

wać z separowania całych bloków i przeanalizować pary. Zamiast dodatkowego warunku 1,2,4-3, powstają wówczas dwa warunki 1,2-3 i 1,4-3, gdyż 2 i 4 przechodzą w 1. Łatwo można sprawdzić, że warunek oddzielenia 1,2-3 jest spełniony przez τ_1 , a 1,4-3 — przez τ_2 , a zatem kodowanie zgodnie z tymi podziałami nie powoduje wyścigów krytycznych i prowadzi do najprostszej realizacji.

Na podstawie powyższych rozważań i przykładów można zestawić następującą kolejność czynności przy kodowaniu:

- 1) wypisać podziały wewnętrzne;
- 2) na ich podstawie utworzyć podziały prawidłowe;
- 3) wybrać takie podziały prawidłowe, aby ich iloczynem był podział zerowy;
- 4) sprawdzić czy wybrane podziały separują wszystkie bloki podziałów wewnętrznych; jeśli tak — wykonać p. 9, jeśli nie — wykonać p. 5;
- 5) wypisać nie spełnione warunki separacji bloków i zamienić je na warunki separacji par;
- 6) sprawdzić, czy wybrane podziały prawidłowe separują te pary; jeśli tak — wykonać p. 9, jeśli nie — wykonać p. 7;
- 7) sprawdzić, czy wprowadzenie przejść cyklicznych likwiduje warunki separacji; jeśli tak — wprowadzić je i wykonać p. 9, jeśli nie — wykonać p. 8;
- 8) wprowadzić dodatkowy podział prawidłowy, realizujący warunki separacji i wykonać p. 9;
- 9) na podstawie wybranych podziałów prawidłowych utworzyć kod i zakodować tablice przejść i wyjść.

Piąty punkt tego algorytmu wiąże się ze zwiększeniem liczby elementów pamięciowych i jest stosowany tylko w ostateczności, gdy zawiodą inne sposoby uniknięcia wyścigu krytycznego.

Jeśli istnieje kilka wariantów rozwiązania, o wyborze mogą decydować pary podziałów lub podziały zewnętrzne, podobnie jak w układach synchronicznych.

4.3.4. REALIZACJA PAMIĘCI I FUNKCJE WZBUDZEŃ

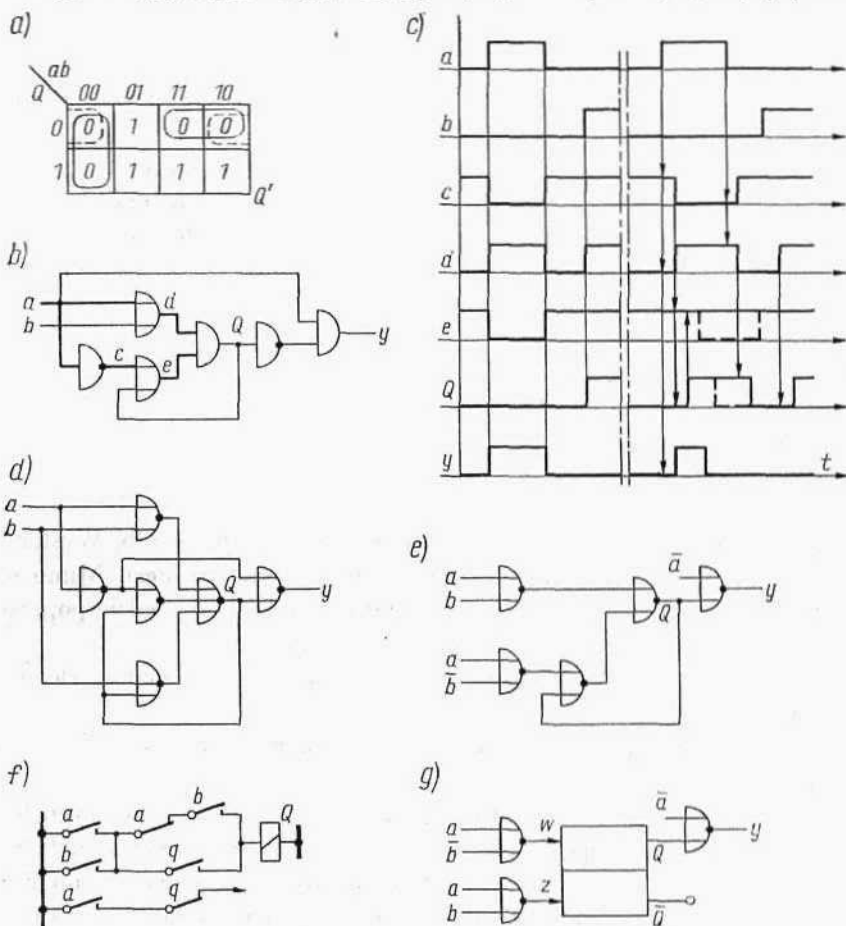
W układach asynchronicznych statycznych są stosowane dwie metody realizacji pamięci.

