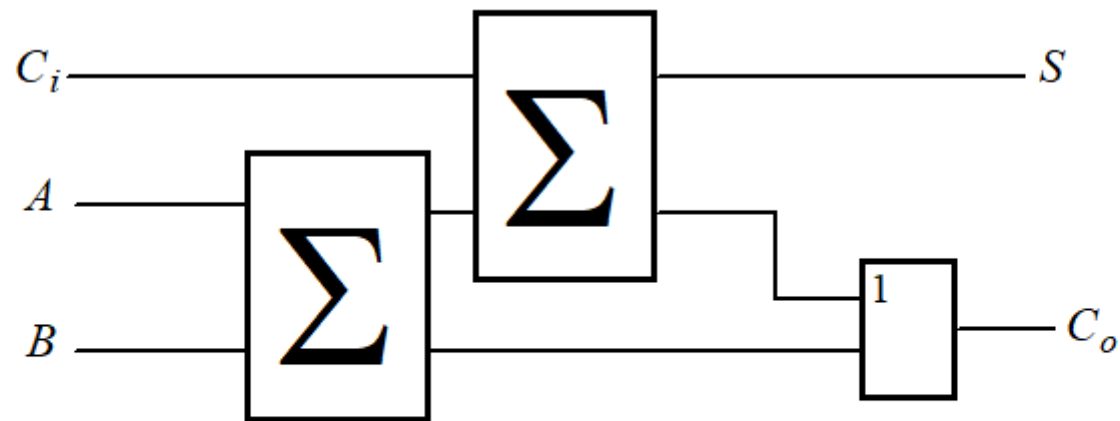
4.4.5 Obvody na realizáciu aritmetických operácií



Obr. 4.9 Realizácia jednotbitovej úplnej sčítačky

4.4.5 Obvody na realizáciu aritmetických operácií

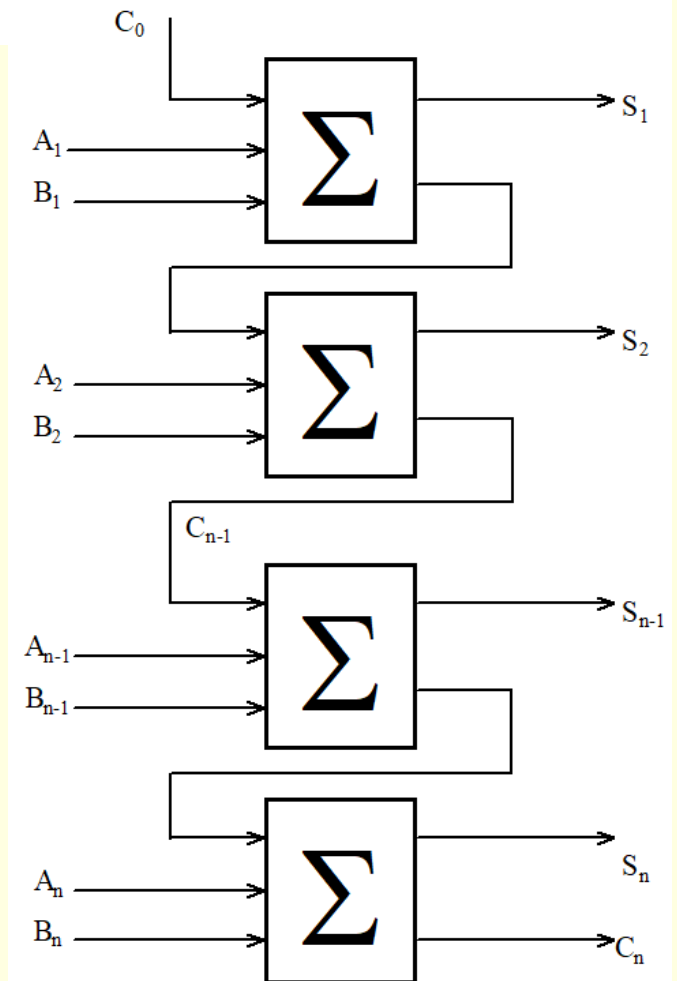
Úplnú jednobitovú sčítačku možno realizovať tiež pomocou dvoch neúplných sčítačiek podľa Obr. 4.10.



Obr. 4.10 Realizácia jednobitovej úplnej sčítačky pomocou dvoch neúplných sčítačiek

4.4.5 Obvody na realizáciu aritmetických operácií

Viacbitové sčítačky možno vytvoriť kaskádnym zapojením úplných jednobitových sčítačiek podľa Obr. 4.11.



