

УДК 004.274

**С.О. Цололо (канд. техн. наук)**

ДВНЗ «Донецький національний технічний університет», м. Донецьк  
кафедра комп'ютерної інженерії  
E-mail: s.solos@gmail.com

## ОПТИМІЗАЦІЯ СХЕМИ АВТОМАТА МУРА В БАЗИСІ FPGA

*Запропонований метод зменшення числа елементів табличного типу LUT у схемі мікропрограмного автомата Мура при реалізації у базисі FPGA. Метод заснований на використанні надлишкових входів вбудованих блоків пам'яті ЕМВ для формування кодів класів псевдоеквівалентних станів. Для оптимізації схеми використовуються три джерела кодів класів. В статті наведено необхідні умови застосування запропонованого методу та практичний приклад його використання.*

**Ключові слова:** FPGA, LUT, ЕМВ, автомат Мура, синтез, витрати апаратури.

### Вступ

У сучасний час програмувальні логічні інтегральні схеми (ПЛІС) типу FPGA (field-programmable gate arrays) широко застосовуються при реалізації складних цифрових систем [1, 2]. Одним з важливих блоків цифрових систем є пристрій керування (ПК) [3], який часто реалізується у вигляді мікропрограмного автомата (МПА) Мура [4]. Характерними особливостями МПА Мура є наявність класів псевдоеквівалентних станів (ПЕС) та регулярний характер системи вихідних функцій [5]. Ці особливості використовуються для зменшення апаратних витрат в схемі МПА Мура [6].

Сучасні ПЛІС FPGA включають елементи табличного типу LUT (Look-Up-Tables) і вбудовані блоки пам'яті ЕМВ (Embedded Memory Blocks) [7, 8]. Елементи LUT мають обмежене число входів (до 6), що визиває необхідність декомпозиції функцій, які потрібно реалізувати [9]. Це в свою чергу призводить до зменшення швидкодії та збільшення споживаної потужності [1]. В роботах [10, 11] наведені методи оптимізації схем МПА Мура в базисі CPLD (Complex Programmable Logic Devices). Ці методи використовують наявність великої кількості входів (до 30) в макроосередках кристалів CPLD, що дозволяє використовувати до трьох джерел кодів станів. Ці методи не можуть бути використані безпосередньо при реалізації схем на ПЛІС FPGA через невелику кількість входів LUT елементів. В роботі пропонується модифікація одного з методів оптимізації, орієнтованих на CPLD, для використання у базисі FPGA.

Метою досліджень, представлених в роботі, є можливість зменшення кількості елементів LUT завдяки використанню трьох джерел кодів класів ПЕС.

Задачею, що вирішується в роботі, є розробка методу синтезу МПА Мура, який дозволяє зменшити число LUT елементів в схемі формування функцій збудження тригерів пам'яті. При цьому алгоритм керування цифровою системою представляється у вигляді граф-схеми алгоритму (ГСА) [4].

### Особливості реалізації МПА Мура на FPGA

Нехай алгоритм керування цифровою системою представлений ГСА  $\Gamma = \Gamma(V, E)$ , де  $V = \{b_0, b_E\} \cup E_1 \cup E_2$  – множина вершин,  $E = \{<b_q, b_t> \mid b_q, b_t \in V\}$  – множина дуг. Тут  $b_0$  – початкова вершина ГСА,  $b_E$  – кінцева вершина ГСА,  $E_1$  – множина операторних вершин,  $E_2$  – множина умовних вершин. У вершинах  $b_q \in E_1$  записуються набори мікрооперацій  $Y(b_q) \subseteq Y$ , де  $Y = \{y_1, \dots, y_N\}$  – множина мікрооперацій операційного автомата цифрової системи [4]. У вершинах  $b_q \in E_2$  записуються елементи множини логічних умов  $X = \{x_1,$

