

А.А. Баркалов¹, д-р техн. наук, проф.,
Л.А. Титаренко¹, д-р техн. наук, проф.,
К.Н. Ефименко², канд. техн. наук, доц.,
И.Я. Зеленева², канд. техн. наук, доц.

¹Університет Зеленогурський, г. Зелена Гора, Польща,

²Донецький національний технічний університет, г. Донецьк, Україна

A.Barkalov@iie.uz.zgora.pl

Реализация КМУУ с элементарными цепями на гибридных FPGA

Предлагается метод уменьшения аппаратных затрат в схеме КМУУ с элементарными цепями, ориентированный на технологию гибридных FPGA. Метод основан на замене LUT-элементов схемой, состоящей из встроенных блоков PLA (programmable logic array), что возможно при использовании данной технологии. Такой подход позволит уменьшить число LUT элементов в схеме адресации КМУУ. Приведен пример применения предложенного метода.

Ключевые слова: КМУУ, ГСА, ЭОЛЦ, гибридные FPGA, логическая схема.

Введение

Задача разработки и усовершенствования устройств управления (УУ) цифровыми системами не утратила своей актуальности и в настоящее время. Одним из путей реализации схемы УУ по линейной граф-схеме алгоритма (ГСА) является использование модели композиционного микропрограммного устройства управления (КМУУ) с элементарными цепями [1,2]. При этом для реализации схем УУ широко используются программируемые логические интегральные схемы (ПЛИС) вида FPGA (field-programmable gate arrays) [3-6]. Основу FPGA составляют макроячейки LUT (look-up table), имеющие ограниченное число входов (4-6) [7,8]. Для оптимизации схемы УУ на FPGA необходимо уменьшить количество аргументов и термов в реализуемых системах булевых функций [9]. В настоящей работе предлагается метод решения этой задачи для КМУУ с элементарными цепями. Метод основан на использовании двух источников кодов классов псевдоэквивалентных элементарных операторных линейных цепей (ЭОЛЦ) и замене LUT-элементов схемой, состоящей из встроенных блоков PLA (programmable logic array). Последнее возможно при использовании технологии гибридных FPGA [10,11], которая активно развивается в настоящее время. Предлагаемый метод является развитием идей из работ [12,13]. Используемые в данной работе алгоритмы управления, представлены в виде граф-схемы алгоритма (ГСА) [9]. Этот выбор определен наглядностью подобного представления и широким применением аппарата ГСА в практике инженерного проектирования

Целью исследования является оптимизация схемы КМУУ с элементарными цепями за счет использования блоков PAL гибридных FPGA.

Задачей исследования является разработка метода синтеза КМУУ с элементарными цепями, позволяющего уменьшить число LUT-элементов в комбинационной части КМУУ.

Базовая информация о КМУУ с элементарными цепями

Пусть ГСА $\Gamma = \Gamma(V, E)$ представлена множествами вершин V и соединяющих их дуг E . Пусть $V = b_0 \cup b_E \cup E_1 \cup E_2$, где b_0 – начальная вершина, b_E – конечная вершина, E_1 – множество операторных вершин и E_2 – множество условных вершин ГСА Γ . В операторных вершинах $b_q \in E_1$ записываются наборы микроопераций $Y(b_q) \subseteq Y$, где $Y = \{y_1, \dots, y_N\}$ – множество микроопераций. В условных вершинах $b_q \in E_2$ записываются элементы множества логических условий $X = \{x_1, \dots, x_L\}$.

Введем ряд определений, взятых из [2].

Определение 1. Операторной линейной цепью (ОЛЦ) ГСА Γ называется конечная последовательность операторных вершин

$\alpha_g = \langle b_{g1}, \dots, b_{gE_g} \rangle$ такая, что для любой пары соседних компонент b_{gi}, b_{gi+1} , где i – номер компоненты кортежа α_g , существует дуга $\langle b_{gi}, b_{gi+1} \rangle \in E$.

Определение 2. Вершина $b_q \in D^g$, где D^g – множество вершин, входящих в ОЛЦ α_g , называется входом ОЛЦ α_g , если существует дуга $\langle b_t, b_q \rangle \in E$, где $b_t \notin D^g$. Элементарная ОЛЦ (ЭОЛЦ) имеет только один вход.

