Opracował: dr hab. inż. Jan Magott

KATEDRA INFORMATYKI TECHNICZNEJ

Ćwiczenia laboratoryjne z Logiki Układów Cyfrowych

ćwiczenie 207

Temat: Automaty Moore'a i Mealy

1. Cel ćwiczenia

Celem ćwiczenia jest zapoznanie studentów z dwoma podstawowymi kategoriami automatów oraz metodami transformacji automatu Moore'a w automat Mealy i odwrotnie.

2. Program ćwiczenia

1. Synteza automatów Moore'a i Mealy realizujących zadane przekształcenie
2. Transformacja automatu Moore'a w automat Mealy i odwrotnie

3. Problematyka ćwiczenia

Automaty Moore'a i Mealy są automatami z pamięcią realizującymi przekształcenie sekwencji (ciągów liter) wejściowych w sekwencje wyjściowe.

Automaty te rozpoczynają swoje działanie w stanie początkowym. Następnie, kolejno podawane litery alfabetu wejściowego powodują zmianę stanu automatu oraz generację liter alfabetu wyjściowego.

Badane kategorie automatów różnią się sposobem generacji liter alfabetu wyjściowego. W przypadku automatu Moore'a, litera generowana na wyjściu zależy od aktualnego stanu. Automat Mealy wytwarza literę na wyjściu na podstawie stanu, w którym została podana litera alfabetu wejściowego oraz na podstawie tej litery.

Porównajmy automaty Moore'a i Mealy realizujące identyczne przekształcenie zbioru sekwencji wejściowych w zbiór sekwencji wyjściowych. Ponadto zakładamy, że oba automaty są w postaci minimalnej. Przy podanych wymaganiach, automat Moore'a posiada nie mniejszą liczbę stanów niż automat Mealy.

4. Wiadomości podstawowe

4.1. Definicje automatu Moore'a i Mealy

Automatem skończonym nazywamy dyskretny przetwornik informacji przekształcający sekwencje wejściowe w sekwencje wyjściowe, w sposób z góry zadany. Zbiór Z liter alfabetu wejściowego, zbiór Y liter alfabetu wyjściowego oraz zbiór Q stanów wewnętrznych są zbiorami skończonymi – stąd nazwa automatu.

Automat pracuje w dyskretnej skali czasu, przy czym punkty tej skali mogą być ponumerowane kolejnymi liczbami naturalnymi. Symbolem $z(t) \in Z$ oznaczamy literę podaną na wejście automatu w punkcie t dyskretnej chwili czasu. W podobny sposób interpretowane są symbole $q(t) \in Q$ i $y(t) \in Y$.

Definicja 1

Automat skończony typu Mealy jest szóstką uporządkowaną:

$A = < Z, Q, Y, \Phi, \Psi, q_0 >$

gdzie:

Z – alfabet wejściowy,
Q – zbiór stanów,
Y – alfabet wyjściowy,
$\Phi : Z \text{ x } Q \rightarrow Q$ – funkcja przejść określająca zmiany stanów,
$\Psi : Z \text{ x } Q \rightarrow Y$ – funkcja wyjść wyznaczająca literę generowaną na wyjściu,
q_0 – stan początkowy.

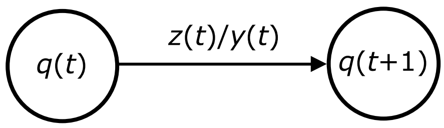
Funkcja przejść określa stan $q(t+1) = \Phi(z(t), q(t))$, który automat osiągnie po podaniu na jego wejście litery $z(t)$, jeśli litera ta została podana w stanie $q(t)$.

Funkcja wyjść definiuje literę alfabetu wyjściowego:

$y(t) = \Psi(z(t), q(t))$

generowaną na wyjściu, jeśli w stanie q(t) podano na wejście literę $z(t)$.

Interpretacja graficzna obu funkcji zawarta została na rysunku 1.



Rys. 1. Funkcje przejść i wyjść automatu Mealy

Definicja 2

Automat skończony typu Moore'a jest szóstka uporządkowaną:

$A = < Z, Q, Y, \Phi, \Psi, q_0 >$

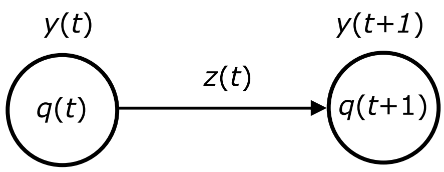
gdzie:

składowe Z, Q, Y, Φ oraz q_0 określone są identycznie jak dla automatu Mealy, zaś:

$\Psi : Q \rightarrow Y$ jest funkcją wyjść.

