

liczników, układ jednostopniowy wymaga wprowadzenia na wejście elementów I sygnałów jednego i drugiego licznika (rys. 5-9b). Tworzy się w ten sposób charakterystyczny *dekoder współrzędnościowy*. Liczba sygnałów w grupach wejściowych ograniczona jest liczbą wejść elementów wyjściowych, ale i tu można stosować dekodowanie uproszczone. Szczególnie proste układy uzyskuje się, gdy sygnały wejściowe tworzą kody pseudopierścieniowe; element I o czterech wejściach może wówczas dekodować stan np. dwóch dowolnie długich liczników. Kody w różnych grupach sygnałów wejściowych mogą też być różne, a liczba współrzędnych układu nie musi być ograniczona do dwóch.

### 5.2.3. TRANSLATORY KODÓW I UKŁADY UZUPEŁNIAJĄCE

Te konwertery kodów, które nie są koderami i dekoderami, bywają nazywane *translatorami kodów* i w zasadzie mogą być budowane przez połączenie wyjść odpowiedniego dekodera z wejściami kodera. Taka realizacja w większości przypadków nie jest jednak optymalna. Najlepsze wyniki daje synteza tych układów jako typowych układów wielowyjściowych, metodami wprowadzonymi wyżej. Na przykład konwerter kodu Gray'a na naturalny kod dwójkowy, w przypadku trzech zmiennych można opisać tablicą z rys. 5-10, z której otrzymuje się

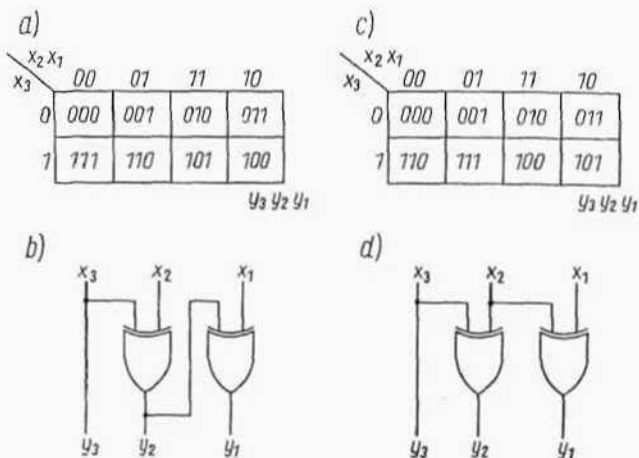
$$\begin{aligned} y_3 &= x_3 \\ y_2 &= x_2 \bar{x}_3 + \bar{x}_2 x_3 = x_2 \oplus x_3 \\ y_1 &= x_1 \bar{x}_2 \bar{x}_3 + \bar{x}_1 x_2 \bar{x}_3 + \bar{x}_1 \bar{x}_2 x_3 + x_1 x_2 x_3 = \\ &= x_1 (\bar{x}_2 \bar{x}_3 + x_2 x_3) + \bar{x}_1 (\bar{x}_2 x_3 + x_2 \bar{x}_3) = x_1 \oplus y_2 \end{aligned}$$

Zaobserwowana w tych zależnościach prawidłowość umożliwia budowanie konwerterów o większej liczbie wejść, z zachowaniem iteracyjnej struktury układu z rys. 5-10b, tzn. ze związku

