

УДК 004.274

А. А. БАРКАЛОВ, Л. А. ТИТАРЕНКО, И. Я. ЗЕЛЕНЕВА, С. С. ГРУШКО

МЕТОД УМЕНЬШЕНИЯ ЧИСЛА ТЕРМОВ ПРИ РЕАЛИЗАЦИИ СХЕМЫ СОВМЕЩЕННОГО МИКРОПРОГРАММНОГО АВТОМАТА В БАЗИСЕ CPLD

Рассмотрена задача синтеза совмещенного микропрограммного автомата (СМПА) в базисе CPLD. Понятие «совмещенный» подразумевает, что схема функционирует одновременно и как автомат Мили, и как автомат Мура, что часто применяется в практике разработки цифровых систем. Выбор базиса CPLD объясняется его широким использованием в настоящее время. Внутренние макроячейки (PAL, PLA) микросхемы CPLD, на которых реализуется схема совмещенного автомата, имеют строгие ограничения по числу термов. В связи с этим, в работе предложен метод уменьшения числа термов, реализуемых в схеме СМПА. Метод основан на кодировании классов псевдоэквивалентных состояний автомата и позволяет существенно сократить аппаратные затраты в схеме за счет уменьшения необходимого числа макроячеек. Эффективность предложенного метода показана на примере синтеза СМПА по исходной граф-схеме алгоритма.

Ключевые слова: совмещенный микропрограммный автомат, CPLD, макроячейка, PAL, псевдоэквивалентные состояния, граф-схема

Розглянуто задачу синтезу суміщеного мікропрограмного автомату (СМПА) в базисі CPLD. Під поняттям «суміщений» мається на увазі, що схема функціонує одночасно як автомат Мілі та як автомат Мура, що часто використовується в практиці проектування цифрових схем. Вибір базису CPLD пояснюється його широким застосуванням в наш час. Внутрішні макрокомірки (PAL, PLA) мікросхеми CPLD, на яких реалізована схема суміщеного автомата, мають жорсткі обмеження щодо числа термів. У зв'язку з цим, в роботі запропоновано метод зменшення числа термів, які потрібно реалізувати в схемі СМПА. Метод засновано на кодуванні класів псевдоеквівалентних станів автомата, що дозволяє ефективно скоротити апаратні витрати в схемі за рахунок зменшення необхідного числа макрокомірок. Ефективність запропонованого методу показана на прикладі синтезу СМПА по граф-схемі алгоритму.

Ключові слова: суміщений мікропрограмний автомат, CPLD, макрокомірка, PAL, псевдоеквівалентні стани, граф-схема алгоритму.

We consider the problem of synthesis of the combined finite state machine (CFSM) in the basis CPLD. Notion "combined" means that the scheme operates as both Mealy and Moore FSM that is often used in the practice of the development of digital systems. Selecting the CPLD is due to its wide using today. The logic circuit of CFSM is implemented on internal macrocells (PAL, PLA) of CPLD chip, which are restrictive on the number of terms. Therefore, in this paper we propose a method to reduce the number of terms to be implemented in the scheme CFSM. The method is based on coding of classes pseudoequivalent states of CFSM and significantly reduces hardware expenses in the scheme by reducing the required number of macrocells. The effectiveness of the proposed method is shown by the example of the synthesis of CFSM on the original graph-scheme of algorithm.

Keywords: combined finite state machine, CPLD, macrocell, PAL, pseudoequivalent states, graph-scheme of algorithm.

Введение. Модель микропрограммного автомата (МПА) часто используется для реализации схем устройств управления [1, 2]. При синтезе МПА возникает ряд оптимизационных задач, одной из которых является задача уменьшения аппаратных затрат в схеме микропрограммного автомата [3, 4]. Решение этой задачи, как правило, позволяет уменьшить потребляемую мощность, необходимую для функционирования МПА, и увеличить его быстродействие [3–6]. Методы решения данной задачи во многом зависят от модели МПА и особенностей элементного базиса [5, 6].

В настоящей работе рассматривается метод синтеза совмещенного МПА (СМПА) в базисе CPLD (complex programmable array logic devices) с макроячейками типа PAL (programmable array logic). Понятие «совмещенный автомат» подразумевает, что МПА функционирует одновременно как автомат Мили и автомат Мура. Интерес авторов к модели СМПА объясняется практически полным отсутствием статей, связанных с синтезом схем такого автомата. При этом СМПА часто используется при реализации реальных цифровых систем [5]. Авторы данной статьи предложили ряд методов реализации схем СМПА в базисе FPGA (field programmable gate arrays) [7–9]. При этом необходимо отметить, что микросхемы CPLD также часто используются для реализации устройств управления [6, 10]. Этими факторами и объясняется интерес к развитию методов [8, 7] для базиса CPLD.

Анализ особенностей совмещенного автомата и базиса CPLD. Особенностью СМПА является наличие выходных сигналов двух типов [1]. Выходные переменные автомата Мили образуют

множество Y^1 , где $|Y^1| = N_1$. Выходные переменные автомата Мура образуют множество Y^2 , где $|Y^2| = N_2$. Множество выходных переменных $Y = \{y_1, \dots, y_N\}$ СМПА образуется как $Y^1 \cup Y^2$. При этом $Y^1 \cap Y^2 \neq \emptyset$ и $N_1 + N_2 = N$.

