

Ćwiczenia laboratoryjne z Logiki Układów Cyfrowych

ćwiczenie 209

Temat: Komputerowa analiza automatów skończonych

1. Cel ćwiczenia

Celem ćwiczenia jest opanowanie umiejętności przeprowadzania analizy automatu skończonego w zakresie identyfikacji grafu automatu przy pomocy komputera.

2. Program ćwiczenia

1. Zapoznanie się z metodą analizy automatu, związaną z określeniem grafu przejść automatu o nieznanym działaniu.
2. Testowanie programowego wariantu automatu Moore'a sekwencją sygnałów wejściowych.
3. Opracowanie wyników testowania automatu.
4. Określenie na podstawie wyników testowania grafu przejść analizowanego automatu.
5. Opracowanie sprawozdania z ćwiczenia.

3. Problematyka ćwiczenia

Analiza automatów w zakresie identyfikacji grafu przejść ma duże praktyczne zastosowanie zarówno na etapie projektowania, jak i eksploatacji dyskretnych przetworników informacji. Najczęściej jednak analizę tę przeprowadzamy wtedy, kiedy chcemy określić działanie automatu, którego graf przejść nie jest znany oraz wtedy, gdy zbudowaliśmy automat i chcemy sprawdzić, czy działa on zgodnie z zadaniem algorytmem. W prezentowanym ćwiczeniu obiektem analizy jest automat typu Moore'a, którego graf nie jest określony, znany jest jednak zbiór sygnałów wejściowych i wyjściowych. Aby określić graf automatu, należy podać na jego wejście odpowiednią sekwencję sygnałów wejściowych i dla każdego z tych sygnałów odnotować sygnał pojawiający się na wyjściu automatu.

Znając sekwencję sygnałów wejściowych i odpowiadającą im sekwencję sygnałów wyjściowych, należy określić stany wewnętrzne automatu i możliwe przejścia między stanami, czyli graf automatu. Na podstawie pojedynczego sygnału wyjściowego nie można określić przyporządkowanego mu stanu wewnętrznego. Problem jest wystarczająco złożony, ponieważ w automacie Moore'a ten sam sygnał wyjściowy automatu może być przyporządkowany kilku stanom wewnętrznym. Rozwiązanie problemu sprowadza się do określenia takich związków między elementami sekwencji sygnałów wejściowych i wyjściowych, na podstawie których można jednoznacznie określić graf automatu. Możliwa do przyjęcia i stosowana w niniejszym ćwiczeniu metoda analizy automatu przedstawiona zostanie poniżej. W ćwiczeniu przyjmuje się, że działanie automatu realizowane jest programowo. Analiza sprowadza się więc do wprowadzania do komputera odpowiedniej sekwencji sygnałów wejściowych i obserwacji na ekranie wyników testowania.

4. Wiadomości podstawowe

4.1. Metoda analizy automatu w zakresie identyfikacji grafu automatu

W celu rozpoznania stanów wewnętrznych automatu o nieznanym grafie przejść powinna być ściśle określona kolejność podawania sygnałów wejściowych na wejście automatu. Kolejność ta określana jest w trakcie testowania automatu, gdyż każdy wyraz sekwencji sygnałów wejściowych jest funkcją poprzednich wyrazów, jak również odpowiadających im sygnałów na wyjściu automatu.

Formułując algorytm analizy automatu, należy określić dwie podstawowe reguły, a więc regułę, według której będzie tworzona sekwencja sygnałów wejściowych oraz regułę, według której zapisywane będą wyniki testowania automatu w takiej postaci analitycznej, która byłaby jednoznaczna z grafem automatu.

Pierwszą regułę można uzyskać przekształcając graf automatu do postaci takiego grafu o strukturze drzewiastej, który jednoznacznie wskazywałby na kolejność sygnałów wejściowych w sekwencji testującej automat.

Druga reguła wynika z faktu, że każdy graf automatu możemy przedstawić w postaci odpowiedniego wyrażenia symbolicznego, umożliwiającego transformację odwrotną, czyli przejście z wyrażenia symbolicznego na graf.

Postacią takiego grafu automatu, który zgodnie z regułą pierwszą umożliwiłby wybór odpowiedniej sekwencji sygnałów wejściowych testujących automat, jest "drzewo z pętlami". Charakteryzuje się ono tym, że każda ścieżka drzewa reprezentuje sobą sekwencję składową w sekwencji sygnałów wejściowych testujących automat.

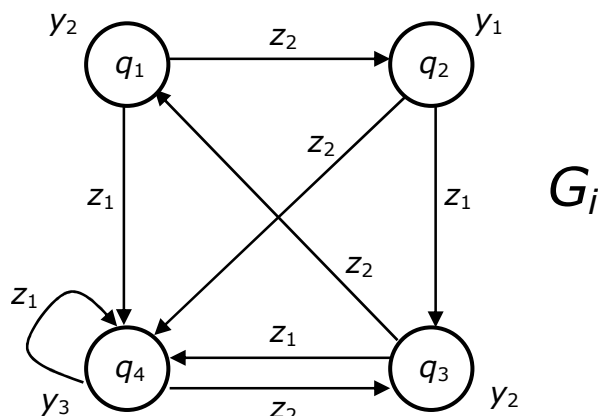
Charakterystyczny sposób budowy wyrażenia symbolicznego reprezentującego "drzewo z pętlami" oraz własności struktury takiego grafu, jako odmiennej postaci grafu automatu, umożliwiają zapisywanie wyników testowania automatu bezpośrednio w postaci wyrażenia symbolicznego reprezentującego "drzewo z pętlami". Mając takie wyrażenie możemy je przekształcić na wyrażenie opisujące graf automatu a następnie na jego podstawie narysować graf automatu. Formalna interpretacja tych przekształceń podana zostanie dalej.

4.2. Operacje na wyrażeniach symbolicznych reprezentujących grafy

Aby można było przeprowadzić analizę automatu według scharakteryzowanej wyżej metody, należy znać sposób budowy wyrażenia symbolicznego reprezentującego graf nazywany tutaj "drzewem z pętlami". Z kolei mając takie wyrażenie należy umieć je przekształcić na wyrażenie symboliczne reprezentujące graf automatu. Reguły takiego przekształcenia można określić tylko wtedy, gdy się zna transformację odwrotną, tj. przejście z wyrażenia reprezentującego graf automatu na wyrażenie symboliczne reprezentujące "drzewo z pętlami". Stąd też w pierwszej kolejności omówiony zostanie sposób przejścia z wyrażenia symbolicznego opisującego graf automatu na odpowiadające mu wyrażenie reprezentujące "drzewo z pętlami".

Przykładowy graf automatu Moore'a przedstawiony został na rys.1. Wyrażenie symboliczne reprezentujące ten graf, przy założeniu że stanem początkowym jest q_2 , ma następującą postać:

$$G_i^+ = {}^0(q_2^1(z_1q_3^2(z_1q_4^3(z_2q_3, z_1q_4)^3, z_2q_1^3(z_2q_2, z_1q_4)^3)^2, z_2q_4)^1)^0 \quad (1)$$



Rys. 1. Przykładowy graf automatu Moore'a

Dla pełnej reprezentacji grafu automatu Moore'a wyrażeniu (1) przyporządkowana jest tabela \bar{T}_i , opisująca funkcję wyjść:

Tabela \bar{T}_i

q	y
q_1	y_2
q_2	y_1
q_3	y_2
q_4	y_3

Wyrażenie G_i^+ możemy przekształcić do postaci takiego wyrażenia G_i^{++} , które będzie reprezentować graf d_i o strukturze drzewiastej, będący odpowiednikiem grafu G_i z rys. 1. Przed omówieniem reguł takiego przekształcenia zwróćmy uwagę na pewną cechę charakterystyczną wyrażenia G_i^+ . Cechą tą jest to, że między wyrazami typu q_r tego wyrażenia występuje ściśle określona podległość. Jeżeli na przykład rozpatrujemy człon o postaci $q_r^k(\dots)^k$, to wszystkie elementy q_i , zarówno dla $i \neq r$ jak i dla $i = r$, występujące wewnątrz nawiasu podlegają pod element q_r stojący przed nawiasem.

