

Jerzy MIKULSKI

Krzysztof ZYCH

METODY PROGRAMOWEJ GENERACJI ZALEŻNOŚCI LOGICZNYCH W TECHNICIE STEROWANIA RUCHEM KOLEJOWYM

Streszczenie. Artykuł przedstawia po omówieniu programowej generacji układów automatyki cyfrowej opis realizacji komputerowej podstawowych automatów wykorzystywanych w układach zależnościowych urządzeń sterowania ruchem kolejowym.

Uwzględniając specyfikę wynikającą z synchronicznego charakteru zmian sygnałów podczas realizacji komputerowej przeprowadzono syntezę automatu sygnałowego, manewrowego, utwierdzenia i sygnalizatora świetlnego. Dla każdego z automatów przedstawione zostały logiczne funkcje wyjściowe, które mogą być realizowane dowolną z przedstawionych metod programowej generacji zależności logicznych.

METODY ПРОГРАМНОГО ГЕНЕРИРОВАНИЯ ЛОГИЧЕСКИХ ЗАВИСИМОСТЕЙ В ТЕХНИКЕ ТЕЛЕУПРАВЛЕНИЯ СТРЕЛКАМИ И СИГНАЛАМИ

Резюме. Статья представляет описание компьютерной реализации основательных автоматов, использованных в схемах телеуправления стрелками и сигналами.

Учитывая специфику, которая возникает из синхронического характера изменений сигналов во время компьютерной реализации проведено синтез сигнального автомата, маневрового, замыкающего и светофора. Для каждого из автоматов представлено логические выходные функции, которые могут быть реализованы любым из представленных методов программного генерирования логических зависимостей.

METHODS FOR PROGRAMMED GENERATION OF LOGIC RELATIONS IN RAILWAY TRAFFIC CONTROL ENGINEERING

Summary. The paper discusses a programmed generation of automatic digital control systems and computer control of

basic automatic devices used in slave systems of railway traffic control devices.

Bearing in mind a synchronous nature of signal changes during the computer control, a synthesis of signalling, switching and confirmation automatic devices was carried out. For each automatic device, starting logical functions that can be executed by any presented method for programmed generation of logic relations are discussed.

Analiza zależności logicznych w przypadku programowej (komputerowej) generacji zależności logicznych odbywa się zgodnie z zasadami stosowanymi dla typowych układów automatyki [6]. Ze względu na synchroniczny charakter zmian sygnałów, w trakcie komputerowej generacji zależności logicznych możliwe jest przyjęcie zasad projektowania obowiązujących dla układów synchronicznych, pomijając tym samym zjawisko hazardu. Należy jednak eliminować możliwość powstania wyścigu w przypadku układów sekwencyjnych.

Wśród metod generowania zależności logicznych można wyróżnić trzy zasadnicze grupy [5]:

- Metoda bezpośrednia - charakteryzująca się tym, że dla każdego wyrażenia logicznego tworzony jest oddzielny program.
- Metody tablicowe, w których zdefiniowanie realizowanych funkcji logicznych wiąże się z wypełnieniem określonej tablicy wartościami funkcji w sposób zbliżony do stosowanego w konwencjonalnej teorii automatów. Program obsługujący te tablice nie zależy od realizowanych zależności i dzięki temu jest uniwersalny.
- Metody widmowe związane z rozwinięciem funkcji logicznej w szereg, które trzeba zaliczyć także do specyficznych metod tablicowych.

W metodzie bezpośredniej, wychodząc ze zminimalizowanego wyrażenia logicznego określającego żądane zależności, tworzy się program bezpośrednio wykorzystujący rozkazy logiczne. W wielu przypadkach program może działać szybciej i zajmować mniej miejsca w pamięci operacyjnej niż w przypadkach metod uniwersalnych, dotyczy to jednak automatów o niewielkiej liczbie wejść.

Spośród metod tablicowych najczęściej wykorzystywane są: metoda przeglądania tablic i metoda adresowa.

