

УДК 004.274

Баркалов А.А., д.т.н.,
Визор Я.Е., к.т.н.,
Матвиенко А.В.

УСОВЕРШЕНСТВОВАНИЕ КОМПОЗИЦИОННОГО УСТРОЙСТВА УПРАВЛЕНИЯ С ОПТИМАЛЬНОЙ АДРЕСАЦИЕЙ МИКРОКОМАНД

Зеленогурский университет, Республика Польша
Институт кибернетики им. В.М. Глушкова НАН Украины

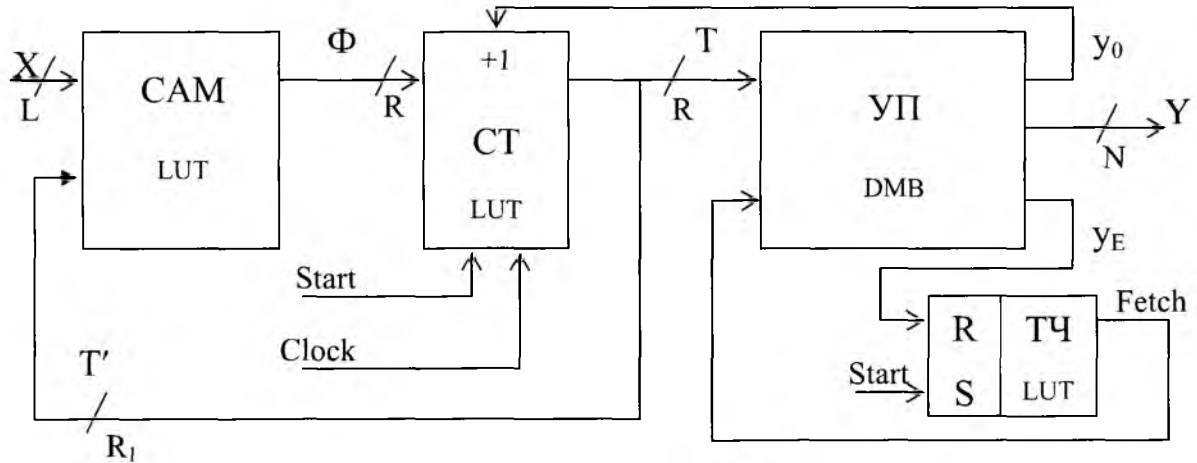
Предложен метод, позволяющий упростить схему адресации композиционного микропрограммного устройства управления с оптимальной адресацией микрокоманд, реализованную на FPGA. Метод основан на использовании процедуры элементаризации операторных линейных цепей с разделением кодов и оптимальным кодированием операторных линейных цепей. Применение данного метода обеспечивает уменьшение числа LUT-элементов в FPGA микросхеме. Приведен пример применения предложенного метода.

Введение

Развитие компьютерной техники неуклонно ведет к ее внедрению практически во все сферы деятельности человека, что в свою очередь, предъявляет все более высокие требования к характеристикам проектируемых устройств. При этом основное внимание уделяется как увеличению быстродействия, так и снижению аппаратных затрат, что в конечном итоге влияет на снижение себестоимости изделий цифровой техники. Это относится как к универсальным, так и к специализированным вычислительным системам, при реализации которых широко используются ПЛИС типа «система-на-кристалле» (*SoPC* – system-on-a-programmable-chip) [1-4]. В состав *SoPC* входят средства для реализации произвольной логики (*FPGA* или *CPLD*), средства для реализации памяти, встроенные микропроцессоры, встроенные средства реконфигурации. Производительность таких ПЛИС определяется эффективностью взаимодействия всех встроенных компонентов [5].