Funkcja wyjść określa literę $y(t) = \Psi(q(t))$ generowaną przez automat po osiągnięciu stanu $q(t)$.

Funkcje przejść i wyjść automatu Moore’a zilustrowane są na rysunku 2.



Rys. 2. Funkcje przejść i wyjść automatu Moore'a

4.2. Transformacja między automatami Moore'a i Mealy

Niech Z^* będzie zbiorem wszystkich sekwencji skończonej długości utworzonych z liter alfabetu Z wraz ze słowem pustym.

Symbolem $\overline{\Psi}$ oznaczamy uogólnioną funkcję wyjść automatu $\overline{\Psi} : Z^* \text{ x } Q \to Y^*$. Wartość tej funkcji $\overline{\Psi}(\overline{z}, q)$ jest sekwencją wyjściową $\overline{y} \in Y^*$ generowaną w wyniku podania sekwencji $\overline{z} \in Z^*$ na wejście automatu znajdującego się w stanie q.

Definicja 3

Dwa automaty $A_1 = < Z, Q_1, Y, \Phi_1, \Psi_1, q_{01} >$ oraz $A_2 = < Z, Q_2, Y, \Phi_2, \Psi_2, q_{02} >$ nazywamy równoważnymi, jeśli dla każdej sekwencji $\overline{z} \in Z^*$ spełniona jest równość:

$$\overline{\Psi}_1(\overline{z}, q_{01}) = \overline{\Psi}_2(\overline{z}, q_{02})$$

Twierdzenie 1

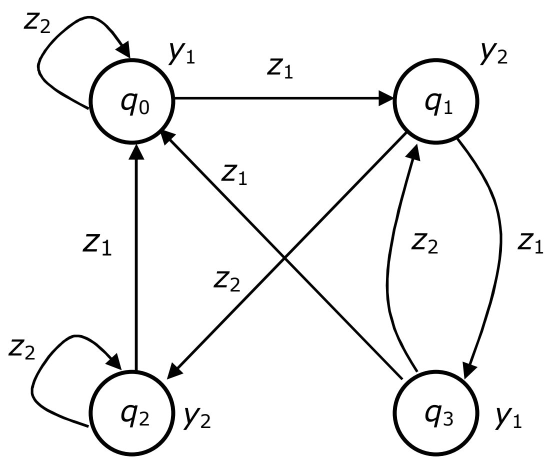
Dla danego automatu Moore'a $A_1 = < Z, Q_1, Y, \Phi_1, \Psi_1, q_{01} >$ istnieje równoważny automat Mealy $A_2 = < Z, Q_2, Y, \Phi_2, \Psi_2, q_{02} >$ taki, że:

$$Q_2 = Q_1, \quad \Phi_2 = \Phi_1, \quad \Psi_2(z, q) = \Psi_1(\Phi_1(z, q)), \quad q_{02} = q_{01}.$$

Transformację automatu Moore'a w automat Mealy zobrazujemy postacią graficzną.

Przykład 1

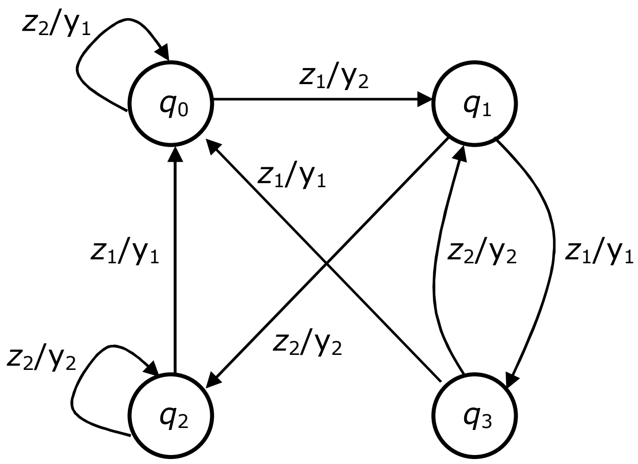
Niech dany będzie automat Moore'a, jak na rysunku 3.



