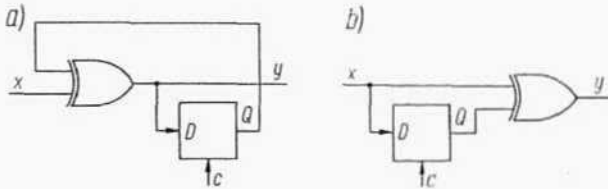


szeregową wówczas, gdy konwerter równoległy jest układem iteracyjnym. Na przykład z rys. 5-10 otrzymuje się bez trudu schematy odpowiednich konwerterów szeregowych, przy wprowadzaniu sygnałów w kolejności $x_n, x_{n-1} \dots x_1$ (rys. 5-19).



Rys. 5-19. Konwertery szeregowy: a) kodu Gray'a na kod naturalny; b) kodu naturalnego na kod Gray'a

5.3. SUMATORY I KOMPARATORY

5.3.1. SUMATORY JEDNOBITOWE

Podstawowymi członami układów realizujących sumę arytmetyczną są *sumatory jednobitowe*, opisywane już w rozdz. 3. Ich działanie jest określone funkcjami:

$$s = \bar{a}\bar{b}\bar{p} + \bar{a}\bar{b}p + \bar{a}b\bar{p} + abp$$

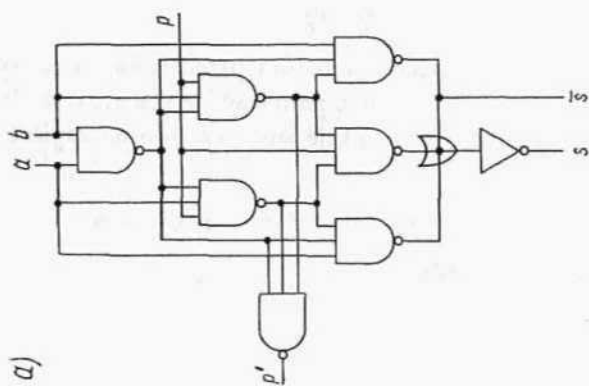
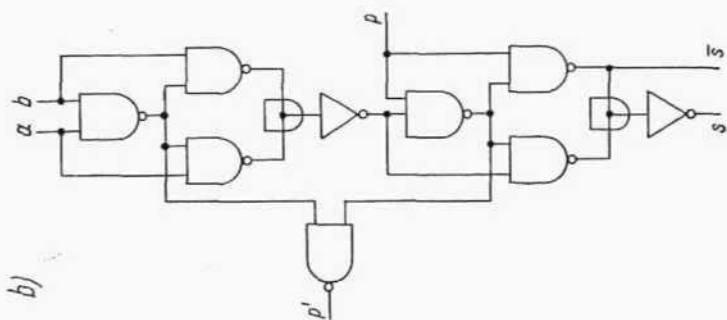
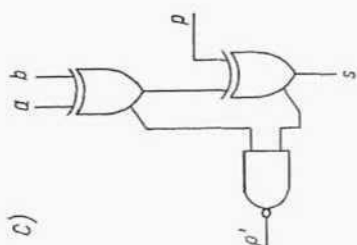
$$p' = ab + ap + bp$$

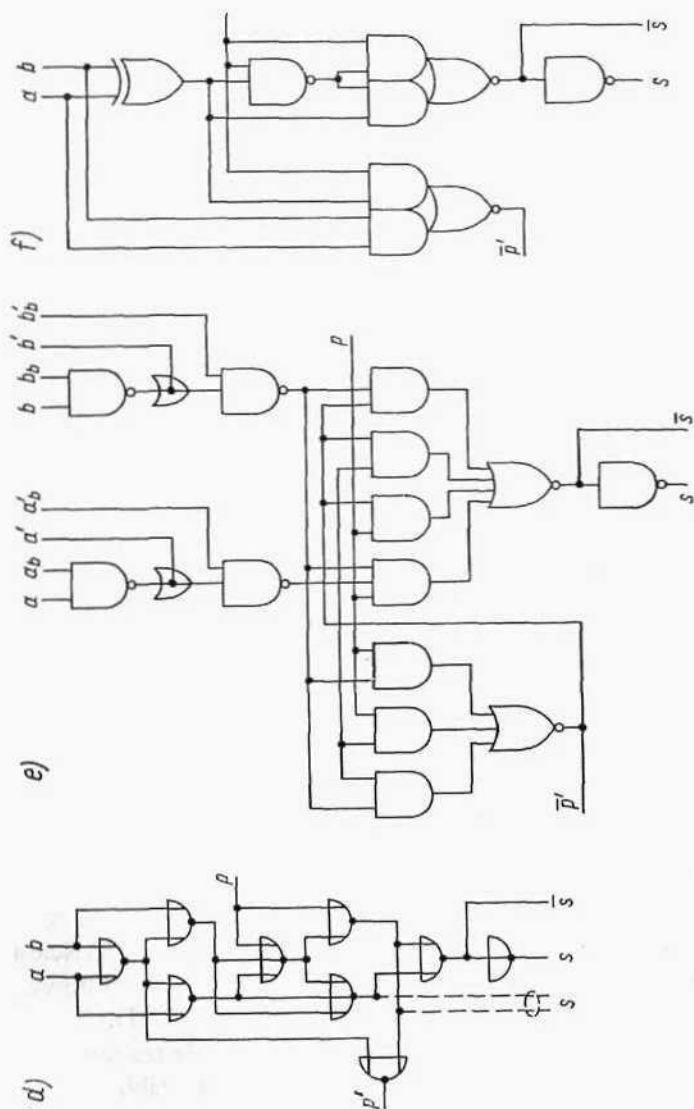
Praktyczny schemat takiego sumatora przedstawiono na rys. 3-38, a jego odmianę na rys. 5-20a. Warto pamiętać, że schematy z elementów NOR i NAND są jednakowe. Funkcje sumatora można zapisać w nieco innej postaci:

