

ale fakt że jedna z liczb jest podana w postaci równoległej umożliwia zastąpienie licznika rewersyjnego licznikiem odejmującym,

Gdy liczba  $X_1$  jest ustawiana za pomocą przełączników lub przycisków, równość liczb

$$X_1 = L_T(x_2)$$

może być sygnalizowana w układzie z rys. 7-20c (przystosowanym do kodu dwójkowo-dziesiętnego). Jest to wprawdzie układ bardziej złożony niż poprzedni, ale za to może być wykorzystany do wyznaczenia momentów, w których liczba  $X_2$  jest równa kilku innym liczbom typu  $X_1$ , czyli

$$y_i = 1 \quad \text{gdy} \quad X_{1i} = L_T(x_2) \quad i = 1, 2, \dots$$

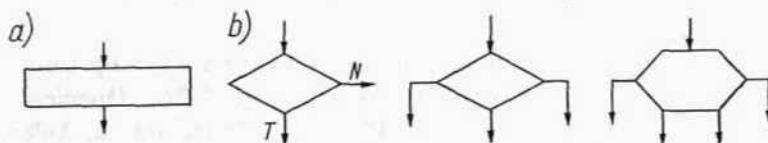
3.  $X_1^{2R} \geq X_2^{2R}$ ,  $X_1^{2S} \geq X_2^{2S}$ . Do porównania liczb w tych przypadkach służą komparatory.

### 7.3. STEROWANIE

#### 7.3.1. SIEĆ DZIAŁAŃ

Pracę złożonego układu cyfrowego wygodnie jest opisywać za pomocą tzw. *sieci działań* (*schematów czynności*), budowanych z dwóch podstawowych elementów:

— *klatek operacyjnych* (rys. 7-21a), opisujących jedną lub kilka równocześnie lub w dowolnej kolejności wykonywanych czynności, zwykle w formie polecenia (załącz, wykonaj, drukuj itp.),

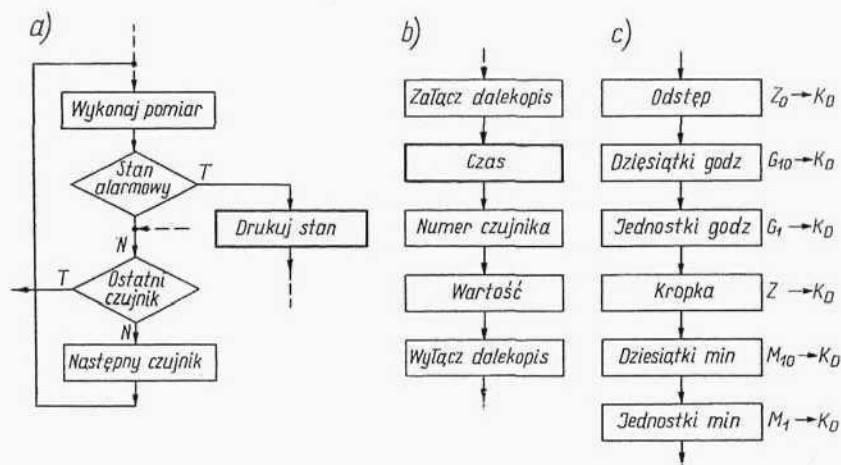


Rys. 7-21. Elementy sieci działań: a) klatka operacyjna; b) klatka warunkowa

— *klatek warunkowych* (rys. 7-21b), opisujących warunki, jakie są stawiane przy przejściu do następnej czynności, zazwyczaj w postaci pytania (czy silnik jest załączony, czy licznik jest wyzerowany itp.). Jeśli odpowiedź może być typu „tak — nie”, klatka warunkowa ma dwa wyjścia ( $T - N$ ), jeśli możliwych odpowiedzi jest więcej, wyjść jest również więcej (np. określenie ruchu: w górę — w dół — stop).

Dla uproszczenia zapisu można wprowadzić dodatkowe klatki sieci działań, np. dla początku i końca ciągu czynności, dla odczekania przez czas  $T$  itd., ale klatki operacyjna i warunkowa są elementami podstawowymi.

Sieć działań może dotyczyć różnych poziomów pracy urządzenia i — odpowiednio — operować operacjami więcej lub mniej złożonymi. Na przykład urządzenie, które automatycznie przeprowadza pomiary za pomocą kilkunastu czujników, a w przypadku stanu alarmowego



Rys. 7-22. Przykład sieci działań o kilku poziomach

zmierzonej wartości drukuje odpowiednie dane, może być opisane siecią działań, której fragment przedstawiono na rys. 7-22a. Operacje typu DRUKUJ STAN, WEŹ NASTĘPNY CZUJNIK itd. są kolejnymi stanami urządzenia, a przejściami między tymi stanami dyrygują warunki: CZY STAN ALARMOWY?, CZY OSTATNI CZUJNIK itp. Każdą z tak ogólnych operacji można rozbić na bardziej dokładne czynności; np. DRUKUJ STAN, w przypadku dalekopisu na sieć jak na rys. 7-22b (w opisach klatek pominięto słowo DRUKUJ). Ale i te operacje nie są jeszcze przetłumaczalne na sygnały sterujące współpracę odpowiednich bloków, gdyż zawierają sekwencję bardziej drobnych czynności (np. dla DRUKUJ CZAS — sieć z rys. 7-22c). Te operacje mają cechy działań

elementarnych, gdyż wynikają z nich bezpośrednio sygnały sterujące bloki. Na przykład operacja DRUKUJ DZIESIĄTKI GODZIN wymaga przesłania stanu odpowiednich wyjść zegara ( $G_{10}$ ) na wejścia konwertera kodu dalekopisowego ( $K_D$ ), co sprowadza się do podania impulsu na odpowiednie bramki. Wprawdzie i druk cyfry w przypadku dalekopisu jest sekwencją znaków binarnych, ale ze względu na jednorodny charakter tych sekwencji, mogą one być uważane za jedną czynność elementarną. Tego typu operacje podstawowe, związane bezpośrednio z wytworzeniem sygnałów sterujących bloki są nazywane *mikrooperacjami*, a sieć działań wyrażona mikrooperacjami opisuje *mikroprogram* działania urządzenia.

