

Ćwiczenia laboratoryjne z Logiki Układów Cyfrowych

ćwiczenie 212

Temat: Automat asynchroniczny

1. Cel ćwiczenia

Celem ćwiczenia jest nabycie praktycznej umiejętności projektowania automatów asynchronicznych statycznych z uwzględnieniem metod eliminacji hazardu i wyścigów, realizacji technicznej tych układów oraz zbadanie wpływu hazardu i wyścigów na ich pracę.

2. Program ćwiczenia

1. Zapoznanie się ze zjawiskami hazardu i wyścigów występujących w układach logicznych oraz ich wpływem na pracę tych układów.
2. Poznanie metod projektowania automatów asynchronicznych.
3. Modelowanie schematów automatów asynchronicznych na zestawach UNILOG.
4. Sprawdzenie wpływu zjawiska hazardu na poprawność pracy automatu.

3. Wiadomości podstawowe

Automaty asynchroniczne należą do grupy automatów pozbawionych zewnętrznej organizacji czasu - nie występuje tu zewnętrzne taktowanie. Zmiany stanów wewnętrznych następują samoczynnie, bezpośrednio po wprowadzeniu sygnałów wejściowych.

Automaty asynchroniczne dzielą się na statyczne i dynamiczne. Styczne automaty wykorzystują do sterowania tylko poziomy sygnałów. Automaty te wykazują wiele zalet w stosunku do synchronicznych, między innymi możliwość budowy z prostszych i tańszych elementów. Jednakże brak zewnętrznego synchronizatora stwarza dodatkowe problemy podczas projektowania tych układów, które związane są z występowaniem hazardu i wyścigów.

W automatach asynchronicznych sygnały wejściowe oddziałują bezpośrednio na stan wewnętrzny automatu. Oznaczają przez: $x(t)$ - stan wejść w chwili t , $q(t)$ - stan wewnętrzny automatu w chwili t , $y(t)$ - stan wyjść w chwili t , dla automatu asynchronicznego zachodzi:

$$q(t + \tau) = \Psi[q(t), x(t)] \quad (1)$$

Wartość τ jest określona naturalnymi opóźnieniami wnoszonymi na drodze sygnałów przez elementy, z których zbudowany jest automat. Funkcja Ψ jest funkcją przejść automatu. Jeśli przez Φ oznaczymy funkcję wyjść, wówczas dla automatu typu Mealy obowiązuje:

$$y(t + \tau') = \Phi[q(t), x(t)] \quad (2)$$

oraz dla automatu Moore'a:

$$y(t + \tau') = \Phi[q(t)] \quad (3)$$

Wartość τ' jest, podobnie jak τ , pewnym opóźnieniem układów logicznych.

W przypadku automatów asynchronicznych wyróżniamy stany stabilne, trwające przez cały czas, w którym stan wejść się nie zmienia, oraz stany niestabilne występujące w momencie zmiany stanu wejść.

W celu zapewnienia prawidłowej pracy automatu asynchronicznego będziemy zakładać, że:

- niemożliwa jest zmiana dwóch sygnałów wejściowych równocześnie,
- dystans czasowy pomiędzy dwiema kolejnymi zmianami stanu wejść jest dłuższy od czasu ustalania się stanu wewnętrznego automatu.

4. Przykład syntezy abstrakcyjnej automatu

Kolejne etapy syntezy abstrakcyjnej automatu asynchronicznego obejmują:

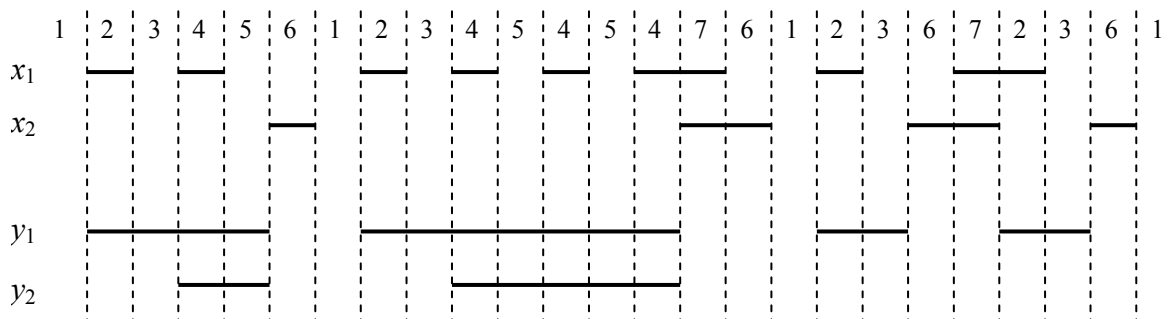
- określenie pierwotnej tabeli przejść i wyjść,
- skonstruowanie tabeli o minimalnej liczbie stanów wewnętrznych,
- kodowanie stanów z uwzględnieniem eliminacji zjawiska wyścigów.