... $x_L$ }. Початкова та кінцева вершини ГСА відповідають стану  $a_1 \in A = \{a_1, \dots, a_M\}$ , де  $A$  – множини станів автомата Мура, а кожна вершина  $b_q \in E_1$  відповідає одному з елементів множини  $A$  [4]. Логічна схема МПА Мура задається системою рівнянь

$$\Phi = \Phi(T, X), \quad (1)$$

$$Y = Y(T), \quad (2)$$

де  $T = T\{T_1, \dots, T_R\}$  – множина внутрішніх змінних, що кодують стани  $a_m \in A$ ,  $R = \lceil \log_2 M \rceil$ ;  $\Phi = \Phi\{D_1, \dots, D_R\}$  – множина функцій збудження тригерів пам'яті станів. Системи (1)-(2) формуються на основі прямої структурної таблиці (ПСТ) зі стовпцями:  $a_m$  – поточний стан;  $K(a_m)$  – код стану  $a_m \in A$ ;  $a_s$  – стан переходу;  $K(a_s)$  – код стану  $a_s \in A$ ;  $X_h$  – кон'юнкція деяких елементів множини  $X$  (або їх заперечень), що визначає перехід  $\langle a_m, a_s \rangle$ ;  $\Phi_h$  – набір функцій збудження пам'яті МПА, що приймають одиничне значення для перемикавання пам'яті з  $K(a_m)$  в  $K(a_s)$ ;  $h = 1, \dots, H_1(\Gamma)$  – номер рядка таблиці. У стовпці  $a_m$  записується набір мікрооперацій  $Y(a_m) \subseteq Y$ , що було сформовано у стані  $a_m \subseteq A$ . Очевидно, що  $Y(a_m) = Y(b_q)$ , де вершина  $b_q \in E_1$  відзначена станом  $a_m \in A$ .

Як правило, число переходів  $H_1(\Gamma)$  більше числа переходів  $H_2(\Gamma)$  еквівалентного автомата Мілі [4]. Це приводить до збільшення числа PAL у схемі МПА Мура в порівнянні з цим показником еквівалентного автомата Мілі. Параметр  $H_1(\Gamma)$  можна зменшити завдяки наявності псевдоеквівалентних станів (ПЕС) МПА Мура [5]. Стани  $a_m, a_s \in A$  називаються псевдоеквівалентними, якщо виходи відповідних їм вершин з'єднані із входом однієї й тієї ж вершини ГСА  $\Gamma$ . Нехай  $\Pi_A = \{B_1, \dots, B_I\}$  – розбивка множини  $A$  на класи ПЕС ( $I \leq M$ ). Поставимо у відповідність класу  $B_i \in \Pi_A$  двійковий код  $K(B_i)$  розрядності  $R_1 = \lceil \log_2 I \rceil$  та використовуємо змінні  $\tau_r \in \tau$  для такого кодування, де  $|\tau| = R_1$ . У цьому випадку МПА Мура представляється у вигляді структури  $U_1$  (рис. 1). Оскільки МПА Мура  $U_1$  є орієнтованим на реалізацію в базисі FPGA, то до схеми входять блоки, що складаються з LUT елементів (LUTer1, LUTer2) та EMB (EMBer).

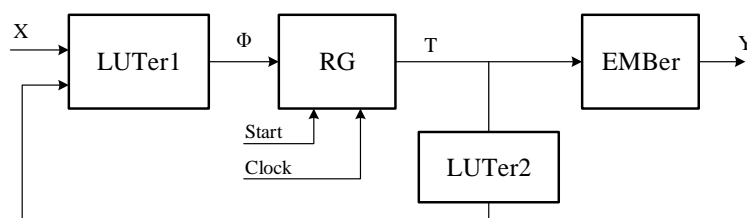


Рисунок 1 – Структурна схема автомату  $U_1$ , що орієнтована на базис ПЛІС FPGA

В автоматі  $U_1$  блок LUTer1 реалізує систему функцій

$$\Phi = \Phi(\tau, X), \quad (3)$$

а блок EMBer реалізує систему (2). Регістр RG представляє собою пам'ять станів, за сигналом Start в RG заноситься нульовий код початкового стану  $a_1 \in A$ , за сигналом Clock відбувається зміна кодів у регістрі. Перетворювач кодів станів LUTer2 реалізує систему функцій

$$\tau = \tau(T), \quad (4)$$

при цьому код  $K(B_i)$  формується на основі коду  $K(a_m)$ , де  $a_m \in B_i$ .

У роботі [10] доведено, що для МПА  $U_1$  число переходів зменшується до  $H_2(\Gamma)$ . Недоліком моделі  $U_1$  є наявність блоку LUTer2, схема якого споживає деякі ресурси кристалу. У цій роботі пропонується метод синтезу МПА Мура, що дозволяє зменшити апаратні витрати в блоці LUTer2. Запропонований метод заснований на таких особливостях мікросхем FPGA [7, 8]: логічні елементи складаються з елементів LUT, програмувального тригеру, мультиплексорів та трьохстабільних буферів, що дозволяє організувати мультиплексування різних LUT елементів; вбудовані блоки пам'яті мають постійну ємність ( $V_0$ ), але змінну кількість осередків ( $V$ ) та виходів ( $t_F$ ).

