



**Аннотация.** Предложен метод уменьшения площади кристалла, занимаемой схемой совмещенного автомата. Метод основан на кодировании классов псевдоэквивалентных состояний автомата Мура дополнительными переменными. Этот подход порождает четырехуровневую схему, реализуемую в виде двух нано-ПЛИМ, и позволяет уменьшить площадь нано-ПЛИМ, формирующей микрооперации автомата Мура и дополнительные переменные. Рассмотрен пример синтеза схемы с использованием предложенного метода. Приведены результаты исследований эффективности метода с использованием библиотеки стандартных тестовых автоматов.

**Ключевые слова:** совмещенный микропрограммный автомат, синтез, нано-ПЛИМ, ASIC, псевдоэквивалентные состояния.

### ВВЕДЕНИЕ

Широкое применение встроенных, автономных и мобильных систем [1–3] тесно связано с проблемой уменьшения площади кристалла, занимаемой цифровой системой. Решение этой проблемы во многом зависит от особенностей элементного базиса [4, 5]. При массовом производстве наноэлектронных изделий целесообразно использование заказных СБИС типа ASIC (application specific integrated circuit) [6, 7]. Методы оптимизации площади для отдельных блоков зависят от их особенностей [8].

Одним из важнейших блоков цифровых систем является устройство управления [9], координирующее взаимодействие остальных блоков. Для синтеза схем устройств управления используются различные модели. Одна из них — модель совмещенного микропрограммного автомата (СМПА) [10]. Подобная модель позволяет реализовать автоматы, имеющие характеристики, как у автоматов Мили и Мура [8].

Схема СМПА задается системой булевых функций (СБФ), для реализации которой в базисе ASIC целесообразно использовать нанопрограммируемую логическую матрицу (нано-ПЛИМ) [11–13]. В известных публикациях не рассматривались методы синтеза схем СМПА в базисе нано-ПЛИМ. В настоящей статье предлагается метод уменьшения площади нано-ПЛИМ, основанный на кодировании классов псевдоэквивалентных состояний (ПЭС) автомата Мура [14]. Для спецификации СМПА используется язык граф-схем алгоритма (ГСА) [8], который наглядно иллюстрирует методы синтеза автоматов.

### ХАРАКТЕРИСТИКИ ОСОБЕННОСТЕЙ СМПА И БАЗИСА НАНО-ПЛИМ

Для синтеза схемы СМПА необходимо отметить исходную ГСА Г состояниями автомата Мура [8]. Это связано с тем, что СМПА генерирует два типа выходных сигналов (микроопераций). Микрооперации автомата Мили из множества

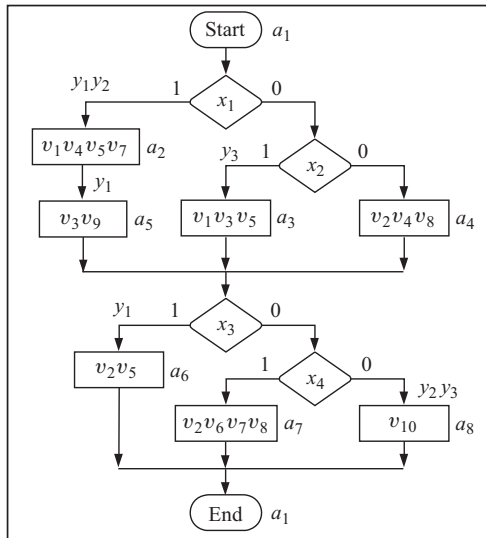


Рис. 1. Исходная ГСА  $\Gamma_1$

автомат задан ГСА  $\Gamma_1$  (рис. 1).

Анализ ГСА  $\Gamma_1$  позволяет найти следующие множества:  $A = \{a_1, \dots, a_8\}$ ,  $Y = \{y_1, y_2, y_3\}$ ,  $V = \{v_1, \dots, v_{10}\}$ ,  $X = \{x_1, \dots, x_4\}$ . Это дает  $M = 8$ ,  $N_1 = 3$ ,  $N_2 = 10$  и  $L = 4$ . Кроме того, можно определить число переходов между состояниями  $H = 16$ .

Состояния  $a_m \in A$  являются абстрактными объектами. Для синтеза схемы СМПА их необходимо закодировать двоичными кодами разрядности  $R$ . В случае реализации автоматов на ПЛИМ [13, 15] параметр  $R$  определяется как

$$R = \lceil \log_2 M \rceil. \quad (1)$$

Формула (1) применима и в базисе нано-ПЛИМ [11].

Для кодирования состояний используются внутренние переменные из множества  $T = \{T_1, \dots, T_R\}$ . Коды  $K(a_m)$  хранятся в регистре  $RG$ , состоящем из  $D$ -триггеров. Для записи кода в  $RG$  используются функции возбуждения памяти (ФВП), образующие множество  $\Phi = \{D_1, \dots, D_R\}$ . Замена содержимого  $RG$  иницируется импульсом синхронизации Clock. Как правило, начальное состояние  $a_1 \in A$  имеет нулевой код, который заносится в регистр  $RG$  по сигналу Start.

