

## РАСШИРЕНИЕ ФОРМАТА МИКРОКОМАНД ПРИ РЕАЛИЗАЦИИ УСТРОЙСТВА УПРАВЛЕНИЯ НА ГИБРИДНЫХ FPGA

В статье предлагается метод синтеза композиционного микропрограммного устройства управления в базе гибридных FPGA. Предлагается использование кодов классов псевдоэквивалентных операторных линейных цепей для уменьшения аппаратных затрат в схеме и снижения общей стоимости реализации устройства управления. Приведен пример применения предложенного метода синтеза. Показано, что экономия аппаратных ресурсов достигает 30 % при сохранении временных характеристик устройства.

: композиционное микропрограммное устройство управления, расширение формата микрокоманд, LUT элемент, гибридные FPGA, снижение аппаратных затрат, псевдоэквивалентные ОЛЦ.

Композиционные микропрограммные устройств управления (КМУУ) являются эффективным средством реализации линейных алгоритмов управления [1, 2]. При реализации схем КМУУ возникает задача уменьшения аппаратных затрат, которая является актуальной для синтеза любых устройств управления [3]. Методы решения этой задачи во многом зависят от особенностей элементного базиса [4].

В настоящее время активно развивается базис программируемых логических интегральных схем (ПЛИС) типа гибридных FPGA (Field-Programmable Gate Array) [5, 6]. В состав таких ПЛИС входят элементы табличного типа (LUT, look-up table) и встроенные блоки программируемых логических матриц (ПЛИМ, PLA, programmable logic array). Примером могут служить микросхемы АРЕХ20К [7], которые включают блоки PLA, имеющие 32 входа ( $S = 32$ ), 16 выходов ( $t = 16$ ) и 32 термина ( $q = 32$ ). Подобные FPGA не включают встроенных блоков памяти, на которых обычно реализуется система выходных функций КМУУ [8]. В настоящей работе предлагается метод синтеза КМУУ в базе гибридных FPGA, основанный на идеях, изложенных в работах [9, 10].

Целью исследования является уменьшение аппаратных затрат в схеме КМУУ при его реализации в базе гибридной FPGA за счет введения в формат микрокоманд кодов классов псевдоэквивалентных операторных линейных цепей (ПОЛЦ). Задачей исследования является разработка метода синтеза КМУУ, позволяющего уменьшить количество макроячеек ПЛИМ в схеме формирования функций возбуждения памяти. При этом алгоритм управления представляется в виде граф-схемы алгоритма (ГСА) [3].

### 1

Обозначим в некоторой ГСА  $\Gamma$  множество вершин  $B = \{b_0, b_E\} \cup B_1 \cup B_2$  и множество дуг  $E$ , соединяющих

© Мирошкин А. Н., 2014

эти вершины. Начальную вершину обозначим  $b_0$ , конечную –  $b_E$ , множество операторных вершин –  $B_1$ , множество условных вершин –  $B_2$ . Операторная вершина  $b_q \in B_1$  содержит набор микроопераций  $Y(b_q) \subseteq Y$ , где  $Y = \{y_1, \dots, y_N\}$  – множество микроопераций (выходных сигналов), вырабатываемых устройством управления. Условная вершина  $b_p \in B_2$  содержит один элемент множества логических условий  $X = \{x_1, \dots, x_L\}$  (входных сигналов). Под линейной понимается ГСА, в которой не менее 75 % от общего числа вершин являются операторными.