Определение 3. Вершина $b_q \in D^g$, называется выходом ОЛЦ α_g , если существует дуга $\langle b_q, b_t \rangle \in E$, где $b_t \notin D^g$.

Определение 4. ЭОЛЦ α_i, α_j называются псевдоэквивалентными ЭОЛЦ, если их выходы связаны со входом одной и той же вершины $b_q \in V$.

Пусть для некоторой ГСА Γ сформировано множество ЭОЛЦ $C = \{\alpha_1, \dots, \alpha_G\}$, определяющее разбиение на множестве E_1 [3], и пусть $|E_1| = M$. Поставим в соответствие каждой вершине $b_q \in E_1$ микрокоманду MI_q с адресом $A(b_q)$, имеющим разрядность

$$R = \lceil \log_2 M \rceil. \quad (1)$$

Пусть $F_{\max} = \max(F_1, \dots, F_G)$ – максимальное число компонент в ЭОЛЦ. Закодируем каждую ЭОЛЦ $\alpha_g \in C$ двоичным кодом $K(\alpha_g)$, имеющим R_1 разрядов, где

$$R_1 = \lceil \log_2 G \rceil. \quad (2)$$

Для определения любой вершины $b_q \in D^g$ достаточно R_2 разрядов, представляющих код $K(b_q)$.

При этом

$$R_2 = \lceil \log_2 F_{\max} \rceil. \quad (3)$$

Пусть для ГСА Γ выполняется следующее условие:

$$R_1 + R_2 = R. \quad (4)$$

В этом случае для реализации алгоритма Γ целесообразно использовать модель КМУУ с элементарными цепями (рис. 1). В этой модели для кодирования ЭОЛЦ используются переменные $\tau_t \in \tau$, где $|\tau| = R_1$. Для кодирования компонент ЭОЛЦ используются переменные $T_t \in T$, где $|T| = R_2$. Коды компонент выбраны так, чтобы выполнялась естественная адресация микрокоманд [1]. Для этого код первой компоненты любой ЭОЛЦ равен 0, второй – 1 и так далее. Естественно, что эти десятичные числа представлены их двоичными R_2 -разрядными эквивалентами.

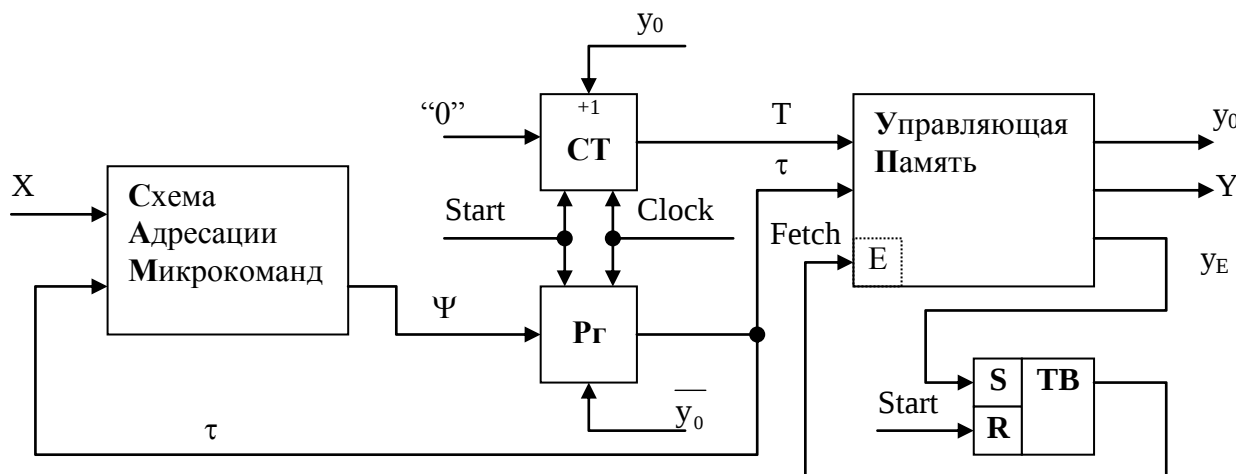


Рисунок 1 – Структурная схема КМУУ с элементарными цепями

Условимся в дальнейшем обозначать КМУУ (рис. 1) символом U_1 .