Таким образом, СМПА может быть представлен как вектор [1]:

$$S = \langle A, X, Y^1, Y^2, \delta, \lambda_1, \lambda_2, \alpha_1 \rangle. \quad (1)$$

В векторе (1) имеются следующие компоненты:

$A = \{\alpha_1, \dots, \alpha_M\}$ – множество внутренних состояний; $X = \{x_1, \dots, x_L\}$ – множество входных переменных; δ – функция переходов; λ_1 – функция выходов автомата Мили; λ_2 – функция выходов автомата Мура; $\alpha_1 \in A$ – исходные состояния СМПА.

Функции δ , λ_1 и λ_2 определяются следующим образом:

$$a_s = \delta(a_m, X), \text{ где } (a_m, a_s) \in A; \quad (2)$$

$$y_n = \lambda_1(a_m, X), \text{ где } y_n \in Y^1; \quad (3)$$

$$y_n = \lambda_2(a_m, X), \text{ где } y_n \in Y^2. \quad (4)$$

Функция δ служит для нахождения состояния перехода $a_s \in A$ в зависимости от текущего состояния $a_m \in A$ и вектора входных переменных. Как видно из (3), выходные переменные автомата Мили зависят от состояний и входных переменных. Из (4) следует, что выходные переменные автомата Мура зависят только от состояний.

© А. А. Баркалов, Л. А. Титаренко, И. Я. Зеленева, С. С. Грушко. 2016

Наличие функций (3) и (4) позволяет использовать для схемы СМПА известные методы оптимизации, эффективные как для МПА Мили, так и для МПА Мура. Это и есть главная особенность совмещенного микропрограммного автомата.

Микросхема CPLD состоит из макроячеек PAL [11, 12]. Каждая макроячейка представляет собой q программируемых схем «И», имеющих s общих входов. Схемы «И» соединены схемой «ИЛИ». Вход схемы «ИЛИ» может быть соединен (либо нет) с входом D-триггера. Триггер имеет входы синхронизации, обнуления и установки в единицу. Таким образом, выход макроячейки может быть комбинационным или регистровым. Макроячейки связаны между собой при помощи программируемой матрицы межсоединений. Эта же матрица связывает макроячейки с входами и выходами микросхемы.

Основной особенностью CPLD является достаточно небольшое значение параметра q ($q \leq 8$). Как правило, это приводит к необходимости раздельной минимизации булевских функций, задающих схему автомата [5, 6]. Целью минимизации является уменьшение числа термов в булевских функциях.

В настоящей работе предлагается один из возможных методов решения этой задачи. При этом СМПА синтезируется на основе граф-схемы алгоритма (ГСА) [1].

Реализация СМПА по граф-схеме алгоритма.

При синтезе СМПА имеется ряд этапов, которые не зависят от элементного базиса. К ним относятся [1]:

1. Отметка исходной ГСА состояниями $a_m \in A$.

2. Кодирование состояний $a_m \in A$ двоичными кодами $K(a_m)$ разрядности R , где

$$R = \lceil \log_2 M \rceil. \tag{5}$$

3. Построение прямой структурной таблицы (ПСТ) автомата.

4. Формирование систем функций, соответствующих (2)–(4) по ПСТ.

Коды состояний $a_m \in A$ хранятся в регистре RG, состоящем из R триггеров. В случае CPLD этот регистр распределен между ячейками PAL [10]. Триггеры имеют общие входы синхронизации, на которые получают импульс Clock, и обнуления, на которые поступает импульс Start. Для представления состояний используются внутренние переменные, образующие множество $T = \{T_1, \dots, T_R\}$. Для изменения содержимого RG используют функции возбуждения памяти, образующие множество $\varphi = \{D_1, \dots, D_R\}$. Переменная $D_r \in \varphi$ поступает на вход D r -го триггера ($r = \overline{1, R}$) распределенного регистра RG.

ПСТ совмещенного автомата строится по правилам [1] и включает следующие столбцы: a_m – текущее состояние; $K(a_m)$ – код состояния $a_m \in A$; a_s – состояние перехода; $K(a_s)$ – код состояния $a_s \in A$; x_h – входной сигнал, определяющий переход $\langle a_m, a_s \rangle$; Y_h^1 – набор входных переменных, формируемых на

переходе $\langle a_m, a_s \rangle$; φ_h – набор функций возбуждения, принимающих единичные значения для переключения регистра RG из $K(a_m)$ в $K(a_s)$; h – номер перехода ($h = \overline{1, H}$). Кроме того, в столбцах a_m записываются выходные переменные $y_n \in Y^2$, формируемые в состоянии $a_m \in A$.

На следующем этапе формируются функции, задающие схему СМПА:

$$\varphi = \varphi(T, X); \tag{6}$$

$$Y^1 = Y^1(T, X); \tag{7}$$

$$Y^2 = Y^2(T). \tag{8}$$

Система (6) определяет функцию переходов (2), система (7) – функцию выходов (3), и система (8) – соответственно, функцию (4).

