3. SYNTEZA UKŁADÓW KOMBINACYJNYCH

3.1. ZASADY OGÓLNE

3.1.1. ZAPIS FUNKCJI

Synteza układów przełączających to zespół czynności, które na podstawie założeń dotyczących działania układów — doprowadza ją do schematu logicznego układu, przy czym schemat ten powinien zawierać tylko elementy przewidzianego typu i spełniać pewne wymagania optymalności. W przypadku układu kombinacyjnego schemat logiczny można jednoznacznie opisać rodziną funkcji przełączających

$$y_i = f_i(x_1, x_2, ..., x_n) \qquad i = 1, 2, ..., m$$

więc za cel syntezy można uważać uzyskanie tych funkcji o odpowiedniej postaci.

Założenia dotyczące działania układu najczęściej są podawane w postaci opisu słownego, tablicy wartości funkcji lub wykresu czasowego.

Opis słowny funkcji realizowanych przez układ musi jednoznacznie określić przypadki, w których sygnały wyjściowe mają wartość 0 albo 1. Przykładem poprawnego opisu może być zdanie: „zaprojektować układ z elementów I, LUB, NIE o trzech wejściach x_1, x_2, x_3, wyróżniający sygnałem $y = 1$ przypadki, gdy na wejściu pojawi się liczba dwójkowa nieparzysta lub podzielna przez 3 (x_3 odpowiada pozycji najmniej znaczącej)".

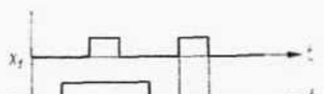
Tablica wartości funkcji to zestaw wszystkich możliwych wartości sygnałów wejściowych i odpowiadających im wartości sygnałów wyjściowych. Dla układu zadanego powyższym opisem słownym tablica wartości funkcji ma np. postać:

	x_1	x_2	x_3	y
0	0	0	0	0
1	0	0	1	1
2	0	1	0	0
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	1

Dla ułatwienia podano tu z boku odpowiednią liczbę dziesiętną.

Tablica jest bardziej konkretną i pełną postacią zapisu funkcji niż opis słowny i dlatego często z opisu słownego przechodzi się do tablicy, sprawdzając przy tym, czy posiadane informacje o funkcji są pełne i wystarczają do wypełnienia wszystkich pozycji. Niekiedy zdarza się, że wartość funkcji przy pewnych kombinacjach sygnałów wejściowych jest dowolna lub nieokreślona, np. gdy z jakichś względów takie kombinacje wejściowe nigdy w rzeczywistości nie występują. Funkcja nosi wówczas nazwę *funkcji niepełnej* (*niezupełnej, nie w pełni określonej*), a w jej tablicy, w odpowiednich pozycjach kolumny y, stawia się kreskę.

Wykres czasowy jest rzadziej spotykaną postacią zapisu układu kombinacyjnego, ale niekiedy bywa pomocny. Na rys. 3-1 przedstawiono przebiegi dla funkcji niepełnej (nieokreślonej wartości y odpowiada linia



Rys. 3-1. Wykres czasowy układu kombinacyjnego

przerywana). Zwykle zakłada się, że przypadki nie pokazane na wykresie dotyczą nieokreślonej wartości y i dlatego rysując wykres należy uwzględniać wszystkie kombinacje robocze. Łatwo zauważyć, że wykres czasowy czytany z góry w dół dla określonej chwili t opisuje jeden wiersz tablicy wartości funkcji, więc i między tymi postaciami zapisu istnieje pełna równoważność.

