

УМЕНЬШЕНИЕ ЧИСЛА LUT-ЭЛЕМЕНТОВ В УСТРОЙСТВЕ УПРАВЛЕНИЯ С ОБЩЕЙ ПАМЯТЬЮ

БАРКАЛОВ А.А., ТИТАРЕНКО Л.А.,
МИРОШКИН А.Н.

Предлагается метод уменьшения числа LUT-элементов в схеме композиционного микропрограммного устройства управления. Метод основан на кодировании входов операторных линейных цепей, позволяющем однозначно представлять коды классов псевдоэквивалентных цепей. Коды классов преобразовываются в адреса микрокоманд и некоторые микрооперации. Показываются условия применения предложенного метода. Рассматривается пример его применения.

1. Введение

Одной из важных проблем, возникающих при реализации схем устройств управления (УУ) в базе FPGA (Field Programmable Gate Array), является уменьшение числа LUT (Look-Up Table) элементов в них [1]. Для решения этой задачи необходимо учитывать особенности как интерпретируемого алгоритма управления, так и используемых для этого микросхем РИ, 2009, № 3

[2]. Если алгоритм управления представлен линейной граф-схемой, а микросхемы FPGA включают встроенные блоки памяти EMB (Embedded Memory Blocks), то для интерпретации может быть использована модель композиционного микропрограммного устройства управления (КМУУ) с общей памятью [3]. В настоящей работе предлагается метод уменьшений числа LUT элементов в схеме КМУУ, основанный на преобразовании кодов входов операторных линейных цепей (ОЛЦ) в адреса микрокоманд. *Основная идея* метода заключается в представлении классов псевдоэквивалентных ОЛЦ обобщенными интервалами кодирующего пространства. *Научная новизна* метода состоит в использовании особенностей EMB современных микросхем FPGA (конфигурируемость в пределах заданной емкости) для уменьшения аппаратных затрат в схеме КМУУ. Алгоритм управления представляется в виде граф-схемы алгоритма (ГСА).

2. Основные определения и общие положения

Пусть алгоритм управления цифровой системы представлен в виде ГСА Γ , которая характеризуется множеством вершин $B = E_1 \cup E_2 \cup \{b_0, b_E\}$ и дуг E , соединяющих эти вершины. Здесь E_1 – множество операторных вершин, содержащих наборы микро-

операций из множества микроопераций $Y = \{y_1, \dots, y_N\}$; E_2 – множество условных вершин, содержащих элементы множества логических условий (ЛЮ) $X = \{x_1, \dots, x_L\}$; b_0 – начальная вершина; b_E – конечная вершина ГСА Γ . Введем ряд определений [3].

Определение 1. Операторной линейной цепью ГСА Γ называется конечная последовательность операторных вершин $\alpha_g = \langle b_{g1}, \dots, b_{gF_g} \rangle$, такая, что для любой пары ее соседних вершин существует дуга $\langle b_{gi}, b_{gi+1} \rangle \in E$, где $i = 1, \dots, F_g - 1$.

Определение 2. Операторная 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 $b_t \notin D^g$.

Определение 3. Операторная 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. Операторные линейные цепи называются псевдоэквивалентными ОЛЦ (ПОЛЦ), если их выходы связаны с входом одной и той же вершины ГСА Γ .

Любая ОЛЦ α_g имеет произвольное число входов, обозначаемых I_g^k ($k = 1, \dots, F_g$) и образующих множество I_g , и один выход, обозначаемый символом O_g .

Определение 5. Граф-схема алгоритма Γ является линейной ГСА, если число ее операторных вершин не менее, чем в два раза превосходит число ОЛЦ.