Poszczególne etapy tworzenia sieci działań — od programu ogólnego do mikroprogramu — są bardzo pomocne przy określaniu struktury urządzenia (jeśli nie jest ona znana wcześniej). Na przykład z rys. 7-22a wynika, że odpowiedni fragment urządzenia musi zawierać zespoły funkcjonalne:

- pomiaru i badania przekroczeń wartości,
- drukowania,
- komutacji punktów pomiarowych.

Z rys. 7-22b wynika, że zespół drukowania ma 4 podstawowe takty pracy:

- drukowanie czasu,
- drukowanie numeru czujnika,
- drukowanie wartości mierzonej,
- wyłączonej.

Z rys. 7-22c wynika, że drukowanie czasu wymaga sześciu taktów o odpowiednim czasie trwania. Jeśli drukowanie numeru i drukowanie wartości można też zawrzeć w 6-taktowych cyklach, to całe drukowanie sprowadza się do wykonania trzech cykli po 6 taktów.

Oczywiście tworzenie sieci działań dla różnych poziomów nie zawsze jest potrzebne; często z sieci podstawowej wynikają mikrooperacje, czasem łatwiej utworzyć sieć mikrooperacji niż inną itd.

W sieci opisującej mikroprogram najniższego poziomu zazwyczaj nie stosuje się rozkazów słownych WPISZ, DODAJ itp., lecz umowne symbole, określające zmiany stanu różnych bloków z pamięcią. Mogą to być na przykład:

- 1)  $R_1 := L_2$  — wpisz do  $R_1$  zawartość  $L_2$  (równolegle);
- 2)  $R_1 := 0$  — wyzeruj  $R_2$ ;
- 3)  $L_1 := L_1 + 1$  — dodaj 1 do wartości  $L_1$ ;
- 4)  $RP := l_1(RP)$  — przesun o 1 bit w lewo zawartość  $RP$ ;  
 $RP := p_2(RP)$  — przesun o 2 bity w prawo zawartość  $RP$   
 (zazwyczaj przesunięcie w lewo oznacza powiększenie liczby);
- 5)  $A := A + R$  — dodaj do zawartości akumulatora zawartość  $R$ .

Każdy taki rozkaz jest realizowany przez odpowiednie wartości sygnałów  $z, w, o, c$  bloków sekwencyjnych lub sygnałów  $b$  bloków kombinacyjnych, a więc właściwe napisanie mikroprogramu określa działanie bloku sterowania.

Postać sieci działań opisującej mikroprogram wyznacza w pewnym stopniu rodzaj pracy układu sterowania. Jeśli każde dwie klatki operacyjne sieci są przedzielone klatką warunkową, to układ sterowania będzie *asynchroniczny*. Jeśli warunek ten nie jest spełniony, to układ sterowania będzie *synchroniczny*, gdyż dla przejścia od jednej klatki operacyjnej do drugiej jest potrzebny wewnętrzny sygnał układu. W sieci działań układu synchronicznego oczywiście też mogą występować klatki warunkowe; przerywają one synchroniczne wykonywanie operacji aż do spełnienia odpowiedniego warunku, lub wybierają jedną z kilku możliwości dalszego wykonywania operacji.

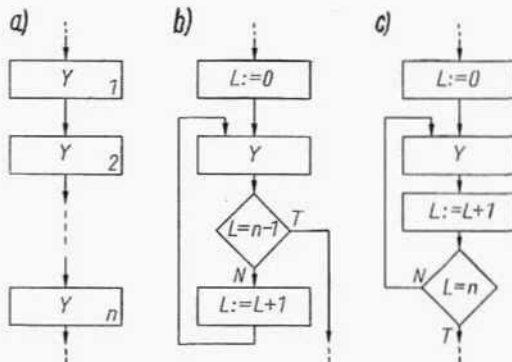
Niekiedy sieć działań układu synchronicznego można zmienić w sieć układu asynchronicznego przez układowe wygenerowanie potrzebnych warunków, np. sygnału zakończenia dodawania, sygnału wpisania do rejestru itp. Jest to celowe wówczas, gdy ważny jest czas wykonywania działań, gdyż układ asynchroniczny działa szybciej od układu synchronicznego.

Jeśli warunki działania urządzenia, opisywane siecią działań, są formułowane w sposób logiczny, to sieć ma postać jednoznaczną i jej realizacja nie prowadzi do sprzeczności czy błędów. Są jednak pewne możliwości budowania sieci, na które warto zwrócić uwagę.

Zazwyczaj linia, wychodząca z dowolnej klatki sieci, wchodzi do jednej tylko klatki, wyznaczając następstwo działań. Może jednak okazać się celowe wykonywanie pewnych czynności w układzie równocześnie, a wówczas w sieci działań wystąpią dwie gałęzie równoległe: rozpoczynające się w jednym miejscu. Sytuacja taka jest dopuszczalna pod warun-

kiem, że zabezpieczone będzie prawidłowe ponowne połączenie obu gałęzi, a więc gdy liczba taktów pracy (bloków operacyjnych w układzie synchronicznym) będzie w tych gałęziach taka sama, lub gdy specjalne warunki wymuszą oczekiwanie w jednej gałęzi na zakończenie działań w drugiej.

