

УДК 681.3

А. А. Баркалов (доктор техн. наук, проф.),

Р. В. Мальчева (канд. техн. наук, доц.),

К. А. Солдатов (ассистент)

Донецкий национальный технический университет

raisa@cs.dgtu.donetsk.ua

МАТРИЧНАЯ РЕАЛИЗАЦИЯ АВТОМАТА МУРА С КОДИРОВАНИЕМ НАБОРОВ МИКРООПЕРАЦИЙ

Предлагается метод уменьшения аппаратных затрат в схеме микропрограммного автомата Мура, ориентированный на использование заказных матриц. Метод основан на представлении кода состояния перехода в виде конкатенации кодов класса псевдоэквивалентных состояний и набора микроопераций. Такой подход позволяет устранить зависимость между состояниями и микрооперациями. В результате оптимизируется как блок формирования функций возбуждения, так и блок формирования микроопераций. Приведен пример применения предложенного метода.

Автомат Мура, граф-схема алгоритма, псевдоэквивалентные состояния, заказные матрицы, логическая схема

Введение

Модель микропрограммного автомата (МПА) Мура [1] часто используется при реализации устройств управления цифровых систем [2,3]. Прогресс в области микроэлектроники привел к появлению разнообразных программируемых логических устройств [4,5], используемых при реализации схемы МПА. Однако при массовом производстве изделий микроэлектроники по-прежнему широко используются заказные схемы типа ASIC (Application-Specified Integrated Circuits) [6]. При этом, как правило, схемы МПА реализуются в виде матричных структур, использующих принцип распределенной логики [7].

Одной из важных проблем, возникающих при реализации схемы МПА на заказных матрицах, является уменьшение площади кристалла, занимаемой схемой. Одним из путей решения этой проблемы является оптимальное кодирование состояний [8]. Однако этот подход не позволяет оптимизировать схему формирования выходных сигналов.

В настоящей работе мы предлагаем метод оптимизации, основанный на представлении кода состояния МПА в виде конкатенации кодов класса псевдоэквивалентных состояний и кода набора микроопераций (микрокоманды). Такой подход позволяет уменьшить аппаратные затраты в комбинационных блоках МПА и не приводит к потере быстродействия.

Целью исследования является оптимизация схемы МПА Мура за счет нестандартного представления кодов состояний.

Задачей исследования является разработка метода синтеза МПА Мура, позволяющего уменьшить площадь матриц в схеме автомата. При этом алгоритм управления представляется в виде граф-схемы алгоритма (ГСА) [1].

Общие положения и основная идея предлагаемого метода

Пусть автомат Мура задан прямой структурной таблицей (ПСТ) со столбцами [1]: $a_m, K(a_m), a_s, K(a_s), X_h, \Phi_h, h$. Здесь a_m – исходное состояние МПА; $K(a_m)$ – код состояния $a_m \in A$ разрядности $R_A = \lceil \log_2 M \rceil$, для кодирования состояний используются внутренние переменные из множества $T = \{T_1, \dots, T_{R_A}\}$; $a_s, K(a_s)$ – соответственно состояние перехода и его код; X_h – входной сигнал, определяющий переход $\langle a_m, a_s \rangle$, и равный конъюнкции некоторых элементов (или их отрицаний) множества логических условий $X = \{x_1, \dots, x_L\}$; Φ_h – набор функций возбуждения триггеров памяти МПА, принимающих единичное значение для переключения памяти из $K(a_m)$ в $K(a_s)$, $\Phi_h \subseteq \Phi = \{\varphi_1, \dots, \varphi_{R_A}\}$; $h=1, \dots, H$ – номер перехода. В столбце a_m записывается набор микроопераций Y_q , формируемых в состоянии $a_m \in A$, где $Y_q \subseteq Y = \{y_1, \dots, y_N\}$, $q=1, \dots, Q$. Эта таблица является основой для формирования систем функций

$$\Phi = \Phi(T, X), \quad (1)$$

$$Y = Y(T), \quad (2)$$

задающих логическую схему МПА. Системы (1)-(2) определяют матричную схему МПА Мура U_1 , состоящую из четырех матриц (рис. 1).