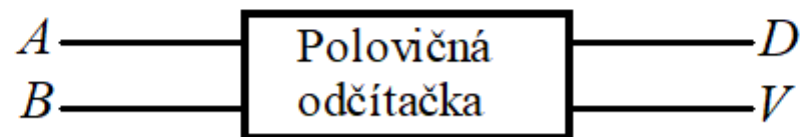
Obr. 4.11 Viacbitová sčítačka

4.4.5 Obvody na realizáciu aritmetických operácií

4.4.5.3 Odčítačky

Odčítačky realizujú operáciu binárneho odčítania. Analogicky ako pri sčítaní možno definovať polovičnú a úplnú sčítačku, pri odčítaní sa definuje *polovičná a úplná odčítačka*.

Polovičná odčítačka má dva vstupy A, B a dva výstupy D a V , kde D je rozdiel $A - B$ a V je výpožička (Obr. 4.12). Činnosť polovičnej odčítačky je daná pravdivostnou tabuľkou Tab. 4.9.



Obr. 4.12 Bloková schéma jednobitovej polovičnej odčítačky

4.4.5 Obvody na realizáciu aritmetických operácií

Tab. 4.9 Pravdivostná tabuľka jednobitovej polovičnej odčítačky.

Vstupy		Výstupy	
<i>A</i>	<i>B</i>	<i>D</i>	<i>V</i>
0	0	0	0
0	1	1	1
1	0	1	0
1	1	0	0

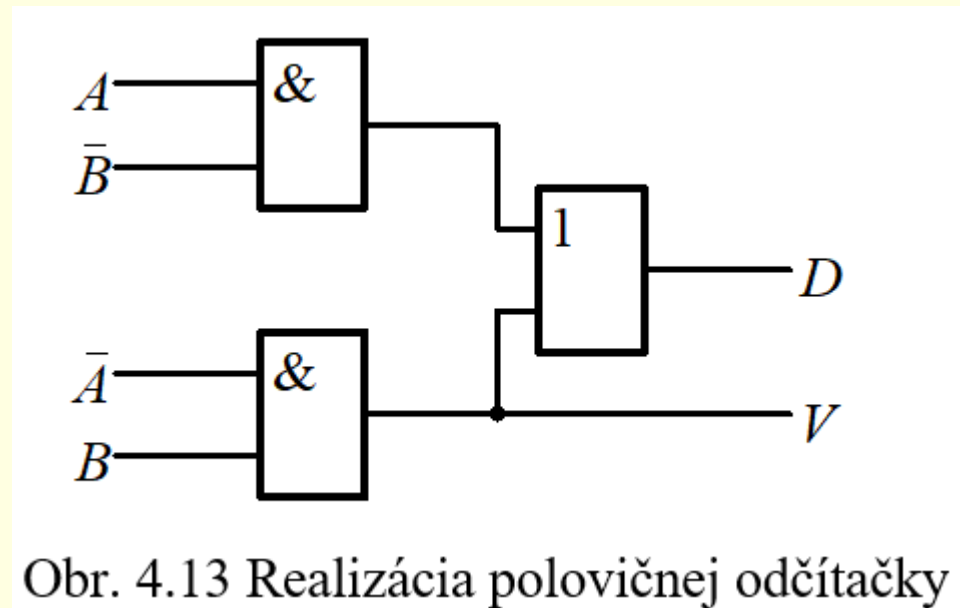
Po prepísaní tabuľky Tab. 4.9 do Karnaughových máp možno polovičnú odčítačku opísať vzťahmi

$$D = \bar{A}B + A\bar{B} = A \oplus B \quad (4.16)$$

$$V = \bar{A}B \quad (4.17)$$

Realizácia polovičnej odčítačky je na Obr. 4.13. Polovičná odčítačka nezohľadňuje výpožičku z predošlého rádu.

