

А.А. Баркалов, Я.Е. Визор, А.В. Матвиенко

Реализация схемы устройства управления на *FPGA*

Предложена реализация схемы устройства управления на *FPGA* для цифровых устройств, для чего использованы особенности автомата Мура и базиса *FPGA*, что приводит к сокращению числа необходимых *LUT*-элементов.

Implementation of the scheme of the control device on *FPGA* for digital devices is suggested. To implement the scheme used by Moore and features automatic basis *FPGA*, which leads to a decrease in the number of *LUT*-elements required for the implementation of the scheme.

Запропоновано реалізацію схеми пристрою керування на *FPGA* для цифрових пристроїв. Для реалізації схеми використовуються особливості автомата Мура і базису *FPGA*, що призводить до зменшення числа необхідних *LUT*-елементів.

Введение. Цифровые устройства часто реализуются в базисе программируемых логических интегральных схем (ПЛИС) и содержат устройство управления (УУ), которое координирует работу всех элементов системы [1]. В качестве модели для построения УУ часто используется модель автомата Мура, поэтому актуальна задача уменьшения аппаратных затрат логической схемы автомата Мура, а значит и всей схемы цифрового устройства.

Сегодня наиболее популярны микросхемы ПЛИС с архитектурой программируемых вентиляционных матриц (*FPGA*, *field-programmable logic arrays*). При этом *FPGA* используются для реализации нерегулярной логики, например систем булевых функций [2, 3] и в большинстве случаев включают универсальные табличные элементы типа *LUT* (*look-up table*) с ограниченным числом входов [2]. Такое ограничение приводит к необходимости функциональной декомпозиции реализуемых функций, что связано с увеличением числа уровней схемы, влекущим увеличение времени такта, а также усложнение задач размещения и трассировки [2, 4]. Для устранения этих негативных явлений необходимо уменьшать число аргументов реализуемых функций.

Для уменьшения аппаратных затрат схемы автомата Мура могут быть использованы такие особенности модели автомата, как зависимость выходных сигналов только от состояний и возможность ввода псевдоэквивалентных состояний. В статье предлагается метод синтеза автомата Мура, использующий как особенности модели этого автомата, так и особенности базиса *FPGA*. Их сочетание позволяет

уменьшить число *LUT*-элементов, необходимых для реализации схемы.

Модели автомата Мура

Пусть автомат Мура задан прямой структурной таблицей (ПСТ) [1] со столбцами: a_m – исходное состояние, входящее в множество состояний $A = \{a_1, \dots, a_M\}$; $K(a_m)$ – код состояния $a_m \in A$; a_s – состояние перехода; $K(a_s)$ – код состояния $a_s \in A$; X_h – конъюнкция входных сигналов (логических условий), входящих в множество $X = \{x_1, \dots, x_L\}$, которая определяет переход $\langle a_m, a_s \rangle$; Φ_h – набор функций возбуждения памяти, входящих во множество $\Phi = \{D_1, \dots, D_R\}$ и принимающих единичное значение для переключения памяти из $K(a_m)$ в $K(a_s)$; $h = \overline{1, H}$ – номер перехода. Кроме того, в столбце a_m записывается выходной набор $Y(a_m) \subseteq Y$, формируемый в состоянии a_m . Здесь $Y = \{y_1, \dots, y_N\}$ – множество микроопераций. Эта таблица – основа для формирования систем булевых функций:

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

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

В системах (1) – (2) множество $T = \{T_1, \dots, T_R\}$ включает внутренние переменные, используемые для кодирования состояний автомата, при этом их количество определяется формулой

$$R = \lceil \log_2 M \rceil. \quad (3)$$

Структурная схема автомата Мура U_1 показана на рис. 1.

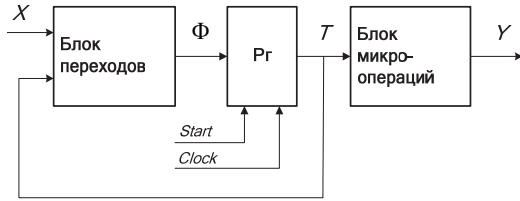


Рис. 1. Структурная схема автомата Мура U_1

В автомате U_1 блок переходов (БП) реализует систему (1), а блок микроопераций (БМО) – систему (2). Схемы обоих блоков реализуются на LUT -элементах. Память состояний автомата реализуется на регистре Рг, имеющем R триггеров. По сигналу *Start* регистр обнуляется, и автомат переходит в начальное состояние. По сигналу *Clock* в Рг записывается код состояния перехода.