В настоящей работе предлагается уменьшить площадь нано-ПЛИМ, закодировав классы ПЭС. Отметим, что каждый класс ПЭС автомата Мура соответствует одному состоянию эквивалентного автомата Мили [14]. Если множество состояний  $A$  может быть разбито на  $I$  классов ПЭС  $B_i \in \Pi_A = \{B_1, \dots, B_I\}$ , то для их кодирования достаточно  $R_A$  переменных  $\tau_A \in \tau = \{\tau_1, \dots, \tau_{R_A}\}$ , где  $R_A = \lceil \log_2 I \rceil$ .

Состояния  $a_m, a_s \in B_i$ , если выходы отмеченных ими вершин соединены с входом одной и той же вершины ГСА  $\Gamma$  [4]. Используя это определение, можно найти следующее разбиение  $\Pi_A$  для ГСА  $\Gamma_1$ :  $\Pi_A = \{B_1, \dots, B_4\}$ , где  $B_1 = \{a_1\}$ ,  $B_2 = \{a_2\}$ ,  $B_3 = \{a_3, a_4, a_5\}$  и  $B_4 = \{a_6, a_7, a_8\}$ . Итак, имеем  $I = 4$ , что определяет  $R_A = 2$  и  $\tau = \{\tau_1, \tau_2\}$ . Используя (1), имеем  $R = 3$  для  $T = \{T_1, T_2, T_3\}$  и  $\Phi = \{D_1, D_2, D_3\}$  для СМПА заданного ГСА  $\Gamma_1$ .

Нано-ПЛИМ представляет собой последовательное соединение матриц: конъюнктивной  $M_1$  и дизъюнктивной  $M_2$  [11]. Для каждой матрицы используется идея распределенной логики [7], т.е. она представляется в виде совокупности  $N_V$  вертикальных и  $N_H$  горизонтальных шин. На пересечении шин находятся элементы с односторонней проводимостью.

$Y = \{y_1, \dots, y_{N_1}\}$  формируются на переходах между состояниями автомата, которые образуют множество  $A = \{a_1, \dots, a_M\}$ . Микрооперации автомата Мура из множества  $V = \{v_1, \dots, v_{N_2}\}$  формируются на протяжении всего такта работы СМПА. Таким образом, наборы микроопераций (НМО)  $Y_q \subseteq Y$  ассоциируются с переходами между состояниями автомата, а НМО  $V(a_m) \subseteq V$  — с состояниями автомата. Переходы между состояниями зависят от входных переменных, которыми являются логические условия (ЛУ)  $x_l \in X = \{x_1, \dots, x_L\}$ . В начальном такте функционирования автомат находится в исходном состоянии  $a_1 \in A$ . Пусть некоторый

На рис. 2 показана нано-ПЛМ, реализующая СБФ  $y_n \in Y$ , зависящих от  $L$  булевых переменных  $x_l \in X$  ( $|Y| = N$ ).

Матрица  $M_1$  реализует термы  $F_h$ , образующие множество  $F = \{F_1, \dots, F_H\}$ . Термы зависят как от прямых, так и от инверсных значений  $x_l \in X$ . Поэтому для  $M_1$  имеем  $N_V = 2L$ , а для матрицы  $M_2$  —  $N_V = H$  и  $N_H = N$ .

Для сравнения различных вариантов матричной реализации СБФ принято использовать площади матриц, выраженные в условных единицах площади (у.е.п.) [8, 11]. Площадь  $S(M_i)$  определяется произведением  $N_V \times N_H$ . Для схемы, изображенной на рис. 2, имеем

$$S(M_1) = 2L \times H, \quad (2)$$

$$S(M_2) = H \times N. \quad (3)$$

Площадь этой схемы определяется суммой выражений (2) и (3).

Как следует из формул (2) и (3), для уменьшения площади необходимо уменьшать параметры  $L$ ,  $N$  или  $H$ . Поэтому для оптимизации матричной схемы следует выполнять совместную минимизацию СБФ [9]. Оптимизация подобного типа должна выполняться и для СМПА.

#### ТРИВИАЛЬНАЯ РЕАЛИЗАЦИЯ СМПА В БАЗИСЕ НАНО-ПЛМ

Построим прямую структурную таблицу (ПСТ) СМПА [10]. Каждая строка ПСТ соответствует одному переходу между состояниями автомата и имеет  $H$  строк. Номер строки из столбца  $h$  соответствует терму  $F_h$  ( $h \in \{1, \dots, H\}$ ), входящему в ДНФ функций  $y_n \in Y$  и  $D_r \in \Phi$  [8]. Эта таблица имеет следующие столбцы [16, 17]:  $a_m$  — исходное состояние, из которого осуществляется переход;  $K(a_m)$  — код состояния  $a_m \in A$ ;  $V(a_m) \subseteq V$  — НМО, формируемый в состоянии  $a_m \in A$ ;  $a_s$  — состояние перехода;  $K(a_s)$  — код состояния  $a_s \in A$ ;  $X_h$  — входной сигнал, определяющий переход  $\langle a_m, a_s \rangle$  и равный конъюнкции некоторых элементов множества ЛУ или их отрицаний;  $Y_h \subseteq Y$  — НМО, формируемый на переходе  $\langle a_m, a_s \rangle$ ;  $\Phi_h \subseteq \Phi$  — набор ФВП, принимающих единичное значение для замены содержимого  $RG$ , а именно кода  $K(a_m)$  кодом  $K(a_s)$ ;  $h$  — номер перехода,  $h \in \{1, \dots, H\}$ .