4.4.5 Obvody na realizáciu aritmetických operácií

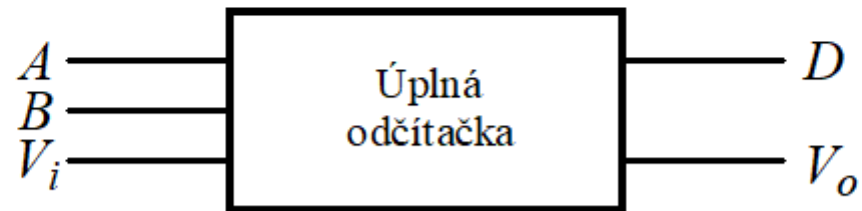


Obr. 4.13 Realizácia polovičnej odčítacky

4.4.5 Obvody na realizáciu aritmetických operácií

Úplná odčítačka má tri vstupy A, B, V_i a dva výstupy D a V_o , kde V_i je výpožička z predošlého rádu a V_o je výpožička na výstupe.

Bloková schéma úplnej odčítačky je na Obr. 4.14 a jej činnosť je daná pravdivostnou tabuľkou Tab. 4.10.



Obr. 4.14 Bloková schéma jednobitovej úplnej odčítačky

4.4.5 Obvody na realizáciu aritmetických operácií

Tab. 4.10 Pravdivostná tabuľka jednobitovej úplnej odčítacky.

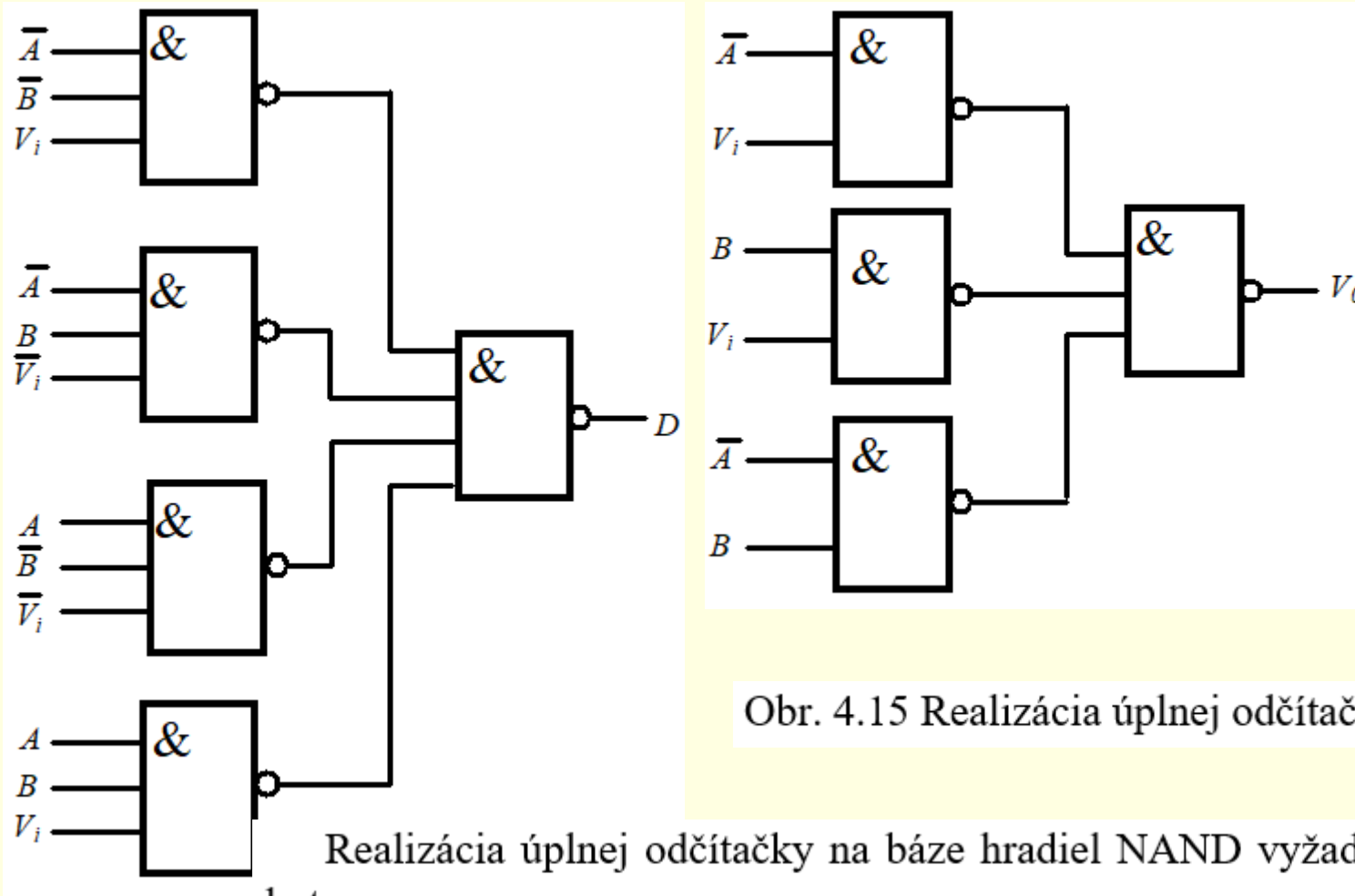
Vstupy			Výstupy	
A	B	V_i	D	V_o
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1