$$s = (\bar{a}b + a\bar{b})\bar{p} + (ab + \bar{a}\bar{b})p = a \oplus b \oplus p$$

$$p' = ab\bar{p} + \bar{a}bp + \bar{a}b\bar{p} + abp = ab + (a \oplus b)p$$

co umożliwia wykorzystanie półsumatorów, tj. układów realizujących sumę mod 2. Przykładowy układ jest przedstawiony na rys. 5-20b; w podobny sposób można zastosować również inne wersje z rys. 5-11, co symbolicznie zaznaczono na rys. 5-20c. Wykorzystanie wiązki prowadzi do układu z rys. 5-20d. Jeśli sygnał s może być również przekazany



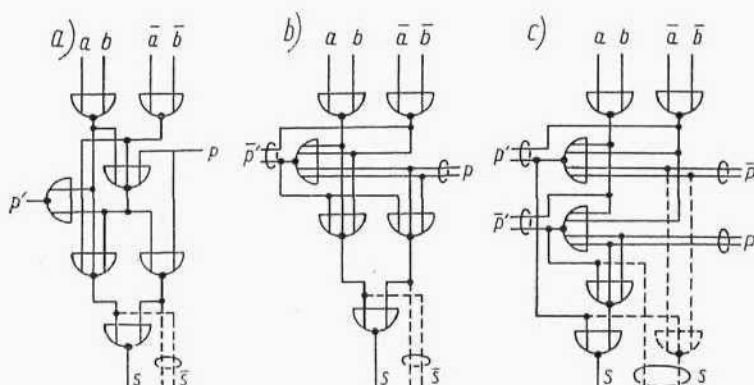


Rys. 5-20. Sumatory jednobitowe (bez negacji argumentów)

wiązką przewodów, realizacja ta dodatkowo uprości się. Układ z rys. 5-20e jest stosowany w realizacjach scalonych MSI. Jego pierwsze warstwy stanowią elementy pomocnicze, a właściwy sumator wykorzystuje zależność p.3.2.2

$$s = abp + \bar{p}'(a + b + p)$$

Układy upraszczają się, gdy do dyspozycji są także negacje argumentów. Schemat z rys. 5-21 jest szczególnym przypadkiem rozwiązania



Rys. 5-21. Sumatory jednobitowe (z negacjami argumentów)

z rys. 5-20c. Jeśli zamiast jednego przewodu p' następny stopień może wykorzystać \bar{p}' w postaci wiązki, to z układu można usunąć jeden element (rys. 5-21b). Dodatkową ważną zaletą takiego rozwiązania jest przyspieszenie pracy, gdyż sygnał przeniesienia przechodzi tu tylko przez jeden element, a w układzie iteracyjnym od szybkości rozchodzenia się przeniesień zależy całkowity czas sumowania. Wadą układu jest negowanie przeniesienia, co komplikuje (w niewielkim stopniu) układ wielobitowy. Wolny od tej wady jest sumator (rys. 5-21c) z szybkim przeniesieniem w postaci wiązki, zarówno p jak \bar{p} . Jeśli i wyjście s może być wykorzystane w postaci wiązki, pełny sumator jednobitowy można zbudować z pięciu zaledwie elementów NOR albo NAND.