Załóżmy, że należy zaprojektować automat w wersji Moore'a, który ma dwa wejścia w postaci niestabilizowanych przycisków x_1 i x_2 i dwa wyjścia y_1 i y_2 oraz działa w następujący sposób:

- pierwsze naciśnięcie przycisku x_1 powoduje $y_1 = 1$, drugie naciśnięcie powoduje $y_2 = 1$, lecz następne naciśnięcia nie powodują zmiany wyjść,
- każdorazowe naciśnięcie przycisku x_2 powoduje $y_1 = y_2 = 0$.

Etap I

Na podstawie opisu słownego rysujemy wykres czasowy jak na rys. 1, który podaje wszystkie istotne zależności między sygnałami wejściowymi a wyjściowymi. Na wykresie wyróżniamy kolejne stany całkowite, odpowiadające odcinkom czasu, w których sygnały wejściowe nie zmieniają się. Stany, które oznaczone zostały na rys. 1 liczbami naturalnymi, odpowiadają różnym kombinacjom sygnałów wejściowych i wyjściowych. Przykładowo, kombinacji $x_1 x_2 y_1 y_2$ równej 0000 odpowiada stan 1, kombinacji 1010 odpowiada stan 2, kombinacji 0010 stan 3 itd.



Rys. 1. Wykres czasowy dla automatu z przykładu

Etap II

Określiwszy liczbę stanów całkowitych, rysujemy pierwotną tabelę przejść, która dla każdego stanu całkowitego w chwili t wyznacza stan następny w chwili $t + \tau$. Analizując przypadki objęte wykresem czasowym, wypełniamy kratki wewnątrz tabeli przejść. Stany stabilne w tabeli wyróżniamy oznaczając je kółkami. Z wykresu czasowego wynika, że stan 1 przechodzi do stanu stabilnego ① pod wpływem sygnałów wejściowych $x_1 x_2 = 00$. Ze stanu 1 automat przechodzi do stanu 2 pod wpływem 10. W kratce odpowiadającej stanowi 1 i kombinacji wejściowej 11 piszemy kreskę (oznaczającą symbol nieokreślony), gdyż zakładamy niemożliwość jednoczesnej zmiany obu sygnałów wejściowych (z 00 na 11).

Po przeanalizowaniu wszystkich możliwych przejść otrzymujemy tabelę przejść automatu jak na rys. 2. Z prawej strony tabeli przejść wypisujemy sygnały wyjściowe odpowiadające kolejnym stanom całkowitym.

$x_1 x_2$ q	00	01	11	10	$y_1 y_2$
1	①	6	-	2	00
2	3	-	7	②	10
3	③	6	-	4	10
4	5	-	7	④	11
5	⑤	6	-	4	11
6	1	⑥	7	-	00
7	-	6	⑦	2	00

Rys. 2. Tabela przejść/wyjść dla automatu z przykładu

Etap III

Następnym etapem syntezy abstrakcyjnej jest minimalizacja pierwotnej tabeli przejść. Ponieważ automat ma być zaprojektowany w wersji Moore'a, należy znaleźć wszystkie pary stanów pseudorównoważnych. W czasie projektowania automatu w wersji Mealy należałoby odszukać pary stanów zgodnych [2, 3].

$x_1 x_2$ q	00	01	11	10	$y_1 y_2$
(6, 7)	①	①	①	2	00
	3	-	1	②	10
	③	1	-	4	10
(4, 5)	④	1	1	④	11

a) zminimalizowana

$x_1 x_2$ $Q_1 Q_2$	00	01	11	10	$y_1 y_2$
(1)	①①	①①	①①	01	00
(2)	10	-	00	①①	10
(3)	①①	00	-	11	10
(4)	11	①①	00	00	①①

b) przykładowo zakodowana

Rys. 3. Tabela przejść automatu

Dwa stany nazywamy pseudorównoważnymi, jeśli dla odpowiadających tym stanom wierszy w tabeli przejść spełnione są warunki:

- wyjścia odpowiadające obu wierszom są jednakowe,
- podczas dowolnej zmiany stanu wejść automat przechodzi z rozpatrywanych stanów do stanów jednakowych lub pseudorównoważnych.