Важнейшей составной частью цифровой системы является устройство

управления [1,5], которое может быть реализовано как композиционное микропрограммное устройство управления (КМУУ) [6]. При реализации КМУУ в составе *SoPC* схема адресации микрокоманд строится на *FPGA* (field-programmable gate array) – программируемых пользователем матрицах вентилях, состоящих из миллионов элементов табличного типа (*LUT*-элементов) [6,7], а система микроопераций реализуются на встроенных блоках памяти *DMB* (dedicated memory block). Ограниченное (до 6) число входов *LUT*-элементов приводит к необходимости декомпозиции реализуемых булевых функций [7], что увеличивает число *LUT*-элементов (и их уровней) в схеме адресации КМУУ. В связи с этим актуальной остается задача разработки новых и совершенствования известных методов синтеза КМУУ. В данной работе предлагается усовершенствование метода синтеза КМУУ с оптимальной адресацией микрокоманд [9], основанное на использовании процедуры элементаризации операторных линейных цепей (ОЛЦ) с разделением кодов.

Рис. 1. Структурная схема КМУУ U_1

Постановка задачи

Пусть алгоритм управления цифровой системы задан в виде граф-схемы алгоритма (ГСА) Γ [6], содержащей начальную b_0 , конечную b_E , операторные и условные вершины. Операторные вершины образуют множество V_1 , имеющее M элементов. В вершинах $b_q \in V_1$ записываются микрокоманды $Yq \subseteq Y$, где $Y = \{y_1, \dots, y_N\}$ – множество микроопераций. В условных вершинах, образующих множество V_2 , записываются элементы множества логических условий $X = \{x_1, \dots, x_L\}$.

Вершины ГСА образуют множество $V = V_1 \cup V_2 \cup \{b_0, b_E\}$, элементы которого связаны дугами из множества E .

Введем ряд определений [7], необходимых для дальнейшего изложения материала.

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

Определение 2. Вершина $b_q \in D^g$, где $D^g \subseteq V_1$ – 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. 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. Вход ОЛЦ называется главным входом, если отсутствует связь этого входа с выходами операторных вершин.

Определение 5. Операторные линейные цепи $\alpha_i, \alpha_j \in C$ называются псевдоэквивалентными ОЛЦ, если их выходы связаны с входом одной и той же вершины ГСА Γ , и образуют множество классов псевдоэквивалентных ОЛЦ $\Pi_C = \{V_1, \dots, V_I\}$. $I = |\Pi_C|$.

Пусть для ГСА Γ найдено разбиение $C = \{\alpha_1, \dots, \alpha_G\}$ множества V_1 на операторные линейные цепи и пусть для каждой пары соседних вершин ОЛЦ $\alpha_g \in C$ выполняется условие

$$A(b_{gi+1}) = A(b_{gi}) + 1 \quad (i=1, \dots, F_g - 1), \quad (1)$$

где $A(b_g)$ – адрес микрокоманды, соответствующей вершине $b_g \in V_1$. В этом случае устройство управления цифровой системы может быть реализовано в виде КМУУ U_1 с оптимальной адресацией микрокоманд (рис. 1) [9], синтез которого основан на наличии в ГСА псевдоэквивалентных ОЛЦ.

По сигналу *Start* в счетчик *CT* заносится адрес первой микрокоманды интерпретируемого алгоритма, а триггер чтения *ТЧ* устанавливается в единицу (*Fetch*=1), что разрешает выборку микрокоманд Y из управляющей памяти *УП*. Управляющая память *УП* хранит наборы микроопераций Y_q , где $Y_q \subseteq Y = \{y_1, \dots, y_N\}$ – микрооперации, записанные в вершине $b_q \in V_1$ ГСА Γ , и состоит из $2^R \cdot (N+2)$ бит. Первый

дополнительный разряд используется для хранения сигнала y_0 , обеспечивающего естественную адресацию компонент ОЛЦ $\alpha_g \in C$. Второй – для организации режима останова КМУУ (сигнал y_E). Комбинационная схема адресации микрокоманд (САМ) благодаря оптимальной адресации микрокоманд, имеет $R_1 = \lceil \log_2 I \rceil \leq R$ сигналов обратной связи и реализует систему функций

$$\Phi = \Phi(T', X), \quad (2)$$

формирующих в счетчике СТ адрес $A(I_g^j)$ j -го входа ОЛЦ $\alpha_g \in C$.

Оптимальная адресация микрокоманд выполняется с помощью модифицированной карты Карно. Модификация карты заключается в том, что по вертикали записываются двоичные наборы, следующие в естественном порядке.

