

Synteza liczników synchronicznych

Załóżmy, iż mamy zaprojektować licznik synchroniczny liczący w kodzie 0123654. Ponieważ jest tutaj 7 wartości, więc do zakodowania potrzeba 3 bitów (które oznaczymy jako Q_2, Q_1, Q_0). Narysujmy tabelę stanów. Na podstawie kodu wypełnimy Tab.1. Natomiast w Tab.2 jest zakodowaną tabelą Tab.1

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

Interpretacja tabeli Tab.2 jest następująca: np. W stanie 0 (000) w chwili t przerzutnik Q_0 przechodzi ze stanu 0 (w chwili t) na 1 (w chwili t+1). Pozostałe przerzutniki (Q_1 i Q_2) przechodzą z 0 na 0. Analogiczna jest interpretacja pozostałych wierszy Tab.2. Aby przerzutnik przeszedł ze stanu 0 na 1 to należy na jego wejście informacyjne podać takie wartości aby zmienił stan na 1. Jakże to mają być wartości, to dostarcza tablica przejść przerzutnika. Dla przerzutników JK i D odpowiednie tablice przedstawiono w tabeli niżej (* oznacza symbol obojętny-dowolny). Wynika z nich, iż aby spowodować takie przejście, to w przypadku przerzutnika JK należy przed wystąpieniem chwili t+1 na wejście J podać 1, na wejście K sygnał dowolny (0 lub 1). W przypadku przerzutnika D należy podać 1.

Q(t)	Q(t+1)	D(t)	J(t)	K(t)
0	0	0	0	*
0	1	1	1	*
1	0	0	*	1
1	1	1	*	0

Na podstawie tabeli przejść można dla tabeli Tab2 określić pobudzenia przedstawione w Tab.3

Tab.3

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

Tabelę tą należy czytać następująco (np. wiersz 1): aby uzyskać przejście ze stanu 000 w chwili t na stan 001 w chwili t+1 należy podać :

dla przerzutnika Q_2 : 0 na J, stan dowolny na K

dla przerzutnika Q_1 : 0 na J, stan dowolny na K

dla przerzutnika Q_0 : 1 na J, stan dowolny na K

W przypadku realizacji na przerzutnikach typu D należy podać:

0 na wejście D przerzutnika Q_2

0 na wejście D przerzutnika Q_1

1 na wejście D przerzutnika Q_0

Podczas syntezy tabele 1 i 2 pomija się od razu przystępując do wypełnienia tabeli 3.

Aby w takim razie zsyntezować sygnał pobudzenia dla wejść informacyjnego należy potraktować stan $Q_2Q_1Q_0$ w chwili t jako dane wejściowe, a wymagane pobudzenie jako funkcję którą należy zsyntezować. Możemy to zrobić metoda formalną (np. przedstawiając funkcję w postaci sum iloczynów) lub korzystając z metody tablic Karnough. Np. dla wejścia D_0 uzyskamy następujące wyrażenie $D_0 = \overline{Q_2}/Q_1/\overline{Q_0} + Q_2Q_1/\overline{Q_0} + Q_2/\overline{Q_1}Q_0$ (symbol / przed zmienną oznacza jej negację). Analogicznie będzie dla pozostałych wejść. Te uzyskane wyrażenia można dalej minimalizować metodą formalną lub od razu zastosować inną metodą np. tablic Karnough. W przypadku układu zbudowanego z przerzutników D te tablice będą wyglądały następująco (w miejsce nieistniejącego stanu 111 w Tab.3 wpisano symbol obojętny) :

Tabela dla przerz. Q_0

$Q_2Q_1 \backslash Q_0$	0	1
00	1	0
01	1	0
11	1	*
10	0	0

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

Tabela dla przerz. Q_1

$Q_2Q_1 \backslash Q_0$	0	1
00	0	1
01	1	1
11	0	*
10	0	0

$$D_1 = \overline{Q_2}Q_0 + Q_2Q_1$$

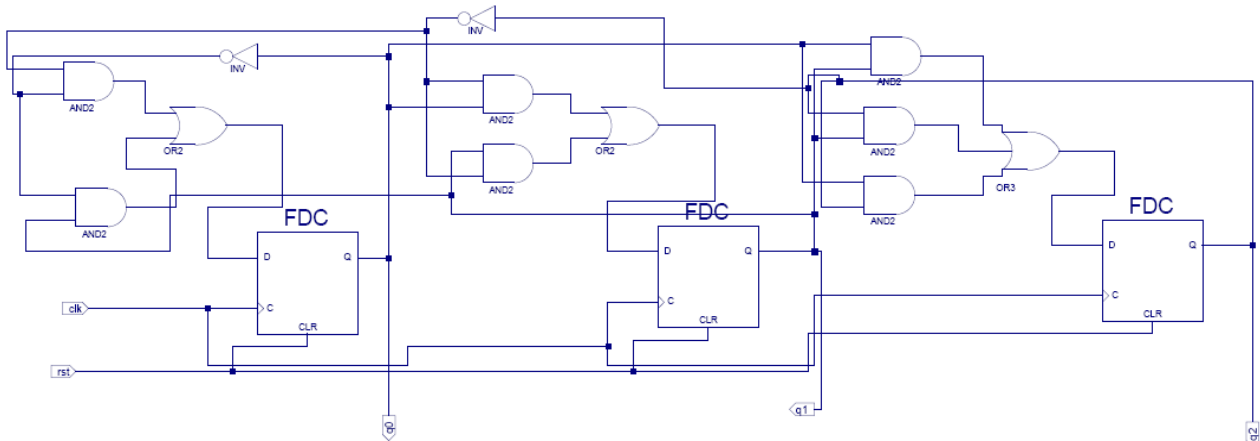
Tabela dla przerz. Q_2

$Q_2Q_1 \backslash Q_0$	0	1
00	0	0
01	0	1
11	1	*
10	0	1

$$D_2 = Q_1Q_0 + Q_2Q_1 + Q_2Q_0$$

Na podstawie poniższych równań można narysować schemat.

W przypadku przerzutników JK tabel będzie dwa razy więcej (bo osobno dla J i K), ale równania częściej wychodzą prostsze niż w przypadku przerzutnika D.



Rys.1 Zsyntezowany licznik na przerzutnikach typu D

Synteza liczników dwukierunkowych

Jest ona analogiczna do przedstawionej. Różnica polega na tym, iż nasza tabela 3 w stanie t zawiera dodatkowy sygnał kierunku - czyli stan bieżący określa stan wyjść $Q_2Q_1Q_0$ oraz kierunek. Ten sygnał pojawi się zatem w tabelach Karnough i w konsekwencji w równaniach po minimalizacji.