Każdy element q_r ma zawsze przyporządkowane sobie elementy q_i jemu podległe. Szczególnym przypadkiem takiego przyporządkowania jest to, że element q_r może sam sobie podlegać. Aby określić, co podlega pod element q_r w ciągu typu G_i^+ należy odszukać taki człon $^k(\dots)^k$, przed którym stoi rozpatrywany element q_r . Powyższe uwagi o relacjach podporządkowania w ciągu G_i^+ jednych elementów drugim ułatwią nam zrozumienie zasady przekształcenia wyrażenia G_i^+ na wyrażenie opisujące graf o innej strukturze, niż graf przedstawiony na rys. 1.

Przekształcenie wyrażenia G_i^+ na wyrażenie opisujące dendryt z cyklami d_i^* , równoważny grafowi G_i , sprowadza się do znalezienia w ciągu G_i^+ takich wyrazów typu q_r , które mogłyby być potraktowane jako wierzchołki końcowe dendrytu d_i^* . Korzystamy przy tym z trzech następujących warunków.

Warunek W0

Wierzchołkami końcowymi dendrytu d_i^\bullet mogą być, przy spełnieniu odpowiednich warunków, te elementy q_r , za którymi w wyrażeniu G_i^+ stoi przecinek lub nawias zamykający. Oznacza to, że dany element q_r wystąpił już w wyrażeniu G_i^+ na wcześniejszej pozycji.

Warunek W1

Jeśli element q_r , za którym wystąpił przecinek lub nawias zamykający, podlega pod wyraz oznaczony tym samym symbolem q_r lecz występujący na wcześniejszej pozycji ciągu G_i^+ , wówczas dany element q_r traktujemy jako wierzchołek końcowy dendrytu d_i^\bullet i oznaczamy dodatkowo indeksem górnym.

Warunek W2

Jeśli element q_r wymieniony w warunku W1 nie podlega pod wyraz oznaczony w ciągu G_i^+ tym samym symbolem q_r , wówczas element ten nie może być wierzchołkiem końcowym. W tym przypadku za rozpatrywanym elementem q_r znajdującym się w członie $^k(\dots q_r \dots)^k$ zapisuje się człon $^{k+1}(\dots)^{k+1}$ zawierający wszystkie te elementy q_i , które podlegają pod dany element q_r . Względem elementów q_i znajdujących się w dopisanym członie $^{k+1}(\dots)^{k+1}$ stosuje się w dalszym ciągu warunki W0, W1, W2 aż do momentu, dopóki nie zostaną określone te elementy q_i , które reprezentują wierzchołki końcowe dendrytu d_i^\bullet .

Przestrzegając przedstawionych wyżej warunków przy przekształceniu ciągu G_i^+ , zadanego wyrażeniem (1), otrzymamy następujący ciąg:

$$G_i^{++} = {}^0(q_2^1(z_1q_3^2(z_1q_4^3(z_2q_3', z_1q_4')^3, z_2q_1^3(z_2q_2', z_1q_4^4(z_2q_3', z_1q_4')^4)^3)^2, \quad (2)$$

$$z_2q_4^2(z_2q_3^3(z_1q_4', z_2q_1^4(z_2q_2', z_1q_4')^4)^3, z_1q_4'^2)^1)^0$$

W automacie Moore'a każdemu elementowi $q_r \in Q$ przyporządkowany jest odpowiedni element $y_i \in Y$, w rozpatrywanym przykładzie przyporządkowanie to ma następującą postać:

$$q_1 \leftrightarrow y_2, \quad q_2 \leftrightarrow y_1, \quad q_3 \leftrightarrow y_2, \quad q_4 \leftrightarrow y_3, \quad q_2' \leftrightarrow y_1', \quad q_3' \leftrightarrow y_2', \quad q_4' \leftrightarrow y_3'.$$

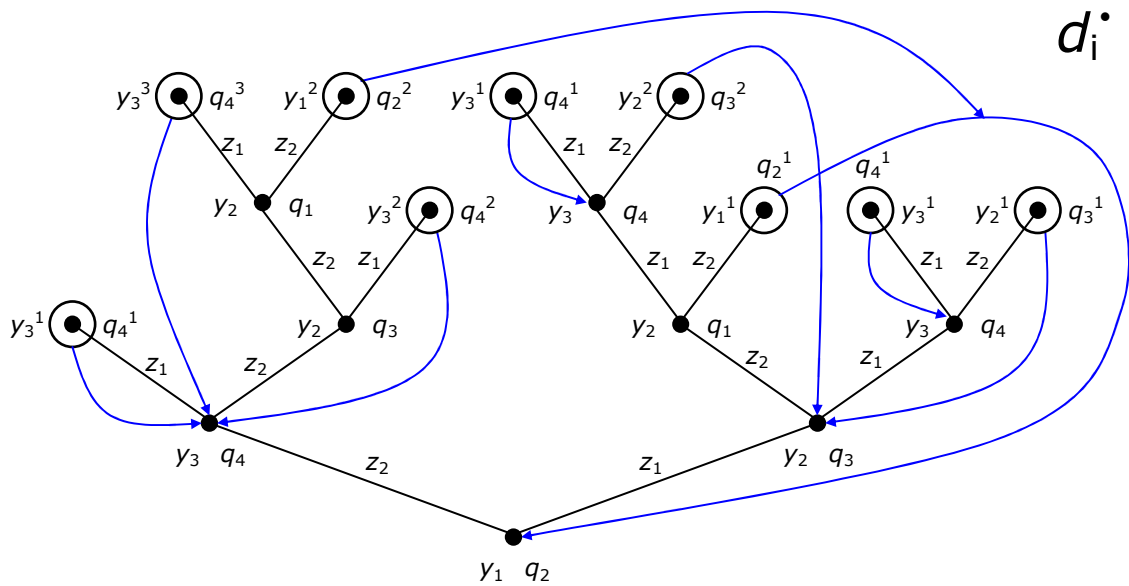
Elementy q_r' noszą nazwę stanów wewnętrznych pośrednich, a elementy y_i' noszą nazwę sygnałów przejść, gdyż sprowadzają one automat ze stanów wewnętrznych pośrednich do stanów wewnętrznych normalnych. Podstawiając w wyrażeniu G_i^{++} w miejsce elementów q_r odpowiadające im elementy y_i otrzymujemy następującą postać:

$$d_i^{++} = {}^0(y_1^1(z_1y_2^2(z_1y_3^3(z_2y_2', z_1y_3')^3, z_2y_2^3(z_2y_1', z_1y_3^4(z_2y_2', z_1y_3')^4)^3)^2, \quad (3)$$

$$z_2y_3^2(z_2y_2^3(z_1y_3', z_2y_2^4(z_2y_1', z_1y_3')^4)^3, z_1y_3'^2)^1)^0$$

Na podstawie wyrażenia G_i^{++} można narysować drzewo z pętlami d_i^\bullet będące odpowiednikiem grafu G_i z rys. 1. Drzewo d_i^\bullet przedstawione zostało na rys. 2. Na rysunku tym uwzględniono również przyporządkowanie:

$$q_r \leftrightarrow y_i \text{ oraz } q_r' \leftrightarrow y_i'.$$



Rys. 2. Drzewo z pętlami dla grafu z rys. 1

Wyrażenie G_i^{++} można łatwo przekształcić z powrotem na wyrażenie G_i^+ . Nie znając więc wyrażenia G_i^+ , a znając wyrażenie G_i^{++} można określić wyrażenie G_i^+ , a tym samym graf automatu G_i . Możliwość przejścia z G_i^{++} na G_i^+ ma duże znaczenie w analizie automatów, gdyż wynikiem analizy automatu o nieznanym grafie G_i jest wyrażenie G_i^{++} .

Przy przejściu z wyrażenia G_i^{++} z powrotem na wyrażenie G_i^+ korzysta się z reguł stosowanych przy budowie wyrażenia G_i^+ i przy transformacji G_i^+ na G_i^{++} . Podstawową własnością wynikającą z tych reguł jest to, że dowolny element q_r wyrażenia G_i^+ (1) może znajdować się przed nawiasem otwierającym tylko wtedy, gdy na wcześniejszych pozycjach wyrażenia G_i^+ nie występuje jego imiennik. Zgodnie z powyższym można określić następujące reguły przekształcenia.

