

ОПТИМИЗАЦИЯ УПРАВЛЯЮЩЕЙ ПАМЯТИ КОМПОЗИЦИОННЫХ УСТРОЙСТВ УПРАВЛЕНИЯ

Баркалов А.А.

University Zelenogursky Institution of Informatics and Electronics, Poland

A.Barkalov@iie.uz.zgora.pl

Abstract

Barkalov A.A. Optimization of control memory of microprogram control units. The method of control memory's optimization based on existence of identical operational linear chains is proposed. An algorithm of control unit's design with minimized control memory is discussed. An example of application of proposed method is given.

1. Введение

Устройство управления (УУ) является одной из центральных частей любой цифровой системы [1]. В настоящее время для синтеза УУ широко используются программируемые логические устройства (ПЛУ) [2, 3]. Этот базис отличается высокой стоимостью, поэтому одна из проблем при синтезе УУ на ПЛУ – минимизация стоимости схемы при заданных характеристиках быстродействия [3]. Методы решения этой проблемы зависят от типа УУ [4-6]. В данной работе предлагается метод решения этой задачи при синтезе композиционных микропрограммных устройств управления (КМУУ) [4].

2. Основная идея метода

Пусть для граф-схемы алгоритма (ГСА) Γ получено множество операторных линейных цепей (ОЛЦ) $C = \{\alpha_1, \dots, \alpha_G\}$, каждая из которых представляет собой последовательность операторных вершин $b_q \in V$, где $V = \{b_1, \dots, b_Q\}$ – множество операторных вершин ГСА Γ . Любая ОЛЦ $\alpha_g \in C$ определяется как вектор [4]

$$\alpha_g = \langle b_{g1}, \dots, b_{gF_g} \rangle \quad (g = \overline{1, G}), \quad (1)$$

такой, что для каждой пары соседних компонент $b_{gi}, b_{gj} \in V$ существует дуга $\langle b_{gi}, b_{gj} \rangle$ в ГСА Γ ($j = i+1, i \in \{1, \dots, F_g-1\}$). Пусть $A(b_q)$ – адрес микрокоманды, соответствующей вершине $b_q \in V$, разрядности

$$R_1 = \lceil \log_2 Q \rceil \quad (2)$$

Если адреса микрокоманд удовлетворяют условию

$$A(b_{gj}) = A(b_{gi}) + 1 \quad (j = i+1, i = \overline{1, F_g-1}), \quad (3)$$

то УУ может быть реализовано как КМУУ U_1 (рис. 1).

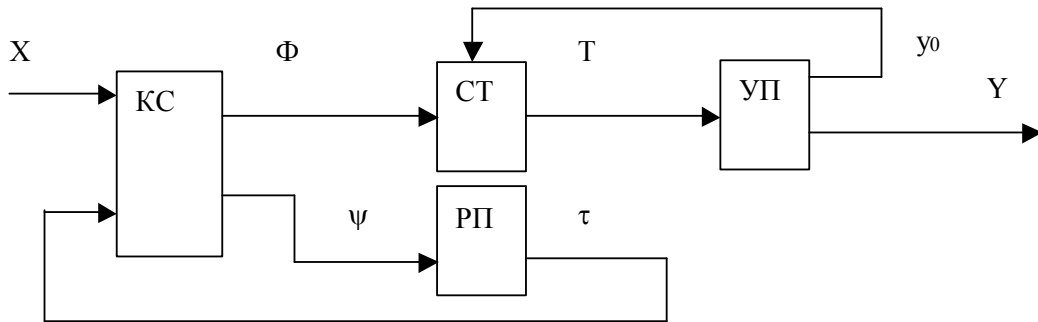


Рис. 1. Структурная схема композиционного микропрограммного устройства управления

Здесь схема КС и регистр памяти РП образуют автомат адресации микрокоманд, состояния которого $a_m \in A = \{a_1, \dots, a_M\}$ закодированы R_2 -разрядными кодами $K(a_m)$ с использованием переменных $\tau_r \in \tau$, $|\tau| = R_2$, $R_2 = \lceil \log_2 M \rceil$. Схема КС формирует функции возбуждения регистра ψ и счетчика Φ в виде

$$\begin{aligned} \Phi &= \Phi(\tau, X), \\ \psi &= \psi(\tau, X), \end{aligned} \quad (4)$$

где $X = \{x_1, \dots, x_L\}$ – множество логических условий. Функции Φ формируют в СТ адреса входов ОЛЦ [4], кодируемые переменными $T_r \in T$ ($r = \overline{1, R_1}$). Управляющая память УП формирует микрооперации $y_n \in Y$, где $Y = \{y_1, \dots, y_N\}$, и сигнал y_0 , по которому к содержимому СТ прибавляется единица для поддержания режима естественной адресации (3).

