

SYNTEZA AUTOMATÓW SKOŃCZONYCH Z WYKORZYSTANIEM METOD KODOWANIA WIELOKROTNEGO

Arkadiusz Bukowiec

Instytut Informatyki i Elektroniki, Uniwersytet Zielonogórski
65-246 Zielona Góra, ul. Podgórna 50
e-mail: a.bukowiec@iie.uz.zgora.pl

STRESZCZENIE

W referacie zostaną przedstawione i porównane metody sprzętowej optymalizacji układu automatu skończonego z wyjściami typu Mealy'ego realizowanego w strukturach programowalnych. Zaprezentowane metody bazują na realizacji dwupoziomowego automatu, w którym zostanie zastosowane kodowanie wielokrotne. Zaprezentowane metody kodowania wielokrotnego polegają na przypisaniu różnych kodów stanom przejść lub zbiorom mikrooperacji dla poszczególnych podzbiorów tych parametrów. Podział na podzbiory dokonywany jest na bazie aktualnego stanu lub aktualnie wykonywanej mikrooperacji.

1. WPROWADZENIE

Skończone automaty stanów (ang. *Finite State Machine*, FSM), a w szczególność automat z wyjściami typu Mealy'ego [1, 6], są wciąż jedną z najpopularniejszych metod realizacji jednostek sterujących. Programowalne układy PLD (ang. *Programmable Logic Device*, PLD) takie jak CPLD lub FPGA są bardzo często stosowane do realizacji układu logicznego takiego automatu [5]. Wysoki koszt realizacji takiego układu powoduje, że optymalizacja wykorzystywanych zasobów sprzętowych stanowi wciąż ważny problem. Jednym z rozwiązań tego problemu jest synteza układu z zastosowaniem struktur wielopoziomowych [2]. Metody takie wymagają utworzenia dodatkowych zmiennych wewnętrznych oraz bardzo często wykorzystania dodatkowych zasobów sprzętowych. W referacie został przedstawiony sposób wielokrotnego kodowania mikroinstrukcji lub stanów wewnętrznych umożliwiający zmniejszenie liczby funkcji boolowskich realizowanych przez część kombinacyjną układu, co prowadzi do zmniejszenia wymaganych zasobów sprzętowych potrzebnych do implementacji danego układu w strukturach programowalnych.

2. DEFINICJE I IDEA KODOWANIA WIELOKROTNEGO

Automat skończony z wyjściami typu Mealy'ego może zostać opisany za pomocą tablicy przejść-wyjść automatu [1, 6] z kolumnami: $a_m, K(a_m), a_s, K(a_s), X_h, Y_h, \Phi_h, h$. $a_m \in A$ jest

aktualnym stan automatu; $K(a_m)$ jest binarnym kodem stanu a_m , zakodowanym na $R = \lceil \log_2 M \rceil$ bitach, reprezentowanych przez zmienną $Q_r \in Q = \{Q_1, \dots, Q_R\}$; $a_s \in A$ stanowi następny stan automatu a $K(a_s)$ jest jego kodem binarnym; X_h jest koniunkcją pewnych elementów ze zbioru warunków logicznych $X = \{x_1, \dots, x_L\}$, powodującą przejście $\langle a_m, a_s \rangle$; $Y_h \subseteq Y$ stanowi zbiór mikrooperacji (mikroinstrukcja) formowany podczas przejścia $\langle a_m, a_s \rangle$, gdzie Y jest zbiorem wszystkich mikrooperacji; $\Phi_h \subseteq \Phi$ stanowi zbiór funkcji wzbudzeń, które są równe 1 i powodują przełączenie pamięci automatu z $K(a_m)$ na $K(a_s)$; $h = 1, \dots, H$ jest liczbą przejść automatu.

Automat Mealy'ego opisany w taki sposób może zostać zrealizowany jako jednopoziomowy układ cyfrowy (automat-P) [2], w którym układ kombinacyjny realizuje system funkcji wyjść automatu:

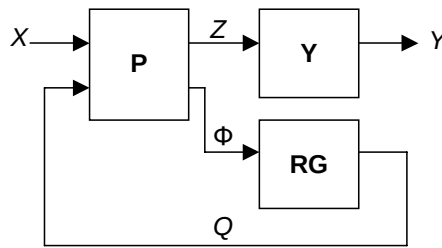
$$Y = Y(Q, X), \quad (1)$$

i system funkcji wzbudzeń:

$$\Phi = \Phi(Q, X). \quad (2)$$

natomiast pamięć automatu stanowi rejestr, zbudowany z przerzutników typu D.