Wiersze odpowiadające stanom pseudorównoważnym łączymy w ten sposób, że jeśli w

kolumnie jednego z łączonych wierszy był stan stabilny, to będzie on stabilny w wierszu połączonym.

Z tabeli na rys. 2 wynika, że pseudorównoważne są stany 1 i 6, 1 i 7 oraz 4 i 5. Wynika stąd, że stany 1, 6, 7 można zastąpić jednym stanem 1 a stany 4 i 5 innym stanem 4. Zminimalizowaną tabelę przejść z rys. 2 dla automatu w wersji Moore'a przedstawiono na rys. 3.

Etap IV

Następnym etapem syntezy jest zakodowanie zminimalizowanej tabeli przejść automatu, które gwarantuje poprawność jego pracy, eliminuje tzw. zjawisko wyścigów oraz minimalizuje złożoność struktury logicznej układu.

Zjawisko wyścigów

Rozpatrzmy zjawisko wyścigów w projektowanym automacie, zakładając, że tabela przejść (z rys. 3a) została zakodowana przykładowo tak jak na rys. 3b. Załóżmy, że automat znajduje się w stanie stabilnym ② tzn. $x_1 x_2 = 10$, $Q_1 Q_2 = 01$. Przypuśćmy, że stan wejść zmieni się z 10 na 00. Wówczas zgodnie z tabelą przejść automat powinien przejść do stanu ③, zakodowanego jako 10. Na skutek niejednakowych opóźnień wprowadzanych przez elementy, z których zbudowany jest automat, przejście ze stanu 01 do 10 może odbywać się dwójako:

$$01 \rightarrow 00 \rightarrow 10 \quad \text{lub} \quad 01 \rightarrow 11 \rightarrow 10.$$

W pierwszym przypadku sygnał Q_2 zmienił się szybciej niż sygnał Q_1 . W konsekwencji automat przechodząc przez stan "pośredni" 00, przy nadal trwającym stanie wejść 00, przejdzie do stanu stabilnego (00) (stan ①) zamiast do stanu (10) (stan ③). Zatem dla tak zakodowanej tabeli przejść automatu jak na rys. 3b występuje wyścig krytyczny, który może doprowadzić do błędnej pracy automatu.

Warunkiem wystarczającym do uniknięcia wyścigów krytycznych jest takie zakodowanie stanów, aby podczas każdej zmiany stanów zmieniał się co najwyżej jeden sygnał pamięci. Nie zawsze jednak takie zakodowanie jest możliwe.

Częstokroć dobre rezultaty kodowania uzyskujemy stosując rachunek podziałów [2]. Metoda ta pozwala nie tylko na uniknięcie wyścigów krytycznych, ale często prowadzi do uproszczenia układu. Na rys. 4 przedstawiono zakodowaną tabelę przejść automatu wolną już od wyścigów krytycznych.

	$x_1 x_2$	00	01	11	10	$y_1 y_2$
(1)	00	00	00	00	01	00
(2)	01	11	00	00	01	10
(3)	11	11	00	-	10	10
(4)	10	10	00	00	10	11

Rys. 4. Prawidłowo zakodowana tabela przejść automatu

Czasami zdarza się, że nie można zapewnić kodowania eliminującego zjawisko wyścigu krytycznego. Wówczas należy przeanalizować możliwość wprowadzenia tzw. przejścia cyklicznego lub, gdy nawet to jest niemożliwe, należy wprowadzić dodatkowy stan wewnętrzny [2].

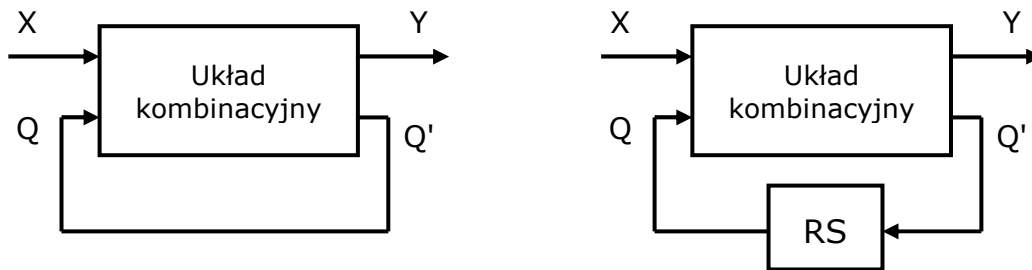
Etap V

4.1. Synteza strukturalna - eliminacja zjawiska hazardu

Strukturę automatu asynchronicznego, dla którego mamy prawidłowo zakodowaną tabelę przejść (tzn. z wyeliminowaniem zjawiska wyścigów) możemy zaprojektować dwoma sposobami:

1. Jako układ kombinacyjny ze sprzężeniami zwrotnymi,
2. Jako układ zawierający przerzutniki statyczne.

Schematy blokowe poszczególnych wersji automatu przedstawiono na rys. 5.