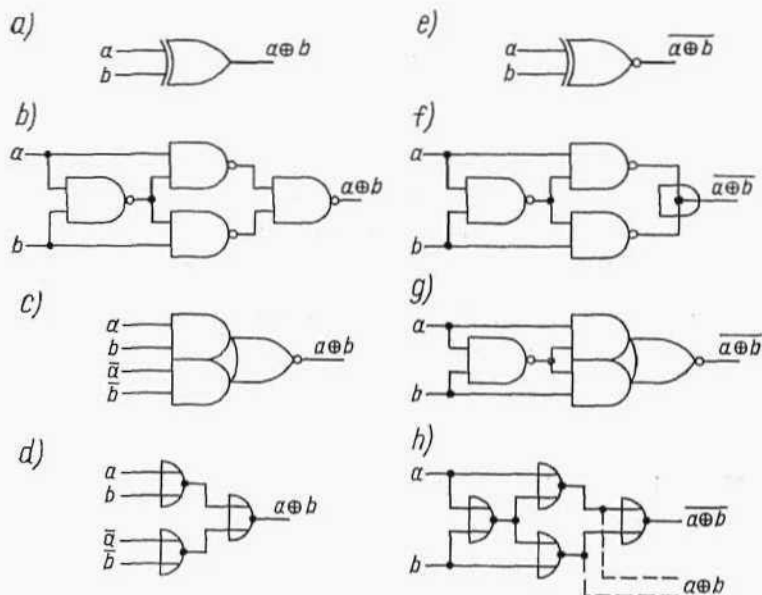
$$y_i = x_i \oplus y_{i+1}$$

W podobny sposób dla konwertera naturalnego kodu dwójkowego na kod Gray'a otrzymuje się z tablicy (rys. 5-10c)

$$\begin{aligned} y_3 &= x_3 \\ y_2 &= x_2 \bar{x}_3 + \bar{x}_2 x_3 = x_2 \oplus x_3 \\ y_1 &= x_1 \bar{x}_2 + \bar{x}_1 x_2 = x_1 \oplus x_2 \end{aligned}$$



Rys. 5-10. Tablice i schematy konwerterów: a,b kodu Gray'a na kod naturalny; b,c) kodu naturalnego na kod Gray'a

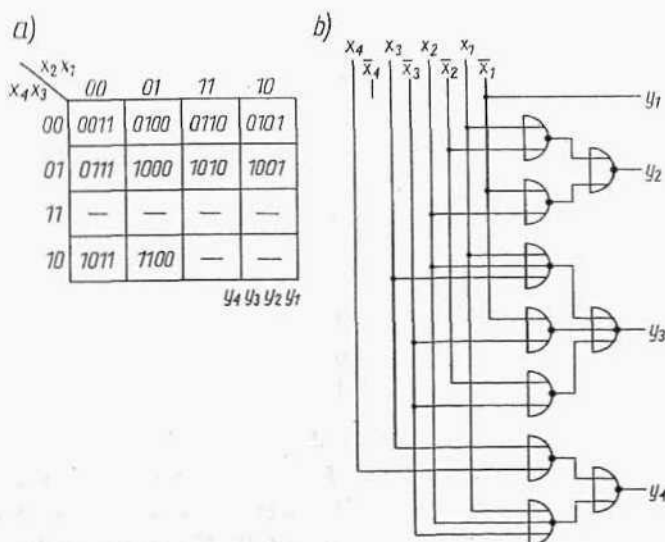


Rys. 5-11. Oznaczenie (a) i przykłady realizacji (b,c,d) układu sumy modulo 2, oraz oznaczenie (e) i przykłady realizacji (f,g,h) układu równoważności

Odpowiedni układ jest przedstawiony na rys. 5-10d i może być łatwo rozbudowany, przy zachowaniu zależności

$$y_i = x_i \oplus x_{i+1}$$

W schematach z rys. 5-10 użyto symbolu układu realizującego funkcję sumy modulo 2, co umożliwiło pokazanie struktury, bez narzucania konkretnego rozwiązania. Przykłady rozwiązań układu sumy modulo 2 przedstawiono na rys. 5-11b, c, d. Równie często jest stosowany symbol układu, realizującego funkcję równoważności (rys. 5-11e). W podanych przykładach wykorzystano znane już układy (rys. 3-29, 3-34) i ich modyfikacje. Rozwiązanie z rys. 5-11c stosuje jeden tylko element wchodzący w skład wielu systemów, zwłaszcza scalonych (rys. 2-11). W rozwiązaniu z rys. 5-11f wyjścia elementów NAND są zwarte, a tranzystory ostatnich stopni w tych elementach mają wspólną rezystancję w obwodzie kolektorów, co sprawia, że sygnał 1 na wyjściu może się pojawić tylko wówczas, gdy nie przewodzi żaden z tych tranzystorów. Zwarcie wyjść wnosi więc do układu funkcję iloczynu, co symbolicznie zaznaczono na schemacie.



