

2. *Funk, P.* Advances in Case-Based Reasoning / P. Funk, P. A. Gonzalez-Calero // 7th European Conference, ECCBR 2004. – Madrid, Spain. – P. 375–380.
3. *Рассел, С.* Искусственный интеллект / С. Рассел, П. Норвиг. – М. ; СПб. ; К. : Вильямс, 2006. – 1408 с.
4. *Gruber, T. R.* A translation approach to portable ontologies / T.R.Gruber // Knowledge Acquisition. – 1993. – N 5 (2). – P. 199–220.
5. Інтелектуальні системи, базовані на онтологіях // Д. Г. Досин, В. В. Литвин, Ю. В. Нікольський, В. В. Пасічник. – Львів : Цивілізація, 2009. – 414 с.
6. *Литвин, В. В.* Мультиагентні системи підтримки прийняття рішень, що базуються на прецедентах та використовують адаптивні онтології / В. В. Литвин // Радіоелектроніка, інформатика, управління. – 2009. – № 2 (21). – С. 120–126.
7. *Sowa, J.* Conceptual graphs for a database interface / J.Sowa // IBM Journal of Research and Development. – Vol. 20. – № 4. – 1976. – P. 336–357.
8. *Свами, М.* Графы, сети и алгоритмы / М. Свами, К. Тхула-сираман. – М. : Наука, 1984. – 408 с.
9. *Литвин, В. В.* Проектування інтелектуальних агентів прийняття рішень в просторі ознак з використанням онтологічного підходу / В. В. Литвин, Р. Р. Даревич, Д. Г. Досин, Н. В. Шкутяк // Штучний інтелект. – 2010. – № 3. – С. 254–262.
10. *Скобелев, В. Г.* Локальные алгоритмы на графах / В. Г.Скобелев. – Донецк : ИПММ НАН Украины, 2003. – 217 с.
11. *Крайовський, В. Я.* Основні підходи до розроблення програмного комплексу автоматичного реферування текстових документів / В. Я. Крайовський В. В. Литвин, Н. Б. Шаховська // Інститут проблем моделювання в енергетиці. – 2009. – Випуск 51. – С. 178–186.
12. *Даревич, Р. Р.* Оцінка подібності текстових документів на основі визначення інформаційної ваги елементів бази знань / Р. Р. Даревич, Д. Г. Досин, В. В. Литвин, З. Т. Назарчук // Штучний інтелект. – 2006. – № 3. – С. 500–509.
13. *Литвин, В. В.* Метод автоматизованого реферування текстових документів з використанням онтологій / В. В. Литвин, В. А. Гайдін, О. Ю. Пшеничний // Складні системи і процеси. – Запоріжжя. – 2009. – № 1. – С. 81–87.
14. *Цветков, А. М.* Разработка алгоритмов индуктивного вывода с использованием деревьев решений / А. М. Цветков // Кибернетика и системный анализ. – № 1. – 1993. – С. 174–178.
15. *Bulskov, H.* On Querying Ontologies and Databases / H. Bulskov, R. Knappe, R. Andreasen // FQAS. – 2004. – P. 191–202.
16. *Фаулер, М.* UML в кратком изложении / М. Фаулер, К. Скотт. – М. : Мир, 1999. – 340 с.

Стаття надійшла до редакції 02.12.2010.

Лытвын В. В.

#### МОДЕЛИРОВАНИЕ ИНТЕЛЛЕКТУАЛЬНЫХ СИСТЕМ ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ С ИСПОЛЬЗОВАНИЕМ ОНТОЛОГИЧЕСКОГО ПОДХОДА

Исследовано функционирование интеллектуальных систем поддержки принятия решений, ядром баз знаний которых являются онтологии. Рассматриваются интеллектуальные агенты четырех типов. Для каждого типа разработана метрика, которая используется для определения релевантности предлагаемых системой решений.

**Ключевые слова:** интеллектуальная система поддержки принятия решений, интеллектуальный агент.

Lytvyn V. V.

#### INTELLIGENT DECISION SUPPORT SYSTEMS MODELING USING ONTOLOGICAL APPROACH

The functioning of intelligent decision support systems with knowledge base based on ontology is investigated. Four types of intelligent agents are considered. Metrics for each of them is developed and used to determine the relevance of solutions proposed by the system.

**Keywords:** Intelligent Decision Support System, Intelligent Agent.

УДК 004.3

Баркалов А. А.<sup>1</sup>, Зеленева И. Я.<sup>2</sup>, Цололо С. А.<sup>3</sup>, Биайрак Х.<sup>4</sup>

<sup>1</sup> Д-р техн. наук, профессор Донецкого национального технического университета

<sup>2,3</sup> Канд. техн. наук, доцент Донецкого национального технического университета

<sup>4</sup> Аспирант Донецкого национального технического университета

## УМЕНЬШЕНИЕ ПЛОЩАДИ МАТРИЧНОЙ СХЕМЫ УСТРОЙСТВА УПРАВЛЕНИЯ С РАЗДЕЛЕНИЕМ КОДОВ

В статье предложена модель композиционного микропрограммного устройства управления с разделением кодов, ориентированная на реализацию логической схемы устройства в базе заказных матриц. В модели используется представление адреса вершины алгоритма управления в виде конкатенации кодов ОЛЦ и кода компоненты ОЛЦ. Этот подход позволяет уменьшить число входов и выходов схемы формирования функций возбуждения.

