

УДК 004.3

А.А. Баркалов<sup>1</sup>, И.Я. Зеленева<sup>2</sup>, Хатгут Байрек<sup>2</sup>, Е.Р. Татолов<sup>2</sup><sup>1</sup> Зеленогурский университет, г. Зеленая гора, Польша

A.Barkalov@iie.uz.zgora.pl

<sup>2</sup> Донецкий национальный технический университет, г. Донецк, Украина

irina@cs.dgtu.donetsk.ua, allah\_karem90@yahoo.com, tatolov@bk.ru

## Матричная реализация композиционного микропрограммного устройства управления с общей памятью

*Впервые предложен метод реализации композиционного микропрограммного устройства управления в базисе заказных интегральных схем (ASIC). Приведены теоретические сведения и практические аспекты синтеза; показан пример применения метода*

*Композиционное микропрограммное устройство управления, операторная линейная цепь, логическая схема, линейный алгоритм управления, общая память, заказная интегральная схема, ASIC*

### Введение

Одним из методов реализации устройства управления цифровой системы является использование модели композиционного микропрограммного устройства управления (КМУУ) [1, 2]. Эти устройства идеально подходят для использования в современных СБИС типа FPGA [3], где имеются средства для реализации произвольной логики и встроенные блоки памяти [4, 5]. В тоже время при массовом производстве изделий электронной промышленности по-прежнему широко используются заказные схемы типа ASIC (Application Specific Integrated Circuits) [6]. В этом случае для реализации схем устройств управления используются заказные матрицы [7, 8, 9], основанные на идее распределенной логики [10]. В настоящее время в литературе практически отсутствуют методы, ориентированные на этот базис. В данной статье рассматривается метод синтеза КМУУ с общей памятью на заказных матрицах и анализируются пути уменьшения площади кристалла, занимаемой его логической схемой.

Целью исследования является реализация логической схемы устройства управления на заказных СБИС при интерпретации линейного алгоритма управления.

Задачей исследования является разработка метода синтеза КМУУ с общей памятью, позволяющего уменьшить площадь заказных матриц в его логической схеме. При этом алгоритм управления представляется в виде граф-схемы алгоритма (ГСА) [7].

### Основные определения и общие положения

Пусть исходная ГСА  $\Gamma$  имеет начальную вершину  $b_0$ , конечную вершину  $b_E$ , операторные вершины из множества  $B_1$  и условные вершины из множества  $B_2$ . В вершинах  $b_q \in B_1$  содержатся микрокоманды  $Y(b_q) \subseteq Y$ , где  $Y = \{y_1, \dots, y_N\}$  – множество микроопераций. В вершинах  $b_q \in B_2$  записаны элементы множества логических условий  $X = \{x_1, \dots, x_L\}$ . Вершины ГСА  $\Gamma$  связаны дугами  $\langle b_t, b_q \rangle$ , образующими множество дуг  $E$ .

Введем ряд определений [1, 2], необходимых для дальнейшего изложения материала.

Определение: Операторной линейной цепью (ОЛЦ) ГСА  $\Gamma$  называется конечная последовательность операторных вершин

$\alpha_g = \langle b_{g_1}, \dots, b_{g_{F_g}} \rangle$ , такая, что для любой пары

соседних компонент вектора  $\alpha_g$  существует дуга

$\langle b_{g_i}, b_{g_{i+1}} \rangle \in E$ .

Определение: Входом ОЛЦ  $\alpha_g$  называется вершина  $b_q \in D^g$ , где  $D^g$  – множество вершин, входящих в ОЛЦ  $\alpha_g$ , если существует дуга

$\langle b_t, b_q \rangle \in E$ , такая, что  $b_t = b_0$  или  $b_t \in B_2$  или  $b_t \notin D^g$ .

**Определение:** Выходом ОЛЦ  $\alpha_g$  называется вершина  $b_q \in D^g$ , такая, что существует дуга  $\langle b_t, b_q \rangle \in E$ , где  $b_t = b_E$  или  $b_t \in B_2$  или  $b_t \notin D^g$ .