Reguła S1

Jeżeli w wyrażeniu G_i^{++} element q_r znajduje się przed nawiasem otwierającym $^k($ i nie wystąpił dotychczas na wcześniejszych pozycjach wyrażenia G_i^{++} , to przy przekształceniu $G_i^{++} \rightarrow G_i^+$ do wyrażenia G_i^+ przechodzi zarówno dany element q_r jak również te elementy $z_j q_i$, które znajdując się wewnątrz nawiasu $^k(\dots)^k$ podległego pod element q_r nie znajdują się w nawiasach wewnętrznych $^{k+1}(\dots)^{k+1}$ zawartych w danej parze nawiasów $^k(\dots)^k$.

Reguła S2

Jeżeli w członie $^k(\dots)^k$ wyrażenia G_i^{++} , spełniającym warunki z reguły S1 na wprowadzenie tego członu do wyrażenia G_i^+ , znajdują się elementy q_r oznaczone górnymi indeksami, czyli elementy typu q_r^s , to przy przejściu do wyrażenia G_i^+ elementy te tracą swoje górne indeksy.

Reguła S3

Jeżeli w wyrażeniu G_i^{++} element q_r znajduje się przed nawiasem otwierającym $^k($ i jeżeli element ten wystąpił już w wyrażeniu G_i^{++} na wcześniejszej pozycji, to przy przekształceniu $G_i^{++} \rightarrow G_i^+$, do wyrażenia G_i^+ przechodzi tylko rozpatrywany element q_r natomiast podległy mu człon $^k(\dots)^k$ zostanie pominięty.

4.3. Metoda analizy programowego wariantu automatu Moore'a

Zgodnie z punktem 4.1, automat powinien być testowany taką sekwencją sygnałów wejściowych $z_j \in Z$, aby na podstawie uzyskiwanych wyników testowania, w postaci sekwencji sygnałów wyjściowych $y_j \in Y$, można było w trakcie analizy automatu budować wyrażenia d_i^{++} i G_i^{++} reprezentujące dendryt z pętlami d_i^\bullet , będący odpowiednikiem grafu G_i . Przy budowaniu wyrażeń d_i^{++} i G_i^{++} , reprezentujących wynik testowania automatu, korzystamy z pewnych własności struktury tych wyrażeń wynikających z przyjętego algorytmu budowy tych wyrażeń. Na podstawie tych własności możemy sformułować następujące reguły interpretacji wyników testowania automatu:

Reguła M1

Jeżeli na wyjściu automatu pojawił się taki element y_i , który nie wystąpił jeszcze w wyrażeniu d_i^{++} , to element ten związany jest z takim stanem wewnętrznym, który nie wystąpił jeszcze w wyrażeniu G_i^{++} . Jeżeli najwyższy indeks porządkowy posiadał stan b_r , to nowy stan otrzymuje indeks $r+1$.

Reguła M2

Jeżeli na wyjściu automatu pojawił się element y_i , który już wystąpił w wyrażeniu d_i^{++} i zapisany na aktualnej pozycji wyrażenia d_i^{++} podlega pod swój imiennik znajdujący się w przodzie tego wyrażenia, to zachodzi następująca alternatywa: badany element y_i związany jest z takim samym stanem b_r co jego imiennik znajdujący się w przodzie wyrażenia d_i^{++} lub też element ten związany jest z nowym stanem b_i , który jeszcze nie wystąpił w wyrażeniu G_i^{++} . Oznaczmy badany stan symbolem b^* , natomiast stan związany z imiennikiem danego elementu y_i symbolem b_r . Możemy sformułować następujący wniosek:

Jeżeli wszystkie elementy $z_j \in Z$ podane na wejście automatu przy stanie b^* sprowadzają automat do takich samych stanów, do których automat przechodzi, gdy znajduje się w stanie b_r , to wówczas $b^* = b_r$, a w przypadku niezgodności wyników $b^* \neq b_r$. Jeżeli powyższa własność jest spełniona i spełniony jest pierwszy składnik alternatywy, to wówczas badany stan $b^* = b_r$ jest wierzchołkiem końcowym grafu d_i^\bullet i przyjmuje się oznaczać go symbolem b_r' .

Reguła M3

Jeżeli na wyjściu automatu pojawił się element y_i , który już wystąpił w wyrażeniu d_i^{++} lecz nie podlega pod swój imiennik występujący w przodzie wyrażenia d_i^{++} , to zachodzi taka sama alternatywa jak w regule M2, z tym, że przy spełnieniu pierwszej części alternatywy wierzchołek $b^* = b_r$ nie może być wierzchołkiem końcowym grafu d_i^\bullet .

Reguły budowania wyrażenia G_i^{++} i d_i^{++} na podstawie wyników testowania można sformułować następująco:

Reguła M4

Jeżeli w budowanym wyrażeniu G_i^{++} pojawił się w wyniku testowania automatu taki element b_r (bez indeksu górnego), który jeszcze nie wystąpił w wyrażeniu G_i^{++} , bądź też wystąpił lecz nie podlega pod swój imiennik w przodzie, to za tym symbolem w wyrażeniu G_i^{++} stawiamy kolejny nawias otwierający. Reguła ta dotyczy również wyrażenia d_i^{++} .

Reguła M5

Jeżeli w budowanym wyrażeniu G_i^{++} pojawił się w wyniku testowania automatu taki element b_j' , za którym wystąpił przecinek, to należy wrócić przed nawias otwierający i testować automat przy stanie wewnętrznym b_j zapisanym przed tym nawiasem takim elementem $z_r \in Z$, który dla stanu b_j nie był jeszcze rozpatrywany.

Reguła M6

Jeżeli w budowanym wyrażeniu G_i^{++} wystąpił w wyniku testowania automatu nawias zamykający $)^k$ z indeksem $k > 1$, wówczas należy zmniejszyć indeks nawiasów k o jeden i wrócić przed nawias otwierający z aktualną wartością indeksu k , przed którym stoi symbol b_j stanu wewnętrznego.

Na podstawie częściowo już skompletowanego wyrażenia G_i^{++} należy określić, czy przy stanie b_j zostały już rozpatrzone wszystkie elementy $z_r \in Z$. Jeżeli tak, to wówczas w wyrażeniu G_i^{++} stawiamy nawias zamykający $)^k$ z aktualną wartością indeksu k i powtarzamy czynności określone w regule M6. Jeżeli nie, to wówczas testujemy automat przy stanie b_j kolejnym elementem $z_r \in Z$, jeszcze nie rozpatrywanym przy tym stanie.

4.4. Przykład analizy automatu

Zadany jest programowy wariant automatu Moore'a zapisany w pamięci komputera. Znany jest zbiór sygnałów wejściowych $Z = \{z_1, z_2\}$ i zbiór sygnałów wyjściowych $Y = \{y_1, y_2, y_3\}$ badanego automatu. Nie jest natomiast znany graf przejść automatu i jego tabela wyjść. Celem analizy jest określenie grafu przejść oraz przyporządkowania jego wierzchołkom sygnałów wyjściowych $y_i \in Y$. Poszczególne kroki tej analizy, przeprowadzonej na komputerze, scharakteryzowane zostaną poniżej.

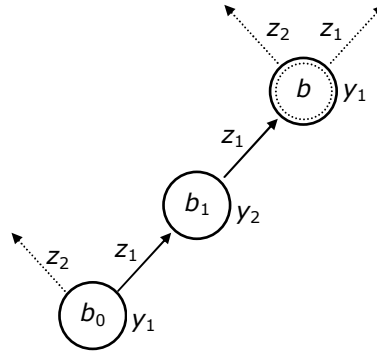
W pierwszej kolejności na wejściu automatu podano (z klawiatury komputera) sekwencję sygnałów $z_1 z_1$, w wyniku czego na wyjściu automatu (czyli na ekranie) otrzymano sekwencję sygnałów wyjściowych $y_1 y_2 y_1$. Na podstawie tych dwóch sekwencji zbudowany został początkowy fragment wyrażenia d_i^{++} :

$$d_i^{++} = {}^0(y_1^1(z_1 y_2^2(z_1 y_1 \dots \tag{4}$$

Stany wewnętrzne analizowanego automatu przyjęto oznaczać symbolami $b_j \in B$. Zgodnie z regułą M1 wyrażenie d_i^{++} (4) można przekształcić na odpowiadające mu wyrażenie G_i^{++} :

$$G_i^{++} = {}^0(b_0^1(z_1 b_1^2(z_1 b \dots \tag{5}$$

gdzie symbol b (bez indeksu) oznacza nieokreślony jeszcze stan wewnętrzny automatu $\langle A_i \rangle$. Na podstawie wyrażenia G_i^{++} (5) można narysować odpowiadający mu fragment grafu d_i^* o strukturze drzewiastej. Fragment tego grafu przedstawiony na rys. 3 ułatwia podjęcie decyzji dotyczącej określenia stanu oznaczonego w (5) symbolem b .