Pierwsza metoda wykorzystuje zależność (4-4)

$$A^{t+\tau} = \delta(A^t, X^t)$$

którą po zamianie stanów na sygnały (po zakodowaniu) można przedstawić w postaci rodziny równań

$$Q_i' = \delta_i(Q_1, Q_2, \dots, Q_k, x_1, x_2, \dots, x_n) \quad (i = 1, 2, \dots, k) \quad (4-14)$$



Rys. 4-38. Likwidowanie hazardu statycznego: a) zakodowana tablica przejść; b) schemat układu z hazardem statycznym; c) wykres czasowy układu z hazardem; d) schemat układu bez hazardu; e) inna postać schematu bez hazardu; f) układ przekaźnikowy; g) układ z wydzielonym przerzutnikiem

Opóźnienie τ (symbolizowane przez znak ') w układach asynchronicznych nie musi mieć ściśle określonej — jak w synchronicznych — wartości, wobec tego do jego realizacji nie są potrzebne żadne dodatkowe elementy i zależność (4-14) można uważać za równanie opisujące układ kombinacyjny o wejściach Q_i i x_j oraz wyjściach Q'_i . Każdy fizyczny układ kombinacyjny wprowadza pewne opóźnienie i dlatego zachowanie różnej symboliki dla wejść Q i wyjść Q' jest uzasadnione, oznacza jednak tylko kolejność występowania, gdyż są to te same sygnały.

Do wyrażenia związków (4-14) w postaci równań logicznych wystarczy zakodowana tablica przejść (najlepiej — jako tablica Karnaugh), którą uważa się za typową tablicę układu kombinacyjnego. O tym, że jest to układ sekwencyjny świadczą jedynie sprzężenia zwrotne, powstałe po połączeniu wyjść Q'_i z wejściem Q_i . Na przykład tablica przejść z rys. 4-32d, po przyjęciu $Q = A$, przybiera postać jak na rys. 4-38a, skąd można otrzymać równania

$$Q' = (a+b)(Q+\bar{a})$$

$$Q' = \bar{a}b + aQ$$

Z tablicy wyjść na rys. 4-32e uzyskuje się

$$y = a\bar{Q}$$

więc schemat logiczny układu może mieć postać z rys. 4-38b. Występująca tu pętla sprzężenia zwrotnego realizuje obwód pamięci. Mimo logicznej poprawności uzyskanego schematu jego działanie będzie poprawne, jeśli spełnione będą dwa dodatkowe wymagania:

- w obwodzie objętym sprzężeniem zwrotnym występuje element czynny,
- opóźnienia wprowadzane przez elementy nie zmieniają zasad pracy układu.

Pierwszy warunek wynika z konieczności wzmacniania sygnałów obiegających pętlę sprzężenia zwrotnego oraz separowania układów biernych; łatwo można sprawdzić, że objęcie sprzężeniem elementów diodowych nie daje żadnych możliwości zapamiętywania.

Drugi warunek wiąże się z faktem, że w układach często jeden sygnał przesyłany jest kilkoma drogami, o różnej liczbie elementów, a więc i różnych opóźnieniach. W rezultacie na końcowy element działają jak

gdyby różne sygnały, co może spowodować błędną pracę. Właśnie na rys. 4-38b występuje taka sytuacja, gdyż sygnał a wpływa na stan Q dwiema różnymi drogami (pogrubionymi), o różnej liczbie elementów. Fragment działania tego układu jest przeanalizowany na rys. 4-38c. Po lewej stronie pokazano idealne przebiegi czasowe, nie uwzględniające opóźnień wnoszonych przez elementy, natomiast po prawej — przebiegi realne, z dowolnie przyjętymi opóźnieniami. Z rysunku wynika, że na skutek opóźnienia wnoszonego przez negator (c), sygnał e nie zmieni się w tej samej chwili co sygnał d , więc przy zmianie z $a = 0$ na $a = 1$ powstaną warunki pojawienia się $Q = 1$. Gdyby nie działało sprzężenie zwrotne, impuls Q byłby krótkotrwały (przerywana linia na rys. 4-38c), ponieważ jednak sprzężenie istnieje — sygnał Q podtrzymuje się w trwały sposób zniekształcając sygnał wyjściowy.