**Ключевые слова:** композиционное устройство управления, матричная схема, операторная линейная цепь (олц), разделение кодов.

### ВВЕДЕНИЕ

При реализации схемы устройства управления (УУ) необходимо учитывать особенности элементного бази-

са и алгоритмы управления [1]. Если алгоритм управления представлен линейной граф-схемой алгоритма (ГСА), то для его реализации целесообразно использо-

вать модель композиционного микропрограммного устройства управления (КМУУ) [2]. В настоящей работе рассматривается задача реализации схемы КМУУ в базе заказных матричных схем [3], которые широко используются при массовом производстве средств вычислительной техники и автоматики. В этом случае возникает задача уменьшения площади кристалла, занимаемого схемой КМУУ [4; 5]. Решение этой задачи позволяет уменьшить потребляемую мощность и повысить выпуск годных кристаллов. При этом рассматривается случай выполнения условий, позволяющих использовать модель КМУУ с разделением кодов [2].

Целью исследования является уменьшение аппаратных затрат в схеме КМУУ с разделением кодов за счет использования нескольких источников классов псевдоэквивалентных ОЛЦ.

Задачей исследования является разработка метода, позволяющего уменьшить площадь, занимаемую схемой адресации микрокоманд в КМУУ с разделением кодов.

**РЕАЛИЗАЦИЯ КМУУ С РАЗДЕЛЕНИЕМ КОДОВ НА ЗАКАЗНЫХ МАТРИЦАХ**

Пусть ГСА  $\Gamma$  представлена множествами вершин  $V$  и дуг  $E$ , соединяющих эти вершины. При этом  $V = \{b_0, b_E\} \cup E_1 \cup E_2$ , где  $b_0$  – начальная вершина ГСА,  $b_E$  – конечная вершина ГСА,  $E_1$  – множество операторных вершин, где  $|E_1| = M$ ,  $E_2$  – множество условных вершин. В вершинах  $b_q \in E_1$  записаны наборы микроопераций  $Y(b_q) \subseteq Y$ , где  $Y = \{y_1, \dots, y_N\}$  – множество микроопераций. В вершинах  $b_q \in E_2$  записаны элементы множества логических условий  $X = \{x_1, \dots, x_L\}$ . Пусть ГСА является линейной, то есть включает более 75% операторных вершин [3].

Сформируем множество ОЛЦ  $C = \{\alpha_1, \dots, \alpha_G\}$  ГСА  $\Gamma$ , где каждая из ОЛЦ является последовательностью операторных вершин и каждой паре ее соседних компонент  $b_i, b_j$  соответствует дуга  $\langle b_i, b_j \rangle \in E$ . Каждая ОЛЦ имеет только один выход  $O_g$  и произвольное число входов  $I_g (g = 1, \dots, G)$ . Формальные определения ОЛЦ, их входов и выходов можно найти в [3]. Каждая вершина

$b_q \in E_1$  соответствует микрокоманде  $MI_q$ , хранимой в управляющей памяти (УП) КМУУ по адресу  $A(b_q)$ . Для адресации микрокоманд достаточно

$$R = \lceil \log_2 M \rceil \tag{1}$$

бит, представленных переменными  $T_r \in T$ , где  $|T| = R$ .

Пусть ОЛЦ  $\alpha_g \in C$  включает  $F_g$  компонент и пусть  $Q = \max(F_1, \dots, F_G)$ . В этом случае для кодирования компонент достаточно

$$R_C = \lceil \log_2 Q \rceil \tag{2}$$

переменных, образующих множество  $\tau$ , где  $|\tau| = R_C$ . Для кодирования ОЛЦ  $\alpha_g \in C$  достаточно

$$R_G = \lceil \log_2 G \rceil \tag{3}$$

переменных, образующих множество  $T$ , где  $|T| = R_G$ .

Пусть  $K(\alpha_g)$ ,  $K(b_q)$  соответственно код ОЛЦ  $\alpha_g \in C$  и код компоненты некоторой ОЛЦ. Тогда адрес микрокоманды, соответствующей вершине  $b_q \in E_1$ , может быть представлен в виде конкатенации

$$A(b_q) = K(\alpha_g) * K(b_q). \tag{4}$$

В выражении (4)  $b_q$  является компонентой ОЛЦ  $\alpha_g \in C$ , а операция конкатенации обозначается знаком  $*$ . Выполним адресацию компонент так, чтобы для каждой ОЛЦ  $\alpha_g \in C$  их коды возрастали в естественном порядке. При этом первая компонента любой ОЛЦ имеет код 0, вторая – 1, и так далее до  $Q$ .