Обозначим символом PALer схему, состоящую из элементов PAL. Из (6) и (7) следует, что функции (2) и (3) определяют схему с входами $x_i \in X$ и $T_r \in T$. Назовем ее схемой PALer1. Функция (8) соответствует схеме PALer2, имеющей только входы $T_r \in T$. Блок PALer1 включает регистр RG. Следовательно, блок PALer1 должен иметь входы Clock и Start. Системы (6)–(8) определяют структурную схему СМПА U_1 (рис. 1). Как следует из рис.1, выходами блока PALer1 являются внутренние переменные $T_r \in T$, а не функции $D_r \in \varphi$.

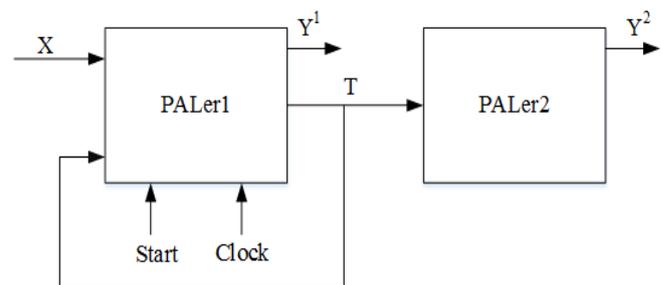


Рис. 1 – Структурная схема СМПА U_1

Функции (6), (7) зависят от термов F_h , соответствующих строкам ПСТ:

$$F_h = A_m \cdot X_h \quad (h = \overline{1, H}), \tag{9}$$

где A_m – конъюнкция переменных $T_r \in T$, соответствующая состоянию $a_m \in A$ из h -й строки ПСТ. Функции (6), (7) определяются следующим образом:

$$D_r = \bigvee_{h=1}^H C_{rh} F_h \quad (r = \overline{1, R}); \tag{10}$$

$$y_n = \bigvee_{h=1}^H C_{nh} F_h \quad (n = \overline{1, N_1}). \tag{11}$$

Здесь $C_{rh} (C_{nh})$ – булева переменная, равная 1,

если и только если переменная $D_r(y_n)$ записана в h -й строке ПСТ.

Функция (8) зависит от термов A_m и определяется следующим образом:

$$y_n = \bigvee_{m=1}^M C_{mn} A_m \quad (n = \overline{1, N_2}) \quad (12)$$

Здесь C_{mn} – булева переменная, равная 1, если и только если переменная $y_n \in Y^2$ формируется в состоянии $a_m \in A$.

Образуем множество функций $O = \varphi \cup Y$, включающее K элементов, где $K = R + N$. Пусть $H(O_k)$ – множество термов в дизъюнктивной нормальной форме (ДНФ) функции $O_k \in O$. Очевидно, уменьшение числа термов необходимо только для функций, у которых

$$q < H(O_k) \quad (k = \overline{1, K}). \quad (13)$$

Как правило, для минимизации систем функций $D_r \in \varphi$ и $y_n \in Y$ используют различные методы кодирования состояний [13–15]. В настоящей работе мы предлагаем метод оптимизации, основанный на наличии псевдоэквивалентных состояний (ПЭС) автомата Мура [15].

Предлагаемый метод синтеза СМПА с уменьшением числа термов. Состояния $(a_m, a_s) \in A$ являются псевдоэквивалентными, если отмеченные ими вершины ГСА Γ связаны с входом одной и той же вершины ГСА [15]. Это определение позволяет построить разбиение π_A множества A на классы ПЭС: $\pi_A = \{B_1, \dots, B_l\}$.

Закодируем классы $B_i \in \pi_A$ двоичными кодами $K(B_i)$ разрядности

$$R_i = \lceil \log_2 l \rceil. \quad (14)$$

Используем для кодирования классов ПЭС элементы множества $\tau = \{\tau_1, \dots, \tau_{R_i}\}$.

Очевидно, классы ПЭС соответствуют состояниям автомата Мили. Представим функции (6), (7) в следующем виде:

$$\varphi = \varphi(\tau, X); \quad (15)$$

$$Y^1 = Y^1(\tau, X). \quad (16)$$

Для кодирования классов $B_i \in \pi_A$ необходимо реализовать систему функций

$$\tau = \tau(T). \quad (17)$$

Сравнение функций (6) и (15), а также (7) и (16) показывает, что переменные $\tau_r \in \tau$ заменяют переменные $T_r \in T$. При этом функции (8) не меняются.

Проведенный анализ позволяет предложить модель U_2 (рис. 2).

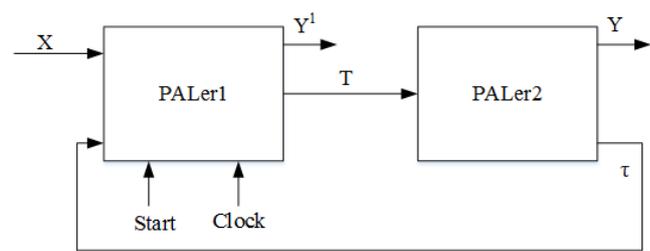


Рис. 2 – Структурная схема СМПА U_2

В СМПА U_2 блок PALer1 реализует функции (15) и (16). Блок PALer2 реализует системы (8) и (17).

