Synteza liczników synchronicznych

Załóżmy, że chcemy zaprojektować licznik synchroniczny o następującej sekwencji:

0 → 1 → 2 → 3 → 6 → 5 → 4 → [0 → ... sekwencja jest powtarzana]

Ponieważ licznik ma 7 stanów, więc do ich zakodowania potrzebne są 3 bity (oznaczymy je jako Q_2, Q_1, Q_0). Na podstawie sekwencji stanów można stworzyć tabelę stanów (tabela 1). Po zakodowaniu stanów na trzech bitach otrzymujemy tabelę 2.

Tab. 1

t	t+1
0	1
1	2
2	3
3	6
4	0
5	4
6	5

⇨

Tab. 2

t			t+1		
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0
0	0	0	0	0	1
0	0	1	0	1	0
0	1	0	0	1	1
0	1	1	1	1	0
1	0	0	0	0	0
1	0	1	1	0	0
1	1	0	1	0	1

Na przykład pierwszy wiersz tabeli 2 pokazuje, że jeśli w chwili t licznik jest w stanie 0 ($Q_2\,Q_1\,Q_0 = 000$), to w chwili t+1 powinien przejść do stanu 1 ($Q_2\,Q_1\,Q_0 = 001$). Oznacza to zmianę 0 → 1 dla przerzutnika Q_0 i zmianę 0 → 0 (czyli brak zmiany stanu) dla pozostałych przerzutników (Q_1 i Q_2). Analogiczna jest interpretacja pozostałych wierszy tabeli 2.

Aby uzyskać żądaną zmianę stanu przerzutnika należy podać na jego wejście informacyjne odpowiednią wartość logiczną. Można ją ustalić na podstawie tabeli wzbudzeń przerzutnika. Tabele wzbudzeń dla przerzutników D i JK przedstawione są w tabeli 3 (symbol Ø oznacza dowolną wartość logiczną).

Tab. 3

$Q(t)$	$Q(t+1)$	$D(t)$	$J(t)$	$K(t)$
0	0	0	0	Ø
0	1	1	1	Ø
1	0	0	Ø	1
1	1	1	Ø	0

Na podstawie zakodowanej tabeli stanów (tabela 2) i tabeli wzbudzeń można określić funkcje pobudzeń dla wejść przerzutników przedstawione w tabeli 4. Zasada tworzenia tej tabeli zilustrowana jest poniżej na przykładzie pierwszego wiersza:

t			t+1			⇨	D_2	D_1	D_0		J_2	K_2	J_1	K_1	J_0	K_0
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0											
0	0	**0**	0	0	**1**		0	0	**1**		0	Ø	0	Ø	**1**	**Ø**

0 → 1

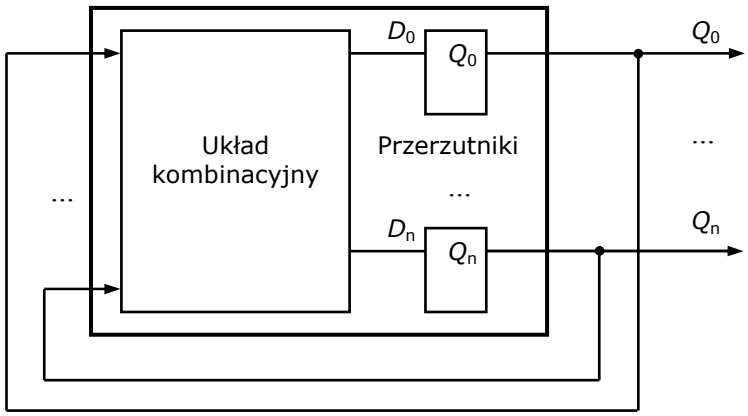
drugi wiersz tabeli wzbudzeń

Zwykle podczas syntezy tabele 1 i 2 są pomijane, od razu wypełniana jest tabela 4. Można zauważyć, że dla przerzutnika D przejście przez tabelę wzbudzeń jest właściwie zbędne. W związku z tym, że $Q(t+1) = D(t)$ do kolumn $D_2\,D_1\,D_0$ można po prostu przepisać zawartość kolumn $Q_2\,Q_1\,Q_0$ z chwili $t+1$.