В этом случае для реализации схемы УУ может быть использована модель КМУУ с разделением кодов (рис. 1).

Назовем эту модель КМУУ  $U_1$ . В КМУУ  $U_1$  матрицы  $M_1$  и  $M_2$  образуют блок адресации микрокоманд (БАМ), а матрицы  $M_3$  и  $M_4$  – управляющую память (УП). Блок БАМ реализует системы функций возбуждения триггеров счетчика СТ и регистра RG :

$$\Phi = \Phi(X, \tau), \tag{5}$$

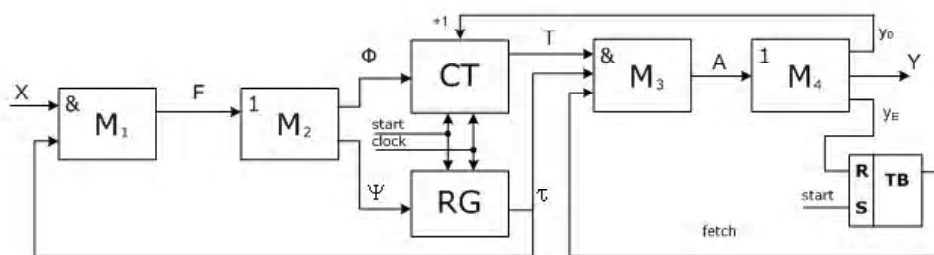


Рис. 1. Матричная реализация КМУУ с разделением кодов

$$\Psi = \Psi(X, \tau). \tag{6}$$

При этом матрица  $M_1$  реализует систему термов  $F = \{F_1, \dots, F_H\}$ , входящих в функции (5)–(6). Матрица  $M_2$  реализует функции (5)–(6), как дизъюнкции термов  $F_h \in F$ :

$$\left. \begin{aligned} \Phi_r &= \bigvee_{h=1}^H C_{rh} F_h (r = \overline{1, R_0}); \\ \Psi_r &= \bigvee_{h=1}^H C_{rh} F_h (r = \overline{1, R_C}). \end{aligned} \right\} \tag{7}$$

В функциях (7)  $C_{rh} \in \{0,1\}$  и  $C_{rh} = 1$ , если и только если терм  $F_h$  входит в функцию  $\Phi_r$  или  $\Psi_r$ .

Матрица  $M_3$  реализует систему функций  $A$ , соответствующих адресам микрокоманд (4):

$$A(b_q) = \left( \bigwedge_{r=1}^{R_0} \tau_r^{l_{gr}} \right) \cdot \left( \bigwedge_{r=1}^{R_C} T_r^{l_{qr}} \right). \tag{8}$$

В формуле (8)  $l_{gr} \in \{0,1\}$  – значение  $r$ -го разряда кода  $K(\alpha_g)$ , где  $\alpha_g \in C$  и  $b_q$  входит в эту ОЛЦ;  $l_{qr} \in \{0,1\}$  – значение  $r$ -го разряда кода  $K(b_q)$ ;  $\tau_r^0 = \overline{\tau_r}$ ,  $\tau_r^1 = \tau_r (r = \overline{1, R_0})$ ,  $\tau_r^0 = \overline{\tau_r}$ ,  $\tau_r^1 = \tau_r (r = \overline{1, R_C})$ . Отметим, что некоторые разряды этих кодов могут быть несущественными, однако этот случай в данном случае не рассматривается. Для формирования терма  $A_m \in A(m = \overline{1, M})$  соответствующий терм  $A(b_q)$  умножается на переменную Fetch. При Fetch=1 разрешается выборка микрокоманд из УП.

Матрица  $M_4$  реализует систему функций

$$y_n = \bigvee_{m=1}^M C_{nm} A_m, \tag{9}$$

где  $n \in \{0,1, \dots, N, E\}$ , а  $C_{nm} = 1$ , если и только если функция  $y_n$  формируется в микрокоманде с индексом  $m$ . Переменная  $y_0$  служит для увеличения содержимого СТ на единицу, что позволяет адресовать компонент одной ОЛЦ без использования блока БАМ. Переменная  $y_E$  формируется при достижении конечной вершины ГСА  $\Gamma$  и служит признаком окончания алгоритма. Если  $y_E = 1$ , то триггер ТФ обнуляется, переменная Fetch=1 и выборка микрокоманд прекращается.

Как видно из рис. 1, сигнал Start служит для установки начальных значений (Fetch=1, СТ=RG=0). Сигнал Clock используется для переключения элементов памяти (СТ и RG).

Очевидно, КМУУ  $U_1$  является автоматом Мура, состояния которого представлены функциями  $\tau_r \in \tau$ . Не-

достатком этого устройства является значительное число термов  $F_h \in F$ , реализуемых матрицей  $M_1$ . Для уменьшения этого параметра можно ввести преобразователь кодов (ПК) [6]. Этот подход позволяет уменьшить число термов в системе (5)–(6) до некоторой величины  $H_0$ , определяемой характеристиками эквивалентного автомата Мили. Однако ПК потребляет некоторые ресурсы кристалла. В настоящей статье предлагается подход, позволяющий гарантировано уменьшить параметр  $H$  до  $H_0$  и уменьшить площадь, занимаемую схемой ПК.