Для синтеза схемы U_2 по ГСА необходимо построить ПСТ, которая отличается от ПСТ автомата U_1 . Отличие сводится к замене столбцов a_m и $K(a_m)$ столбцами B_i и $K(B_i)$ соответственно. При этом выполняется условие

$$H_1 < H, \quad (18)$$

где H_1 – число строк ПСТ автомата U_2 , эквивалентного автомату U_1 . Очевидно, автоматы U_1 и U_2 являются эквивалентными, если они синтезируются по одной и той же ГСА Γ .

Предлагаемый подход позволяет для эквивалентных автоматов U_1 и U_2 выполнить следующее:

1. Уменьшить число термов в функциях (15), (16) по сравнению с функциями (6) и (7), соответственно. Это следует из условия (18).

2. Кодирование состояний $a_m \in A$ автомата U_2 можно выполнить так, чтобы уменьшить число термов в функциях системы (8). Это возможно, так как коды $K(a_m)$ практически не влияют на число термов в функциях (15) и (16).

Недостатком U_2 является наличие системы (17), для реализации которой необходимы некоторые ресурсы кристалла. Однако кодирование состояний можно выполнить так, чтобы минимизировать число макроячеек PAL, необходимых для реализации системы (17).

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

1. Формирование отмеченной ГСА Γ .
2. Формирование разбиения π_A на множестве A .
3. Кодирование классов $B_i \in \pi_A$.
4. Кодирование состояний СМПА.
5. Формирование прямой структурной таблицы автомата U_2 .
6. Формирование систем функций, задающих схему U_2 .
7. Реализация схемы СМПА в заданном базисе.

Рассмотрим применение этого метода для случая ГСА Γ_1 (рис. 3).

Пример синтеза СМПА U_2 . Вершины ГСА Г1 отмечены состояниями автомата Мура [1]. Анализ ГСА Г1 позволяет найти следующие множества и их параметры: $A = \{a_1, \dots, a_9\}$, $M = 9$, $X = \{x_1, \dots, x_4\}$, $L = 4$, $Y^1 = \{y_1, \dots, y_5\}$, $N_1 = 5$, $Y^2 = \{y_6, \dots, y_{10}\}$, $N_2 = 5$, $N = 10$.

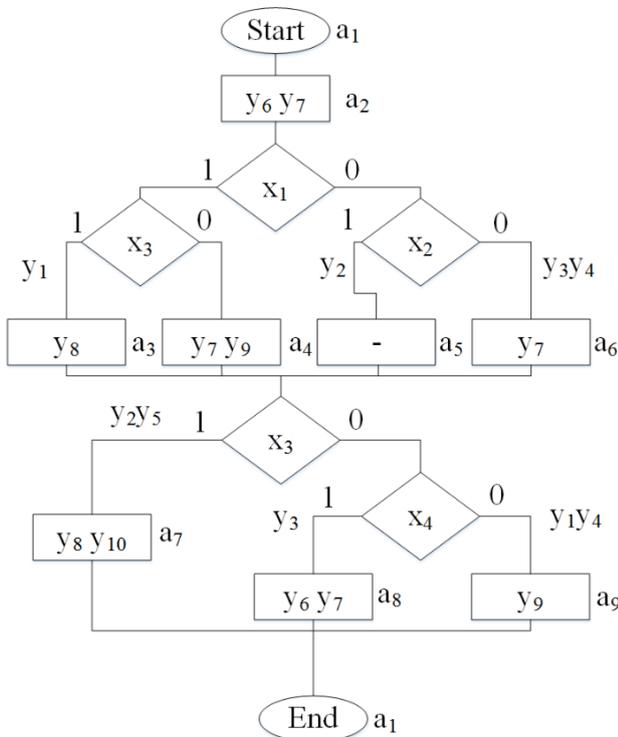


Рис. 3 – Исходная ГСА Г1

Используя определение ПЭС, найдем разбиение $\pi_A = \{B_1, \dots, B_4\}$. Итак, $I = 4$. Используя (5), найдем $R = 4$, что дает множество $T = \{T_1, \dots, T_4\}$ и $\varphi = \{D_1, \dots, D_4\}$. Используя (14), найдем $R_I = 2$, что дает множество $\tau = \{\tau_1, \tau_2\}$.

Для уменьшения числа термов в системе (17) закодируем классы $B_i \in \pi_A$ следующим образом: чем больше состояний включает класс, тем меньше единиц имеет его код. В нашем случае

$$B_1 = \{a_1\}, B_2 = \{a_2\}, B_3 = \{a_3, a_4, a_5\}, \\ B_4 = \{a_7, a_8, a_9\}.$$

Закодируем классы $B_i \in \pi_A$ следующими кодами

$$K(B_3) = 00, K(B_1) = 01, K(B_2) = 10 \text{ и} \\ K(B_4) = 11.$$

Заметим, что класс (B_4) имеет код 11, так как переходы для этого класса не включаются в ПСТ. Это следует из ГСА Г1, для которой нет условных переходов в состояние a_1 из состояний $a_m \in B_4$. Таким образом, код 11 можно использовать для минимизации функций (17).