**Головна ідея методу, що пропонується**

Закодуємо стани  $a_m \in A$  оптимальним чином. Представимо множину  $\Pi_A$  у вигляді об'єднання множин  $\Pi_B$  і  $\Pi_C$ . При цьому розподіл класів виконується в такий спосіб:

$$(T(B_i) = 1) \rightarrow B_i \in \Pi_B, (T(B_i) > 1) \rightarrow B_i \in \Pi_C. \quad (6)$$

Очевидно, що перетворенню підлягають тільки коди станів  $a_m \in B_i$  для блоків  $B_i \in \Pi_C$ . Поставимо у відповідність кожному класу  $B_i \in \Pi_C$  двійковий код  $K(B_i)$  розрядності

$$R_c = \lceil \log_2(I_c + 1) \rceil, \quad (7)$$

де  $I_c = |\Pi_C|$ . Призначення одиниці у формулі (7) пояснимо пізніше.

Нехай  $t_F$  – фіксоване число виходів блоку ЕМВ і нехай  $q$  – число слів у блоці при  $t_F = 1$ . Для блоку ЕМВег параметр  $t_F$  визначається як

$$t_F = \lceil q / 2^R \rceil, \quad (8)$$

а сумарне число виходів у блоках ЕМВ, що утворюють схему ЕМВег, визначається як

$$t_1 = \lceil N / t_F \rceil \cdot t_F. \quad (9)$$

При цьому  $\Delta_t$  виходів не використовуються для представлення мікрооперацій, де

$$\Delta_t = t_1 - N. \quad (10)$$

Ці виходи можна використовувати для представлення розрядів коду  $K(B_i)$ . При цьому блок LUTer2 буде реалізовувати

$$R_K = R_c - \Delta_t \quad (11)$$

розрядів коду. При виконанні умови

$$R_c \leq \Delta_t \quad (12)$$

блок LUTer2 відсутній. Для загального випадку, коли  $\Delta_t \neq 0$  й  $R_K > 1$ , автомат Мура представляється моделлю  $U_2$  (рис. 2).

У цій моделі є три джерела кодів станів. Стани  $a_m \in B_i$ , де  $B_i \in \Pi_B$ , визначаються вмістом регістру RG. Стани класів  $B_i \in \Pi_C$  представляються змінними  $\tau_r \in \tau^1$  (виходи ЕМВег) і  $\tau_r \in \tau^2$  (виходи LUTer2). Очевидно, що  $\tau^1 \cup \tau^2 = \tau$  й  $|\tau| = R_c$ . Для ідентифікації джерела необхідно ввести спеціальний код, що визначається як

$$\tau_r = 0 \quad (r = \overline{1, R_c}). \quad (13)$$

Цим пояснюється наявність одиниці у вираженні (7). Нульовий код (13) використовується для керування мультиплексором MX, який служить для вибору джерела коду класу ПЕС. Для вибору використовується змінна  $y_M$ :

$$y_M = \bigwedge_{r=L}^{R_c} \bar{\tau}_r. \quad (14)$$

У МПА  $U_2$  блок LUTer1 формує систему функцій:

$$\Phi^1 = \Phi^1(T, X). \quad (15)$$

Блок LUTer3 формує функції:

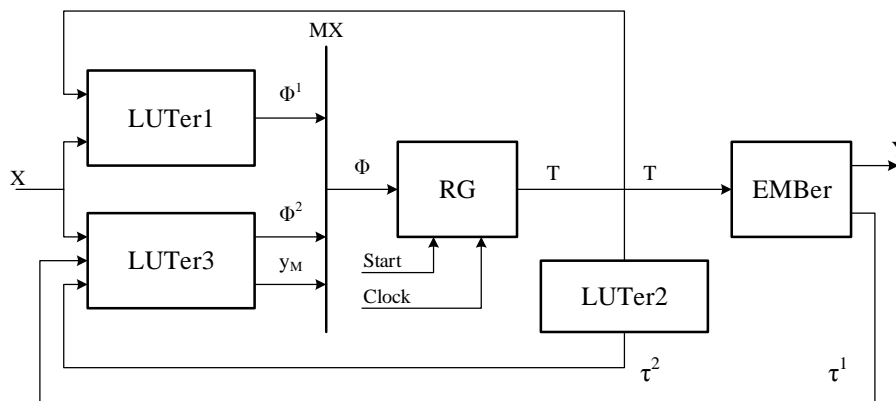
$$\Phi^2 = \Phi^2(\tau, x). \quad (16)$$