Opisane zjawisko, polegające na pojawieniu się niewłaściwego stanu Q na skutek różnicy w czasie przesyłania sygnału po dwóch różnych drogach, nosi nazwę *hazardu (ryzyka) statycznego*. Hazard ten jest zawsze związany z obecnością w formule logicznej, opisującej Q , argumentu z negacją i bez negacji, gdyż właśnie negator jest głównym powodem błędów. Dwa człony funkcji, w których występuje a i \bar{a} można w tablicy Karnaugh'a poznać po dwóch grupach kratek, stykających się ze sobą na linii zmiany wartości a . Na rys. 4-38a linią tą jest brzeg tablicy — linia styczności dwóch zakreslonych grup zer. Hazard oznacza, że już przestał działać element realizujący jedną grupę, a jeszcze nie zaczął działać element realizujący drugą grupę (lub odwrotnie) i dlatego przejście między grupami jest niebezpieczne. Z analizy tej wynika uniwersalna metoda zwalczania hazardu statycznego, gdyż wystarczy na miejsce styczności dwóch grup w tablicy Karnaugh'a nałożyć trzecią grupę, by przejęła ona obowiązki utrzymywania właściwej wartości Q w przejściowym okresie, gdy nie działają dwie poprzednie grupy. Na rys. 4-38a odpowiednią grupę zaznaczono linią przerywaną, a wolna od hazardu funkcja Q' ma postać

$$Q' = (a+b)(Q+\bar{a})(Q+b)$$

Realizacja tej funkcji z elementów NOR (czynnych) jest przedstawiona na rys. 4-38d. Jeśli negacje argumentów są do dyspozycji, bardziej korzystna jest postać

$$Q' = (a+b)(Q+\bar{a}b)$$

o schematach jak na rys. 4-38e (z elementów NOR) i 4-38f (z elementów stykowych). Opisana tu metoda budowania układu sekwencyjnego bez wydzielania elementów pamięci szczególnie dobrze nadaje się do syntezy układów z przełącznikami i elementami pneumatycznymi (z systemów, w których elementy pamięci nie występują jako odrębny blok).

W schematach z rys. 4-38 wykorzystywano funkcję Q' , wyrażoną jako iloczyn sum i otrzymaną z zer tablicy Karnaugh'a. W podobny sposób, łącząc w grupy jedyńki, można uzyskać postać sumy iloczynów

$$Q' = \bar{a}b + aQ$$

Również w tym przypadku utworzone grupy stykają się, więc do realizacji nadaje się wolna od hazardu postać

$$Q' = \bar{a}b + aQ + bQ$$

Poprzednio hazard groził pojawieniem się sygnału $Q = 1$ tam, gdzie powinno być $Q = 0$; obecnie brak zabezpieczenia przed hazardem spowodowałby pojawienie się sygnału $Q = 0$ tam, gdzie powinno być $Q = 1$. Dla odróżnienia pierwszy przypadek nazywany jest „hazardem w zerach”, a drugi — „hazardem w jedynkach”. Zwalczenie jednego z nich automatycznie likwiduje drugi.

Walka z hazardem statycznym przez dołączenie dodatkowych elementów, sprowadza się ogólnie do realizowania funkcji w postaci iloczynu wszystkich implicantów albo sumy wszystkich implikantów, zdarza się jednak, że niektóre przejścia między grupami Karnaugh'a nie są wykorzystywane w tablicy przejść i wtedy zabezpieczać ich nie trzeba.

Inna metoda walki z hazardem polega na takim przyspieszaniu lub opóźnianiu reakcji poszczególnych elementów metodami układowymi, aby warunki powstania hazardu nie wystąpiły. Jest to metoda niekiedy tańsza, ale bardziej kłopotliwa i mniej uniwersalna.