Карта Карно, содержащая адреса микрокоманд, предоставляет $\Delta = 2^R - M_1$ клеток для адресации микрокоманд ОЛЦ $\alpha_g \in C'$, здесь $C' \subseteq C$ – множество ОЛЦ, выходы которых не связаны с входом вершины b_E ; M_1 – число компонент в ОЛЦ $\alpha_g \notin C'$. При выполнении условия

$$\Delta \geq \sum_{i=1}^I \Delta_i \quad (3)$$

адреса компонент любой ОЛЦ $\alpha_g \in C'$ могут быть расположены в соседних клетках карты Карно, причем компоненты всех ОЛЦ, принадлежащих i -му классу псевдоэквивалентных ОЛЦ B_i , будут расположены в одном кубе размерности Δ_i .

Предложенный метод оптимальной адресации микрокоманд [9] позволяет уменьшить число сигналов обратной связи в схеме адресации микрокоманд за счет уменьшения числа аргументов и числа термов в системе функций (2). Однако его применение не всегда возможно. Для целесообразности использования метода

необходимо выполнение ряда условий [9].

В настоящей работе предлагается для усовершенствования метода синтеза КМУУ с оптимальной адресацией микрокоманд использовать процедуру элементаризации ОЛЦ с разделением кодов. При этом элементарной ОЛЦ будем считать ОЛЦ, имеющую только один вход [6].

Основная идея метода

Структура КМУУ U_2 , интерпретирующего элементарные ОЛЦ с разделением кодов, представлена на рис. 2.

Здесь комбинационная схема САМ формирует систему функций

$$\Phi = \Phi(T', X), \quad (4)$$

задающих в регистре РК код $K(\alpha_g)$ текущей ОЛЦ $\alpha_g \in C$. Адрес микрокоманды представлен в виде конкатенации кодов [6].

$$A(Y_q) = K(\alpha_g) * K(b_q),$$

где $A(Y_q)$ – адрес микрокоманды Y_q , записанной в вершине b_q , входящей в ОЛЦ α_g и хранящейся в управляющей памяти УП;

$K(\alpha_g)$ – код ОЛЦ $\alpha_g \in C = \{\alpha_1, \dots, \alpha_G\}$ разрядности $R_{11} = \lceil \log_2 G \rceil$;

$K(b_q)$ – код вершины b_q кортежа $\alpha_g \in C$ разрядности

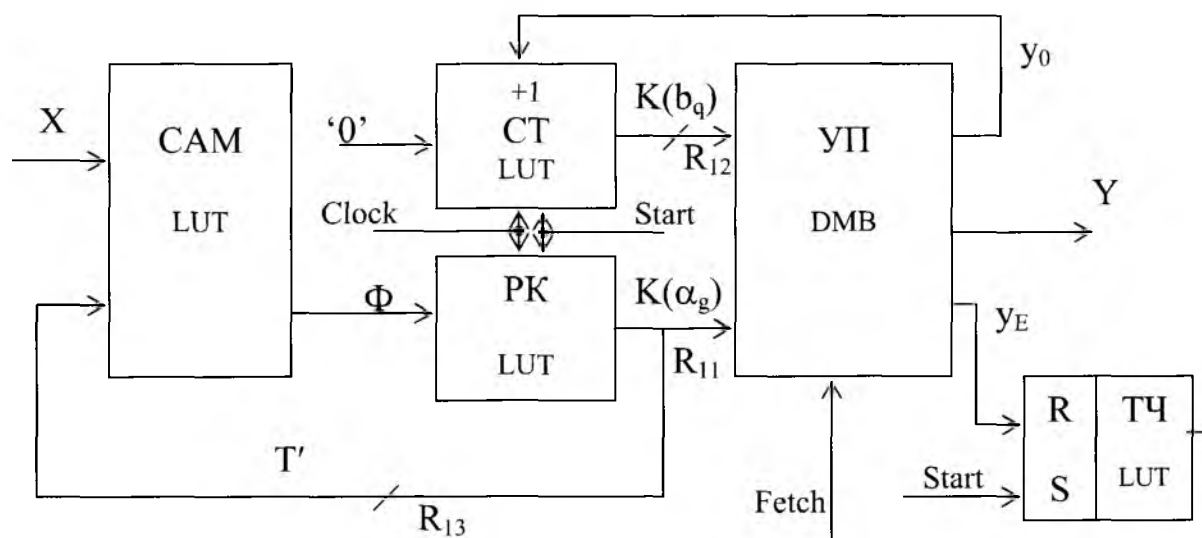
$$R_{12} = \lceil \log_2 F_{\max} \rceil, \quad (5)$$

где F_{\max} – максимальное число компонент ОЛЦ в множестве C .

