

**А.А. Баркалов¹, Л.А. Титаренко¹,
К.Н. Ефименко², И.Я. Зеленёва²**

Университет Зеленогурский, г. Зеленая Гура, Польша¹,

A.Barkalov@iie.uz.zgora.pl¹

ГВУЗ «Донецкий национальный технический университет», г. Донецк².

РЕАЛИЗАЦИЯ КМУУ С ОБЩЕЙ ПАМЯТЬЮ НА ГИБРИДНЫХ FPGA

Предлагается метод уменьшения аппаратных затрат в схеме КМУУ с общей памятью, ориентированный на технологию гибридных FPGA. Метод основан на использовании трех источников кодов классов псевдоэквивалентных ОЛЦ и реализации схемы адресации микрокоманд на программируемых логических матрицах. Такой подход позволяет уменьшить площадь кристалла, занимаемую схемой устройства управления. Приведен пример применения предложенного метода.

КМУУ, ГСА, ОЛЦ, гибридные FPGA, логическая схема

Введение

В случае если алгоритм управления некоторой системы представлен линейной граф-схемой алгоритма (ГСА) [1], для реализации схемы устройства управления (УУ) может быть использована модель композиционного микропрограммного устройства управления (КМУУ) с общей памятью [2,3]. В настоящее время для реализации схем УУ широко используются программируемые логические интегральные схемы (ПЛИС) вида FPGA (field-programmable gate arrays) [4,5]. Основу FPGA представляют макроячейки LUT (look-up table), имеющие ограниченное число входов (4-6) [6,7]. Кроме того, в настоящее время развивается технология гибридных FPGA [8,9], в состав которых входят LUT элементы, встроенные блоки памяти (EMB, embedded memory blocks) и встроенные программируемые логические матрицы (PLA, programmable logic array). При этом появляется возможность реализации части схемы на блоках PLA. Это позволяет уменьшить площадь кристалла, занимаемую схемой КМУУ. Метод основан на использовании трех источников кодов классов псевдоэквивалентных операторных линейных цепей (ОЛЦ). Предложенный метод развивает идеи из работы [10].

Целью исследования является оптимизация схемы КМУУ с общей памятью за счет использования возможностей, представляемых технологией гибридных FPGA.

Задачей исследования является разработка метода синтеза КМУУ с общей памятью, позволяющего уменьшить число LUT-элементов в комби-

национной части КМУУ.

Композиционное МУУ с общей памятью

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 [1].

Определение 1. Операторной линейной цепью ГСА Γ называется конечная последовательность операторных вершин $\bar{b}_g = \langle b_{g1}, \dots, b_{gF_g} \rangle$ такая, что для любой пары соседних компонент b_{gi}, b_{gi+1} , где i – номер компоненты кортежа \bar{b}_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 \bar{b}_g , называется входом ОЛЦ \bar{b}_g , если существует дуга $\langle b_t, b_q \rangle \in E$, где $b_t \notin D^g$.

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

Определение 4. ОЛЦ \bar{b}_i, \bar{b}_j называются псевдоэквивалентными ОЛЦ, если их выходы связаны со входом одной и той же вершины $b_q \in V$.

Пусть для некоторой ГСА Γ сформировано множество ОЛЦ $S = \{\bar{b}_1, \dots, \bar{b}_G\}$, 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

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

Используем для адресации микрокоманд переменные $T_r \in T$, где $|T| = R$. Адресация выполняется таким образом, чтобы выполнялось условие

$$A(b_{gi+1}) = A(b_{gi}) + 1, \quad (2)$$

где $b_{gi}, b_{gi+1} \in D^g$ и $\langle b_{gi}, b_{gi+1} \rangle \in E$.

В этом случае УУ может быть реализовано в виде КМУУ U_1 (рис. 1), называемом КМУУ с общей памятью [2,3]. Принципы функционирования КМУУ U_1 достаточно полно представлены в литературе [1-3].