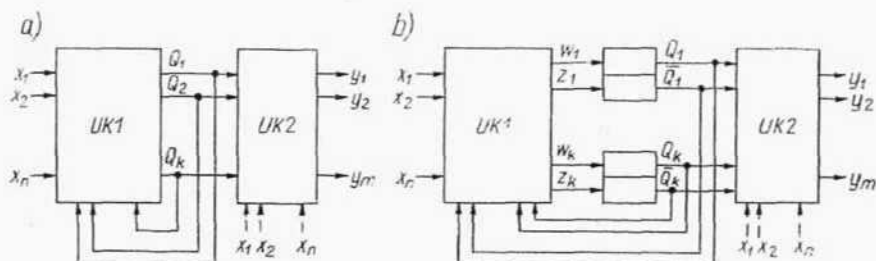
Jeśli sygnał w układzie jest przesyłany trzema lub więcej drogami o różnych opóźnieniach, to może wystąpić *hazard dynamiczny*, polegający na wygenerowaniu sekwencji 0101 zamiast 01, albo 1010 — zamiast 10. Hazard taki może powstać tylko w układach faktoryzowanych o dużej liczbie poziomów, gdy jedna ze zmiennych wchodzi na trzy lub więcej różne poziomy. Jest to przypadek stosunkowo rzadko występujący, a jego złe skutki najprościej można zlikwidować przez zmianę struktury układu (zmniejszenie liczby poziomów).

Hazard statyczny i dynamiczny występuje właściwie w układzie kombinacyjnym i powinien być zwalczany we wszystkich układach kombinacyjnych, w których pojawienie się krótkotrwałych, nie przewidzianych programem, przerw lub impulsów na wyjściu jest szkodliwe. W szczególności jest to szkodliwe, gdy (jak w układach sekwencyjnych asynchronicznych) wyjścia układu kombinacyjnego bezpośrednio oddziałują na elementy pamięci lub podtrzymują swój stan pętlami sprzężenia zwrotnego.

Jeśli w układzie sekwencyjnym z więcej niż jednym obwodem pamięciowym występują elementy o bardzo różnych czasach reakcji, istnieje niebezpieczeństwo powstania *hazardu podstawowego*. Polega on na tym, że jedne obwody pamięciowe — uzależnione od jakiegoś x — działają szybciej niż inne, uzależnione od opóźnionego sygnału \bar{x} . W rezultacie przy zmianie wartości x , powstaje inny stan wewnętrzny niż wynika to z tablicy przejść. Jeśli stan ten ma warunki stabilizacji, występuje trwały błąd w pracy układu. Hazard podstawowy może powstać, gdy czas reakcji negatora jest kilkakrotnie dłuższy niż czas reakcji innych elementów układu. W nowoczesnych układach negator ma identyczną budowę jak inne elementy (niekiedy nawet jest prostszy i szybszy), więc problem walki z hazardem podstawowym nie występuje.

W schemacie z rys. 4-38e można wydzielić dwa elementy NOR objęte pętlą sprzężenia, gdyż już wcześniej zostały stwierdzone interesujące właściwości takiego układu (rys. 2-15), polegające na możliwości przebywania w dwóch różnych stanach stabilnych, zmienianych sygnałami oznaczonymi przez w (wpisywanie) i z (zerowanie). Wprowadzenie symbolu takiego przerzutnika wz zmienia schemat układu z rys. 4-38e do postaci z rys. 4-38g. Różnica w oznaczeniach nie jest istotna, ale nowy rysunek sugeruje możliwość wyznaczania wzbudzeń w i z zamiast poprzednio tworzonej funkcji Q' . W ten sposób dochodzi się do drugiej metody realizacji pamięci w układach asynchronicznych statycznych, polegającej na wydzieleniu przerzutników wz . Ogólny schemat blokowy układu sekwencyjnego z rys. 4-1 w przypadku układów synchronicznych statycznych może więc przybrać jedną z dwóch postaci, przedstawionych na rys. 4-39. Rozwiązanie bez wydzielania bloku przerzutników jest bardziej ogólne i umożliwia uzyskanie większej liczby wariantów (niekiedy prostszych niż przy zastosowaniu przerzutników), natomiast roz-

wiązanie z blokiem pamięci upraszcza wyznaczanie wzbudzeń i umożliwia łatwą realizację zarówno sygnałów Q jak i \bar{Q} . Ze względu na te cechy, metodę wydzielania przerzutników stosuje się zwykle tylko w tych przypadkach, gdy niezbędne są zarówno sygnały Q jak i \bar{Q} . Gdy dla realizacji funkcji y są potrzebne tylko negacje Q , można je łatwo uzyskać i w pierwszej metodzie, zmieniając kod w ramach tych samych podziałów.