Po prepísaní pravdivostnej tabuľky do Karnaughových máp možno pre úplnú odčítacku odvodiť vzťahy

$$D = \bar{A}\bar{B}V_i + \bar{A}B\bar{V}_i + A\bar{B}\bar{V}_i + ABV_i \quad (4.18)$$

$$V = \bar{A}V_i + BV_i + \bar{A}B \quad (4.19)$$

4.4.5 Obvody na realizáciu aritmetických operácií



Obr. 4.15 Realizácia úplnej odčítacky

Realizácia úplnej odčítacky na báze hradieľ NAND vyžaduje úpravu vzťahov do tvaru

$$D = \overline{\overline{\overline{ABV_i}} \cdot \overline{\overline{ABV_i}} \cdot \overline{\overline{ABV_i}} \cdot \overline{\overline{ABV_i}}} \quad (4.20)$$

$$V = \overline{\overline{\overline{AV_i}} \cdot \overline{\overline{BV_i}} \cdot \overline{\overline{AB}}} \quad (4.21)$$

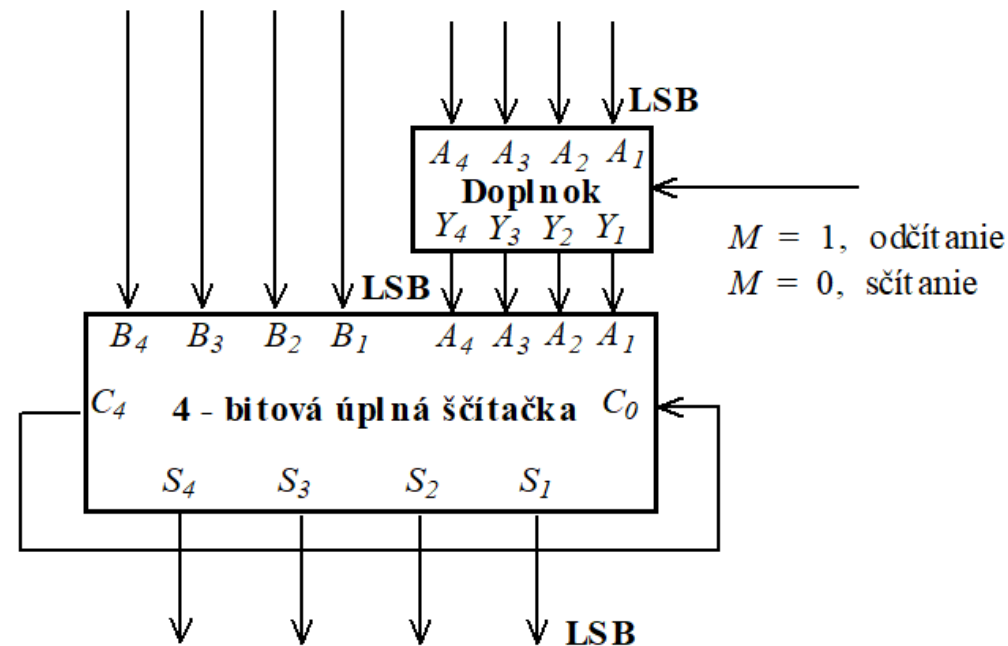
4.4.5 Obvody na realizáciu aritmetických operácií

4.4.5.5 Realizácia sčítačiek a odčítačiek

Operáciu odčítania možno realizovať dvoma spôsobmi:

- metódou 1. doplnku,
- metódou 2. doplnku.