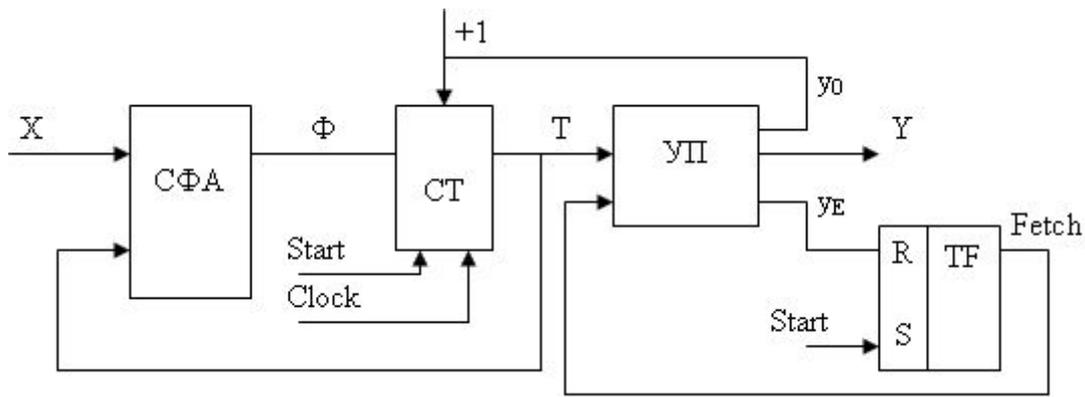


Рисунок 1 – Структурная схема КМУУ U_1

В КМУУ U_1 схема формирования адреса (СФА) реализует систему функций возбуждения триггеров счетчика СТ, определяемую как

$$\Phi = \Phi(T, X). \quad (3)$$

При этом, как правило, счетчик имеет информационные входы типа D [6,7]. По сигналу Start счетчик СТ устанавливается в ноль, что соответствует адресу первой микрокоманд реализуемого алгоритма. По сигналу $y_0 = 1$ содержимое СТ увеличивается на единицу, что соответствует режиму (2). Управляющая память (УП) хранит наборы микроопераций $Y(b_q) \subseteq Y$ и переменные y_0 (управление СТ) и y_E (признак окончания алгоритма). Триггер считывания TF формирует сигнал Fetch, разрешающий выборку микрокоманд из УП. При достижении окончания алгоритма формируется переменная $y_E = 1$, что приводит к $\text{Fetch} = 0$ и прекращению выборки из УП.

При использовании FPGA схемы СФА, СТ и TF реализуются на LUT, а схема УП – на встроенных блоках памяти ЕМВ. Основным недостатком КМУУ U_1 является значительное число термов в системе функций (3). Это приводит к увеличению числа LUT элементов и их уровней в схеме СФА. Для устранения этого недостатка используют оптимальную адресацию микрокоманд, что приводит к КМУУ U_2 [10]. Однако такая адресация не всегда возможна. Уменьшение числа термов системе (3) гарантируется в КМУУ U_3 , где осуществляется преобразование адресов микрокоманд в коды классов псевдоэквивалентных ОЛЦ [2,3]. Однако это связано с введением в схему КМУУ дополнительного блока преобразователя адресов (БПА), который потребляет некоторые ресурсы кристалла. В настоящей работе предлагается метод синтеза КМУУ, позволяющий уменьшить как число термов в системе (3), так и число LUT элементов в схемах БПА и СФА.

Основная идея предлагаемого метода

Найдем разбиение $\Pi_C = \{B_1, \dots, B_l\}$ множества ОЛЦ C на классы

псевдоэквивалентных ОЛЦ. Выполним адресацию микрокоманд так, чтобы при выполнении условия (2) максимально возможное число классов $V_i \in \Pi_C$ выражалось одним интервалом R -разрядного булева пространства. Представим разбиение Π_C в виде $\Pi_C = \Pi_1 \cup \Pi_2$, где $\Pi_1 \cap \Pi_2 = \emptyset$. Пусть $V_i \in \Pi_1$, если этот класс представляется более, чем одним интервалом пространства кодирования. Остальные классы принадлежат множеству Π_2 .

Характерной особенностью блоков ЕМВ является фиксированное число выходов $t_F \in \{1, 2, 4, 8, 16, 32, 64\}$. Пусть V_0 означает число ячеек ЕМВ при $t_F = 1$. Тогда число ячеек V при некотором фиксированном значении t_F можно определить как

$$V = \lceil V_0 / t_F \rceil. \quad (4)$$

Для реализации управляющей памяти КМУУ достаточно M ячеек ЕМВ. При этом блок будет иметь

$$t_M = \lceil V_0 / M \rceil \quad (5)$$

выходов. Пусть следующее условие выполняется для некоторой ГСА Γ и блоки ЕМВ используемой микросхемы FPGA