Rys. 4-39. Struktury układów asynchronicznych statycznych: (a) bez bloku pamięci; b) z blokiem pamięci

Wyjściami przerzutnika w - z są sygnały Q i \bar{Q} tylko w tym przypadku, gdy obydwa wejścia nie są wzbudzone równocześnie, tzn. gdy $wz = 0$. Warunek ten sprawia, że działanie przerzutnika wz staje się bardzo podobne do działania elementu SR , a do wyznaczania wzbudzeń w i z można wykorzystać odpowiednie kolumny SR z tablic 4-1 i 4-2. Na przykład tablica przejść układu bramkowania generatora z rys. 4-38a może być za pomocą tabl. 4-1 przekształcona w tablicę wzbudzeń z rys. 4-40a, z której otrzymuje się

$$w = \bar{a}b \quad z = \bar{a}\bar{b}$$

Odpowiada to schematowi z rys. 4-38g. Podobnie jak w układach synchronicznych, przy bardziej złożonych tablicach jest korzystne stosowanie uniwersalnych tablic wzbudzeń powstałych przez podkreślenie tych sygnałów w tablicy przejść, które ulegają zmianie. W ten sposób z rys. 4-38a powstaje rys. 4-40b. Sygnał w musi mieć wartość 1 w tych kratkach tablicy, w których występuje 1 i musi mieć wartość 0 tam gdzie występuje jakiegokolwiek 0; kreski i 1 oznaczają dowolną wartość w . Sygnał z musi mieć wartość 1 w tych kratkach, gdzie występuje 0 oraz musi mieć wartość 0 tam gdzie występuje jakiegokolwiek 1; kreski i 0 oznaczają

dowolną wartość z (tabl. 4-2). Na podstawie tych reguł z rys. 4-40 wyznacza się wprost formuły dla w i z , a i w bardziej złożonych tablicach określenie wzbudzeń nie jest trudne.

a)

$Q \backslash ab$	00	01	11	10
0	0,-	1,0	0,-	0,-
1	0,1	-,0	-,0	-,0

w, z

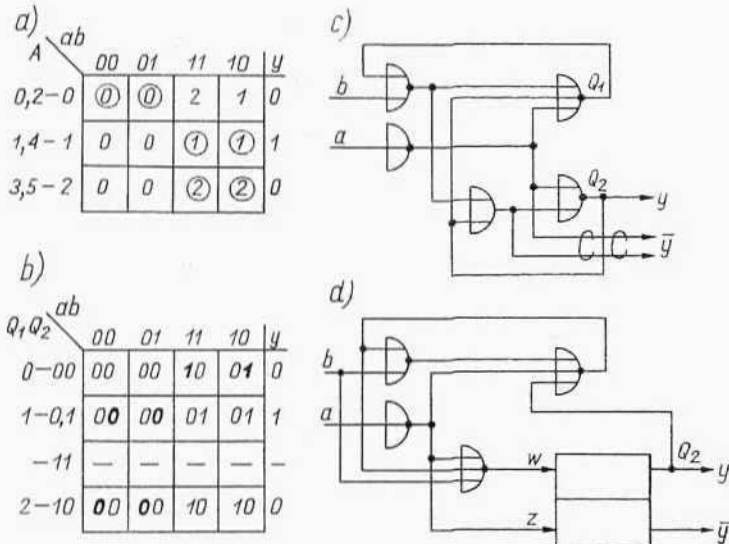
b)

$Q \backslash ab$	00	01	11	10
0	0	1	0	0
1	0	1	1	1

Q'

Rys. 4-40. Tablice wzbudzeń układu bramkowania generatora

Próba realizacji układu bramkowania generatora wg tablicy z rys. 4-32c prowadzi do rozwiązania identycznego jak w wersji Mealy'ego, natomiast odmienny układ Moore'a można uzyskać przez inne łączenie stanów przy minimalizacji (rys. 4-41a). Po zakodowaniu otrzymuje się tablicę jak na rys. 4-41b. Jeśli układ ma być budowany z elementów



