

INSTYTUT CYBERNETYKI TECHNICZNEJ POLITECHNIKI WROCŁAWSKIEJ
ZAKŁAD SZTUCZNEJ INTELIGENCJI I AUTOMATÓW

Ćwiczenia laboratoryjne z Logiki Układów Cyfrowych

ćwiczenie 205

temat: ZASTOSOWANIE JĘZYKA WYRAŻEŃ REGULARNYCH DO SYNTEZY I ANALIZY
AUTOMATÓW SKOŃCZONYCH

1. CEL ĆWICZENIA

Celem ćwiczenia jest nabycie praktycznej umiejętności projektowania i technicznej realizacji automatów przy zastosowaniu języka wyrażeń regularnych.

2. PROGRAM ĆWICZENIA

1. Na podstawie wyrażenia regularnego opisującego automat określić graf przejść pomiędzy stanami automatu.
2. Przeprowadzić syntezę automatu realizowanego jako automat Moore'a.
3. Realizacja techniczna automatu z zastosowaniem elementów scalonych TTL serii UCY-74.
4. Sprawdzenie poprawności działania modelu automatu.

3. PROBLEMATYKA ĆWICZENIA

Analizę abstrakcyjną automatów przeprowadza się różnymi metodami. Przy prostych automatach proces ten polega na intuicyjnym opisie zachowania automatu na podstawie jego modelu abstrakcyjnego. Zastosowanie języka wyrażeń regularnych do syntezy i analizy automatów skończonych umożliwia przeprowadzenie tego procesu w sposób formalny.

Język wyrażeń regularnych powstał w wyniku poszukiwania prostszych i bardziej funkcjonalnych od tradycyjnych sposobów opisu działania automatów.

4. WIADOMOŚCI PODSTAWOWE

Definicja 1

Automat skończony jest modelem matematycznym systemu dyskretnego działającego w dyskretnych chwilach czasu. Jego działanie jest określone na zbiorach skończonych sygnałów wejściowych, stanów wewnętrznych i sygnałów wyjściowych. Automat skończony można zrealizować sprzętowo lub programowo.

Ogólnie, schemat blokowy automatu skończonego można przedstawić następująco:

(schemat)

gdzie: Z – alfabet wejściowy

Q – zbiór stanów wewnętrznych

Y – alfabet wyjściowy

Automat ten akceptuje słowa należące do języka regularnego. Język regularny jako zbiór słów reprezentowany jest przez wyrażenia regularne.

Założmy, że alfabet wejściowy automatu jest następującym zbiorem:

$$Z = \{z_1, z_2, \dots, z_i, \dots, z_n\}$$

Z symboli tego zbioru możemy zbudować określone słowa, np.:

$$z_1 z_2 z_2 z_1; z_2 z_1 z_3 z_9; \dots, z_1 z_1 z_5; \dots$$

Zbiór wszystkich możliwych słów jest zbiorem nieskończonym Z^*

$$Z^* = \{z_1, z_1 z_2, \dots, z_2 z_1 z_2, \dots, z_9 z_{10}, z_{12}, \dots\}$$

Na zbiorze Z^* można określić rodzinę zbiorów S^* .

$$S^* = \{S_1, S_2, \dots, S_i, \dots, S_n\}$$

Na słowach $S_i \in S^*$ jak również na słowach przynależnych do dowolnego zbioru $S_i \in S^*$ przeprowadzane są określone operacje. Dowolny zbiór $S_i \in S^*$ zawierający słowa wejściowe automatu nazywamy zdarzeniem.

Definicja 2

Do oznaczenia zbiorów powstałych w wyniku wykonania operacji sumy, konkatenacji i iteracji posługujemy się wyrażeniem nazywanym wyrażeniem regularnym. W skład wyrażenia regularnego wchodzi określone słowa połączone znakami reprezentującymi powyższe operacje.

Każde wyrażenie regularne reprezentuje sobą język regularny.

Twierdzenie 1

Jeżeli r jest wyrażeniem regularnym, to istnieje automat NFA with ϵ -moves, który akceptuje słowa języka regularnego reprezentowanego przez to wyrażenie.

Mając wyrażenie regularne możemy wykonywać następując transformacje:

- wyrażenie regularne \rightarrow zbiór słów.
- wyrażenie regularne \rightarrow graf przejść automatu akceptującego język reprezentowany przez to wyrażenie – $r \rightarrow S(r)$.
- wyrażenie regularne \rightarrow gramatyka bezkontekstowa regularna, generująca słowa danego języka $S(r)$.