Rys. 5-12. Tablica i schemat konwertera kodu 8421 na „+3”

Układy z rys. 5-11 można stosować w bardzo różnych konfiguracjach, pamiętając, że zanegowanie jednej dowolnej zmiennej w funkcji  $a \oplus b$  daje funkcję  $a \oplus \bar{b}$ , i przeciwnie. Na przykład schemat z rys. 5-10b można przystosować do realizacji z rys. 5-11g, wprowadzając zanegowane sygnały  $x$ , a jeśli w układzie z rys. 5-10d potrzebne są negacje sygnałów  $y$  wystarczy zamiast układów  $a \oplus b$  zastosować tu którąś z realizacji  $a \oplus \bar{b}$ .

Przykładem innego konwertera może być konwerter dwójkowo-dziesiętnego kodu 8421 (I) na kod „+3” (V), opisany tablicą z rys. 5-12a, z której otrzymuje się:

$$y_4 = x_4 + x_1 x_3 + x_2 x_3 = (x_3 + x_4)(x_1 + x_2 + \bar{x}_3)$$

$$y_3 = x_2 \bar{x}_3 + x_1 \bar{x}_3 + \bar{x}_1 \bar{x}_2 x_3 = (x_1 + x_2 + x_3)(\bar{x}_1 + \bar{x}_3)(\bar{x}_2 + \bar{x}_3)$$

$$y_2 = x_1 x_2 + \bar{x}_1 \bar{x}_2 = (x_1 + \bar{x}_2)(\bar{x}_1 + x_2)$$

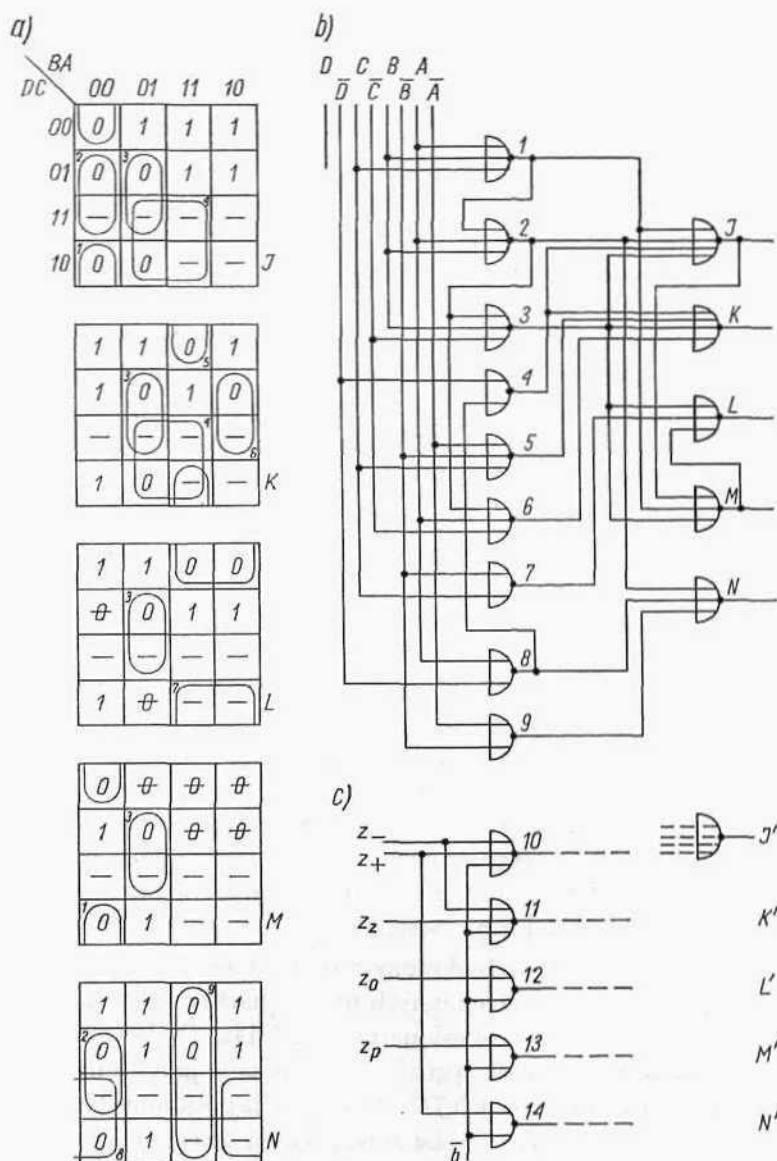
$$y_1 = \bar{x}_1$$

Jedna z możliwych realizacji jest przedstawiona na rys. 5-12b.

