

УДК 004.3

## Применение преобразователя кодов для уменьшения аппаратурных затрат в схеме устройства управления с разделением кодов

Баркалов А.А.<sup>1</sup>, Ковалёв С.А., Лаврик А.С.<sup>2</sup>  
University of Zielona Gora, Zielona Gora1, Poland, ДонНТУ2, Донецк, Украина  
a.barkalov@iie.uz.zgora.pl, alexandrlavrik@cs.donntu.edu.ua

### Abstract

*Barkalov A., Kovalyov S., Lavrik A. Application of code converter for control unit with code sharing hardware amount reduction. The method of hardware reduction is proposed which is oriented on compositional microprogram control units with code sharing and PAL-based CPLD chips. The method is based on a wide fan-in of PAL macrocells allowing using more than one source for codes of operational linear chains. An example of proposed method application is given.*

### Введение

Одним из важных блоков цифровых систем является устройство управления (УУ) [1]. Если реализуемый алгоритм имеет линейный характер, для его представления можно использовать модель композиционного микропрограммного устройства управления (КМУУ) с разделением кодов [4]. В настоящее время для реализации логических схем устройств управления широко используются программируемые логические интегральные схемы (ПЛИС) типа CPLD (Complex Programmable Logic Devices) [6]. Такие ПЛИС включают в себя макроячейки программируемой матричной логики (ПМЛ), обладающие большим коэффициентом объединения по входу (несколько десятков) [2,3]. Высокая стоимость этого базиса требует решения актуальной задачи уменьшения числа корпусов ПЛИС в схеме. При её решении следует учитывать не только специфику интерпретируемого алгоритма управления, но и особенности элементного базиса. В таком случае, благодаря большому коэффициенту объединения по входу макроячеек ПМЛ, коды классов псевдоэквивалентных операторных линейных цепей (ОЛЦ) могут быть представлены более чем одним источником [7,8]. В данной работе предлагается метод уменьшения аппаратурных затрат в схеме КМУУ, основанный на использовании трёх источников кодов.

Целью исследования является уменьшение аппаратурных затрат в комбинационной схеме КМУУ за счёт использования нескольких источников кода классов псевдоэквивалентных операторных линейных цепей.

Задачей исследования является разработка метода синтеза, позволяющего уменьшить число

макроячеек ПМЛ в схеме блока преобразователя кода.

### Особенности КМУУ с разделением кодов

Пусть ГСА  $\Gamma$  представлена множеством вершин  $V$  и дуг  $E$ . Пусть  $B = \{b_0, b_E\} \cup E_1 \cup E_2$ , где  $b_0$  – начальная вершина,  $b_E$  – конечная вершина,  $E_1$  – множество операторных вершин, где  $|E_1| = M$ ,  $E_2$  – множество условных вершин. Вершина  $b_q \in E_1$  содержит набор микроопераций (МО)  $Y(b_q) \subseteq Y$ , где  $Y = \{y_1, \dots, y_N\}$  – множество микроопераций [1]. Каждая вершина  $b_q \in E_2$  содержит элементы множества логических условий  $X = \{x_1, \dots, x_L\}$ .

Сформируем набор операторных линейных цепей  $C = \{\alpha_1, \dots, \alpha_G\}$  для ГСА  $\Gamma$ , где каждая ОЛЦ  $\alpha_g \in C$  – это последовательность операторных вершин, причём каждой паре её соседних вершин соответствует одна из дуг ГСА. Каждая ОЛЦ  $\alpha_g \in C$  имеет только один выход  $O_g$  и произвольное количество входов. Определения ОЛЦ, их входов выходов могут быть найдены в [4].

Назовём ГСА  $\Gamma$  линейной ГСА (ЛГСА), если выполняется следующее условие:

$$\frac{M}{G} \geq 2. \quad (1)$$

Каждая вершина  $b_q \in E_1$  соответствует микрооперации  $MI_q$ , хранимой в управляющей памяти (УП) КМУУ по адресу  $A_q$ . Для адресации микрокоманд достаточно

$$R = \lceil \log_2 M \rceil \quad (2)$$

бит. Пусть ОЛЦ  $\alpha_g \in C$  включает  $F_g$  компонент и пусть  $Q = \max(F_1, \dots, F_G)$ . Закодируем ОЛЦ  $\alpha_g \in C$  двоичными кодами  $K(\alpha_g)$  разрядности  $R_G$ , где