Каждая строка ПСТ соответствует терму

$$F_h = \bigwedge_{r=1}^R T_r^{l_{mr}} X_h \quad (h = \overline{1, H}), \quad (4)$$

где первый член выражения соответствует конъюнкции  $A_m$ , определяемой кодом состояния  $a_m \in A$ ;  $l_{mr} \in \{0, 1\}$  — значение  $r$ -го разряда  $T_r^0 = \overline{T_r}$  и  $T_r^1 = T_r$ ,  $r \in \{1, \dots, R\}$ . Отметим, что дизъюнкция всех термов (4) с одинаковым первым членом  $A_m$  дает конъюнкцию  $A_m$ . Таким образом, если конъюнкция  $A_m$  входит в ДНФ функции  $v_n \in V$ , то в эту ДНФ должны войти все термы (4), содержащие конъюнкцию  $A_m$ . Термы (4) входят в ДНФ функций  $y_n \in Y$  и  $D_r \in \Phi$ .

Используя ПСТ, можно найти множество  $F$  термов (4) и следующие функции:

$$Y = Y(F), \quad \Phi = \Phi(F), \quad V = V(F). \quad (5)$$

Эта система определяет тривиальную схему СМПА  $U_1$  в базисе нано-ПЛМ (рис. 3).

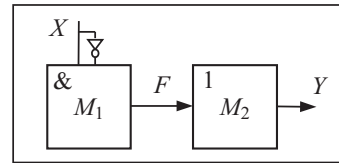


Рис. 2. Реализация СБФ в базисе нано-ПЛМ

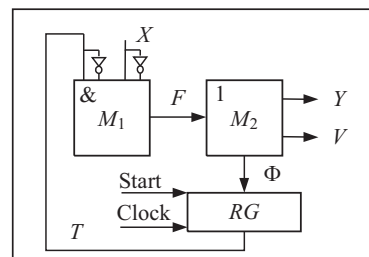


Рис. 3. Структурная схема СМПА  $U_1$

В схеме на рис. 3 матрица  $M_1$  реализует систему термов  $F$  и имеет площадь

$$S(M_1) = 2(L + R) \times H. \quad (6)$$

Матрица  $M_2$  реализует функции (5) и имеет площадь

$$S(M_2) = (R + N1 + N2) \times H. \quad (7)$$

Результат суммирования (6) и (7) дает площадь  $S(U_1)$  схемы СМПА  $U_1$ :

$$S(U_1) = (2L + 3R + N1 + N2) \times H. \quad (8)$$

Известно [8], что тривиальные матричные схемы автоматов значительно избыточны. Для уменьшения площади, занимаемой схемой автомата, используют различные методы структурной редукции [18, 19]. В настоящей статье предлагается оптимизировать площадь, используя кодирование классов ПЭС, в результате чего будет получен СМПА  $U_2$ , структура и метод синтеза которого рассмотрены далее.

#### ОСНОВНАЯ ИДЕЯ И МЕТОД СИНТЕЗА СМПА $U_2$

Найдем разбиение  $\Pi_A = \{B_1, \dots, B_I\}$  для некоторой ГСА  $\Gamma_j$  и выполним кодирование состояний для автомата  $U_1(\Gamma_j)$ . Здесь запись  $U_i(\Gamma_j)$  означает, что автомат со структурой  $U_i$  синтезируется по ГСА  $\Gamma_j$ . Используя ГСА  $\Gamma_j$ , построим систему обобщенных формул переходов (ОФП) [20].

Например, для ГСА  $\Gamma_1$  можно построить следующую систему ОФП:

$$\begin{aligned} B_1 &\rightarrow x_1 a_2 \vee \overline{x_1 x_2 a_3} \vee \overline{x_1 x_2 a_4}; & B_2 &\rightarrow a_5; \\ B_3 &\rightarrow x_3 a_6 \vee \overline{x_3 x_4 a_7} \vee \overline{x_3 x_4 a_8}; & B_4 &\rightarrow a_1. \end{aligned} \quad (9)$$

Как следует из (9), в левой части каждой ОФП вместо состояний  $a_m \in A$  находятся классы  $B_i \in \Pi_A$ , включающие эти состояния.