Trudniejsza jest synteza konwerterów przetwarzających cyfry w kodzie 8421 na kod dalekopisowy i przeciwnie. Zależność między tymi kodami można przedstawić tablicą

	D	C	B	A	Ƴ	K	L	M	N
0	0	0	0	0	0	1	1	0	1
1	0	0	0	1	1	1	1	0	1
2	0	0	1	0	1	1	0	0	1
3	0	0	1	1	1	0	0	0	0
4	0	1	0	0	0	1	0	1	0
5	0	1	0	1	0	0	0	0	1
6	0	1	1	0	1	0	1	0	1
7	0	1	1	1	1	1	1	0	0
8	1	0	0	0	0	1	1	0	0
9	1	0	0	1	0	0	0	1	1

z której wynika, że:  $F^1(D) \subseteq F^0(C)$ ,  $F^1(D) \subseteq F^0(B)$ ,  
 $F^1(B) \cup F^1(C) \subseteq F^0(D)$  oraz  $F^1(\text{Ƴ}) \subseteq F^0(M)$ ,  $F^1(L) \subseteq F^0(M)$ ,  
 $F^1(M) \subseteq F^0(\text{Ƴ})$  oraz  $F^1(M) \subseteq F^0(L)$ . Relacje te są bardzo przydatne przy budowaniu układów z elementów NOR. Konwerter kodu 8421 na kod dalekopisowy jest opisany tablicami z rys. 5-13a. Jeśli do realizacji



Rys. 5-13. Tablice (a) i schemat (b) konwertera kodu 8421 na kod dalekopisowy oraz dodatkowy układ (c) sygnałów pomocniczych

mają być użyte elementy NOR, układ można optymalizować dwoma sposobami:

- przedstawiając  $\mathcal{F}$  jako funkcję  $M$ , a  $M$  jako funkcję  $L$ .
- przedstawiając  $L$  jako funkcję  $M$ , a  $M$  jako funkcję  $\mathcal{F}$ . Drugi sposób okazuje się prostszy. Na rys. 5-13a przekreślono te zera, które są wymuszane innymi sygnałami wyjściowymi (jak na rys. 5-13b). Pozostałe zera w tablicach ujęto w grupy Karnaugh'a, dobierając grupy tak, aby mogły być wykorzystane w maksymalnej liczbie tablic. Dla ułatwienia, grupy i odpowiadające im elementy (rys. 5-13b) ponumerowano, a do wyznaczenia jednych grup wykorzystano inne, co zmniejszyło obciążenie sygnałów wejściowych. Jeśli wejście  $A$  nie może sterować czterech elementów, trzeba nieco zmienić schemat i grupę 8 wyznaczyć za pomocą 4 (a nie przeciwnie).