Для кодирования состояний построим систему функций, задающих связь между классами $B_i \in \pi_A$ (а также переменными $y_n \in Y^2$) и состояниями $a_m \in A$. В нашем случае это система (19):

$$B_1 = A_1; y_7 = A_2 \vee A_4 \vee A_6 \vee A_8; \\ B_2 = A_2; y_8 = A_3 \vee A_7; \\ B_3 = A_3 \vee A_4 \vee A_5 \vee A_6; y_9 = A_4 \vee A_9; \\ y_6 = A_2 \vee A_8; y_{10} = A_7. \tag{19}$$

Пусть для реализации СМПА U_2 используется CPLD, у которых параметры $s = 12$, $q = 3$. Из (13) следует, что оптимизация должна быть выполнена для функций B_3 и y_7 системы (19). Однако код $K(B_3) = 00$, поэтому класс B_3 не вносит термы в функции (17). Итак, оптимизация необходима только для функции y_7 . Закодируем состояния $a_m \in A$ так, как показано в карте Карно (рис. 4).

Состояния a_2, a_4, a_6, a_8 входят в куб 1 карты. Это дает $y_7 = T_2$. Кроме того, имеем $y_8 = T_1$. Следовательно, для реализации функций y_7 и y_8 не используются макроячейки. При этом нужно стремиться, чтобы состояния были закодированы с минимально возможным числом единиц. Это позволяет уменьшить число термов в системе (15).

Построим ПСТ автомата U_2 для нашего примера (табл. 1). Эта таблица имеет $H_1 = 8$ строк.

		$T_1 T_2$			
		00	01	11	10
$T_3 T_4$	00	a_1	a_2	*	a_3
	01	a_5	a_4	*	a_7
	11	*	a_8	*	*
	10	a_9	a_6	*	*

Рис. 4 – Коды состояний СМПА

Отметим, что в эквивалентном автомате U_1 ПСТ имеет $H = 17$ строк. Таким образом, наш подход позволяет в два раза уменьшить число термов в системах $D_r \in \phi$ и $y_n \in Y^1$ (для данного примера).

ПСТ используется для формирования функций (15) и (16). Например, из табл. 1 можно получить следующие функции:

$$D_1 = \overline{\tau_1} \overline{\tau_2} x_1 x_3 \vee \overline{\tau_1} \cdot \overline{\tau_2} x_3 \\ y_1 = \overline{\tau_1} \overline{\tau_2} x_1 x_3 \vee \overline{\tau_1} \cdot \overline{\tau_2} \cdot \overline{x_3} \cdot \overline{x_4}$$

Таблиця 1 – ПСТ автомата U_2

B_i	$K(B_i)$	a_s	$K(a_s)$	X_h	Y_h^1	ϕ^i	h
B_1	01	a_2	0100	1	-	D_2	1
B_2	10	a_3	1000	$x_1 x_3$	y_1	D_1	2
		a_4	0101	$x_1 \overline{x_3}$	-	$D_2 D_4$	3
		a_5	0001	$\overline{x_1} x_2$	y_2	D_4	4
		a_6	0110	$\overline{x_1} \cdot \overline{x_2}$	$y_3 y_4$	$D_2 D_3$	5
B_3	00	a_7	1001	x_3	$y_2 y_5$	$D_1 D_4$	6
		a_8	0111	$\overline{x_3} x_4$	y_3	$D_2 D_3 D_4$	7
		a_9	0010	$\overline{x_3} \cdot \overline{x_4}$	$y_1 y_4$	D_3	8

Системы функций (8) и (17) строятся на основе системы (19), карты Карно (рис. 4) и кодов классов $B_i \in \pi_A$. Уравнение для τ_1 строится на основе уравнения для B_2 , а для τ_2 - на основе B_1 . Итак, имеем следующие функции:

$$\begin{aligned} \tau_1 &= T_2 \overline{T_3} \cdot \overline{T_4}; & y_7 &= T_2; & y_{10} &= T_1 T_4; \\ \tau_2 &= \overline{T_1} \cdot \overline{T_2} \cdot \overline{T_3} \cdot \overline{T_4}; & y_8 &= T_1; \\ y_6 &= T_2 \overline{T_3} \cdot \overline{T_4} \vee T_3 T_4; & y_9 &= T_2 \overline{T_3} T_4 \vee \overline{T_2} T_3; \end{aligned} \quad (20)$$

Для реализации систем (15), (16) достаточно $R+N_1 = 9$ макроячеек PAL, имеющих $q=3$. Для реализации системы (20) достаточно 5 макроячеек, что меньше чем $R_1+N_2=7$. Итак, для реализации схемы U_2 необходимо 14 макроячеек. Отметим, что для кодов (рис. 4) схема PALer1 автомата U_1 имеет 28 макроячеек, а PALer2 – всего 3. Таким образом, применение модели U_2 для нашего примера позволило в 2,2 раза уменьшить число макроячеек PAL по сравнению с моделью U_1 .

Выводы. В настоящей работе предлагается метод синтеза совмещенного МПА по ГСА, ориентированный на базис CPLD с макроячейками типа PAL. Метод основан на использовании классов псевдоэквивалентных состояний.

Применение метода позволяет уменьшить число термов в системе функций возбуждения памяти СМПА. Благодаря использованию ПЭС, число термов уменьшается до величины, соответствующей эквивалентному автомату Мили. Кроме того, кодирование состояний можно выполнить так, чтобы уменьшить аппаратные затраты в схеме, формирующей выходные переменные автомата Мура.