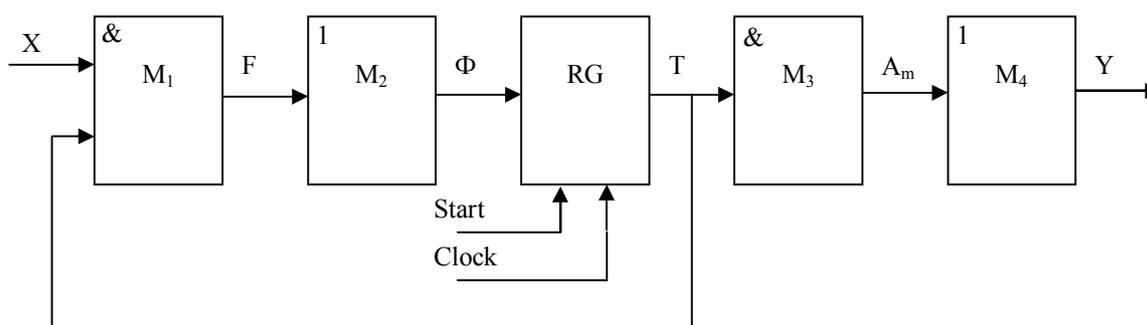


Рисунок 1 – Матричная реализация автомата Мура U_1

В автомате U_1 конъюнктивная матрица M_1 реализует систему термов $F = \{F_1, \dots, F_H\}$; дизъюнктивная матрица M_2 реализует систему (1);

конъюнктивная матрица M_3 реализует термы $A_m(m=1, \dots, M)$, соответствующие состоянием МПА; дизъюнктивная матрица M_4 реализует функции системы (2). Регистр RG служит для хранения кодов состояний и управляется сигналами Start (обнуление) и Clock (изменение содержимого в зависимости от функции Φ). Матрицы M_1 и M_2 определяют блок функций возбуждения памяти (БФП), а матрицы M_3 и M_4 – блок микроопераций (БМО). Для уменьшения площади блока БФП может использоваться метод оптимального кодирования состояний [8], позволяющий уменьшить число термов в системе (1) до N_0 . Здесь N_0 – число переходов эквивалентного автомата Мили. Площадь блока БМО может быть уменьшена за счет уточненного кодирования состояний [9]. В пределе каждая микрооперация $y_n \in Y$ выражается одной конъюнкцией матрицы M_3 и матрица M_4 отсутствует. Для оптимального и уточненного кодирования может использоваться, например, известный алгоритм ESPRESSO [3]. Однако оба этих метода не могут быть использованы одновременно, то есть кодирование состояний позволяет уменьшить площадь либо БФП, либо БМО.

В настоящей работе предлагается метод, позволяющий уменьшить площадь матричной реализации для обоих блоков МПА Мура.

Одной из особенностей МПА Мура является наличие псевдоэквивалентных состояний [8], то есть состояний с одинаковыми переходами под воздействием одинаковых входных сигналов. Такие состояния соответствуют операторным вершинам [1] алгоритма управления, выходы которых связаны со входом одной и той же вершины алгоритма.

Пусть $\Pi_A = \{B_1, \dots, B_k\}$ – разбиение множества A на классы псевдоэквивалентных состояний. Закодируем классы $B_i \in \Pi_A$ двоичными кодами $K(B_i)$ разрядности

$$R_B = \lceil \log_2 I \rceil. \quad (3)$$

Пусть исходная ГСА Γ включает Q попарно различных наборов микроопераций (НМО) $Y_q \subseteq Y$. Закодируем набор Y_q двоичным кодом $K(Y_q)$ разрядности

$$R_Y = \lceil \log_2 Q \rceil. \quad (4)$$

Пусть $E_1 = \{b_1, \dots, b_D\}$ – множество операторных вершин исходной ГСА Γ . Зададим отношение α на множестве E_1

$$b_i \alpha b_j \leftrightarrow Y(b_i) = Y(b_j), \quad (5)$$

где $Y(b_i), Y(b_j) \subseteq Y$ – НМО, записанный соответственно в вершинах b_i и b_j ($(i, j \in \{1, \dots, D\})$). Отношение α симметрично, рефлексивно, транзитивно и, следовательно, является отношением эквивалентности [10]. Отношение α