### ОСНОВНАЯ ИДЕЯ ПРЕДЛАГАЕМОГО МЕТОДА

Напомним, что ОЛЦ  $\alpha_i, \alpha_j \in C$  являются псевдоэквивалентными (ПОЛЦ), если их выходы связаны с входом одной и той же вершиной ГСА  $\Gamma$  [2]. Пусть  $\Pi_C = \{B_1, \dots, B_I\}$  разбиение множества ОЛЦ  $C_1$  на классы ПОЛЦ. При этом ОЛЦ  $\alpha_g \in C_1$ , если ее выход не связан с вершиной  $b_E$ . Выполним кодирование ОЛЦ  $\alpha_g \in C$  так, чтобы максимально возможное число классов  $B_i \in \Pi_C$  входило в один обобщенный интервал  $R_0$ -мерного булева пространства.

Теперь множество  $\Pi_C$  может быть представлено в виде  $\Pi_C^1 \cup \Pi_C^2$ . Пусть  $B_i \in \Pi_C^1$ , если класс  $B_i \in \Pi_C$  представляется одним обобщенным интервалом  $R_0$ -мерного булева пространства. В противном случае  $B_i \in \Pi_C^2$ . Очевидно,  $\Pi_C^1 \cap \Pi_C^2 = \emptyset$  и  $\Pi_C^1 \cup \Pi_C^2 = \Pi_C$ . Закодируем классы  $B_i \in \Pi_C^2$  двоичными кодами  $K(B_i)$  разрядности

$$R_2 = \lceil \log_2 G_2 \rceil, \tag{10}$$

где  $G_2 = |\Pi_C^2| + 1$ . Используем для кодирования классов  $B_i \in \Pi_C^2$  переменные  $z_r \in Z$ , где  $|Z| = R_2$ .

Исходная ГСА  $\Gamma$  служит для нахождения системы обобщенных формул переходов [2]. Разделим эту систему  $S$  на две подсистемы:  $S = S_1 \cup S_2$ . Пусть подсистема  $S_1$  задает переходы для классов  $B_i \in \Pi_C^1$ , а подсистема  $S_2$  – для классов  $B_i \in \Pi_C^2$ . В этом случае для реализации схемы КМУУ на заказных матрицах предлагается модель  $U_2$  (рис. 2).

В КМУУ  $U_2$  блок БАМ представлен матрицами  $M_1^1$ ,  $M_1^2$  и  $M_2$ . Матрица  $M_1^1$  реализует термы  $F_h \in F^1$ , входящие в подсистему формул перехода  $S_1$ . Термы  $F_h \in F^1$  задаются формулами:

$$F_h = \left( \bigwedge_{r=1}^{R_0} \tau_r^{l_{gr}} \right) \cdot X_h (h = \overline{1, H_1}). \tag{11}$$

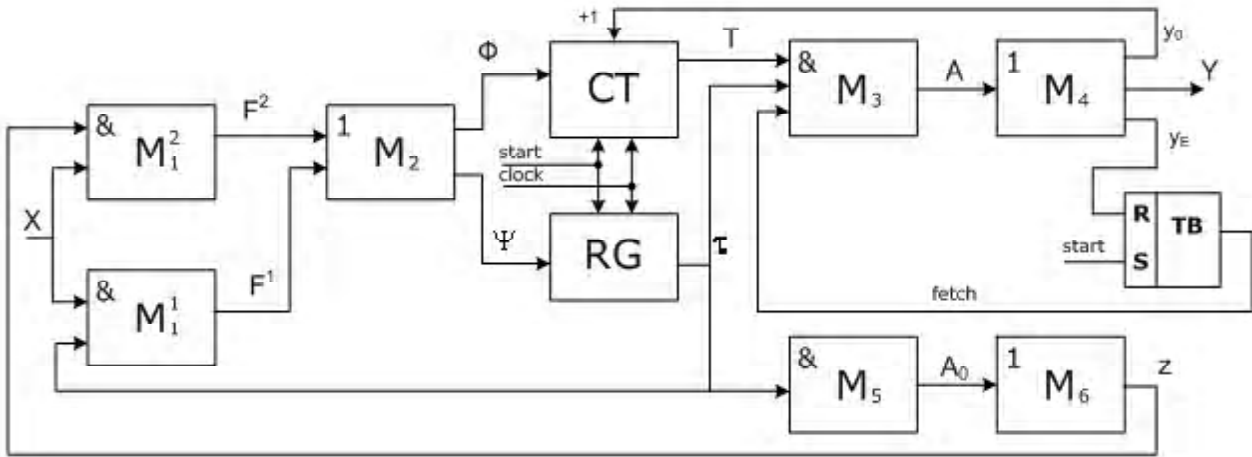


Рис. 2. Структурная схема КМУУ

В формуле (11) конъюнкция  $X_h$  соответствует части формул переходов, содержащей логические условия  $x_l \in X$ . Отметим, что  $l_{gr} \in \{0,1,*\}$  и  $\tau_r^*$ . Это связано с тем, что классы  $B_i \in \Pi_C^1$  представляются обобщенными интервалами, то есть некоторые кодирующие переменные имеют неопределенные значения.