Используя систему ОФП, построим ПСТ автомата  $U_2$  со столбцами  $B_i$ ,  $K(B_i)$ ,  $a_s$ ,  $K(a_s)$ ,  $X_h$ ,  $\Phi_h$ ,  $h$ , в которой  $B_i$  — класс ПЭС из левой части ОФП, а  $K(B_i)$  — его код. Таблица имеет  $H_0$  строк, каждая из которых соответствует одному терму системы ОФП. Если для класса  $B_i \in \Pi_A$  выполняются переходы в состояние с нулевым кодом и не формируются НМО  $Y_h \subseteq Y$ , то такие переходы в ПСТ не включаются.

Таким образом, ПСТ автомата  $U_2$  определяет множество термов  $F_0$ ,  $|F_0| = H_0$ . Каждый терм  $F_h \in F_0$  соответствует конъюнкции

$$F_h = \bigwedge_{r=1}^{R_A} \tau_r^{l_{ir}} X_h \quad (h = \overline{1, H_0}), \quad (10)$$

где первый член определяется кодом  $K(B_i)$ ,  $l_{ir} \in \{0, 1\}$  — значение  $r$ -го разряда  $K(B_i)$ . Это конъюнкция переменных  $\tau_r \in \tau$ , где  $\tau_r^0 = \overline{\tau_r}$  и  $\tau_r^1 = \tau_r$  ( $r = \overline{1, R_A}$ ).

Из ПСТ  $U_2$  формируются системы функций  $D_r \in \Phi$  и  $y_n \in Y$ . Система функций  $v_n \in V$  формируется по ГСА  $\Gamma_1$ . При этом функции из систем  $\Phi$  и  $Y$  зависят от термов (10), а функции  $v_n \in V$  — от термов  $A_m$ . Эти термы образуют соответственно множества  $F_0$ ,  $|F_0| = H_0$ , и  $A_V$ ,  $|A_V| = M_V$ . Схема СМПА  $U_2$  задается системами функций

$$Y = Y(F_0), \quad (11)$$

$$\Phi = \Phi(F_0), \quad (12)$$

$$V = V(A_V). \quad (13)$$

Термы (10) зависят от переменных  $\tau_r \in \tau$ . В силу зависимости классов  $B_i \in \Pi_A$  от состояний [14] термы системы  $\tau = \tau(A)$ , также как и функции

зависят от  $A_m \in A_V$ .

В результате проведенного анализа можно предложить структурную схему СМПА  $U_2$  (рис. 4). Эта схема включает две нано-ПЛИМ, образованные матрицами  $M_1, M_2$  и  $M_3, M_4$ . При этом матрица  $M_1$  формирует систему термов (10), матрица  $M_2$  — СБФ (11) и (12), матрица  $M_3$  — термы (4), а матрица  $M_4$  — функции (13) и (14).

В настоящей работе предлагается такой метод синтеза схемы СМПА  $U_2$ .

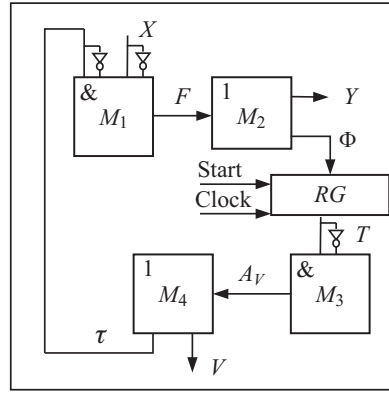


Рис. 4. Структурная схема СМПА  $U_2$

1. Отметить исходную граф-схему состояниями автомата Мура.

2. Найти разбиение  $\Pi_A = \{B_1, \dots, B_I\}$  множества  $A$ .

3. Закодировать классы  $B_i \in \Pi_A$ .

4. Закодировать состояния  $a_m \in A$ .

5. Построить систему ОФП и ПСТ СМПА  $U_2$ .

6. Построить термы для  $A_V$  и (10), а также СБФ (11)–(14).

7. Построить схему СМПА  $U_2$  на нано-ПЛИМ.

#### ПРИМЕР СИНТЕЗА СХЕМЫ СМПА $U_2$

В рассматриваемом примере  $I = 4, R_A = 2$ . Закодируем классы  $B_i \in \Pi_A$  следующим образом: чем больше состояний включает класс, тем больше единиц содержит его код. Из анализа разбиения  $\Pi_A$  можно получить такие коды:  $K(B_1) = 10, K(B_2) = 11, K(B_3) = 00, K(B_4) = 01$ .

Кодирование состояний  $a_m \in A$  необходимо выполнить так, чтобы минимизировать величину  $M_V = |A_V|$ . Построим системы функций  $\tau(A)$  и  $V(A)$  для рассматриваемого примера:

$$\begin{aligned} v_1 &= A_2 \vee A_3, v_2 = A_4 \vee A_6 \vee A_7, v_3 = A_3 \vee A_5, \\ v_4 &= A_2 \vee A_4, v_5 = A_2 \vee A_3 \vee A_6, v_6 = A_7, \end{aligned} \quad (15)$$

$$\begin{aligned} v_7 &= A_2 \vee A_7, v_8 = A_4 \vee A_7, v_9 = A_5, v_{10} = A_8; \\ \tau_1 &= A_1 \vee A_2, \tau_2 = A_2 \vee A_4. \end{aligned} \quad (16)$$

Для минимизации систем (15), (16) используем алгоритм из [21]. Коды состояний  $a_m \in A$  приведены на рис. 5.