3.1.2. POSTAĆ KANONICZNA FUNKCJI

Ogólną postać funkcji przełączającej można zawsze rozbić na dwa składniki lub czynniki, wg zależności:

$$f(x_1, x_2, \ldots, x_n) = x_1 \cdot f(1, x_2, \ldots, x_n) + \bar{x}_1 \cdot f(0, x_2, \ldots, x_n)$$
$$f(x_1, x_2, \ldots, x_n) = [x_1 + f(0, x_2, \ldots, x_n)][\bar{x}_1 + f(1, x_2, \ldots, x_n)]$$

Prawdziwość tych związków można łatwo sprawdzić, podstawiając obustronnie dwie możliwe wartości x_1: 0 i 1. W podobny sposób każdy z otrzymanych składników (czynników) można dalej rozłożyć względem innej zmiennej, np.

$$\begin{aligned} f(x_1, x_2, \ldots, x_3) = \\ = x_1 x_2 \cdot f(1, 1, x_3, \ldots, x_n) + x_1 \bar{x}_2 \cdot f(1, 0, x_3, \ldots, x_n) + \\ + \bar{x}_1 x_2 \cdot f(0, 1, x_3, \ldots, x_n) + \bar{x}_1 \bar{x}_2 \cdot f(0, 0, x_3, \ldots, x_n), \end{aligned}$$

Jeśli operację tę przeprowadzi się dla wszystkich zmiennych x, to rezultatem przekształcenia będzie zależność

$$\begin{aligned} f(x_1, x_2, \ldots, x_n) = x_1 x_2 \ldots x_n \cdot f(1, 1, \ldots, 1) +, \\ + x_1 x_2 \ldots x_{n-1} \bar{x}_n \cdot f(1, 1, \ldots, 1, 0) + \ldots, \\ \ldots + x_1 \bar{x}_2 \ldots \bar{x}_n \cdot f(1, 0, \ldots, 0) + \bar{x}_1 \bar{x}_2 \ldots \bar{x}_n \cdot f(0, 0, \ldots, 0) \end{aligned} \tag{3-1}$$

Podobnie, z rozkładania na czynniki otrzymuje się

$$\begin{aligned} \mathrm{f}(x_1 x_2, \ldots, x_n) = [x_1 + x_2 + \ldots + x_n + f(0, 0, \ldots, 0)][x_1 + x_2 + \ldots \\ \ldots + x_{n-1} + \bar{x}_n + f(0, 0, \ldots, 0, 1)] \ldots [x_1 + \bar{x}_2 + \ldots \\ \ldots + \bar{x}_n + f(0, 1, \ldots, 1)][\bar{x}_1 + \bar{x}_2 + \ldots + \bar{x}_n + f(1, 1, \ldots, 1)] \end{aligned} \tag{3-2}$$

W tych bardzo ważnych zależnościach występują pewne prawidłowości, które łatwiej będzie określić wprowadzając dodatkowe oznaczenia.

Iloczyn wszystkich argumentów funkcji (z negacjami lub bez) będzie nazywany *iloczynem pełnym* i oznaczany przez K z odpowiednim indeksem. Indeks jest liczbą dwójkową (lub równoważną — dziesiętną) utworzoną przez przyporządkowanie każdej zmiennej x_i symbolu 1, a zmiennej $\bar{x}_i$ —

symbolu 0. Tak więc iloczynowi pełnemu x_1x_2 odpowiada indeks (11), czyli 3 i symbol K_3, pełnemu iloczynowi $\bar{x}_1x_2\bar{x}_3$ — indeks (010), czyli 2 i symbol K_2, itd.

Suma wszystkich argumentów funkcji z negacjami lub bez będzie nazywana *pełną sumą* i oznaczana przez D z indeksem. Indeks tworzy się tu odwrotnie niż przy *iloczynie pełnym* — przyporządkowując zmiennej x_i–0, a zmiennej $\bar{x}_i$–1. Sumie pełnej x_1+x_2 odpowiada więc indeks (00), czyli 0 i symbol D_0, natomiast sumie pełnej $\bar{x}_1+x_2+\bar{x}_3$ — indeks (101), czyli 5 i D_5.

Symbole K_i i D_i jednoznacznie określają iloczyn pełny i sumę pełną, gdy jest znana liczba zmiennych n.