Функції збудження тригерів RG формуються в такий спосіб:

$$\Phi = \overline{y_M} \Phi^1 \vee y_M \Phi^2. \quad (17)$$

Для реалізації системи (17) використовується MX. Блок LUTer2 формує функції:

$$\tau^2 = \tau^2(T). \quad (18)$$

Рисунок 2 – Структурна схема автомата Мура  $U_2$ 

Блок EMBer формує вихідні функції (2) і функції

$$\tau^1 = \tau^1(T). \quad (19)$$

Таким чином, в роботі пропонується метод синтезу МПА Мура  $U_2$ , що включає наступні етапи:

1. Формування розбивки  $\Pi_A = \{B_1, \dots, B_I\}$ .
2. Оптимальне кодування станів  $a_m \in A$ .
3. Формування множин  $\Pi_B$  і  $\Pi_C$ .
4. Кодування класів  $B_i \in \Pi_C$ .
5. Формування таблиць блоків LUTer1 і LUTer3.
6. Формування вмісту блоку мікрооперацій (EMBer).
7. Формування таблиці блоку перетворювача кодів (LUTer2).
8. Формування системи функцій, що задають схему МПА.
9. Реалізація схеми в заданому елементному базисі.

Розглянемо приклад застосування запропонованого методу.

#### Приклад застосування запропонованого методу

Використовуємо завдання автомата у вигляді системи узагальнених формул переходу (УФП) [6]. Нехай автомат Мура  $U_1(\Gamma)$  визначається наступною системою УФП:

$$\begin{aligned} B_1 &\rightarrow x_1 a_2 \vee \bar{x}_1 a_3; & B_2 &\rightarrow x_2 a_4 \vee \bar{x}_2 x_3 a_5 \vee \bar{x}_2 \bar{x}_3 a_6; \\ B_3 &\rightarrow x_3 a_6 \vee \bar{x}_3 x_4 a_8 \vee \bar{x}_3 \bar{x}_4 a_1; & B_4 &\rightarrow x_5 a_8 \vee \bar{x}_5 x_6 a_{11} \vee \bar{x}_5 \bar{x}_6 a_{13}; \\ B_5 &\rightarrow a_{10}; & B_6 &\rightarrow x_1 a_{12} \vee \bar{x}_1 a_1; & B_7 &\rightarrow x_4 a_1 \vee \bar{x}_4 x_5 a_7 \vee \bar{x}_4 \bar{x}_5 x_6 a_{11} \vee \bar{x}_4 \bar{x}_5 \bar{x}_6 a_{13}. \end{aligned} \quad (20)$$

Нехай при цьому отримана розбивка  $\Pi_A = \{B_1, \dots, B_7\}$ , де  $B_1 = \{a_1\}$ ,  $B_2 = \{a_2, a_3\}$ ,  $B_3 = \{a_4\}$ ,  $B_4 = \{a_5, a_6, a_7\}$ ,  $B_5 = \{a_8, a_9\}$ ,  $B_6 = \{a_{10}\}$ ,  $B_7 = \{a_{11}, a_{12}, a_{13}\}$ . Нехай система мікрооперацій автомата  $U_1(\Gamma)$  представляється наступною системою рівнянь:

$$\begin{aligned} y_1 &= a_2 \vee a_3 \vee a_{12}; & y_2 &= a_4 \vee a_8 \vee a_9 \vee a_{10}; \\ y_3 &= a_3 \vee a_7 \vee a_8 \vee a_{11}; & y_4 &= a_5 \vee a_{11}; \\ y_5 &= a_6 \vee a_7 \vee a_8; & y_6 &= a_4 \vee a_6 \vee a_{10}; & y_7 &= a_3 \vee a_{11} \vee a_{12} \vee a_{13}. \end{aligned} \quad (21)$$

Система (21) будується тривіальним образом: якщо мікрооперація  $y_n \in A$  формується в стані  $a_m \in A$ , то в рівняння для  $y_n$  включається терм  $a_m$ .