Система ОФП (9) и коды (см. рис. 5) позволяют построить ПСТ СМПА  $U_2(\Gamma_1)$  (табл. 1).

Таблица имеет  $H_0 = 7$  строк, так как переходы происходят из состояний  $a_m \in B_4$  в состояние  $a_1 \in A$  с кодом  $K(a_1) = 000$ . Из табл. 1 следует, что  $F_0 = \{F_1, \dots, F_7\}$ , где  $F_1 = \tau_1 \tau_2 x_1, F_2 = \tau_1 \tau_2 x_1 x_2, \dots, F_7 = \tau_1 \tau_2 x_3 x_4$ . Термы  $F_h \in F_0$  формирует матрица  $M_1$ .

Из табл. 1 можно найти следующие СБФ, реализуемые матрицей  $M_2$  тривиальным образом:

$$\begin{aligned} y_1 &= F_1 \vee F_4 \vee F_5, y_2 = F_1 \vee F_7, y_3 = F_2 \vee F_7; \\ D_1 &= F_3 \vee \dots \vee F_6, D_2 = F_1 \vee F_2 \vee F_4 \vee F_6, D_3 = F_2 \vee F_4 \vee F_5 \vee F_7. \end{aligned}$$

	$T_1$	$T_2$		
$T_3$	00	01	11	10
0	$a_1$	$a_2$	$a_7$	$a_4$
1	$a_8$	$a_3$	$a_5$	$a_6$

Рис. 5. Коды состояний СМПА  $U_2(\Gamma_1)$

**Таблица 1.** Прямая структурная таблица автомата  $U_2(\Gamma_1)$

$B_i$	$K(B_i)$	$a_s$	$K(a_s)$	$X_h$	$Y_h$	$\Phi_h$	$h$
$B_1$	10	$a_2$	010	$x_1$	$y_1 y_2$	$D_2$	1
		$a_3$	011	$\overline{x_1 x_2}$	$y_3$	$D_2 D_3$	2
		$a_4$	100	$x_1 x_2$	—	$D_1$	3
$B_2$	11	$a_5$	111	1	$y_1$	$D_1 D_2 D_3$	4
$B_3$	00	$a_6$	101	$x_3$	$y_1$	$D_1 D_3$	5
		$a_7$	110	$\overline{x_3 x_4}$	—	$D_1 D_2$	6
		$a_8$	001	$\overline{x_3 x_4}$	$y_2 y_3$	$D_3$	7

Найдем термы систем (13), (14), образующие множество  $A_V = \{F_8, \dots, F_{H1}\}$ . Начнем с  $F_8$ , поскольку множество  $F_0$  включает  $H_0 = 7$  элементов. Используя коды из рис. 4 и законы алгебры логики [9], преобразуем системы (15), (16) следующим образом:

$$\begin{aligned} v_1 &= \overline{T_1} T_2 = F_8, \quad v_2 = T_1 \overline{T_3} \vee T_1 \overline{T_2} = F_9 \vee F_{10}, \quad v_3 = T_2 T_3 = F_{11}, \\ v_4 &= A_2 \vee A_4 = F_{12} \vee F_{13}, \quad v_5 = \overline{T_1} T_2 \vee A_6 = F_8 \vee F_{14}, \quad v_6 = A_7 = F_{15}, \\ v_7 &= T_2 \overline{T_3} = F_{16}, \quad v_8 = T_1 \overline{T_3} = F_9, \quad v_9 = A_5 = F_{17}, \quad v_{10} = A_8 = F_{18}; \\ \tau_1 &= \overline{T_1} \overline{T_3} = F_{15}, \quad \tau_2 = A_2 \vee A_4 = v_4. \end{aligned}$$

Эта система имеет  $M_V = 11$  термов, реализуемых на матрице  $M_3$ . При этом  $N_0 = 8$  функций представляются в виде одного терма. Таким образом, эти восемь функций формируются непосредственно матрицей  $M_3$  и образуют множества  $V^1 \subseteq V$  и  $\tau^1 \subseteq \tau$ . Следовательно, матрица  $M_4$  формирует остальные функции из множества  $V^2 = V/V^1$  и  $\tau^2 = \tau/\tau^1$ .

В рассматриваемом случае  $V^1 = \{v_1, v_3, v_6, \dots, v_{10}\}$ ,  $\tau^1 = \{\tau_1\}$ ,  $V^2 = \{v_2, v_4, v_5\}$ ,  $\tau^2 = \{\tau_2\}$ . В силу равенства  $\tau_2 = v_4$  матрица  $M_4$  формирует  $N_1 = N_2 + R_A - N_0 - 1 = 3$  функции. При этом только  $M_{V_1} = 6$  термов из множества  $A_{V_0} = \{F_9, F_{10}, F_{12}, F_{13}, F_8, F_{14}\}$  являются входами матрицы  $M_4$ .