Synteza abstrakcyjna automatów skończonych

Definicja 3

Synteza abstrakcyjna automatu to określenia takiego opisu formalnego automatu, na podstawie którego można zbudować tablice przejść i wyjść automatu. Synteza ta sprowadza się do przejścia od algorytmu działania automatu do grafu przejść automatu.

Poszczególne etapy tej syntezy to:

1. algorytm słowny
2. przedstawienie algorytmu słownego w postaci wyrażen regularnych
3. określenie grafu przejść

Przykład 1

$$S_1 = z_1 z_2 + z_1 z_1 z_1 \mid y_1$$

$$S_2 = z_1 z_2 z_2 + z_2 z_2 \mid y_2$$

$$S_3 = \overline{S_1 + S_2} \mid y_3 = \varepsilon$$

gdzie: S_1 - zdarzenie warunkujące pojawienie się na wyjściu automatu y_1
 S_2 - zdarzenie warunkujące pojawienie się na wyjściu automatu y_2
 ε - sygnał pusty

W celu określenia stanów projektowanego automatu wprowadza się pojęcie „miejsca” w wyrażeniu regularnym. Miejscem jest położenie pomiędzy literami, między literą i znakiem dysjunkcji (OR) oraz początek i koniec wyrażenia. Miejscom tym przyporządkowuje się stany automatu.

$$S_1 = \begin{array}{c|c|c|c|c} 0 & z_1 & 1 & z_2 & 2 \\ \hline \end{array} + \begin{array}{c|c|c|c|c} 0 & z_1 & 1 & z_1 & 3 \\ \hline \end{array} z_1 \begin{array}{c|c} 4 \\ \hline \end{array}$$

$$S_2 = \begin{array}{c|c|c|c|c} 0 & z_1 & 1 & z_2 & 2 \\ \hline \end{array} z_2 \begin{array}{c|c} 5 \\ \hline \end{array} + \begin{array}{c|c|c|c} 0 & z_2 & 6 & z_2 \\ \hline \end{array} 7$$

Dla uproszczenia rozpatrujemy automat Moore'a.

tablica 1

wyjście	y_3	y_3	y_1	y_3	y_1	y_2	y_3	y_2	y_3
stany wejście	0	1	2	3	4	5	6	7	8
z_1	1	3	*	4	*	*	*	*	*
z_2	6	2	5	*	*	*	7	*	*



* - do powyższego zapisu wprowadzamy stan dodatkowy, do którego automat przechodzi, gdy pojawi się słowo należące do S_3 (w tablicy 1 stan 8 to *)

Po otrzymaniu tablicy 1, przeprowadzamy minimalizację tablicy stanów $5 \equiv 7$, tworząc tablicę 2

tablica 2

wyjście	y_3	y_3	y_1	y_3	y_1	y_2	y_3	y_3
stany wejście	q_0	q_1	q_2	q_3	q_4	q_5	q_6	q_7
z_1	q_1	q_3	q_7	q_4	q_7	q_7	q_7	q_7
z_2	q_6	q_2	q_5	q_7	q_7	q_7	q_5	q_7

Na podstawie tablicy 2 można narysować graf automatu.

Wyznaczenie stanów automatów staje się bardziej złożone, gdy w wyrażeniu regularnym występuje operacja iteracji. W tym przypadku wyrażenie regularne dzieli się na miejsca „podstawowe” i „przedpodstawowe”.

Miejscami „podstawowymi” nazywamy te miejsca w wyrażeniu regularnym, na lewo od których stoi litera oraz miejsce początkowe.

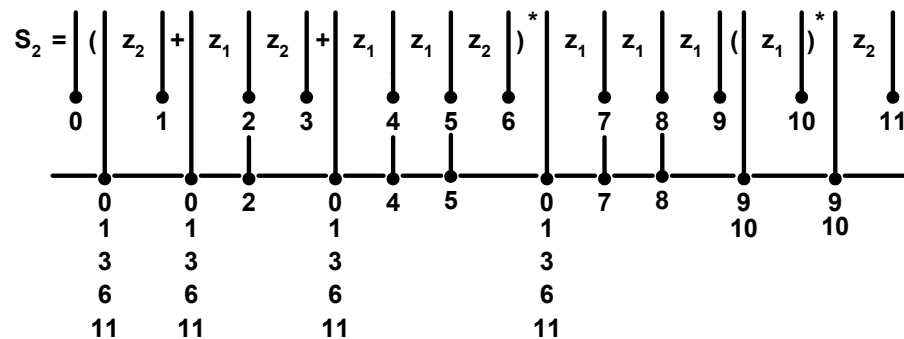
Miejscem „przedpodstawowym” nazywamy te miejsca w wyrażeniu regularnym, na prawo od których stoi litera.