Отже, для автомата  $U_1(\Gamma)$  маємо  $M=13$ ,  $R=4$ ,  $T = \{T_1, \dots, T_4\}$ ,  $\Phi = \{D_1, \dots, D_4\}$ ,  $I=7$ ,  $N=8$ . Сформуємо систему функцій, що виражає залежність класів ПЕС від станів:

$$\begin{aligned}
 B_1 &= a_1 ; B_2 = a_2 \vee a_3 ; \\
 B_3 &= a_4 ; B_4 = a_5 \vee a_6 \vee a_7 ; \\
 B_5 &= a_8 \vee a_9 ; B_6 = a_{10} ; B_7 = a_{11} \vee a_{12} \vee a_{13} .
 \end{aligned}
 \tag{22}$$

Один з можливих результатів оптимального кодування станів для автомата  $U_1(\Gamma)$  показаний картою Карно на рис. 3.

		$T_3T_4$			
		00	01	11	10
$T_1T_2$	00	$a_1$	$a_2$	*	$a_4$
	01	$a_5$	*	$a_6$	$a_{10}$
	11	$a_{11}$	$a_3$	$a_7$	$a_8$
	10	$a_{13}$	$a_{12}$	*	$a_9$

Рисунок 3 – Коды станів автомата Мура  $U_1(\Gamma)$

Аналіз цієї карти показує, що класи  $B_1, B_3, B_5, B_6 \in \Pi_B$  і класи  $B_2, B_4, B_7 \in \Pi_C$ . Таким чином,  $I_C = 3$ ,  $R_C = 2$ ,  $\tau = \{\tau_1, \tau_2\}$ . Закодуємо класи  $B_i \in \Pi_C$  в такий спосіб:  $K(B_2) = 11$ ,  $K(B_4) = 01$ ,  $K(B_7) = 10$ . Таким чином, чим більше станів входить до класу  $B_i \in \Pi_C$ , тим менше одиниць містить код цього класу. Відзначимо, що код 00 відповідає ситуації  $B_i \in \Pi_B$ . Цим пояснюється наявність одиниці у формулі (7). З карти Карно маємо  $K(B_1) = 0000$ ,  $K(B_3) = 001*$ ,  $K(B_5) = 1*10$ ,  $K(B_6) = 0110$ .

Тепер переходимо до моделі  $U_2(\Gamma_1)$ . Побудуємо таблиці блоків LUTer1 і LUTer3. Для побудови таблиць використовуються відповідні формули системи (20). Таблиця блоків LUTer1 має наступні стовпці:  $B_i, K(B_i), a_s, K(a_s), X_h, \Phi_h^1, h$ . Наприклад, для класів  $B_1, B_3 \in \Pi_B$  ця таблиця має 5 рядків (табл. 1).

Таблиця 1

Фрагмент таблиці блоку LUTer1 автомату  $U_2(\Gamma_1)$

$B_i$	$K(B_i)$	$a_s$	$K(a_s)$	$X_h$	$\Phi_h^1$	$h$
$B_1$	0000	$a_2$	0001	$x_1$	$D_4$	1
		$a_3$	1101	$\bar{x}_1$	$D_1D_2D_4$	2
$B_3$	001*	$a_6$	0111	$x_3$	$D_2D_3D_4$	3
		$a_8$	1110	$\overline{x_3x_4}$	$D_1D_2D_3$	4
		$a_1$	0000	$\overline{x_3x_4}$	—	5

Таблиця блоку LUTer3 має аналогічні стовпці, але замість стовпця  $\Phi_h^1$  вводиться стовпець  $\Phi_h^2$ . Наприклад, для класу  $B_7 \in \Pi_C$  ця таблиця має 4 стовпця (табл. 2).