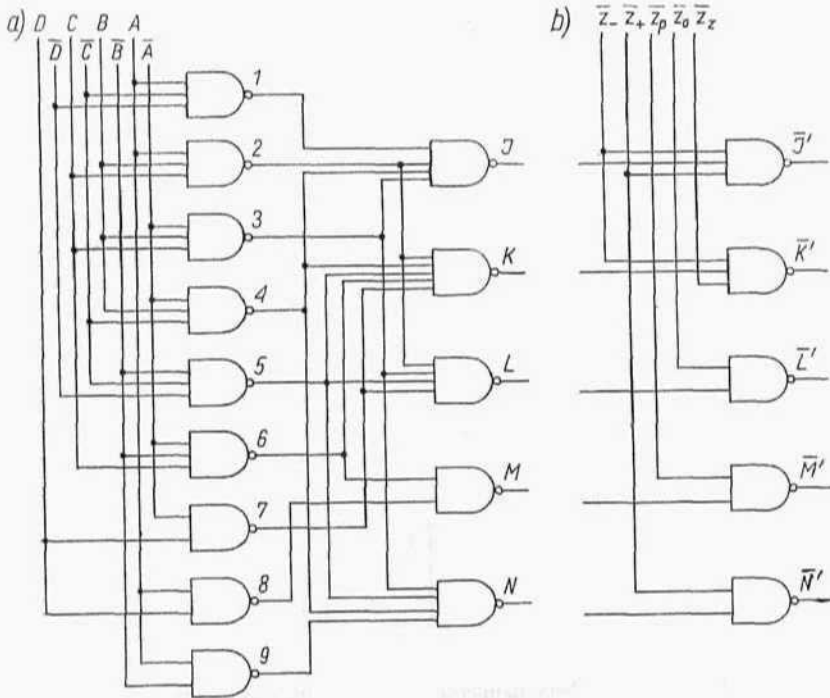
W układach praktycznych do opisanego typowego konwertera dodawane są na ogół układy pomocnicze, umożliwiające powstawanie na wyjściach  $\mathcal{F}KLMN$  sygnałów, odpowiadających poleceniom: „powrót karetki” ( $z_p$ ), „zmiana wiersza” ( $z_z$ ), „odstęp” ( $z_o$ ), a także — w miarę potrzeby — „+”, „-”, „.” itd. Zespół takich poleceń jest kodowany kodem „1 z  $n$ ” i może być wprowadzony do konwertera np. za pomocą układu z rys. 5-13c. Sygnał bramkujący  $b = 1$  powinien wymusić 0 na wyjściach elementów 1...9, a wówczas wyjścia 10...14, oddziaływując na elementy wyjściowe konwertera, generują odpowiednie znaki. Należy zwrócić uwagę, że dodatkowe sygnały wejściowe zmieniają relacje między sygnałami wyjściowymi konwertera, więc przy przyjętych na rys. 5-13b sprzężeniach nie wszystkie sygnały użyteczne pomocnicze mogą być realizowane. Na przykład wprowadzenie kropki wymaga zmiany połączeń elementu z wyjściem  $L$ , gdyż przestaje być prawdą, że  $F^1(M) \subseteq F^0(L)$ .

Jeśli konwerter ma być zbudowany z elementów NAND, wykorzystanie jednych wyjść do realizacji innych nie jest możliwe. Jedna z wersji takiego konwertera jest przedstawiona na rys. 5-14a. Dodatkowy układ (rys. 5-14b) może wprowadzać sygnały pomocnicze  $z$ , jeśli sygnał bramkujący wymusi stan 1 na wyjściach  $\mathcal{F}KLMN$ : wyjścia połączonych układów oznaczono przez  $\mathcal{F}'K'L'M'N'$ . Liczbę wejść  $z$  można dowolnie powiększać.

W podobny sposób buduje się konwertery kodu dalekopisowego na kod 8421 i wiele innych. Jeśli zachodzi potrzeba, wprowadzany jest

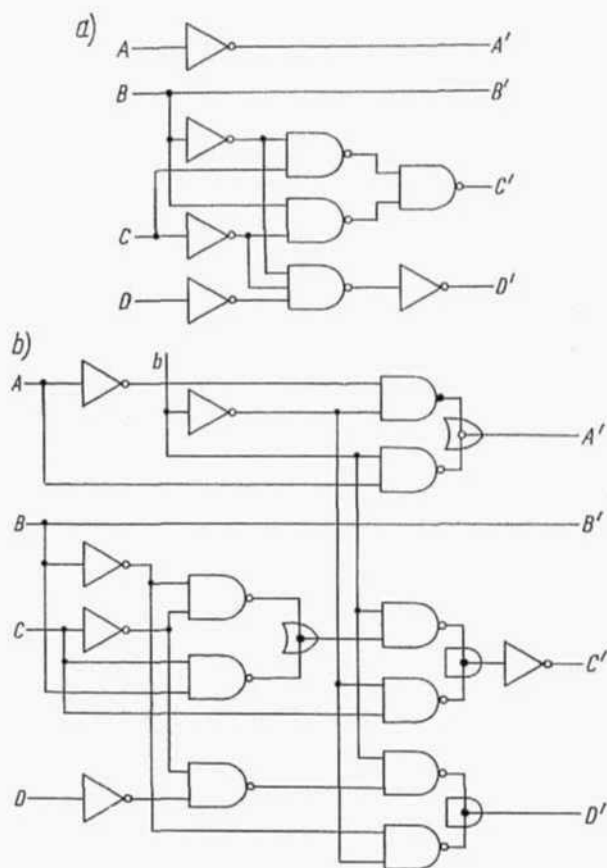
sygnał pomocniczy  $b$ , blokujący działanie układu, lub zmieniający realizowaną funkcję.