В работе показано, что минимизация подлагает только функции, для которых выполняется условие (13). Анализ библиотеки [16] показал, что это условие выполняется для 17–19 % функций, задающих схему автомата. Этот факт значительно упрощает метод кодирования состояний по сравнению с известными подходами.

Направлением дальнейших исследований является разработка методов синтеза СМПА, позволяющих исключить дополнительные переменные, кодирующие классы ПЭС. Кроме того, интерес представляет случай, когда число входов PAL меньше, чем $R+L$. В этом случае необходимо искать некоторые разбиения на множестве термов [10].

Список литературы:

1. Baranov, S. Logic and System Design of Digital Systems [Text] / S. Baranov. – Tallinn: TUT Press, 2008. – 267 p.
2. Micheli, G. Synthesis and Optimization of Digital Circuits [Text] / G. Micheli. – New York: Mc Graw-Hill, 1994. – 636 p.
3. Sklyarov, V. Synthesis and Optimization of FPGA-Based Systems [Text] / V. Sklyarov, I. Sklyarova, A. Barkalov, L. Titarenko. – Springer International Publishing, 2014. – 432 p. doi: [10.1007/978-3-319-04708-9](https://doi.org/10.1007/978-3-319-04708-9)
4. Sklyarova, I. Design of FPGA – based circuits using Hierarchical Finite State Machines [Text] / I. Sklyarova, V. Sklyarov, A. Sudnitson. – Tallinn: TUT Press, 2012. – 240 p.
5. Соловьев, В. В. Логическое проектирование цифровых систем на основе программируемых логических интегральных схем [Текст] / В. В. Соловьев, А. Климович. – Москва: Горячая Линия–Телеком, 2008. – 376 с.
6. Czerwinski, R. Finite State Machine Logic Synthesis for Complex Programmable Logic Devices [Text] / R. Czerwinski D. Kania. – Springer Science & Business Media, 2013. – 172 p.
7. Баркалов, А. А. Реалізація суміщеного мікропрограмного автомата в базисі FPGA [Текст] / А. А. Баркалов Л. А. Титаренко І. Я. Зеленева // Наукові праці Донецького національного технічного університету. Серія : Інформатика, кібернетика та обчислювальна техніка. – 2015. – № 2. – С. 84–88.
8. Баркалов, А. А. Оптимізація схеми суміщеного автомата в базисі FPGA способом заміни вхідних змінних [Текст] / А. А. Баркалов, Л. А. Титаренко, І. Я. Зеленева, С. С. Грушко // Науковий вісник Чернівецького національного університету імені Юрія Федьковича. Серія: Комп'ютерні системи та компоненти. – 2015. – № 6 (2). – С. 49 – 54.
9. Баркалов, А. А. Способы оптимизации схем совмещенных микропрограммных автоматов в базисе FPGA [Текст]: мат. межн. науч.-практ. конф. / А. А. Баркалов, И. Я. Зеленева, С. С. Грушко, Д. А. Михайлютенко // Современные проблемы и достижения в области радиотехники, телекоммуникаций и информационных технологий, 2016. – С. 281–282.
10. Barkalov, A. Reduction in the Number of PAL Macrocells in the Circuit of a Moore FSM [Text] / A. Barkalov, L. Titarenko, S. Chmielewski // International Journal of Applied Mathematics and Computer Science. – 2007. – № 17 (4). – P. 565–575. doi: [10.2478/v10006-007-0046-8](https://doi.org/10.2478/v10006-007-0046-8)
11. Accelerating the Smart and Connected World [Electronic resource]. – Available at: www.altera.com
12. All programmable [Electronic resource]. – Available at: www.xilinx.com