Таблиця 2

Фрагмент таблиці блоку LUTer3 автомату  $U_2(\Gamma_1)$ 

$B_i$	$K(B_i)$	$a_s$	$K(a_s)$	$X_h$	$\Phi_h^2$	$h$
$B_7$	10	$a_1$	0000	$x_4$	–	1
		$a_7$	1111	$\overline{x_4 x_5}$	$D_1 D_2 D_3 D_4$	2
		$a_{11}$	1100	$\overline{x_4 x_5 x_6}$	$D_1 D_2$	3
		$a_{13}$	1000	$\overline{x_4 x_5 x_6}$	$D_1$	4

Формування вмісту блоку мікрооперацій зводиться до формування таблиці зі стовпцями  $a_m, K(a_m), Y(a_m), \tau_m^1, m$ . Тут  $Y(a_m) \subseteq Y$  – набір мікрооперацій, що формується в стані  $a_m \in A$ . Ця інформація міститься у операторних вершинах ГСА  $\Gamma$ , а в нашому прикладі задана системою (21). Стовпець  $\tau_m^1$  містить змінні  $\tau_r \in \tau^1$ , рівні одиниці в частині коду  $K(B_i)$ , що формується блоком ЕМВег. Цей стовпець може бути відсутнім, якщо  $\Delta_i = 0$ .

Нехай для реалізації блоку ЕМВег використовуються блоки ЕМВ, у яких число  $t_F$  вибирається з множини  $T_0 = \{1, 2, 4\}$ , і при  $t_F = 1$  ( $q = 64$ ). Для автомату  $U_2(\Gamma_1)$   $M = 13$ ,  $R = 4$  і, згідно (8), маємо  $t_F = 4$ . Оскільки  $t_F \in T_0$ , те вибираємо його в якості фіксованого числа виходів блоків ЕМВ. Відзначимо, що якщо  $t_F \notin T_0$ , то в якості  $t_F$  необхідно вибрати найближчий менший елемент безлічі  $T_0$ . Для автомату  $U_2(\Gamma_1)$   $N = 7$ , тому  $t_1 = 8$  і  $\Delta_i = 1$ . Отже, один розряд коду  $K(B_i)$  може бути реалізований на ЕМВег.

Нехай  $\tau^1 = \{\tau_1\}$ , тоді ця змінна включається в рядки 2, 3, 11, 12, 13 таблиці вмісту блоку ЕМВег (табл. 3).

Таблиця 3

Вміст блоку ЕМВег автомату  $U_2(\Gamma_1)$ 

$a_m$	$K(a_m)$	$Y(a_m)$	$\tau_m^1$	$m$	$a_m$	$K(a_m)$	$Y(a_m)$	$\tau_m^1$	$m$
$a_1$	0000	–	–	1	$a_8$	1110	$y_2 y_3 y_5$	–	8
$a_2$	0001	$y_1$	$\tau_1$	2	$a_9$	1010	$y_2$	–	9
$a_3$	1101	$y_1 y_3 y_7$	$\tau_1$	3	$a_{10}$	0110	$y_2 y_6$	–	10
$a_4$	0010	$y_2 y_6$	–	4	$a_{11}$	1100	$y_3 y_4 y_7$	$\tau_1$	11
$a_5$	0100	$y_4$	–	5	$a_{12}$	1001	$y_1 y_7$	$\tau_1$	12
$a_6$	0111	$y_5 y_6$	–	6	$a_{13}$	1000	$y_7$	$\tau_1$	13
$a_7$	1111	$y_3 y_5$	–	7	–	****	***	*	*

Блок LUTer2 реалізується на елементах LUT. При цьому доцільно задати кожну функцію  $\tau_r \in \tau^2$  у вигляді карти Карно. Для нашого прикладу функція  $\tau_2 \in \tau^2$  задається картою Карно, що наведена на рис. 4.

Система (15) будується на основі таблиці блоку LUTer1. Наприклад, наступне рівняння (з урахуванням мінімізації) можна знайти з табл. 1:  $D_4 = \overline{T_1 T_2 T_3 T_4} \vee \overline{T_1 T_2} T_3 x_3$ . Система (16) будується на основі таблиці блоку LUTer3. Наприклад наступне рівняння (з урахуванням мінімізації) можна знайти з табл. 2:  $D_1 = \tau_1 \overline{\tau_2 x_4}$ . Система  $\tau^2(T)$  формується по відповідних

картах Карно. Наприклад, наступне рівняння може бути знайдене з рис. 4:  $\tau_2 = \overline{T_1}T_4 \vee T_2T_4 \vee \overline{T_1}T_2\overline{T_3}$ . Для реалізації схеми рівняння мультиплексором використовується рівняння (14). У нашому прикладі маємо  $y_M = \overline{\tau_1}\overline{\tau_2}$ .