Jeśli czynność wymuszona klatką operacyjną kończy się przed przejściem do innej klatki, do jej opisu używa się nazw typu ZAMKNIJ ZAWÓR, DRUKUJ CZAS, albo ZAMYKANIE ZAWORU, DRUKOWANIE CZASU itp. Jeśli w czasie trwania czynności, zainicjowanej klatką operacyjną, można wykonać kilka dalszych czynności, opis powinien to uwzględniać, formą np. POCZĄTEK ZAMYKANIA ZAWORU, START DRUKOWANIA CZASU, a w układzie odpowiedni przerzutnik powinien podtrzymywać realizację tej czynności przez kilka taktów, aż do rozkazów ze słowami KONIEC czy STOP. Pewna różnica występuje tylko przy odmierzaniu czasu, gdzie wystarczy operacja START (POCZĄTEK) oraz warunek sprawdzający, czy czas już minął.

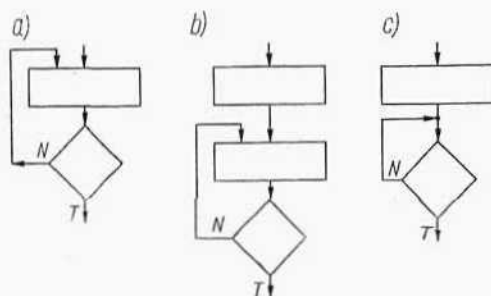


Rys. 7-23. Odcinki równoważne sieci działań z powtarzającymi się fragmentami

Jeśli w sieci działań pewne fragmenty (pojedyncze klatki lub ich zespoły) powtarzają się, można tych powtórzeń uniknąć, wprowadzając specjalny indeks i określający go licznik, zliczający powtórzenia. Na przykład, zamiast sieci z rys. 7-23a można narysować sieć jak na rys. 7-23b lub 7-23c. Jeśli w sieci z rys. 7-23b wstępnie wpisze się do licznika

$n-1$  ( $L := n-1$ ), a licznik będzie zmniejszał swą wartość ( $L := L-1$ ), to warunek przybierze postać  $L = 0$ . Podobnie można zmienić drugą sieć. Indeks określony licznikiem może być również wykorzystany do pewnych zmian operacji w kolejnych obrotach pętli, podstawiania coraz to innych argumentów realizowanego  $n$  razy działania itp.

Pętle obejmujące jedną tylko klatkę operacyjną (rys. 7-24a) mogą niekiedy mieć różne znaczenia. W układach asynchronicznych oznacza



Rys. 7-24. Różne postaci pętli

to po prostu oczekiwanie na spełnienie warunku, ale w układach synchronicznych może oznaczać oczekiwanie lub wielokrotne wykonywanie tej samej czynności. Aby uniknąć dwuznaczności, należy w przypadku oczekiwania rysować pętlę jak na rys. 7-24c (co jednak narzuca wersję układu Mealy'ego — p. 7.3.2) lub wprowadzić dodatkową klatkę oczekiwania (rys. 7-24b).

### 7.3.2. TABLICE PRZEJŚĆ

Wykonanie rozkazu, określonego przez klatkę operacyjną, odbywa się przez wysłanie z bloku sterowania odpowiedniego sygnału  $y$  do arytmometru (lub pamięci). Sygnały te blok sterowania określa na podstawie stanu, opisanego klatkami warunków, a więc warunki są wejściami  $x$  bloku sterowania. Ponadto sygnały wyjściowe tego bloku zależą od stanu, w jakim układ się znajduje, co jest określone miejscem w sieci działań. Jak z tego wynika, dla układu sekwencyjnego jakim jest blok sterowania, stan wejść  $X$  jest określony przez klatki warunków, stan wyjść  $Y$  — przez

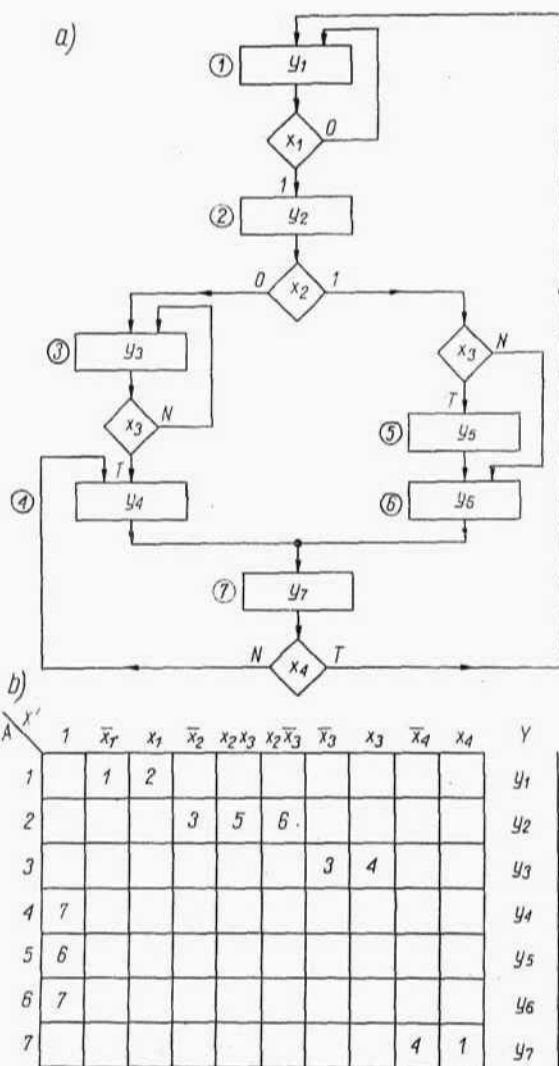
klatki operacyjne, a stan wewnętrzny  $A$  — przez konfigurację sieci działań.