Niech w tablicy przejść-wyjść automatu występuje T różnych mikroinstrukcji $Y_t \subseteq Y$. Zakodujemy każdy taki zbiór $Y_t \subseteq Y$ binarnym kodem $K(Y_t)$ na $R_1 = \lceil \log_2 T \rceil$ bitach. Do reprezentacji tego kodu wykorzystamy zmienną $z_r \in Z = \{z_1, \dots, z_{R_1}\}$. W takim przypadku automat Mealy'ego może zostać przedstawiony jako dwupoziomowa struktura – automat-PY (rys. 1) [2].



Rys. 1. Schemat automatu-PY

W przypadku takim układ kombinacyjny **P** realizuje system funkcji wzbudzeń (2) oraz system kodujący mikroinstrukcje:

$$Z = Z(Q, X), \quad (3)$$

Natomiast układ **Y** pełni funkcję dekodowania mikroinstrukcji i realizuje funkcję:

$$Y = Y(Z). \quad (4)$$

Ideą zastosowania kodowania wielokrotnego, przedstawionego w tym referacie, jest zmniejszenie liczby wyjść układu kombinacyjnego **P**. Metody kodowania wielokrotnego polegają na przypisaniu różnym parametrom wewnętrznym, tj. stanom przejść lub mikroinstrukcją, różnych kodów dla poszczególnych podzbiorów tych parametrów. Podział na podzbiory dokonywany jest na bazie aktualnego stanu lub aktualnie wykonywanej mikroinstrukcji.

3. METODY KODOWANIA WIELOKROTNEGO

3.1. Metoda z wielokrotnymi kodami mikroinstrukcji

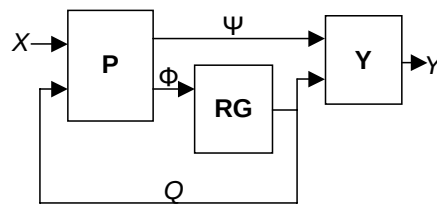
Metoda ta bazuje na przypisaniu wielokrotnych kodów mikroinstrukcją [4]. Podział zbioru mikroinstrukcji Y_h na podzbiory dokonywany jest poprzez aktualny stan a_m . Niech $Y_h(a_m)$ będzie zbiorem mikroinstrukcji realizowanych podczas przejść ze stanu $a_m \in A$ i $|Y_h(a_m)| = T_m$. Przez T_0 oznaczmy maksymalną liczbę mikroinstrukcji realizowanych w jednym stanie a_m :

$$T_0 = \max(T_1, \dots, T_M). \quad (5)$$

Zakodujemy każdy zbiór mikrooperacji $Y_t \subseteq Y_h(a_m)$ binarnym kodem $K_m(Y_t)$. Kod ten będzie posiadał $R_2 = \lceil \log_2 T_0 \rceil$ bitów i do jego reprezentacji wykorzystane zostaną zmienne ze zbioru $\Psi = \{\psi_1, \dots, \psi_{R_2}\}$. W sytuacji takiej kod mikroinstrukcji $K(Y_t)$ będzie reprezentowany przez wielokrotny kod mikroinstrukcji $K_m(Y_t)$ oraz kod aktualnego stanu $K(a_m)$:

$$K(Y_t) = K_m(Y_t) * K(a_m). \quad (6)$$

Automat Mealy'ego z takim kodowaniem może zostać zrealizowany w dwupoziomowej strukturze PY_0 (rys. 2).