$$R_G = \lceil \log_2 G \rceil. \quad (3)$$

Закодируем каждый компонент ОЛЦ  $\alpha_g \in C$  двоичными кодами  $K(b_q)$  разрядности  $R_Q$ , где

$$R_Q = \lceil \log_2 Q \rceil. \quad (4)$$

Пусть компоненты закодированы таким образом, что

$$K(b_{gi+1}) = K(b_{gi}) + 1, \quad (5)$$

где  $i = 1, \dots, F_g - 1, g = 1, \dots, G$ . Пусть для кодирования компонент ОЛЦ используются переменные  $T_r \in T$ , а для кодирования ОЛЦ – переменные  $\tau_r \in \tau$ , где  $|\tau| = R_G, |T| = R_Q$ . В случае, если выполняется условие

$$R_G + R_Q = R, \quad (6)$$

ЛГСА  $\Gamma$  может быть представлена моделью КМУУ с разделением кодов  $U_1$  (Рис. 1).

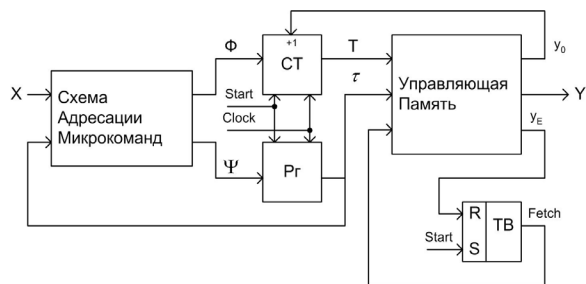


Рисунок 1 – Структурная схема КМУУ  $U_1$

В КМУУ  $U_1$ , схема адресации микрокоманд (САМ) реализует систему функций возбуждения счётчика СТ и регистра Рг:

$$\Phi = \Phi(\tau, X), \quad (7)$$

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

При этом, адрес микрокоманды  $MI_q$  представляется в виде:

$$A(b_q) = K(\alpha_g) * K(b_q), \quad (8)$$

где вершина  $b_q$  – это компонент ОЛЦ  $\alpha_g \in C$  и “\*” – знак операции конкатенации.

КМУУ  $U_1$  работает следующим образом. По сигналу  $Start = 1$  начальный адрес (все нули) заносится в Рг и СТ. В тоже время, триггер выборки ТВ устанавливается в единичное состояние. При этом  $Fetch = 1$ , что разрешает выборку микрокоманд из УП. Каждая ячейка УП содержит микрооперации  $y_n \in Y$  и специальные

переменные  $y_0$  и  $y_E$ . Если  $y_0 = 1$ , то к текущему содержимому СТ прибавляется единица и адресуется следующий компонент текущей ОЛЦ. Иначе – значения СТ и Рг загружаются из САМ. Первый случай соответствует переходу из любого компонента ОЛЦ кроме выхода. Второй случай соответствует переходу из выхода ОЛЦ. Если  $y_E = 1$ , триггер выборки обнуляется, сигнал  $Fetch = 0$  и работа КМУУ прекращается. Это соответствует переходу из вершины  $b_q \in E_1$ , где  $\langle b_q, b_E \rangle \in E$ .

Отметим, что ОЛЦ  $\alpha_i, \alpha_j \in C$  являются псевдоэквивалентными ОЛЦ (ПОЛЦ) [4], если их выходы соединены с одной и той же вершиной ГСА  $\Gamma$ . Аппаратурные затраты в логической схеме САМ могут быть уменьшены введением специального блока преобразователя кодов (ПК), преобразующего коды ОЛЦ в коды классов ПОЛЦ. В этом случае САМ реализует  $H_0$  термов, где  $H_0$  – число переходов эквивалентного автомата Мили, что является минимально возможным числом переходов [4]. Однако этот блок потребляет некоторые ресурсы ПЛИС или ППЗУ, из которых строится УП.

В данной работе предлагается метод синтеза КМУУ  $U_2$ , в котором используется три источника кодов классов ПОЛЦ, что позволяет снизить аппаратные затраты, потребляемые блоком ПК, а в определённых случаях, этот блок вообще может не использоваться.