Rys. 5. Struktury automatu asynchronicznego

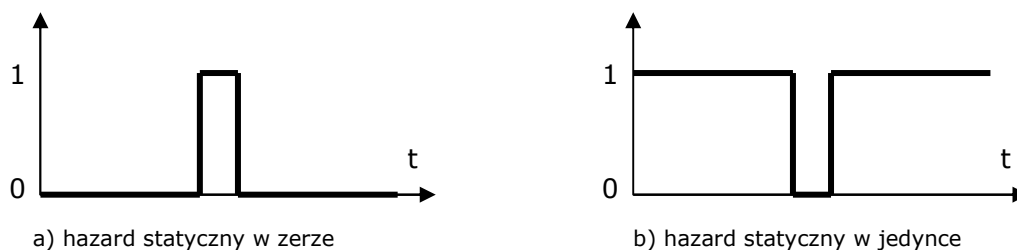
Automat o K stanach wewnętrznych musi zawierać k sprzężeń zwrotnych lub k przerzutników, gdzie k wyznacza się z zależności:

$$2^{k-1} < K \leq 2^k$$

4.1.1. Realizacja automatu jako układu kombinacyjnego ze sprzężeniami zwrotnymi

W torze sprzężenia zwrotnego układu kombinacyjnego muszą znajdować się elementy czynne (np. spełniające funkcje funktorów, elementy NOR lub NAND). Na obecnym etapie syntezy należy w taki sposób określić funkcje przejść automatu, aby wyeliminować ewentualne zjawisko hazardu w układzie kombinacyjnym. Hazard statyczny w układzie kombinacyjnym polega na krótkotrwałych zmianach sygnału wyjściowego podczas zmiany jednego z sygnałów wejściowych. Zobrazowano to na rysunku 6.

W układzie asynchronicznym ten krótkotrwały, błędny sygnał może być podtrzymany na skutek istnienia sprzężenia zwrotnego i spowodować tym samym błędne działanie automatu. Hazard statyczny łatwo jest rozpoznać w tabeli Karnaugh'a po tym, że sąsiadujące ze sobą grupy kratek biorące udział w minimalizacji funkcji stykają się na linii zmiany wartości jednego z argumentów.



Rys. 6. Hazard statyczny w układzie kombinacyjnym

Przypuścimy, że automat będziemy budować z elementów NOR. W celu określenia funkcji wzbudzeń i funkcji wyjść przedstawimy tabelę przejść z rys. 7 osobno dla sygnałów Q_1 i Q_2 (rys. 8). Oznaczenie Q_1' zostało użyte dla wyróżnienia sygnału na wyjściu, w rzeczywistości jest to ten sam sygnał, co sygnał wejściowy Q_1 (sprzężenie zwrotne).

	$x_1 x_2$				
$Q_1 Q_2$	00	01	11	10	$y_1 y_2$
(1)	00	00	00	01	00
(2)	01	11	00	01	10
(3)	11	11	00	-	10
(4)	10	10	00	10	11

Rys. 7. Tabela przejść automatu z rys. 4

		Q_1'				Q_2'					
$x_1 x_2$		00	01	11	10	$x_1 x_2$		00	01	11	10
$Q_1 Q_2$						$Q_1 Q_2$					
00		0	0	0	0	00	0	0	0	1	
01		1	0	0	0	01	1	0	0	1	
11		1	0	-	1	11	1	0	-	0	
10		1	0	0	1	10	0	0	0	0	

Rys. 8. Tabele przejść dla sygnałów Q_1' i Q_2'

Z tabel na rys. 8 otrzymujemy następujące zależności:

$$Q_1' = \overline{x_2}(Q_1 + Q_2)(Q_1 + \overline{x_1}) \quad Q_2' = \overline{x_2}(Q_2 + x_1)(\overline{Q_1} + \overline{x_1})$$