$$t_M > N + 3. \quad (6)$$

Закодируем классы $V_i \in \Pi_1$ двоичными кодами $K(V_i)$ разрядности

$$R_1 = \lceil \log_2 I_1 \rceil, \quad (7)$$

где $I_1 = |\Pi_1|$. Используем для кодирования переменные $\tau_r \in \tau$, где $|\tau| = R_1$. Пусть $\Pi_1 \neq \emptyset$, $\Pi_2 \neq \emptyset$ и следующее условие выполняется для ЕМВ одновременно с условием (6):

$$t_M < N + 3 + R_1. \quad (8)$$

Очевидно, что часть разрядов кодов $K(V_i)$ для классов $V_i \in \Pi_1$ может быть сформирована схемой УП. Теперь УП формирует R_2 разряда кода $K(V_i)$, а БПА – R_3 разряда:

$$R_2 = t_M - (N + 3); \quad (9)$$

$$R_3 = R_1 - R_2. \quad (10)$$

При этом множество τ может быть представлено в виде $\tau = \tau^1 \cup \tau^2$, где $|\tau^1| = R_3$, $|\tau^2| = R_2$. Предлагаемая в работе модель КМУУ основана на представлениях $\Pi_C = \Pi_1 \cup \Pi_2$ и $\tau = \tau^1 \cup \tau^2$. На этих представлениях основана и модель U_4 из работы [11]. Однако в предлагаемой структуре U_5 схема СФА реализуется на блоках PLA (рис. 2).

Схема СФА реализует функции Φ в следующем виде:

$$\Phi = \Phi(T, \tau, X). \quad (11)$$

Блок БПА реализует часть разрядов кодов $K(V_i)$, образующих множество τ^2 . В отличие от КМУУ U_4 [11] в схеме отсутствует мультиплексор выбо-

ра источника функций Φ , что является несомненным преимуществом.

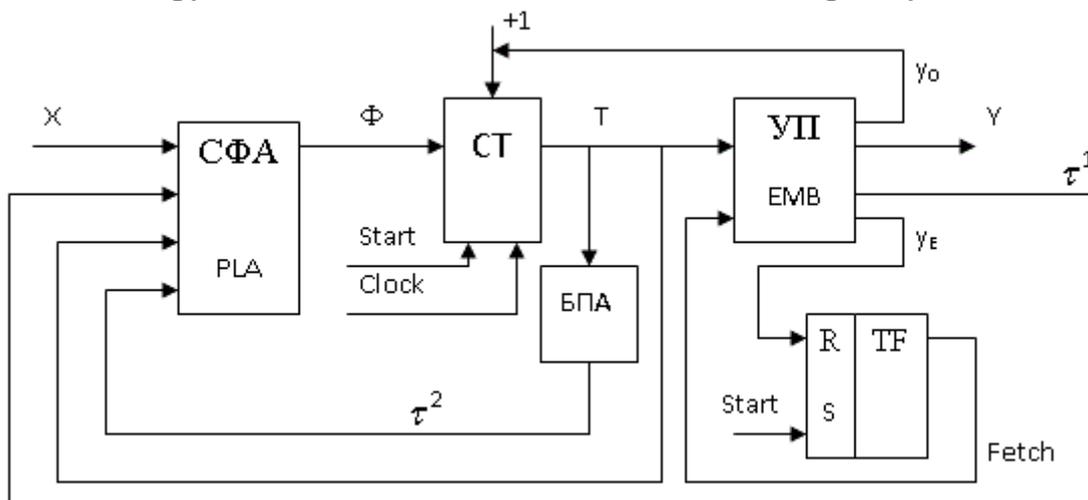


Рисунок 2 – Структурная схема КМУУ U_5

Такой подход позволяет уменьшить число термов в системе (3) до абсолютно возможного минимума. Кроме того, уменьшается сложность блока БПА по сравнению с КМУУ U_3 . Отметим, что при $\Pi_1 = \emptyset$ КМУУ U_5 вырождается в U_2 . При $\Pi_2 = \emptyset$ КМУУ U_5 превращается в КМУУ U_3 . Недостатком является увеличение числа выходов блока УП. Однако этот блок строится из реконфигурируемых блоков ЕМВ, которые имеют строго определенное число выходов [6,7]. При этом имеется высокая вероятность наличия неиспользованных выходов.