Również wartość funkcji dla konkretnych wartości argumentów dogodnie jest oznaczyć symbolem α, z indeksem w postaci liczby dziesiętnej, odpowiadającej wartościom argumentów, np.

$$f(1, 1) = \alpha_3 \qquad f(1, 0, 1) = \alpha_5 \text{ itd.}$$

Wprowadzając te oznaczenia, wzory (3-1) i (3-2) dla funkcji dwóch zmiennych można zapisać w postaci

$$f(x_1, x_2) = x_1x_2\cdot f(1, 1)+x_1\bar{x}_2\cdot f(1, 0)+\bar{x}_1x_2\cdot f(0, 1)+$$

$$+\bar{x}_1\bar{x}_2\cdot f(0, 0) = K_3\cdot\alpha_3+K_2\cdot\alpha_2+K_1\cdot\alpha_1+K_0\cdot\alpha_0 = \sum_{i=0}^{3} K_i\cdot\alpha_i$$

$$f(x_1, x_2) = [x_1+x_2+f(0, 0)][x_1+\bar{x}_2+f(0, 1)][\bar{x}_1+x_2+f(1, 0)]$$

$$[\bar{x}_1+\bar{x}_2+f(1, 1)] = (D_0+\alpha_0)(D_1+\alpha_1)(D_2+\alpha_2)(D_3+\alpha_3) =$$

$$= \prod_{i=0}^{3} (D_i+\alpha_i)$$

Podobnie dla n zmiennych

$$f(x_1, x_2, \ldots, x_n) = \sum_{i=1}^{2^n-1} \alpha_i\cdot K_i \qquad (3\text{-}3)$$

$$f(x_1, x_2, \ldots, x_n) = \prod_{i=0}^{2^n-1} (\alpha_i+D_i) \qquad (3\text{-}4)$$

Symbol $\sum$ oznacza tu sumę logiczną; a $\prod$ — iloczyn logiczny.

Powyższe zależności umożliwiają łatwe przejście od tablicy wartości funkcji (lub innego równoważnego zapisu) do wyrażenia logicznego. Tablica przedstawia kolejne wartości α_i. Ponieważ $0 \cdot K_i = 0$ oraz $1 \cdot K_i = K_i$ — do przedstawienia funkcji wg zależności (3-3) należy wypisać sumę tych K_i, dla których $\alpha_i = 1$. Dla podanej wyżej tablicy będzie to wyrażenie

$$y(x_1, x_2, x_3) = K_1 + K_3 + K_5 + K_6 + K_7$$

czyli

$$y = \bar{x}_1 \bar{x}_2 x_3 + \bar{x}_1 x_2 x_3 + x_1 \bar{x}_2 x_3 + x_1 x_2 \bar{x}_3 + x_1 x_2 x_3$$

Formułę taką można otrzymać bezpośrednio z tablicy, biorąc pod uwagę jedynie te wiersze, dla których $y = 1$, i przypisując wartościom $x_i = 0$ zmienną $\bar{x}_i$, a wartościom $x_i = 1$ — zmienną x_i. Tak utworzone iloczyny pełne należy dodać. Uzyskana postać funkcji nosi nazwę ***postaci kanonicznej sumy***. Z zasad jej tworzenia wynika, że każda funkcja może mieć tylko jedną taką postać.

Przy przedstawianiu funkcji wg zależności (3-4) należy zauważyć, że $0 + D_i = D_i$, $1 + D_i = 1$, więc w odpowiednim wyrażeniu należy wypisać iloczyn tych D_i, dla których $\alpha_i = 0$. Dla rozpatrywanego przykładu będzie to wyrażenie

$$y(x_1, x_2, x_3) = D_0 \cdot D_2 \cdot D_4$$

czyli

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

Formułę taką można otrzymać bezpośrednio z tablicy, biorąc pod uwagę jedynie te wiersze, dla których $y = 0$ i przypisując wartościom $x_i = 0$ zmienną x_i, a wartościom $x_i = 1$ — zmienną $\bar{x}_i$. Tak utworzone pełne sumy należy pomnożyć. Uzyskana postać funkcji nosi nazwę ***postaci kanonicznej iloczynu*** i również jest jedna dla każdej funkcji.