Сравним варианты реализации функций  $V(T)$  и  $\tau(T)$  в виде, показанном на рис. 4 и рис. 6 (для рассматриваемого примера).

Для схемы на рис. 4 площади  $S(M_3)$ ,  $S(M_4)$  оцениваются следующим образом:

$$S(M_3) = 2R \cdot M = 48 \text{ у.е.п.}, \quad (17)$$

$$S(M_4) = M \cdot (N_2 + R_A) = 96 \text{ у.е.п.} \quad (18)$$

Для схемы на рис. 6 имеем для рассматриваемого примера

$$S(M_3) = 2R \cdot M_V = 6 \cdot 11 = 66 \text{ у.е.п.}, \quad (19)$$

$$S(M_4) = M_{V_1} \cdot N_1 = 6 \cdot 3 = 18 \text{ у.е.п.} \quad (20)$$

Суммирование (17) и (18) дает 144 у.е.п., а суммирование (19) и (20) дает 84 у.е.п. Второй вариант по сравнению с первым дает выигрыш в 1.71 раза. Такой эффект достигнут, благодаря использованию метода [21] для кодирования  $a_m \in A$ .

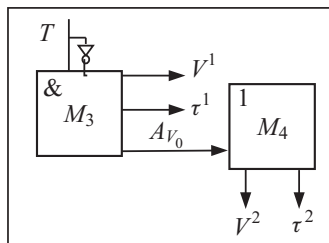


Рис. 6. Реализация функций  $V(T)$  и  $\tau(T)$  с учетом оптимизации

Последний этап предложенного метода выполняется тривиальным образом. Отметим, что физическая реализация схемы нано-ПЛМ связана с изготовлением литографических масок и другими сложными технологическими процессами [6].

#### АНАЛИЗ ЭФФЕКТИВНОСТИ ПРЕДЛОЖЕННОГО МЕТОДА

Условимся понимать под эффективностью метода отношение площади некоторой базовой схемы к площади схемы, полученной в результате применения предложенного метода. Площадь  $S(U_1)$  определена в (8). Найдем подобное выражение для  $S(U_2)$ .

Площади матрицы автомата  $U_2$  определяются по формулам:

$$S(M_1) = 2(L + R_A) \times H_0, \quad (21)$$

$$S(M_2) = H_0 \times (N1 + R), \quad (22)$$

$$S(M_3, M_4) = 2R \times M_V + M_{V_1} \times N_1. \quad (23)$$

Равенство (23) найдено как сумма равенств (19) и (20). Если  $M_V = M_{V_1} = M$  и  $N_1 = N_2 + R_A$ , то равенства (19), (20) превращаются в (17), (18) соответственно. Результат суммирования (21)–(23) дает площадь  $S(U_2)$ .

Для  $U_1(\Gamma_1)$  число термов (4)  $H = 16$ . Используя параметры  $L = 4$ ,  $R = 3$ ,  $N1 = 3$ ,  $N2 = 10$ , найдем  $S(U_1) = 480$  у.е.п. Для  $U_2(\Gamma_1)$  имеем  $S(M_1) = 2 \cdot (2 + 4) \cdot 7 = 84$ ,  $S(M_2) = 7 \cdot (3 + 3) = 42$ ,  $S(M_3) + S(M_4) = 84$ . Это дает  $S(U_2) = 210$ . Отношение  $S(U_1) / S(U_2) = 2.28$ , т.е. предложенный метод позволяет на 57 % уменьшить площадь матричной схемы СМПА.

Естественно, такая эффективность получена только для рассматриваемого примера. Для общего случая необходимо исследовать некоторую совокупность стандартных примеров (benchmarks) с помощью библиотеки [22], собранной в 90-х годах XX века. Во многих работах, описывающих схемы автоматов, именно эта библиотека используется для сравнения различных методов синтеза.

Как показали проведенные исследования, предложенный метод всегда давал лучшие результаты, чем метод  $U_1$ . Выигрыш от применения модели  $U_2$  увеличился по мере роста отношения числа кодирующих переменных  $R / R_A$  и отношения числа строк ПСТ  $H / H_0$ . При этом второй фактор имел большее влияние. Кроме того, уменьшение площади  $S(M_3, M_4)$  зависит от того, какая часть функций  $v_n \in V$  и  $\tau_r \in T$  представляется в виде одного терма (конъюнкции внутренних переменных  $T_r \in T$ ). В предельном случае матрица  $M_3$  генерирует все функции из множества  $V \cup T$ , а матрица  $M_4$  имеет нулевую площадь. Отметим, что для примеров из [22] такая ситуация не возникала.

Переход от  $U_1$  к  $U_2$  позволил в среднем уменьшить площадь схемы на 26 %, в данном случае максимальный выигрыш достигал 42 %, а минимальный — 11.5 %. Отметим, что теоретически выигрыш невозможен, если  $I = M$ . При этом каждый класс  $B_i \in \Pi_A$  включает только один элемент. Для стандартных примеров из [22] такая ситуация также не возникала.