W pierwszej z nich wyrażenia logiczne doprowadza się do normalnej postaci sumy (lub iloczynu) i zapamiętuje się w tablicy implikanty (lub impliency), których suma (lub iloczyn) reprezentują rozważane wyrażenie logiczne. Wartość słowa zmiennych wejściowych jest porównywana z kolejnymi elementami tablicy¹. Jeśli któryś z elementów tablicy jest identyczny ze słowem wejściowym, generowane wyrażenie przyjmuje wartość jedynki logicznej (lub w przypadku impliency zera logicznego). Jeżeli nie znaleziono w tablicy słowa zgodnego ze słowem wejściowym, wyrażenie ma wartość zera (jedynki) logicznego. Jest to metoda bardzo nieefektywna, ponieważ czas pracy zależy nie tylko od wielkości tablicy (liczby implikantów lub impliency), ale także od słowa wejściowego. Ponieważ jednak zależności logiczne, stosowane w przemysłowych układach sterowania należą do klasy funkcji zwanych funkcjami słabo określonymi, czyli zawierającymi wiele stanów obojętnych (nieokreślonych), metodę tę można zmodyfikować korzystając ze zminimalizowanej postaci wyrażenia logicznego. Przeglądana tablica będzie zawierać wtedy implikanty (impliency) proste uzupełnione o słowo maski, informujące, które zmienne wejściowe w nim występują [5].

W metodzie adresowej wartości wyrażeń są zapamiętywane w tablicy, podobnie jak przy realizacji zależności logicznych przy korzystaniu z pamięci stałych ROM. Słowo wejściowe zestawiane tak samo, jak poprzednio z poszczególnych zmiennych stanowi adres, pod którym są zapamiętywane wartości generowanych wyrażeń. Tablica wartości wyrażeń logicznych odpowiada tutaj wprost tablicy zależności (prawdy) w klasycznej teorii automatów. Założenia odnośnie do przekazywania parametrów

¹W przedstawianych metodach zakłada się, że sygnały wejściowe (zmienne logiczne) są zestawiane w jedno słowo o długości jednego (lub więcej) bajtu, w którym każdy bit reprezentuje jedną zmienną.

wejściowych, jak i wyjściowych są takie same, jak w poprzedniej metodzie [5].

AUTOMATY SYGNAŁOWE

Zadaniem automatu sygnałowego jest kontrola następujących zależności:

- właściwe położenie wszystkich zwoznic uczestniczących w przebiegu, to znaczy zwoznic, przez które przejeżdża pociąg, zwoznic w drodze ochronnej i zwoznic oraz wykolejnic ochronnych,
 - niezajętość odcinków kontrolowanych torowych i zwoznicowych leżących w drodze przebiegu pociągu, drodze ochronnej oraz odcinków kontrolowanych wchodzących w skrajnię taboru drogi przebiegu,
 - wykluczenie przebiegów sprzecznych,
- a ponadto sprawdzenie, czy wymagane jest nastawienie przebiegu (kontrola naciśnięcia przycisków nastawczych).

Automat sygnałowy jest automatem sekwencyjnym, asynchronicznym, w swojej elementarnej postaci przyporządkowanym jednemu przebiegowi p_1 [4]. W praktycznych rozwiązaniach układów przekąźnikowych występują najczęściej układy sygnałowe wspólne dla grup przebiegów, jednak w proponowanym systemie przyjęto zasadę przyporządkowania każdemu przebiegowi oddzielnego automatu sygnałowego, co ułatwia matematyczną analizę i pozwala na tworzenie systemu uniwersalnego.

Zasadniczymi elementami wejściowymi automatu sygnałowego przyporządkowanego przebiegowi p_1 są:

- wartość funkcji $P_z(Z_1)$ określająca zgodność rzeczywistego położenia zwoznic z wymaganym (w układach przekąźnikowych są to przekąźniki kontroli położenia zwoznic oznaczone zazwyczaj skrótem Kn),
- wartość funkcji $Z_0(T_1)$ określająca zgodność rzeczywistego stanu obwodów torowych z wymaganym (w układach przekąźnikowych są to przekąźniki torowe oznaczane skrótem IT lub IZ),

- wartość funkcji $U_S(p_1)$ (w przypadku przebiegów sekcjonowanych funkcji $U'_S(p_1)$) określająca brak utwierdzenia przebiegu sprzecznego z przebiegiem p_1 (w układach przekaźnikowych są to przekaźniki utwierdzenia oznaczone symbolem U),
- przyciski nastawcze².