Zauważmy, że wszystkie łączone grupy krerek zazębiają się wzajemnie, a zatem nie występuje tu zjawisko hazardu. Ponieważ automat jest w wersji Moore'a a zatem już na podstawie tabeli z rys. 7 można podać następujące funkcje wyjść:

$$y_1 = Q_1 + Q_2 \quad y_2 = Q_1 \overline{Q_2}$$

Uzyskane w wyniku minimalizacji funkcje logiczne zostaną przekształcone tak, aby mogły być zrealizowane przy użyciu dwuwejściowych funkcji NOR:

$$Q_1' = \overline{x_2}(Q_1 + Q_2)(Q_1 + \overline{x_1}) = \overline{\overline{\overline{x_2}(Q_1 + Q_2)(Q_1 + \overline{x_1})}} = \overline{\overline{\overline{x_2} + \overline{Q_1 + Q_2} + \overline{Q_1 + \overline{x_1}}}}$$

$$Q_2' = \overline{x_2}(Q_2 + x_1)(\overline{Q_1} + \overline{x_1}) = \overline{\overline{\overline{\overline{x_2}(Q_2 + x_1)(\overline{Q_1} + \overline{x_1})}}} = \overline{\overline{\overline{x_2} + \overline{Q_2 + x_1} + \overline{\overline{Q_1} + \overline{x_1}}}}$$

$$y_1 = Q_1 + Q_2 = \overline{\overline{Q_1 + Q_2}}$$

$$y_2 = Q_1 \overline{Q_2} = \overline{\overline{Q_1 \overline{Q_2}}} = \overline{\overline{Q_1} + \overline{Q_2}}$$

Schemat logiczny automatu zbudowanego z dwuwejściowych elementów typu NOR pokazany został na rys 14.

4.1.2. Realizacja automatu z przerzutników asynchronicznych RS

Sposób określenia struktury automatu jest analogiczny jak w przypadku automatów synchronicznych. Wykorzystujemy do tego tabelę przejść automatu z rys. 8 oraz tabelę wzbudzeń dla przerzutników RS, z których ma być zbudowany automat. Tabelę wzbudzeń przerzutnika (poziom aktywny dla wejść R i S wysoki) pokazano na rys. 9.

$Q(t)$	$Q(t+1)$	$R(t)$	$S(t)$
0	0	-	0
0	1	0	1
1	0	1	0
1	1	0	-

Rys. 9. Tabela wzbudzeń przerzutnika RS

Na podstawie tabel z rysunków 8 i 9 tworzymy tabele dla poszczególnych wejść przerzutników (rys. 10), a na ich podstawie określimy funkcje wzbudzeń.

$x_1 x_2$	00	01	11	10
00	-	-	-	-
01	0	-	-	-
11	0	1	1	0
10	0	1	1	0

$x_1 x_2$	00	01	11	10
00	0	0	0	0
01	1	0	0	0
11	-	0	0	-
10	-	0	0	-

$x_1 x_2$	00	01	11	10
00	-	-	-	0
01	0	1	1	0
11	0	1	1	1
10	-	-	-	-

$x_1 x_2$	00	01	11	10
00	0	0	0	1
01	-	0	0	-
11	-	0	0	0
10	0	0	0	0

Rys. 10. Tabele wzbudzeń dla wejść przerzutników

Uzyskane w wyniku minimalizacji funkcje wzbudzeń są następujące:

$$R_1 = x_2 \qquad S_1 = Q_2 \overline{x_1} \overline{x_2}$$

$$R_2 = x_2 + Q_1 x_1 \qquad S_2 = \overline{Q_1} x_1 \overline{x_2}$$

Funkcje wyjść określamy bezpośrednio z tabeli przejść automatu. Są one identyczne jak w przypadku wersji automatu ze sprzężeniami zwrotnymi.

$$y_1 = Q_1 + Q_2 \qquad y_2 = Q_1 \overline{Q_2}$$

Funkcje logiczne po przekształceniu do postaci zawierającej tylko dwuwejściowe funkcje NOR uzyskują następującą postać:

$$R_1 = x_2$$

$$S_1 = \overline{Q_2} \overline{x_1} \overline{x_2} = \overline{Q_2} \overline{x_1} \overline{x_2} = \overline{Q_2 + x_1 + x_2} = \overline{Q_2} + \overline{x_1} + \overline{x_2}$$

$$R_2 = x_2 + Q_1 x_1 = x_2 + \overline{Q_1} x_1 = x_2 + \overline{Q_1} + x_1$$

$$S_2 = \overline{Q_1} x_1 \overline{x_2} = \overline{Q_1} x_1 \overline{x_2} = \overline{Q_1 + x_1 + x_2} = \overline{Q_1} + \overline{x_1} + \overline{x_2}$$