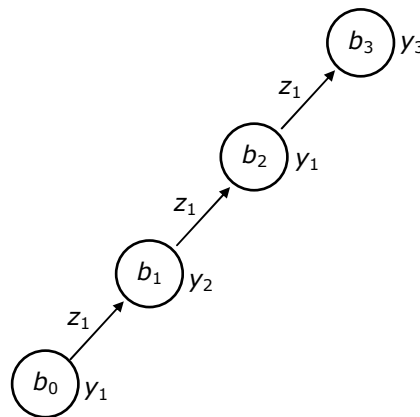
Rys. 3. Wyniki testów - krok 1

Zgodnie z tym grafem symbol y_1 przyporządkowany wierzchołkowi b podlega pod swój imiennik y_1 w przodzie wyrażenia d_i^{++} , któremu przyporządkowany jest stan b_0 . Zachodzi więc alternatywa, zgodnie z regułą M2, że stan b jest stanem b_0 lub że stan b jest stanem b_2 , który nie wystąpił jeszcze w wyrażeniu G_i^{++} . Aby upewnić się, która część alternatywy jest spełniona sprawdzono zachowanie się automatu przy stanie b po podaniu na jego wejście sygnału z_1 . Pod wpływem tego sygnału na wyjściu automatu pojawił się sygnał y_3 . Wynika stąd, że symbol b reprezentuje sobą nowy stan b_2 , gdyż przy stanie b_0 i przy sygnale z_1 na wyjściu pojawia się sygnał y_1 . Po wykonaniu tej operacji wyrażenia d_i^{++} i G_i^{++} uzupełnione zostały o nowe elementy i otrzymały następującą postać:

$$d_i^{++} = {}^0(y_1^1(z_1 y_2^2(z_1 y_1^3(z_1 y_3 \dots$$
 (6)

$$G_i^{++} = {}^0(b_0^1(z_1 b_1^2(z_1 b_2^3(z_1 b_3 \dots$$
 (7)

Graf narysowany na podstawie wyrażenia G_i^{++} (7) przedstawiony został na rys 4.



Rys. 4. Wyniki testów - krok 2

Kontynuując testowanie automatu $\langle A_i \rangle$ przy stanie b_3 (rys. 4) sygnałem z_1 , a więc przy

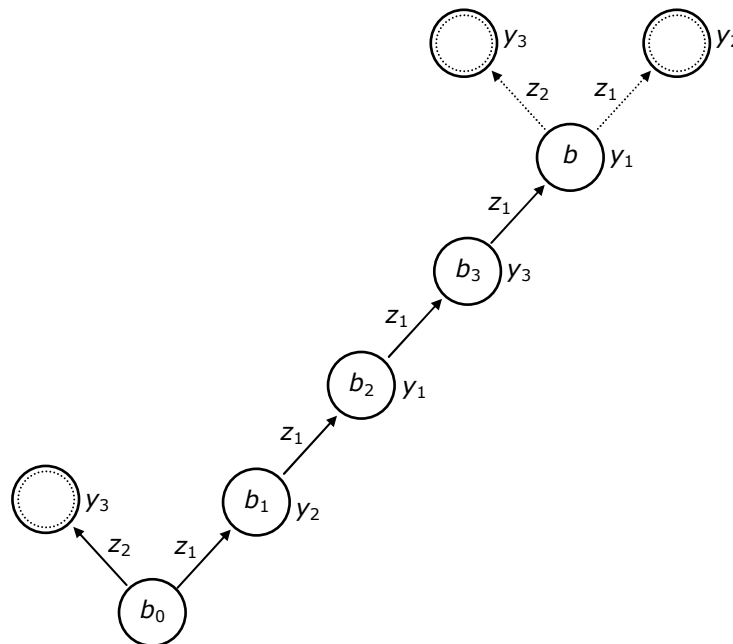
słowie wejściowym $z_1 z_1 z_1 z_1$, na ekranie monitora wyświetlona została sekwencja $y_1 y_2 y_1 y_3 y_1$. Na podstawie tej sekwencji wyrażenie d_i^{++} (6) uzupełnione zostaje o człon ${}^4(z_1 y_1$ i przyjmuje następującą postać:

$$d_i^{++} = {}^0(y_1^1(z_1 y_2^2(z_1 y_1^3(z_1 y_3^4(z_1 y_1 \dots \quad (8)$$

Ponieważ symbol y_1 występujący w członie ${}^4(z_1 y_1$ podlega pod dwa swoje imienniki y_1 występujące w przodzie wyrażenia d_i^{++} (8), należy rozpatrzeć warunki określone w regule M2 w celu określenia stanu wewnętrznego automatu, generującego rozpatrywany y_1 . Po oznaczeniu poszukiwanego stanu symbolem b wyrażenie G_i^{++} (7) przyjmuje teraz następującą postać:

$$G_i^{++} = {}^0(b_0^1(z_1 b_1^2(z_1 b_2^3(z_1 b_3^4(z_1 b \dots \quad (9)$$

gdzie symbol b w członie ${}^4(z_1 b$ oznacza stan podlegający określeniu. Graf reprezentowany przez wyrażenie G_i^{++} , ze wskazaniem linią przerywaną na dalsze kierunki testowania, pokazano na rys. 5.



Rys. 5. Wyniki testów - krok 3

Testując automat sygnałami z_1 i z_2 zarówno przy stanie b_0 jak i b stwierdzono, że stan b zapisany w termie ${}^4(z_1 b$ wyrażenia G_i^{++} jest stanem b_0 . W związku z powyższym wierzchołek b w termie ${}^4(z_1 b$ traktowany jest jako wierzchołek końcowy budowanego drzewa d_i^* , a odpowiadający temu wierzchołkowi symbol y_3 w wyrażeniu d_i^{++} traktowany jest jako pośredni sygnał wyjściowy y_3' . Ponieważ wierzchołek b_0 wystąpił już w wyrażeniu G_i^{++} , to zgodnie z regułami budowy wyrażenia symbolicznego, za symbolem b_0' w G_i^{++} stawiamy przecinek i wychodzimy przed nawias otwierający. Znajdujemy się w wierzchołku b_3 stojącym przed nawiasem otwierającym 4 . Krawędź z_1 wychodząca z tego wierzchołka była już rozpatrzona, należy więc rozpatrzeć teraz drugą krawędź z_2 . W tym celu w omawianym przykładzie analizowany automat $\langle A_i \rangle$ sprowadzony został do stanu początkowego b_0 , a

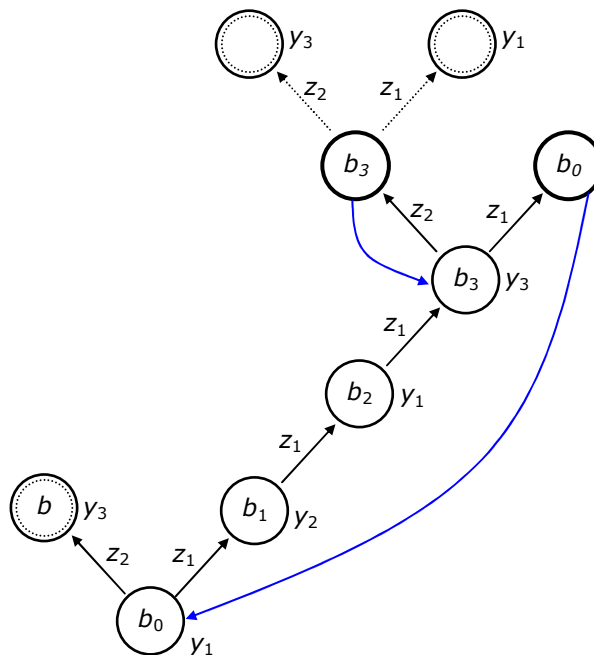
następnie na jego wejście podano sekwencję $z_1 z_1 z_1 z_2$ w wyniku czego na wyjściu automatu otrzymano sekwencję $y_1 y_2 y_1 y_3 y_3$. Testując automat zgodnie z regułą M2 stwierdzono, że obydwa sygnały y_3 generowane są przez jeden i ten sam stan wewnętrzny b_3 . W wyniku powyższego testowania wyrażenia d_i^{++} (8) i G_i^{++} (9) rozwinięte zostały do postaci:

$$d_i^{++} = {}^0(y_1^1(z_1 y_2^2(z_1 y_1^3(z_1 y_3^4(z_1 y_1', z_2 y_3')^4) \dots \tag{10}$$

$$G_i^{++} = {}^0(b_0^1(z_1 b_1^2(z_1 b_2^3(z_1 b_3^4(z_1 b_0', z_2 b_3')^4) \dots \tag{11}$$

Graf narysowany na podstawie wyrażenia G_i^{++} (11) z uwzględnieniem kroków testowania przedstawiony został na rys. 6.