Разрядность адреса микрокоманды составляет $R = R_{11} + R_{12}$ и для его представления используется множество внутренних переменных

$$T = \{T_1, \dots, T_{R_{11}}, T_{R_{11}+1}, \dots, T_{R_{11}+R_{12}}\}.$$

Использование элементарных ОЛЦ позволяет подавать на вход счетчика СТ нулевой код разрядности R_{12} , соответствующий коду первой компоненты текущей ОЛЦ $\alpha_g \in C$.

Рис. 2. Структурная схема КМУУ U_2

При таком подходе коды ОЛЦ и коды их компонент являются взаимно независимыми и на вход комбинационной схемы *CAM* подаются только R_{11} переменных обратной связи, образующих множество $\Gamma' = \{T_1, \dots, T_{R_{11}}\}$. Это позволяет уменьшить количество *LUT*-элементов, использующихся для реализации схемы *CAM*, по сравнению с КМУУ U_1 .

Функционирование КМУУ U_2 происходит следующим образом. По сигналу *Start* содержимое СТ и РК обнуляется, что является кодом первой компоненты первой ОЛЦ $\alpha_g \in C$, то есть адресу первой микрокоманды соответствующей ГСА Γ . Триггер чтения ТЧ устанавливается в единичное состояние и происходит считывание микрокоманды из УП.

Если адрес $A(Yq)$ компоненты не является адресом выхода текущей ОЛЦ $\alpha_g \in C$, то одновременно с микрооперациями $Yq \in Y$ формируется сигнал $y_0 = 1$. Таким образом, содержимое счетчика увеличивается на 1 и происходит переход к следующей компоненте Yt текущей ОЛЦ $\alpha_g \in C$.

Если адрес $A(Yq)$ компоненты является адресом выхода Og текущей ОЛЦ, то сигнал $y_0 = 0$, счетчик обнуляется, и схема *CAM* формирует код $K(\alpha q)$ следующей ОЛЦ $\alpha_q \in C$ в соответствии с системой

(4). Для представления кода используются переменные $\varphi_r \in \Phi = \{\varphi_1, \dots, \varphi_{R_{11}}\}$.

При формировании сигнала y_E триггер ТЧ сбрасывается и функционирование КМУУ U_2 завершается.

Предложен метод синтеза КМУУ U_2 .

Метод синтеза КМУУ U_2

Пусть алгоритм управления некоторой цифровой системой задан ГСА Γ_1 (рис. 3). Выполним синтез КМУУ U_2 по ГСА Γ_1 .

1. Формирование множества ОЛЦ

С. Этот этап выполняется по известной методике [7] и в нашем случае имеем $C = \{\alpha_1, \dots, \alpha_5\}$, где $\alpha_1 = \langle b_1, b_2, b_3, b_4 \rangle$, $\alpha_2 = \langle b_5, b_6 \rangle$, $\alpha_3 = \langle b_7, b_8, b_9, b_{10} \rangle$, $\alpha_4 = \langle b_{11}, b_{12} \rangle$, $\alpha_5 = \langle b_{13}, b_{14} \rangle$, $G = 5$.

2. Формирование множества элементарных ОЛЦ $C_E = \{\alpha_1, \dots, \alpha_{G_E}\}$.