Пусть для ГСА Γ получено множество ОЛЦ $C = \{\alpha_1, \dots, \alpha_G\}$, где каждая ОЛЦ $\alpha_g \in C$ включает максимально возможное число компонент [3]. Пусть $I(\Gamma)$, $O(\Gamma)$ – соответственно множество входов и выходов ОЛЦ ГСА Γ . Каждая вершина $b_q \in E_1$ соответствует микрокоманде MI_q , имеющей адрес $A(b_q)$. Выполним адресацию микрокоманд так, чтобы выполнялось условие

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

где $g \in \{1, \dots, G\}$, $i \in \{1, \dots, F_g - 1\}$. В этом случае для интерпретации линейной ГСА Γ можно использовать модель КМУУ с общей памятью (рис. 1), обозначаемую в дальнейшем символом U_1 . Эта модель включает блок адресации микрокоманд (БАМ), счетчик (СТ), блок микроопераций (БМО) и триггер выборки (ТВ). Устройство U_1 функционирует следующим образом.

По сигналу *Start* в СТ записывается нулевой адрес, соответствующий началу микропрограммы, интерпретирующей ГСА Γ , одновременно триггер ТВ устанавливается в единичное состояние ($Fetch = 1$) и микрокоманды могут выбираться из блока памяти БМО.

Если СТ содержит адрес $A(b_q)$ и $b_q \notin O(\Gamma)$, то одновременно с набором микроопераций $Y(b_q)$, записанных в вершине $b_q \in E_1$, БМО формирует сигнал y_0 . Если $y_0 = 1$, то содержимое СТ увеличивается на единицу по сигналу *Clock*. При этом выполняется безусловный переход, соответствующий равенству (1). В том случае, если $b_q \in O(\Gamma)$, то сигнал y_0 не формируется, а блок БМО вырабатывает функции возбуждения СТ:

$$\Phi = \Phi(T, X). \quad (2)$$

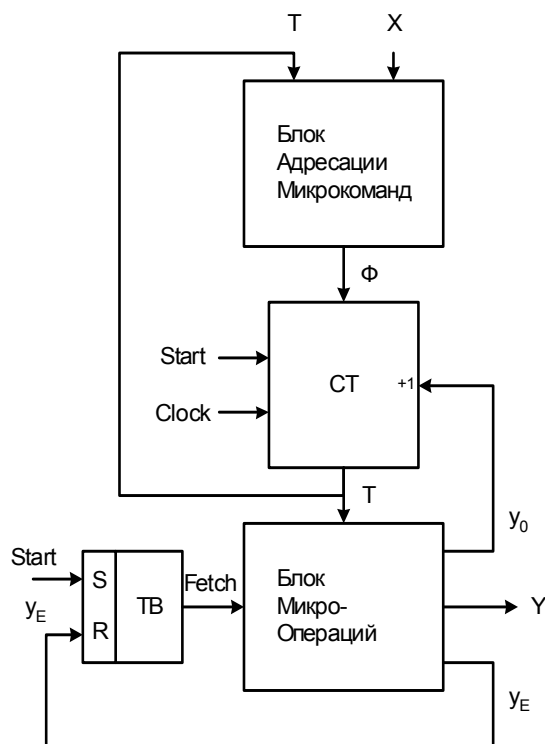


Рис. 1. Структурная схема КМУУ U_1

В этом случае по сигналу *Clock* в СТ формируется адрес перехода из выхода некоторой ОЛЦ $\alpha_g \in C$. Если $\langle b_q, b_E \rangle \in E$, то блок БМО формирует сигнал y_E , вызывающий установку триггера ТВ в нулевое состояние. При этом $Fetch = 0$, выборка микрокоманд прекращается и КМУУ U_1 прекращает функционирование.

При реализации схем КМУУ U_1 на FPGA схемы БАМ, СТ и ТВ строятся на LUT элементах, а блоки БМО используются для реализации БМО. Недостатком модели U_1 является значительное число переменных обратной связи, определяемое как

$$R_A = \lceil \log_2 M \rceil, \quad (3)$$

где $M = |E_1|$. Для уменьшения числа LUT элементов в схеме БАМ необходимо уменьшить число его входов и выходов. Один из методов решения этой задачи предлагается в данной статье.