В настоящей работе предлагается метод синтеза КМУУ U_5 , включающий следующие этапы:

1. Формирование множества ОЛЦ S для ГСА Γ .
2. Формирование разбиения Π_C множества S .
3. Оптимальная адресация микрокоманд.
4. Кодирование классов $V_i \in \Pi_1$.
5. Формирование таблицы переходов КМУУ.
6. Формирование таблицы блока преобразователя адреса.
7. Формирование таблицы содержимого управляющей памяти.
8. Реализация схемы КМУУ в заданном элементном базисе.

Пример применения предложенного метода

Пусть для некоторой ГСА Γ получено множество ОЛЦ $S = \{b_1, \dots, b_{12}\}$, где $b_1 = \langle b_1, \dots, b_4 \rangle$, $b_2 = \langle b_5 \rangle$, $b_3 = \langle b_6, b_7 \rangle$, $b_4 = \langle b_8, b_9, b_{10}, b_{11}, b_{12} \rangle$, $b_5 = \langle b_{13}, b_{14} \rangle$, $b_6 = \langle b_{15}, b_{16} \rangle$, $b_7 = \langle b_{17}, \dots, b_{20} \rangle$, $b_8 = \langle b_{21}, \dots, b_{24} \rangle$, $b_9 = \langle b_{25}, b_{26} \rangle$, $b_{10} = \langle b_{27}, b_{28} \rangle$, $b_{11} = \langle b_{29}, b_{30}, b_{31} \rangle$, $b_{12} = \langle b_{32} \rangle$. Пусть эти ОЛЦ могут быть разбиты на $I = 7$ классов, где

$V_1 = \{b_1\}$, $V_2 = \{b_2, b_3\}$, $V_3 = \{b_4\}$, $V_4 = \{b_5, b_6\}$, $V_5 = \{b_7, b_8, b_9\}$, $V_6 = \{b_{10}, b_{11}\}$, $V_7 = \{b_{12}\}$. Пусть выход ОЛЦ $b_{12} \in C$ связан с входом вершины b_E . Как известно, переходы из таких ОЛЦ не рассматриваются, так как их последняя вершина должна включать переменную u_E [2].

Выполним адресацию микрокоманд так, чтобы выполнялось условие (2) и максимально возможное число классов представлялось одним обобщенным интервалом R -мерного булева пространства. В рассматриваемом примере $M = 32$, то есть $R = 5$ и $T = \{T_1, \dots, T_5\}$. Один из возможных вариантов оптимальной адресации микрокоманд приведен на рис. 3. Этот рисунок содержит видоизмененную карту Карно, которая достаточна для получения обобщенных интервалов, соответствующих кодам классов $V_i \in \Pi_C$. Символ $U_5(\Gamma)$ означает, что КМУУ U_5 реализуется по ГСА Γ .

Следующие интервалы могут быть найдены для классов $V_i \in \Pi_C$. Класс V_1 соответствует интервалу 000^{**} ; класс V_2 – интервалам 0010^* и 00110 ; класс V_3 – интервалам 010^{**} и 00111 ; класс V_4 – интервалу 011^{**} ; класс V_5 – интервалам 10^{***} и 1100^* ; класс V_6 – интервалам 1101^{**} , 1110^* и 11110 . Итак, имеем следующие классы для разбиения Π_C : $\Pi_1 = \{V_2, V_3, V_5, V_6\}$ и $\Pi_2 = \{V_1, V_4\}$. Для кодирования классов $V_i \in \Pi_1$ необходимо $R_1 = 2$ переменных $\tau_r \in \{\tau\}$.

		$T_1 T_2 T_3$			V_2				
		000	001	010	011	100	101	110	111
$T_4 T_5$	00	b_1	b_5	b_9	b_{13}	b_{17}	b_{21}	b_{25}	b_{29}
	01	b_2	b_6	b_{10}	b_{14}	b_{18}	b_{22}	b_{26}	b_{30}
10	b_3	b_7	b_{11}	b_{15}	b_{19}	b_{23}	b_{27}	b_{31}	
11	b_4	b_8	b_{12}	b_{16}	b_{20}	b_{24}	b_{28}	b_{32}	
		V_1	V_3	V_4	V_5	V_6	V_7		

Рисунок 3 – Адреса микрокоманд КМУУ $U_5(\Gamma)$

Однако необходим код, позволяющий идентифицировать тот факт, что $V_i \notin \Pi_1$. В этой связи (в данном примере) разрядность R_1 увеличивается до 3. Пусть код 000 соответствует условию $V_i \notin \Pi_1$. Закодируем классы $V_i \in \Pi_1$ следующим образом: $K(V_2) = 001$, $K(V_3) = 010$, $K(V_5) = 100$ и $K(V_6) = 101$.