13. Villa, T. NOVA: state assignment of finite state machines for optimal two-level implementation [Text] / T. Villa, A. Sangiovanni-Vincentelli // IEEE Transactions on CAD of Integrated Circuits and Systems. – 1990. – № 9 (9). – P. 905–924. doi: [10.1109/43.59068](https://doi.org/10.1109/43.59068)
14. Yang, S. Optimum and suboptimum algorithms for input encoding and its relationships to logic minimization [Text] / S. Yang, M. J. Ciesielski // IEEE Transactions on CAD of Integrated Circuits and Systems. – 1991. – № 10 (1). – P. 4–12. doi: [10.1109/43.62787](https://doi.org/10.1109/43.62787)
15. Баркалов, А. А. Принципы оптимизации логической схемы микропрограммного автомата Мура [Текст] / А. А. Баркалов // Кибернетика и системный анализ. – 1998. – № 1. – С. 65–72.
16. Yang, S. Logic Synthesis and optimization benchmarks user guide [Text] / Yang S. – Microelectronics Centre of North Carolina, 1991. – 45 p.
8. Barkalov, A. A., Ty'tarenko, L. A., Zeleneva, I. Ya., Grushko, S. S. (2015). Opty'mizaciya sxemy' sumishhenogo avtomata v bazy'si FPGA sposobom zaminy' vxidny'x zminny'x. Naukovy'j visny'k Chernivec'kogo nacional'nogo universy'tetu imeni Yuriya Fed'kovy'cha. Seriya: Komp'yuterni sy'stemy' ta komponenty', 6 (2), 49–54.
9. Barkalov, A. A., Zeleneva, I. Ja., Grushko, S. S., Mihajljutenko, D. A. (2016). Sposoby optimizacii shem sovmeshhennyh mikroprogrammnyh avtomatov v bazise FPGA. Sovremennye problemy i dostizheniya v oblasti radiotekhniki, telekommunikacij i informacionnyh tehnologij, 281–282.
10. Barkalov, A., Titarenko, L., Chmielewski, S. (2007). Reduction in the Number of PAL Macrocells in the Circuit of a Moore FSM. International Journal of Applied Mathematics and Computer Science, 17 (4), 565–575. doi: [10.2478/v10006-007-0046-8](https://doi.org/10.2478/v10006-007-0046-8)
11. Accelerating the Smart and Connected World. Available at: www.altera.com
12. All programmable. Available at: www.xilinx.com
13. Villa, T., Sangiovanni-Vincentelli, A. (1990). NOVA: state assignment of finite state machines for optimal two-level logic implementation. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 9 (9), 905–924. doi: [10.1109/43.59068](https://doi.org/10.1109/43.59068)
14. Yang, S., Ciesielski, M. J. (1991). Optimum and suboptimum algorithms for input encoding and its relationship to logic minimization. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 10 (1), 4–12. doi: [10.1109/43.62787](https://doi.org/10.1109/43.62787)
15. Barkalov, A. A. (1998). Principy optimizacii logicheskoy shemy mikroprogrammnoy avtomata Mura. Kibernetika i sistemnyy analiz, 1, 65–72.
16. Yang, S. (1991). Logic Synthesis and optimization benchmarks user guide. Microelectronics Centre of North Carolina, 45.

Поступила (received) 06.11.2016

Бібліографічні описи / Библиографические описания / Bibliographic descriptions

Метод уменьшения числа термов при реализации схемы совмещенного микропрограммного автомата в базисе CPLD/ А. А. Баркалов, Л. А. Титаренко И. Я. Зеленева, С. С. Грушко// Вісник НТУ «ХПІ». Серія: Механіко-технологічні системи та комплекси. – Харків : НТУ «ХПІ», 2016. – No 49(1221). – С.25–31. – Бібліогр.: 16 назв. – ISSN 2079-5459.

Метод зменшення числа термів при реалізації схеми сумішеного мікропрограмного автомата в базисі CPLD/ О. О. Баркалов, Л. О. Титаренко, І. Я. Зеленьова, С. С. Грушко// Вісник НТУ «ХПІ». Серія: Механіко-технологічні системи та комплекси. – Харків : НТУ «ХПІ», 2016. – No 49(1221). – С.25–31. – Бібліогр.: 16 назв. – ISSN 2079-5459.

Method of reducing the number of terms for combined finite state machine implementing in CPLD/ A. Barkalov, L. Titarenko, I. Zeleneva, S. Hrushko//Bulletin of NTU “KhPI”. Series: Mechanical-technological systems and complexes. – Kharkov: NTU “KhPI”, 2016. – No 49 (1221).– P.25–31. – Bibliogr.: 16. – ISSN 2079-5459.

Відомості про авторів / Сведения об авторах / About the Authors

Баркалов А. А. – доктор технічних наук, професор, Университет Зеленогурский, Кафедра комп'ютерних наук; ул. Средней школы, 9, г. Зелена Гура, Польща, 65-424, e-mail: a.barkalov@imei.uz.zgora.pl.

Титаренко Л. А. – доктор технічних наук, професор, Университет Зеленогурский, Кафедра комп'ютерних наук; ул. Средней школы, 9, г. Зелена Гура, Польща, 65-424, e-mail: L.Titarenko@iie.uz.zgora.pl.

Зеленева І. Я. – кандидат технічних наук, доцент, Запорожский национальный технический университет, Кафедра комп'ютерних систем и сетей; ул. Жуковського, 64, г. Запорожье, Украина, 69063, e-mail: irina.zeleneva@gmail.com.

Грушко С. С. – аспірант, Запорожский национальный технический университет, Кафедра комп'ютерних систем и сетей; ул. Жуковського, 64, г. Запорожье, Украина, 69063; e-mail: grushkosveta@gmail.com.

Баркалов А. А. – доктор технічних наук, професор, Университет Зеленогурський, Кафедра комп'ютерних наук; вул. Середньої школи, 9, м. Зелена Гура, Польща, 65-424, e-mail: a.barkalov@imei.uz.zgora.pl.

Титаренко Л. А. – доктор технічних наук, професор, Университет Зеленогурський, Кафедра комп'ютерних наук; вул. Середньої школи, 9, м. Зелена Гура, Польща, 65-424, e-mail: L.Titarenko@iie.uz.zgora.pl.

Зеленева І. Я. – кандидат технічних наук, доцент, Запорізький національний технічний університет, Кафедра комп'ютерних систем та мереж; вул. Жуковського, 64, Запоріжжя, Україна, 69063, e-mail: irina.zeleneva@gmail.com.

Грушко С. С. – аспірант, Запорізький національний технічний університет, Кафедра комп'ютерних систем та мереж; вул. Жуковського, 64, Запоріжжя, Україна, 69063; e-mail: grushkosveta@gmail.com.