Операторная линейная цепь (ОЛЦ) является последовательностью операторных вершин граф-схемы алгоритма. Сформируем множество ОЛЦ  $C = \{\alpha_1, \dots, \alpha_G\}$ . При этом каждая пара соседних вершин ОЛЦ  $\alpha_g \in C$  соединяется дугой  $e_i \in E$  [2]. Каждая ОЛЦ  $\alpha_g \in C$  имеет произвольное число входов  $I_g^k$  и только один выход  $O_g$ . Формальные определения ОЛЦ, их входов и выходов приведены в работе [2]. Отметим, что каждая вершина  $b_q \in B_1$  соответствует микрокоманде  $MI_q$ , которая хранится в управляющей памяти (УП) по адресу  $A_q$ . Для адресации микрокоманд достаточно

$$R = \log_2(M) \text{ бит}, \quad (1)$$

где  $M = |B_1|$ . Пусть каждая ОЛЦ  $\alpha_g \in C$  состоит из  $F_g$  операторных вершин, и пусть  $F_{\max} = \max(F_1, \dots, F_G)$ . Поставим в соответствие каждой ОЛЦ  $\alpha_g \in C$  двоичный код  $K(\alpha_g)$  разрядности

$$R_1 = \log_2(G), \quad (2)$$

а каждой компоненте  $b_q \in B_1$  – двоичный код  $K(b_q)$  разрядности

$$R_2 = \log_2(F_{\max}) . \tag{3}$$

Для кодирования ОЛЦ используем элементы  $\tau_r \in \tau$ , а для кодирования их компонент – элементы  $T_r \in T$ , при этом  $|\tau| = R_1$  и  $|T| = R_2$ . Кодирование компонент выполняется в естественном порядке, то есть

$$K[(b_g)_i] = K[(b_g)_{i-1}] + 1, \tag{4}$$

где  $g = \overline{1, G}$ ;  $i = \overline{1, F_G}$ . Если для ГСА  $\Gamma$  выполняется условие

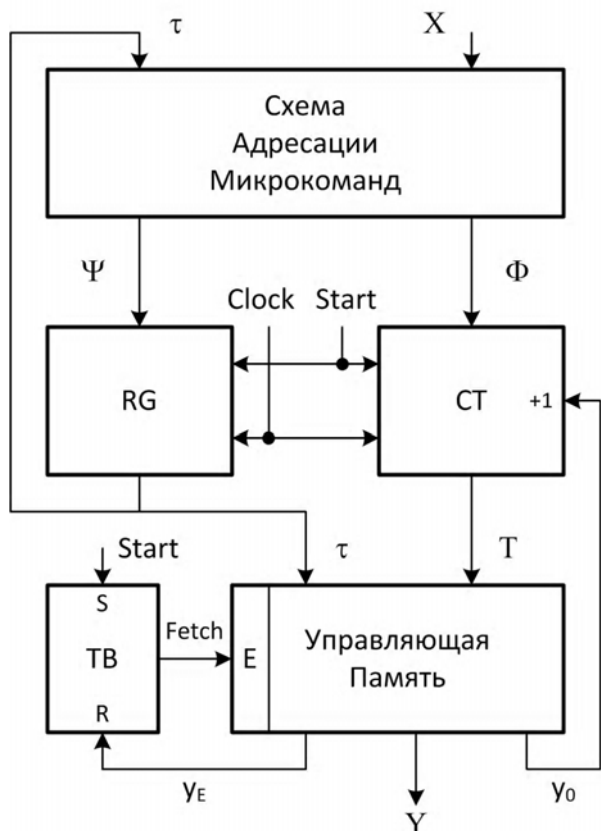
$$R_1 + R_2 = R , \tag{5}$$

то для ее интерпретации может быть использована модель КМУУ с разделением кодов (рис. 1), которую обозначим  $U_1$ .

В КМУУ  $U_1$  схема адресации микрокоманд (САМ) реализует систему функций возбуждения счетчика СТ (6) и регистра RG (7):

$$\Phi = \Phi(\tau, X) ; \tag{6}$$

$$\Psi = \Psi(\tau, X) . \tag{7}$$



. 1. Структурная схема КМУУ  $U_1$

При таком подходе адрес микрокоманды  $MI_q$  определяется как

$$A(b_q) = K(\alpha_g) * K(b_q), \tag{8}$$

где вершина  $b_q$  входит в ОЛЦ  $\alpha_g \in C$ , а символом «\*» обозначается операция конкатенации.