Tab. 4

t			$t+1$			⇨	D_2	D_1	D_0		J_2	K_2	J_1	K_1	J_0	K_0
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0											
0	0	0	0	0	1		0	0	1		0	Ø	0	Ø	1	Ø
0	0	1	0	1	0		0	1	0		0	Ø	1	Ø	Ø	1
0	1	0	0	1	1		0	1	1		0	Ø	Ø	0	1	Ø
0	1	1	1	1	0		1	1	0		1	Ø	Ø	0	Ø	1
1	0	0	0	0	0		0	0	0		Ø	1	0	Ø	0	Ø
1	0	1	1	0	0		1	0	0		Ø	0	0	Ø	Ø	1
1	1	0	1	0	1		1	0	1		Ø	0	Ø	1	1	Ø

Ustalenie sygnałów pobudzeń dla wejść informacyjnych przerzutników sprowadza się do syntezy zwykłego układu kombinacyjnego, w którym sygnałami wejściowymi są stany $Q_2\,Q_1\,Q_0$ w chwili t a wyjścia tego układu są podawane na wejścia przerzutników. Pokazuje to poniższy rysunek:



Syntezę układu kombinacyjnego można przeprowadzić dowolną metodą. Na przykład dla przerzutnika Q_0 funkcję D_0 można zapisać wprost z postaci kanonicznej sumy:

$$D_0 = \overline{Q_2}\,\overline{Q_1}\,\overline{Q_0} + \overline{Q_2}\,Q_1\,\overline{Q_0} + Q_2\,Q_1\,\overline{Q_0}$$

Otrzymane wyrażenie można dalej minimalizować metodą przekształceń formalnych. W podobny sposób można wyprowadzić funkcje dla pozostałych przerzutników.

Zwykle jednak lepsze efekty (przy mniejszym nakładzie pracy) można uzyskać stosując minimalizację metodą siatek Karnaugha. Trzeba pamiętać, że w projektowanym liczniku stan $Q_2\,Q_1\,Q_0 = 111$ nie jest wykorzystywany, można więc dla tej kombinacji wpisać w siatce symbol Ø, co może ułatwić minimalizację. Dla przejrzystości puste komórki w siatkach reprezentują wartość 0. Pełne siatki Karnaugha dla przerzutników D wyglądają następująco:

Tab. 5 D_2

Q_2Q_1 \ Q_0	0	1
00		
01		1
11	1	Ø
10		1

$$D_2 = Q_2\,Q_1 + Q_1\,Q_0 + Q_2\,Q_0$$

Tab. 6 D_1

Q_2Q_1 \ Q_0	0	1
00		1
01	1	1
11		Ø
10		

$$D_1 = \overline{Q_2}\,Q_1 + \overline{Q_2}\,Q_0$$

Tab. 7 D_0

Q_2Q_1 \ Q_0	0	1
00	1	
01	1	
11	1	Ø
10		

$$D_0 = \overline{Q_2}\,\overline{Q_0} + Q_1\,\overline{Q_0}$$

Uzyskane z siatek Karnaugha funkcje można jeszcze uprościć (stosując wyłączenie przed nawias) i przekształcić według zadanego typu bramek (przyjęto dwuwejściowe bramki NOR). W przekształceniach wykorzystane zostały prawa de Morgana i zasada neutralności podwójnej negacji:

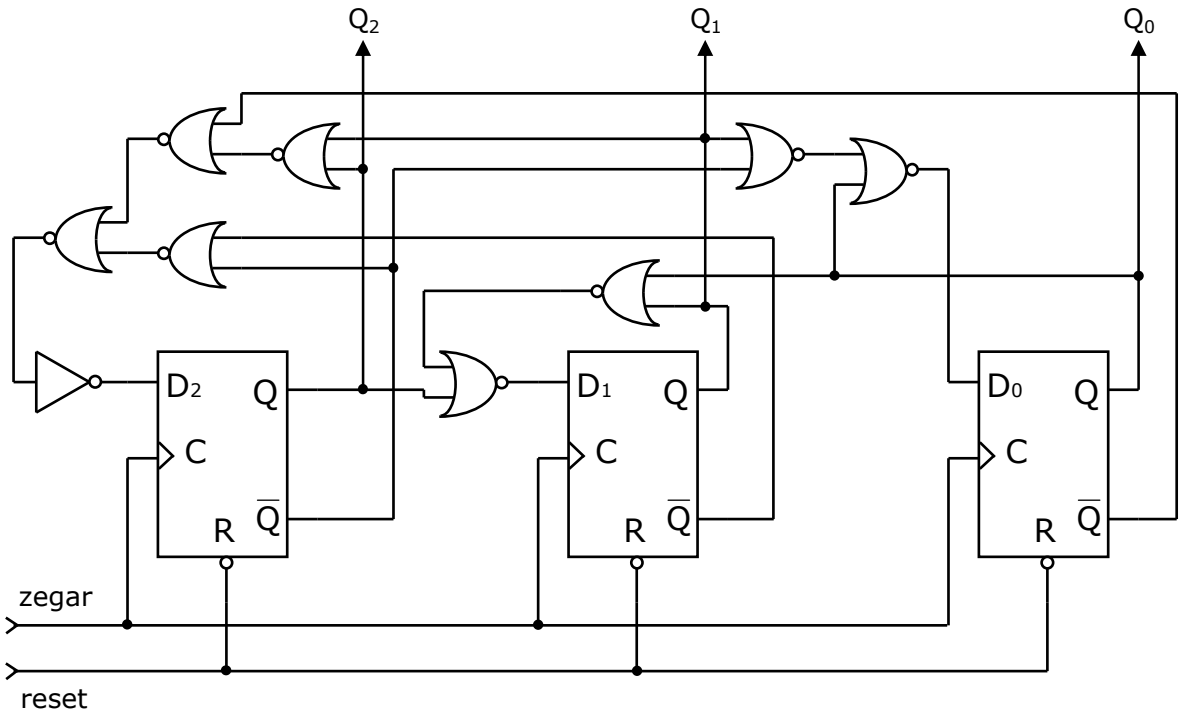
$$D_2 = Q_2\,Q_1 + Q_1\,Q_0 + Q_2\,Q_0 = Q_2\,Q_1 + Q_0\,(Q_1 + Q_2) = \overline{\overline{\overline{Q_2}\,\overline{Q_1}} + \overline{\overline{Q_0}\,\overline{(Q_1 + Q_2)}}} =$$

$$\overline{\overline{\overline{Q_2} + \overline{Q_1}} + \overline{\overline{Q_0} + \overline{(Q_1 + Q_2)}}}$$

$$D_1 = \overline{Q_2}Q_1 + \overline{Q_2}Q_0 = \overline{Q_2}\,(Q_1 + Q_0) = \overline{\overline{\overline{Q_2}\,(Q_1 + Q_0)}} = \overline{Q_2 + \overline{(Q_1 + Q_0)}}$$

$$D_0 = \overline{Q_2}\,\overline{Q_0} + Q_1\,\overline{Q_0} = \overline{Q_0}\,(\overline{Q_2} + Q_1) = \overline{\overline{\overline{Q_0}\,(\overline{Q_2} + Q_1)}} = \overline{Q_0 + \overline{(\overline{Q_2} + Q_1)}}$$

Na podstawie końcowych postaci funkcji można narysować schemat licznika.



Dla przerzutników JK synteza jest bardziej pracochłonna (osobne funkcje dla wejść J i K stąd dwukrotnie większa liczba siatek Karnaugha). Często jednak można uzyskać funkcje logiczne prostsze niż dla przerzutnika D.

Siatki Karnaugha dla wariantu z przerzutnikami JK wyglądają następująco:

Tab. 8 J_2

Q_2Q_1 \ Q_0	0	1
00		
01		1
11	Ø	Ø
10	Ø	Ø

$J_2 = Q_1 Q_0$

Tab. 9 K_2

Q_2Q_1 \ Q_0	0	1
00	Ø	Ø
01	Ø	Ø
11		Ø
10	1	

$K_2 = \overline{Q_1}\,\overline{Q_0}$

Tab. 10 J_1

Q_2Q_1 \ Q_0	0	1
00		1
01	Ø	Ø
11	Ø	Ø
10		

$J_1 = \overline{Q_2}\,Q_0$

Tab. 11 K_1

Q_2Q_1 \ Q_0	0	1
00	Ø	Ø
01		
11	1	Ø
10	Ø	Ø

$K_1 = Q_2$

Tab. 12 J_0

Q_2Q_1 \ Q_0	0	1
00	1	Ø
01	1	Ø
11	1	Ø
10		Ø

$J_0 = \overline{Q_2} + Q_1$

Tab. 13 K_0

Q_2Q_1 \ Q_0	0	1
00	Ø	1
01	Ø	1
11	Ø	Ø
10	Ø	1

$K_0 = 1$

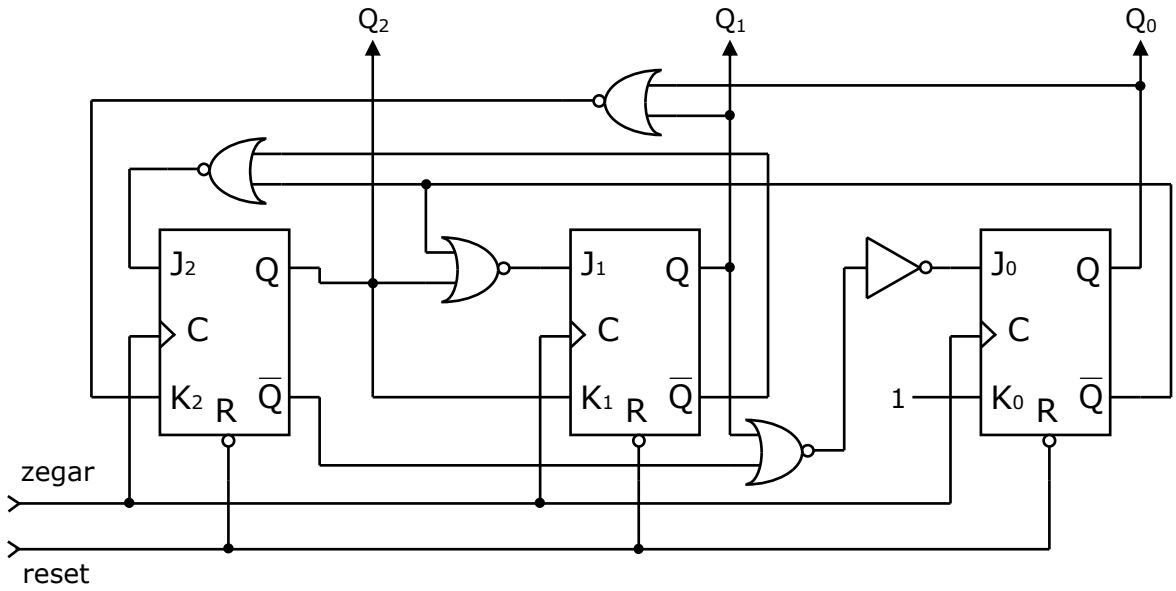
Przekształcenia funkcji dla dwuwejściowych bramek NOR:

$$J_2 = Q_1 Q_0 = \overline{\overline{Q_1 Q_0}} = \overline{\overline{Q_1} + \overline{Q_0}} \qquad K_2 = \overline{Q_1}\,\overline{Q_0} = \overline{\overline{\overline{Q_1}\,\overline{Q_0}}} = \overline{Q_1 + Q_0}$$

$$J_1 = \overline{Q_2}\,Q_0 = \overline{\overline{\overline{Q_2}\,Q_0}} = \overline{Q_2 + \overline{Q_0}} \qquad K_1 = Q_2$$

$$J_0 = \overline{Q_2} + Q_1 = \overline{\overline{\overline{Q_2} + Q_1}} \qquad K_0 = 1$$

Zgodnie z oczekiwaniem układ logiczny dla przerzutników JK (przy tym samym typie użytych bramek) jest znacznie prostszy (potrzeba 4 x NOR + 1 x NOT, dla przerzutników D było 8 x NOR + 1 x NOT). Schemat licznika w wariancie z przerzutnikami JK przedstawiony jest poniżej.