#### ЗАКЛЮЧЕНИЕ

Уменьшение площади кристалла ASIC, занимаемой схемой устройства управления, позволяет снизить потребляемую мощность, увеличить быстродействие и повысить надежность схемы [9]. Предложенный в работе метод ориентирован на решение этой задачи для СМПА. Поскольку схема СМПА задается системами булевых функций, для ее реализации выбран базис нано-ПЛМ.

Предлагаемый метод основан на кодировании классов ПЭС автомата Мура. Именно состояниями автомата Мура отмечена ГСА, задающая закон функционирования СМПА. При этом происходит переход от двухуровневой схемы  $U_1$  (см. рис. 3)

к четырехуровневой схеме  $U_2$  (см. рис. 4). Кодирование классов ПЭС выполняется дополнительными переменными, которые необходимо генерировать как выходы одной из матриц.

Для оптимизации схемы, формирующей микрооперации автомата Мура и дополнительные переменные, используется метод кодирования состояний, основанный на подходе [21]. В предельном случае эти микрооперации и дополнительные переменные формируются на третьем уровне схемы, что позволяет увеличить быстродействие.

Исследования с использованием библиотеки [22] показали высокую эффективность предложенного метода. В среднем площадь схем автоматов  $U_2$  уменьшается до 0.76 площади автоматов  $U_1$ . Основываясь на результатах [5, 7, 11], можно ожидать, что снизится и потребляемая мощность.

Увеличение числа уровней в схеме  $U_2$  (относительно  $U_1$ ) свидетельствует, что предложенный метод относится к группе методов структурной декомпозиции [19], в которую входят методы замены логических условий, кодирования полей совместимых микроопераций, кодирования термов ПСТ и др. [23, 24]. В дальнейших исследованиях планируется применить эти подходы для оптимизации схем СМПА в базе ASIC, PLD [25] и в нейронных сетях [26].

#### СПИСОК ЛИТЕРАТУРЫ

1. Barkalov A., Titarenko L., Mazurkiewicz M. Foundations of embedded systems. Berlin: Springer, 2019. 167 p.
2. Marwedel P. Embedded system design: Embedded systems, foundations of cyber-physical systems and the Internet of Things. Berlin: Springer, 2017. 612 p.
3. Gajski D.D., Abdi S., Gerstlauer A., Schirner G. Embedded system design: Modeling, synthesis and verification. New York: Springer, 2009. 352 p.
4. Baranov S. Logic and system design of digital systems. Tallin: TUT Press, 2008. 267 p.
5. Czerwinski R., Kania D. Finite state machines logic synthesis for complex programmable logic devices. *Lecture Notes in Electrical Engineering*. Vol. 231. Berlin; Heidelberg: Springer-Verlag, 2013. 172 p.
6. Smith M. Application specific integrated circuits. Boston: Addison-Wesley, 1997. 632 p.
7. Nababi Z. Embedded core design with FPGAs. New York: McGraw-Hill, 2008. 418 p.
8. Baranov S. Logic synthesis for control automata. Dordrecht: Kluwer Academic Publishers, 1994. 312 p.
9. DeMicheli G. Synthesis and optimization of digital circuits. New York: McGraw-Hill, 1994. 634 p.
10. Баркалов А.А., Титаренко Л.А., Визор Я.Е., Матвиенко А.В., Горина В.В. Уменьшение числа LUT элементов в схеме совмещенного автомата. *Управляющие системы и машины*. 2016. № 3. С. 16–22.
11. Baranov S., Levin L., Keren O., Karpovsky M. Designing fault tolerant FSM by nano-PLA. *Proc. of 15th International On-Line Testing Symposium*. Lisbon, 2009. P. 216–220.
12. Naemi H., DeHon A. A greedy algorithm for tolerating crosspoints in NanoPLA design. *Proc. of IEEE International Competence on Field-Programmable Technology*. Piscataway, NJ, 2004. P. 49–56.
13. Баранов С.И., Складов В.А. Цифровые устройства на программируемых БИС с матричной структурой. Москва: Радио и связь, 1986. 272 с.
14. Баркалов А.А. Принципы оптимизации логической схемы микропрограммного автомата Мура. *Кибернетика и системный анализ*. 1998. № 1. С. 65–72.
15. Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем. Москва: Горячая линия — ТЕЛЕКОМ, 2001. 636 с.
16. Баркалов А.А., Титаренко Л.А., Визор Я.Е., Матвиенко А.В. Синтез совмещенного микропрограммного автомата в базе FPGA. *Колп'ютерні засоби, мережі та системи*. Київ: Ін-т кібернетики ім. В.М. Глушкова НАН України, 2015. Вип. 14. С. 32–39.
17. Баркалов А.А., Титаренко Л.А., Визор Я.Е., Матвиенко А.В. Реализация схемы совмещенного микропрограммного автомата в базе FPGA. *Проблеми інформатизації та управління*. Київ: Національний авіаційний університет, 2015. Вип. 3(51). С. 5–13.
18. Баркалов А.А. Структуры многоуровневых схем микропрограммных автоматов на программируемых логических матрицах. *Кибернетика и системный анализ*. 1994. № 4. С. 22–29.
19. Баркалов А.А., Титаренко Л.А., Баркалов А.А. (мл.). Структурная декомпозиция как средство оптимизации схемы автомата Мили на ПЛИС. *Кибернетика и системный анализ*. 2012. № 2. С. 177–187.