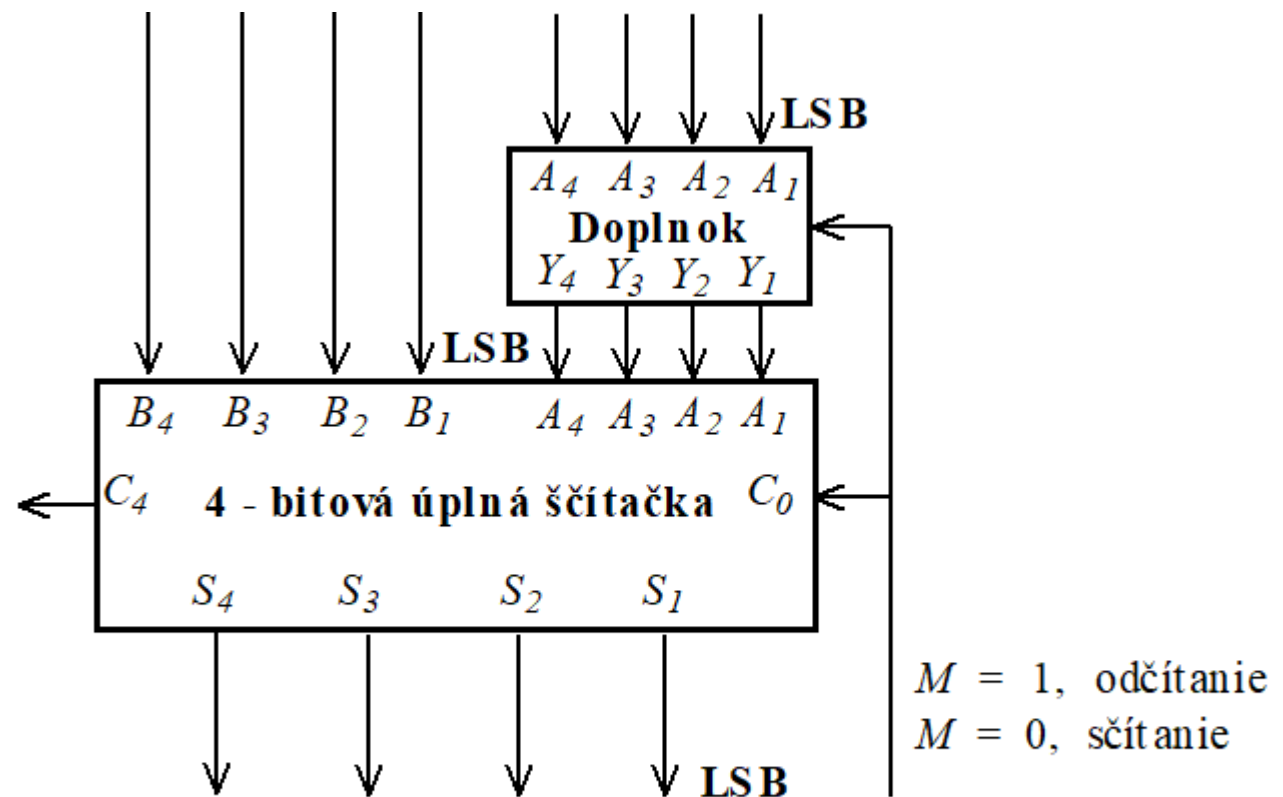
Princíp oboch metód spočíva v transformácii odčítania na sčítanie. Tento princíp umožňuje využiť sčítačku vo funkcii odčítačky. Blokova schéma odčítačky, realizujúca odčítanie metódou 1. doplnku je na Obr. 4.16.