3. Основная идея предлагаемого метода

Поставим в соответствие каждому входу $I_g^j \in I(\Gamma)$ двоичный код $K(I_g^j)$ разрядности

$$R_I = \lceil \log_2 M_I \rceil, \quad (4)$$

где $M_I = |I(\Gamma)|$. Используем для кодирования входов переменные $v_r \in V$, где $|V| = R_I$. Пусть ОЛЦ $\alpha_g \in C'$, если $\langle b_q, b_E \rangle \notin E$, где $b_q = O_g$ и $C' \subseteq C$. Найдем разбиение $\Pi_C = \{B_1, \dots, B_I\}$ множества ОЛЦ $\alpha_g \in C'$ на классы псевдоэквивалентных ОЛЦ. Пусть для кодирования классов $B_i \in \Pi_C$ достаточно

$$R_C = \lceil \log_2 I \rceil \quad (5)$$

переменных.

Закодируем входы $I_g^j \in I(\Gamma)$ так, чтобы каждый класс $B_i \in \Pi_C$ представлялся одним обобщенным интервалом R_I -мерного булева пространства. Пусть при этом только R_O переменных $v_r \in V$ являются значимыми для всех интервалов, которые можно рассматривать как коды $K(B_i)$ классов $B_i \in \Pi_C$. Очевидно, в пределе выполняется условие

$$R_O = R_C. \quad (6)$$

Современные микросхемы FPGA [5,6] включают блоки ЕМВ, конфигурация которых (число слов N_W и число выходов N_O) может меняться при сохранении постоянной емкости:

$$V_{EMB} = N_W \cdot N_O. \quad (7)$$

У типичных представителей семейства FPGA $V_{EMB} = 4K$ бит при конфигурациях $4K \times 1$, $2K \times 2$, $1K \times 4$, 512×8 , т.е. $N_O \in \{1, 2, 4, 8\}$. Для реализации блока БМО необходимы ЕМВ с $N_W = 2^{R_A}$, при этом число выходов t_F определяется как

$$t_F = \frac{V_{EMB}}{2^{R_A}}. \quad (8)$$

Таким образом, для реализации схемы блока БМО необходимо

$$n_1 = \left\lceil \frac{N+2}{t_F} \right\rceil \quad (9)$$

блоков ЕМВ.

Для преобразования кодов входов ОЛЦ в адреса микрокоманд необходимы блоки ЕМВ, имеющие $N_W = 2^{R_I}$ слов и

$$t_I = \frac{V_{EMB}}{2^{R_I}}. \quad (10)$$

При этом

$$\Delta t = t_I - R_A \quad (11)$$

выходов остаются свободными.

Разобьем множество микроопераций Y на классы Y^1 и Y^2 . При этом $y_n \in Y^1$, если эта микрооперация записана только в вершинах $b_q \in I(\Gamma)$. Пусть выполняются следующие условия:

$$\left\lceil \frac{N - N_1 + 2}{t_F} \right\rceil < n_1, \quad (12)$$

$$\Delta t \geq N_1, \quad (13)$$

$$|Y^1| \geq N_1. \quad (14)$$

В этом случае N_1 микроопераций $y_n \in Y^1$ могут быть реализованы на преобразователе кодов входов, что уменьшит число блоков ЕМВ, необходимых для реализации схемы БМО.

Если условия (12)-(14) выполняются, то для интерпретации линейной ГСА Γ предлагается КМУУ U_2 (рис. 2). По сравнению с КМУУ U_1 в состав U_2 входит блок преобразователя кодов (БПК) и регистр кодов блоков (РКБ).