Rys. 2. Schemat automatu- PY_0

W przypadku takim układ **P** realizuje system funkcji wzbudzeń (2) i system :

$$\Psi = \Psi(Q, X). \quad (7)$$

Układ **Y** implementuje zaś funkcje konwertera kodu:

$$Y = Y(\Psi, Q). \quad (8)$$

Gdy zachodzi warunek:

$$R_2 < R_1 \quad (9)$$

struktura ta pozwala na zmniejszenie liczby wyjść układu **P** w porównaniu z automatem-PY do

$$t(PY_0) = R_2 + R. \quad (10)$$

3.2. Metody z wielokrotnymi kodami stanów wewnętrznych

Metody ta bazują na przypisaniu wielokrotnych kodów stanom wewnętrznym automatu. Pierwsza metoda dokonuje podziału zbioru stanów wewnętrznych A automatu na podzbiory ze względu na aktualnie wykonywaną mikroinstrukcję Y_t [3]. Druga metoda jako parametr dzielący zbioru stanów wewnętrznych A obiera aktualny stan a_m .

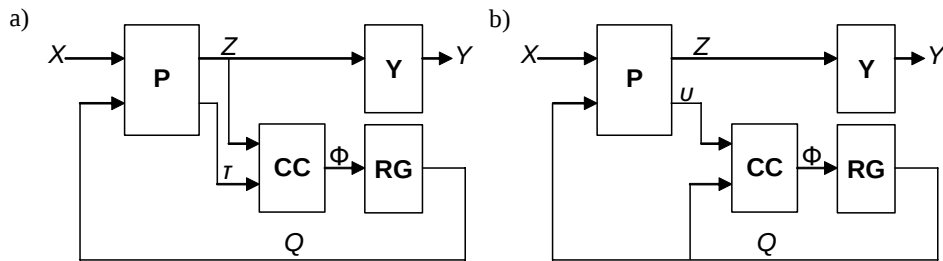
Niech $A(Y_t) \subseteq A$ będzie zbiorem możliwych stanów wewnętrznych dla mikroinstrukcji Y_t i $|A(Y_t)| = B_t$. Przez B_0 oznaczmy maksymalną liczbę różnych stanów przejść dla mikroinstrukcji Y_t :

$$B_0 = \max(B_1, \dots, B_T). \quad (11)$$

Zakodujemy każdy stan $a_s \in A(Y_t)$ R_3 -bitowym kodem $K_t(a_s)$ gdzie $R_3 = \lceil \log_2 B_0 \rceil$. Do reprezentacji tego kodu wykorzystane zostaną zmienne ze zbioru $\tau = \{\tau_1, \dots, \tau_{R_3}\}$. W sytuacji takiej każdy stan $a_s \in A(Y_t)$ będzie reprezentowany przez wielokrotny kod stanu wewnętrznego $K_t(a_s)$ i kod mikroinstrukcji $K(Y_t)$:

$$K(a_s) = K_t(a_s) * K(Y_t). \quad (12)$$

Struktura takiego dwupoziomowego automatu Mealy'ego może zostać zrealizowana jako automat-PYY (rys. 2a).



Rys. 3. Schematy a) automatu-PYY b) automatu-PAY

Przy takiej strukturze układ **P** realizuje system funkcji (3) oraz:

$$\tau = \tau(Q, X). \quad (13)$$

Układ **Y** funkcjonuje tak samo jak w strukturze PY i realizuje system (4). Wprowadzony zostaje dodatkowy układ **CC** konwersji kodu realizujący funkcję:

$$\Phi = \Phi(Z, \tau). \quad (14)$$

Gdy zachodzi warunek:

$$R_3 < R \quad (15)$$

struktura PYY pozwala na zmniejszenie liczby wyjść układu **P** w porównaniu z automatem-PY do

$$t(PYY) = R_1 + R_3. \quad (16)$$

Przy drugim sposobie wielokrotnego kodowania stanów wewnętrznych niech $A(a_m) \subseteq A$ będzie zbiorem stanów przejść ze stanu $a_m \in A$ i $|A(a_m)| = B'_m$. Przez B'_0 oznaczmy maksymalną liczbę różnych stanów przejść z jednego stanu a_m :

$$B'_0 = \max(B'_1, \dots, B'_M) \quad (17)$$

i niech $R_4 = \lceil \log_2 B'_0 \rceil$. Zakodujmy każdy stan $a_s \in A(a_m)$ R_4 -bitowym kodem $K_m(a_s)$. Niech kod ten będzie reprezentowany przez zmienne ze zbioru $v = \{v_1, \dots, v_{R_4}\}$. W przypadku takim kod $K(a_s)$ każdego stanu a_s jest reprezentowany:

$$K(a_s) = K_m(a_s) * K(a_m). \quad (18)$$

Automat Mealy'ego z takim kodowaniem stanów wewnętrznych może zostać zrealizowany jako automat-PAY (rys. 3b).