### Основная идея предлагаемого метода

Отметим, что схемы блоков САМ, СТ, Рг и ТВ реализуются на макроячейках ПМЛ. Для реализации УП требуются внешние ППЗУ, имеющие  $t$  выходов, где  $t \in \{1, 2, 4, 8, 16\}$  [2, 3].

Пусть  $C_1 \subseteq C$  будет множеством ОЛЦ, где  $\alpha_g \in C_1$ , если её выход не соединён с конечной вершиной  $b_E$ . Пусть  $\Pi_C = \{B_1, \dots, B_l\}$  – разбиение множества  $C_1$  на классы ПОЛЦ. Закодируем ОЛЦ  $\alpha_g \in C_1$  таким образом, чтобы большинство классов  $B_i \in \Pi_C$  было представлено одним интервалом  $R_G$ -мерного булева пространства. Для этого может быть использован широко известный алгоритм ESPRESSO [5]. Пусть  $\Pi_C = \Pi_A \cup \Pi_B$ , где  $B_i \in \Pi_A$ , если классу соответствует один интервал, иначе  $B_i \in \Pi_B$ .

Если выполняется условие

$$\Pi_B = \emptyset, \quad (9)$$

то блок ПК отсутствует и регистр Рг является источником всех кодов  $K(B_i)$  для классов  $B_i \in \Pi_C$ . В противном случае преобразованию

подлежат только адреса выходов ОЛЦ, входящих в классы  $B_i \in \Pi_B$ . Для кодирования этих классов достаточно

$$R_B = \lceil \log_2(|\Pi_B| + 1) \rceil \quad (10)$$

Единица добавляется к  $|\Pi_B|$  для обозначения ситуации  $B_i \notin \Pi_B$ . Отметим, что часть кодов может быть реализована на ППЗУ. При использовании унитарного кодирования микроопераций [6], каждое слово в УП состоит из  $N+2$  бит. Число 2 добавляется к  $N$  для учёта дополнительных переменных  $y_0$  и  $y_E$  (Рис. 1).

Если каждое ППЗУ имеет  $t$  выходов и не менее  $M$  слов, то достаточно  $K_0$  микросхем для реализации УП, где

$$K_0 = \left\lceil \frac{N+2}{t} \right\rceil. \quad (11)$$

Очевидно, что остаётся  $R_0$  свободных выходов микросхем ППЗУ, где

$$R_0 = K_0 t - N - 2. \quad (12)$$

Если выполняется условие

$$R_0 \geq R_B, \quad (13)$$

то все классы  $B_i \in \Pi_B$  могут быть представлены с помощью УП и блок ПК отсутствует. В противном случае множество  $\Pi_B$  представляется в виде  $\Pi_E \cup \Pi_D$ . При этом

$$|\Pi_E| = 2^{R_0} - 1, \quad (14)$$

$$R_D = \lceil \log_2(|\Pi_D| + 1) \rceil. \quad (15)$$

Единица в (14) вычитается, чтобы отразить ситуацию  $B_i \notin \Pi_E$ . Единица прибавляется в (15), для отражения ситуации  $B_i \notin \Pi_D$ . Итак, преобразованию подлежат только адреса выходов ОЛЦ  $\alpha_g \in B_i$ , где  $B_i \in \Pi_D$ . Если все множества  $\Pi_A, \Pi_E$  и  $\Pi_D$ , не являются пустыми, то для интерпретации ЛГСА  $\Gamma$  предлагается КМУУ  $U_2$  (Рис. 2).

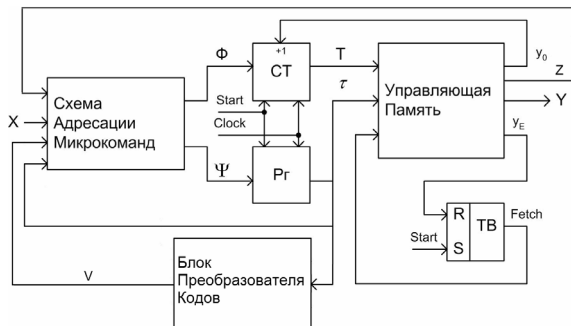


Рисунок 2 – Структурная схема КМУУ  $U_2$

В КМУУ  $U_2$ , блок САМ реализует функции:

$$\Phi = \Phi(\tau, V, Z, X), \quad (16)$$

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

а блок ПК реализует функцию

$$V = V(\tau).$$

Переменные  $v_r \in V$  используются для кодирования классов  $B_i \in \Pi_D$ ,  $|V| = R_D$ . Переменные  $z_r \in Z$  используются для кодирования классов  $B_i \in \Pi_E$ ,  $|Z| = R_0$ . В зависимости от переменных  $\tau$  и  $T$  управляющая память реализует функции  $Z, Y, y_0$  и  $y_E$ . Принципы действия КМУУ  $U_1$  и  $U_2$  практически идентичны.