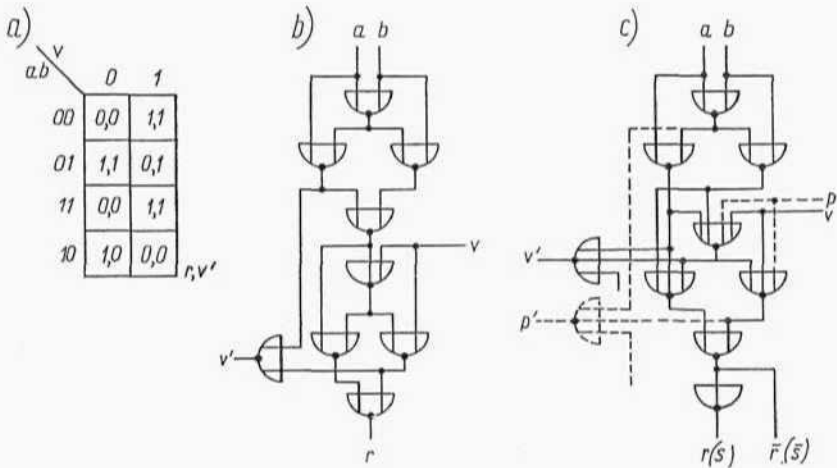
Nazwą „sumator” określa się często także układy realizujące różnicę. Dla działania ab różnicę r i pożyczkę v można określić z tablicy na rys. 5-22a

$$r = (\bar{a}\bar{b} + ab)v + (\bar{a}b + a\bar{b})\bar{v}$$

$$v' = \bar{a}b + \bar{a}v + bv$$

Jak widać, jedyna istotna różnica między tymi wzorami a odpowiednimi zależnościami dla układu sumy polega na występowaniu \bar{a} w równaniu pożyczki. Wynikają stąd dwie możliwości zrealizowania układu różnicy:

1) w układach sumy podstawić \bar{a} w miejsce a ; wówczas p zamieni się na v , \bar{s} — na r ;

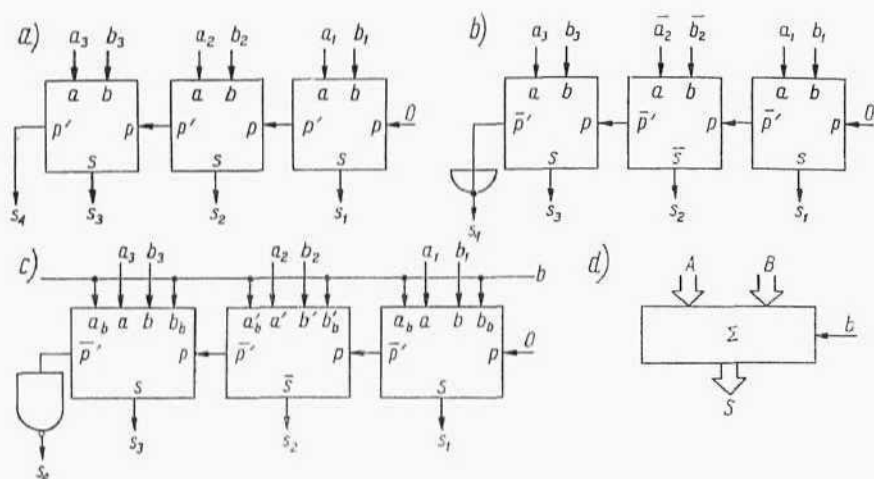


Rys. 5-22. Układy różnicy: a) tablica; b, c) schematy

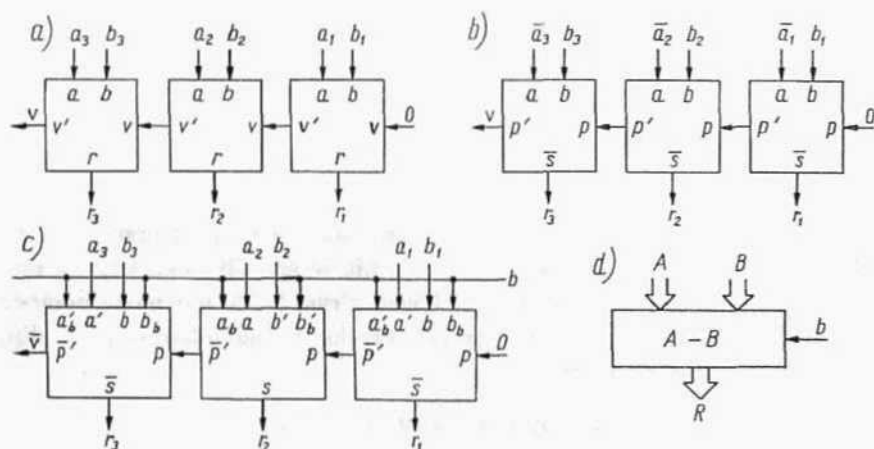
2) dla r zbudować układ identyczny jak dla s , natomiast jeden ze składowych sygnałów przeniesienia zmienić w sposób pokazany na rys. 5-22b,c, przez co p zmieni się w v . Układ z rys. 5-22c można za pomocą dodatkowych sygnałów bramkujących przełączać na odejmowanie albo dodawanie (linia przerywana).