W sytuacji tej układ **P** realizuje system funkcji (3) oraz:

$$v = v(Q, X), \quad (19)$$

układ **Y** realizuje system (4), Natomiast układ CC realizujący funkcję:

$$\Phi = \Phi(Q, v). \quad (20)$$

Gdy zachodzi warunek:

$$R_4 < R \quad (21)$$

struktura PAY pozwala na zmniejszenie liczby wyjść układu **P** w porównaniu z automatem-PY do

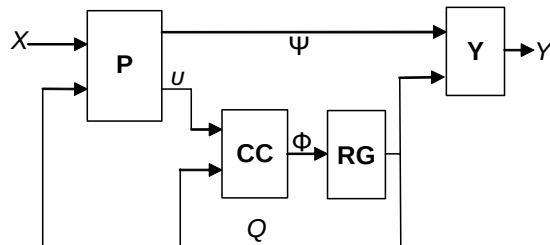
$$t(PAY) = R_1 + R_4. \quad (22)$$

Aby struktura PAY była efektywniejsza od struktury PYY musi zachodzić warunek:

$$t(PAY) < t(PYY). \quad (23)$$

3.3. Metoda z wielokrotnymi kodami mikroinstrukcji i stanów wewnętrznych

Należy tu jednak wspomnieć, że struktura PAY posiada jedną przewagę nad strukturą PYY. W strukturze tej istnieje możliwość dalszej modyfikacji, poprzez dodanie wielokrotnego kodowania mikroinstrukcji. Zabieg taki doprowadzi do powstania automat-PAY₀ (rys. 4).



Rys. 4. Schemat automatu-PAY₀

Metody kodowania i reprezentacji zmiennych są analogiczne z tymi przedstawionymi w rozdziałach 3.1. i 3.2. W strukturze tej układ **P** realizuje funkcje (7) i (19), układ **Y** system (8), natomiast układ **CC** funkcję (20). W strukturze tej liczba wyjść układu **P** wynosi:

$$t(\text{PAY}_0) = R_2 + R_4. \quad (24)$$

4. ZAKOŃCZENIE

Metody z wielokrotnymi kodami zaprezentowane w referacie, w pewnych warunkach, zapewnią zmniejszenie liczby wyjść układu **P**, co prowadzi do zmniejszenia wymaganych zasobów sprzętowych potrzebnych do jego implementacji.

Przeprowadzone badania analityczne, w oparciu o wyniki pracy [2], wykazały, że proponowane metody są lepsze od standardowej. Zysk osiąga 5%-20% w zależności od parametrów układu i doboru metody.

LITERATURA

- [1] S. Baranov: *Logic Synthesis for Control Automata*, Kluwer Academic Publishers, Dordrecht, 1994
- [2] A. Barkalov: *Structures of the multilevel circuits of microprogram automata on PLA*, Cybernetics and System Analysis, No. 4, 1994, s. 22-29
- [3] A. Bukowiec: *Synteza automatów Mealy'ego z wielokrotnymi kodami stanów wewnętrznych*, Materiały konferencji naukowej Informatyka – sztuka czy rzemiosło – KNWS '04, Zielona Góra, 2004, s. 29-34
- [4] A. Bukowiec: *Synteza automatów skończonych Mealy'ego z wielokrotnymi kodami stanów wewnętrznych lub zbiorów mikrooperacji*, Archiwum Konferencji PTETiS – VI Międzynarodowe Warsztaty Doktoranckie – OWD '04, Gliwice, 2004, Vol. 19, z. 5, s. 367-372
- [5] T. Łuba, et. al.: *Synteza układów cyfrowych*, Wydawnictwa Komunikacji i Łączności, Warszawa, 2003
- [6] D. Gajski: *Principles of Digital Design*, Prentice Hall, New Jersey 1997