Произвольная ОЛЦ  $\alpha_g$  может иметь более одного входа; обозначим  $j$ -й вход ОЛЦ  $\alpha_g$  как  $I_g^j$ . Каждая ОЛЦ имеет только один выход  $O_g$ , входящий в множество  $O(G)$  выходов ОЛЦ ГСА Г.

Пусть для ГСА Г сформировано множество ОЛЦ  $C = \{\alpha_1, \dots, \alpha_G\}$ , удовлетворяющее условию:

$$\begin{cases} D^i \cap D^j = \emptyset & (i \neq j; i, j \in \{1, \dots, G\}) \\ D^1 \cup D^2 \cup \dots \cup D^G = B_1 \\ G \Rightarrow \min. \end{cases} \quad (1)$$

При выполнении условия (1) каждая вершина  $b_q \in B_1$  входит точно в одну ОЛЦ  $\alpha_g \in C$ , число которых минимально.

Пусть  $A(b_q)$  – двоичный адрес, соответствующий вершине  $b_q \in B_1$  и имеющий

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

разрядов, где  $M = |B_1|$ . Выполним адресацию так, чтобы выполнялось условие

$$A(b_{g_{i+1}}) = A(b_{g_i}) + 1, \quad (3)$$

где  $i \in \{1, \dots, F_g - 1\}$ ,  $g \in \{1, \dots, G\}$ . Такой режим соответствует естественной адресации микрокоманд [2], сопоставленных операторным вершинам ГСА Г. Назовем ГСА Г линейной ГСА, если выполняется условие

$$\frac{M}{G} \geq 2, \quad (4)$$

т. е. если количество операторных вершин хотя бы в два раза превышает количество ОЛЦ.

В этом случае для интерпретации ГСА Г можно использовать модель КМУУ с общей памятью [2], обозначаемую в дальнейшем символом  $U_1$  (рис. 1).

Это устройство функционирует следующим образом. По сигналу Start в счетчик СТ загружается нулевой адрес первой микрокоманды микропрограммы, соответствующей ГСА Г. Одновременно триггер

выборки TF устанавливается в единичное состояние (Fetch=1). Очередная микрокоманда (МК) выбирается из управляющей памяти СМ и, если эта МК соответствует вершине  $b_q \neq O_g$ , то одновременно с набором микроопераций  $Y(b_q)$  формируется переменная  $y_0 = 1$ . Если  $y_0 = 1$ , то содержимое СТ увеличивается на 1 по сигналу Clock. Это соответствует режиму (3), т. е. переход происходит внутри некоторой ОЛЦ  $\alpha_g \in C$ . Если  $b_q = O_g$ , то переменная  $y_0 = 0$ . В этом случае комбинационная схема СС формирует функции возбуждения СТ

$$\Phi = \Phi(T, X), \quad (5)$$

необходимые для загрузки в СТ адреса входа некоторой ОЛЦ. Адрес записывается в СТ по сигналу Clock. Если  $\langle b_q, b_E \rangle \in E$ , где  $b_E$  – конечная вершина ГСА Г, то формируется переменная  $y_E = 1$ . Если  $y_E = 1$ , то триггер TF устанавливается в нулевое состояние (Fetch=0). В этом случае выборка микрокоманд из СМ прекращается и КМУУ  $U_1$  завершает свою работу.

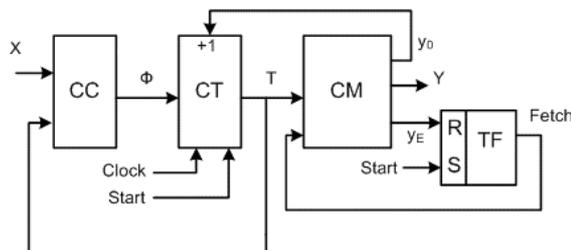


Рисунок 1 – Структурная схема КМУУ с общей памятью

Очевидно, КМУУ  $U_1$  является автоматом Мура, так как выходные сигналы  $y_0, y_E$  и  $Y$  зависят только от содержимого СТ. При этом адрес микрокоманды может рассматриваться как код состояния автомата. Однако, в отличие от классического автомата Мура, память КМУУ  $U_1$  реализуется на счетчике. В этом случае ОЛЦ  $\alpha_g \in C$  являются аналогами состояний автомата.