Jeśli pojęcie stanu wewnętrznego przypisze się klatkom operacyjnym, to sygnały  $y$  będą związane ze stanami  $A$ , więc układ sterowania będzie automatem Moore'a. Przykład sieci działań z wpisanymi symbolami  $x$ ,  $y$  oraz  $i$  (zamiast  $A_i$ ) podano na rys. 7-25a. W odróżnieniu od rozważanych wyżej grafów układu sekwencyjnego, sieć taka określa przejścia ze stanu do stanu tylko pod wpływem niektórych sygnałów wejściowych, aktualnie potrzebnych. Narysowanie na tej podstawie pełnej tablicy przejść wymagałoby szczegółowego rozważania przypadków w sieci nie uwzględnianych (np. zachowanie się układu, gdy w stanie  $A_1$  pojawi się sygnał  $x_3 = 1$ ), gdyż nie zawsze działanie układu przy wymuszeniach nie uwidocznionych jest obojętne (tzn. odpowiada kreskom wewnątrz tablicy przejść). Inną trudnością przy budowie pełnej tablicy jest duża zwykle liczba sygnałów  $x$ , a więc i stanów  $X$ , co prowadzi do ogromnych tablic. W takiej sytuacji jest celowe stosowanie uproszczonych tablic przejść, uwzględniających tylko te warunki zmiany stanu wewnętrznego, które wynikają z sieci działań. Dla sieci z rys. 7-25a będzie to tablica z rys. 7-25b. Kolumny tablicy są tu opisane taką funkcją  $X'$ , że przejście z jednego stanu wewnętrznego do drugiego następuje wówczas, gdy  $X' = 1$ . Dla kolumny oznaczonej przez 1 przejście następuje więc zawsze, bez dodatkowych warunków. Funkcja  $X'$  może mieć postać sumy iloczynów.

Ponieważ o pustych polach wewnątrz tablicy nic nie wiadomo, za stany zgodne można tu uznać tylko takie, które mają identyczne stany następne i stany wyjść. Dalsze zmniejszenie liczby stanów wewnętrznych tablicy, bez szczegółowego badania stanów następnych, jest możliwe po wprowadzeniu dodatkowej informacji o tych sygnałach  $x$ , które w stanie  $A_i$  dają się określić i nie ulegają zmianie. Na przykład, jeśli po klatce operacyjnej ( $A_i$ ) ZAMKNIJ ZAWÓR następuje klatka warunkowa ( $x_j$ ) CZY ZAWÓR ZAMKNIĘTY?, to zazwyczaj na podstawie poprzednich klatek można stwierdzić, że w  $A_i$  jest zawsze  $\bar{x}_j$ , natomiast w następnym stanie jest zawsze  $x_j$ <sup>1)</sup>.

Dla każdego stanu  $A_i$  funkcja określająca stan niezmiennych warunków  $x$  będzie oznaczana przez  $W_i$ . Postać funkcji  $W$  nie zawsze wy-

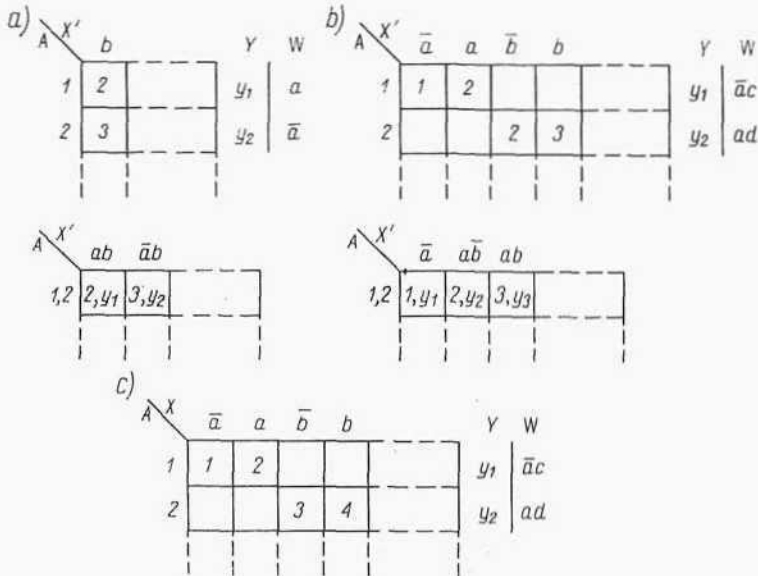
<sup>1)</sup> „Jest  $x_j$ ” oznacza tu  $x_j = 1$ .



Rys. 7-25. Przykład sieci działań (a) i opisująca ją tablica przejść układu Moore'a



nika z sieci działań, lecz musi być wyznaczona na podstawie przewidywanego zachowania się układu. Na przykład, jeśli w sieci z rys. 7-25a, stan  $A_1$  jest stanem oczekiwania, a  $x_1$  to START pochodzący od niestabilnego przycisku, to można napisać, że czynnikiem  $W_1$  jest  $\bar{x}_1$ , ale  $x_1$  nie może być czynnikiem  $W_2$ , gdyż w czasie trwania stanu  $A_2$  przycisk startowy może być zwolniony, co nie powinno mieć wpływu na pracę układu.