По сигналу Start в RG и СТ заносится начальный адрес микропрограммы, а триггер выборки ТВ устанавливается в единичное состояние, разрешая этим выборку команд из УП. Если считанная микрокоманда не соответствует выходу ОЛЦ, то одновременно с микрооперациями  $Y(b_q)$  формируется сигнал  $y_0$ , по которому к содержимому СТ прибавляется единица, формируя тем самым адрес следующей компоненты текущей ОЛЦ. Если микрокоманда соответствует выходу ОЛЦ, сигнал  $y_0$  не формируется. При этом адрес входа следующей ОЛЦ формируется схемой САМ. При достижении окончания микропрограммы формируется сигнал  $y_E$ , триггер ТВ обнуляется, и выборка команд из УП прекращается.

Число термов в схеме САМ может быть уменьшено путем введения преобразователя кодов ОЛЦ в коды классов псевдоэквивалентных ОЛЦ [2]. ОЛЦ  $\{\alpha_i, \alpha_j\} \subseteq C$  называются псевдоэквивалентными, если их выходы связаны с входом одной и той же вершины ГСА  $\Gamma$ . Следует заметить, однако, что реализация такого преобразователя требует дополнительных аппаратных ресурсов микросхемы FPGA.

В настоящей работе предлагается метод синтеза КМУУ в базе гибридных FPGA, основной целью которого является уменьшение аппаратных затрат в схеме устройства управления.

## 2

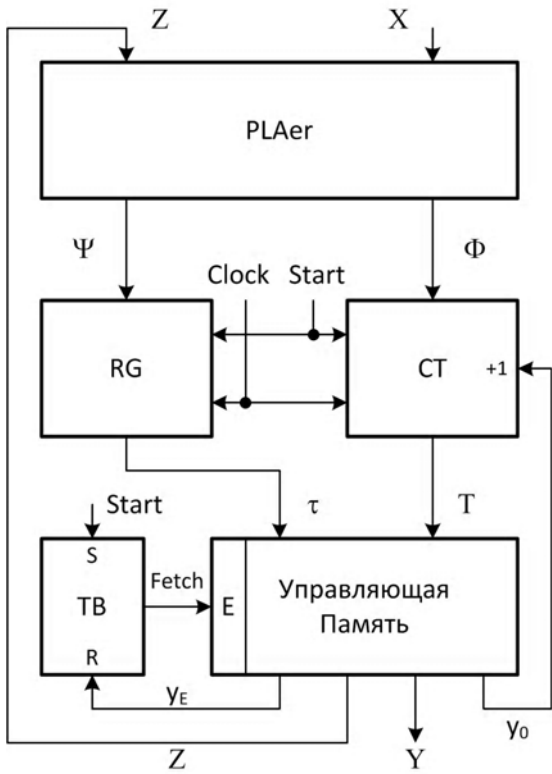
Пусть ОЛЦ  $\alpha_g \in C_1$ , если  $\alpha_g \in C_1$  и ее выход не связан с входом конечной вершины ГСА  $\Gamma$ . Найдем разбиение  $\Pi_C = \{B_1, \dots, B_I\}$  множества  $C_1$  на классы ПОЛЦ. Закодируем классы  $B_i \in \Pi_C$  двоичными кодами  $K(B_i)$  разрядности

$$R_I = \log_2(I) . \tag{9}$$

В работах [9, 10] предлагается ввести в формат микрокоманд поле  $K(B_i)$ . При этом управляющая память реализуется на встроенных блоках памяти. Однако в случае гибридных FPGA такие блоки отсутствуют. Функцию блоков встроенной памяти могут выполнять LUT элементы, которые можно рассматривать как блок памяти с  $S_L$  входами и одним выходом. Очевидно, LUT элемент имеет  $2^{S_L}$  ячеек памяти.