**Матричная реализация КМУУ с общей памятью**

На рис. 2 показана матричная реализация КМУУ с общей памятью.

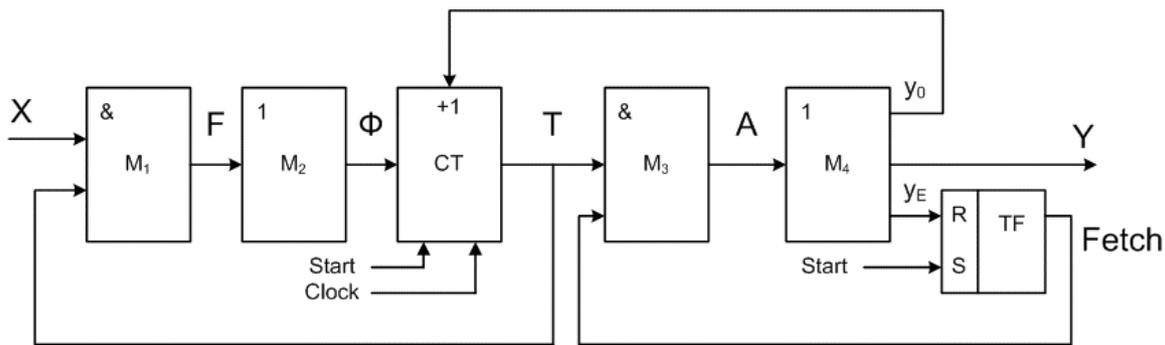


Рисунок 2 – Матричная реализация КМУУ с общей памятью

При реализации схемы КМУУ  $U_1$  на заказных матрицах, предлагаемой в данной работе, схема СС представляется в виде конъюнктивной матрицы  $M_1$  и дизъюнктивной матрицы  $M_2$ . Аналогично, управляющая память СМ отображается в виде конъюнктивной матрицы  $M_3$  и дизъюнктивной матрицы  $M_4$ .

В этой схеме матрица  $M_1$  реализует систему термов  $F = \{F_1, \dots, F_H\}$ , соответствующих строкам таблицы переходов КМУУ. При этом терм  $F_h$  представляется в следующем виде:

$$F_h = \left( \bigcap_{r=1}^{R_A} T_r^{e_{rh}} \right) \cdot X_h, \quad (h = 1, \dots, H). \quad (6)$$

Первый член в формуле (6) соответствует адресу  $A(O_g)$  выхода ОЛЦ  $\alpha_g \in C$ , переход из которого рассматривается в первой строке таблицы. При этом  $e_{rh} \in \{0, 1\}$  – значение  $r$ -го разряда адреса  $A(O_g)$ ;  $T_r^0 = \bar{T}_r$ ,  $T_r^1 = T_r$ ,  $r = (1, \dots, R_A)$ .

Естественно, что переменные  $T_r \in T$ , где  $|T| = R_A$ , используются для адресации микрокоманд.

Матрица  $M_2$  реализует систему (5), элементы которой определяют функции возбуждения триггеров D-типа, поступающие на информационные входы СТ:

$$D_r = \bigcup_{h=1}^H C_{rh} \cdot F_h, \quad (r = 1, \dots, R_A). \quad (7)$$

В выражении (7) переменная  $C_{rh} = 1$ , если и только если функция  $D_r$  записана в  $h$ -й строке таблицы переходов ( $h = 1, \dots, H$ ).