Назовем ОЛЦ $\alpha_g, \alpha_k \in C$ идентичными ОЛЦ, если $F_g = F_k$ и для вершин, соответствующих компонентам с одинаковыми номерами, выполняется условие

$$Y(b_{gi}) = Y(b_{ki}) \quad (i = \overline{1, F_g}), \quad (5)$$

где $Y(b_i)$ – набор микроопераций, записанных в вершине $b_i \in V$.

Поясним это определение на примере ГСА Γ_1 (Рис. 2), для которой $C = \{\alpha_1, \dots, \alpha_6\}$, где $\alpha_1 = \langle b_1, b_2 \rangle$, $\alpha_2 = \langle b_3, b_4 \rangle$, $\alpha_3 = \langle b_5, b_6 \rangle$, $\alpha_4 = \langle b_7, b_8 \rangle$, $\alpha_5 = \langle b_9, b_{10} \rangle$, $\alpha_6 = \langle b_{11}, b_{12} \rangle$. Отметим, что все ОЛЦ в данном случае имеют только один вход, то есть являются элементарными ОЛЦ [4]. Здесь b_0 и b_k соответственно начальная и конечная вершина ГСА Γ_1 .

Зададим на множестве C отношение идентичности β , причем $\alpha_i \beta \alpha_j$, если и только если α_i, α_j – идентичные ОЛЦ ($i, j \in \{1, \dots, G\}$). Отношение β рефлексивно, симметрично, транзитивно и, следовательно, является

отношением эквивалентности, определяющим разбиение π_β множества S на классы идентичных ОЛЦ: $\pi_\beta = \{B_1, \dots, B_l\}$.

Идея оптимизации заключается в сопоставлении каждому классу $B_i \in \pi_\beta$ единой области УП, состоящей из F_g ячеек, где F_g – длина ОЛЦ $\alpha_g \in B_i$.

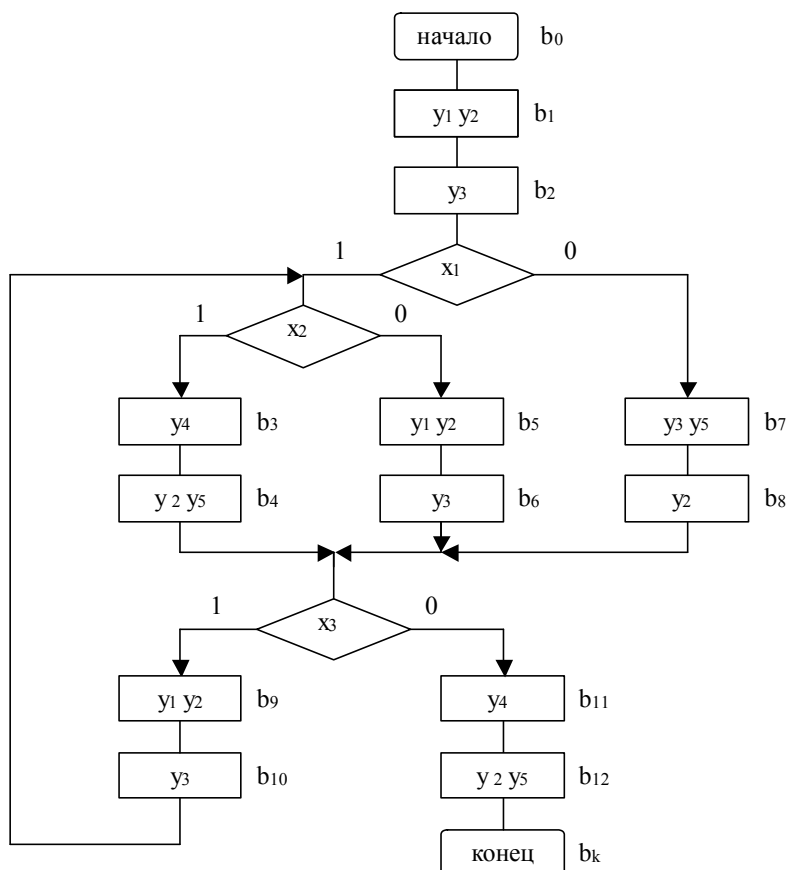


Рис. 2. Исходная граф-схема алгоритма Γ_1

Если класс $B_i \in \pi_\beta$ включает G_i ОЛЦ, то при традиционном подходе [4] ему соответствует