$T_3T_4$	00	01	11	10
$T_1T_2$ 00	0	1	*	0
01	1	*	1	0
11	0	1	1	0
10	0	0	*	0

Рисунок 4 – Карта Карно для функції  $\tau_2$

Синтез схеми МПА зводиться до використання стандартних пакетів САПР [7, 8]. Це питання досить докладно розглянуте в літературі [1, 2]. В статті цей етап реалізації запропонованого методу не розглядається.

#### Висновки

У роботі пропонується метод зменшення числа LUT елементів у схемі автомата Мура, що реалізується в складі кристала FPGA. Запропонований метод засновано на використанні трьох джерел коду класів псевдоеквівалентних станів. Вибір джерела коду виконується за допомогою мультиплексора, на виходах якого формуються остаточні значення функцій збудження пам'яті автомата.

Наявність мультиплексору вносить додаткову затримку під час такту пристрою. Однак схеми, що реалізують функції збудження, спрощуються в порівнянні з базовою моделлю  $U_1$ . Це спрощення може призвести до зменшення числа рівнянь LUT елементів у відповідних схемах. При цьому затримка, що вноситься мультиплексором, може бути компенсована.

Дослідження, проведені для стандартних прикладів з бібліотеки [12] показали, що при виконанні умови (10) МПА  $U_2(\Gamma)$  завжди має менше LUT елементів, ніж еквівалентний МПА  $U_1(\Gamma)$ . Максимальний вигреш може досягати 32%. При цьому для 94% стандартних прикладів блок EMVer реалізується у вигляді одного блока EMV. Крім цього, у 86% розглянутих МПА з бібліотеки [12] швидкодія моделей  $U_1(\Gamma)$  та  $U_2(\Gamma)$  співпадає. В 14% прикладів, що залишилися, автомат  $U_2(\Gamma)$  має трохи меншу швидкодію (знижується на значення близько 10%).

Наукова новизна запропонованого методу полягає в використанні особливостей МПА Мура та вбудованих блоків пам'яті мікросхем FPGA для зменшення кількості LUT елементів в схемі МПА.

Практичне значення методу полягає в зменшенні площі кристалу FPGA, що займає схема автомата Мура, в порівнянні з відомими з літератури аналогами.

#### Список використаної літератури

1. Grout I. Digital Systems Design with FPGAs and CPLDs. – Amsterdam: Elsevier, 2008. – 328 pp.
2. Skliarova I., Skliarov V., Sudnitson A. Design of FPGA-based Hierarchical Finite State Machines. – Tallinn: TUT Press, 2012. – 286 pp.
3. De Micheli G. Synthesis and Optimization of Digital Circuits. – New York: McGraw Hill, 1994. – 636 pp.

4. Baranov S. *Logic Synthesis for Control Automata*. – New York: Kluwer Academic Publishers, 1994. – 312 pp.
5. Баркалов А.А. Принципы оптимизации логической схемы микропрограммного автомата Мура. // Кибернетика и системный анализ. – 1998, № 1. – с. 65-72
6. Barkalov A., Titarenko L. *Logic Synthesis for FSM-based Control Units*. – Berlin: Springer, 2009. – 234 pp.
7. Altera Corporation. *FPGA, CPLD and Structured ASIC* [Электронный ресурс]. – Режим доступа: <http://www.altera.com>.
8. Xilinx Inc. *FPGA, CPLD Solutions* [Электронный ресурс]. – Режим доступа: <http://www.xilinx.com>.
9. Scholl C. *Functional Decomposition with Application to FPGA Synthesis*. – Norwell: Kluwer Academic Publishers, 2001. – 319 pp.
10. А.А. Баркалов, Л.А. Титаренко, С.А. Цололо Оптимизация схемы автомата Мура, реализуемой в базисе ПЛИС / Кибернетика и системный анализ. – Киев, 2009. – №5. – С. 180-186.
11. Баркалов А.А., Цололо С.А. Оптимизация автомата Мура, реализуемого в базисе CPLD / // Управляющие системы и машины. – Киев, 2008. – №4 – С. 43-48.
12. Yang S. *Logic Synthesis and Optimization Benchmarks user guide. Technical Report*. – North Carolina: MCNC, 1991. – 44 pp.