Z wyrażenia G_i^{++} (11) wynika, że obydwie krawędzie z_1 i z_2 wychodzące z wierzchołka b_3 zostały rozpatrzone, w związku z czym za nawiasem $)^4$ stawiamy przecinek i przechodzimy do wierzchołka b_2^3 (w G_i^{++}). Do rozpatrzenia pozostaje teraz druga krawędź wychodząca z b_2 , reprezentująca sygnał wejściowy z_2 . Automat $\langle A_i \rangle$ sprowadzony został więc do stanu początkowego b_0 a następnie na jego wejście podano sekwencję $z_1 z_1 z_2$ w wyniku czego na wyjściu automatu otrzymano sekwencję $y_1 y_2 y_1 y_1$. Ponieważ aktualnie rozpatrywany ostatni symbol y_1 w sekwencji $y_1 y_2 y_1 y_1$ podlega pod swoje imienniki w przodzie wyrażenia d_i^{++} , przy testowaniu zastosowano regułę M2. W wyniku testowania ustalono, że stan b , przy którym generowany jest sygnał y będący ostatnim elementem ciągu $y_1 y_2 y_1 y_1$, jest stanem b_0 , który wystąpił już wcześniej w wyrażeniu G_i^{++} .



Rys. 6. Wyniki testów - krok 4

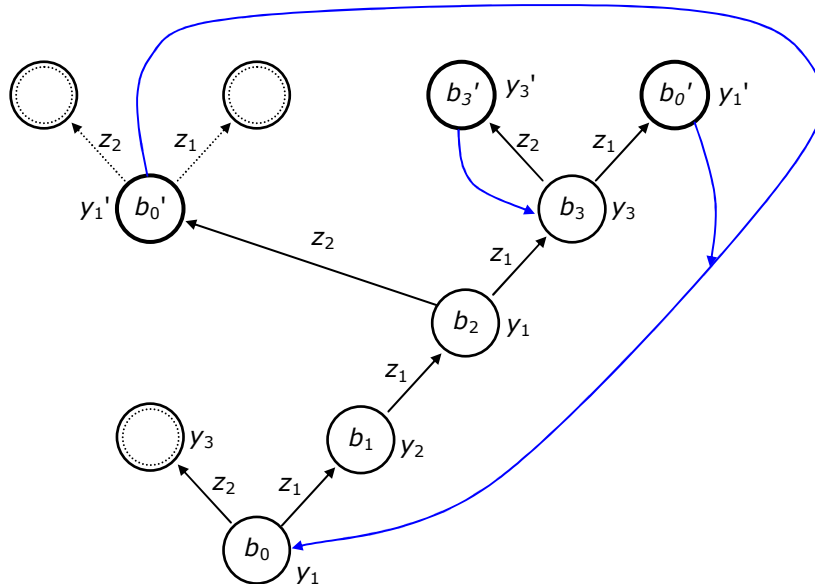
Na podstawie uzyskanych wyników wyrażenia d_i^{++} (10) i G_i^{++} (11) rozszerzone zostały do następującej postaci:

$$d_i^{++} = {}^0(y_1^1(z_1 y_2^2(z_1 y_1^3(z_1 y_3^4(z_1 y_1', z_2 y_3')^4, z_2 y_1')^3, \dots \tag{12}$$

$$G_i^{++} = {}^0(b_0^1(z_1 b_1^2(z_1 b_2^3(z_1 b_3^4(z_1 b_0', z_2 b_3')^4, z_2 b_0')^3, \dots \tag{13}$$

Drzewo z pętlami narysowane na podstawie wyrażenia G_i^{++} (13) przedstawione zostało na rys. 7.

Na podstawie grafu d_i^* z rys. 7 można określić następny etap testowania automatu $\langle A_i \rangle$. Należy rozpocząć go od wierzchołka b_1 przy sygnale z_2 . Zgodnie z powyższym automatem $\langle A_i \rangle$ sprowadzono (w rozpatrywanym przykładzie) do stanu początkowego b_0 , a następnie na jego wejście podano sekwencję $z_1 z_2$ otrzymując na wyjściu automatu sekwencję $y_1 y_2 y_2$.



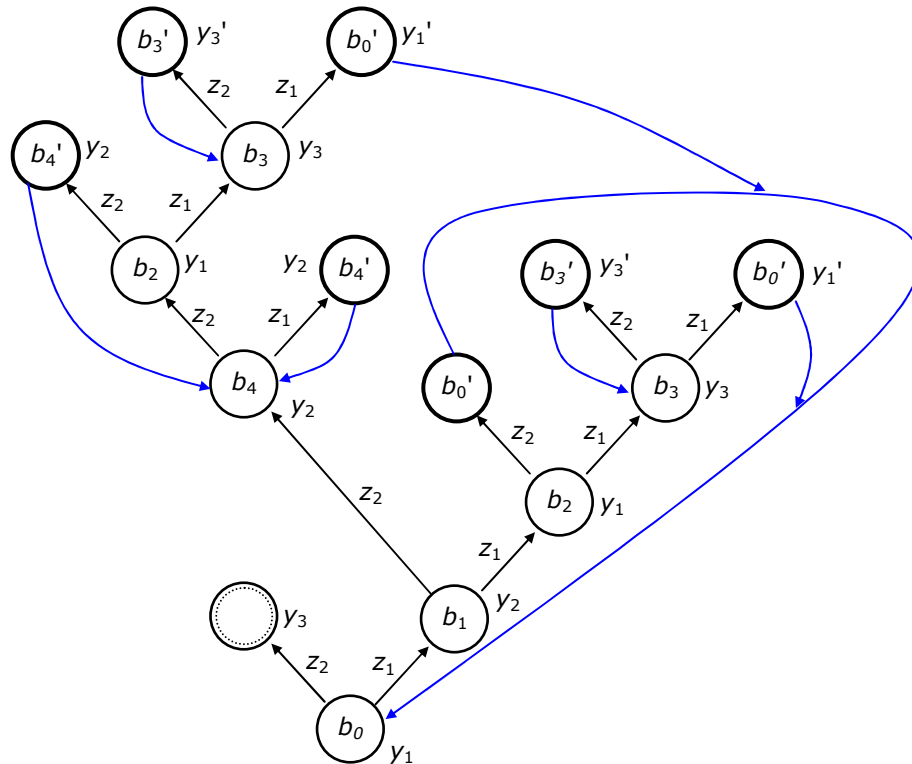
Rys. 7. Wyniki testów - krok 5

Ponieważ w otrzymanej sekwencji $y_1 y_2 y_2$ ostatni element y_2 podlega pod swój imiennik w przodzie wyrażenia d_i^{++} (12), do ustalenia stanu b generującego y_2 przeprowadzono operacje wynikające z reguły M2. Po przeprowadzeniu tych operacji, jak również po przeprowadzeniu testowania dla wierzchołków będących potomnymi rozpatrywanego stanu b , na podstawie uzyskanych wyników wyrażenia d_i^{++} (12) i G_i^{++} (13) rozwinięte zostają do postaci jak niżej:

$$d_i^{++} = {}^0(y_1^1(z_1 y_2^2(z_1 y_1^3(z_1 y_3^4(z_1 y_1', z_2 y_3')^4, z_2 y_1')^3, z_2 y_2^3(z_1 y_2', z_2 y_1^4(z_1 y_3^5(z_1 y_1', z_2 y_3')^5, z_2 y_2)^4)^3)^2, \dots) \quad (14)$$

$$G_i^{++} = {}^0(b_0^1(z_1 b_1^2(z_1 b_2^3(z_1 b_3^4(z_1 b_0', z_2 b_3')^4, z_2 b_0')^3, z_2 b_4^3(z_1 b_4', z_2 b_4^4(z_1 b_3^5(z_1 b_0', z_2 b_3')^5, z_2 b_4')^4)^3)^2, \dots) \quad (15)$$

Graf d_i^* narysowany na podstawie wyrażenia G_i^{++} (15) przedstawiony został na rys. 8.