Zmiany stanów wejściowych można przedstawić dzieląc umownie elementy wejściowe na:

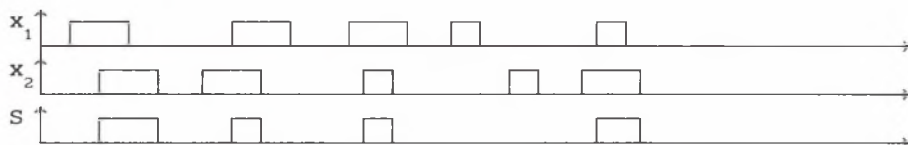
- x_1 - elementy, o zmianie których decyduje nastawniczy,
- x_2 - elementy, których zmiany następują automatycznie.

Do pierwszej grupy należą tylko przyciski (w układach przekaźnikowych są to również elementy pośredniczące w przekazywaniu informacji o stanach przycisków), do drugiej - wszystkie pozostałe elementy wejściowe [1].

Działanie automatu polega najogólniej na sprawdzeniu, czy został wygenerowany sygnał $x_2 = 1$ określający właściwy stan urządzeń uczestniczących w nastawianym przebiegu i podany przez dyżurnego ruchu sygnał $x_1 = 1$ wymuszający nastawienie przebiegu. Sygnały te powodują pojawienie się sygnału wyjściowego $S = 1$, który umożliwia wyświetlenie światła zezwalającego na sygnalizatorach (semafor, powtarzacze, tarcza ostrzegawcza lub tarcza manewrowa) związanych z daną drogą przebiegu. Przejście do stanu $S = 0$ następuje w momencie zmiany stanu urządzeń uczestniczących w nastawianym przebiegu, to znaczy w chwili, gdy sygnał wejściowy $x_2 = 0$. Na rys.1 przedstawiony został wykres czasowy programu pracy automatu sygnałowego, a na rys. 2, 3 i 4 - projekt automatu sygnałowego.

²W przypadku stosowania blokady stacyjnej oraz blokady półsamoczynnej sygnałami wejściowymi są również:

- przekaźniki blokady stacyjnej pośredniczące w przenoszeniu zależności z innych okręgów nastawczych tej samej stacji (oznaczone skrótami On lub Oz),
- przekaźniki lub bloki blokady liniowej pośredniczące w przenoszeniu zależności z obiektów współpracujących z daną stacją (oznaczone skrótami Po, Poz, Pw1).



Rys.1. Wykres czasowy programu pracy automatu sygnałowego

Fig.1. A relation of the program of signalling automatic device operation vs. time

 $x_1 x_2$

	00	01	11	10	S
1	1	5	-	2	0
1	1	-	3	2	0
-	-	4	3	-	1
1	1	4	3	-	1
1	1	5	3	-	0

□ - stan stabilny

Rys.2. Tablica programu automatu sygnałowego

Fig.2. A table of the program for signalling automata device

Przedstawiony automat sygnałowy zapewnia poprawne działanie w przypadku przebiegów pociągowych, natomiast podczas realizacji przebiegów manewrowych wymagane jest, aby sygnał wyjściowy przechodził do stanu biernego ($S = 0$) dopiero po opuszczeniu pierwszego odcinka kontrolowanego za sygnalizatorem. Taka zasada działania zabezpiecza przed przedwczesnym

osłonięciem drogi przebiegu sygnałem zabraniającym³.

$x_1 x_2$		00	01	11	10	S
	1	1	5	3	2	0
	0	1	4	3	2	1

Rys.3. Zredukowana tablica programu automatu sygnałowego (dla automatu Moore'a)

Fig.3. A reduced table of the program for signalling automatic device (for Moore's automata)

$x_1 x_2$		00	01	11	10	S
q	0	0	0	1	0	0
	1	0	1	1	0	1

$$Q = x_1 x_2 + q x_2$$

$$S = Q$$

Q

Rys.4. Siatka stanów automatu sygnałowego

Fig.4. A network of the states of signalling automatic device

Z tego względu w celu realizacji przebiegów manewrowych został skonstruowany osobny automat sygnałowy manewrowy (zwany dalej automatem manewrowym). Sygnałami wejściowymi automatu

³Podczas manewrów lokomotywa może znajdować się na końcu składu, dlatego wyświetlenie sygnału zabraniającego po wjechaniu taboru na pierwszy odcinek drogi przebiegu powodowałoby, że maszynista otrzymałby błędną informację nakazującą zatrzymanie taboru.