Матрица  $M_3$  формирует термы  $A_m \in A$ , соответствующие адресам микрокоманд:

$$A_m = \left( \bigcap_{r=1}^{R_A} T_r^{e_{mr}} \right) \cdot y_E, \quad (m = 1, \dots, M). \quad (8)$$

Здесь  $e_{mr} \in \{0, 1\}$  – значение  $r$ -го разряда адреса  $A(b_m)$ ;  $T_r^0 = \bar{T}_r$ ,  $T_r^1 = T_r$ ,  $r = (1, \dots, R_A)$ .

Матрица  $M_4$  формирует функции  $y_0, y_E$  и  $y_n \in Y$ , как следующие дизъюнкции:

$$\begin{aligned} y_0 &= \bigcup_{m=1}^M C_{0m} \cdot A_m; \\ y_E &= \bigcup_{m=1}^M C_{Em} \cdot A_m; \\ y_n &= \bigcup_{m=1}^M C_{nm} \cdot A_m; \quad (n = 1, \dots, N). \end{aligned} \quad (9)$$

Буква-переменная  $C_{im} = 1$ , если и только если при выполнении микрокоманды, соответствующей вершине  $b_m \in B_1$ , формируется функция  $y_i$  ( $i = 0, E, 1, \dots, N$ ).

В настоящей работе предлагается метод матричной реализации КМУУ  $U_1$ , включающий следующие этапы.

1. Формирование множества ОЛЦ С. Этот этап выполняется по известной методике [1,2].

2. Естественная адресация микрокоманд. Данный этап сводится к последовательной нумерации компонент кортежа  $\alpha = \alpha_1 * \alpha_2 * \dots * \alpha_G$ , где \* – знак конкатенации. Нумерация начинается с нуля, а  $R_A$  – разрядные двоичные эквиваленты номеров операторных вершин, являются адресами соответствующих микрокоманд [1, 2].

3. Формирование системы формул перехода. Эта система формируется для выходов ОЛЦ  $\alpha_g \in C_1$ , где  $\alpha_g \notin C_1$ , если ее выход связан

с конечной вершиной ГСА  $\Gamma$ . Система формул перехода содержит  $G_1 = |C_1|$  формулы, каждая из которых имеет следующий вид:

$$O_g \rightarrow \bigcup_{m=1}^M C_{g_i b_i} \cdot X_{g_i} \quad (g = 1, \dots, G). \quad (10)$$

В формуле (10) булева переменная  $C_{g_i b_i} = 1$ , если и только если существует переход из вершины  $b_q = O_g$  в вершину  $b_i \in I(\Gamma)$ ;  $X_{g_i}$  – конъюнкция некоторых элементов (или их отрицаний)  $x_e \in X$ , определяющая переход из  $b_q = O_g$  в  $b_i \in I(\Gamma)$ . Количество термов в системе (10) равняется количеству строк таблицы переходов, а  $X_{g_i} = X_h$  для терма номер  $h$ .

4. Формирование таблицы переходов КМУУ. Эта таблица строится по системе (10) и имеет столбцы:  $O_g, A(O_g), b_i, A(b_i), X_h, \Phi_h, h$ . Здесь  $A(O_g)$  – адрес выхода ОЛЦ  $\alpha_g \in C$ ,  $\Phi_h \subseteq \Phi$  – набор функций возбуждения триггеров СТ, принимающих единичное значение для записи в СТ адреса  $A(b_i)$ . Строки этой таблицы соответствуют термам (6), а функции  $D_r \in \Phi_h$  представляются в виде (7).

5. Формирование содержимого управляющей памяти. Эта таблица имеет столбцы  $b_q, A(b_q), y_0, y_E, Y(b_q), q$ , где  $q \in \{1, \dots, M\}$ . Если  $b_q \neq O_g$ , то столбец  $y_0 = 1$  для адреса  $A(b_q)$ . Если  $\langle b_q, b_E \rangle \in E$ , то столбец  $y_E = 1$  для адреса  $A(b_q)$ . Эта таблица является основой для формирования термов (8) и функций (9).