$$K_i = G_i F_g^i \quad (6)$$

ячеек УП, а при предлагаемом – только F_g . В целом в УП будет храниться

$$Q_1 = \sum_{i=1}^l F_g^i \quad (7)$$

микрокоманд, где F_g^i – длина ОЛЦ $\alpha_g \in B_i$ ($i = \overline{1, l}$). Пусть $R_3 = \lceil \log_2 Q_1 \rceil$, тогда при выполнении условия

$$R_3 < R_1 \quad (8)$$

применение предлагаемого метода уменьшает требуемую емкость УП по сравнению с этим показателем КМУУ U_1 .

3. Синтез КМУУ с идентичными ОЛЦ

Для ГСА $\Gamma \pi_{\beta} = \{B_1, B_2, B_3\}$, где $B_1 = \{\alpha_1, \alpha_3, \alpha_5\}$, $B_2 = \{\alpha_2, \alpha_6\}$, $B_3 = \{\alpha_4\}$, $G_1 = 3$, $F_g^1 = 2$, $G_2 = 2$, $F_g^2 = 2$, $G_3 = 1$, $F_g^3 = 2$, $Q=12$. Согласно [6], $Q_1=6$ тогда $R_3=3$ и условие (8) выполняется, то есть учет идентичных ОЛЦ может сократить разрядность адреса УП.

Как видно из рис. 2, переходы из выходов идентичных ОЛЦ $\alpha_g \in B_i$ различны. Поэтому для организации адресации микрокоманд, соответствующей исходной ГСА, необходимо ввести идентификаторы, позволяющие однозначно определить представителя каждого класса $B_i \in \pi_{\beta}$. Пусть V – множество переменных, кодирующих идентификаторы, тогда КМУУ U_2 с идентичными ОЛЦ имеет следующую структуру (Рис. 3).

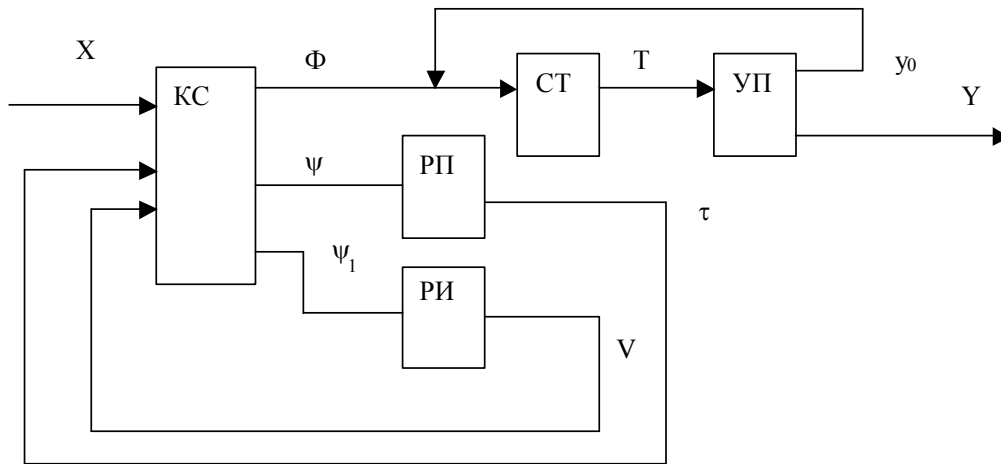


Рис. 3. Структурная схема КМУУ U_2

Здесь РИ – регистр идентификаций, управляемый формируемыми КС функциями ψ_1 .

В настоящей работе предлагается методика синтеза КМУУ U_2 , включающая следующие этапы:

1. Определение мощности R_4 множества V . Число идентификаторов для класса $B_i \in \pi_{\beta}$ совпадает с мощностью $G_i = |B_i|$. Пусть $R(G_i) = \lceil \log_2 G_i \rceil$ – число переменных, кодирующих идентичные ОЛЦ $\alpha_g \in B_i$, тогда для кодирования идентификаторов достаточно

$$R_4 = \max(R(G_1), \dots, R(G_n)) \quad (9)$$

переменных $v_r \in V$.

2. Кодирование операторных линейных цепей. Поставим в соответствие каждой ОЛЦ $\alpha_g \in B_i$ двоичный код $K(\alpha_g)$ разрядности $R(G_i) \leq R_4$, используя для кодирования $R(G_i)$ младших переменных из вектора $\langle v_1, \dots, v_{R_4} \rangle$.