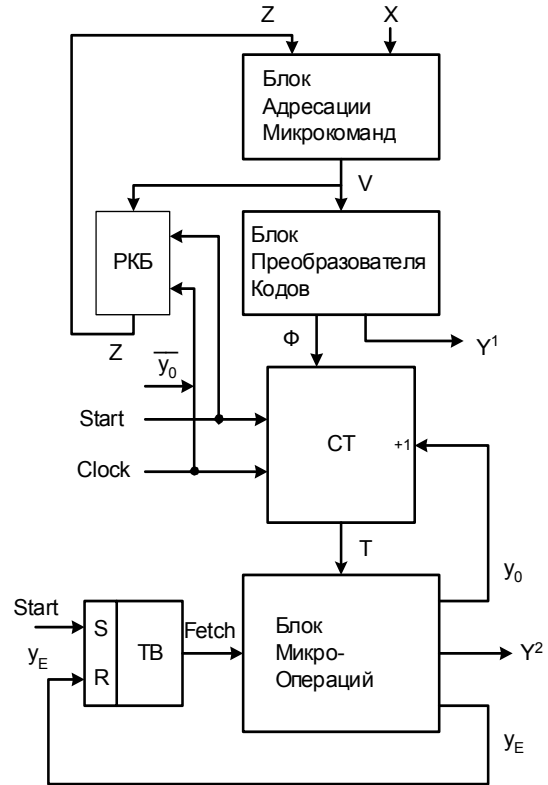


Рис. 2. Структурная схема КМУУ U_2

Блок БПК реализует системы функций

$$\Phi = \Phi(V), \quad (15)$$

$$Y^1 = Y^1(V), \quad (16)$$

регистр РКБ хранит часть кода $K(I_g^j)$, представляющую собой код $K(B_i)$. Содержимое РКБ меняется при $y_0 = 0$. Блок БАМ реализует систему функций

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

имеющих меньшее число аргументов, чем функции (2). Принцип функционирования КМУУ U_2 очевиден из предыдущего материала.

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

1. Формирование множеств C , C' и P_C для ГСА Γ .
2. Естественная адресация микрокоманд.
3. Кодирование входов ОЛЦ $\alpha_g \in C$.
4. Разбиение множества микроопераций на классы Y^1 и Y^2 .
5. Спецификация блока БАМ.
6. Спецификация блока БПК.
7. Спецификация блока БМО.
8. Реализация схемы КМУУ в заданном элементном базисе.

Рассмотрим применение этого метода на примере реализации КМУУ $U_2(\Gamma_1)$, где ГСА Γ_1 представлена на рис. 3. Здесь и далее символ $U_i(\Gamma_j)$ означает, что модель КМУУ U_i используется для интерпретации ГСА Γ_j .

4. Пример применения предложенного метода

Применение методов из [3] позволяет найти множество ОЛЦ $C = \{\alpha_1, \dots, \alpha_6\}$,

где $\alpha_1 = \langle b_1, b_2, b_3 \rangle$, $\alpha_2 = \langle b_4, \dots, b_8 \rangle$,

$\alpha_3 = \langle b_9, \dots, b_{11} \rangle$, $\alpha_4 = \langle b_{12}, \dots, b_{14} \rangle$,

$\alpha_5 = \langle b_{15}, b_{16} \rangle$, $\alpha_6 = \langle b_{17}, b_{18} \rangle$,

при этом $\alpha_6 \notin C'$. Анализ ГСА Γ_1 позволяет найти элементы множества входов ОЛЦ $I(\Gamma_1)$: $I_1^1 = b_1$, $I_1^2 = b_2$, $I_2^1 = b_4$, $I_2^2 = b_6$, $I_3^1 = b_9$, $I_4^1 = b_{12}$, $I_5^1 = b_{15}$, $I_6^1 = b_{17}$. Аналогично можно найти элементы множества выходов ОЛЦ $O(\Gamma_1)$: $O_1 = b_3$, $O_2 = b_8$, $O_3 = b_{11}$, $O_4 = b_{14}$, $O_5 = b_{16}$ и $O_6 = b_{18}$. Итак, для ГСА Γ_1 количество ОЛЦ $G = 6$,

количество операторных вершин $M = 18$, т.е. $\frac{M}{G} = 3$

и ГСА Γ_1 является линейной. Таким образом, применение модели КМУУ для интерпретации ГСА Γ_1 целесообразно.

Используя определение 4, можно найти множество ПОЛЦ $P_C = \{B_1, B_2\}$, где $B_1 = \{\alpha_1, \alpha_2\}$, $B_2 = \{\alpha_3, \alpha_4, \alpha_5\}$. Адресация микрокоманд согласно (1) выполняется тривиальным образом [3]. В нашем примере для адресации $M = 18$ вершин достаточно кода разрядности $R_A = 5$; $A(b_0) = 00000$, $A(b_1) = 00001$, ..., $A(b_{18}) = 10010$.