6. Реализация схемы КМУУ. Матрица  $M_1$  реализует термы (6), матрица  $M_2$  – функции (7), матрица  $M_3$  – термы (8), матрица  $M_4$  – функции (9). Соединение этих матриц и счетчика СТ дает схему КМУУ  $U_1$ . Сложность каждой из матриц определяется ее площадью в условных единицах:  $S(M_j) = S_j \cdot t_j$ , где  $S_j$  – количество входов и  $t_j$  – количество выходов матрицы  $M_j$  ( $j = 1, \dots, 4$ ).

Рассмотрим пример синтеза КМУУ  $U_1$  по ГСА  $\Gamma_1$  (рис. 3)

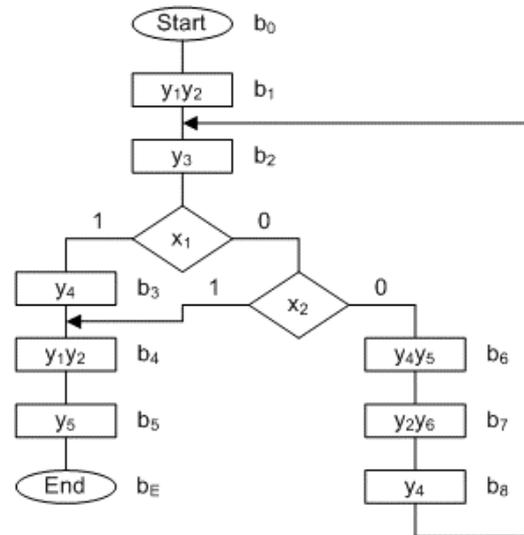


Рисунок 3 – Исходная ГСА  $\Gamma_1$

**Пример матричной реализации КМУУ с общей адресацией**

Используя методы из [2], можно получить множество  $C = \{\alpha_1, \alpha_2, \alpha_3\}$ , где  $\alpha_1 = \langle b_1, b_2 \rangle$ ,  $I_1^1 = b_1, I_1^2 = O_1 = b_2$ ;  $\alpha_2 = \langle b_3, b_4, b_5 \rangle, I_2^1 = b_3, I_2^2 = b_4, O_2 = b_5$ ;  $\alpha_3 = \langle b_6, b_7, b_8 \rangle, I_3^1 = b_6, O_3 = b_8$ .

Итак,  $G = 3, M = 8, R = 3, T = \{T_1, T_2, T_3\}, \Phi = \{D_1, D_2, D_3\}, X = \{x_1, x_2\}, L = 2, Y = \{y_1, \dots, y_6\}, N = 6, I(\Gamma_1) = \{b_1, b_2, b_3, b_4, b_6\}, O(\Gamma_1) = \{b_2, b_5, b_8\}, C_1 = \{\alpha_1, \alpha_3\}$ .

Естественная адресация микрокоманд [1] позволяет получить следующие адреса:  $A(b_1) = 000, A(b_2) = 001, \dots, A(b_8) = 111$ . Формулы перехода (10) строятся для выходов ОЛЦ  $\alpha_g \in C_1$ , т. е. для  $O_1$  и  $O_3$ . Эта система имеет следующий вид:

$$O_1 \rightarrow x_1 b_3 \cup \bar{x}_1 x_2 b_4 \cup \bar{x}_1 \bar{x}_2 b_6; \quad O_3 \rightarrow b_2. \quad (11)$$

Система (11) имеет  $H = 4$  терма, следовательно, таблица переходов КМУУ имеет  $H = 4$  строки (табл. 1).

Здесь символ  $U_1(\Gamma_1)$  означает, что модель КМУУ  $U_1$  используется для интерпретации ГСА  $\Gamma_1$ .