Obr. 4.16 Odčítačka realizujúca odčítanie metódou 1. doplnku

4.4.5 Obvody na realizáciu aritmetických operácií

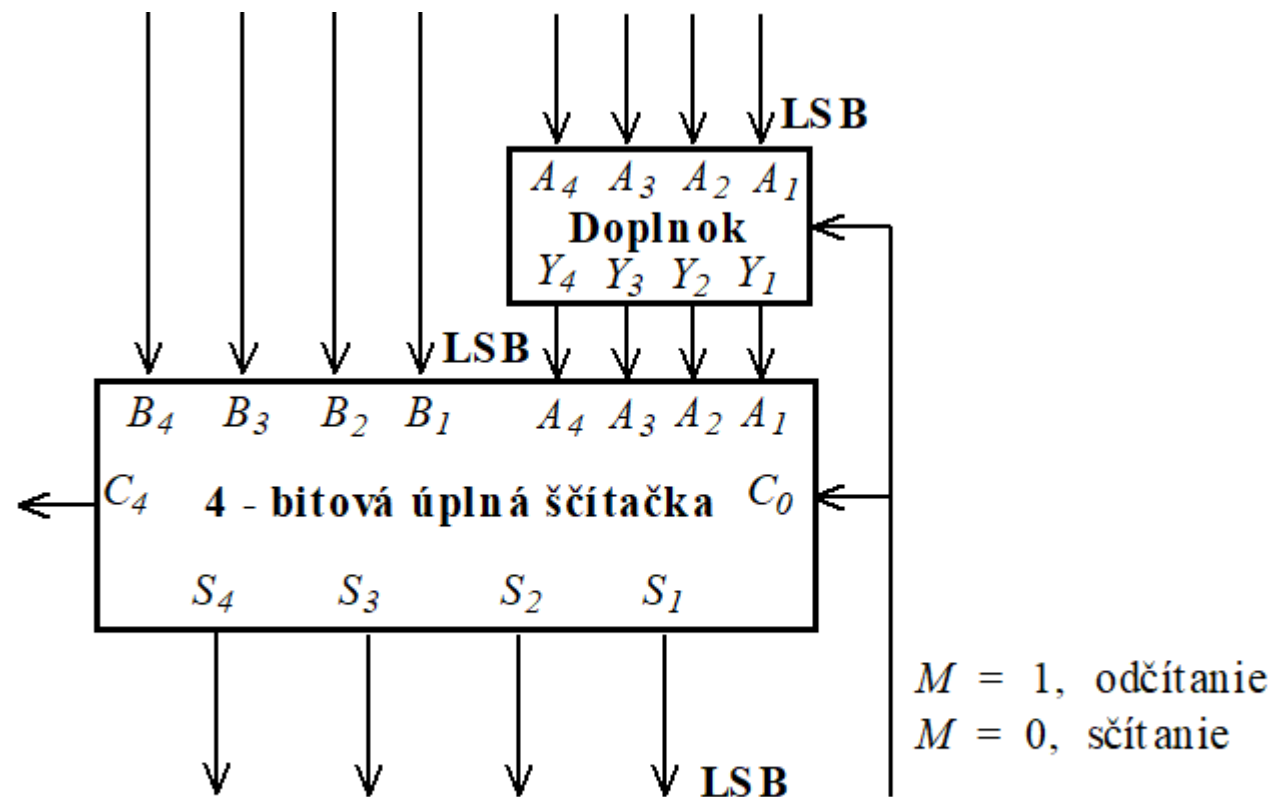
Odčítačka na Obr. 4.17 realizuje odčítanie metódou 2. doplnku. Ak $M=1$, potom obvod pre výpočet doplnku poskytuje 1. doplnok k sčítancu $A_4A_3A_2A_1$ a pretože $C_0=1$, realizuje sa pričítanie 2. doplnku. Ak $M=0$, sčítačka realizuje sčítanie.



Obr. 4.17 Odčítačka realizujúca odčítanie metódou 2. doplnku

4.4.5 Obvody na realizáciu aritmetických operácií

Odčítačka na Obr. 4.17 realizuje odčítanie metódou 2. doplnku. Ak $M=1$, potom obvod pre výpočet doplnku poskytuje 1. doplnok k sčítancu $A_4A_3A_2A_1$ a pretože $C_0=1$, realizuje sa pričítanie 2. doplnku. Ak $M=0$, sčítačka realizuje sčítanie.



Obr. 4.17 Odčítačka realizujúca odčítanie metódou 2. doplnku

4.4.5 Obvody na realizáciu aritmetických operácií

4.4.5.6 Aritmeticko-logické jednotky

Okrem sčítačiek a odčítačiek sa realizujú aj kombinačné obvody, ktoré umožňujú vykonávať aj logické operácie. Tieto obvody teda realizujú aritmetické a logické operácie, preto sa označujú ako *aritmeticko-logické jednotky*. Ako príklad môžeme uviesť rýchlu aritmeticko-logickú jednotku 74181, ktorá vykonáva 16 aritmetických a logických operácií s dvoma 4-bitovými slovami.

5 Sekvenčné obvody

■ 5.1 Klopné obvody

- 5.1.1 *RS klopný obvod*
- 5.1.2 *D klopný obvod*
- 5.1.3 *T klopný obvod*
- 5.1.4 *JK klopný obvod*

■ 5.2 Čítače

- 5.2.1 *Asynchrónne čítače*
- 5.2.2 *Synchrónne čítače*

■ 5.3 Posuvné registre

- 5.3.1 *Typy posunov*
- 5.3.2 *Realizácia posuvných registrov*