определяет разбиение $\Pi_\alpha = \{C_1, \dots, C_\eta\}$ на множестве E_1 . Пусть $G = \max(|C_1|, \dots, |C_\eta|)$, закодируем каждую вершину $b_q \in C_j$ двоичным кодом $K(b_q)$ разрядности

$$R_\alpha = \lceil \log_2 G \rceil \quad (6)$$

и используем переменные $z_r \in Z_1$ для такого кодирования, где $|Z_1| = R_\alpha$. Тогда код состояния $a_m \in A$ можно представить в следующем виде:

$$K(a_m) = K(Y_q) * K(b_q), \quad (7)$$

где $b_q \in E_1$ – операторная вершина, отмеченная состоянием $a_m \in A$, $Y_q = Y(b_q)$, * – знак конкатенации.

Построим систему функций

$$B = B(A), \quad (8)$$

задающую зависимость классов $B_i \in \Pi_A$ от состояний $a_m \in A$. При этом каждая функция $B_i \in B$ представляется в виде

$$B_i = \bigvee_{i=1}^I C_{im} A_m \quad (i = 1, \dots, I), \quad (9)$$

где C_{im} – булевская переменная, равная единице, если и только если $a_m \in B_i$. Теперь можно перейти к предлагаемой матричной реализации МПА Мура U_2 (рис. 2).

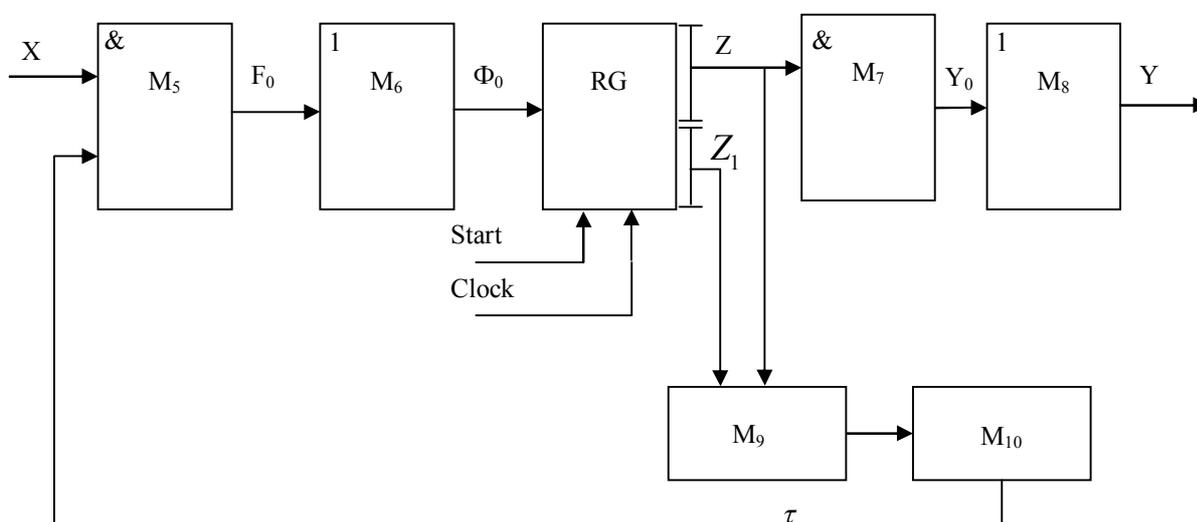


Рисунок 2 – Матричная реализация автомата Мура U_2

В автомате U_2 матрица M_5 реализует систему термов F_0 , соответствующих строкам преобразованной ПСТ и зависящих от логических условий $x_i \in X$ и переменных $\tau_r \in \tau$, кодирующих классы $B_i \in \Pi_A$, где $|\tau| = R_B$. Матрица M_6 реализует систему функций возбуждения RG

$$\Phi_0 = \Phi_0(\tau, X) \quad (10)$$

зависящую от термов $F_h \in F_0$. Система (10) включает $R_Y + R_\alpha$ функций, каждая из которых управляет соответствующим D-триггером регистра RG. Матрица M_7 реализует термы Y_0 , входящие в систему функций $y_n \in Y$ и зависящие от переменных $z_r \in Z$, где $|z| = R_Y$. Матрица M_8 реализует функции $y_n \in Y$, зависящие от термов $\Delta_q \in Y_0$. Матрица M_9 реализует термы A_0 , входящие в систему (9), а матрица M_{10} – функции $\tau_r \in \tau$, используемые для кодирования классов $B_i \in \Pi_A$, при этом $|\tau| = R_B$.