Таблица 1 – Таблица переходов КМУУ  $U_1(\Gamma_1)$

$O_g$	$A(O_g)$	$b_i$	$A(b_i)$	$X_h$	$\Phi_h$	$h$
$O_1$	001	$b_3$	010	$x_1$	$D_2$	1
		$b_4$	011	$\overline{x_1}x_2$	$D_2D_3$	2
		$b_6$	101	$\overline{x_1x_2}$	$D_1D_3$	3
$O_3$	111	$b_2$	001	1	$D_3$	4

Из табл. 1 формируется система (6), имеющая следующий вид для КМУУ  $U_1(\Gamma_1)$ :

$$F_1 = \overline{T_1}\overline{T_2}T_3x_1; \quad F_2 = \overline{T_1}\overline{T_2}T_3\overline{x_1}x_2; \\ F_3 = \overline{T_1}\overline{T_2}T_3\overline{x_1}\overline{x_2}; \quad F_4 = T_1T_2T_3. \quad (12)$$

Эти термы входят в систему (7), имеющую для КМУУ  $U_1(\Gamma_1)$  следующей вид:

$$D_1 = F_3; \quad D_2 = F_1 \cup F_2; \\ D_3 = F_2 \cup F_3 \cup F_4. \quad (13)$$

Содержимое управляющей памяти КМУУ  $U_1(\Gamma_1)$  показано в табл. 2, имеющей  $M = 8$  строк.

Таблица 2 – Содержимое управляющей памяти КМУУ  $U_1(\Gamma_1)$

$b_q$	$A(b_q)$	$y_0$	$y_E$	$Y(b_q)$	$q$
$b_1$	000	1	0	$y_1y_2$	1
$b_2$	001	0	0	$y_3$	2
$b_3$	010	1	0	$y_4$	3
$b_4$	011	1	0	$y_1y_2$	4
$b_5$	100	0	1	$y_5$	5
$b_6$	101	1	0	$y_4y_5$	6
$b_7$	110	1	0	$y_2y_6$	7
$b_8$	111	0	0	$y_4$	8

Из табл. 2 формируются термы (8), имеющие следующий вид для КМУУ  $U_1(\Gamma_1)$ :

$$A_1 = \overline{T_1}\overline{T_2}\overline{T_3}; \quad A_2 = \overline{T_1}\overline{T_2}T_3; \quad A_3 = \overline{T_1}T_2\overline{T_3}; \\ A_4 = \overline{T_1}T_2T_3; \quad A_5 = T_1\overline{T_2}\overline{T_3}; \quad A_6 = T_1\overline{T_2}T_3;$$

$$A_7 = T_1T_2\overline{T_3}; \quad A_8 = T_1T_2T_3. \quad (14)$$

Эти термы образуют функции (9), имеющие следующий вид для КМУУ  $U_1(\Gamma_1)$ :

$$y_0 = A_1 \cup A_3 \cup A_4 \cup A_6 \cup A_7; \quad y_E = A_5; \\ y_1 = A_1 \cup A_4; \quad y_2 = A_1 \cup A_4 \cup A_7; \quad y_3 = A_2; \\ y_4 = A_3 \cup A_6 \cup A_8; \quad y_5 = A_5 \cup A_6; \quad y_6 = A_7. \quad (15)$$

Матричная схема КМУУ  $U_1(\Gamma_1)$  строится по системам (12)-(15) и приведена на рис. 4.