Для формирования таблицы переходов необходимо построить систему обобщенных формул перехода [2,3] для классов $V_i \in \Pi_C$. Пусть для классов $V_1, V_2 \in \Pi_C$ получены следующие формулы:

$$\begin{aligned}
V_1 &\rightarrow x_1 b_5 \vee \overline{x_1} b_8; \\
V_2 &\rightarrow x_2 b_{17} \vee \overline{x_2} x_3 b_8 \vee \overline{x_2} \overline{x_3} b_{13}.
\end{aligned}
\tag{12}$$

Фрагмент таблицы переходов КМУУ $U_5(\Gamma)$, соответствующий (12), приведен в табл.1.

Таблица 1. Фрагмент таблицы переходов КМУУ $U_5(\Gamma)$

V_i	$K(V_i)2$	$K(V_i)1$	b_q	$A(b_q)$	X_h	Φ_h	h
V_1	000**	000	b_5	00100	x_1	D_3	1
			b_8	00111	$\overline{x_1}$	$D_3 D_4 D_5$	2
V_2	*****	001	b_{17}	10000	x_2	D_1	3
			b_8	00111	$\overline{x_2} x_3$	$D_3 D_4 D_5$	4
			b_{13}	01100	$\overline{x_2} \overline{x_3}$	$D_2 D_3$	5

В столбце $K(V_i)1$ указан код класса $V_i \in \Pi_1$, в столбце $K(V_i)2$ – код класса $V_i \in \Pi_2$. Если рассматриваются переходы из класса $V_i \in \Pi_2$, то в строке записан код 000. В противном случае, код из столбца $K(V_i)2$ игнорируется.

Таблица переходов служит для формирования функций (11). Например, из табл.1 может быть получена функция

$$D_3 = \overline{T_1} \overline{T_2} \overline{T_3} \tau_1 \tau_2 \tau_3 \vee \tau_1 \tau_2 \tau_3 x_2.$$

Остальные этапы синтеза выполняются аналогично их выполнению в [10]. Только этап 7 имеет некоторые отличия. Рассмотрим следующий пример.

Пусть в вершине b_{31} ГСА Γ записан набор микроопераций $y_{3,7}$. Из предыдущего материала ясно, что вершина b_{31} является выходом ОЛЦ b_{11} , которая входит в класс V_6 . Класс V_6 имеет код $K(V_6) = 101$, а вершине b_{31} соответствует адрес 11110 (рис. 3). Таким образом, в ячейку ЕМВ с адресом 11110 должен быть помещен код набора $y_{3,7}$ и переменная τ_3 .

Итак, переменные $\tau_r \in \{\tau^1\}$ помещаются в ячейки УП, соответствующие выходам ОЛЦ, входящим в классы $V_i \in \Pi_1$. Схема БПА реализуется на LUT элементах. Очевидно, что разбиение множества τ на классы τ^1 и τ^2 надо производить следующим образом. В класс τ^2 помещаются переменные $\tau_r \in \{\tau\}$, которым соответствуют схемы с наименьшим числом LUT элементов. Такой подход позволяет уменьшить аппаратные затраты в схеме БПА. Дальнейшая реализация схемы КМУУ U_5 сводится к реализации системы функций на PLA и УП на ЕМВ. Для решения этой задачи используются стандартные промышленные пакеты [6,7]. Этот этап выходит за пределы нашей статьи.

Отметим, что реализация предложенного подхода возможна только благодаря большому числу входов PLA. Например, в блоках PLA микросхем APX20K имеется $S=32$ входа [7]. В общем случае предложенный метод целесообразен при выполнении условия

$$R_1 + R + L \leq S. \quad (13)$$

Анализ стандартных примеров [12] показал, что условие (13) выполняется для 87% МПА из этой библиотеки.