3. Преобразование исходной ГСА Γ . Введем в вершины – входы ОЛЦ $\alpha_g \in V_i$ функции $v_r \in V$, принимающие единичное значение в коде $K(\alpha_g)$. Удалим из (G_i-1) -й ОЛЦ $\alpha_g \in V_i$ все компоненты, начиная со второй, при этом ОЛЦ, компоненты которой не удаляются, назовем базовой ОЛЦ блока $V_i \in \pi_\beta$. Построим систему формул перехода [6]

$$b_i^i \rightarrow b_2(V_i), \quad (i = \overline{2, G_i}) \quad (10)$$

где $b_2(V_i)$ – вторая компонента базовой ОЛЦ блока $V_i \in \pi_\beta$, b_1^i – первая компонента i -й ОЛЦ блока $V_i \in \pi_\beta$. Для всех вершин – выходов базовых ОЛЦ построим систему формул перехода

$$b_q \rightarrow v_{qt} X_h b_t, \quad (11)$$

где b_q – выход ОЛЦ $\alpha_g \in V_i$, v_{qt} – конъюнкция переменных $v_r \in V$, соответствующая коду $K(\alpha_g)$, X_h – конъюнкция логических условий, определяющая переход из b_q в b_t . Преобразованная ГСА Γ' строится по системам (10) – (11) и формуле перехода для вершины b_0 .

4. Адресация микрокоманд. Выполним естественную адресацию микрокоманд только для базовых ОЛЦ. Адреса, соответствующие компонентам с одинаковыми номерами для идентичных ОЛЦ, дублируются. Зададим содержимое УП, используя эти адреса.

5. Формирование прямой структурной таблицы автомата адресации. Преобразуем ГСА Γ' по известной методике [4], оставив в ней только вершины b_0, b_k , входы ОЛЦ $\alpha_g \in C$ и все условные вершины. Отметим преобразованную ГСА Γ'' состояниями автомата Мили $a_m \in A = \{a_1, \dots, a_m\}$. Закодируем состояния $a_m \in A$ R_2 – разрядными кодами $K(a_m)$ и построим прямую, структурную таблицу (ПСТ), из которой получим систему функций

$$\begin{aligned} \Phi &= (r, V, x); \\ \Psi &= (r, V, x); \\ \Psi_1 &= (r, V, x). \end{aligned} \quad (12)$$

6. Построение схемы КМУУ U_2 . Схема КС строится по системе (12), а управляющая память – по таблице содержимого УП. Методы решения этих задач достаточно освещены в литературе [2,3].

4. Пример применения предложенного метода синтеза

Рассмотрим пример синтеза КМУУ U_2 по ГСА Γ_1 , для которой $R(G_1)=2, R(G_2)=1, R(G_3)=0$ и, согласно (8), $R_4=2, V=\{v_1, v_2\}$.

Закодируем ОЛЦ $\alpha_g \in C$ следующим образом $K(\alpha_1)=00, K(\alpha_3)=01, K(\alpha_5)=10, K(\alpha_2)=0, K(\alpha_6)=1, K(\alpha_4)=0$. Теперь в вершины b_5, b_{11} вводится переменная v_2 , а в вершину $b_9 - v_1$.

Система формул перехода (10) имеет вид: $b_1, b_5, b_9 \rightarrow b_2$; $b_3, b_{11} \rightarrow b_4$; $b_7 \rightarrow b_8$, то есть базовыми ОЛЦ выбраны соответственно $\alpha_1, \alpha_2, \alpha_4 \in C$.

Система формул перехода (11) имеет вид:

$$\begin{aligned} b_2 &\rightarrow \overline{v_1} \overline{v_2} (x_1 x_2 b_3 \vee x_1 \overline{x_2} b_5 \vee \overline{x_1} b_7) \vee \overline{v_1} v_2 (x_3 b_9 \vee \overline{x_3} b_{11}) \vee v_1 (x_2 b_3 \vee \overline{x_2} b_5); \\ b_4 &\rightarrow \overline{v_2} (x_3 b_9 \vee \overline{x_3} b_{11}) \vee v_2 b_k; \\ b_8 &\rightarrow x_3 b_9 \vee \overline{x_3} b_{11} \end{aligned}$$

Используя эти системы и формулу перехода $b_0 \rightarrow b_1$, построим преобразованную ГСА Γ_1' (Рис.4).