Разбиение $\alpha_g \in C$ на элементарные ОЛЦ имеет смысл только в случае, если

$$|M(\Gamma)| < |I(\Gamma)|, \quad (6)$$

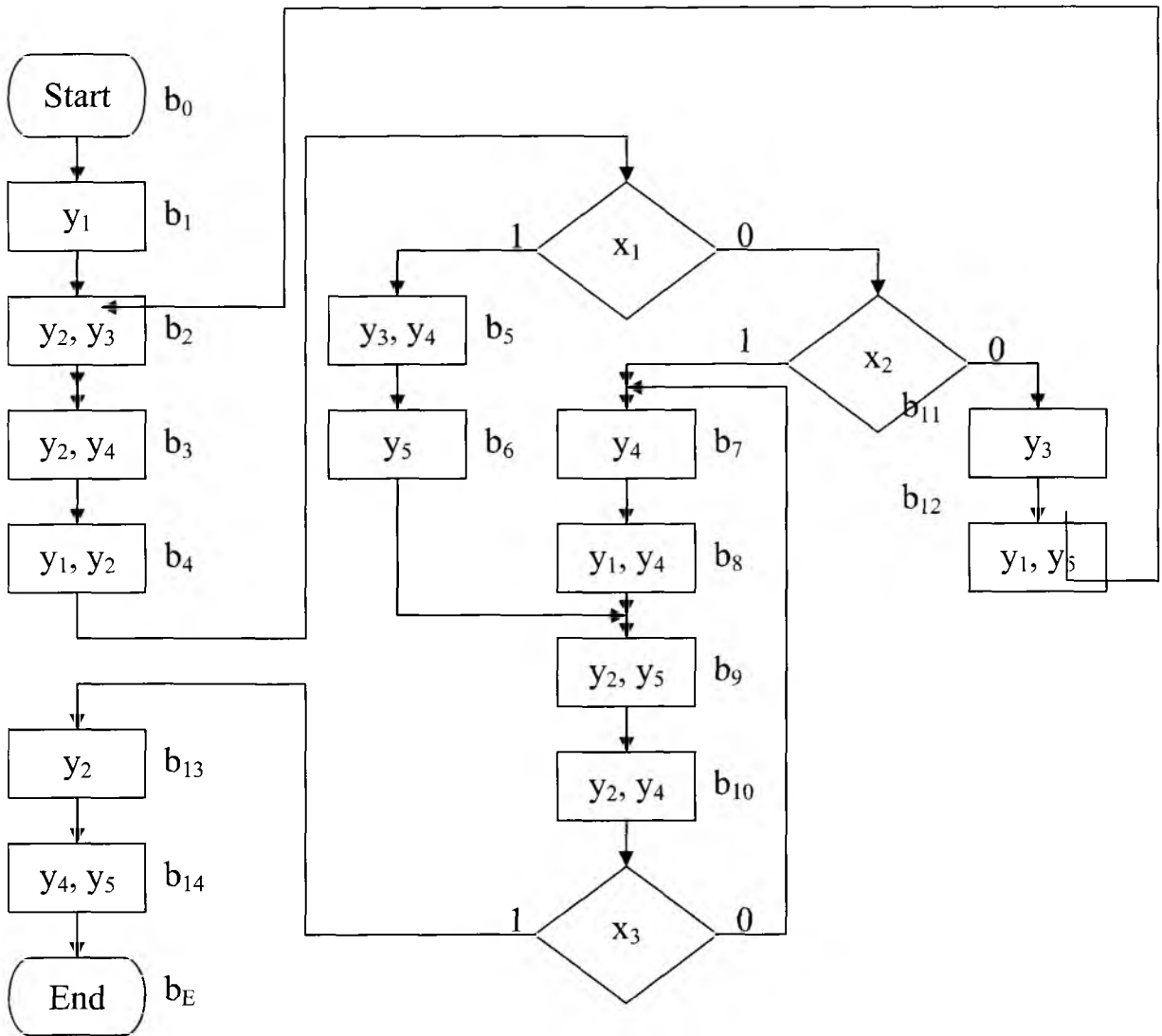


Рис. 3. Исходная граф-схема алгоритма Γ

где $MI(\Gamma)$ – множество главных входов ГСА Γ ; $I(\Gamma)$ – множество всех входов ГСА Γ . Для ГСА Γ_1 имеем $|MI(\Gamma_1)| = 5$ и $|I(\Gamma_1)| = 7$, следовательно, условие (6) выполняется.