Rys. 4-41. Projekt układu bramkowania generatora (wersja Moore'a): a) minimalna tablica przejść; b) zakodowana tablica przejść; c) układ z wydzielonym przerzutnikiem

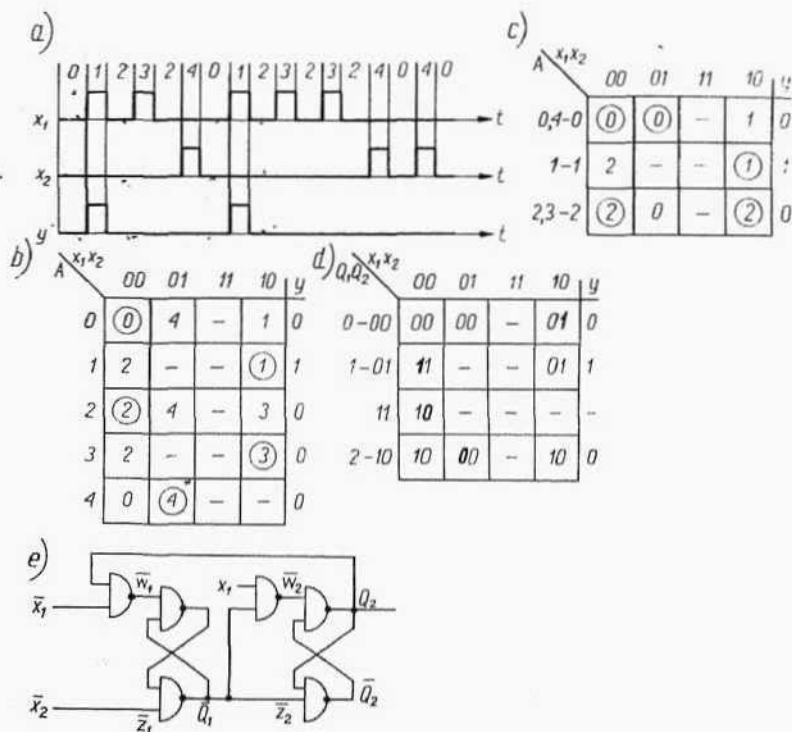
NOR lub NAND, do jego syntezy można wykorzystać metody z p. 3.3:

$$F^1(Q_1) = \begin{pmatrix} 1 & 1 & - \\ - & 0 & 1 & 1 \end{pmatrix} \quad F^1(Q_2) = \begin{pmatrix} - & 1 & 1 & - \\ 0 & - & 1 & 0 \end{pmatrix}$$

$$F^0(Q_1) = \begin{pmatrix} - & 0 & - \\ - & 1 & - \\ 0 & - & 0 \end{pmatrix} \quad F^0(Q_2) = \begin{pmatrix} 1 & - & - \\ - & 0 & - \\ - & 0 & - & 1 \end{pmatrix} = \begin{pmatrix} 1 & 0 & - \\ - & 0 & - \\ - & 0 & - \end{pmatrix} = \begin{pmatrix} - & 0 & - & 0 \\ - & 0 & - & - \end{pmatrix}$$

$$(0 \text{---} 0 | -)$$

przy czym zmienne wypisywane są w kolejności (Q_1, Q_2, a, b) . W pierwszym przekształceniu $F^0(Q_2)$ zrezygnowano z postaci minimalnej, aby



Rys. 4-42. Projekt układu statycznego: a) wykres czasowy; b) tablica pierwotna; c) tablica minimalna; d) tablica zakodowana; e) schemat układu z elementów NAND

upodobnić do siebie rdzenie. Prowadzi to do minimalnego schematu przedstawionego na rys. 4-41c. Negację sygnału wyjściowego można uzyskać w postaci wiązki (jak na tym rysunku), dodając negator na wyjściu układu, albo realizując Q_2 za pomocą przerzutnika. Z tablicy na rys. 4-41b uzyskuje się wówczas

$$w_2 = \bar{Q}_1 \bar{a}b; \quad z_2 = \bar{a}$$

a odpowiedni schemat zawiera jeden element więcej (rys. 4-41d) niż w przypadku wiązki.