manewrowego są, podobnie jak w przypadku automatu sygnałowego, umowne elementy wejściowe:

x_1 - elementy, o zmianie których decyduje nastawniczy,

x_2 - elementy, których zmiany następują automatycznie.

oraz

t - wartości funkcji zajętości odcinka kontrolowanego za sygnalizatorem,

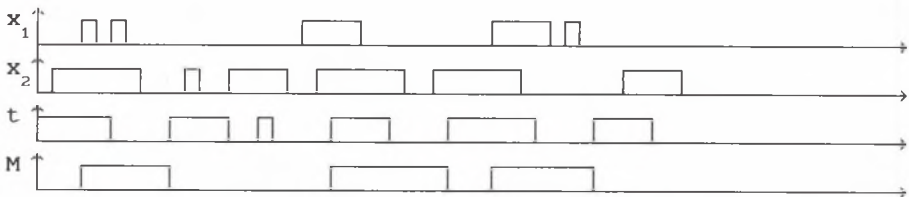
- $t=0$ - odcinek zajęty,

- $t=1$ - odcinek wolny.

W ten sposób sygnały $x_1 = 1$, $x_2 = 1$ i $t = 1$ powodują pojawienie się sygnału wyjściowego $M = 1$, który umożliwia wyświetlenie sygnału zezwalającego na sygnalizatorze manewrowym. Przejście do stanu biernego $M = 0$ wymaga pojawienia się sekwencji:

- zajęcie przez tabor odcinka kontrolowanego ($t=1 \rightarrow t=0$),

- zwolnienie przez tabor odcinka kontrolowanego ($t=0 \rightarrow t=1$).



Rys.5. Wykres czasowy programu pracy automatu manewrowego

Fig.5. A relation of the program of switching automatic device operation vs. time

$x_1 x_2 t$	000	001	011	010	110	111	101	100	M
7	1	2	-	-	-	-	8	-	0
-	1	2	9	-	3	-	-	-	0
-	-	4	-	10	3	14	-	-	1
-	16	4	5	-	3	-	-	-	1
6	-	4	5	10	-	-	-	-	1
6	1	-	9	-	-	-	-	15	1
7	1	-	9	-	-	-	-	11	0
-	1	-	-	-	3	-	8	11	0
7	-	2	9	12	-	-	-	-	0
-	-	-	5	10	3	-	-	15	1
7	-	-	-	12	-	-	8	11	0
-	-	-	9	12	3	-	-	11	0
-	-	2	-	12	13	8	-	-	0
-	16	-	-	-	3	-	14	15	1
6	-	-	-	12	-	-	8	15	1
6	16	4	-	-	-	-	14	-	1

Rys. 6. Tablica programu automatu manewrowego

Fig. 6. A table of the program for switching automata

$x_1 x_2 t$		000	001	011	010	110	111	101	100	M
		7	1	2	9	12	3	3	11	0
	6	16	4	3	10	3	14	15	1	
	6	1	2	9	12	13	8	15	1	
	7	1	2	9	12	13	8	11	0	

Rys.7. Zredukowana tablica programu automatu manewrowego (dla automatu Moore'a)

Fig.7. A reduced table of the program for switching automatic device (for Moore's automata)

$x_1 x_2 t$		000	001	011	010	110	111	101	100	M
$q_1 q_2$	0 0	7	1	2	9	12	3	8	11	0
	0 1	6	16	4	5	10	3	14	15	1
	1 1	6	1	2	9	12	13	8	15	1
	1 0	7	1	2	9	12	13	8	11	0

Rys.8. Zakodowana tablica programu automatu manewrowego (dla automatu Moore'a)

Fig.8. A coded table of the program for switching automatic device (for Moore's automata)

$q_1 q_2$	$x_1 x_2 t$								M
	000	001	011	010	110	111	101	100	
0 0	00	00	00	00	00	01	00	00	0
0 1	11	01	01	01	01	01	01	11	1
1 1	11	10	10	10	10	10	10	11	1
1 0	00	00	00	00	00	10	00	00	1