Матрица  $M_1^2$  реализует термы  $F_h \in F^2$ , входящие в подсистему формул  $S_2$ . Термы  $F_h \in F^2$  задаются формулами:

$$F_h = \left( \bigwedge_{r=1}^{R_2} z_r^{l_{ir}} \right) \cdot X_h \quad (h = \overline{1, H_2}). \quad (12)$$

В формуле (12)  $l_{ir} \in \{0,1,*\}$  – значение  $r$ -го разряда кода  $K(B_i)$  класса  $B_i \in \Pi_C^2$ . При этом неопределенное значение \* может появиться, если число возможных кодов больше числа классов  $B_i \in \Pi_C^1$ . Очевидно,  $z_r^0 = \overline{z_r}$ ,  $z_r^1 = z_r$  и  $z_r^* = 1$  ( $r = \overline{1, R_2}$ ). Кроме того, выполняется равенство  $H_0 = H_1 + H_2$ .

Матрица  $M_2$  реализует функции (7), однако теперь они зависят от термов  $F_h \in F^1 \cup F^2$ . Матрицы  $M_3$  и  $M_4$  реализуют управляющую память; они полностью идентичны соответствующим матрицам КМУУ  $U_1$ . Это же справедливо и для триггера ТФ. Матрицы  $M_5$  и  $M_6$  реализуют преобразователь кодов ПК. Матрица  $M_5$  реализует термы  $A_i \in A_0$ , входящие в систему функций

$$Z = Z(\tau). \quad (13)$$

Термы  $A_i$  определяются выражением

$$A_i = \bigwedge_{r=1}^{R_0} \tau_r^{l_{ir}} \quad (i = \overline{1, M_0}), \quad (14)$$

где  $l_{ir} \in \{0,1,*\}$  – значение  $r$ -го разряда кода класса  $B_i \in \Pi_C^2$ ,  $\tau_r^0 = \overline{\tau_r}$ ,  $\tau_r^1 = \tau_r$  и  $\tau_r^* = 1$  ( $r = \overline{1, R_0}$ ). Параметр  $M_0$  определяют в ходе синтеза КМУУ  $U_2$ ; он зависит от результата кодирования ОЛЦ  $\alpha_g \in C$ . Матрица  $M_6$  формирует функции (13), как некоторые дизъюнкции от термов (14):

$$z_r = \bigvee_{i=1}^{M_0} C_{ri} A_i \quad (r = \overline{1, R_0}). \quad (15)$$

В формуле (15) булева переменная  $C_{ri} = 1$ , если и только если функция  $z_r$  зависит от термина  $A_i$ .

Принцип функционирования КМУУ  $U_2$  очевиден. Разница между КМУУ  $U_1$  и  $U_2$  заключается в следующем:

1. Используются два источника кодов классов псевдоэквивалентных ОЛЦ. Один источник – регистр RG, второй – блок преобразователя кодов ОЛЦ в коды классов ПОЛЦ.

2. Число термов в схеме БАМ гарантировано равняется этому параметру эквивалентного автомата Мили. При этом площадь, занимаемая блоком ПК, уменьшается по сравнению с подходом, основанным на преобразовании кодов всех ОЛЦ  $\alpha_g \in C_1$ .

Отметим, что КМУУ с разделением кодов реализуется только при выполнении условия

$$R_0 + R_C = R. \quad (16)$$

**МЕТОД СИНТЕЗА И РАЗВИТИЕ  
ОСНОВНОЙ ИДЕИ**

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

1. Формирование множества ОЛЦ  $C$  и  $C_1$  по исходной граф-схеме алгоритма  $\Gamma$ .
2. Определение параметров  $R, R_0$  и  $R_C$ . Если условие (16) выполняется, то процесс синтеза продолжается.
3. Формирование разбиения  $\Pi_C$  множества ОЛЦ  $C_1$  на классы псевдоэквивалентных ОЛЦ.
4. Кодирование ОЛЦ  $\alpha_g \in C$  и их компонент.
5. Формирование содержимого управляющей памяти.
6. Формирование множеств  $\Pi_C^1$  и  $\Pi_C^2$ . Кодирование классов  $B_i \in \Pi_C^2$ .
7. Формирование системы обобщенных формул переходов  $S$  и разбиение ее на подсистемы  $S_1$  и  $S_2$ .
8. Формирование таблицы переходов для классов  $B_i \in \Pi_C^1$  по системе  $S_1$ . Построение системы функций  $\Phi = \Phi(\tau, X)$  и системы  $\Psi = \Psi(\tau, X)$ .
9. Формирование таблицы переходов для классов  $B_i \in \Pi_C^2$  по системе  $S_2$ . Построение системы функций  $\Phi = \Phi(Z, X)$  и системы  $\Psi = \Psi(Z, X)$ .
10. Построение таблицы преобразователя кодов и формирование системы функций  $Z = Z(\tau)$ .
11. Реализация схемы КМУУ на заказных матрицах по полученным системам функций.