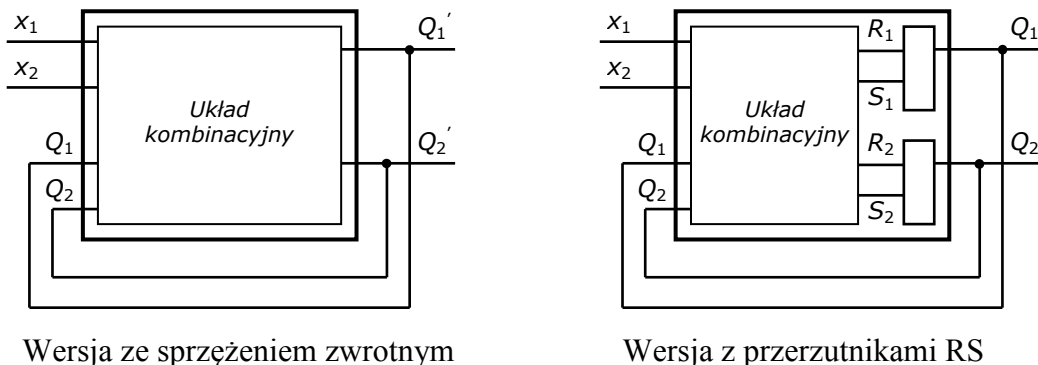
$$y_1 = Q_1 + Q_2 = \overline{Q_1} + \overline{Q_2}$$

$$y_2 = Q_1 \overline{Q_2} = \overline{Q_1} \overline{Q_2} = \overline{Q_1} + \overline{Q_2}$$

Schemat automatu na przerzutnikach RS, zbudowanego z dwuwejściowych elementów NOR przedstawiono na rys. 13.

4.2. Testowanie automatu

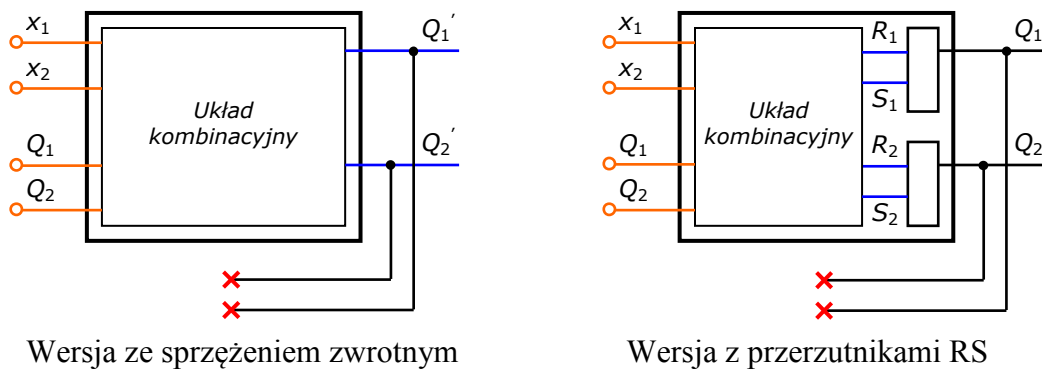
Uruchamianie automatu asynchronicznego wymaga odpowiedniego podejścia w związku z istnieniem sprzężeń zwrotnych, które utrudniają wykrycie przyczyn błędnego działania automatu. Sprzężenie zwrotne występuje w obu wariantach automatu, pokazane zostało to na rys. 11. W wariacie z przerzutnikami RS sprzężenie jest mniej widoczne, ale zmiany stanu przerzutników wpływają asynchronicznie na stan ich wejść RS, co może prowadzić do kolejnych zmian stanów.



Rys. 11. Sprzężenia zwrotne w automacie asynchronicznym

Jeśli zostaną popełnione błędy przy syntezie lub w czasie łączenia układu zachowanie automatu będzie odbiegać od spodziewanego. Może być obserwowany na przykład całkowity brak reakcji na zmianę sygnałów wejściowych x_1 i x_2 , nieprawidłowa zmiana stanów lub w najgorszym przypadku ciągle generowanie pewnej sekwencji stanów, bez osiągnięcia stanu stabilnego.