В КМУУ U_1 схема адресации микрокоманд (САМ) реализует систему функций возбуждения триггера Pr

$$\Psi = \Psi(\tau, X). \quad (5)$$

При этом адрес микрокоманды MI_q представляется в виде

$$A(b_q) = K(\alpha_g) * K(b_q), \quad (6)$$

где вершина b_q входит в состав ЭОЛЦ $\alpha_g \in C$, * – знак операции конкатенации.

Принцип функционирования КМУУ U_1 следующий. По сигналу $Start$ в Pr и CT заносится начальный адрес микропрограммы, а триггер выборки TB устанавливается в единичное состояние. При этом $Fetch = 1$, что разрешает выборку ко-

манд из управляющей памяти (УП). Если считанная микрокоманда не соответствует выходу ЭОЛЦ, то одновременно с микрооперациями $Y(b_q)$ формируется сигнал y_0 . Если $y_0 = 1$, то к содержимому CT прибавляется единица и адресуется следующая компонента текущей ЭОЛЦ. Если выход ОЛЦ достигнут, то $y_0 = 0$. При этом адрес входа следующей ЭОЛЦ формируется схемой САМ. При достижении окончания микропрограммы формируется сигнал y_e , триггер TB обнуляется, и выборка микрокоманд прекращается.

Число LUT-элементов в схеме САМ зависит от числа аргументов и термов в системе (5). В настоящей работе предлагается метод, позволяющий уменьшить сложность функций в системе (5) и, следовательно, уменьшить аппаратные затраты в схеме САМ.

Основная идея предлагаемого метода

$$Z = Z(\tau). \quad (9)$$

Пусть ЭОЛЦ $\alpha_g \in C_1$, если O_g не связан с конечной вершиной ГСА Г. Найдем разбиение $\Pi_C = \{B_1, \dots, B_I\}$ множества C_1 на классы псевдо-эквивалентных ЭОЛЦ (ПЭОЛЦ). Выполним кодирование $\alpha_g \in C$ так, чтобы максимально возможное число классов $B_i \in \Pi_C$, где $|\Pi_C| = I$, представлялось одним обобщенным интервалом R_1 -мерного булева пространства. Пусть n_i – число обобщенных интервалов, представляющих класс. Представим множество Π_C в виде $\Pi_C = \Pi_A \cup \Pi_B$. При этом множества Π_A и Π_B строятся следующим образом:

$$\begin{aligned} (n_i = 1) &\rightarrow B_i \in \Pi_A, \\ (n_i > 1) &\rightarrow B_i \in \Pi_B. \end{aligned} \quad (7)$$

Источником кодов классов $B_i \in \Pi_A$ является регистр Рг. При этом код класса $B_i \in \Pi_A$ определяется соответствующим интервалом R_1 -мерного булева пространства.

Закодируем классы $B_i \in \Pi_B$ двоичными кодами $C(B_i)$ разрядности

$$R_3 = \lceil \log_2(|\Pi_B| + 1) \rceil. \quad (8)$$

Используем для кодирования классов $B_i \in \Pi_B$ переменные из множества $Z = \{z_1, \dots, z_{R_3}\}$. Для формирования кодов $C(B_i)$ необходим блок преобразователя кодов (БПК). Этот блок реализует систему функций

Очевидно, для реализации схемы БПК необходимы некоторые ресурсы кристалла FPGA. В настоящей работе предлагается уменьшить эти ресурсы за счет использования избыточности встроенных блоков памяти ЕМВ, входящих в состав FPGA [7,8]. Эти блоки имеют фиксированную емкость, однако их конфигурация может меняться.