Rys. 8. Wyniki testów - krok 6

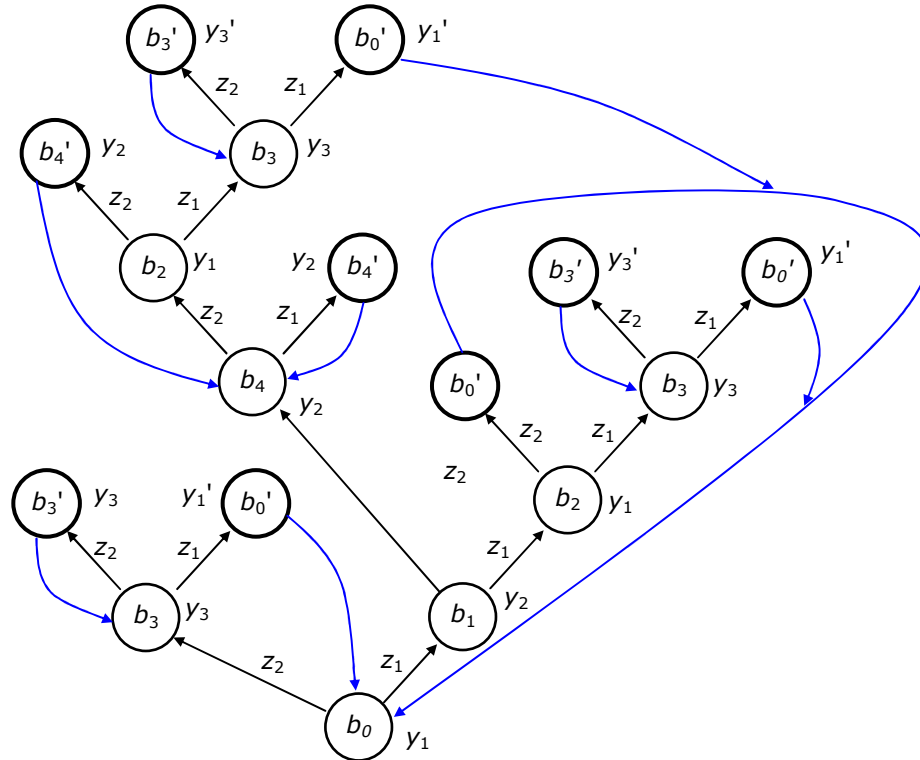
Z wyrażenia G_i^{++} (15) jednoznacznie wynika, że dalszy etap testowania automatu $\langle A_i \rangle$ należy przeprowadzić wzdłuż krawędzi z_2 wychodzącej z wierzchołka początkowego b_0 .

Po podaniu na wejście automatu sygnału z_2 , przy stanie b_0 , na wyjściu automatu pojawił się sygnał y_3 . Z wyrażenia d_i^{++} (14) wynika, że symbol y_3 wystąpił już w tym wyrażeniu na wcześniejszych pozycjach. Jednakże analizowany y_3 nie podlega pod swoje imienniki w przodzie wyrażenia d_i^{++} (14). Zachodzi więc alternatywa, że stan wewnętrzny b , który spowodował wygenerowanie rozpatrywanego sygnału y_3 jest stanem nowym, bądź jest to jeden ze stanów już rozpoznanych i przyporządkowanych w wyrażeniu G_i^{++} (15) imiennikom rozpatrywanego elementu y_3 . Zastosowano więc tu regułę M3. Na wejście automatu, przy stanie początkowym b_0 , podano $z_2 z_1$ i otrzymano $y_1 y_3 y_1$. Następnie sprowadzono automat do stanu początkowego i podano sekwencję $z_2 z_2$, a na wyjściu otrzymano $y_1 y_3 y_3$. Uzyskane wyniki porównano z pozycjami y_3^4 (i y_3^5 (wyrażenia d_i^{++} (14). Stwierdzono, że w rozpatrywanym przypadku automat zachował się tak samo. Ponieważ elementom y_3^4 (i y_3^5 (przyporządkowany jest stan b_3 w wyrażeniu G_i^{++} (15), stan b generujący rozpatrywany y_3 jest stanem b_3 . Zwróćmy uwagę na to, że zachowanie się automatu $\langle A_i \rangle$ przy stanie b_3 , zarówno pod wpływem sygnału z_1 jak i z_2 , zostało już określone w wyrażeniu G_i^{++} (15). W związku z powyższym proces testowania automatu należy uważać za zakończony. Wyrażenia d_i^{++} (14) i G_i^{++} (15) uzupełnione o omówione wyżej wyniki ostatniego etapu testowania przyjmują teraz następującą postać:

$$d_i^{++} = {}^0(y_1^1(z_1 y_2^2(z_1 y_1^3(z_1 y_3^4(z_1 y_1', z_2 y_3')^4, z_2 y_1')^3, z_2 y_2^3(z_1 y_2', z_1 y_1^4(z_1 y_3^5(z_1 y_1', z_2 y_3')^5, z_2 y_2^4)^3)^2, z_2 y_3^2(z_1 y_1', z_2 y_3')^2)^1)^0 \tag{16}$$

$$G_i^{++} = {}^0(b_0^1(z_1b_1^2(z_1b_2^3(z_1b_3^4(z_1b_0', z_2b_3')^4, z_2b_0')^3, z_2b_4^3(z_1b_4', z_2b_2^4(z_1b_3^5(z_1b_0', z_2b_3')^5, z_2b_4')^4)^3)^2, z_2b_3^2(z_1b_0', z_2b_3')^2)^1)^0 \quad (17)$$

Na podstawie wyrażenia G_i^{++} (17) można narysować drzewo d_i^* ilustrujące wyniki uzyskane w procesie analizy przykładowego automatu $\langle A_i \rangle$ typu Moore'a. Graf d_i^* (drzewo z pętlami) przedstawiony został na rys. 9.



Rys. 9. Wyniki testów - końcowe drzewo z pętlami

Porównując wyrażenia d_i^{++} i G_i^{++} na pozycjach stojących przed nawiasami otwierającymi można określić przyporządkowanie stanom wewnętrznym b_j sygnałów wyjściowych y_i analizowanego automatu. W wyniku tego porównania otrzymano następującą tabelę wyjść T_i :

Tabela \bar{T}_i

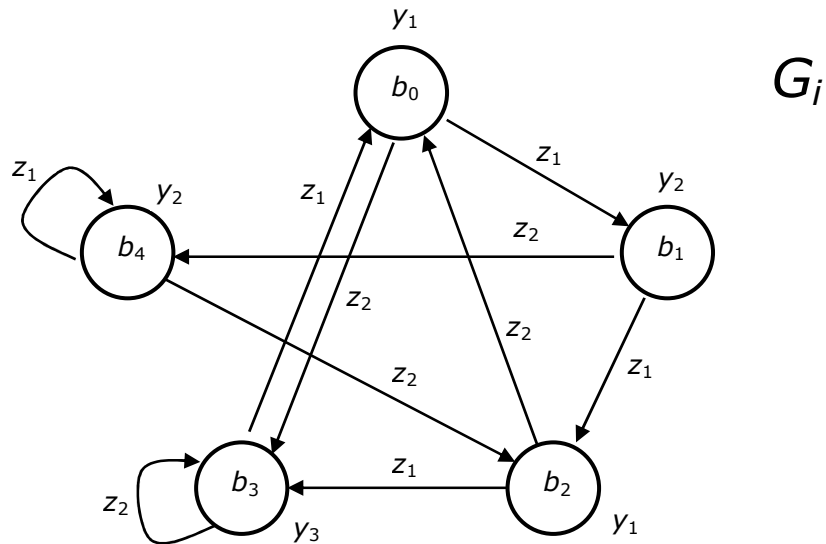
q	y
b_0	y_1
b_1	y_2
b_2	y_1
b_3	y_3
b_4	y_2