Адресация микрокоманд, соответствующих базовым ОЛЦ $\alpha_g \in C$, приводит к содержимому УП (Табл. 1).

Построение преобразованной ГСА Γ_1'' (Рис.5) по методике [4] не представляет труда.

Таблица 1

Содержимое управляющей памяти КМУУ U_2 для ГСА Γ_1

Адрес	Микрооперации	Примечание
000	$y_0 y_1 y_2$	$b_1 b_5 b_9$ B_1
001	y_3	$b_2 b_6 b_{10}$
010	$y_0 y_4$	$b_3 b_{11}$ B_2
011	$Y_2 y_5$	$b_4 b_{12}$
100	$Y_0 y_3 y_5$	b_7 B_3
101	Y_2	b_8

Закодируем состояния $\alpha_m \in A$ следующим образом: $K(a_1)=00, \dots, K(a_4)=11$ ($R_2=2, \tau=\{\tau_1, \tau_2\}$) и построим ПСТ автомата адресации (табл. 2), имеющую $H=13$ строк.

Так как все элементы памяти в нашем случае имеют входы D-типа, то адресам микрокоманд соответствуют функции D_1-D_3 , функции возбуждения РП – D_4, D_5 , функции возбуждения РИ – D_6 (соответствует v_1) и D_7 (соответствует v_2). Символ “1” в ПСТ соответствует конъюнкции пустого множества переменных. Система (12) строится в виде

$$\varphi_r = \bigvee_{h=1}^H C_{rh} A_m^h X_h V_h (r=1, R_3+R_2+R_4), \quad (13)$$

где C_{rh} – булева переменная, равная единице, если и только если в h -й строке ПСТ записана переменная $\varphi_r \in \Phi \cup \Psi \cup \Psi_1$, A_m^h – конъюнкция переменных $\tau_r \in \tau$, соответствующая коду $K(a_m)$ исходного состояния из h -й строки ПСТ.