Jedynym skutecznym wyjściem w tej sytuacji jest dokonanie takich zmian w układzie, które umożliwią sprawdzenie poprawności działania układu kombinacyjnego. Potrzebna jest możliwość ustawienia wszystkich możliwych kombinacji sygnałów wejściowych testowanego układu czyli x_1 , x_2 , Q_1 , Q_2 aby sprawdzić zgodność funkcji z jej tabelą prawdy. Można to osiągnąć przez rozwarcie pętli sprzężeń zwrotnych i zadawanie sygnałów wejściowych Q_1 i Q_2 , podobnie jak x_1 , x_2 , bezpośrednio z przełączników. Układ umożliwiający przeprowadzenie takich testów pokazany został na rys. 12.



Rys. 12. Układ testowy z rozwartymi sprzężeniami zwrotnymi

Warto zwrócić uwagę na różnice w przebiegu testów z rozwartym sprzężeniem zwrotnym dla obu wariantów automatu. Wynikają one wprost z różnic w syntezy strukturalnej (jej rezultatem są inne funkcje logiczne). Dla wariantu z bezpośrednim sprzężeniem zwrotnym obserwowane powinny być bezpośrednio stany Q_1' i Q_2' , dla wariantu z przerzutnikami RS należy sprawdzić funkcje sterujące wejściami przerzutników czyli R_1 , S_1 , R_2 , S_2 .

Możliwość rozwarcia sprzężeń zwrotnych należy przewidzieć już w trakcie łączenia układu logicznego tak, aby nie wymagało to poważnych zmian w układzie i było możliwe przez przełączenie pojedynczych sygnałów. Aby to osiągnąć należy podzielić połączenia sygnałów Q_1' i Q_2' na dwie grupy:

- połączenia stałe czyli obserwacja stanu i sygnały wykorzystywane przez funkcje wyjść,
- połączenia związane ze sprzężeniem zwrotnym czyli prowadzące do wejść układu kombinacyjnego.

Dzięki temu możliwe będzie jednoczesne podłączenie wszystkich sygnałów z drugiej grupy do grupy pierwszej (tryb roboczy) lub do przełączników (tryb testowy według rys. 12).

5. Przebieg ćwiczenia

Ćwiczenie może być wykonane w dwóch wariantach, zależnie od wskazówek prowadzącego. Jeśli zostanie podany automat asynchroniczny do zaprojektowania, to synteza przedstawiona w instrukcji powinna służyć jako przykład, na którym można się oprzeć wykonując własny projekt. W przeciwnym razie należy uruchomić i przetestować automat asynchroniczny przedstawiony w instrukcji.

1. Zapoznanie się ze zjawiskami hazardu i wyścigów występującymi w układach logicznych.
2. Opanowanie metod syntezy automatu asynchronicznego z uwzględnieniem eliminacji zjawisk wyścigów i hazardu.
3. Zmontowanie na zestawie UNILOG dwóch układów automatów asynchronicznych, zgodnie ze schematami podanymi na rysunkach 13 i 14.
4. Sprawdzenie poprawności działania zmontowanych układów.
5. W celu zbadania wpływu zjawiska hazardu przeprowadzić jedną z podanych metod syntezy automatu dla celowo zakodowanych stanów wewnętrznych w sposób nieprawidłowy (np. dla tabeli z rys. 3b). Zmontować i sprawdzić działanie tak zaprojektowanego automatu obserwując występujące zjawisko hazardu.