Przykład 2

$$S_1 = (z_2 + z_1 z_2 + z_1 z_1 z_2)^* z_1 z_1 z_1 \mid y_1$$

$$S_2 = (z_2 + z_1 z_2 + z_1 z_1 z_2)^* z_1 z_1 z_1 (z_1)^* z_2 \mid y_2$$

$$S_3 = S_1 + S_2 \mid y_3 = \varepsilon$$



Miejsca „przedpodstawowe” oznacza się odpowiednimi symbolami miejsc „podstawowych”. Stosujemy następujące reguły:

Reguła 1

Symbole miejsca „podstawowego” przed nawiasem iteracyjnym rozmieszcza się w miejscach „przedpodstawowych” we wszystkich miejscach początkowych wszystkich członów dysjunktywnych stojących w danym nawiasie.

Reguła 2

Symbol miejsca końcowego dowolnego członu dysjunktywnego zamkniętego w nawiasy iteracyjne rozmieszczamy w miejscach początkowych („przedpodstawowych”) wszystkich członów dysjunktywnych zamkniętych w danym nawiasie.

Reguła 3

Symbole miejsc „podstawowych”, na lewo i prawo od których stoją litery nie rozmieszcza się nigdzie więcej.

Reguła 4

Symbol miejsca końcowego wyrażenia rozmieszcza się we wszystkich tych miejscach „przedpodstawowych”, gdzie znajduje się symbol miejsca początkowego.

Reguła 5

Symbol miejsca końcowego dowolnego członu dysjunktywnego zamkniętego w nawiasy iteracyjne rozmieszcza się w miejscu „przedpodstawowym” bezpośrednio za danym nawiasem.

Reguła 6

Symbol miejsca przed nawiasem iteracyjnym zapisuje się w miejscu „przedpodstawowym” znajdującym się za tym nawiasem. Następnie przeprowadza się minimalizację stanów. Jeżeli kilka miejsc „przedpodstawowych” oznakowane jest jednakowym zbiorem symboli i na prawo od tych miejsc zapisane są takie same litery, to wówczas miejsca „podstawowe” położone na prawo od tych liter są sobie równoważne.

Wracając do przykładu 2:

Stany $2 \equiv 4 \equiv 7$ są sobie równoważne ($4, 7 \rightarrow 2$)
 $5 \rightarrow 4, 6 \rightarrow 5, 8 \rightarrow 6, 9 \rightarrow 7, 10 \rightarrow 8, 11 \rightarrow 9$

W następnej fazie $4 \equiv 6$ są sobie równoważne
 $7 \rightarrow 6, 8 \rightarrow 7, 9 \rightarrow 8$

Tworzy się tablicę przejść:

tablica 3

wyjście	y_3	y_3	y_3	y_3	y_3	y_3	y_1	y_3	y_2
<u>stany</u> wejście	0	1	2	3	4	5	6	7	8
z_1	2	2	4	2	6	2	7	7	2
z_2	1	1	3	1	5	1	8	8	1

$0 \equiv 1 \equiv 3 \equiv 5$

5. PRZEBIEG ĆWICZENIA

Ćwiczenie przeprowadzane jest z wykorzystaniem zestawu AUT-01. Na płycie czołowej modelu znajduje się 13 lampek sygnalizacyjnych. Lampka umieszczona nad przełącznikiem klawiszowym opisanym MAINS sygnalizuje podłączenie napięć zasilających. Ośmiu lampek sygnalizacyjnych nad przełącznikami STATE 0 – STATE 7 służy do sygnalizacji stanów automatu, 4 lampki OUTPUT INDICATORS służy do sygnalizacji wyjść automatu oznaczonych symbolami $Y_0 - Y_3$.

Elementy manipulacyjne

Przełącznik MAINS służy do przyłączenia napięć zasilających. Zespół czterech przełączników klawiszowych $X_0 - X_3$ służy do wprowadzania liter alfabetu wejściowego modelowanego automatu. Wciśnięcie dowolnego z przycisków $X_0 - X_3$ odpowiada wprowadzeniu pojedynczej litery alfabetu wejściowego. Na jednym z gniazd FUNCTION X&S pojawia się sygnał utworzony z iloczynu stanu i wprowadzanej litery. Sygnał ten doprowadzony do odpowiednich gniazd STATE EXCITATION lub do gniazd OUTPUT EXCITATION powoduje wzbudzenie następnego stanu, wyzerowanie poprzedniego lub wzbudzenie odpowiedniego wyjścia. Zespół ośmiu przełączników STATE0 – STATE7 służy do ustawiania stanów początkowych automatu. Naciśnięcie dowolnego z tych przełączników powoduje ustawienie nowej zawartości rejestru stanu i wyzerowanie rejestru wyjść.