Матрица M_5 и M_6 образуют блок БФП, реализующий функции $D_r \in \Phi_0$. Матрицы M_7 и M_8 образуют блок БМО, реализующий функции

$$Y = Y(Z). \quad (11)$$

Матрица M_9 и M_{10} образуют блок преобразователя кодов (БПК), реализующий функции

$$\tau = \tau(z, z_1). \quad (12)$$

Предлагаемый метод имеет ряд положительных качеств:

1. Коды наборов микроопераций не зависят от кодов состояний. Поэтому наборы $Y_q \subseteq Y$ можно закодировать так, чтобы максимально уменьшить площадь матриц БМО.

2. Число строк таблицы переходов МПА U_2 всегда равняется N_0 . При этом классы $B_i \in \Pi_A$ можно закодировать так, чтобы уменьшить сложность матриц блока БФП.

Как было отмечено ранее, для кодирования состояний МПА U_1 достаточно

$$R_A = \lceil \log_2 M \rceil \quad (13)$$

переменных. Очевидным недостатком автомата U_2 является увеличение числа входов БФП при выполнении условия

$$R_Y + R_\alpha > R_A. \quad (14)$$

Кроме того, модель U_2 включает блок БПК, который потребляет некоторые ресурсы кристалла. Однако эти недостатки компенсируются значительным уменьшением площадей блоков БФП и БМО по сравнению с моделью U_1 .

Метод синтеза МПА Мура с кодирование наборов микроопераций

В данной работе предлагается метод синтеза МПА Мура U_2 по ГСА Γ , включающий следующие этапы:

1. Отметка ГСА Γ и формирование множества состояний A .
2. Разбиение множества A на классы псевдоэквивалентных состояний.

3. Кодирование наборов микроопераций $Y_q \subseteq Y$.
4. Формирование разбиения Π_α и кодирование операторных вершин $b_q \in E_1$. Определение кодов состояний.
5. Кодирование классов $B_i \in \Pi_A$.
6. Построение преобразованной ПСТ автомата U_2 .
7. Формирование системы (12) по таблице БПК.
8. Реализация матриц M_5 - M_{10} .

Первый этап выполняется по известной методике [1], когда каждая вершина $b_q \in E_1$ отмечается отдельным состоянием. При этом начальная и конечная вершина отмечаются начальным состоянием $a_1 \in A$. Второй этап выполняется тривиальным образом, используя определение псевдоэквивалентных состояний [8].

Третий этап выполняется так, чтобы максимально уменьшить число термов в системе Y_0 . В предельном случае каждая микрооперация $y_n \in Y$ представляется одним термом и матрица M_8 отсутствует [1]. Для кодирования наборов микроопераций можно использовать известный алгоритм ESPRESSO [3].

Четвертый этап выполняется на основе выражения (5). При этом коды операторных вершин назначаются с учетом неопределенностей. Коды состояний $a_m \in A$ определяются на основе формулы (7).

Классы $B_i \in \Pi_A$ кодируются так, чтобы уменьшить число термов в системе (12). Этот этап сводится к известной задаче символьного кодирования [3]. Для оптимизации системы (12) возможна перекодировка операторных вершин.