Применение процедуры элементаризации [6] позволяет построить множество $C_E = \{\alpha_1, \dots, \alpha_7\}$ элементарных ОЛЦ, где $\alpha_1 = \langle b_1, b_2 \rangle$, $\alpha_2 = \langle b_3, b_4 \rangle$, $\alpha_3 = \langle b_5, b_6 \rangle$, $\alpha_4 = \langle b_7, b_8 \rangle$, $\alpha_5 = \langle b_9, b_{10} \rangle$, $\alpha_6 = \langle b_{11}, b_{12} \rangle$, $\alpha_7 = \langle b_{13}, b_{14} \rangle$, $G_E = 7$. Для кодирования элементарных ОЛЦ $\alpha_g \in C_E$ потребуется $R_{11} = 3$ переменных $T_\Gamma \in T$.

В общем случае, число элементарных ОЛЦ больше числа исходных ОЛЦ, что увеличивает длину таблицы переходов КМУУ, но в свою очередь, позволяет уменьшить число переменных обратной

связи на величину $\lceil \log_2(|I(\Gamma)| - |MI(\Gamma)|) \rceil$.

3. Формирование разбиения

$\Pi_{C_E} = \{V_1, \dots, V_I\}$ множества ОЛЦ

C_E на классы псевдоэквивалентных ОЛЦ. Разбиение на классы псевдоэквивалентных ОЛЦ выполняется только для $\alpha_g \in C_E$ таких, что выход этой ОЛЦ не связан с конечной вершиной ГСА. В нашем случае $C_E = \{\alpha_1, \dots, \alpha_6\}$ и формирование разбиения множества ОЛЦ C_E на классы псевдоэквивалентных ОЛЦ дает $\Pi'_{C_E} =$

$\{V_1, \dots, V_4\}$, где $V_1 = \{\alpha_1, \alpha_6\}$, $V_2 = \{\alpha_2\}$, $V_3 = \{\alpha_3, \alpha_4\}$, $V_4 = \{\alpha_5\}$, $I = 4$.

4. Оптимальное кодирование элементарных ОЛЦ. Оптимальное кодирование элементарных ОЛЦ $\alpha_g \in C_E$ вы-

полняется аналогично кодированию псевдоэквивалентных состояний автомата Мура [8]. При этом каждой элементарной ОЛЦ $\alpha_g \in C'_E$ ставится в соответствие R_{11} -разрядный код и все ОЛЦ $\alpha_g \in V_i$ должны располагаться в одном кубе булевого пространства. Вариант оптимального кодирования приведен на рис. 4, представляющем собой модифицированную карту Карно.

Как видно из рис. 4, полученные кубы булевого пространства дают коды, однозначно идентифицирующие классы $V_i \in \Pi_{C'_E}$: $K(V_1) = 00^*$, $K(V_2) = 01^*$, $K(V_3) = 10^*$, $K(V_4) = 11^*$. Это приводит к уменьшению числа переменных обратной связи $T_r \in T'$ от R_{11} до $R_{13} = \lceil \log_2 I \rceil = 2$.

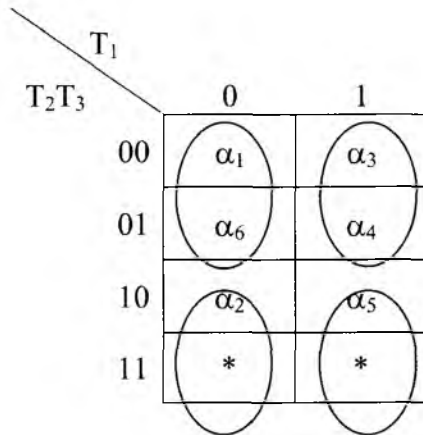


Рис. 4. Оптимальное кодирование ОЛЦ КМУУ $U_2(\Gamma_1)$

5. Формирование содержимого управляющей памяти УП. Результаты кодирования элементарных ОЛЦ приведены в табл. 1, коды ОЛЦ $\alpha_g \in C'_E$ взяты из карты Карно на рис. 4. Для кодирования ОЛЦ $\alpha_7 \notin C'_E$ взят один из несуществующих кодов.