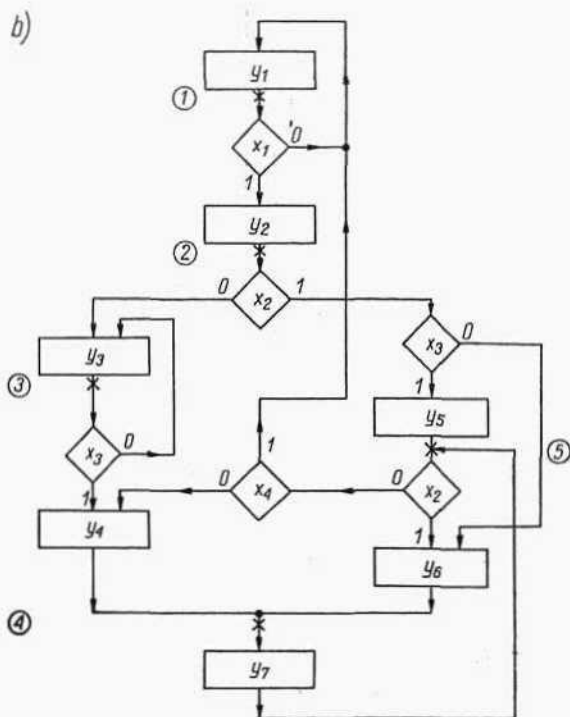


Rys. 7-26. Przykłady minimalizacji liczby stanów (a,b) i przypadek nie podlegający uproszczeniu (c)

Jeśli  $W_i = x$  oraz  $W_j = \bar{x}$ , to istnieją wstępne warunki połączenia stanów  $A_i$  oraz  $A_j$  w jeden stan, gdyż dochodzi się do nich z dwóch różnych stanów, przygotowujących warunki  $x$  oraz  $\bar{x}$ , a zatem, uzależniając dalsze działanie wspólnego stanu  $A_{ij}$  od  $x$ , można będzie zrealizować wszystkie przewidziane stany tablicą przejścia. Przykład ilustrujący to rozumowanie jest przedstawiony na rys. 7-26a. Ponieważ  $W_1 = a$ , więc przejście z  $A_1$  pod wpływem  $X' = \bar{a}b$  jest nieokreślone, podobnie jak przejście z  $A_2$  pod wpływem  $X' = ab$ ; stany  $A_1$  i  $A_2$  można więc połączyć, rozbudowując funkcję  $X'$ . Ogólnie, jeśli  $W_i \cdot W_j = 0$ , to stany  $A_i$  i  $A_j$

a)

A	X'	1	$\bar{x}_1$	$x_1$	$\bar{x}_2$	$x_2 x_3$	$x_2 \bar{x}_3$	$\bar{x}_3$	$x_3$	$x_2$	$\bar{x}_2 \bar{x}_4$	$\bar{x}_2 x_4$
1			1, $y_1$	2, $y_2$								
2					3, $y_3$	5, $y_5$	6, $y_6$					
3								3, $y_3$	4, $y_4$			
4,6		7, $y_7$										
5,7										6, $y_6$	4, $y_4$	1, $y_1$



Rys. 7-27. Tablica Mealy'ego dla przykładu z rys. 7-25 (a) i odpowiadająca jej sieć działań (b)

można połączyć, mnożąc  $X'$ , dla których  $A_i$  ma stany następne, przez  $W_i$  (lub tylko tę literę z  $W_i$ , która sprawia, że  $W_i \cdot W_j = 0$ ) oraz mnożąc  $X'$ , dla których  $A_j$  ma stany następne, przez  $W_j$  (lub odpowiednią literę). Dodatkowym warunkiem jest, aby po połączeniu stanów zachowane były wszystkie poprzednie związki z innymi stanami. Na przykład połączenie stanów w tablicy z rys. 7-26b ruguje jedną kolumnę tablicy, ale zachowuje związek ze stanem  $A_2$ , natomiast próba połączenia stanów w tablicy z rys. 7-26c prowadzi do usunięcia jedyne go przejścia do stanu  $A_2$ , wierszy tych nie można więc zastąpić jednym.

Jeśli dwa stany (układu Moore'a) o różnych wyjściach sprowadza się do jednego stanu, to prowadzi to oczywiście do układu Mealy'ego. Sposób wpisywania stanów wyjść wynika z rys. 7-26. Blokom operacyjnym może odpowiadać nie jeden, lecz kilka sygnałów wyjściowych  $y$ .

Zastosowanie omówionych reguł minimalizacji tablicy do rys. 7-25b umożliwia uzyskanie tablicy układu Mealy'ego z rys. 7-27a. Na tej podstawie można odtworzyć zmodyfikowaną sieć działań (rys. 7-27b), w której stanom wewnętrznym będą odpowiadać już nie klatki operacyjne, lecz punkty charakterystyczne sieci za klatką operacyjną. Rysunki 7-25a i 7-27b są sobie równoważne tylko pod warunkiem spełnienia funkcji  $W$  z rys. 7-25b. Układ Mealy'ego skraca niektóre sygnały  $y$ .

### 7.3.3. STRUKTURY UKŁADU STEROWANIA

Stany wewnętrzne układu sterowania, wyznaczone na podstawie sieci działań lub tablic przejść, można następnie zakodować metodami przedstawionymi wyżej. Ponieważ jednak prostota strukturalna układu sterowania jest szczególnie ważna, często rezygnuje się z kodu optymalnego, stosując kod narzucony wyjściami (jeśli to możliwe), a zwłaszcza najprostszy kod „1 z  $k$ ”. Dalej będą rozważane przypadki układów sterowania z tym kodem, aczkolwiek w wielu przypadkach inne kodowanie może przynieść znaczną oszczędność elementów.