В работе предлагается следующий алгоритм построения КМУУ  $U_2$ :

1. Построение множеств  $C, C_1$  и  $\Pi_C$  для ЛГСА  $\Gamma$ .
2. Кодирование ОЛЦ  $\alpha_g \in C$  и их компонентов.
3. Построение разбиений  $\Pi_A, \Pi_E$  и  $\Pi_D$ .
4. Кодирование классов  $B_i \in \Pi_E \cup \Pi_D$ .
5. Построение таблицы переходов КМУУ.
6. Спецификация управляющей памяти.
7. Спецификация блока ПК.
8. Реализация схемы КМУУ в данном элементном базисе.

### Пример применения предложенного метода

Пусть для некоторой ГСА  $\Gamma_1$  построены множества  $C = \{\alpha_1, \dots, \alpha_8\}$ ,  $C_1 = \{\alpha_1, \dots, \alpha_7\}$  и  $\Pi_C = \{B_1, B_2, B_3\}$ , где  $\alpha_1 = \langle b_1, \dots, b_4 \rangle$ ,  $\alpha_2 = \langle b_5, \dots, b_8 \rangle$ ,  $\alpha_3 = \langle b_9, \dots, b_{11} \rangle$ ,  $\alpha_4 = \langle b_{12}, \dots, b_{14} \rangle$ ,  $\alpha_5 = \langle b_{15}, \dots, b_{17} \rangle$ ,  $\alpha_6 = \langle b_{18}, \dots, b_{21} \rangle$ ,  $\alpha_7 = \langle b_{22}, \dots, b_{24} \rangle$ ,  $\alpha_8 = \langle b_{25}, \dots, b_{28} \rangle$ ,  $B_1 = \{\alpha_1\}$ ,  $B_2 = \{\alpha_2, \alpha_3, \alpha_4\}$ ,  $B_3 = \{\alpha_5, \alpha_6, \alpha_7\}$ . Итак  $G=8$ ,  $R_G=3$ ,  $Q=4$ ,  $R_Q=2$ ,  $M=28$ ,  $R=5$ . Таким образом, условие (6) выполняется и метод разделения кодов может быть применён. Условие (1) также выполняется, следовательно, ГСА  $\Gamma_1$  является линейной.

Закодируем ОЛЦ  $\alpha_g \in C$ , как показано на рис. 3, используя переменные  $\tau_r \in T$ .

Компоненты ОЛЦ  $\alpha_g \in C$  закодированы тривиально [4]: первый компонент имеет код 00, второй – 01 и так далее, для соответствия условию (5). Для кодирования компонентов используются переменные  $T_r \in T = \{T_1, T_2\}$ .

	$\tau_2\tau_3$			
$\tau_1$	00	01	11	10
0	$\alpha_1$	$\alpha_2$	$\alpha_3$	$\alpha_4$
1	$\alpha_5$	$\alpha_6$	$\alpha_7$	$\alpha_8$

Рисунок 3 – Коды ОЛЦ для ГСА  $\Gamma_1$ .

Из рис. 3 видно, что класс  $B_1$  представлен одним интервалом булева пространства и представлен кодом  $K(B_1)=000$ . Классу  $B_2$  соответствуют интервалы 001 и 01\*, а классу  $B_3$  интервалы 10\* и 111. Таким образом,  $\Pi_A = \{B_1\}$ ,  $\Pi_B = \{B_2, B_3\}$  и  $R_B = 2$ .