20. Barkalov A., Titarenko L., Kolopenczyk M., Mielcarek K., Bazydło G. Logic synthesis for FPGA-based finite state machines. Berlin: Springer, 2016. 280 p.
21. Ачасова С.М. Алгоритмы синтеза автоматов на программируемых логических матрицах. Москва: Сов. радио, 1987. 132 с.
22. Yang S. Logic synthesis and optimization benchmarks user guide. Version 3.0. Microelectronics Center of North Carolina, 1991. 43 p.
23. Баркалов А.А., Титаренко Л.А., Цололо С.А. Оптимизация схемы автомата Мура, реализуемой в базе ПЛИС. *Кибернетика и системный анализ*. 2009. № 5. С. 180–186.
24. Баркалов А.А., Титаренко Л.А., Лаврик А.С. Уменьшение аппаратных затрат в устройстве управления с разделением кодов. *Кибернетика и системный анализ*. 2013. № 3. С. 113–123.
25. Palagin A.V., Opanasenko V.N. 3 Design and application of the PLD-based reconfigurable devices. In: Design of Digital Systems and Devices. *Lecture Notes in Electrical Engineering*. Adamski M., Barkalov A., Wegrzyn M. (Eds.). Berlin; Heidelberg: Springer, 2011. Vol 79. P. 59–91.
26. Opanasenko V.N., Kryvyi S.L. Synthesis of neural-like networks on the basis of conversion of cyclic hamming codes. *Cybernetics and Systems Analysis*. 2017. Vol. 53, N 4. P. 627–635. <https://doi.org/10.1007/s10559-017-9965-z>.

*Надійшла до редакції 17.01.2020*

**О.О. Баркалов, Л.О. Титаренко, А.В. Баєв, О.В. Матвієнко**  
**ОПТИМІЗАЦІЯ СХЕМИ СУМІЩЕНОГО АВТОМАТА В БАЗИСІ ASIC**

**Анотація.** Запропоновано метод зменшення площі кристала, яку займає схема суміщеного автомата. Метод базується на кодуванні класів псевдоеквівалентних станів автомата Мура додатковими змінними. Цей підхід породжує чотирирівневу схему, яка реалізується у вигляді двох нано-ПЛМ, і дає змогу зменшити площу нано-ПЛМ, яка формує мікрооперації автомата Мура та додаткові змінні. Розглянуто приклад синтезу схеми із застосуванням запропонованого методу. Наведено результати досліджень ефективності методу з використанням бібліотеки стандартних тестових автоматів.

**Ключові слова:** суміщений мікропрограмний автомат, синтез, нано-ПЛМ, ASIC, псевдоеквівалентні стани.

**A.A. Barkalov, L.A. Titarenko, A.V. Baiev, A.V. Matviienko**  
**OPTIMIZING THE SCHEME OF A COMBINED AUTOMATON IN THE ASIC BASIS**

**Abstract.** A method is proposed for decreasing the area of the ASIC occupied by the scheme of a combined automaton. The method is based on encoding of the classes of pseudoequivalent states of Moore automaton by additional variables. This approach leads to a four-level scheme implemented as two nano-PLAs and decreases the area of nano-PLA generating microoperations of the Moore automaton and additional variables. An example of synthesis with the use of the proposed scheme is considered. The results of the efficiency analysis of the proposed method with the use of a library of benchmarks are presented.

**Keywords:** combined microprogrammed automaton, synthesis, nano-PLA, ASIC, pseudo-equivalent states.

**Баркалов Александр Александрович,**  
 доктор техн. наук, профессор Университета Зеленогурского (Польша), профессор Донецкого национального университета имени Василя Стуса, Винница, e-mail: A.Barkalov@iie.uz.zgora.pl.

**Титаренко Лариса Александровна,**  
 доктор техн. наук, профессор Университета Зеленогурского (Польша), профессор Харьковского национального университета радиоелектроники, e-mail: L.Titarenko@iie.uz.zgora.pl.

**Баєв Артем Викторович,**  
 кандидат физ.-мат. наук, декан Донецкого национального университета имени Василя Стуса, Винница; руководитель направления по искусственному интеллекту, фирма Peoly, Винница, e-mail: a.baev@donnu.edu.ua.

**Матвієнко Александр Владимирович,**  
 научный сотрудник Института кибернетики им. В.М. Глушкова НАН Украины, Киев, e-mail: avmatv@ukr.net.