Zakodować a tablica przejść układu sterowania opisuje układ sekwencyjny o stanach  $X$ ,  $Y$ ,  $A$ , który może być dalej projektowany znanymi metodami. Specyficzny charakter układu sterowania sprawia jednak, że zazwyczaj, zamiast wyznaczania odrębnych funkcji wzbudzeń dla poszczególnych elementów pamięci, przyjmuje się blok pamięci w postaci re-

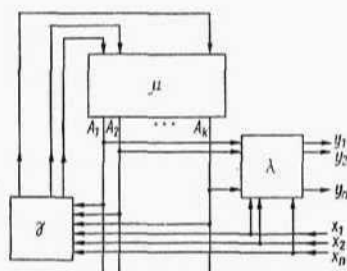
jestru lub licznika, a wzbudzenia realizuje się w bloku dekodera, co ułatwia wprowadzanie ewentualnych zmian.

Znaną, ogólną strukturę układu sekwencyjnego (rys. 4-1) wygodnie będzie w przypadku układu sterowania przedstawić w postaci z rys. 7-28. Układy realizujące funkcje  $\gamma$  i  $\lambda$  to proste układy kombinacyjne, złożone zazwyczaj z iloczynów lub sum iloczynów zmiennych, natomiast realizacja funkcji  $\mu$  jest bardziej złożona i różnorodna.

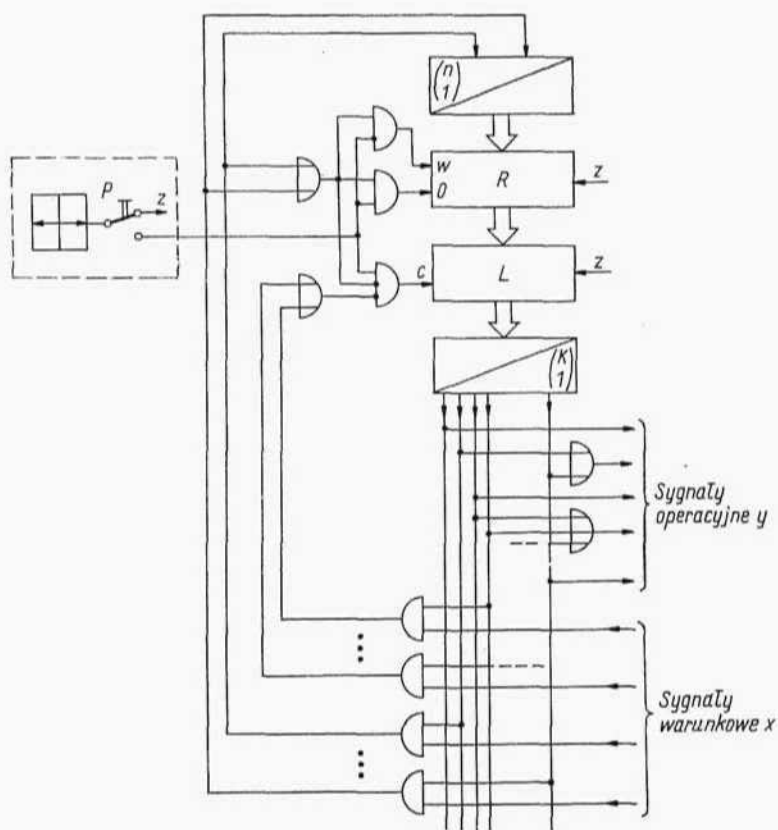
Na rys. 7-29 przedstawiono przykładową strukturę układu sterowania, o rozbudowanej części zmiany stanu. Jest to układ Moore'a, kodowany kodem „1 z  $k$ ”, więc sygnały wyjściowe (operacyjne)  $y$  otrzymuje się wprost z sygnałów oznaczających stan wewnętrzny lub przez sumowanie logiczne tych sygnałów, gdy jakiś sygnał  $y$  występuje w kilku stanach. Układ jest przystosowany do pracy synchronicznej i dlatego zawiera licznik, napędzany generatorem impulsów. Stan licznika jest dekodowany dla uzyskania kodu „1 z  $k$ ”; zastosowanie licznika pierścieniowego umożliwia usunięcie dekodera. W niektórych stanach licznika i przy niektórych sygnałach  $x$  praca ciągła licznika jest przerywana, aż do spełnienia odpowiednich warunków. Napęd licznika jest również przerywany, gdy zachodzi potrzeba zmiany stanu licznika o więcej niż jeden (skok w programie). Wówczas koincydencja odpowiednich sygnałów  $x$  i stanów  $A$  blokuje wejście zegarowe licznika, wpisując przy tym do rejestru nowy stan  $A$  (w kodzie licznika). Stan ten jest następnie przepisowywany do licznika, stając się stanem aktualnym i czynności powtarzają się.

Zastosowanie rejestru i licznika jest celowe wówczas, gdy stosuje się prosty licznik, którego wszystkie wejścia dynamiczne biorą udział w operacji zliczania i wpisać równolegle nowy stan można tylko asynchronicznie, wejściami statycznymi, wykorzystując do tego specjalny takt. W układzie z rys. 7-29 podział czasu na dwa takty został zrealizowany przez wpisywanie do rejestru  $R$ , gdy  $g = 0$  i wpisywanie do licznika  $L$ , gdy  $g = 1$  ( $g$  — sygnał z generatora). Przełącznik  $P$  na rys. 7-29 symbolizuje przełącznik uruchamiający; w czasie przerwy w działaniu wszystkie pamięci są zerowane.

Uproszczenie struktury z rys. 7-29 jest możliwe przez zastosowanie specjalnego licznika (z możliwością dynamicznego wpisywania równoległego), lub gdy sieć działań ma pewne szczególne cechy.