Пусть для данной ГСА  $\Gamma$  выполняется отношение

$$S_L \geq R . \tag{10}$$



. 2. Структурная схема КМУУ  $U_2$

В КМУУ  $U_2$  схема САМ реализуется в виде совокупности блоков PLA, называемой PLAer. Схема УП реализуется из совокупности элементов LUT, называемой LUTer. Блоки СТ и RG также реализованы на LUT элементах.

Классы  $B_i \in \Pi_C$  закодированы переменными  $z_r$ , образующими множество  $Z = \{z_1, \dots, z_{R_I}\}$ . Блок PLAer реализует функции

$$\Phi = \Phi(Z, X); \tag{11}$$

$$\Psi = \Psi(Z, X). \tag{12}$$

Блок LUTer реализует функции

$$Y = Y(\tau, T); \tag{13}$$

$$Z = Z(\tau, T) \tag{14}$$

$$y_0 = y_0(\tau, T). \tag{15}$$

В настоящей работе предлагается метод синтеза КМУУ  $U_2$ , включающий следующие этапы:

1. Формирование множеств  $C$ ,  $C_1$  и  $\Pi_C$  для ГСА  $\Gamma$ .
2. Кодирование ОЛЦ, их компонент и классов  $B_i \in \Pi_C$ .
3. Формирование содержимого блока LUTer.
4. Формирование таблицы блока PLAer.
5. Синтез логической схемы КМУУ.

3

Пусть ГСА  $\Gamma_1$  содержит  $G = 7$  ОЛЦ:  $C = \{\alpha_1, \dots, \alpha_7\}$ , где  $\alpha_7 \notin C_1$ . В множестве  $C_1$  выделены  $I=3$  класса ПОЛЦ:  $\Pi_C = \{B_1, B_2, B_3\}$ , где  $B_1 = \{\alpha_1\}$ ,  $B_2 = \{\alpha_2, \alpha_3\}$ ,  $B_3 = \{\alpha_4, \alpha_5, \alpha_6\}$ . ОЛЦ  $\alpha_i$  сформированы из следующих последовательностей операторных вершин:  $\alpha_1 = \langle b_1, b_2, b_3 \rangle$ ,  $\alpha_2 = \langle b_4, b_5, b_6, b_7 \rangle$ ,  $\alpha_3 = \langle b_8, b_9 \rangle$ ,  $\alpha_4 = \langle b_{10}, b_{11}, b_{12} \rangle$ ,  $\alpha_5 = \langle b_{13}, b_{14}, b_{15}, b_{16} \rangle$ ,  $\alpha_6 = \langle b_{17}, b_{18} \rangle$ ,  $\alpha_7 = \langle b_{19}, b_{20} \rangle$ . Согласно (2) для кодирования ОЛЦ достаточно  $R_1 = 3$  переменных из множества  $\tau = \{\tau_1, \tau_2, \tau_3\}$ . Максимальное количество компонент  $F_{\max} = 4$ , для их кодирования согласно (3) достаточно  $R_2 = 2$  переменных из множества  $T = \{T_1, T_2\}$ . В общем случае для кодирования  $M = 20$  операторных вершин согласно (1) достаточно  $R = 5$  двоичных разрядов, следовательно, условие (5) выполняется, и применение метода разделения кодов целесообразно. При этом для кодирования  $I=3$  классов ПОЛЦ согласно (9) необходимы  $R_I = 2$  переменные, которые формируют множество  $Z = \{z_1, z_2\}$ .