Согласно (5) имеем $F_{\max} = 2$, то есть для кодирования компонент ОЛЦ $\alpha_g \in C_E$ необходимо $R_{12} = 1$ переменных $T_r \in T$. При кодировании первой компоненты ОЛЦ $\alpha_g \in C_E$ переменная $T_4 = 0$, второй компоненты — $T_4 = 1$.

Таблица 1. Кодирование элементарных ОЛЦ ГСА Γ_1

α_g	α_1	α_2	α_3	α_4	α_5	α_6	α_7
$K(\alpha_g)$	000	010	100	101	110	001	011

Таким образом, для адресации микрокоманд КМУУ $U_2(\Gamma_1)$ необходимо использовать $R = R_{11} + R_{12} = 3 + 1 = 4$ переменных, что не превышает соответствующий показатель для КМУУ $U_1(\Gamma_1)$. Однако, число переменных обратной связи $R_{13} = 2$ в КМУУ $U_2(\Gamma_1)$, меньше, чем $R_1 = 4$ в КМУУ $U_1(\Gamma_1)$. Следовательно, применение КМУУ U_2 , интерпретирующего элементарные ОЛЦ с разделением кодов, целесообразно.

Содержимое управляющей памяти УП для КМУУ $U_2(\Gamma_1)$ приведено в табл. 2. Здесь сигнал u_0 введен во все вершины, не являющиеся выходами ОЛЦ $\alpha_g \in C_E$, сигнал u_E введен в вершину b_{14} , связанную с конечной вершиной b_E .

6. Формирование таблицы переходов КМУУ. До построения таблицы переходов КМУУ $U_2(\Gamma_1)$ выполняется формирование системы формул перехода, в которых выходы ОЛЦ $\alpha_g \in V_i$ заменяются соответствующими классами $V_i \in \Pi_{C'_E}$, а входы ОЛЦ заменяются ОЛЦ $\alpha_q \in C_E$, в которые происходит переход.

В нашем случае имеем

$$\begin{aligned} V_1 &\rightarrow \alpha_2; \\ V_2 &\rightarrow x_1\alpha_3 \vee x_1x_2\alpha_4 \vee x_1x_2\alpha_6; \\ V_3 &\rightarrow \alpha_5; V_4 \rightarrow x_3\alpha_7 \vee x_3\alpha_4. \end{aligned} \quad (7)$$

При этом не используется формула для класса $V_5 \in \Pi_{C'_E}$, в который входит ОЛЦ $\alpha_7 \in C_E$, содержащая вершину b_{14} , связанную с конечной вершиной b_E ГСА Γ_1 . Таблица переходов КМУУ $U_2(\Gamma_1)$ строится по системе (7) (табл. 3).

Таблиця 2. Содержимое управляющей памяти КМУУ U_2 (Γ_1)

Адрес $A(b_q)$	Микрокоманда $Y(b_q)$							Примечания	
	y_0	y_1	y_2	y_3	y_4	y_5	y_E	b_q	α_g
0000	1	1	0	0	0	0	0	b_1	α_1
0001	0	0	1	1	0	0	0	b_2	
0100	1	0	1	0	1	0	0	b_3	α_2
0101	0	1	1	0	0	0	0	b_4	
1000	1	0	0	1	1	0	0	b_5	α_3
1001	0	0	0	0	0	1	0	b_6	
1010	1	0	0	0	1	0	0	b_7	α_4
1011	0	1	0	0	1	0	0	b_8	
1100	1	0	1	0	0	1	0	b_9	α_5
1101	0	0	1	0	1	0	0	b_{10}	
0010	1	0	0	1	0	0	0	b_{11}	α_6
0011	0	1	0	0	0	1	0	b_{12}	
0110	1	0	1	0	0	0	0	b_{13}	α_7
0111	0	0	0	1	0	1	1	b_{14}	

Таблиця 3. Таблица переходов КМУУ U_2 (Γ_1)