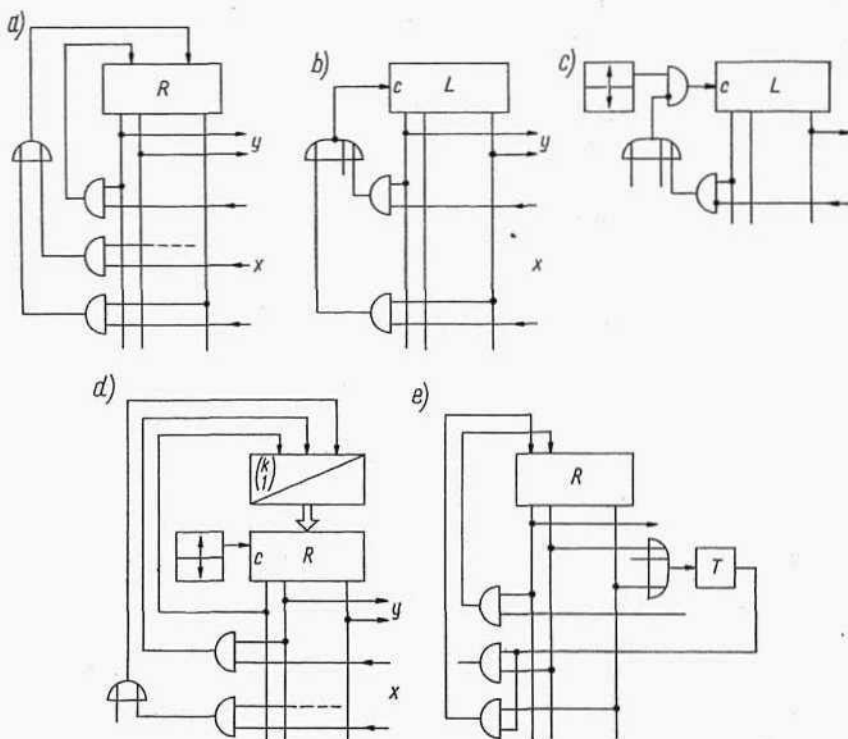


Rys. 7-28. Struktura układu sterowania



Rys. 7-29. Rozbudowany schemat układu sterowania

Układ sterowania jest najprostszy, jeśli w sieci działań występuje bardzo niewiele pętli, obejmujących więcej niż jedną klatkę operacyjną (a pętle obejmujące jedną klatkę mają charakter sprzężenia zwrotnego). Taki program jest nazywany *liniowym* i może być zrealizowany, w przypadku układu asynchronicznego, prostym układem z rejestrem (rys.



Rys. 7-30. Przykłady realizacji zmiany stanu wewnętrznego

7-30a). W stanie  $A_i$  oraz po spełnieniu odpowiednich warunków  $X'$  zostaje wpisany stan 1 do przerzutnika wyznaczającego stan  $A_{i+1}$ , co wpisuje stan 0 do przerzutnika wyznaczającego stan  $A_i$ . Spełnienie odpowiednich warunków powoduje wpisanie 1 do  $A_{i+2}$ , przy zerowaniu  $A_{i+1}$  itd. Przez rozbudowę systemu zerowań można ten układ wykorzystać także w przypadku, gdy w programie występują skoki i pętle.

Inna wersja układu asynchronicznego dla programów liniowych zamiast rejestru stosuje licznik (rys. 7-30b), którego stan zmienia się po spełnieniu odpowiednich warunków  $x$ .

Jeśli program liniowy ma być realizowany w układzie synchronicznym, licznik jest napędzany generatorem impulsów (rys. 7-30c), a ewentualne klatki warunkowe (sygnały  $x$ ) mogą tylko zatrzymywać pracę licznika, aż do spełnienia warunku.

Jeśli w sieci działań układu synchronicznego odcinki liniowe prawie nie występują, może być celowe zrezygnowanie z licznika i generowanie nowego stanu wewnętrznego tak jak w klasycznych układach synchronicznych sekwencyjnych. Zespół przerzutników tworzy rejestr (rys. 7-30d), którego stan jest zmieniany w takt generatora, a narzucany przez stan poprzedni i stan wejść. Gdy rejestr jest zbudowany z przerzutników typu D i pracuje w kodzie „1 z  $k$ ” — koder w układzie staje się zbędny.