5.3.2. SUMATORY DWÓJKOWE RÓWNOLEGŁE

Działania arytmetyczne wykonuje się najprościej, gdy argumenty występują w naturalnym kodzie dwójkowym. Sumatory są wówczas prostymi układami iteracyjnymi, złożonymi z sumatorów jednobitowych. W zależności od rodzaju użytych członów, zmienia się nieco sposób połączeń w układzie, co pokazano na rys. 5-23. Jeśli każdy człon generuje p i s , układ jest oczywisty (rys. 5-23a). Jeśli każdy człon generuje \bar{p} , to można wykorzystać szczególną cechę sumatora jednobitowego: zanego-



Rys. 5-23. Równoległe sumatory dwójkowe (a,b,c) i ich symbol (d)



Rys. 5-24. Równoległe układy różnicy (a,b,c) i ich symbol (d)

wanie wszystkich sygnałów wejściowych powoduje pojawienie się zanegowanych sygnałów wyjściowych. Skoro więc w drugim członie sygnał wejściowy przeniesienia musi być zanegowany, trzeba dostarczyć zanegowane a_2 i b_2 , wynik (s_2) brać z \bar{s} , a przeniesienie do trzeciego członu będzie już bez negacji. Po każdym członie nieparzystym prze-

niesienie jest zanegowane i dlatego potrzebny jest negator. Negowanie wejść członów parzystych może być realizowane poza układem (rys. 5-23b) lub wewnątrz układu (rys. 5-23c). Człony wg rys. 5-20 umożliwiają bramkowanie sygnałów wejściowych (oddzielne lub łączne); w innych rozwiązaniach takie sygnały również można wprowadzić. Do pierwszego członu wprowadza się przeniesienie 0, albo zastępuje się go innym układem, tj. półsumatorem (o dwóch wejściach).

Struktury układów różnicy przedstawiono na rys. 5-24. Jeśli członem układu iteracyjnego jest jednobitowy układ różnicy — połączenia są oczywiste (rys. 5-24a). Jeśli stosuje się jednobitowe układy sumy, trzeba negować bity odjemnika i korygować zmiany sumowania. W przypadku gdy człon generuje przeniesienia bez negacji (rys. 5-24b), wynik pojawi się na wyjściach \bar{s} , jeśli natomiast człony negują przeniesienie (rys. 5-24c) — tylko co drugie wyjście będzie z \bar{s} .

5.3.3. SUMATORY DWÓJKOWO-DZIESIĘTNE RÓWNOLEGŁE

Dodawanie liczb przedstawionych w kodach dwójkowo-dziesiętnych wymaga, zgodnie z rozważaniami z p.1.4.2, wprowadzenia korekcji przesuwałcej na skali liczbowej wyniki trafiające w niewykorzystywany zakres.

W przypadku kodu 8421 korekcja polega na dodaniu liczby 6, gdy pierwotna suma jest większa od 9. Wtedy również przeniesienie do następnej tetrady jest równe 1, co można wyrazić funkcją

$$p' = p_5 + s_4s_3 + s_4s_2 = (p_5 + s_4)(p_5 + s_3 + s_2)$$

Dodawanie liczby 6 realizuje się łatwo dodatkowym sumatorem, a więc ogólna struktura całego układu jest taka jak na rys. 5-25a. Dwa sumatory jednobitowe z dolnej warstwy można uprościć.

W przypadku kodu Aikena (IV) korekcja jest potrzebna, gdy pierwotna suma jest większa od (0100) i mniejsza od (1011), a polega na dodaniu liczby 6, gdy $p' = 0$, oraz odjęciu liczby 6 — gdy $p' = 1$. Zamiast odejmować 6 można dodać 10, gdyż sumator 4-bitowy bez przeniesienia działa modulo 16 (a $16 - 6 = 10$). Struktura całego układu jest przedstawiona na rys. 5-25b. Blok K generuje sygnał o wartości 1, gdy są spełnione podane wyżej warunki korekcji.