Пусть для ГСА  $\Gamma_1$   $N=13$  и пусть для реализации УП используются микросхемы ППЗУ с  $t = 4$ . В таком случае, из (11) имеем  $K_0 = 4$ , а из условия (12) –  $R_0 = 1$ . Условие (13) нарушается и в КМУУ  $U_2(\Gamma_1)$  используется блок ПК. Пусть  $\Pi_E = \{B_2\}$ , тогда  $\Pi_D = \{B_3\}$ . Итак, множества  $\Pi_A$ ,  $\Pi_E$ ,  $\Pi_D$  сформированы. Переменная  $z_1 \in Z$  используется для кодирования класса  $B_2$ . Пусть  $K(B_2)=1$ , где  $z_1=0$  показывает ситуацию  $B_i \notin \Pi_E$ . Пусть  $K(B_3)=1$ , где  $v_1=0$  показывает ситуацию  $B_i \notin \Pi_D$ .

Пусть переходы для классов  $B_i \in \Pi_C$  задаются следующей системой обобщённых формул перехода (ОФП)[4]:

$$B_1 \rightarrow x_1 b_5 \vee x_1 x_2 b_9 \vee x_1 x_2 x_3 b_{11} \vee x_1 x_2 x_3 b_{18};$$

$$B_2 \rightarrow \overline{x_4} b_{15} \vee x_4 \overline{x_5} b_{19} \vee x_4 x_5 \overline{x_6} b_{22} \vee x_4 x_5 x_6 b_{13}; \quad (18)$$

$$B_3 \rightarrow x_7 b_{11} \vee x_7 \overline{x_8} b_9 \vee x_7 x_8 b_{25}.$$

Эта система является основой для построения таблицы переходов, имеющей следующие столбцы:  $B_i$ ,  $K_A(B_i)$ ,  $K_E(B_i)$ ,  $K_D(B_i)$ ,  $b_q$ ,  $A(b_q)$ ,  $X_h$ ,  $\Psi_h$ ,  $\Phi_h$ ,  $h$ . Здесь  $K_A(B_i)$  – это код  $K(B_i)$ , где  $B_i \in \Pi_A$ ;  $K_E(B_i)$  – код для  $B_i \in \Pi_E$ ;  $K_D(B_i)$  – код для  $B_i \in \Pi_D$ ;  $X_h$  – входной сигнал получаемый из ОФП;  $\Psi_h$  – набор выходных функций загружающих код  $K(\alpha_g)$  в Рг;  $\Phi_h$  – набор выходных функций загружающих код  $K(b_q)$  в СТ;  $h=1, \dots, H_0$  номер перехода. В этом случае  $\Psi = \{D_1, D_2, D_3\}$ ,  $\Phi = \{D_4, D_5\}$ . Определим несколько адресов

микроопераций. Например, вершина  $b_{22}$  является первым компонентом ОЛЦ  $\alpha_7 \in C$ , следовательно,  $A(b_{22})=11000$ . По аналогии мы можем найти  $A(b_{12})=01100$ ,  $A(b_4)=00011$  и  $A(b_{19})=10101$ .

Пусть символ  $U_i(\Gamma_j)$  используется для обозначения интерпретации ГСА  $\Gamma_j$  с помощью модели КМУУ  $U_i(i=1,2)$ .

Часть таблицы переходов для классов  $B_1, \dots, B_3 \in \Pi_C$  показана в табл. 1.

Эта таблица используется для построения системы (16). Например, следующие выражения получены из табл. 1:

$$D_2 = \tau_1 \tau_2 \tau_3 z_1 x_1 x_2 \vee \tau_1 \tau_2 \tau_3 z_1 x_1 x_3 \vee z_1 v_1 x_4 x_5 \vee z_1 v_1 \quad (\text{после минимизации});$$

$$D_4 = z_1 v_1 x_7.$$

Таблица 1. Часть таблицы переходов для КМУУ  $U_2(\Gamma_1)$

$B_i$	$K_A(B_i)$	$K_E(B_i)$	$K_D(B_i)$	$b_q$	$A(b_q)$	$X_h$	$\Psi_h$	$\Phi_h$	$h$
	$\tau_1 \tau_2 \tau_3$	$z_1$	$v_1$						
$B_1$	000	0	0	$b_5$	001 00	$\overline{x_1}$	$D_3$	-	1
				$b_9$	010 00	$x_1 \overline{x_2}$	$D_2$	-	2
....									
$B_2$	***	1	0	$b_{15}$	100 00	$\overline{x_4}$	$D_1$	-	5
				....					
				$b_{13}$	011 01	$x_4 x_5 x_6$	$D_2 D_3$	$D_5$	8
$B_3$	***	0	1	$b_{11}$	010 10	$\overline{x_7}$	$D_2$	$D_4$	9
				....					