Детализируем некоторые этапы синтеза, связанные с построением таблиц переходов. Пусть система  $S_1$  включает формулу перехода:

$$B_3 \rightarrow x_1x_2b_3 \vee x_1x_2b_8 \vee x_1b_{12}. \tag{17}$$

Пусть  $K(B_3) = 011, A(B_3) = 10010, A(B_8) = 11001$  и  $A(B_{12}) = 11100$ . Очевидно,  $R_0 = 2, T = \{T_1, T_2\}, R_C = 3, \tau = \{\tau_1, \tau_2, \tau_3\}$ . Таблица переходов для классов  $B_i \in \Pi_C^1$  имеет следующие столбцы:  $B_i$  – класс ПОЛЦ;  $K(B_i)$  – код класса  $B_i \in \Pi_C^1$ ;  $A(b_q)$  – адрес входа некоторой ОЛЦ  $\alpha_g \in C$ ;  $X_h$  – входной сигнал, определяющий переход из входов ОЛЦ  $\alpha_g \in B_i$  в вершину  $b_q$ ;  $\Psi_h$  – функция возбуждения триггеров регистра RG, принимающие единичное значение на переходе  $\langle B_i, b_q \rangle$ ;  $\Phi_h$  – функция возбуждения триггеров счетчика СТ, принимающие единичное значение на переходе  $\langle B_i, b_q \rangle$ ;  $h = \overline{1, H_1}$  – номер перехода. Для формулы (17) этот фрагмент таблицы имеет следующий вид (табл. 1).

Как следует из табл. 1,  $\Psi = \{D_1, D_2, D_3\}$  и  $\Phi = \{D_4, D_5\}$ . Из табл. 1 имеем, например, следующие функции:

$$\begin{aligned} D_2 &= \overline{\tau_1\tau_2\tau_3x_1x_2} \vee \overline{\tau_1\tau_2\tau_3x_1}; \\ D_4 &= \overline{\tau_1\tau_2\tau_3x_1x_2}. \end{aligned} \tag{18}$$

Термы системы (18) реализуются на матрице  $M_1^1$ , а сами функции – на матрице  $M_2$ .

Пусть система  $S_2$  включает формулу перехода

$$B_5 \rightarrow x_2b_{19} \vee x_3b_{26}. \tag{19}$$

Пусть  $K(B_5) = 01, A(B_{19}) = 00101, A(B_{26}) = 01100$ , то есть  $R_2 = 2, Z = \{z_1, z_2\}$ . Таблица переходов для системы  $S_2$  имеет такой же вид, как и таблица переходов для системы  $S_1$  (табл. 2).

Из табл. 2 можно, например, получить функции

$$D_2 = \overline{z_1z_2x_3}; D_5 = \overline{z_1z_2x_3}. \tag{20}$$

Термы системы (20) реализуются на матрице  $M_1^2$ , а сами функции – на матрице  $M_2$ .

Пусть  $\Pi_C^2 = \{B_2, B_5, B_6\}, K(B_2) = 01, K(B_5) = 10$  и  $K(B_6) = 11$ . При этом код 00 является признаком того, что переходы происходят из классов  $B_i \in \Pi_C^1$ . Этому факту соответствует переменная  $y_{C1}$ ; в данном случае  $y_{C1} = \overline{z_1z_2}$ . Пусть  $B_2 = \{\alpha_2, \alpha_3\}, B_5 = \{\alpha_4, \alpha_5\}$  и  $B_6 = \{\alpha_6, \alpha_7\}$ , а ОЛЦ имеют коды, соответствующие двоичным эквивалентам их индексов. Таблица преобразователя кодов имеет следующие столбцы:  $\alpha_g, K(\alpha_g), B_i, K(B_i), Z_i, i$ . Здесь столбец  $Z_i$  содержит перемен-

**Таблица 1.** Фрагмент таблицы переходов для формулы (17)

$B_i$	$K(B_i)$	$A(b_q)$	$X_h$	$\Psi_h$	$\Phi_h$	$h$
$B_3$	011	10010	$x_1x_2$	$D_1$	$D_4$	1
		11001	$x_1\overline{x_2}$	$D_1D_2$	$D_5$	2
		11100	$\overline{x_1}$	$D_1D_2D_3$	–	3

**Таблица 2.** Фрагмент таблицы переходов для формулы (19)

$B_i$	$K(B_i)$	$A(b_q)$	$X_h$	$\Psi_h$	$\Phi_h$	$h$
$B_5$	01	00101	$x_3$	$D_3$	$D_5$	1
		01199	$\overline{x_3}$	$D_2D_3$	–	2

ные  $z_r$ , равные единице в коде  $K(B_i)$ . Для нашего примера таблица ПК представлена в табл. 3.