Закодируем ОЛЦ  $\alpha_g \in C$  и их классы произвольным образом:  $K(\alpha_1) = 000, \dots, K(\alpha_7) = 110$ ;  $K(B_1) = 00, \dots, K(B_3) = 10$ . Для выполнения условия (4) присвоим первой компоненте каждой ОЛЦ  $\alpha_g \in C$  код 00, второй – 01, третьей – 10, четвертой – 11. Это позволит определить адреса  $A(b_q)$  микрокоманд КМУУ  $U_2(\Gamma_1)$ , показанные в табл. 1. Здесь и далее запись  $U_i(\Gamma_j)$  обозначает КМУУ  $U_i$ , которое интерпретирует ГСА  $\Gamma_j$ .

Из табл. 1 имеем, например,  $A(b_6) = 00110$ ,  $A(b_{18}) = 10101$  и т. д.

Формат микрокоманд КМУУ  $U_2$  включает поля  $y_0$ ,  $y_E$ ,  $FY$ ,  $FB$ , где поле  $FY$  содержит код набора микроопераций, а поле  $FB$  – код класса  $B_i \in \Pi_C$ . Если  $y_0 = 1$ , то содержимое поля  $FB$  игнорируется.

Содержимое блока LUTer КМУУ  $U_2(\Gamma_1)$  показано в табл. 2. Принцип формирования содержимого блока LUTer тривиален. В строку с адресом  $A(b_q)$  записывается набор микроопераций  $Y(b_q)$ . Если вершина  $b_q \in B_1$  не является выходом ОЛЦ  $\alpha_g \in C$ , то в строку с адресом  $A(b_q)$  записывается микрооперация  $y_0$ . В противном случае в эту строку записывается код  $K(B_i)$ , где  $\alpha_g \in B_i$ . Если вершина  $b_q \in B_1$  связана с конечной вершиной ГСА, то в строку с адресом  $A(b_q)$  заносится микрооперация  $y_E$ .

В табл. 2 дано символічне содериме блоку LUTer, а перехід до бітовим строкам не представляє складностей.

Пусть переходы из выходов ОЛЦ  $\alpha_g \in C_1$  представлены следующей системой обобщенных формул переходов [2]:

$$\begin{aligned} B_1 &\rightarrow x_1 b_4 \vee \overline{x_1 x_2} b_6 \vee \overline{x_1 x_2} b_8; \\ B_2 &\rightarrow x_3 x_4 b_{10} \vee x_3 \overline{x_4} b_{13} \vee \overline{x_3 x_5} b_{19} \vee \overline{x_3 x_5} b_{16}; \\ B_3 &\rightarrow x_5 b_{11} \vee \overline{x_5 x_3} b_{17} \vee \overline{x_5 x_3} b_8. \end{aligned} \tag{16}$$

Подобная система является основой для формирования таблицы блока PLAer со столбцами  $B_i, K(B_i), b_q, A(b_q), X_h, \Psi_h, \Phi_h, h$ . Назначение столбцов ясно из табл. 3, задающей переходы для класса  $B_3 \in \Pi_C$ .

Адреса микрокоманд берутся з табл. 1. Отметим, что  $\Psi = \{D_1, D_2, D_3\}, \Phi = \{D_4, D_5\}$ . Общее число строк  $H_2(\Gamma_j)$  в таблице блока КМУУ  $U_2(\Gamma_j)$  совпадает с числом термов в системе обобщенных формул переходов.

1. Адреса микрокоманд КМУУ  $U_2(\Gamma_1)$

$\tau_1 \tau_2 \tau_3$	000	001	010	011	100	101	110
$T_1 T_2$							
00	$b_1$	$b_4$	$b_8$	$b_{10}$	$b_{13}$	$b_{17}$	$b_{19}$
01	$b_2$	$b_5$	$b_9$	$b_{11}$	$b_{14}$	$b_{18}$	$b_{20}$
10	$b_3$	$b_6$	*	$b_{12}$	$b_{15}$	*	*
11	*	$b_7$	*	*	$b_{16}$	*	*