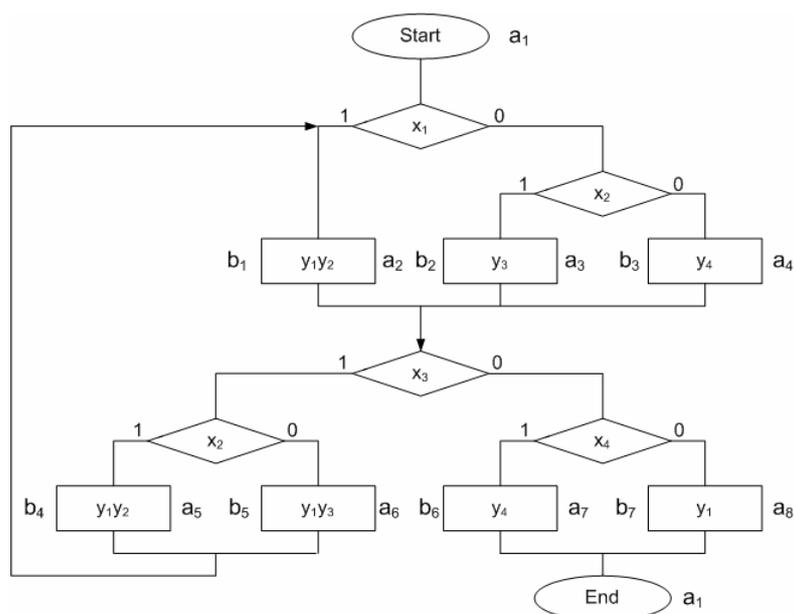
Преобразованная ПСТ автомата U_2 имеет столбцы $B_i, K(B_i), a_s, K(a_s), X_k, \Phi_k, h$. Здесь $\Phi_h \subseteq \Phi_0$ – набор функций возбуждения памяти, принимающих единичное значение для формирования в RG кода $K(a_s)$; $h=1, \dots, N_0$ – номер перехода. Эта таблица строится на основе системы обобщенных формул перехода [11].

Таблица блока БПК включает столбцы $a_m, K(a_m), B_i, K(B_i), \tau_m, m$. Здесь $\tau_m \subseteq \tau$ – переменные, принимающие единичные значения в коде $K(B_i)$ из m -й строки таблицы, $m=1, \dots, M$.

Выполнение последнего этапа не представляет трудностей и рассматривается в примере.

Пример применения предложенного метода

Пусть символ $U_i(\Gamma_j)$ означает, что ГСА Γ_j интерпретируется используя модель U_i ($i=1,2$). Рассмотрим пример синтеза автомата Мура $U_2(\Gamma_1)$, где ГСА Γ_1 представлена на рис. 3.

Рисунок 3 – Исходная ГСА Γ_1

Как следует из ГСА Γ_1 , множество $A = \{a_1, \dots, a_8\}$, $M = 8$ и $R_A = 3$. В данном случае $\Pi_A = \{B_1, \dots, B_4\}$, где $B_1 = \{a_1\}$, $B_2 = \{a_2, a_3, a_4\}$, $B_3 = \{a_5, a_6\}$, $B_4 = \{a_7, a_8\}$. Итак, $I = 4$, $R_B = 2$, $\tau = \{\tau_1, \tau_2\}$. В операторных вершинах ГСА Γ_1 находится $Q=5$ различных наборов микроопераций: $Y_1 = 0$, $Y_2 = \{y_1, y_2\}$, $Y_3 = \{y_3\}$, $Y_4 = \{y_4\}$, $Y_5 = \{y_1, y_3\}$. Для их кодирования достаточно $R_Y = 3$ переменных, то есть $Z = \{z_1, z_2, z_3\}$. Закодируем наборы $Y_q \subseteq Y$ так, как показано на рис. 4.

		$Z_2 Z_3$		
Z_1	00	01	11	10
0	Y_1	*	Y_4	Y_2
1	*	*	Y_3	Y_5

Рисунок 4 – Коды наборов микроопераций МПА U_2 (Γ_1)

Используя рис. 3 и рис. 4 можно получить систему формул:

$$\begin{aligned}
 y_1 &= Y_2 \vee Y_5 = \overline{z_2} z_3 = \Delta_1; \\
 y_2 &= Y_2 = \overline{z_1} z_2 \overline{z_3} = \Delta_2; \\
 y_3 &= Y_3 \vee Y_5 = z_1 = \Delta_3; \\
 y_4 &= Y_4 = \overline{z_1} z_3 = \Delta_4;
 \end{aligned} \tag{15}$$

Найдем разбиение Π_α , которое в нашем примере включает 4 класса: $C_1 = \{b_1, b_4, b_7\}$, $C_2 = \{b_2\}$, $C_3 = \{b_3, b_6\}$, $C_4 = \{b_5\}$. Очевидно, $G = 3$, $R_\alpha = 2$, $r_1 = \{z_4, z_5\}$, то есть мы присваиваем сквозную нумерацию элементам объединения множеств Z и Γ_1 .