$q_1 q_2$	$x_1 x_2 t$								M
	000	001	011	010	110	111	101	100	
0 0	0	0	0	0	0	0	0	0	0
0 1	1	0	0	0	0	0	0	1	1
1 1	1	1	1	1	1	1	1	1	1
1 0	0	0	0	0	0	1	0	0	0

$$Q_1 = q_2 \bar{x}_2 \bar{t} + q_1 q_2 + q_1 x_1 x_2 t \quad Q_1$$

$q_1 q_2$	$x_1 x_2 t$								M
	000	001	011	010	110	111	101	100	
0 0	0	0	0	0	0	1	0	0	0
0 1	1	1	1	1	1	1	1	1	1
1 1	1	0	0	0	0	0	0	1	1
1 0	0	0	0	0	0	0	0	0	0

$$Q_2 = q_2 \bar{x}_2 \bar{t} + \bar{q}_1 q_2 + \bar{q}_1 x_1 x_2 t; \quad M = Q_2 \quad Q_2$$

Rys. 9. Siatki stanów automatu manewrowego

Fig. 9. A network of the states of switching automatic device

Wykres czasowy programu pracy automatu manewrowego przedstawiono na rys.5, a na rys.6, 7, 8 i 9 - projekt automatu manewrowego. Liczby podkreślone w zredukowanej tablicy programu oznaczają stany, które z punktu widzenia klasycznej automatyki są stanami obojętnymi (nieosiągalne w automatach sekwencyjnych asynchronicznych). Podczas realizacji komputerowej, ze względu na możliwość powstania błędu odczytu pamięci, stany te zostały określone, aby można było przewidzieć stan automatu po zaistnieniu takiego błędu.

AUTOMAT UTWIERDZENIA

Każda zwrotnica opisana elementem należącym do zbioru Z_1 musi być utwierdzona w przebiegu p_1 , aż do momentu zakończenia jazdy na odcinku od początku drogi przebiegu dp_1 do rozważanej zwrotnicy włącznie.

W niektórych systemach sterowania ruchem kolejowym przebiegi są zamykane i utwierdzane, w innych tylko utwierdzane. Zamknięcie różni się od utwierdzenia tym, że jego cofnięcie nie jest kontrolowane. W proponowanym systemie wszystkie przebiegi zarówno pociągowe, jak i manewrowe, są utwierdzane.

Automat utwierdzenia jest automatem sekwencyjnym asynchronicznym, w swojej elementarnej postaci przyporządkowanym jednemu przebiegowi p_1 . W rozwiązaniach praktycznych występują zwykle układy utwierdzenia wspólne dla grup przebiegów, natomiast w przedstawianym systemie każdy przebieg ma oddzielny automat utwierdzenia.

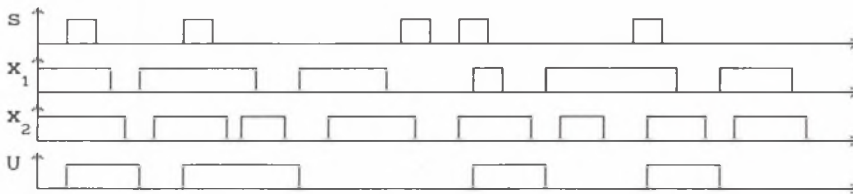
Zasadniczymi elementami wejściowymi automatu utwierdzenia są [1]:

- sygnał wyjściowy automatu sygnałowego oznaczony symbolem S (w układach przekaźnikowych jest to przekaźnik sygnałowy),
- sygnały wyjściowe elementów kontrolujących przemieszczanie się taboru po drodze jazdy d_1 (wartości funkcji zajętości odcinków kontrolowanych); w układzie elementarnym dla dwóch sekcji są to dwa elementy, oznaczone jako x_1 i x_2 , gdzie x_1

jest poprzednim w stosunku do x_2 (w układach przekaźnikowych są to przekaźniki torowe);

- stan przycisku ręcznego zwolnienia sekcji przebiegu.

Zakładając, że utwierdzenie przebiegu jest jednostopniowe, wyjście automatu przechodzi do stanu aktywnego (utwierdzenie) $U = 1$ wówczas, gdy zostanie wygenerowany sygnał wyjściowy automatu sygnałowego $S = 1$ (oznaczony jako wejście s automatu utwierdzenia). Sygnał ten umożliwia wyświetlenie światła zezwalającego na sygnalizatorach dotyczących nastawianej drogi przebiegu, jeśli wartości funkcji zajętości odcinków kontrolowanych (w praktycznych zastosowaniach są to najczęściej sygnały wyjściowe przekaźników torowych kontrolujących przemieszczanie się taboru) wskazują na ich niezajętość ($x_1 = 1$ i $x_2 = 1$).