W układach kombinacyjnych, realizujących sygnały w i z , również mogą powstać hazardy, ale są one znacznie mniej prawdopodobne, gdyż funkcje w i z są dużo prostsze niż Q' , a ponadto hazard w układzie wzbudzenia rzadko jest niebezpieczny, gdyż krótkotrwała przerwa w działaniu wzbudzenia zwykle nie może już zmienić stanu przerzutnika.

Niezależnie od tego czy przerzutnik jest zbudowany z elementów NOR, czy NAND, należy w i z wyznaczać w postaci sum iloczynów, gdyż sumy realizowane są w samym przerzutniku (rys. 2-15e i 2-16c). W przypadku elementów NAND zazwyczaj prościej jest określać \bar{w} i \bar{z} . Na przykład gdy trzeba zaprojektować układ wybierający pierwszy impuls x_1 z serii impulsów dzielonych impulsem x_2 (rys. 4-42a), uzyskuje się tablicę pierwotną z rys. 4-42b. Założenie, że impulsy wejściowe nie mogą występować w tym samym czasie, bardzo upraszcza tablicę. Łatwo można zauważyć zgodność stanów 0 i 4 oraz 2 i 3, co prowadzi do tablicy z rys. 4-42c, dla której:

$$\pi(X_0) = \{\bar{0}; \bar{1}, \bar{2}\} \quad \pi(X_2) = \{\bar{0}, \bar{1}; \bar{2}\} \quad \pi(y) = \{\bar{0}, \bar{2}; \bar{1}\}$$

Przyjęcie do kodowania podziałów wewnętrznych skomplikuje funkcję wyjściową. Ponieważ tablica przejść jest silnie nieokreślona, można założyć, że funkcje przejść będą proste przy każdym kodzie, a wyrugowanie wyścigów krytycznych nie sprawi trudności. W takiej sytuacji każdy kod daje podobne rezultaty. Kodując tak jak na rys. 4-42d otrzymuje się

$$Q'_1 = Q_2 \bar{x}_1 + Q_1 \bar{x}_2 \quad Q'_2 = \bar{Q}_1 Q_2 + \bar{Q}_1 x_1 \quad y = Q_2$$

albo

$$\begin{aligned} w_1 &= Q_2 \bar{x}_1 & w_2 &= \bar{Q}_1 x_1 \\ z_1 &= x_2 & z_2 &= Q_1 \end{aligned}$$

Obydwie metody postępowania dają w tym przypadku takie same wyniki. Na schemacie z elementów NAND (rys. 4-42e) zachowano typowy rysunek przerzutnika i zanegowane sygnały wzbudzeń.

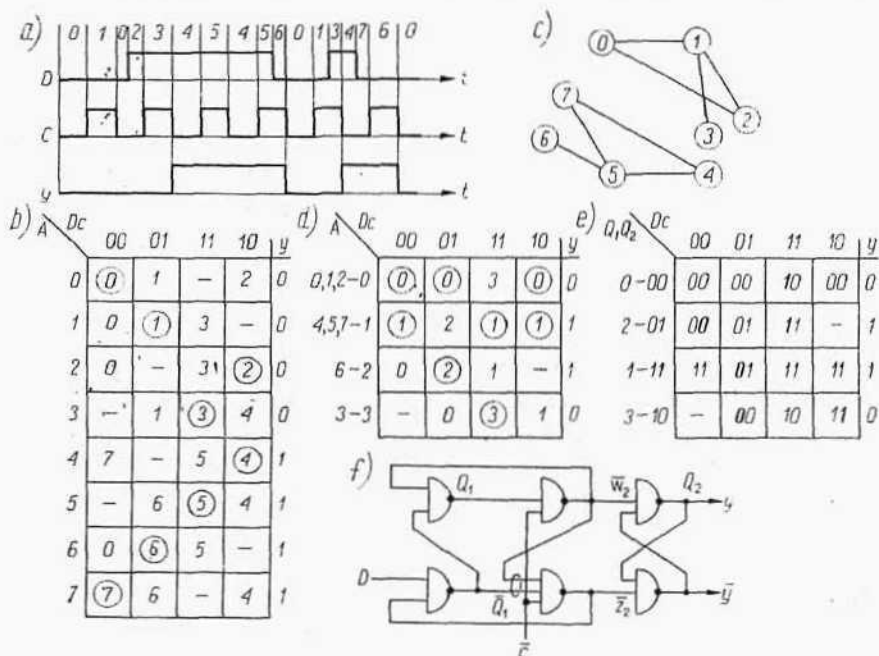
Przy zmianie formuł Q' na w , z (lub odwrotnie) niekiedy mogą być użyteczne podstawowe równania przerzutników (p. 2.3.2):

z elementów NOR — $Q' = (Q + w)\bar{z}$ oraz

z elementów NAND — $Q' = Q\bar{z} + w$

Gdy $wz = 0$ równania te są równoważne.