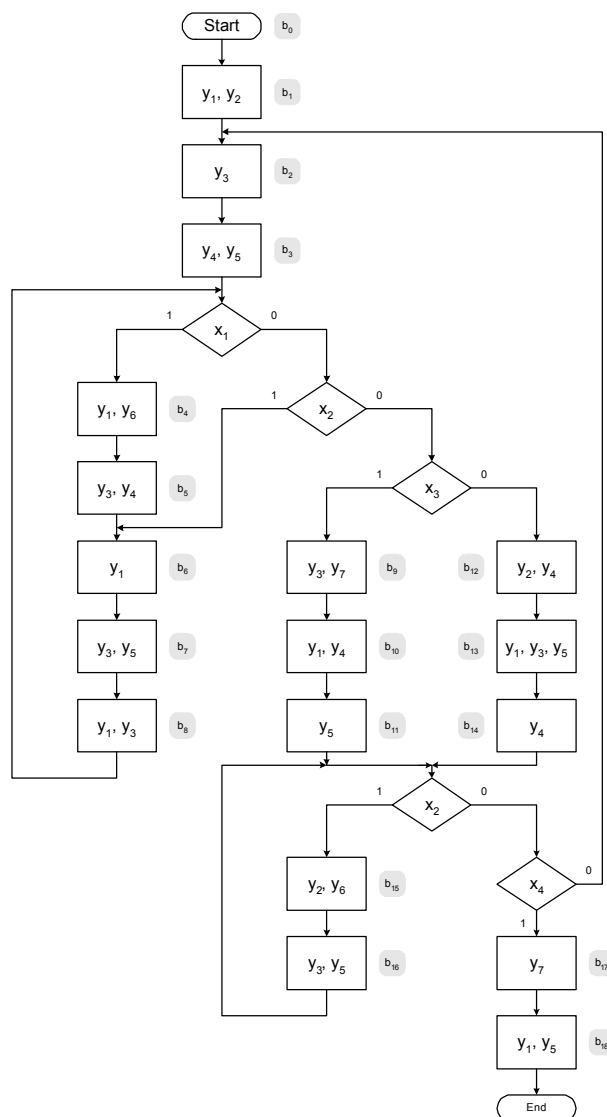


Рис. 3. Исходная ГСА Γ_1

Множество входов ОЛЦ $I(\Gamma_1)$ содержит $M_I = 8$ элементов, для кодирования которых достаточно $R_I = 3$ переменных $v_I \in V$. Так как число классов ПОЛЦ $I = 2$, то достаточно кода разрядности $R_C = 1$. Следовательно, в оптимальном случае входы ОЛЦ $\alpha_g \in C'$ должны быть закодированы так, чтобы количество значимых переменных для кодирования обобщенных интервалов булевого пространства было равным $R_O = 1$. Для этой цели можно использовать, например, алгоритм ESPRESSO [1]. Один из вариантов оптимального кодирования приведен в карте Карно (рис. 4).

		$v_1 v_2$			
v_3	0	00	01	11	10
	1	00	01	11	10
		I_1^1	I_2^1	I_3^1	I_5^1
		I_1^2	I_2^2	I_4^1	I_6^1

Рис. 4. Коды входов ОЛЦ КМУУ $U_2(\Gamma_1)$

Поскольку переходы из выходов ОЛЦ $\alpha_g \notin C'$ не используются для формирования функций (17), то набор 101 в этой карте рассматривается как несущественный. Из рис. 4 следует, что $K(B_1) = 0**$ и $K(B_2) = 1**$. Следовательно, минимум разрядности для $K(B_i)$ достигнут и $Z = \{z_1\}$.

Пусть для реализации блоков БПК и БМО используются ЕМВ, имеющие следующие конфигурации: 64×1 , 32×2 , 16×4 , 8×8 . Так как разрядность кода, необходимого для кодирования операторных вершин ГСА, $R_A = 5$, то для реализации БМО необходимо выбрать конфигурацию 32×2 с количеством выходов $t_F = 2$. При этом для реализации БМО требуется

$$n_1 = \left\lceil \frac{7+2}{2} \right\rceil = 5 \text{ блоков ЕМВ. Для реализации БПК}$$

необходимо выбрать конфигурацию 8×8 , так как разрядность кода входов ОЛЦ $R_I = 3$. Следовательно, количество выходов БПК $t_I = 8$ и $\Delta t = 3$ выходов остаются свободными, т.е. блок БПК можно использовать для реализации трех микроопераций из множества Y^1 . При этом для реализации БМО будет достаточно 3 блока ЕМВ.