#### References

1. Grout, I. (2008), *Systems Design with FPGAs and CPLDs*, Elsevier, Amsterdam, Netherlands.
2. Skliarova, I., Skliarov, V. and Sudnitson, A. (2012), *Design of FPGA-based Hierarchical Finite State Machines*, TUT Press, Tallinn, Estonia.
3. De Micheli, G. (1994), *Synthesis and Optimization of Digital Circuits*, McGraw Hill, New York, USA.
4. Baranov, S. (1994), *Logic Synthesis for Control Automata*, Kluwer Academic Publishers, New York, USA.
5. Barkalov, A.A. (1998), “Principles of optimization logic of Moore FSM”, *Cybernetics and Systems Analysis*, no. 1, pp. 65–72.
6. Barkalov, A. and Titarenko, L. (2009), *Logic Synthesis for FSM-based Control Units*, Springer, Berlin, Germany.
7. Altera Corporation (2014), “FPGA, CPLD and Structured ASIC”, available at: <http://www.altera.com> (Accessed 10 April 2014).
8. Xilinx Inc. (2014), “FPGA, CPLD Solutions”, available at: <http://www.xilinx.com> (Accessed 10 April 2014).
9. Scholl, C. (2001), *Functional Decomposition with Application to FPGA Synthesis*, Kluwer Academic Publishers, Norwell, USA.
10. Barkalov, A.A., Titarenko, L.A. and Tsololo, S.A. (2009), “Optimization of Moore FSM implemented in FPGA basis”, *Cybernetics and Systems Analysis*, no. 5, pp. 180–186.
11. Barkalov, A.A. and Tsololo, S.A., (2008), “Optimization of Moore FSM implemented in CPLD basis”, *Upravljajushhie sistemy i mashiny*, no. 4. pp. 43-48.
12. Yang, S. (1991), *Logic Synthesis and Optimization Benchmarks user guide. Technical Report*, MCNC, North Carolina, USA.

Надійшла до редакції:  
13.05.2014 р.

Рецензент:  
докт. техн. наук, проф. Скобцов Ю.О.



**С.А. Цололо**

**ГВУЗ «Донецкий национальный технический университет»**

**Оптимизация схемы автомата Мура в базе FPGA.** Предлагается метод уменьшения числа элементов табличного типа LUT в схеме микропрограммного автомата Мура при реализации в базе FPGA. Метод основан на использовании избыточных входов встроенных блоков памяти EMB для формирования кодов классов псевдоэквивалентных состояний. Для оптимизации схемы используются три источника кодов классов. В статье приведены необходимые условия применения метода и пример его использования.

**Ключевые слова:** FPGA, LUT, EMB, МПА Мура, синтез, аппаратные затраты.

**S.A. Tsololo**

**Donetsk National Technical University**

**Optimizing circuit of Moore FSM with FPGA.** FPGA (field-programmable gate arrays) are widely used in the implementation of complex digital systems. One of the important elements of digital systems is the control unit (CU), which is often implemented as Moore FSM. Characteristic features of Moore FSM are existence of classes pseudoequivalent states (PES) and regular nature output variables. These features are used to reduce hardware costs in the circuit of Moore FSM.

Modern FPGA elements include LUT (look-up-tables) elements and EMB (embedded memory blocks) units. LUT elements have a limited number of inputs (about six), which necessitates decomposition realizable functions. This in turn leads to a decrease in performance and increase in power consumption. There are many methods of optimization schemes in the basis of Moore FSM CPLD (complex programmable logic devices). These methods use the presence of a large number of inputs (up to 30) in macrocells crystals CPLD, which allows up to three sources of status codes. These methods can not be directly used to implement FPGA PLD circuits because of the small number of inputs in LUT elements. In this paper author propose a modification of one of optimization techniques targeting CPLD.

The purpose of the studies presented in this work is the ability to reduce the number of LUT elements of the three classes of source codes PES. Problem solved in this work is to develop a method for the synthesis of Moore FSM that reduce the number of LUT elements in the circuit triggers the formation of excitation functions of memory. In this case the control algorithm of the digital system is represented as a graph-scheme of algorithm.

The conditions are given for applying the proposed method. An example is shown for its application.

**Keywords:** FPGA, LUT, EMB, Moore FSM, synthesis, hardware optimization.



**Цололо Сергій Олексійович**, Україна, закінчив Донецький національний технічний університет, канд. техн. наук, доцент кафедри комп'ютерної інженерії ДВНЗ «Донецький національний технічний університет» (вул. Артема, 58, м. Донецьк, 83001, Україна). Основний напрям наукової діяльності – методи синтезу та оптимізації пристроїв керування у складі систем на кристалі при реалізації схеми в сучасних базисах CPLD та FPGA.