В нашем случае система (8) имеет следующий вид:

$$\begin{aligned} B_1 &= A_1; B_2 = A_2 \vee A_3 \vee A_4; \\ B_3 &= A_5 \vee A_6; B_4 = A_7 \vee A_8. \end{aligned} \quad (16)$$

Закодируем операторные вершины $b_q \in E_1$ так, чтобы коды состояний (7) определялись из рис. 5.

		$z_1 z_2 z_3$							
		000	001	011	010	110	111	101	100
$z_4 z_5$	00	a_1	*	a_4	a_2	*	a_3	*	*
	01	*	*	*	a_5	a_6	*	*	*
	11	*	*	*	*	*	*	*	*
	10	*	*	a_8	a_7	*	*	*	*

Рисунок 5 – Коды состояний МПА U_2 (Γ_1)

Из карты Карно (рис. 5) следует, что:

$$\begin{aligned} B_1 &= \overline{z_2}; B_2 = \overline{z_2 z_4 z_5}; \\ B_3 &= z_5; B_4 = z_4. \end{aligned} \quad (17)$$

Закодируем классы $B_i \in \Pi_A$ следующим образом $K(B_1) = 01, K(B_2) = 00, K(B_3) = 10, K(B_4) = 11$. Из этих кодов получаем следующую систему уравнений:

$$\begin{aligned} \tau_1 &= B_3 \vee B_4 = z_5 \vee z_4; \\ \tau_2 &= B_1 \vee B_4 = \overline{z_2} \vee z_4. \end{aligned} \quad (18)$$

Система (18) определяет блок БПК. Как следует из (18), матрица M_9 в блоке БПК отсутствует.

Построим систему обобщенных формул перехода из ГСА Γ_1 :

$$\begin{aligned} B_1 &\rightarrow x_1 a_2 \vee \overline{x_1 x_2} a_3 \vee \overline{x_1 x_2} a_4; \\ B_2 &\rightarrow x_3 x_2 a_5 \vee \overline{x_3 x_2} a_6 \vee \overline{x_3 x_4} a_7 \vee \overline{x_3 x_4} a_8; \\ B_3 &\rightarrow a_2; B_4 \rightarrow a_1. \end{aligned} \quad (19)$$

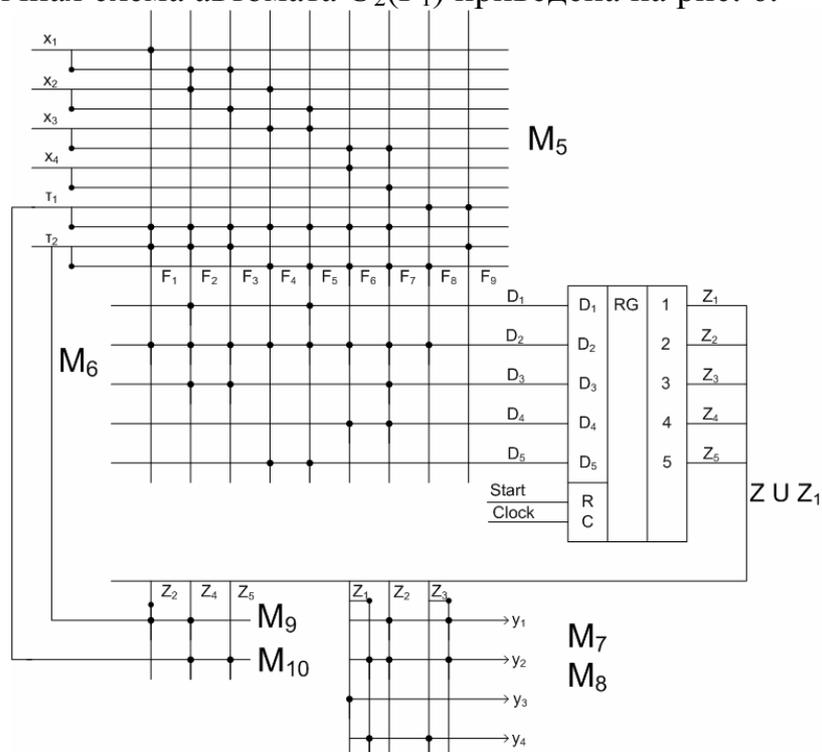
Используя систему (19) и коды состояний из рис. 5, построим преобразованную ПСТ автомата $U_2(\Gamma_1)$, имеющую $N_0=9$ строк (табл. 1).