Rys. 3. Automat Moore'a

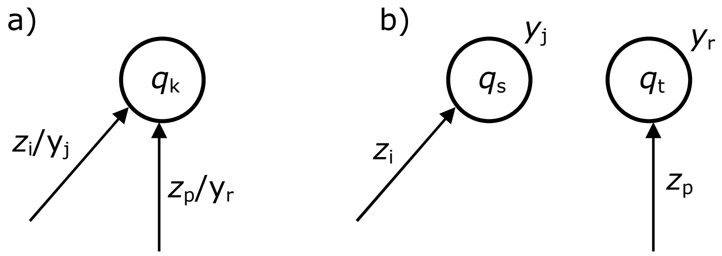
Równoważny mu automat Mealy pokazany jest na rysunku 4. Przeanalizujmy przykładowe wartości funkcji przejść i wyjść obu automatów.

Niech w stanie q_0 automatu Moore'a, na wejściu pojawi się litera z_1. W tym przypadku dla funkcji przejść mamy $\Phi_1(z_1, q_0) = q_1$. Zatem $\Phi_2(z_1, q_0) = q_1$, ponadto dla funkcji wyjść prawdziwa jest równość $\Psi_1(q_1) = y_2$, a więc $\Psi_2(z_1, q_0) = y_2$.



Rys. 4. Automat Mealy równoważny automatowi Moore'a z rys. 3

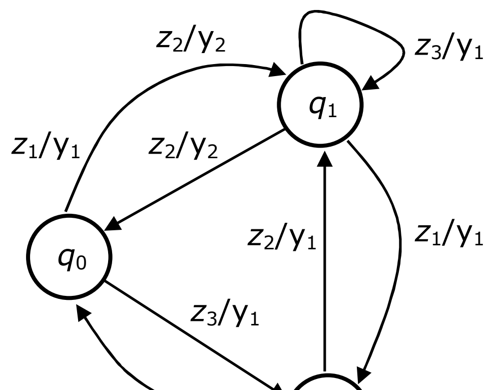
W przypadku transformacji automatu Mealy w automat Moore'a często wymagane jest zwiększenie liczby stanów. Rozważmy fragment grafu automatu Mealy ilustrowany rysunkiem 5a).



Rys. 5. Transformacja stanu automatu Mealy

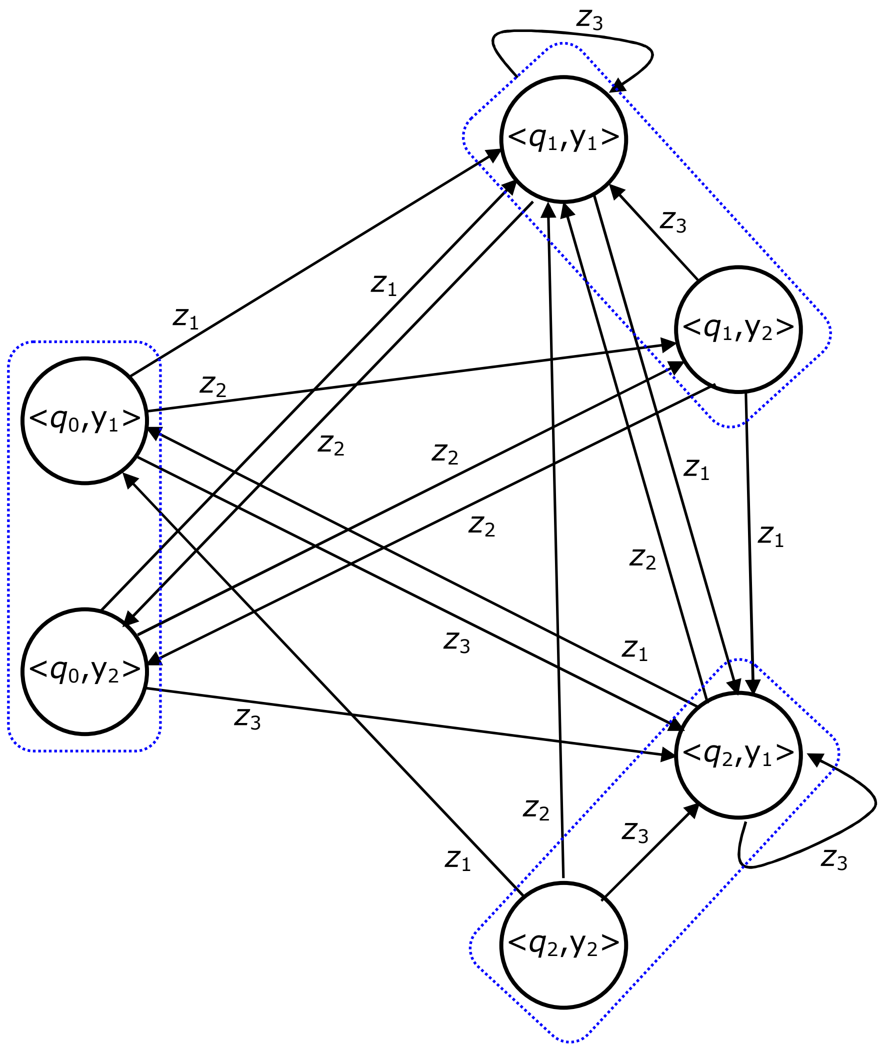
Stanowi q_k automatu Mealy nie można przypisać tylko jednego stanu automatu Moore'a, bowiem stan taki musiałby po osiągnięciu go w wyniku podania na wejście litery z_i generować literę y_j, natomiast po osiągnięciu tego stanu poprzez podanie litery z_p musiałby generować literę y_r. W tym przypadku stan q_k automatu Mealy zostanie odwzorowany w dwa stany automatu Moore'a odpowiadające parom uporządkowanym $<q_k, y_j>$ oraz $<q_k, y_r>$. Stany te można oznaczyć symbolami q_s i q_t - rysunek 5b).