Анализ ГСА Γ_1 показывает, что множество микроопераций, которые встречаются только во входах ОЛЦ, формируют класс $Y^1 = \{y_2, y_6, y_7\}$, остальные микрооперации включены в класс $Y^2 = \{y_1, y_3, y_4, y_5\}$. Итак, условия (12)-(14) выполняются и разбиение множества Y уменьшает число блоков ЕМВ в схеме КМУУ. При этом общее число блоков ЕМВ в схемах БМО и БПК составляет 4, а для КМУУ $U_1(\Gamma_1)$ требуется $n_1 = 5$ таких блоков.

Для спецификации блока БАМ необходимо построить таблицу переходов КМУУ, которая формируется на основе системы обобщенных формул перехода [2]. Для ГСА Γ_1 эта система имеет следующий вид:

$$\begin{aligned} B_1 &\rightarrow x_1 I_2^1 \vee \overline{x_1} x_2 I_2^2 \vee \overline{x_1} x_2 x_3 I_3^1 \vee \overline{x_1} x_2 x_3 I_4^1; \\ B_2 &\rightarrow x_2 I_5^1 \vee \overline{x_2} x_4 I_6^1 \vee \overline{x_2} x_4 I_1^2. \end{aligned} \quad (18)$$

Таблица переходов включает $H_2(\Gamma_j)$ строк, этот параметр равен числу термов в системе вида (18). Таблица содержит столбцы B_i , $K(B_i)$, I_g^j , $K(I_g^j)$, X_h , V_h , h , где X_h – конъюнкция логических условий, входящая в термы системы вида (18), V_h – набор переменных $v_f \in V$, принимающих единичное значение в коде $K(I_g^j)$ для h -й строки таблицы. Для КМУУ $U_2(\Gamma_1)$ эта таблица имеет $H_2(\Gamma_1) = 7$ строк (табл. 1).

Связь табл. 1 с системой (18) и кодами из рис. 4 очевидна. Из таблицы переходов строятся функции:

$$\begin{aligned} v_1 &= \overline{z_1} x_1 x_2 \vee z_1 x_2 \vee z_1 \overline{x_2} x_4; \\ v_2 &= \overline{z_1}; \\ v_3 &= \overline{z_1} x_1 x_2 \vee \overline{z_1} x_1 x_2 x_3 \vee z_1 x_4. \end{aligned} \quad (19)$$

Таблица 1

Таблица переходов КМУУ $U_2(\Gamma_1)$

B_i	$K(B_i)$	I_g^j	$K(I_g^j)$	X_h	V_h	h
B_1	$0**$	I_2^1	010	x_1	v_2	1
		I_2^2	011	$\overline{x_1} x_2$	$v_2 v_3$	2
		I_3^1	110	$\overline{x_1} x_2 x_3$	$v_1 v_2$	3
		I_4^1	111	$\overline{x_1} x_2 x_3$	$v_1 v_2 v_3$	4
B_2	$1**$	I_5^1	100	x_2	v_1	5
		I_6^1	101	$\overline{x_2} x_4$	$v_1 v_3$	6
		I_1^2	001	$\overline{x_2} x_4$	v_3	7

Для спецификации блока БПК строится таблица со столбцами I_g^j , $K(I_g^j)$, $A(I_g^j)$, Φ_h , Y_h^1 , h , где $h = 1, \dots, M_O$. В столбце Φ_h записываются функции $D_r \in \Phi$, равные единице в адресе входа I_g^j , в столбце Y_h^1 – микрооперации $y_n \in Y^1$, равные единице в вершине $b_q \in E_1$, соответствующей входу I_g^j из h -й строки. Для КМУУ $U_2(\Gamma_1)$ блок БПК задан в табл. 2.

Отметим, что СТ и РКБ имеют информационные входы типа D, что соответствует современной тенденции реализации устройств управления на FPGA [7].