W celu otrzymania grafu automatu $\langle A_i \rangle$, wyrażenie G_i^{++} (17) reprezentujące drzewo z pętlami z rys. 9 należy przekształcić na wyrażenie symboliczne G_i^+ reprezentujące graf automatu. Przekształcenie to wykonywane jest na podstawie podanych wyżej reguł S1, S2 i

S3. Wyrażenie G_i^+ wyprowadzone z G_i^{++} (17) ma następującą postać:

$$G_i^+ = {}^0(b_0^1(z_1b_1^2(z_1b_2^3(z_1b_3^4(z_1b_0, z_2b_3)^4, z_2b_0)^3, z_2b_4^3(z_1b_4, z_2b_2)^3)^2, z_2b_3)^1)^0 \quad (18)$$

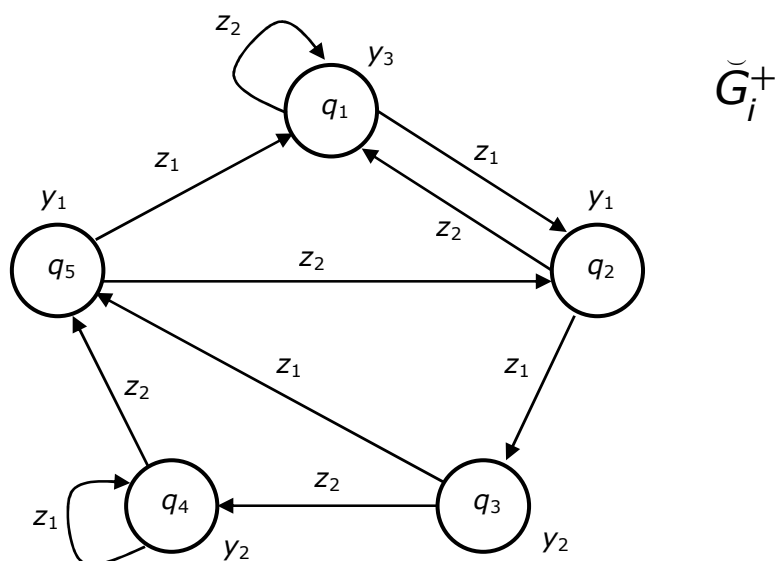
Wyrażenie G_i^+ (18) i tabela T_i jednoznacznie określają graf G_i analizowanego automatu $\langle A_i \rangle$. Graf ten przedstawiony został na rys. 10.



Rys. 10. Wyniki testów - końcowy graf automatu

Założmy, że w rezultacie przeprowadzonych testów automatu uzyskane zostały wyniki w postaci tabeli T_i , wyrażenia G_i^{++} (18) i grafu G_i^+ z rys. 10. Przyjmijmy, że dla analizowanego automatu $\langle A_i \rangle$ znana jest postać źródłowa danych w postaci wyrażenia symbolicznego \check{G}_i^+ (19), reprezentowanego przez graf na rys. 11, oraz tabeli wyjść automatu.

$$\check{G}_i^+ = {}^0(q_2^1(z_1q_3^2(z_2q_4^3(z_2q_5^4(z_1q_1^5(z_1q_2, z_2q_1)^5, z_2q_2)^4, z_1q_4)^3, z_1q_5)^2, z_2q_1)^1)^0 \quad (19)$$



Rys. 11. Wzorcowy graf testowanego automatu

Należy sprawdzić, czy uzyskane wyniki odzwierciedlają rzeczywisty graf przejść badanego automatu $\langle A_i \rangle$. W tym celu porównuje się wyrażenie \tilde{G}_i^+ (19) z wyrażeniem G_i^+ (18) na odpowiadających sobie pozycjach q_r^k i b_i^k w obydwu wyrażeniach. W rozpatrywanym przykładzie w wyniku porównania otrzymano następującą równoważność symboli reprezentujących stany wewnętrzne automatu $\langle A_i \rangle$:

$$q_2 = b_0, \quad q_3 = b_1, \quad q_4 = b_4, \quad q_5 = b_4, \quad q_1 = b_3.$$

Podstawiając symbole q_r pod odpowiadające im symbole b_i w wyrażeniu G_i^+ (18) otrzymujemy wyrażenie \tilde{G}_i^+ (19) co świadczy o tym, że uzyskane wyniki analizy są poprawne.

5. Program do komputerowej analizy automatów skończonych

W poprzednich punktach rozważono możliwość analizy automatów skończonych przy wykorzystaniu komputera. Przedstawiono również algorytm umożliwiający taką analizę. W celu weryfikacji algorytmu powstał program, który zapewnia komputerową realizację następujących funkcji:

- przechowywanie struktury badanego automatu w pamięci komputera,
- testowanie działania automatu (po wprowadzeniu ciągu sygnałów wejściowych program generuje odpowiadający mu ciąg sygnałów wyjściowych),
- wspomaganie procesu odtworzenia grafu automatu na podstawie otrzymanych wyników (wizualizacja wyników i ich wstępne przetworzenie).

Poniżej zostanie opisany sposób korzystania z tego programu.

5.1. Opis rozszerzonej wersji opcji ANALIZA

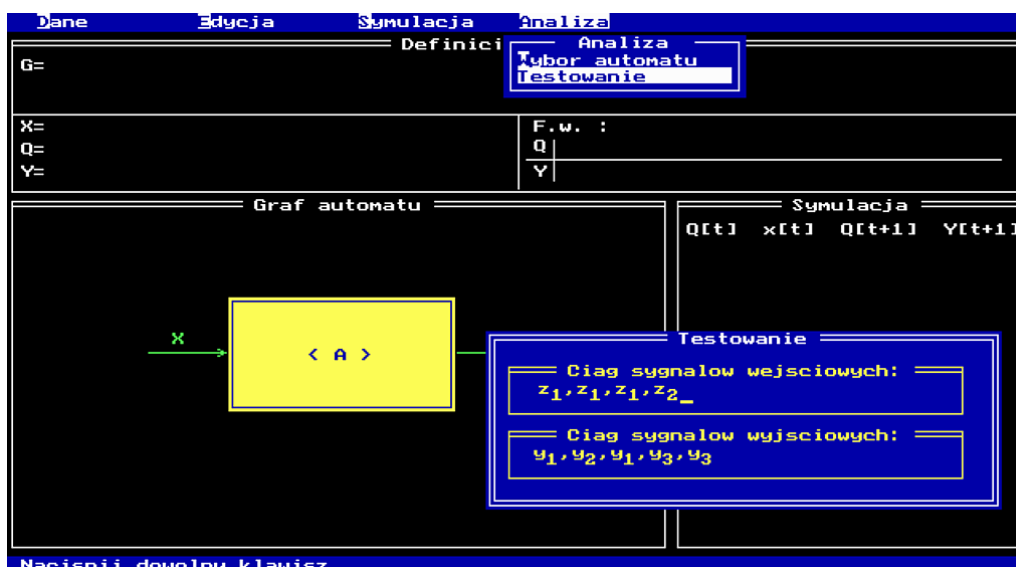
Jedną z opcji programowego wariantu automatu skończonego jest opcja ANALIZA. Po jej wybraniu z menu głównego istnieje możliwość realizacji jednej z dwóch funkcji:

- wczytanie opisu automatu,
- testowanie automatu.

Pierwsza z tych funkcji umożliwia wczytanie automatu z pliku o podanej nazwie. Zbiór sygnałów wejściowych ma postać $Z = \{z_0, z_1, z_2, \dots\}$, używane są zawsze kolejne indeksy. Jeśli zakres indeksów dla symboli z_i nie zostanie podany, to może być on rozpoznany w pierwszej fazie testowania. Jeśli w ciągu testowym zostanie podany symbol z_i , który nie należy do zbioru sygnałów wejściowych, to wyświetlony zostanie komunikat "Przejście nieokreślone!".

Po wczytaniu automatu z pliku można przystąpić do testowania automatu, wybierając drugą z wymienionych wcześniej funkcji.