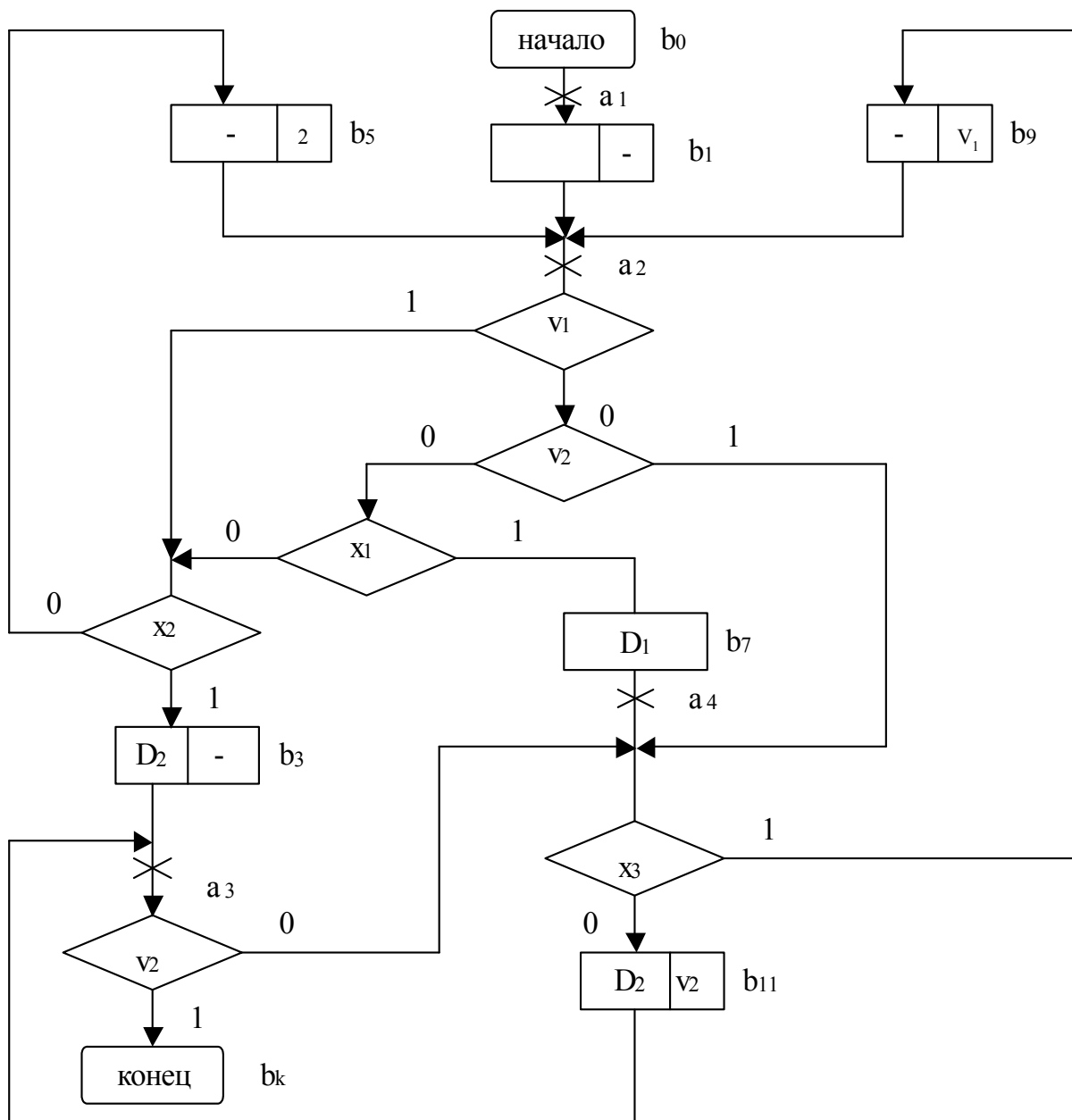


Рис. 5. Преобразованная граф-схема алгоритма Γ_1''

5. Заключение

Предложенный метод позволяет уменьшить стоимость управляющей памяти КМУУ, не увеличивая числа состояний автомата адресации. В данной работе изложена только общая идея метода, которая требует развития в трех направлениях:

1. Минимизация числа дополнительных переменных, кодирующих ОЛЦ, за счет увеличения числа классов разбиения π_β таким образом, чтобы условие (7) выполнялось при минимальной мощности классов $V_i \in \pi_\beta$.

2. Уточнение методики синтеза для случая ОЛЦ с несколькими входами, более распространенным на практике.

Таблица 2

Прямая структурная таблица автомата адресации

a_m	$K(a_m)$	a_s	$K(a_s)$	X_h	V_h	Φ_h	Ψ_h	Ψ_{1h}	h
a_1	00	a_2	01	1	1	—	D_5	—	1
a_2	01	a_2	01	$\overline{x_2}$	v_1	—	D_5	D_7	2
		a_3	10	x_2	$\overline{v_1}$	D_2	D_4	—	3
		a_4	11	x_1	$\overline{v_1} \overline{v_2}$	D_1	$D_4 D_5$	—	4
		a_3	10	$\overline{x_1} \overline{x_2}$	$\overline{v_1} \overline{v_2}$	D_2	D_4	—	5
		a_2	01	$\overline{x_1} \overline{x_2}$	$\overline{v_1} \overline{v_2}$	—	D_5	D_7	6
		a_2	01	x_3	$\overline{v_1} v_2$	—	D_5	D_6	7
a_3	10	a_3	10	$\overline{x_3}$	$\overline{v_1} v_2$	D_2	D_4	—	8
		a_1	00	1	$\overline{v_2}$	—	—	—	9
		a_2	01	x_3	$\overline{v_2}$	—	D_5	D_7	10
a_4	11	a_3	10	x_3	v_2	D_2	D_4	—	11
		a_2	01	x_3	1	—	D_5	D_7	12
		a_3	10	x_3	1	D_2	D_4	—	13

3. Развитие методики синтеза для всех известных структур КМУУ [4], а не только для КМУУ с базовой структурой.

По мнению автора, аналоги этого метода могут быть использованы для минимизации стоимости схем микропрограммных автоматов Мили и Мура, реализуемых на ПЛУ.

6. Литература

1. Баркалов А. А., Палагин А.В. Синтез микропрограммных устройств управления. – Киев: ИК НАН Украины, 1997.- 136 с.
2. Соловьев В.В. Проектирование цифровых систем на основе программируемых логических интегральных схем. – М.: Горячая линия – Телеком, 2001. – 636 с.
3. Грушницкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем на микросхемах программируемой логики. – СПб: БХВ – Петербург, 2002. – 636 с.
4. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. – Донецк: ДонНТУ, 2002. – 262 с.
5. Bursky D. Embedded Logic and Memory Find Home in FPGA. // Electronic Design. – 1999. - №14. p.43 – 56.
6. Baranov S. Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 p.

Дата надходження до редколегії: 6.12.2003 р.