Specjalnym rodzajem konwerterów są *układy uzupełniające*, których zadaniem jest obliczanie uzupełnienia liczby wejściowej (p.1.4). Uzupełnianie do jedności sprowadza się do negowania, uzupełnianie do

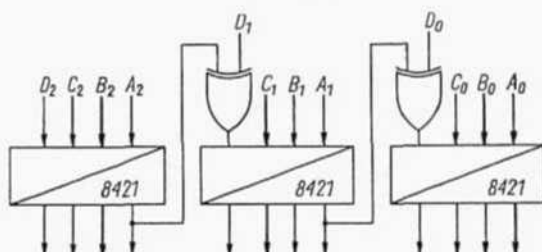


Rys. 5-14. Schemat konwertera kodu 8421 na kod dalekopisowy (a) i dodatkowy układ dla sygnałów pomocniczych (b)

dwóch i dziesięciu wykonuje się zazwyczaj w sumatorach, natomiast uzupełnianie do dziewięciu — w specjalnych układach. Przykłady takich układów dla kodu 8421 pokazano na rys. 5-15. W drugim rozwiązaniu, w zależności od wartości  $b$ , na wyjściu pojawia się liczba wejściowa albo jej uzupełnienie do 9. W przypadku kodów samouzupełniających się (np. IV, V), uzupełnianie polega na zanegowaniu wszystkich bitów. Można to zrealizować za pomocą takich układów, jakie zastosowano



Rys. 5-15. Układy uzupełniające do 9 dla kodu 8421



Rys. 5-16. Schemat blokowy konwertera kodów specjalnych XIII, XIV na kod 8421



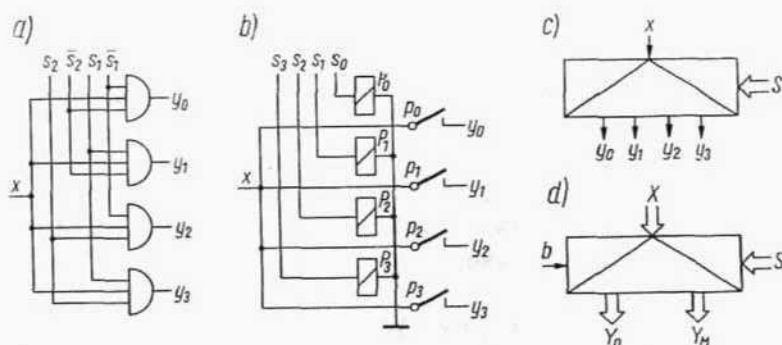
do uzyskania  $A'$  na rys. 5-15 (negator  $b$  będzie oczywiście wspólny dla wszystkich bitów), albo dowolny inny układ z rys. 5-11. W przypadku kodów XIII i XIV (tabl. 1-4) uzupełnianie sprowadza się do negowania  $D$ , a więc wystarczy jeden tylko układ sumy modulo 2 lub równoważności (w zależności od roli  $b$ ).

Kody XIII i XIV charakteryzują się jeszcze tym, że w systemach wielodekadowych wartość każdej tetrady zależy od tego, czy wartość starszej dekady jest parzysta czy nie. Przy przechodzeniu na kod 8421 albo inny, którego ostatni bit wyróżnia parzystość, można zastosować układ z rys. 5-16, na którym prostokąty oznaczają konwerter jednej dekady.