Для реализации схемы $Y \cup \{y_0, y_E\}$ необходимо

$$n_f = \left\lceil \frac{N+2}{t_F} \right\rceil \quad (10)$$

блоков ЕМВ, имеющих не менее M слов. При этом параметр t_F (число выходов блока) определяется как

$$t_F = \left\lceil \frac{V}{2^R} \right\rceil. \quad (11)$$

Этот параметр затем уточняется как ближайшее меньшее целое к одному из элементов множества фиксированных выходов $O_F = \{1, 2, 4, 8, 16\}$ [6,7]. Общее число выходов памяти можно найти как

$$t_O = \lceil n_f \cdot t_F \rceil. \quad (12)$$

Если выполняется условие

$$t_O - (N+3) \geq 1, \quad (13)$$

то для реализации схемы устройства управления предлагается модель КМУУ U_2 (рис. 2). В этой модели схема блока САМ реализуется на встроенных блоках PLA.

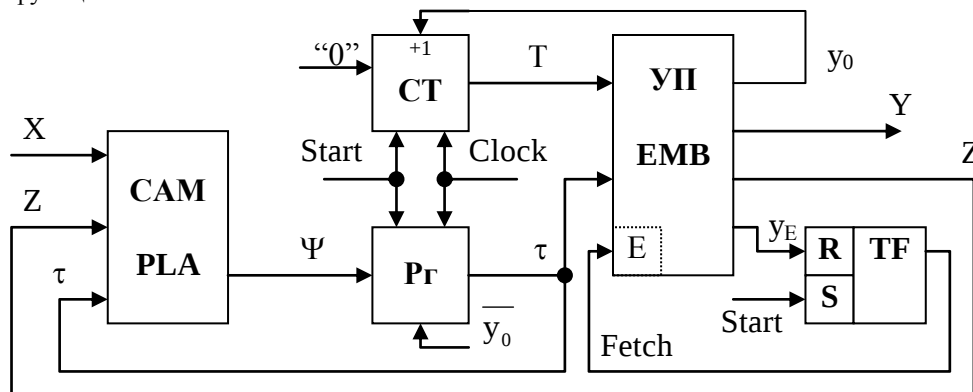


Рисунок 2 – Структурная схема КМУУ U_2

В КМУУ U_2 регистр Рг является источником кодов классов $B_i \in \Pi_A$. Коды классов $B_i \in \Pi_B$ формирует управляющая память. Схема блока САМ реализуется на встроенных PLA, имеющих s входов, t выходов и q термов. При выполнении условий

$$L + R_1 + R_3 \leq s, \quad (14)$$

$$R_1 \leq t, \quad (15)$$

$$H_0 \leq q, \quad (16)$$

блок САМ представляется в виде одного блока

PLA.

Блок САМ формирует систему функций

$$\Psi = \Psi(\tau, Z, X). \quad (17)$$

Эта система имеет H_0 термов, где H_0 – число переходов автомата Мили, реализованного по преобразованной ГСА [3].

Итак, в КМУУ U_2 используются следующие ресурсы кристалла гибридной FPGA: LUT-элементы (схемы Рг и CT), блоки PLA (схема САМ) и ЕМВ (схема УП). Блок УП реализуется на реконфигурируемых блоках памяти ЕМВ [7,8]. Эти блоки имеют строго определенное число вы-

ходов, входящих в множество $\{1, 2, 4, 8, 18, 36\}$. При этом имеется высокая вероятность наличие неиспользованных выходов блоков ЕМВ, входящих в состав УП устройства U_1 .

Особенности реализации схемы КМУУ U_2

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

1. Формирование для ГСА Γ множеств C , C_1 , и Π_C .
2. Оптимальное кодирование ЭОЛЦ $\alpha_g \in C_1$ и кодирование компонент ЭОЛЦ.
3. Формирование множеств Π_A и Π_B .
4. Кодирование классов $B_i \in \Pi_B$ кодами $C(B_i)$.
5. Формирование таблицы переходов КМУУ.
6. Формирование содержимого управляющей памяти.
7. Синтез схемы КМУУ в заданном базисе.

Этапы 1-4 выполняются по известным методикам [1-3]. Этап 7 связан с разработкой VHDL-моделей КМУУ и использованием стандартных промышленных пакетов [6,7]. Эти этапы не представляют особого интереса для иллюстрации синтеза схемы КМУУ U_2 . В этой связи мы не рассматриваем данные этапы в нашей статье.