Rys.10. Wykres czasowy programu pracy automatu utwierdzenia

Fig.10. A relation of the program of confirmation automatic device operation vs. time

s	x_1	x_2	000	001	011	010	110	111	101	100	U
-	7	[1]	6	-	2	-	-	0			
-	-	3	-	8	[2]	9	-	1			
-	4	[3]	10	-	2	-	-	1			
5	[4]	3	-	-	-	9	-	1			
[5]	4	-	6	-	-	-	14	1			
11	-	1	[6]	12	-	-	-	0			
11	[7]	1	-	-	-	13	-	0			
-	-	-	10	[8]	2	-	14	1			
-	17	-	-	-	2	[9]	14	1			
15	-	3	[10]	8	-	-	-	1			
[11]	7	-	6	-	-	-	16	0			
-	-	-	6	[12]	2	-	16	0			
-	7	-	-	-	2	[13]	16	0			
15	-	-	-	8	-	9	[14]	1			
[15]	17	-	10	-	-	-	14	1			
11	-	-	-	12	-	13	[16]	0			
15	[17]	3	-	-	-	9	-	1			

Rys.11. Tablica programu automatu utwierdzenia

Rys.11. A table of the program for confirmation automatic device

s x_1 x_2

000	001	011	010	110	111	101	100	U
[11]	[7]	[1]	[6]	[12]	2	[13]	[16]	0
5	[4]	[3]	10	[8]	[2]	9	14	1
[5]	4	<u>3</u>	6	<u>8</u>	<u>2</u>	<u>9</u>	14	1
[15]	[17]	3	[10]	8	2	[9]	[14]	1

Rys.12. Zredukowana tablica programu automatu utwierdzenia (dla automatu Moore'a)

Fig.12. A reduced table of the program for confirmation automatic device (for Moore's automata)

s x_1 x_2

q_1q_2	000	001	011	010	110	111	101	100	U
00	[11]	[7]	[1]	[6]	[12]	2	[13]	[17]	0
01	[15]	[19]	3	[10]	8	2	[9]	[14]	1
11	5	[4]	[3]	10	[8]	[2]	9	14	1
10	[5]	4	3	6	8	2	9	14	1

Rys.13. Zakodowana tablica programu automatu utwierdzenia (dla automatu Moore'a)

Fig.13. A coded table of the program for confirmation automatic device (for Moore's automata)

		s x ₁ x ₂								
q ₁ q ₂		000	001	011	010	110	111	101	100	U
0 0		0	0	0	0	0	1	0	0	0
0 1		0	0	1	0	1	1	0	0	1
1 1		1	1	1	0	1	1	0	0	1
1 0		1	1	1	0	1	1	1	1	1

Q₁

$$Q_1 = s x_1 x_2 + q_2 s x_1 + q_1 \bar{s} x_2 + q_1 \bar{s} \bar{x}_1 + q_2 x_1 x_2 + q_1 \bar{q}_2 s$$

		s x ₁ x ₂								
q ₁ q ₂		000	001	011	010	110	111	101	100	U
0 0		0	0	0	0	0	1	0	0	0
0 1		1	1	1	1	1	1	1	1	1
1 1		0	1	1	1	1	1	1	1	1
1 0		0	1	1	0	1	1	1	1	1

Q₂

$$Q_2 = \bar{q}_1 q_2 + q_1 s + q_2 x_1 + q_1 x_2 + s x_1 x_2$$

$$U = Q_1 + Q_2$$

Rys. 14. Siatka stanów automatu utwierdzenia

Fig. 14. A network of the states of confirmation automatic device

Przejście automatu utwierdzenia do stanu biernego ($U = 0$) następuje w momencie pojawienia się następującej sekwencji stanu wejść:

- zajęcie przez tabor odcinka kontrolowanego przez przekaźnik x_1 ($x_1=1 \rightarrow x_1=0$),
- zajęcie przez tabor odcinka kontrolowanego przez przekaźnik x_2 ($x_2=1 \rightarrow x_2=0$),
- zwolnienie przez tabor odcinka kontrolowanego przez przekaźnik x_1 ($x_1=0 \rightarrow x_1=1$).

Na rys. 10 przedstawiony został wykres czasowy programu pracy automatu utwierdzenia, a na rys. 11, 12, 13 i 14 - projekt automatu utwierdzenia.