Управляющая память КМУУ  $U_2(\Gamma_1)$  специфицируется используя широко известные методы [4].

Таблица блока преобразователя адресов строится для классов  $B_i \in \Pi_D$ . В нашем примере она имеет 3 строки (Табл. 2).

Таблица 2. Таблица блока ПК устройства  $U_2(\Gamma_1)$

$\alpha_g$	$A(O_g)$	$B_i$	$K_D(B_i)$	$V_i$	$j$
$\alpha_{17}$	100 10	$B_3$	1	$v_1$	1
$\alpha_{21}$	101 11				2
$\alpha_{24}$	110 10				3

Из этой таблицы имеем систему (17) которая в нашем случае представляется следующей ДНФ:

$$v_1 = \tau_1 \tau_2 \vee \tau_1 \tau_3.$$

Отметим, что  $H_0 = 11$  что определено общим количеством термов в системе (18). В случае КМУУ  $U_1(\Gamma_1)$  имеем  $H = 22$ . Таким образом применение предложенного метода позволяет сократить это значение в 2 раза. Ожидается что уменьшение количества макроячеек ПМЛ в логической схеме блока САМ будет иметь тот же порядок [4].

### Заключение

Предложенный метод направлен на уменьшение числа макроячеек ПМЛ в схеме адресации микрокоманд КМУУ с разделением кодов. Это возможно благодаря таким главным факторам как большой коэффициент объединения по входу современных макроячеек ПМЛ, а также естественная избыточность микросхем ППЗУ, обусловленная тем, что количество их выходов ограничено определённым множеством чисел; существование классов псевдоэквивалентных ОЛЦ.

Научная новизна предложенного метода заключается в использовании особенностей

базиса ПЛИС, а именно большого коэффициента объединений по входу, для уменьшения аппаратных затрат в схеме блока ПК. Отметим, что при выполнении условия (13) этот блок вообще отсутствует.

Практическая значимость этого метода заключается в уменьшении числа микросхем при реализации схемы КМУУ, что позволяет получить схемы, обладающие меньшей стоимостью по сравнению с известными из литературы аналогами.

Исследования показали, что общее количество макроячеек уменьшается до 12% для КМУУ  $U_2(\Gamma_1)$  в сравнении с эквивалентным КМУУ  $U_1(\Gamma_1)$ . Стоит отметить, что данный метод применим лишь для интерпретации линейных ГСА при условии выполнения (6).

Дальнейшие направления наших исследований связаны с применением предложенного подхода для реализации КМУУ на микросхемах FPGA [5,10].

### Литература

1. Baranov S. Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 pp.
2. Электронный ресурс. Xilinx CPLDs  
[http://www.xilinx.com/products/silicon\\_solutions/cplds/index.htm](http://www.xilinx.com/products/silicon_solutions/cplds/index.htm).
3. Электронный ресурс. Altera devices overview.  
[http://www.altera.com/products/devices/common/devfamily\\_overview.html](http://www.altera.com/products/devices/common/devfamily_overview.html).
4. Barkalov A., Titarenko L. Logic Synthesis for Compositional Microprogram Control Units. – Berlin: Springer, 2008. – 272 pp.
5. Maxfield C. The Design Warrior's Guide to FPGAs. – Amsterdam: Elsevier, 2004. – 541 pp.
6. Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.
7. Баркалов А.А., Зеленёва И.Я., Лаврик А.С. Использование особенностей ПЛИС для оптимизации схемы устройства управления. / Наукові праці Донецького національного технічного університету. Серія «Інформатика, кібернетика і обчислювальна техніка» (ІКОТ-2008). Випуск 9 (132) – Донецьк: ДонНТУ. – 2008. С. 178-182.
8. Баркалов А.А., Ковалёв С.А., Красичков А.А., Лаврик А.С. Оптимизация устройства управления с преобразователем адреса микрокоманд. /Материалы Девятого Международного научно-практического семинара. В 3-х кн. – Таганрог. Кн. 3. 2008. С. 12-20.
9. Электронный ресурс. CoolRunner CPLD Datasheet.  
<http://www.xilinx.com/support/documentation/coolrunner-ii.htm>
10. Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем с использованием микросхем программируемой логики. – СПб: БХВ. – Петербург, 2002. – 608 с.

Поступила в редакцию 30.03.2010