Postaci kanoniczne często zapisuje się w skrócie, w postaci zbioru indeksów (bez K i D) z odpowiednim symbolem operacji, np.

$$y(x_1, x_2, x_3) = \sum (1, 3, 5, 6, 7)$$

oraz

$$y(x_1, x_2, x_3) = \prod (0, 2, 4)$$

Dla funkcji bez wartości nieokreślonych, występujące w tych dwóch postaciach liczby indeksowe muszą wyczerpywać wszystkie możliwe wartości od 0 do 2^n-1, co umożliwia łatwe wyznaczenie jednej postaci na podstawie znajomości drugiej (przez uzupełnienie liczb). Na przykład

jeśli $y_1(x_1, x_2) = \sum(0,1)$, to $y_1(x_1, x_2) = \prod(2, 3)$

jeśli $y_2(x_1, x_2, x_3) = \prod(4, 5, 7)$, to $y_2(x_1, x_2, x_3) = \sum(0, 1, 2, 3, 6)$

itd.

Gdy funkcja jest niepełna, indeks odpowiadający kombinacji nieokreślonej może wystąpić zarówno w jednej postaci jak i w drugiej, gdyż y może wówczas być zarówno 0 jak i 1. Oznacza się to zwykle przez wzięcie odpowiedniego składnika w nawias. Na przykład z rys. 3-1 otrzymuje się

$$y(x_1, x_2) = \sum[2, (1)], \quad \text{czyli} \quad y(x_1 x_2) = x_1\bar{x}_2 + (\bar{x}_1 x_2)$$

oraz

$$y(x_1, x_2) = \prod[0, 3(1)], \quad \text{czyli} \quad y(x_1, x_2) = (x_1+x_2)(\bar{x}_1+\bar{x}_2)[(x_1+\bar{x}_2)].$$

Warto jeszcze zwrócić uwagę na łatwą interpretację otrzymywanych opisanymi metodami wyników w rachunku zdań. Uzyskaną wyżej funkcję.

$$y = x_1\bar{x}_2 + \bar{x}_1 x_2$$

można czytać w następujący sposób:

„$y = 1$, gdy $x_1 = 1$ i $x_2 = 0$ lub gdy $x_1 = 0$ i $x_2 = 1$"

Odpowiada to dokładnie sytuacji na rysunku. Tak więc — odwracając czynności — można z rysunku lub tablicy otrzymać formułę logiczną, wykorzystując rachunek zdań.

3.1.3. ZASADY MINIMALIZACJI FUNKCJI

Jeśli w skład syntetyzowanego schematu logicznego mają wchodzić elementy I, LUB, NIE, to uzyskane podanymi wyżej metodami postaci kanoniczne umożliwiają już utworzenie takiego schematu. W tym celu należy do negowania zmiennej użyć elementu NIE, do uzyskania ilo-

czynu pełnego oraz iloczynu sum pełnych — elementów I, a do uzyskania sumy pełnej oraz sumy iloczynów pełnych — elementu LUB. Jak łatwo policzyć, realizacja otrzymanej wyżej postaci kanonicznej sumy (dla przykładu ze str. 87) wymaga 3 elementów NIE, 5 elementów I, 1 elementu LUB, natomiast odpowiednia postać iloczynu może być zrealizowana z 2 elementów NIE, 3 elementów LUB, 1 elementu I.
Można przypuszczać, że druga wersja będzie tańsza i prostsza, a więc lepsza.