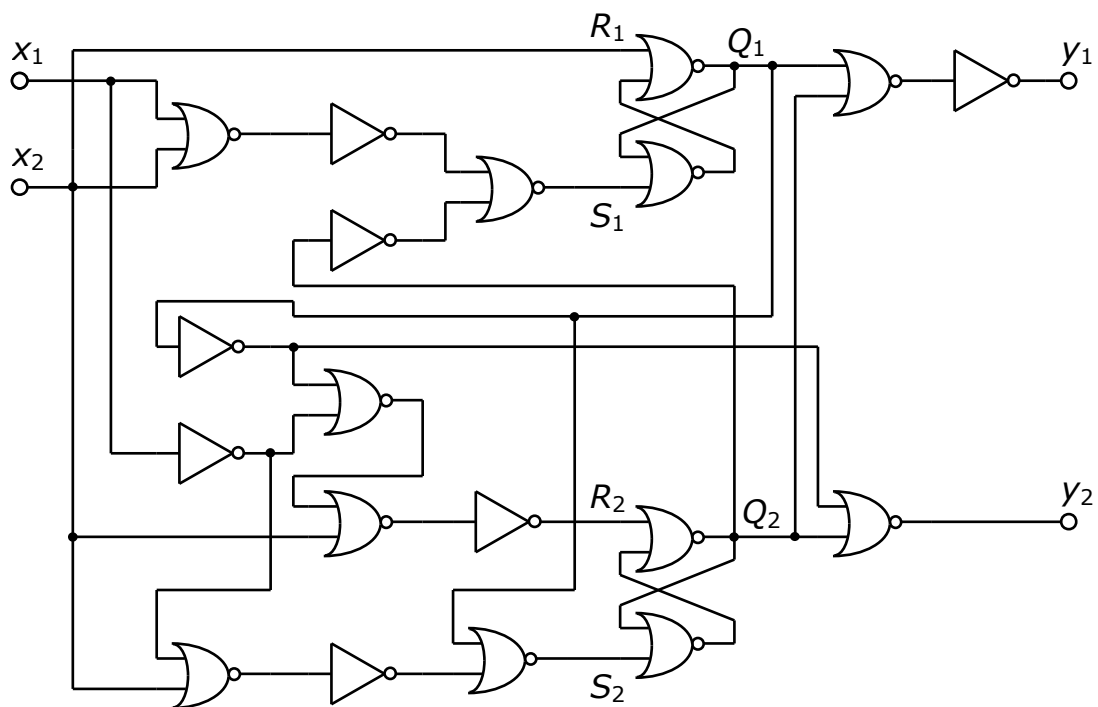
6. Sprawozdanie z ćwiczenia

W sprawozdaniu należy umieścić:

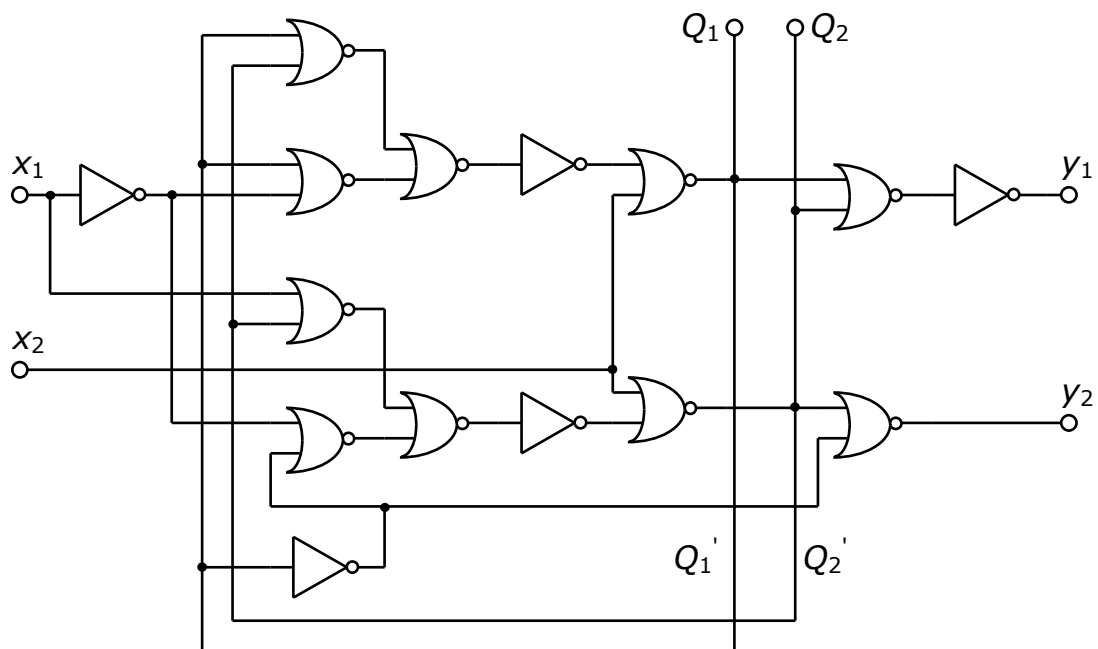
- temat i cel ćwiczenia,
- schematy zamodelowanych automatów,
- wyniki testowania automatów,
- wnioski z ćwiczenia.

7. Spis literatury

- [1]. Kazimierzczak J., Kluska J., Kaczmarek A.: Podstawy teorii automatów - Laboratorium, skrypt Politechniki Rzeszowskiej, Rzeszów 1984
- [2]. Misiurewicz P., Perkowski M.: Teoria automatów. Materiały pomocnicze do ćwiczeń i laboratorium, WPW, Warszawa 1976
- [3]. Traczyk W.: Układy cyfrowe automatyki, WNT, Warszawa 1976



Rys. 13. Schemat automatu asynchronicznego zbudowanego z przerzutników RS



Rys. 14. Schemat automatu asynchronicznego w wersji ze sprzężeniami zwrotnymi