Выводы

Предлагаемый в работе метод уменьшения аппаратных затрат в схеме КМУУ основан на учете особенностей гибридных FPGA, а также наличии классов псевдоэквивалентных ОЛЦ. Использование двух источников кодов классов позволяет гарантированно уменьшить число термов в системе функций возбуждения триггеров счетчика адресов микрокоманд до максимально возможной величины. Если КМУУ с общей памятью рассматривать как автомат Мура, то предлагаемый подход позволяет уменьшить число термов до величины этого параметра у эквивалентного автомата Мили. Кроме того, уменьшается число LUT элементов в схеме преобразователя адреса, так как не все адреса выходов ОЛЦ подлежат преобразованию.

Кроме того, при выполнении условия (13) схема СФА реализуется в виде одного блока PLA. При этом существенно уменьшается площадь кристалла, занимаемая схемой СФА. Это во многом объясняется уменьшением числа межсоединений по сравнению с КМУУ $U_1 - U_4$.

Научная новизна предложенного метода заключается в использовании особенностей базиса гибридных FPGA (большой коэффициент объединения по входам блоков PLA) для уменьшения числа LUT элементов в схеме КМУУ.

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

Дальнейшие направления работы связаны с разработкой метода синтеза КМУУ, уменьшающего число блоков PLA в схеме адресации при нарушении условия (13).

Литература

1. Barkalov A., Titarenko L. Logic synthesis for compositional microprogram control units. – Berlin: Springer, 2008. –272 pp.
2. Баркалов А.А., Титаренко Л.А. Синтез микропрограммных автоматов на заказных и программируемых СБИС. – Донецк: УНИТЕХ, 2009. –336 с.
3. Barkalov A., Titarenko L. Logic synthesis for FSM-based control units. – Berlin:

Springer, 2009. –233 pp.

4. Maxfield S. The Design Warrior's Guide to FPGAs. – Amsterdam: Elsevier, 2004. – 541 pp.

5. Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах с программируемой структурой – С-Пб: БХВ – Петербург, 2006. – 736 с.

6. xilinx.com.

7. altera.com.

8. Kaviani F., Brown S. The Hybrid Field Programmable Architecture// IEEE Design & Test of Computers. – 1999, V.16, N4. – pp. 74-83.

9. 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.

10. Баркалов А.А., Титаренко Л.А., Ефименко К.Н., Липински Я.М. Оптимизация схемы КМУУ с общей памятью // УСиМ. –2011. – №5. – С. 47-52.

11. Баркалов А.А., Титаренко Л.А., Ефименко К.Н., Зеленёва И.Я. Разделение схемы адресации в КМУУ с общей памятью. // Известия ТТИ ЮФУ-ДонНТУ. Материалы Четырнадцатого Международного научно-практического семинара «Практика и перспективы развития партнерства в сфере высшей школы» В 3х кн. – Таганрог. Изд-во ТТИ ЮФУ. Кн. 2. 2013, №13 – С. 11-16.

12. Yang S. Logic synthesis and optimization bench marks user guide. Technical report. – Microelectronic Center of North Carolina, 1991. – 44pp.

Надійшла до редакції 25.10.2013 р.

Рецензент:

О.О. Баркалов¹, Л.О. Титаренко¹, К.М. Єфіменко², І.Я. Зеленьова²
Університет Зеленогурьський (Польща)¹,
ДВНЗ «Донецький національний технічний університет» (Україна)²

Реалізація КМУУ із загальною пам'яттю на гібридних FPGA. Пропонується метод зменшення апаратних витрат в схемі КМПК із загальною пам'яттю, який орієнтовано на технологію гібридних FPGA. Метод засновано на використанні трьох джерел код класів псевдоеквівалентних ОЛЛ і реалізації схеми адресації мікрокоманд на програмованих логічних матрицях. Такий підхід дозволяє зменшити площу кристала, яку займає схема пристрою керування. Наведено приклад використання запропонованого методу.
КМПК, ГСА, ОЛЛ, гібридні FPGA, логічна схема

A.A. Barkalov¹, L.A. Titarenko¹, K.N. Efimenko², I.J. Zelenjova²
University of Zielona góra (Poland)¹, Donetsk National Technical University (Ukraine)²

Implementing CMCU with common memory by hybrid FPGA. A method for reducing the hardware amount in the circuit of CMCU with common memory is proposed oriented on hybrid FPGA technology. The method is based on the use of three sources of codes classes of pseudoequivalent OLC and implementing the block of microinstruction addressing with programmable logic arrays. Such approach allows reducing the chip area occupied by the circuit of CMCU. An example of the proposed method application is given.
CMCU, GSA, OLC, hybrid FPGA, logic circuit