#### 5.2.4. KOMUTATORY

Układy nazywane komutatorami służą do selektywnego przełączania sygnałów i dzielą się na dwie grupy.

*Selektory (demultipleksery)* przekazują sygnał wejściowy do jednego z kilku torów wyjściowych, przy czym numer (adres) toru wyjściowego jest wybierany sygnałami pomocniczymi  $s$ . Na rys. 5-17 przedstawiono



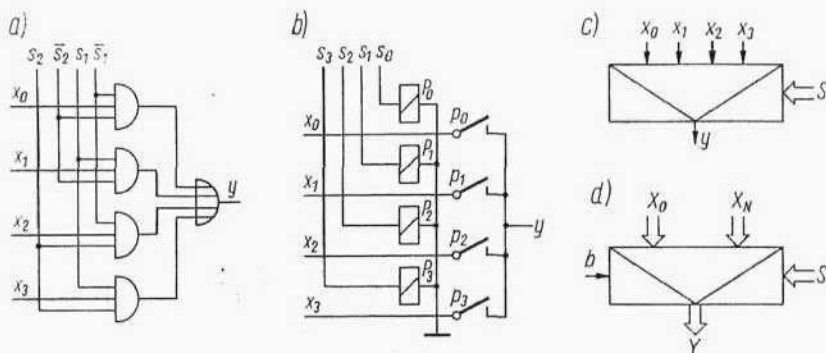
Rys. 5-17. Selektory: a) układ bezstykowy; b) układ stykowy; c) oznaczenie; d) oznaczenie selektora grupowego

typowy układ selektora z wybieraniem w kodzie naturalnym. Niekiedy jest stosowany jeszcze ogólny sygnał bramkujący  $b$ . Komutatory są również budowane w wersji stykowej, gdy współpracują z zewnętrznymi obwodami urządzeń cyfrowych. Umożliwiają wówczas galwaniczne

oddzielenie układów, a przy zastosowaniu odpowiednich przekazywników mogą też przełączać sygnały analogowe. Na rys. 5-17b przedstawiono selektor stykowy, z najczęściej w takich przypadkach stosowanym wybieraniem w kodzie „1 z  $n$ ”. Ogólnie selektor realizuje zespół funkcji

$$y_i = x \cdot f_i(S) \quad i = 0, 1, \dots, m$$

Zamiast przekazywać jeden sygnał wyjściowy na jedno wyjście, specjalne *selektory grupowe* mogą również przekazywać całą liczbę wejściową



Rys. 5-18. Kolektory: a) układ bezstykowy; b) układ stykowy; c) oznaczenie; d) oznaczenie kolektora grupowego

(grupę sygnałów) na jedno z wyjść grupowych. Odpowiedni symbol jest pokazany na rys. 5-17d.

*Kolektory (multipleksery)* przekazują jeden z sygnałów wejściowych (wybrany adresem  $S$ ) na wyjście układu. Typowy schemat przedstawiono na rys. 5-18a, a wersję stykową na rys. 5-18b. Układy realizują funkcję o ogólnej postaci

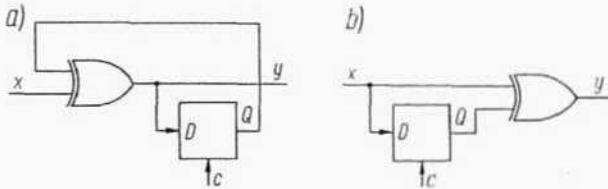
$$y = x_0 \cdot f_0(S) + x_1 \cdot f_1(S) + \dots + x_n \cdot f_n(S)$$

Podobnie jak w przypadku selektorów kody wyrażające  $S$  mogą być różne. Istnieją też kolektory grupowe.

### 5.2.5. KONWERTERY SZEREGOWE

Wszystkie opisywane wyżej konwertery przetwarzają kody o postaci równoległej. Nieliczne konwertery można zbudować również w wersji szeregowej, a szczególnie proste jest przejście od wersji równoległej do

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