2. Сoderиме блоку LUTer КМУУ  $U_2(\Gamma_1)$

$\tau_1 \tau_2 \tau_3$	000	001	010	011	100	101	110
$T_1 T_2$							
00	$y_0 Y(b_1)$	$y_0 Y(b_4)$	$y_0 Y(b_8)$	$y_0 Y(b_{10})$	$y_0 Y(b_{13})$	$y_0 Y(b_{17})$	$y_0 Y(b_{19})$
01	$y_0 Y(b_2)$	$y_0 Y(b_5)$	$z_2 Y(b_9)$	$y_0 Y(b_{11})$	$y_0 Y(b_{14})$	$z_1 Y(b_{18})$	$y_E Y(b_{20})$
10	$Y(b_3)$	$y_0 Y(b_6)$	*	$z_1 Y(b_{12})$	$y_0 Y(b_{15})$	*	*
11	*	$z_2 Y(b_7)$	*	*	$z_1 Y(b_{16})$	*	*

3. Фрагмент таблицы блоку PLAer КМУУ  $U_2(\Gamma_1)$

$B_i$	$K(B_i)$		$b_q$	$A(b_q)$					$X_h$	$\Psi_h$	$\Phi_h$	$h$
	$z_1$	$z_2$		$\tau_1$	$\tau_2$	$\tau_3$	$T_1$	$T_2$				
$B_3$	1	0	$b_{11}$	0	1	1	0	1	$x_5$	$D_2 D_3$	$D_5$	1
			$b_{17}$	1	0	1	0	0	$\overline{x_5 x_3}$	$D_1 D_3$	-	2
			$b_8$	0	1	0	0	0	$\overline{x_5 x_3}$	$D_2$	-	3

В нашем примере,  $H_2(\Gamma_1)=10$ . Отметим, что  $H_1(\Gamma_1)=20$ , где  $H_i(\Gamma_j)$  обозначает число строк в таблице блоку PLAer КМУУ  $U_i$ , интерпретирующей ГСА  $\Gamma_j$ .

Системы (11)–(12) формируются по таблице переходов. Так, из табл. 3 можно построить фрагменты ДНФ:

$$\begin{aligned} D_1 &= \overline{z_1 z_2 x_5 x_3}; \\ D_2 &= z_1 \overline{z_2 x_5} \vee z_1 \overline{z_2 x_5 x_3}; \\ D_3 &= z_1 \overline{z_2 x_5} \vee z_1 z_2 \overline{x_5 x_3}. \end{aligned} \tag{17}$$

При выполнении условий

$$S \geq L + R_1 + R_3; \tag{18}$$

$$t \geq R_1 + R_2; \tag{19}$$

$$q \geq H_2(\Gamma_j), \tag{20}$$

блок PLAer тривиальным образом реализуется на одной макроячейке PLA. Если эти отношения нарушаются, то требуется несколько макроячеек. Для уменьшения числа макроячеек PLA в схеме блоку PLAer можно использовать известные методы [11].

При выполнении условия (10) каждая функция из систем (12)–(14) реализуется на одном элементе LUT. Такое решение является оптимальным. При этом таблица блоку LUTer рассматривается как таблица истинности функций (13)–(15).

Предлагаемый метод расширения формата микрокоманд за счет введения поля с кодом класса псевдоэквивалентных ОЛЦ ориентирован на уменьшение числа макроячеек PLA в схеме формирования адреса микрокоманд. При этом число тактов интерпретации алгоритма управления совпадает с соответствующим значением для базовой структуры КМУУ  $U_1$  с разделением кодов.

Уменьшение числа термов в ДНФ функций возбуждения памяти может привести к уменьшению числа уровней в комбинационной части КМУУ. Это в свою очередь приводит к повышению быстродействия цифровой системы в целом. Рассмотренные нами примеры показали, что число макроячеек PLA в зависимости от их параметров уменьшается на значение до 30 % по сравнению с КМУУ  $U_1$ . Напомним, что применение этого метода целесообразно только для линейных ГСА при выполнении условия (5).