Пусть для некоторой ГСА Γ_1 получено множество ЭОЛЦ $C = \{\alpha_1, \dots, \alpha_{16}\}$ и разбиение $\Pi_C = \{B_1, \dots, B_7\}$. При этом $\alpha_{16} \notin C_1$, а классы $B_i \in \Pi_C$ определяются следующим образом: $B_1 = \{\alpha_1\}$, $B_2 = \{\alpha_2, \alpha_3, \alpha_4\}$, $B_3 = \{\alpha_5, \alpha_6\}$, $B_4 = \{\alpha_7, \alpha_8, \alpha_9\}$, $B_5 = \{\alpha_{10}, \alpha_{11}\}$, $B_6 = \{\alpha_{12}\}$, $B_7 = \{\alpha_{13}, \alpha_{14}, \alpha_{15}\}$. Итак, $G = 16$, $R_1 = 4$, $\tau = \{\tau_1, \dots, \tau_4\}$.

Оптимальное кодирование ЭОЛЦ $\alpha_g \in C_1$ выполняется так, чтобы максимально возможное число классов $B_i \in \Pi_C$ представлялось одним интервалом R_1 -мерного булева пространства [1]. Один из вариантов кодирования представлен на рис. 3.

		$\tau_3 \tau_4$			
		00	01	11	10
$\tau_1 \tau_2$	00	α_1	α_2	α_3	α_4
	01	α_5	α_6	α_{10}	α_{11}
	11	α_7	α_8	α_9	α_{16}
	10	α_{13}	α_{14}	α_{15}	α_{12}

Рисунок 3 – Коды ЭОЛЦ $\alpha_g \in C$

Из карты Карно (рис. 3) можно получить следующие интервалы, определяющие классы $B_i \in \Pi_C$. Класс B_1 определяется интервалом 0000; класс B_2 – интервалами 00*1 и 001*; класс B_3 – интервалом 010*; класс B_4 – интервалами 110* и 11*1; класс B_5 – интервалом 011*; класс B_6 – интервалом 1010; класс B_7 – интервалами 100* и 10*1. Таким образом, $\Pi_A = \{B_1, B_3, B_5, B_6\}$, где $K(B_1) = 0000$, $K(B_3) = 010*$, $K(B_5) = 011*$ и $K(B_6) = 1010$. Очевидно, $\Pi_B = \{B_2, B_4, B_7\}$ и для их кодирования необходимо $R_3 = 2$ разряда. Итак, $Z = \{z_1, z_2\}$. Закодируем классы $B_i \in \Pi_B$ следующим образом: $C(B_2) = 11$, $C(B_4) = 01$, $C(B_7) = 10$.

Таблица переходов формируется на основе обобщённых формул переходов [1]. Пусть, например, из ГСА Γ_1 можно получить следующие формулы:

$$\begin{aligned} B_1 &\rightarrow x_1 b_3 \vee \overline{x_1} x_2 b_8 \vee \overline{x_1} x_2 b_{12}; \\ B_2 &\rightarrow x_4 b_{12} \vee \overline{x_4} b_{17}. \end{aligned} \quad (18)$$

Для правильного функционирования КМУУ U_2 необходимо, чтобы блок САМ различал множества Π_A и Π_B . С этой целью достаточно использовать код 00, формируемый схемой УП.

Таблица переходов включают столбцы: B_i (класс разбиения Π_C); $K(B_i)$; $C(B_i)$; $A(b_q)$ (адрес микрокоманды, соответствующей вершине b_q в формуле перехода для класса B_i); X_h (входной сигнал, определяющий переход из B_i ; в b_q); Ψ_h (набор функций возбуждения триггеров Γ); h (номер перехода, $h = \overline{1, N_0}$). Фрагмент таблицы переходов, соответствующий системе (18), приведен в табл.1.