Из табл. 3 с учетом минимизации имеем:  $z_1 = \tau_1$ ,  $z_2 = \tau_2$ . При этом матрица  $M_5$  в схеме КМУУ отсутствует. Однако, это частный случай. В общем случае система термов, полученных из таблицы ПК, реализуется на матрице  $M_5$ , а функции  $z_r \in Z$  – на матрице  $M_6$ .

Отметим, что термы  $F_h \in F^1$  могут зависеть только от логических условий  $x_l \in X^1$ , где  $X^1 \subset X$ . Аналогично, термы  $F_h \in F^2$  могут зависеть только от логических условий  $x_l \in X^2$ , где  $X^2 \subset X$ . В лучшем случае имеем равенство  $X^1 \cap X^2 = \emptyset$ . Это позволяет уточнить часть схемы КМУУ  $U_2$ , реализующей функции  $\Psi$  и  $\Phi$  (рис. 3).

Отметим, что наличие переменной  $y_{c1}$  не вносит задержку во время такта КМУУ  $U_2$  по сравнению с КМУУ  $U_1$ . Это связано с тем, что значение переменной  $y_{c1}$  формируется до прихода правильных значений логических условий. Итак, площади матриц  $M_1^1$ ,  $M_1^2$  и  $M_2$  могут быть найдены следующим образом:

$$\begin{aligned} S(M_1^1) &= 2(L_1 + R_C + 1) \cdot H_1; \\ S(M_1^2) &= 2(L_2 + R_0) \cdot (H_2 + 1); \\ S(M_2) &= H_0(R_0 + R_C). \end{aligned} \tag{21}$$

Таблица 3. Таблица ПК КМУУ  $U_2$

$\alpha_g$	$K(\alpha_g)$	$B_i$	$K(B_i)$	$Z_i$	$i$
$\alpha_2$	010	$B_2$	01	$z_2$	1
$\alpha_3$	011	$B_2$	01	$z_2$	2
$\alpha_4$	100	$B_5$	10	$z_1$	3
$\alpha_5$	101	$B_5$	10	$z_1$	4
$\alpha_6$	110	$B_6$	11	$z_1 z_2$	5
$\alpha_7$	111	$B_6$	11	$z_1 z_2$	6

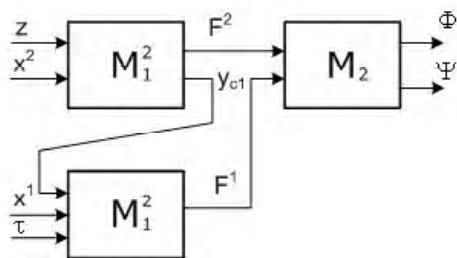


Рис. 3. Уточненная часть схемы КМУУ  $U_2$

В системе (21) имеем  $L_1 = |X^1|$ ,  $L_2 = |X^2|$ ,  $|F^1| = H_1$ ,  $|F^2| = H_2$ ,  $H_1 + H_2 = H_0$ .

**ЗАКЛЮЧЕНИЕ**

В работе предложен метод, позволяющий уменьшить сложность матричной реализации КМУУ с разделением кодов. Этот метод основан на использовании двух источников кодов классов псевдоэквивалентных ОЛЦ. При этом матрица термов схемы адресации КМУУ разбивается на две части.

Такой подход гарантирует уменьшение числа термов в схеме адресации до величины, определяемой эквивалентным автоматом Мили. Выбор между КМУУ  $U_1$  и  $U_2$  может быть выполнен аналитическим путем, то есть без реализации схемы устройства. Отметим, что предложенный метод применим только для линейных граф-схем алгоритма при выполнении условия (16). Если условие (16) не выполняется, то это приводит к резкому увеличению площадей матриц  $M_3$  и  $M_4$  по сравнению с КМУУ с общей памятью [2].

Научная новизна статьи заключается в усовершенствовании метода синтеза КМУУ с разделением кодов. Внесенные изменения позволяют использовать несколько источников кодов псевдоэквивалентных ОЛЦ, что приводит к уменьшению аппаратных затрат в схеме адресации.

Практическая значимость результатов работы заключается в уменьшении числа базовых элементов, которые необходимы для реализации схемы КМУУ на кристалле. Это позволяет удешевить реализацию сложных цифровых систем, в состав которых входят устройства управление, реализованный по модели КМУУ.