Научная новизна предложенного метода состоит в использовании классов псевдоэквивалентных ОЛЦ для уменьшения числа макроячеек PLA в схеме адресации микрокоманд. Практическая значимость метода заключается в уменьшении числа макроячеек при реализации схемы КМУУ, что позволяет получить схемы, обладающие меньшей стоимостью, чем известные аналоги.

Отметим, что при нарушении условий (18)–(20) использование изложенного подхода невозможно, а структура КМУУ и соответствующий метод синтеза нуждаются в модификации, на что и направлены дальнейшие направления исследований.

1. *Barkalov, A. Logic Synthesis for Compositional Microprogram Control Unit / A. Barkalov, L. Titarenko. – Berlin : Springer, 2008. – 272 p.*

2. Synthesis of compositional microprogram control unit with dedicated area of inputs / Alexander Barkalov, Larysa Titarenko, Jacek Bieganowski, A.N. Miroshkin // W: Design of digital systems and devices / eds M. Adamski, A. Barkalov, M. Wegrzyn. – Berlin: Springer-Verlag, 2011. (Lecture Notes in Electrical Engineering; 79). – P. 193–214.
3. *Баркалов, А. А. Синтез устройств управления на программируемых логических устройствах / А. А. Баркалов. – Донецк, ДНТУ, 2002. – 262 с.*
4. *Соловьев, В. В. Проектирование цифровых схем на основе программируемых логических интегральных схем / Соловьев В. В. – М. : Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.*
5. *Kabiani, A. The Hybrid Field Programmable Architecture / A. Kabiani, S. Brown // IEEE Design & Test of Computers. – 1999. – V. 16, No. 4. – P. 74–83.*
6. *Singh, S. K. Design flow of reconfigurable embedded system architecture using LUTs/PLAs / S. K. Singh, R. K. Singh // 2nd IEEE International Conference on Parallel Distributed and Grid Computing (PDGC). – 6–8 Dec. – 2012. – P. 385–390.*
7. Altera Corporation APEX20K PLD Family Data Sheet (2004) [электронный ресурс]. – Режим доступа: [www.altera.com/literature/ds/apex.pdf](http://www.altera.com/literature/ds/apex.pdf)
8. *Баркалов, А. А. Синтез композиционных микропрограммных устройств управления / А. А. Баркалов, Л. А. Титаренко. – Харьков : Коллегиум, 2007. – 302 с.*
9. *Баркалов, А. А. Расширение формата микрокоманд в микропрограммном устройстве управления с разделением кодов / А. А. Баркалов, Л. А. Титаренко, А. Н. Мирошкин // Радиоэлектроника и информатика. – № 2 (41), апрель-июнь. – 2008. – С. 40–45.*
10. *Баркалов, А. А. Расширение формата микрокоманд в композиционном микропрограммном устройстве управления с элементаризацией операторных линейных цепей / А. А. Баркалов, А. А. Красичков, А. Н. Мирошкин // Радиоелектронні і комп'ютерні системи. – 2010. – № 7. – С. 301–305.*
11. *Баранов, С. И. Цифровые устройства на проектируемых БИС с матричной структурой / С. И. Баранов, В. А. Скляров. – М. : Радио и связь, 1986. – 272 с.*

Стаття надійшла до редакції 03.02.2014.

Мірошкін О. М.

Канд. техн. наук, доцент, ДВНЗ «Донецький національний технічний університет», Україна

**FPGA**

У статті пропонується метод синтезу композиційного мікропрограмного пристрою керування в базисі гібридних FPGA. Пропонується використання кодів класів псевдоеквівалентних операторних лінійних ланцюгів для зменшення апаратних витрат в схемі та зниження загальної вартості реалізації пристрою керування. Наведений приклад використання запропонованого метода синтезу. Показано, що економія апаратних витрат досягає 30 % при збереженні часових характеристик пристрою.