Pomijając szczegółowe problemy optymalizacji układu można stwierdzić, że na ogół układ o mniejszej liczbie elementów jest tańszy i bardziej niezawodny, a spośród dwóch układów o takiej samej liczbie elementów logicznych lepszy jest ten, który operuje mniejszą liczbą sygnałów (mniej wejść mają w sumie wszystkie jego elementy). Tak więc, niezależnie od zastosowanych dalej elementów schematu logicznego, bardzo ważnym etapem syntezy jest poszukiwanie takiej postaci funkcji przełączającej, w której występuje minimalna liczba liter (tzn. zmiennych lub ich negacji). Proces poszukiwania tej postaci minimalnej nosi nazwę minimalizacji funkcji i opiera się na tzw. *regułach sklejania*:

$$Ax+A\bar{x} = A$$

$$(B+x)(B+\bar{x}) = B$$

w których A i B — zmienne lub funkcje przełączające. Reguły te można wyrazić następująco: suma lub iloczyn dwóch wyrażeń różniących się między sobą tylko na jednej pozycji — znakiem negacji, mogą być zastąpione jednym wyrażeniem, bez litery stanowiącej różnicę. Na przykład:

$$x_1x_2+x_1\bar{x}_2 = x_1(x_2+\bar{x}_2) = x_1\cdot 1 = x_1$$

$$\bar{x}_1x_2x_3+\bar{x}_1x_2\bar{x}_3 = \bar{x}_1x_2$$

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

Wyrażenia podlegające sklejaniu nazywane są *wyrażeniami sąsiednimi*. Jeśli w otrzymanych postaciach kanonicznych funkcji takie wyrażenia sąsiednie występują, to postać kanoniczną można uprościć, z wyraźną korzyścią dla realizacji technicznej. Postać kanoniczna sumy z przykładu zaczętego wyżej można zminimalizować w następujący sposób:

$$y = \bar{x}_1\bar{x}_2x_3 + \bar{x}_1x_2x_3 + x_1\bar{x}_2x_3 + x_1x_2\bar{x}_3 + x_1x_2x_3 =$$
$$= (\bar{x}_1\bar{x}_2x_3 + x_1\bar{x}_2x_3) + (\bar{x}_1x_2x_3 + x_1x_2x_3) + (x_1x_2\bar{x}_3 + x_1x_2x_3) =$$
$$= \bar{x}_2x_3 + x_2x_3 + x_1x_2 = x_3 + x_1x_2$$

Aby utworzyć pary wyrażeń sąsiednich, wpisano tu dwukrotnie iloczyn $x_1x_2x_3$, ale jest to dozwolone, gdyż w algebrze Boole'a $A = A + A$, więc każdy składnik może się powtarzać, bez zmiany wartości funkcji.

Przeprowadzając sklejanie w postaci kanonicznej iloczynu (dla tego samego przykładu) otrzymuje się

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

Z zasady rozdzielności (1-13) wynika, że obydwa te wyniki są sobie równe, a do realizacji prostszego z nich są potrzebne elementy: $1 \times$ LUB, $1 \times$ I. Różnica między realizacją postaci kanonicznej i postaci minimalnej jest więc bardzo duża.

W wyniku sklejania uzyskuje się wyrażenia, które już nie są postaciami kanonicznymi, ale zachowują postać sumy iloczynów oraz iloczynu sum. Wyrażenia tego typu przyjęto nazywać *postacją normalną sumy* (*normalną postacią alternatywną*) i *postacią normalną iloczynu* (*normalną postacią koniunkcyjną*).

W przypadku złożonych funkcji wielu zmiennych wyszukiwanie wyrażeń sąsiednich i sklejanie w sposób podany wyżej staje się bardzo uciążliwe, zwłaszcza że dla otrzymania postaci minimalnej trzeba dokonać wszystkich możliwych sklejeń. Istnieją inżynierskie metody upraszczające procedurę minimalizacji, gdy liczba zmiennych nie przekracza 6 lub gdy liczba wyrazów zadanej postaci kanonicznej nie jest zbyt duża. W innych przypadkach rozwiązań minimalnych trzeba szukać z pomocą cyfrowych maszyn matematycznych.

3.1.4. METODA KARNAUGHA

Metoda tablic Karnaugha (zwana też metodą kart Veitcha) ułatwia sklejanie przez takie usytuowanie na płaszczyźnie wyrażeń postaci kanonicznej, aby wyrażenia sąsiednie podlegające sklejeniu były umieszczo-