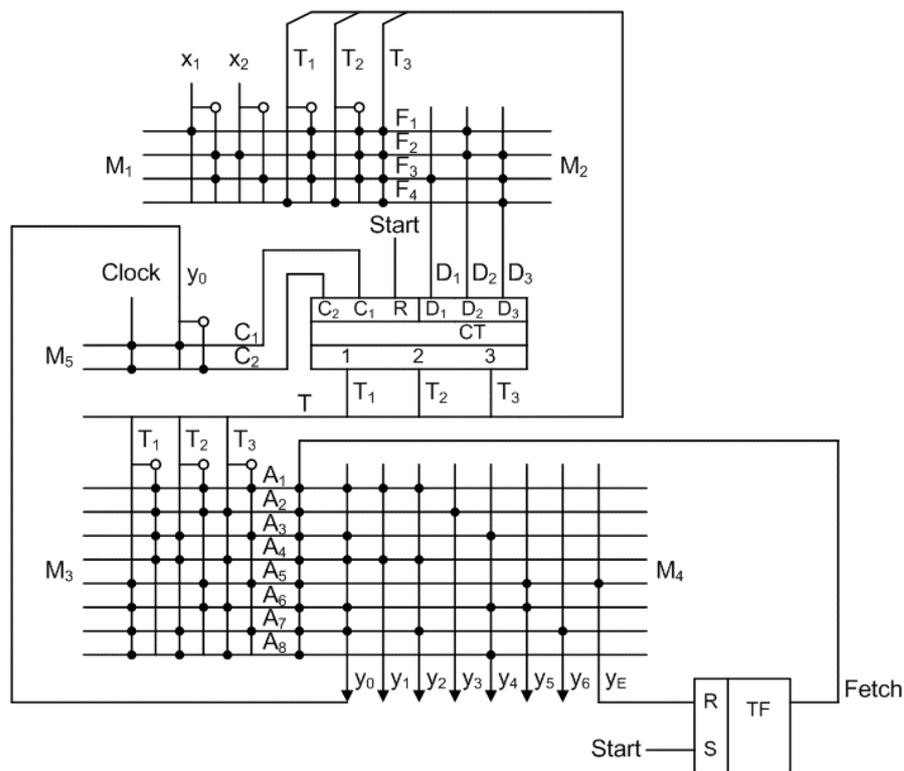
В этой схеме присутствует матрица  $M_5$ , распределяющая синхросигнал Clock между входами  $C_1 \# CT := CT + 1$  и  $C_2 \# CT := \Phi$ . Поскольку содержимое счетчика увеличивается, если  $y_0 = 1$ , а параллельный код принимается, если  $y_0 = 0$ , то матрица  $M_5$  задается следующей системой уравнений:

$$C_1 = y_0 \cdot Clock; \\ C_2 = \overline{y_0} \cdot Clock. \quad (16)$$

На схеме знак "o" означает инвертор, а знак "•" – межсоединение между вертикальными и горизонтальными шинами матриц.

Оценим сложность реализации КМУУ  $U_1(\Gamma_1)$ , используя схему на рис. 4.

Из этой схемы следует, что  $S_1 = 2 \cdot (L + R_A) = 10; \quad t_1 = H = 4; \quad S_2 = H = 4; \\ t_2 = R_A = 3; \quad S_3 = 2 \cdot R_A + 1 = 7; \quad t_3 = M = 8; \\ S_4 = M = 8; \quad O_4 = N + 2 = 8; \quad S_5 = 3; \quad t_5 = 2.$  Итак, площади матриц  $M_1 - M_5$  находятся следующим образом:  $S(M_1) = 40, \quad S(M_2) = 12, \\ S(M_3) = 56, \quad S(M_4) = 64, \quad S(M_5) = 6.$  В целом, схема КМУУ  $U_1(\Gamma_1)$  занимает 178 условных единиц площади (без учета площади, необходимой для счетчика СТ и триггера ТФ). Так как площади матриц не оптимизировались, то схема на рис. 4 является аналогом тривиальной реализации микропрограммного автомата [7].

Рисунок 4 – Матричная реализация КМУУ  $U_1$  ( $\Gamma_1$ )

### Заключення

Предложенный метод матричной реализации КМУУ с общей памятью позволяет получить схему, соответствующую тривиальной реализации микропрограммного автомата. Для оптимизации этой схемы можно использовать как известные методы замены логических условий и кодирования наборов микроопераций [7], так и адаптировать методы оптимизации КМУУ, ориентированные на FPGA [2]. Напомним что эти методы целесообразно использовать для интерпретации линейных

алгоритмов управления, удовлетворяющих условию (4).

Научная новизна работы заключается в том, что впервые были предложены структура и метод синтеза КМУУ на заказных матрицах.