Таблица 1. Фрагмент таблицы переходов для системы (18)

B_i	$K(B_i)$	$C(B_i)$	$A(b_q)$	X_h	Ψ_h	h
B_1	0000	11	000100	x_1	D_4	1
			001000	$\overline{x_1} x_2$	D_3	2
			010101	$\overline{x_1} x_2$	$D_2 D_4$	3
B_2	****	00	010101	x_4	$D_2 D_4$	4
			110010	$\overline{x_4}$	$D_1 D_2$	5

Этот фрагмент построен в предположении, что микрокоманды имеют следующие адреса: $A(b_3) = 000100$, $A(b_8) = 001000$, $A(b_{12}) = 010101$, $A(b_{17}) = 110010$. Система (17) формируется из таблицы переходов. Например, из табл.1 можно получить функции: $D_1 = z_1 z_2 x_4$ (строка 5) и $D_4 = \tau_1 \tau_2 \tau_3 \tau_4 z_1 z_2 (x_1 \vee \overline{x_1} x_2) \vee z_1 z_2 x_4$ (строки 1, 3 и 4).

При формировании содержимого УП в ячейки памяти заносятся микрооперации [2] и коды $C(B_i)$. Коды заносятся следующим образом. Пусть b_q – выход ОЛЦ $\alpha_g \in B_i$. Если $B_i \in \Pi_A$, то

по адресу $A(b_q)$ заносится код признака (в рассматриваемом примере это код 00). Если $B_i \in P_B$, то по адресу $A(b_q)$ заносится код $C(B_i)$.

Заключение

Разработанный метод усовершенствования КМУУ основан на применении технологии гибридных FPGA со встроенными блоками PLA и использовании двух источников кодов классов псевдоэквивалентных ЭОЛЦ, что позволяет гарантированно уменьшить число термов в системе функций возбуждения триггеров регистра и счетчика адресов микрокоманд до максимально возможной величины. Если КМУУ с элементарными цепями рассматривать как автомат Мура, то предлагаемый подход позволяет уменьшить число термов до величины этого параметра у эквивалентного автомата Мили. Применение предложенного метода имеет смысл только при наличии

свободных ресурсов встроенных блоков памяти, входящих в состав микросхемы FPGA. Схема адресации микрокоманд реализуется на встроенных блоках PLA. Такой подход позволяет заменить LUT-элементы встроенными PLA. Это уменьшает число межсоединений, что ведет к уменьшению потребляемой энергии и времени задержки [1].

Научная новизна предложенного метода заключается в использовании особенностей КМУУ (наличие классов псевдоэквивалентных ЭОЛЦ) и гибридных FPGA (фиксированное число выходов блоков ЕМВ и наличие встроенных блоков PLA) для уменьшения числа LUT-элементов в схеме КМУУ с элементарными цепями.

Практическая значимость метода заключается в уменьшении площади кристалла FPGA, занимаемой схемой КМУУ, что позволяет получить схемы, обладающие меньшей стоимостью, чем известные из литературы аналоги.

Список литературы

1. Палагин А.В. Проектирование реконфигурируемых цифровых систем / Палагин А.В., Баркалов А.А., Титаренко Л.А., Опанасенко В.Н. – Луганск: Издательство ВНУ, 2011. – 432 с.
2. Barkalov A., Titarenko L. Logic synthesis for compositional microprogram control units. – Berlin: Springer, 2009. – 272 pp.
3. Баркалов А.А. Синтез микропрограммных автоматов на заказных и программируемых СБИС / А.А. Баркалов, Л.А.Титаренко. – Донецк: УНИТЕХ, 2009. – 336 с.
4. Barkalov A., Titarenko L. Logic synthesis for FSM-based control units. – Berlin: Springer, 2009. – 233 pp.
5. Maxfield S. The Design Warrior's Guide to FPGAs. – Amsterdam: Elsevier, 2004. – 541 pp.
6. Грушвицкий Р.И. Проектирование систем на микросхемах с программируемой структурой / Р.И. Грушвицкий, А.Х. Мурсаев, Е.П. Угрюмов. – СПб: БХВ, 2006. – 736 с.
7. <http://www.xilinx.com>
8. <http://www.altera.com>
9. Baranov S. Logic and System Design of Digital Systems. - Tallinn: TTU, 2008. – 266 pp.
10. Kaviani F., Brown S. The Hybrid Field Programmable Architecture// IEEE Design & Test of Computers. – 1999, V.16, N4. – pp. 74-83.
11. Sigh S., Singh R., Bhatia V. Performance Evaluation of Hybrid Reconfigurable Computing Architectures over Symmetrical FPGA// International Journal of Embedded Systems and Applications. – 2012, V.2, N3. – pp. 107-116.
12. Баркалов А.А. Оптимизация схемы КМУУ с общей памятью / Баркалов А.А., Титаренко Л.А., Ефименко К.Н., Липински Я.М. // УСиМ. – 2011. – №5. – С. 47-52.
13. Баркалов А.А. Оптимизация схемы КМУУ с преобразователем адреса микрокоманд / Баркалов А.А., Титаренко Л.А., Ефименко К.Н., Липински Я.М. // Наукові праці ДонНТУ. Серія „Проблеми моделювання та автоматизації проектування”. – 2011. – Вип.9 (179). – С.26-35.