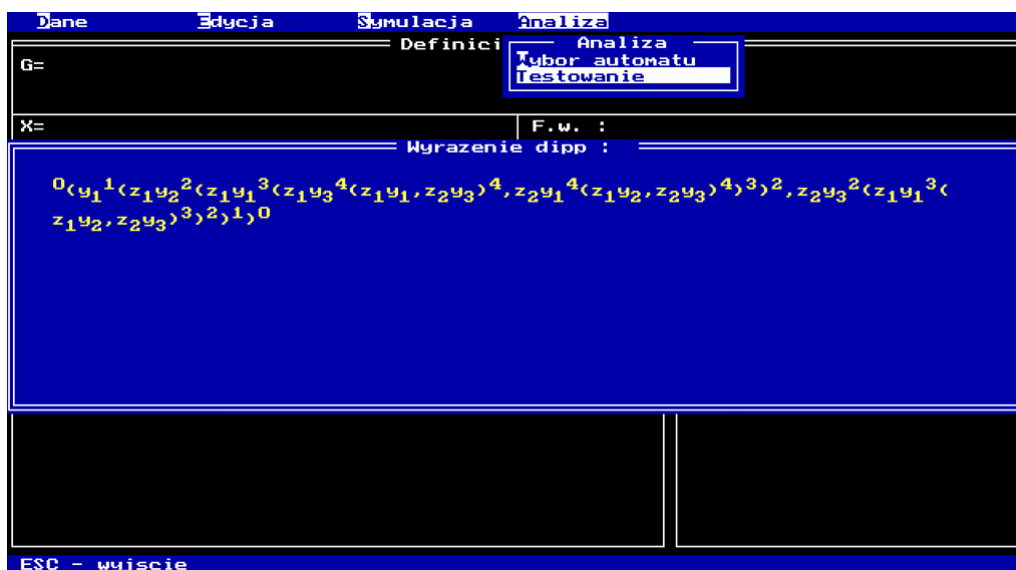
Testowanie automatu rozpoczyna się od podania ciągu sygnałów wejściowych. W tym celu należy nacisnąć klawisz $\langle \text{SPACJA} \rangle$. Na ekranie pojawi się okno przedstawione na rys. 12. Można wówczas podać ciąg sygnałów wejściowych, oddzielając poszczególne elementy tego ciągu przecinkami. Po wciśnięciu klawisza $\langle \text{ENTER} \rangle$ program wygeneruje ciąg sygnałów wyjściowych, odpowiadający danym wejściowym. Należy zwrócić uwagę, że ciąg ten jest zawsze o jeden element dłuższy od ciągu wejściowego, ponieważ zawiera dodatkowo informację o stanie początkowym (aktualnym) automatu przed podaniem pierwszego sygnału wejściowego z ciągu.

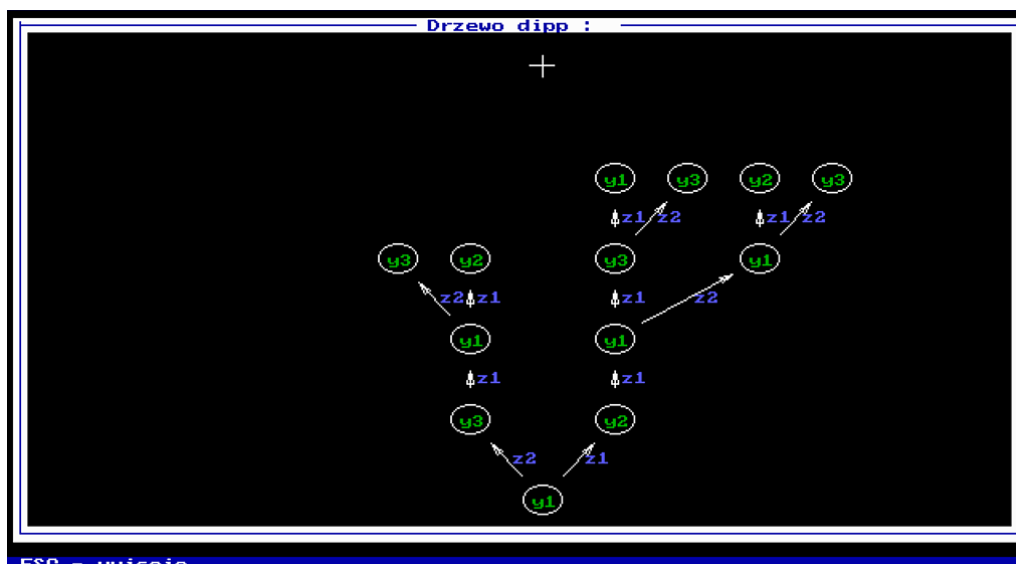


Rys. 12. Widok ekranu testowania automatu

W trakcie testowania powinna być możliwość powrotu badanego automatu do jednego z wcześniejszych stanów. Można to osiągnąć sprowadzając automat do stanu początkowego (klawisz <TAB> - zerowanie), a następnie podając odpowiedni ciąg wejściowy. O ile nie zostanie naciśnięty klawisz <TAB> kolejne testowanie odbywa się od stanu, w którym automat znalazł się po poprzednim kroku.

W celu ułatwienia procesu analizy automatu program, na podstawie otrzymywanych danych (ciąg wejściowy i wyjściowy), po każdym kroku testowania buduje wyrażenie Można je wyświetlić na ekranie naciskając klawisz <F1> (patrz rys. 13). Odpowiadający temu wyrażeniu dendryt można otrzymać na ekranie wciskając klawisz <F2> (patrz rys. 14). W przypadku, gdy dendryt nie mieści się w całości na ekranie komputera, za pomocą klawiszy kierunkowych (strzałek) można wybrać dowolny jego fragment. Do szybszego przewijania ekranu należy dodatkowo wcisnąć klawisz <SHIFT>.

Rys. 13. Wyświetlenie wyrażenia d_i^{++}

Rys. 14. Wyświetlenie drzewa d_i^{++}

6. Przebieg ćwiczenia

Przed przystąpieniem do ćwiczenia sprawdzana jest znajomość metody analizy automatów w zakresie identyfikacji grafu automatu przedstawionej w punktach 4.1, 4.2 i 4.3. Po zapoznaniu się z obsługą programu i jego podstawowymi opcjami, dla automatów ze wskazanych plików należy:

- ustalić alfabet wejściowy Z składający się z symboli z_i (jeśli nie został podany),
- wykonać testowanie automatu, według przykładu podanego w punkcie 4.4, podając ciągi testowe i obserwując sygnały pojawiające się na wyjściu automatu,
- zapisywać wyniki testowania automatu w postaci grafu o strukturze drzewiastej i kolejnych fragmentów wyrażeń symbolicznych G_i^{++} i d_i^{++} ,
- przekształcić uzyskane wyrażenie G_i^{++} do postaci wyrażenia G_i^+ i na jego podstawie narysować graf G_i rozpoznanego automatu,
- przedstawić uzyskane wyniki prowadzącemu do weryfikacji.

7. Sprawozdanie z ćwiczenia

W sprawozdaniu należy umieścić:

- temat i cel ćwiczenia,
- wyniki testowania automatu w postaci kolejnych fragmentów wyrażeń d_i^{++} , G_i^{++} i odpowiadających im kolejnych obrazów drzewa d_i^\bullet ,
- kompletne wyrażenia G_i^{++} i d_i^{++} uzyskane na podstawie wyników testów automatu,
- przyporządkowanie sygnałów wyjściowych stanów wewnętrznym automatu,
- przekształcenie wyrażenia G_i^{++} do postaci wyrażenia G_i^+ ,
- graf G_i analizowanego automatu uzyskany na podstawie wyrażenia G_i^+ ,
- wnioski z ćwiczenia.

8. Literatura

- [1]. Bromirski J.: Teoria automatów, WNT, Wyd. II., Warszawa 1970
- [2]. Kazimierczak J.: System cybernetyczny, Wiedza Powszechna (seria Omega), Warszawa 1978
- [3]. Kazimierczak J.: Elementy syntezy formalnej systemów operacyjnych, Biblioteka WASC, Wydawnictwo Politechniki Wrocławskiej, Wrocław 1979
- [4]. Kazimierczak J.: Automaty rozgrywające parametryczne - Synteza i zastosowanie, Prace Naukowe Instytutu. Cybernetyki. Technicznej PWr., Monografie, Wydawnictwo Politechniki Wrocławskiej, Wrocław 1974
- [5]. Kazimierczak J., Kluska J., Kaczmarek A.: Podstawy teorii automatów - Laboratorium (skrypt), Wydawnictwo Politechniki Rzeszowskiej, Rzeszów 1984