AUTOMAT SYGNALIZATORA ŚWIETLNEGO

Sygnalizatory świetlne służą do przekazywania w sposób zakodowany informacji określających prędkość pojazdów. Informacje te są odbierane przez obsługę pojazdów, umożliwiając jej utrzymanie bezpiecznej prędkości jazdy. Podstawowym zadaniem każdego sygnalizatora jest określenie, czy wjazd na odcinek układu torowego osłaniany nim (lub najbliższym semaforem w przypadku tarcz ostrzegawczych lub sygnalizatorów powtarzających) jest dozwolony. Dla tak opisanego przypadku można automat sygnalizatora traktować jako automat kombinacyjny, którego sygnałami wejściowymi są:

- funkcja stanu automatu sygnałowego (S) oznaczona jako wejście s,
- funkcja stanu automatu utwierdzenia (U) oznaczona jako wejście u.

Sygnałem wyjściowym jest natomiast sygnał (Z) określający, czy wyświetlony ma być sygnał zezwalający ($Z=0$) czy zabraniający na wjazd na odcinek toru osłaniany rozważanym semaforem (lub najbliższym semaforem). Sygnał zezwalający może być podany tylko w przypadku spełnienia wymaganych zależności określonych działaniem automatów sygnałowego (wejście $s=1$) i

utwierdzenia (wejście $u=0$). Tablica zależności automatu sygnalizatora przedstawiona jest na rys.15.

s	u	Z
0	0	0
0	1	0
1	0	1
1	1	0

Rys.15. Tablica zależności automatu sygnalizatora

Fig.15. Truth table of dependence semaphore's automata

Bezpośrednio z tablicy zależności można wyznaczyć funkcję logiczną opisującą warunki wyświetlenia sygnału zezwalającego:

$$Z = s\bar{u}$$

Oznacza to, że sygnał zezwalający może być wyświetlony tylko w przypadku, gdy wszystkie urządzenia sterowania ruchem kolejowym są w stanie zapewniającym bezpieczeństwo jazdy, odcinki kontrolowane w przebiegu są wolne, a przebieg jest utwierdzony.

Przedstawione rozwiązania dotyczące podstawowych automatów sekwencyjnych stosowanych w systemach sterowania ruchem kolejowym mogą być realizowane dowolną z przedstawionych na wstępie metod komputerowej generacji zależności logicznych lub za pomocą tradycyjnych układów synchronicznych. W rozwiązaniach uwzględniono również stany, które teoretycznie są stanami niemożliwymi, mogącymi jednak wystąpić w wyniku nieprawidłowego działania systemu komputerowego. Rozwiązania są więc nadmiarowe w stosunku do spotykanych w literaturze, przez co podwyższa się poziom bezpieczeństwa procesu sterowania.

LITERATURA

- [1] Apuniewicz S.: Zasady modelowania matematycznego obiektów i procesów sterowania ruchem kolejowym, ZN Pol. Świętokrz., Kielce 1977, z 5.
- [2] Mikulski J.: Koncepcja bezstykowych bloków sterowania ruchem kolejowym, ZN Pol. Śl., s. Transport, Gliwice 1986, z. 5.
- [3] Mikulski J.: Realizacja bezstykowych bloków sterowania ruchem kolejowym, ZN Pol. Śl., s. Transport, Gliwice 1986, z. 5.
- [4] Mikulski J., Zych K.: Opis formalny obiektów sterowania ruchem kolejowym, {niniejszy zeszyt}.
- [5] Plaza R., Wróbel E.: Systemy czasu rzeczywistego, WNT, Warszawa 1988.
- [6] Siwiński J.: Automatyka układów przełączających, WNT, Warszawa 1980.

Recenzent: Doc. dr inż. Zbigniew Ginalski

Wpłynęło do Redakcji 22.10.1993 r.

Abstract

The paper discusses a programmed generation of automatic digital control systems and computer control of basic automatic devices used in slave systems of railway traffic control devices.

The sequential systems are disqued by the table of programs with solution of Moore's automata.

Bearing in mind a synchronous nature of signal changes during the computer control, a synthesis of signalling, switching and confirmation automatic devices was carried out. For each automatic device, starting logical functions that can

be executed by any presented method for programmed generation of logic relations are discussed.

The problem of safety of computer control system for the railway movement was also discussed.