Из этой таблицы формируются система (10). Например, из табл. 1 можно найти функции: $D_1 = F_2 \vee F_5 = \overline{\tau_1 \tau_2} x_1 x_2 \vee \overline{\tau_1 \tau_2} x_3 x_2$; $D_2 = F_2 \vee \dots \vee F_8$; $D_3 = F_2 \vee F_3 \vee F_7$; $D_4 = F_6 \vee F_7$; $D_5 = F_4 \vee F_5$. Эти функции в случае их матричной реализации не минимизируются. Таблицу блока БПК в нашем примере строить не нужно, так как система (18) задает функции (12).

Таблица 1. Преобразованная ПСТ автомата Мура $U_2(\Gamma_1)$

B_i	$K(B_i)$	a_s	$K(a_s)$	X_h	Φ_h	h
B_1	01	a_2	01000	x_1	D_2	1
		a_3	11100	$\overline{x_1 x_2}$	$D_1 D_2 D_3$	2
		a_4	01100	$\overline{x_1 x_2}$	$D_2 D_3$	3
B_2	00	a_5	01001	$x_3 x_2$	$D_2 D_5$	4
		a_6	11001	$x_3 \overline{x_2}$	$D_1 D_2 D_5$	5
		a_7	01010	$\overline{x_3 x_4}$	$D_2 D_4$	6
		a_8	01110	$\overline{x_3 x_4}$	$D_2 D_3 D_4$	7
B_3	10	a_2	01000	1	D_2	8
B_4	11	a_1	00000	1	-	9

Матричная схема автомата $U_2(\Gamma_1)$ приведена на рис. 6.

Рисунок 6 – Матричная реализация автомата Мура $U_2(\Gamma_1)$

Определим площадь, занимаемую матрицами M_5 - M_{10} , которая определяется как произведение чисел входов и выходов матрицы. Из рис.6 имеем: $S(M_5) = 2(4 + 2) * 9 = 108$, $S(M_6) = 9 * 5 = 45$, $S(M_7, M_8) = 5 * 4 = 20$ и $S(M_9, M_{10}) = 3 * 2 = 6$. Итак, для реализации схемы МПА $U_2(\Gamma_1)$ необходимо 179 условных единиц площади [1]. Для автомата $U_1(\Gamma_1)$ можно найти, что $H=19$, $S(M_1) = 2(4 + 3) * 19 = 166$, $S(M_2) = 19 * 3 = 57$, $S(M_3) = 2 * 3 * 7 = 42$ и

$S(M_4) = 7 * 4 = 28$. Итак, схема МПА $U_1(\Gamma_1)$ занимает 293 условных единиц площади. Кроме того, схема $U_1(\Gamma_1)$ имеет 4 уровня комбинационной логики, а схема $U_2(\Gamma_1)$ – только три, так как функции τ и Y формируются параллельно. Таким образом, применение метода кодирования наборов микроопераций и представление кодов состояний в виде (7) позволило в 1,7 раза уменьшить площадь и в 1,3 раза повысить быстродействие комбинационной схемы автомата Мура по сравнению с её классической реализацией.

Заключение

Предлагаемый в работе метод представления кодов состояний ориентирован на уменьшение площади матричной реализации микропрограммного автомата Мура. Этот подход позволяет уменьшить число термов в системе функций возбуждения памяти МПА Мура до соответствующего параметра эквивалентного автомата Мили. Кроме того предлагаемый метод позволяет уменьшить число термов в системе микроопераций в силу независимости кодов состояний и кодов наборов микроопераций.