Przykład 2



Rys. 6. Automat Mealy

Wynikiem transformacji automatu Mealy z rysunku 6 jest automat Moore'a pokazany na rysunku 7.



Rys. 7. Automat Moore'a równoważny automatowi Mealy z rys. 6

Stan przyporządkowany parze $<q_2, y_2>$ może być wyeliminowany, bowiem nie może on być osiągnięty z żadnego ze stanów odpowiadających parom $<q_0, y_1>$, $<q_0, y_2>$.

Twierdzenie 2

Dla danego automatu Mealy $A_1 = < Z, Q_1, Y, \Phi_1, \Psi_1, q_{01} >$ istnieje równoważny mu automat Moore'a $A_2 = < Z, Q_2, Y, \Phi_2, \Psi_2, q_{02} >$ zdefiniowany następująco:

$Q_2 = Q_1 \text{ x } Y$,

$\Phi_2 : Z \text{ x } Q_2 \rightarrow Q_2$ taka, że $\Phi_2(z, <q, y>) = < \Phi_1(z, q), \Psi_1(z, q)>$, gdzie $q \in Q_1$,

$\Psi_2 : Q_1 \text{ x } Y \rightarrow Y$ taka, że $\Psi_2(<q, y>) = y$,

q_{02} jest jednym ze stanów $<q_{01}, y>$, gdzie $y \in Y$.

4.3. Minimalizacja automatu

W celu uzyskania automatu z minimalną liczbą stanów, należy wykryć stany równoważne i zastąpić je jednym stanem.

Każdy z rozpatrywanych dotąd automatów charakteryzuje się tym, że w każdym z jego stanów może pojawić się na wejściu każda z liter alfabetu wejściowego. Istnieją automaty, dla których w pewnych stanach na wejście podane mogą być tylko litery z podzbioru Z. Jeśli na wejściu automatu w stanie q_i nie może być podana litera z_j, wówczas stosujemy następujące oznaczenie dla funkcji przejść $\Phi(z_j, q_j) = –$. Funkcja wyjść automatu Mealy zadana jest wówczas równością $\Psi(z_j, q_j) = –$.

Rozważmy sytuację, w której dla automatu będącego w stanie q_i, po podaniu na wejście litery z_j, następuje zmiana stanu zgodnie z funkcją przejść $\Phi(z_j, q_i) = q_k$, natomiast generowana na wyjściu litera może być dowolna. W tym przypadku dla automatu Mealy stosujemy oznaczenie $\Psi(z_j, q_i) = –$, natomiast dla automatu Moore'a $\Psi(q_k) = –$.