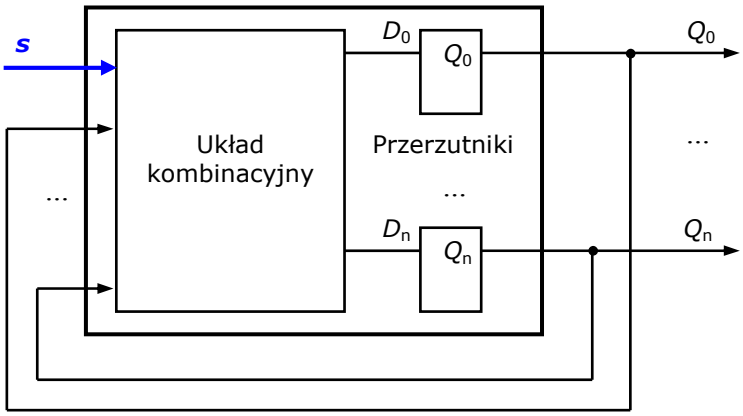
Synteza licznika dwukierunkowego

W pokazanym wcześniej przykładzie sekwencja stanów licznika była stała, dlatego układ nie posiadał żadnych zewnętrznych wejść informacyjnych. Funkcje sterujące wejściami przerzutników zależały tylko od stanów przerzutników w chwili *t*.

Zmodyfikujemy licznik wprowadzając dodatkowe wejście sterujące *s*, wpływające na sposób pracy licznika (odwracające kolejność stanów w sekwencji):

$s = 0$: $0 \rightarrow 1 \rightarrow 2 \rightarrow 3 \rightarrow 6 \rightarrow 5 \rightarrow 4 \rightarrow [0 \rightarrow \ldots]$
$s = 1$: $0 \rightarrow 4 \rightarrow 5 \rightarrow 6 \rightarrow 3 \rightarrow 2 \rightarrow 1 \rightarrow [0 \rightarrow \ldots]$

Strukturę logiczną licznika dwukierunkowego przedstawia poniższy rysunek (nowy element to sygnał *s* na wejściu układu kombinacyjnego):



Synteza w takim przypadku jest podobna do przedstawionej wcześniej. Różnica polega na tym, że tabela 4 (dla kolumn w chwili *t*) zawiera dodatkowy sygnał *s*, służący do wyboru kierunku zliczania. Stan licznika w chwili *t*+1 zależy teraz nie tylko od bieżącego stanów przerzutników $Q_2 Q_1 Q_0$ ale również od sygnału *s*. W tabeli 4 pojawia się dodatkowa, dolna część odpowiadająca odwrotnej sekwencji stanów dla $s = 1$.

Tab. 4a

t				t+1			⇨	D_2	D_1	D_0		J_2	K_2	J_1	K_1	J_0	K_0
s	Q_2	Q_1	Q_0	Q_2	Q_1	Q_0											
0	0	0	0	0	0	1		0	0	1		0	Ø	0	Ø	1	Ø
0	0	0	1	0	1	0		0	1	0		0	Ø	1	Ø	Ø	1
0	0	1	0	0	1	1		0	1	1		0	Ø	Ø	0	1	Ø
0	0	1	1	1	1	0		1	1	0		1	Ø	Ø	0	Ø	1
0	1	0	0	0	0	0		0	0	0		Ø	1	0	Ø	0	Ø
0	1	0	1	1	0	0		1	0	0		Ø	0	0	Ø	Ø	1
0	1	1	0	1	0	1		1	0	1		Ø	0	Ø	1	1	Ø
1	0	0	0	1	0	0		1	0	0		1	Ø	0	Ø	0	Ø
1	0	0	1	0	0	0		0	0	0		0	Ø	0	Ø	Ø	1
1	0	1	0	0	0	1		0	0	1		0	Ø	Ø	1	1	Ø
1	0	1	1	0	1	0		0	1	0		0	Ø	Ø	0	Ø	1
1	1	0	0	1	0	1		1	0	1		Ø	0	0	Ø	1	Ø
1	1	0	1	1	1	0		1	1	0		Ø	0	1	Ø	Ø	1
1	1	1	0	0	1	1		0	1	1		Ø	1	Ø	0	1	Ø

Sygnał sterujący s będzie obecny również w siatkach Karnaugha i w konsekwencji w funkcjach logicznych po minimalizacji.

W podobny sposób można przeprowadzić syntezę w przypadku większej liczby sygnałów sterujących. Na przykład jeden z sygnałów może służyć (tak jak w pokazanym przykładzie) do wyboru kierunku a drugi do wyboru jednej z dwóch różnych sekwencji. Można w ten sposób uzyskać aż cztery różne warianty działania licznika.