Исследование эффективности предложенного метода на стандартных примерах [10] показали, что предлагаемый метод позволяет в среднем уменьшить площадь реализации до 52%. При этом в 86% случаев быстродействие МПА увеличивалось, благодаря отсутствию матрицы M_8 .

Научная новизна предложенного метода заключается в использовании особенностей автомата Мура (наличие классов псевдоэквивалентных состояний) для уменьшения площади матричной схемы автомата.

Практическая значимость метода заключается в уменьшении площади кристалла, занимаемой комбинационной схемой МПА, что позволяет получить схемы, которые обладают меньшей стоимостью, чем известные из литературы аналоги.

Дальнейшие направления работы связаны с исследованием возможности применения предложенного метода для случая реализации устройства управления в базисе FPGA.

Список литературы

1. Baranov S. Logic Synthesis for Control Automata/ S. Baranov – Kluwer Academic Publishers, 1994. – 312 pp.
2. Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем/ В.В. Соловьев – М.: Горячая линия - ТЕЛЕКОМ, 2001. – 636 с.

3. DeMicheli G. Synthesis and Optimization of Digital Circuits/ G. DeMicheli – McGraw-Hill, 1994. – 636 pp.
4. Грушницький Р.И. Проектирование систем с использованием микросхем программируемой логики/ Р.И. Грушницький, А.Х. Мурзаев, Е.П. Угрюмов. – СПб.: БХВ. - Петербург, 2002. – 608 с.
5. Maxfield C. The Design Warrior's Guide to FPGAs/ C. Maxfield – Amsterdam: Elsevier, 2004. – 541 pp.
6. Smith M. Application-Specific Integrated Circuits/ M. Smith – Boston: Addison-Wesley, 1997. – 836 pp.
7. Nababi Z. Embedded Core Design with FPGA/ Z. Nababi – NY: McGraw-Hill, 2008, - 618 pp.
8. Баркалов А.А. Принципы оптимизации логической схемы микропрограммного автомата Мура/ А.А. Баркалов – Кибернетика и системный анализ. – 1998. – № 1. – С. 65-72.
9. Баркалов А.А. Уменьшение аппаратных затрат в схеме микропрограммного автомата Мура на CPLD/ А.А. Баркалов, Л.А. Титаренко, С.А. Цололо – Радіоелектронні та комп'ютерні системи. – 2008. - №7(34). – с. 118-123.
10. Шиханович Ю.А. Введение в современную математику/ Ю.А. Шиханович – М.:Наука, 1965. – 376 с.
11. Yang S. Logic Synthesis and Optimization Benchmarks user guide/ S. Yang – Technical report, №1991 – IWLS-UG-Saryang.-Microelectronics center of North Carolina.

Надійшла до редакції 22.10.2009 р. Рецензент: д.т.н., проф. Скобцов Ю.О.

О.О. Баркалов, Р.В. Мальчева, К.А. Солдатов
Донецький національний технічний університет

Матрична реалізація автомату Мура з кодуванням наборів мікрооперацій. Запропоновано метод зменшення апаратних витрат в схемі мікропрограмного автомату Мура, який орієнтований на використання замовних матриць. Метод заснований на представленні кодів стану переходу у вигляді конкатенації кодів класу псевдоеквівалентних станів і набору мікрооперацій. У результаті оптимізується як блок формування функцій збудження пам'яті, так і блок формування мікрооперацій. Наведено приклад застосування запропонованого методу.

Автомат Мура, граф-схема алгоритму, псевдоеквівалентні стани, замовні матриці, логічна схема

A.A. Barkalov, R.V. Malcheva, K.A. Soldatov
Donetsk National Technical University

Method to reduce hardware amount in the Moore finite state machine, which focused on the use of custom matrixes, is proposed. The method based on representation of the next state code as a concatenation of codes for class of pseudoequivalent states and set of microoperations. This approach allows to eliminate a dependence between states and microoperations. As the result the block of input memory functions and the block of microoperations are optimized. An example of the proposed method is presented.

Moore finite state machine, graph-scheme of algorithm, pseudoequivalent states, custom matrixes, logic circuit