: композиційний мікропрограмний пристрій керування, розширення формату мікрокоманд, LUT елемент, гібридні FPGA, зниження апаратних витрат, псевдоеквівалентні ОЛЦ.

Miroshkin A. N.

Ph.D. in computer engineering, assistant professor at Computer Engineering Department, Donetsk National Technical University, Ukraine

#### **EXTENSION OF MICROINSTRUCTION FORMAT FOR CONTROL UNIT IMPLEMENTATION ON HYBRID FPGA**

A synthesis method of compositional microprogram control unit on hybrid FPGA (that contain table elements, and programmable logic arrays) is proposed. The main purpose of the method is hardware amount reduction in the control unit circuit. Usage of classes codes of pseudo-equivalent operational linear chains for hardware amount reducing in the circuit and overall cost reducing of the control device is proposed. An example of the proposed synthesis method application is given. It is shown that depending on the basis circuit

parameters and characteristics of the control algorithm hardware resource saving can reach 30 % in comparison to the costs of basic circuit control device implementation. Time characteristics of control units of modified and base structured are equivalent.

**Keywords:** compositional microprogram control unit, microinstruction format extension, LUT element, hybrid FPGA, reduction of hardware amount, pseudoequivalent OLC.

## REFERENCES

1. Barkalov A., Titarenko L. Logic Synthesis for Compositional Microprogram Control Unit. Berlin, Springer, 2008, 272 p.
2. Synthesis of compositional microprogram control unit with dedicated area of inputs / Alexander Barkalov, Larysa Titarenko, Jacek Bieganski, A.N. Miroshkin, W: Design of digital systems and devices / eds M. Adamski, A. Barkalov, M. Wegrzyn. Berlin, Springer-Verlag, 2011. (Lecture Notes in Electrical Engineering; 79), pp. 193–214.
3. Barkalov A. A. Sintez ustrojstv upravleniya na programiruemix logicheskix ustrojstvax. Doneck, DNTU, 2002, 262 p.
4. Solov'ev V.V. Proektirovanie cifrovix sxem na osnove programiruemix logicheskix integral'nyx sxem. Moscow, Goryachaya liniya-TELEKOM, 2001, 636 p.
5. Kabiani A., Brown S. The Hybrid Field Programmable Architecture, IEEE Design & Test of Computers, 1999, V.16, No. 4, pp. 74–83.
6. Singh S. K., Singh R. K. Design flow of reconfigurable embedded system architecture using LUTs/PLAs. 2nd IEEE International Conference on Parallel Distributed and Grid Computing (PDGC), 6–8 Dec. 2012, pp. 385–390.
7. Altera Corporation APEX20K PLD Family Data Sheet (2004) [e'lektronnyj resurs], Rezhim dostupa: [www.altera.com/literature/ds/apex.pdf](http://www.altera.com/literature/ds/apex.pdf)
8. Barkalov A. A., Titarenko L. A. Sintez kompozicionnyx mikroprogrammnyx ustrojstv upravleniya. Xar'kov, Kollegium, 2007, 302 p.
9. Barkalov A. A., Titarenko L. A., Miroshkin A. N., Barkalov A. A. Rasshirenie formata mikrokomand v mikroprogrammnom ustrojstve upravleniya s razdeleniem kodov, *Radioelektronika i informatika*, No. 2 (41), aprel'iyun', 2008, pp. 40–45.
10. Barkalov A. A., Krasichkov A. A., Miroshkin A. N. Rasshirenie formata mikrokomand v kompozicionnom mikroprogrammnom ustrojstve upravleniya s e'lementarizaciej operatornyx linejnyx cepej, *Radioelektron. i komp'yut. sistemi*, 2010, No. 7, pp. 301–305.
11. Baranov S. I., Sklyarov V. A. Cifrovyje ustrojstva na proektiruemix BIS s matrichnoj strukturoj. Moscow, Radio i svyaz', 1986, 272 p.