Практическая значимость предложенного метода заключается в уменьшении площади кристалла, занимаемой схемой устройства управления, по сравнению с матричной схемой автомата Мура с регистром, хранящим коды состояний.

Дальнейшее направление исследований связано с адаптацией методов оптимизации схем КМУУ на стандартных СБИС к особенностям матричных схем.

### Список літератури

1. Barkalov A.A. Synthesis of operational and control automata / A.A. Barkalov, L.A. Titarenko. – Donetsk: DonNTU, TechPark DonNTU UNITECH, 2009. – 256 p.
2. Баркалов А.А. Микропрограммное устройство управления как композиция автоматов с программируемой и жесткой логикой / А.А. Баркалов // Автоматика и вычислительная техника. – 1983 – №4 – С. 36-41.
3. Максфилд К. Проектирование на ПЛИС: курс молодого бойца / К. Максфилд. – М.: Издательский дом «Додэка-XXI», 2007. – 408 с.
4. Грушвицкий Р.И. Проектирование систем на микросхемах программируемой логики / Р.И. Грушвицкий, А.Х. Мурсаев, Е.П. Угрюмов. – С-Пб.: БХВ-Петербург, 2002. – 608 с.
5. Соловьев В.В. Проектирование цифровых систем на основе программируемых логических интегральных схем / В.В. Соловьев. – М.: Горячая линия – Телеком, 2001. – 636 с.
6. Smith M. Application-Specific Integrated Circuits / M. Smith. – Boston: Addison Wesley, 1997. – 836 p.

7. Баркалов А.А. Синтез микропрограммных автоматов на заказных и программируемых СБИС / А.А. Баркалов, Л.А. Титаренко. – Донецк: ДонНТУ, Технопарк ДонНТУ УНИТЕХ, 2009. – 336 с.
8. Баранов С.И. Синтез микропрограммных автоматов / С.И. Баранов. – Л.: Энергия, 1979. – 236 с.
9. Бибило П.Н. Синтез комбинационных ПЛИМ-структур для СБИС / П.Н. Бибило. – Минск: Наука и техника, 1992. – 268 с.
10. Navabi Z. Embedded Core Design with FPGAs / Z. Navabi. – NY: McGraw-Hill, 2007. – 435 p.

Надійшла до редколегії 30.01.2011

**О.О. БАРКАЛОВ<sup>1</sup>, І.Я. ЗЕЛЕНЬОВА<sup>2</sup>, ХАТГУТ БАЙРЕК<sup>2</sup>, Є.Р. ТАТОЛОВ<sup>2</sup>**

<sup>1</sup> Зеленогурський університет, м. Зелена гора, Польща

<sup>2</sup> Донецький національний технічний університет

**A.A. BARKALOV<sup>1</sup>, I.J. ZELENYOVA<sup>2</sup>, HATHOT BIAYREK<sup>2</sup>, E.R. TATOLOV<sup>2</sup>**

<sup>1</sup> University of Zielona Gora, Poland,

<sup>2</sup> Donetsk National Technical University

**Матрична реалізація композиційного мікропрограмного пристрою керування з загальною пам'яттю.**

Вперше запропонований метод реалізації композиційного мікропрограмного пристрою керування в базисі інтегральних схем, що зумовлюється користувачем (ASIC). Наведені теоретичні засади та практичні аспекти синтеза; показаний приклад застосування метода.

*композиційний мікропрограмний пристрій керування, операторний лінійний ланцюг, логічна схема, лінійний алгоритм керування, загальна пам'ять, інтегральна схема, що замовлюється користувачем, ASIC*

**Matrix Realization of Compositional Microprogram Control Unit with Common Memory.**

For the first time the method of compositional microprogram control unit realization on ASIC is proposed. Theoretical background and practical approach of synthesis are provided; an example of method application is given.

*compositional microprogram control unit, operational linear object, logical circuit, linear control algorithm, common memory, ASIC*