Definicja 4

Dwa stany q_i, q_j automatu $A = < Z, Q, Y, \Phi, \Psi, q_0 >$ z uogólnioną funkcją wyjść $\overline{\Psi}$ nazywamy równoważnymi, jeśli dla każdej sekwencji wejściowej $z \in Z^*$ prawdziwa jest równość:

$$\overline{\Psi}(z, q_i) = \overline{\Psi}(z, q_j),$$

przy czym symbol "–" w sekwencji wyjściowej może być potraktowany jako równy dowolnej literze alfabetu wyjściowego Y.

Podobnie wartość $\Phi(z, q_k) = –$ funkcji przejść może być interpretowana jako równa dowolnemu stanowi $q \in Q$.

W procesie minimalizacji automatu wykorzystujemy własności, które obecnie przedstawimy.

Rozważmy dwa stany q_i, q_j automatu A. Niech dla każdej litery $z_k \in Z$ spełnione będą równości:

$$\Phi(z_k, q_i) = \Phi(z_k, q_j) \tag{1}$$
$$\Psi(z_k, q_i) = \Psi(z_k, q_j) \tag{2a}$$
$$\Psi(q_i) = \Psi(q_j) \tag{2b}$$

z uwzględnieniem swobody w interpretacji symbolu "–". Wyrażenie (2a) dotyczy automatu Mealy, natomiast wyrażenie (2b), automatu Moore'a.

Warunki (1), (2a) lub (2b) są wystarczające dla równoważności stanów q_i, q_j. Warunki te nie są jednak warunkami koniecznymi.

W warunku koniecznym słabiej sformułowane są wymagania nałożone na funkcję przejść. W warunku tym wartość funkcji Φ muszą spełniać wyrażenie (1) lub stany $\Phi(z_k, q_i)$ i $\Phi(z_k, q_j)$ powinny być stanami równoważnymi.

Możemy zatem sformułować dla funkcji przejść słabszy warunek równoważności stanów niż wymaganie (1). Załóżmy bowiem, że funkcja przejść dla stanów q_i, q_j, dla każdego $z_k \in Z$ spełnia alternatywnie jeden z trzech warunków:

$$\text{warunek (1)} \tag{3}$$
$$\Phi(z_k, q_i) = q_i \quad \text{i} \quad \Phi(z_k, q_j) = q_j \tag{4}$$
$$\Phi(z_k, q_i) = q_j \quad \text{i} \quad \Phi(z_k, q_j) = q_i \tag{5}$$

Wyrażenia (4), (5) wskazują, że wymaganiem dla równoważności stanów q_i, q_j jest równoważność ich stanów następnych. Stąd wyrażenia (3), (4), (5) określają słabszy warunek równoważności niż warunek (1).

Proces wykrywania stanów równoważnych prześledźmy na przykładzie.

Przykład 3

Niech dane będą funkcje przejść i wyjść automatu Mealy opisane tabelą 1.

Tabela 1

q \ z	00	01	11	10
0	1, 1	5, –	—	2, 1
1	4, 1	1, 0	3, 0	4, –
2	3, –	1, 1	4, 1	0, –
3	—	5, 0	5, 0	4, 1
4	2, 0	3, 1	3, 0	5, 0
5	4, –	3, –	5, 0	—

Obecnie przedstawimy znaczenie poszczególnych elementów tablicy. Na przykład, gdy automat znajduje się w stanie 4 i otrzymuje na wejściu literę 01, przechodzi do stanu 3, generując na wyjściu literę 1. Na wejściu automatu znajdującego się w stanie 3 nie może być podana litera 00. W tym przypadku, w celu uproszczenia zapisów, w odpowiedniej kratce tabeli umieszczany jest symbol "—" zamiast "–, –". Automat będąc w stanie 5 i otrzymując na wejściu literę 01 może generować dowolną literę wyjściową.

Szukając stanów równoważnych porównujemy ze sobą wszystkie pary wierszy tabeli.

Dwa wiersze opisujące stany q_i, q_j możemy zastąpić jednym, jeśli dla każdej litery $z_r \in Z$, ich składowe q_k, y_p znajdujące się w tej samej kolumnie (odpowiadające tej samej literze z_r alfabetu wejściowego) spełniają jeden z warunków:

1. obie składowe są równe sobie:

   $q_p, y_s = q_r, y_t$ co oznacza, że $q_p = q_r$ i $y_s = y_t$,

2. jedna ze składowych jest "—" (czyli, równa "–, –") a druga może być dowolna,
3. jedna ze składowych równa jest q_p, –, natomiast druga q_p, y_k dla dowolnego y_k,
4. jedna ze składowych jest postaci q_i, y_k, natomiast druga q_j, y_k.