B_i	$K(B_i)$	α_q	$K(\alpha_q)$	X_h	Φ_h	h
B_1	00^*	α_2	010	–	D_2	1
B_2	01^*	α_3	100	x_1	D_1	2
		α_4	101	$\overline{x_1 x_2}$	$D_1 D_3$	3
		α_6	001	$\overline{x_1 x_2}$	D_3	4
B_3	10^*	α_5	110	–	$D_1 D_2$	5
B_4	11^*	α_7	011	x_3	$D_2 D_3$	6
		α_4	101	$\overline{x_3}$	$D_1 D_3$	7

7. Формирование системы функций Φ . Формирование системы функций (4) выполняется по таблице переходов КМУУ U_2 (Γ_1) (табл. 3) в виде:

$$\varphi_r = \bigvee_{h=1}^N C_{rh} E_i^h X_h \quad (r = \overline{1, R_{13}})$$

где C_{rh} – булева переменная, равная еди-

нице, если и только если в h -й строке таблицы переходов записана функция $\varphi_r = 1$; E_i^h – конъюнкция переменных $T_r \in \Gamma$, соответствующая коду $K(B_i)$ из h -й строки таблицы переходов $h = \overline{1, N}$ равна:

$$E_i^h = \bigwedge_{r=1}^{R_{13}} T_r^{1_{ir}} \quad (i = \overline{1, I})$$

где $1_{ir} \in \{*, 0, 1\}$ – значение r -го разряда кода $K(B_i)$, $T_r^* = 1$, $T_r^0 = \overline{T_r}$, $T_r^1 = T_r$ ($r = \overline{1, R_{13}}$).

Для КМУУ U_2 (Γ_1) имеем, например, $D_2 = \overline{T_1} \overline{T_2} \vee T_1 \overline{T_2} \vee T_1 T_2 X_3$.

8. Синтез логической схемы КМУУ на FPGA. Синтез логической схемы выполняется на *LUT*-элементах (регистр РК реализуется на входящих в состав *FPGA* триггерах) и управляющей памяти УП на блоках *DMB* [6,8].

Заключение

Предложенный метод элементаризации ОЛЦ с разделением кодов и оптимальным кодированием ОЛЦ позволяет при выполнении условия $R_{13} < R_{11}$ уменьшить число переменных обратной связи, необходимых для реализации системы функций (4), формирующих код следующей ОЛЦ ГСА. Применение метода не требует использования дополнительного регистра, так как для его реализации достаточно использовать имеющиеся в составе *FPGA* триггеры. В тоже время, происходит уменьшение разрядности счетчика *СТ*, который используется только для адресации компонент ОЛЦ (5). При этом время такта КМУУ U_1 и U_2 совпадают. Проведенные исследования показали, что использование КМУУ U_2 уменьшает число *LUT*-элементов программируемых логических блоков *FPGA*, необходимых для реализации схемы адресации микрокоманд, до 20% по сравнению с КМУУ U_1 .

Список литературы

1. Baranov S. Logic and System Design of Digital Systems. – Tallinn: TUT Press, 2008. – 266 с.

2. Соловьев В.В., Климович А.С. Логическое проектирование цифровых систем на основе программируемых логических интегральных схем. – М.: Горячая линия – Телеком, 2008. – 376 с.

3. Minks P., Eliot I. FSM – based Digital Design using Verilog. – London: John

Wiley & Sons, 2008. – 351 p.

4. Maxfield C. The Design Warrior's Guide to *FPGAs*. – Elsevier: Amsterdam, 2004. – 541 p.

5. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. – Донецк: ДонНТУ, 2002. – 262 с.

6. Баркалов А.А., Титаренко Л.А. Синтез композиционных микропрограммных устройств управления. – Харьков: Коллегиум, 2007. – 302 с.

7. Баркалов А.А., Титаренко Л.А. Синтез микропрограммных автоматов на заказных и программируемых СБИС. – Донецк: УНИТЕХ, 2009. – 336 с.

8. Баркалов А.А. Принципы оптимизации логической схемы микропрограммного автомата Мура // Кибернетика и системный анализ. – 1998. – №1. – С. 65 – 72.

9. Баркалов А.А., Ковалев С.А., Ефименко К.Н. Оптимизация числа *LUT*-элементов в композиционном микропрограммном устройстве управления с общей памятью // Управляющие системы и машины. – 2006. – №4. – С. 50 – 55.