Gniazda opisane FUNCTION X&S są przeznaczone do łączenia przewodami z gniazdami STATE EXCITATION w celu wzbudzenia stanów i wyjść automatu. Każda

para gniazd FUNCTION X&S umiejscowiona na przecięciu wiersza X_n alfabetu wejściowego X z kolumną S_n stanów S odpowiada iloczynowi X_n & S_n .

Każdemu z ośmiu stanów STATE0 – STATE7 odpowiadają 24 gniazda wzbudzenia stanów STATE EXCITATION.

Gniazda OUTPUT EXCITATION umożliwiają doprowadzenie sygnałów wzbudzających do układu wzbudzenia wyjść automatu.

Sygnały wzbudzające są doprowadzone z gniazd FUNCTION X&S w przypadku automatu Moore'a. Każdemu z czterech wyjść odpowiada 16 gniazd wzbudzenia oznaczonych Y0 – Y3.

Gniazda STATE0 – STATE7 umieszczone w dolnej części płyty czołowej służą do wyprowadzenia z nich sygnałów wzbudzających wyjścia w przypadku automatu Moore'a.

Obsługa modelu:

1. Podłączyć przyrząd do sieci.
2. Włączyć napięcia zasilania przełącznikiem MAINS.
3. W momencie załączenia ustawienie stanów i wyjść jest przypadkowe. Przeprowadzić zerowanie wszystkich stanów i ustawić wymagany stan początkowy jednym z przełączników STATE0 – STATE7.
4. Dla zbudowania modelu automatu połączyć gniazda na płycie czołowej zgodnie z zadanymi tabelami przejść i wyjść. Programowanie przejść polega na połączeniu gniazd FUNCTION X&S z odpowiednimi gniazdami STATE EXCITATION. Na przykład, aby dla litery wejściowej X_1 uzyskać przejście $S5 \rightarrow S3$ należy połączyć gniazdo FUNCTION X_1 & S5 z gniazdem S3 z kolumny STATE EXCITATION. Programowanie wyjść dla automatu Mealy jest analogiczne jak programowanie przejść.

Przejście automatu ze stanu do stanu i wzbudzenie wyjść zachodzi każdorazowo po wprowadzeniu pojedynczej litery alfabetu wejściowego. Należy to zrealizować przez wciśnięcie odpowiedniego przełącznika X0 – X3. W danym momencie na wejście automatu może być podana tylko jedna litera alfabetu wejściowego, wzbudzony jeden stan i jedno wyjście.

6. ZADANIA DO WYKONANIA

Część I

1. Dokonać syntezy automatu Moore'a na podstawie wyrażenia regularnego podanego przez prowadzącego
2. Zamodelować automat na stanowisku.
3. Określić sekwencje rozpoznawane przez automat poprzez zadawanie sekwencji zdarzeń na wejściu.
4. Określić tablice przejść automatu Moore'a i narysować graf tego automatu.
5. Zanalizować w sposób formalny, czy automat realizuje przedstawione wyrażenie regularne.

Część II

1. Napisać wyrażenie regularne opisujące działanie zamka szyfrowego.
2. Zamodelować automat na stanowisku.
3. Rozpoznać sekwencję zdarzeń powodujących alarm zamka szyfrowego.
4. Przeprowadzić analizę automatu ze względu na otwarcie zamka.
5. Określić intuicyjnie definicję negacji wyrażenia regularnego na podstawie porównać obu wyrażeń.

Część III

1. Zamodelować dowolny automat na stanowisku.
2. Określić wyrażenie regularne reprezentowane przez ten automat.
3. Dokonać syntezy automatu na podstawie określonego w 2. wyrażenia.
4. Udowodnić, że oba automaty rozpoznają te same sekwencje zdarzeń.
5. Określić, które stany automatów są równoważne na podstawie analizy modeli abstrakcyjnych.

7. SPRAWOZDANIE Z ĆWICZENIA

W sprawozdaniu należy umieścić:

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

LITERATURA

1. J. Bromirski, Teoria automatów, WNT, Warszawa, 1969
2. J. Kazimierczak, J. Kluska, A. Kaczmarek, Podstawy teorii automatów – laboratorium, Wydawnictwa Politechniki Rzeszowskiej, Rzeszów, 1984
3. E. N. Wawiłow, G. P. Portnoj, Synteza układów elektronicznych maszyn cyfrowych, WNT, Warszawa, 1967