Таблица 2

Спецификация блока БПК КМУУ $U_2(\Gamma_1)$

I_g^j	$K(I_g^j)$	$A(I_g^j)$	Φ_h	Y_h^1	h
I_1^1	000	00000	-	y_2	1
I_1^2	001	00001	D_5	-	2
I_2^1	010	00011	D_4, D_5	y_6	3
I_2^2	011	00101	D_3, D_5	-	4
I_5^1	100	01110	D_2, D_3, D_4	y_2, y_6	5
I_6^1	101	10000	D_1	y_7	6
I_3^1	110	01000	D_2	y_7	7
I_4^1	111	01011	D_2, D_4, D_5	y_2	8

Спецификация блока БМО выполняется тривиальным образом [3] и этот этап не рассматривается. Реализация схемы КМУУ сводится к реализации системы (17) на LUT элементах и блоков БПК и БМО – на блоках ЕМВ. Этот этап выполняется с использованием стандартных промышленных пакетов [5, 6] и в данной статье не рассматривается.

5. Заключение

Предлагаемый метод основан на преобразовании кодов входов ОЛЦ в адреса микрокоманд и коды классов псевдоэквивалентных ОЛЦ. Если множество микроопераций разбивается на непустые множества Y^1 и Y^2 , то при выполнении некоторых условий метод позволяет уменьшить как число LUT элементов, так и блоков ЕМВ по сравнению с КМУУ с общей памятью. Уменьшение числа LUT элементов позволяет уменьшить число уровней в блоке адресации микрокоманд.

Проведенные исследования показали, что для линейных ГСА предложенный метод позволяет до 30% уменьшить число LUT элементов и число уровней уменьшается на 2-3. При этом в большинстве случаев число ЕМВ уменьшилось на один.

Практическая значимость предложенного метода заключается в уменьшении аппаратных затрат по сравнению с известными методами реализации КМУУ с общей памятью.

Дальнейшие направления исследований связаны с разработкой эффективного метода кодирования входов ОЛЦ, позволяющего выполнить условие (6). Применение для этой цели программы ESPRESSO [1] не всегда дает оптимальные результаты. Напомним, что предлагаемый метод применим только для интерпретации линейных ГСА.

Литература: 1. *De Micheli G.* Synthesis and Optimization of Digital Circuits. NY: McGraw-Hill, 1994. 636 p. 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 Compositional Microprogram Control Units – Berlin: Springer, 2008. 272 p. 4. *Baranov S.* Logic and System Design of Digital Systems. Tallinn: TUT Press, 2008. 266 p. 5. Embedded Memory in Altera FPGAs // <http://www.altera.com/technology/memory/embedded/mem-embedded.html>. 6. Embedded Memory // http://www.xilinx.com/products/design_resources/mem_corner/resource/xawembedded.htm. 7. *Соловьев В.В., Климович А.* Логическое проектирование цифровых систем на программируемых логических интегральных схемах. М.: Горячая линия-Телеком, 2008. 376 с.

Поступила в редколлегию 09.09.2009

Рецензент: д-р техн. наук, проф. Скобцов Ю.А.

Баркалов Александр Александрович, д-р техн. наук, профессор кафедры ЭВМ ДонНТУ, профессор Университета Зеленогурского (Польша). Научные интересы: цифровые устройства управления. Увлечения: научная работа, спорт. Адрес: Украина, 83122, Донецк, ул. Артема, 204А, кв.105. (+38062)301-07-35.

Титаренко Лариса Александровна, д-р техн. наук, профессор кафедры ТКС ХНУРЭ, профессор Университета Зеленогурского (Польша). научные интересы: системы телекоммуникаций, цифровые устройства управления. Увлечения: научная работа, спорт. Адрес: Украина, 62418, Харьковская область, пос. Песочин, ул. Зеленая, 14.

Мирошкин Александр Николаевич, ассистент кафедры ЭВМ ДонНТУ. Научные интересы: цифровые устройства управления. Увлечения: научная работа, спорт. Адрес: Украина, 86120, Донецкая область, Макеевка, ул. Курская, д.15, кв.45.