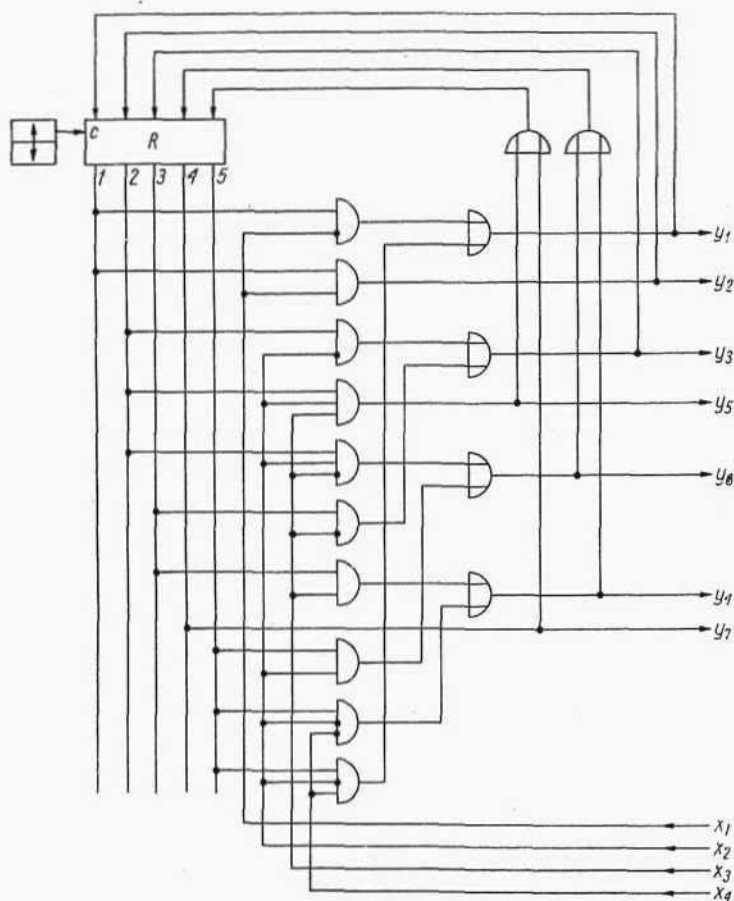
W przypadku, gdy większa część sieci działań ma klatki operacyjne objęte indywidualną pętlą sprzężenia zwrotnego (np. stany  $A_1$  i  $A_3$  na rys. 7-25) warunkującego przejście do innej klatki, zamiast dla pozostałych klatek stosować układ synchroniczny z zegarem i licznikiem, można czas trwania odpowiedniego taktu uzyskać za pomocą układu czasowego. W stanie  $A_i$  zostaje pobudzony układ czasowy (rys. 7-30e), który na impuls wejściowy odpowiada ze zwłoką, wystarczającą na zrealizowanie działań. Przejście do stanu  $A_{i+1}$  jest uzależnione od upływu czasu  $T$  i ewentualnie warunków  $x$ , zwłoka jest więc uważana za dodatkowy warunek.

W układach Mealy'ego zazwyczaj uzyskuje się oszczędniejszą realizację przez połączenie układów kombinacyjnych  $\gamma$  i  $\lambda$ , gdyż wiele warunków ma taką samą postać dla sygnałów wyjściowych jak dla sygnałów wzbudzeń. Na rys. 7-31 przedstawiono jeden z możliwych układów realizujących funkcje zadane tablicą i siecią z rys. 7-27; wspólna część kombinacyjna umożliwiła istotne uproszczenie układu.

Niekiedy rysowanie sieci działań ze wszystkimi operacjami elementarnymi napotyka na trudności, gdyż operacji jest wiele i wielokrotnie powtarzają się (rys. 7-22, 7-23). Blok sterowania może wówczas zawierać kilka rejestrów (liczników), z których każdy realizuje sieć działań innego poziomu. Na przykład pierwszy steruje druk pojedynczych cyfr, drugi

składa z cyfr liczby, trzeci rozkłada liczbę wzdłuż wiersza, zmienia wiersz drukarki itp.

W przypadku urządzeń cyfrowych złożonych istnieje wiele możliwych realizacji zadanych funkcji, a także wiele możliwości podziału zadań między dwie główne części urządzenia: układ sterowania i arytmometr. Można tu wyodrębnić trzy podstawowe koncepcje:



Rys. 7-31. Realizacja sieci działań z rys. 7-27



1. Arytmometr realizuje większość czynności arytmetycznych i logicznych, a porządek ich wykonywania wynika z powiązań schematowych wewnątrz arytmometru. Funkcje układu sterowania sprowadzają się do uruchomienia, zatrzymywania i ewentualnie innych nielicznych czynności.

2. Arytmometr składa się tylko z typowych bloków funkcjonalnych, przystosowanych do realizacji funkcji układu, a współpracą bloków kieruje układ sterowania.

3. Arytmometr zawiera uniwersalny zestaw dodająco-mnożący, a wszystkie działania są programowane przez układ sterujący.

Istnieje, oczywiście wiele wariantów pośrednich, gdyż na wybór koncepcji ma wpływ rodzaj realizowanych działań, ich różnorodność, częstość powtarzania, perspektywy rozszerzenia funkcji, rodzaj elementów itp.

Przykładami różnych wersji mogą być opisane wyżej układy mnożenia. Rozwiązanie z rys. 7-15 jest samowystarczalne; do jego działania wystarczy sygnał  $w = 1$ , ale może ono realizować tylko jedną operację. Liczniki można by wykorzystać również do innych działań, ale wtedy sygnały  $w$ ,  $z$ ,  $c$  muszą być generowane przez układ sterowania. Jeszcze bardziej złożony program, ale i większą uniwersalność otrzymuje się przy zastosowaniu układu z rys. 7-14a, przystosowanego do działań uniwersalnych, a więc bez wejścia  $X_2$  i sprzężeń z  $RP$  do  $R$ . Akumulator może wówczas dodawać kolejne liczby  $X$ , po przesunięciu zawartości  $A$  do  $RP$  i zastosowaniu odpowiedniego sprzężenia uzyskuje się mnożenie:  $A := RP \times X$ , wprowadzenie  $X$  w zapisie dopełniającym umożliwia realizację odejmowania itd. Kosztem rozbudowy programu i układu sterującego w takim uniwersalnym rozwiązaniu można zrealizować dowolne działanie arytmetyczne.

## 7.4. PRZYKŁADY SYNTEZY URZĄDZEŃ

### 7.4.1. URZĄDZENIE STEROWANIA SEKWENCYJNEGO

**Zadanie.** Zaprojektować układ sterowania procesu hartowania kół zębatych. Urządzenie do hartowania składa się z pieca przelotowego oraz wanny. Koła zębate są przemieszczane samotokiem, w grupach. Gdy jedna grupa jest wprowadzana do pieca, druga jest wyprowadzana i powinno się to odbywać przy zwiększonej szybkości samotoku. Odcinek samotoku