Układy asynchroniczne statyczne projektuje się z najbardziej elementarnych składników układu logicznego, jakimi są elementy kombinacyjne. Inne stosowane poprzednio elementy, np. przerzutniki, mogą być uważane za układy złożone z tych elementarnych składników. Na przykład przerzutnik synchronizowany typu D może być uważany za układ asynchroniczny o dwóch wejściach D i c i jednym wyjściu y (pożądana negacja \bar{y}). Z wykresu czasowego (rys. 4-43a) wyznacza się tablicę



Rys. 4-43. Projektowanie realizacji statycznej przerzutnika typu D

pierwotną (b). Stanów pseudorównoważnych nie ma, więc na wykresie skracania każdy stan występuje oddzielnie. W wierszach 3 i 6 tablicy pierwotnej występują stany nieokreślone, więc nie łącząc tych stanów z innymi można spodziewać się pewnego uproszczenia układu. Z minimalnej tablicy Moore'a (rys. 4-43d) otrzymuje się:

$$\begin{aligned}\pi(X_0) &= \{\overline{0}, 2; \overline{1}; (3)\} & \pi(X_1) &= \pi(X_3) = \pi(y) = \{\overline{0}, 3; \overline{1}, 2\} = \pi_1 \\ \pi(X_2) &= \{\overline{0}; \overline{1}, 3; (2)\}, & \text{więc} & \quad \tau_2 = \{\overline{0}, 2; \overline{1}, 3\}\end{aligned}$$

Z zakodowanej tablicy można uzyskać łatwo

$$\begin{aligned}w_1 &= Dc & w_2 &= Q_1\bar{c} \\ z_1 &= \bar{D}c & z_2 &= \bar{Q}_1\bar{c}\end{aligned}$$

co odpowiada strukturze stosowanej w przerzutnikach Master-Slave (rys. 2-23a). Jeśli układ ma być budowany z elementów NAND, to — decydując się na realizację Q_2 za pomocą przerzutnika, można z zakodowanej tablicy otrzymać:

$$\begin{aligned}F^1(w_2) &= (1--0) & F^1(z_2) &= (0--0) \\ F^1(Q_1') &= \begin{pmatrix} 1--0 \\ 1-1- \\ --11 \end{pmatrix} = \begin{pmatrix} 1--0 & | & - \\ -1- & | & 0 \end{pmatrix} & F^0(Q_1') &= \begin{pmatrix} 0--0 \\ --01 \end{pmatrix} \\ & & & \hline & & & (0--0|-) \end{aligned}$$

przy czym zmienne są wypisywane w kolejności: Q_1, Q_2, D, c .

$F^1(w_2)$ oraz $F^1(z_2)$, realizowane jednym elementem NAND, wyznaczają \bar{w}_2 oraz \bar{z}_2 , czyli potrzebne wzbudzenia przerzutnika. Ponieważ w obydwu tych wyrażeniach występuje \bar{c} , dla ujednolicenia usunięto c z $F^1(Q_1')$. Po zastosowaniu wiązki zamiast pojedynczego sygnału \bar{Q}_1 , otrzymuje się schemat (rys. 4-43f) podobny do schematu układu z elementów NOR (rys. 2-22a). W taki sam sposób można zaprojektować rozwiązania statyczne innych typowych przerzutników.

4.4. UKŁADY ASYNCHRONICZNE DYNAMICZNE

4.4.1. PODSTAWOWE PROBLEMY SYNTEZY UKŁADÓW DYNAMICZNYCH

Charakterystyczną cechą układów asynchronicznych dynamicznych jest możliwość wykorzystania dynamicznych stanów wejść, tzn. informacji o charakterze zmian sygnałów wejściowych. Układ o n sygnałach