Jeśli zastąpimy dwa wiersze dla stanów q_i, q_j wierszem q_i, to w tablicy pozostawimy wiersz q_i (usuwając q_j) oraz pozycje, w których występował stan q_j zastępujemy stanem q_i. Dodatkowo dla czytelności, przy wierszu opisanym stanem q_i pozostawiamy oznaczenie (q_i, q_j) wskazujące na rolę stanu q_i.

Proces minimalizacji obrazują tabele 2, 3 i 4.

Tabela 2

	q \ z	00	01	11	10
	0	1, 1	3, –	—	2, 1
	1	4, 1	1, 0	3, 0	4, –
	2	3, –	1, 1	4, 1	0, –
(3, 5)	3	4, –	3, 0	3, 0	4, 1
	4	2, 0	3, 1	3, 0	3, 0

Tabela 3

	q \ z	00	01	11	10
	0	1, 1	1, –	—	2, 1
(1, 3, 5)	1	4, 1	1, 0	1, 0	4, 1
	2	1, –	1, 1	4, 1	0, –
	4	2, 0	1, 1	1, 0	1, 0

Tabela 4

	q \ z	00	01	11	10
(0, 2)	0	1, 1	1, 1	4 ,1	0, 1
(1, 3, 5)	1	4, 1	1, 0	1, 0	4, 1
	4	2, 0	1, 1	1, 0	1, 0

Wynikowy automat ma trzy stany zamiast sześciu stanów automatu pierwotnego.

5. Przebieg ćwiczenia

Zadania do wykonania podane są poniżej ale mogą być również podane indywidualnie przez prowadzącego. Dla automatów zaprojektowanych na poziomie abstrakcyjnym należy również wykonać syntezę strukturalną, wykorzystując zadany typ przerzutników. Zaprojektowane układy należy uruchomić i przetestować na zestawach UNILOG.

6. Zadania do wykonania

Ćwiczenie jest podzielone na cztery moduły, z których każdy może być wykonany niezależnie.

Moduł A

1. Zaprojektować automat Moore’a będący sumatorem szeregowym liczb dwójkowych.
2. Zrealizować automat i sprawdzić jego działanie.
3. Podać równoważny automat Mealy i porównać obydwa modele.
4. Analogiczne czynności przeprowadzić dla subtraktora szeregowego z tym, że najpierw zaprojektować automat Mealy, z potem podać równoważny automat Moore’a.

Moduł B

1. Zaprojektować i zrealizować na stanowisku automat Mealy, który:
   - po wprowadzeniu litery X_1 wyprowadza literę Y_1 niezależnie od następnych liter,
   - po literze X_2 i co najmniej jeszcze jednej dowolnej literze, literę Y_2,
   - po literze X_3 i co najmniej dwu dowolnych liter, literę Y_3,
   - jako czwarte potraktować fakt nie wyprowadzania żadnej litery spośród Y_1, Y_2, Y_3.
2. Sprawdzić poprawność realizacji wyżej wymienionego algorytmu.
3. Podać równoważny automat Moore’a.

Moduł C

1. Zadany automat Moore’a w formie tablicy przejść i wyjść przekształcić w automat Mealy.
2. Wykonać model automatu Mealy i sprawdzić poprawność translacji słów wejściowych w słowa wyjściowe.
3. W analogiczny sposób dokonać zmiany automatu Mealy na model Moore’a.

Moduł D

1. Zaproponować algorytm "otwierania zamka" oraz sygnalizacji alarmu (włamania) w przypadku wprowadzenia nieprawidłowej kombinacji liter wejściowych. Podać tablicę przejść i wyjść oraz graf automatu.
2. Wykonać model "zamka".
3. Dokonać prób "otwarcia" przez osoby nie znające prawidłowej kombinacji otwierającej.

7. Literatura

[1]. Bromirski J.: Teoria automatów, WNT, Warszawa, 1969
[2]. Majewski W., Albicki A.: Algebraiczna teoria automatów, WNT, Warszawa, 1980
[3]. Traczyk W.: Układy cyfrowe automatyki, WNT, Warszawa, 1974