Надійшла до редакції 15.03.2014

О.О. БАРКАЛОВ¹, Л.О. ТИТАРЕНКО¹, К.М. ЄФІМЕНКО², І.Я. ЗЕЛЕНЬОВА²

¹Університет Зеленогурський (Польща)

²Донецький національний технічний університет (Україна)

РЕАЛІЗАЦІЯ КМПК ІЗ ЕЛЕМЕНТАРНИМИ ЛАНЦЮГАМИ НА ГІБРИДНИХ FPGA

В роботі запропоновано метод зменшення апаратних витрат у схемі КМПК із елементарними ланцюгами, який орієнтовано на технологію FPGA. Метод засновано на використанні двох джерел кодів класів псевдоеквівалентних ЕОЛЛ та технології гібридних FPGA. Такій підхід дозволить зменшити число LUT елементів у схемі адресації КМПК. Наведено приклад використання запропонованого методу.

Ключові слова: КМПК, ГСА, ЕОЛЛ, гібридні FPGA, логічна схема.

A.A. BARKALOV¹, L.A. TITARENKO¹, K.N. EFIMENKO², I.J. ZELENJOVA²

¹University of Zielona góra (Poland)²Donetsk National Technical University (Ukraine)**IMPLEMENTING CMCU WITH ELEMENTARY CHAINS BY HYBRID FPGA**

The proposed method is oriented to reducing the hardware amount of the composite microprogramming control unit (CMCU) scheme with elementary chains, using the technology of hybrid FPGA. To optimize the hardware cost in the scheme of the control device in the FPGA it is necessary to reduce the number of arguments and implemented systems in terms of Boolean functions. The idea of the proposed in this paper method is based on the two sources of codes of classes of pseudoequivalent elementary operational linear chains (EOLC) and replacement of LUT-element for a circuit consisting of embedded blocks PLA (programmable logic array). This is possible by using hybrid technology FPGA, which is actively developing now. The proposed structure of compositional microprogram control unit uses the following resources of crystal hybrid FPGA: LUT-elements for the register and the counter of transition address, PLA blocks for scheme of microinstructions addressing in control memory. The control memory is implemented on reconfigurable memory blocks EMB. These blocks have a specific number of outputs and inputs. At the same time there is a high probability of having unused outputs of EMB included in the control memory. In this paper we propose a method for the synthesis of CMCU, comprising the steps of: forming the set of elementary linear chains for a given flow-chart of the control algorithm; optimal encoding of elementary chains and encoding of their components; formation of sets of pseudoequivalent chains, their optimal encoding; formation of the transition table of compositional microprogramming control unit; formation of control memory content; synthesis of scheme CMCU in a given basis.

An example of application of the proposed method is given. Scientific novelty of the proposed method is to use features CMCU (classes of pseudoequivalent EOLC) and hybrid FPGA (fixed number of block outputs and a built- EMB blocks PLA) to reduce the number of elements in the LUT-scheme CMCU with elementary chains. The practical significance of the method is to reduce the chip area FPGA, occupied by CMCU scheme that allows getting schemes which have a lower cost than prior known analogues.

Key words: CMCU, GSA, EOLC, hybrid FPGA, logic circuit.