Barkalov A. – Doctor of Technical Sciences, Professor, University of Zielona Góra, Department of Computer Science, Electrotechnics and Informatics; Licealna, 9, Zielona Góra, Poland, 65-424, e-mail: a.barkalov@imei.uz.zgora.pl.

Titarenko L. – doctor of Technical Sciences, Professor, University of Zielona Góra, Institute of Metrology, Electrotechnics and Informatics; Licealna, 9, Zielona Góra, Poland, 65-424, e-mail: L.Titarenko@iie.uz.zgora.pl.

Zeleneva I. – PhD, associate professor, Zaporizhzhya National Technical University, Department of Computer Systems and Networks; Zhukovskogo str., 64, Zaporozhye, Ukraine, 69063, e-mail: irina.zeleneva@gmail.com.

Hrushko S. – graduate student, Zaporizhzhya National Technical University, Department of Computer Systems and Networks; Zhukovskogo str., 64, Zaporozhye, Ukraine, 69063; e-mail: grushkosveta@gmail.com.

УДК 615.849.19

В. Й. КОТОВСЬКИЙ, В. В. ШЛИКОВ, В. А. ДАНИЛОВА

THE MATHEMATICAL MODEL OF HEAT TRANSFER IN MATLAB FOR DEVICE OF CARDIO-PULMONARY BYPASS IN CARDIOLOGY

Запропоновано модель теплопередачі для пристрою штучного кровообігу у вигляді системи диференціальних рівнянь. Чисельна модель дозволяє оцінити температурні параметри гіпотермії і гіпертермії серця і досліджувати зміни градієнта температури на поверхні міокарда в момент реєстрації теплових зображень серця.

Ключові слова: термограма; міокард; розподіл температури; судинна патологія.

Предложена модель теплопередачи для устройства искусственного кровообращения в виде системы дифференциальных уравнений. Численная модель позволяет оценить температурные параметры гипотермии и гипертермии сердца и исследовать изменения градиента температуры на поверхности миокарда в момент регистрации тепловых изображений сердца.

Ключевые слова: термограмма; миокард; распределение температуры; сосудистая патология.

The heat transfer models for device of cardiopulmonary bypass in the form of a system of ordinary differential equations are presented. The numerical model allows us to estimate the temperature of the process parameters of hypothermia and hyperthermia heart and to investigate changes in the gradient of the temperature on the surface of the myocardium at the time of registration of thermal images of the heart.

The numerical model of heat transfer during extracorporeal circulation in the form of a system of ordinary differential equations was created. It allows to estimate the parameters of temperature propagation and temperature gradients at the surface of the myocardium at the time of registration of thermal images of the heart.

Comparison of the model with real patients intraoperative infra-red thermography shows that this method can provide additional important information regarding temperature and vascular uniformity in time of myocardial cooling and heating. This method increasing safety of myocardium protection and assistive in solutions of weaning from cardiopulmonary bypass.

Keywords: thermogram; myocardium; temperature distribution; vascular pathology.

Introduction. The heat transfer during extracorporeal cardiopulmonary bypass (CPB) is due to the heat exchange between the blood and the water in the heat exchanger device of cardiopulmonary bypass (DCB) and to the heat exchange between the blood and the body of the patient's in the circulatory system. In accordance with the protocol of CPB blood is first using a special catheter enters the oxygenator to the pump (centrifuge pump) which replaces lung function, and then from the oxygenator blood moves to the heat exchanger, which lowers the blood temperature and across a catheter (silicone tube) directed in the patient's system circulation [1].

For the CPB the blood flow is maintained at 2,2 – 2,4 $\frac{L}{min \cdot m^2}$ and temperature at 28 – 35 °C during moderate hypothermia, and 16 – 26 °C during deep hypothermia. Reduction of the patient's body temperature leads to reduced oxygen demand for bodies. To perform the heat balance the heat exchange in the body must be equal to its heat transfer. At rest, the magnitude of the human body heat is about 75kcal/g at about 36,6 °C. When heating blood the amount of heat that is absorbed or excreted of blood, calculated using the formula:

$$Q_{lic} = c_{liq} m_{liq} (T_{lic} - T_{out}), J,$$

where c_{liq} – the specific heat capacity of the blood, $c_{liq} \approx 4200 \frac{J}{kg \cdot s}$, $(T_{lic} - T_{out})$ – the temperature difference at heating blood, °C, m_{liq} – the mass of cooled blood, kg.

The heat balance equation involving DCB for cooling and heating of blood will look:

$$Q_{lic} + Q_{heart} = Q_{dbc},$$

where Q_{heart} – the amount of heat emitted or absorbed by the myocardium (heart) for cooling or heating of blood, J. Q_{dbc} – the amount of heat that is released or absorbed in the DCB, J.

Obviously, the combustion of the blood the temperature difference is $(T_{lic} - T_{out}) > 0$ and correspondingly the amount of heat is $Q_{lic} > 0$, and when the cooling blood the temperature difference is $(T_{lic} - T_{out}) < 0$ and $Q_{lic} < 0$ – blood releases energy in the DCB.

© V. Kotovskyi, V. Shlykov, V. Danilova. 2016