**СПИСОК ЛИТЕРАТУРЫ**

1. Baranov Samary. Logic Synthesis for Control Automata. / Samary Baranov. – Kluwer Academic Publishers, 1994. – 312 p.
2. Barkalov, A. Logic Synthesis for Compositional Microprogram Control Units. / A. Barkalov, L. Titarenko. – Berlin : Springer, 2008. – 273 p.
3. Smith, Michael. Application-Specific Integrated Circuits / Michael Smith. – Boston : Addison-Wesley, 1997. – 1040 p.
4. Баркалов, А. А. Синтез микропрограммных автоматов на заказных и программируемых СБИС. / А. А. Баркалов, Л. А. Титаренко. – Донецк : УНИТЕХ, 2009. – 336 с.
5. Bayrek Hathot Approach to Realization of Compositional Microprogramming Control Unit with Code Converter on Custom-Made Matrix. / A. Barkalov, I. Zelenyova, A. Miroshkin, H. Bayrek // Наукові праці Донецького національного технічного університету. Серія «Інформатика, кібернетика та обчислювальна техніка». Випуск 11(164). – Донецьк : ДонНТУ, 2010. – С. 71–74.
6. Barkalov, A. Basic principles of logic design. / A. Barkalov, L. Titarenko. – Zielona Gora : University of Zielona Gora Press, 2010. – 295 p.

Стаття надійшла до редакції 03.05.2011.

Баркалов О. О., Зеленьова І. Я., Цололо С. О., Біайрак Х.

#### ЗМЕНШЕННЯ ПЛОЩІ МАТРИЧНОЇ СХЕМИ ПРИБРОЮ КЕРУВАННЯ З РОЗДІЛЕННЯМ КОДІВ

У статті запропонована модель композиційного мікропрограми пристрою керування з розділенням кодів, яка орієнтована на реалізацію схеми пристрою у базисі замовних матриць. Запропонована модель використовує представлення адреси вершини алгоритму керування у вигляді конкатенації кодів ОЛЛ та коду компоненти ОЛЛ. Такий підхід зменшує число входів і виходів схеми формування функцій збудження.

**Ключові слова:** композиційний пристрій керування, матрична схема, операційний лінійний ланцюг (олл), розділення кодів.

Barkalov A. A., Zelenyova I. J., Tsololo S. A., Biayarek H.

#### REDUCTION OF TERM MATRIX OF CONTROL UNIT WITH CODE SHARING

The structures of compositional microprogram control unit with code sharing are proposed. Structures allow reducing the complexity of the matrix realization in the device's circuit. The proposed method is based on using of node's address representation as a concatenation of OLC codes and code of OLC components. The proposed method allow reducing the complexity of the matrix realization in the device's circuit.

**Key words:** compositional control unit, matrix circuit, operator linear chain (olc), code shareing.

УДК 378.14:004.421

Пищукина О. А.<sup>1</sup>, Клочок А. Ю.<sup>2</sup>

<sup>1</sup>Канд. техн. наук, доцент Национального аерокосмического университета им. М. Е. Жуковского «ХАИ»

<sup>2</sup>Асистент Национального аерокосмического университета им. М. Е. Жуковского «ХАИ»

## ПОДХОД К ФОРМИРОВАНИЮ ОБРАТНОЙ СВЯЗИ В ИНТЕЛЛЕКТУАЛЬНЫХ ОБУЧАЮЩИХ СИСТЕМАХ В СФЕРЕ ВЫСШЕГО ТЕХНИЧЕСКОГО ОБРАЗОВАНИЯ

Предложен подход к формированию обратной связи процесса обучения с использованием интеллектуальных компьютерных систем, особенностью которого является поэтапный контроль процесса обучения, определение места возникновения ошибок, в отдельных случаях – распознавание причин их возникновения и формирование рекомендаций по их устранению.

**Ключевые слова:** интеллектуальные компьютерные обучающие системы, высшее техническое образование, обратная связь.

### ВВЕДЕНИЕ

Сокращение часов аудиторной нагрузки и увеличение объемов самостоятельной работы студентов представляет собой одну из особенностей процесса обучения в современном вузе, ориентированном на западные стандарты образования. Основную сложность для студентов технических специальностей вызывает усвоение постоянно растущего объема информации и расширения круга задач, умение решать которые необходимо для обеспечения компетентности и конкурентоспособности выпускников технических вузов на современном рынке труда. Вместе с тем, следует учитывать ментальные отличия украинских студентов, связанные со слабой мотивацией и (или) неумением работать самостоятельно, что способствует недостаточному усвоению учебного материала, и как следствие, снижению общего образовательного уровня. Необходимо отметить также недостаточную подготовку учащихся средних школ по базовым техническим дисциплинам для эффективного обучения в высших учебных заведениях.

Эффективным путем разрешения данного противоречия, связанного с необходимостью активного обучения студентов, ограничением энергетических ресурсов преподавателя, фиксированным количеством часов практических и лабораторных занятий согласно разработанным учебным и рабочим программам вузов, является использование компьютерных средств для организации самостоятельной работы студентов, а именно – интеллектуальных обучающих систем [1, 2].

Характерной чертой обучения студентов технических специальностей является необходимость формирования у них конкретных навыков и умений, направленных на практическое решение примеров и задач, требующих конечного ответа, чаще всего в виде числа или математического выражения. Эта особенность накладывает специфические требования к структуре и наполнению интеллектуальной обучающей системы, связанные с многократным повторением и решением предложенных задач, и требует включения в ее состав следующих итерационных блоков [3, 4]: 1) информационного блока, содержащего конкретные рекомендации по способам