Состояния $a_m, a_p \in A$ называются псевдоэквивалентными (ПЭС), если ими отмечены операторные вершины граф-схемы алгоритма (ГСА) управления, выходы которых соединены с входом одной и той же вершины ГСА. Такие состояния относятся к одному классу разбиения $\Pi_A = \{B_1, \dots, B_l\}$ множества состояний, основанному на отношении псевдоэквивалентности [5, 6]. При этом множество Π_A может быть представлено в виде объединения множеств:

$$\Pi_A = \Pi_B \cup \Pi_C. \quad (4)$$

Пусть $I(B_i)$ – число обобщенных интервалов булевого пространства, представляющих код $K(B_i)$ класса $B_i \in \Pi_A$. Множества Π_B и Π_C формируются следующим образом:

$$\begin{aligned} I(B_i) = 1 &\rightarrow B_i \in \Pi_B; \\ I(B_i) > 1 &\rightarrow B_i \in \Pi_C. \end{aligned} \quad (5)$$

Если $\Pi_C = \emptyset$, то автомат Мура реализуется в виде модели U_1 . Если $\Pi_B = \emptyset$, то автомат реализуется в виде модели U_2 (рис. 2).

В автомате U_2 используется кодирование классов $B_i \in \Pi_C$ кодами $K(B_i)$ разрядности

$$R_C = \lceil \log_2 I_C \rceil, \quad (6)$$

где $I_C = |\Pi_C|$.

Для автомата U_2 $I_C = I$, так как $\Pi_C = \Pi_A$. Для кодирования классов используются пере-

менные $\tau_r \in \tau$, где $|\tau| = R_C$. Блок преобразователя кодов (БПК) используется для преобразования кодов $K(a_m)$ в коды $K(B_i)$. Блок БП реализует систему функций

$$\Phi = \Phi(\tau, X), \quad (7)$$

блок БПК – систему функций

$$\tau = \tau(T), \quad (8)$$

а блок БМО – систему (2).

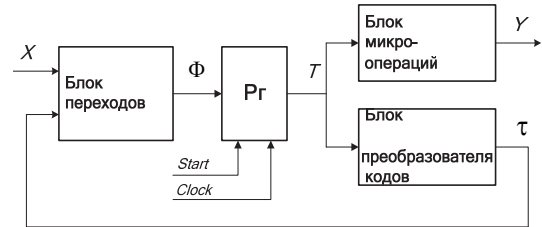


Рис. 2. Структурная схема автомата Мура U_2

В общем случае при выполнении условия $\Pi_B \neq \Pi_C \neq \emptyset$ для реализации автомата Мура предлагается модель U_3 (рис. 3).

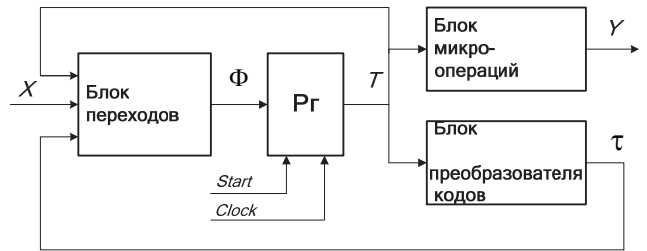


Рис. 3. Структурная схема автомата Мура U_3

В автомате U_3 используется два источника кодов классов $B_i \in \Pi_A$, что позволяет уменьшить число переменных в термах для оптимальной реализации на LUT -элементах. Коды классов $B_i \in \Pi_B$ поступают из регистра Рг и представляются переменными $T_r \in T$. Коды классов $B_i \in \Pi_C$ поступают из БПК и представляются переменными $\tau_r \in \tau$. При этом параметр R_C определяется как

$$R_C = \lceil \log_2 (I_C + 1) \rceil. \quad (9)$$

Единица в (8) добавляется для учета необходимости кодирования ситуации $B_i \in \Pi_C$. При этом блок БП реализует функции

$$\Phi = \Phi(T, \tau, X). \quad (10)$$

Таким образом, для синтеза автомата Мура в базе *FPGA* могут быть использованы три модели, различающиеся количеством источников кодов состояний в блоке переходов и наличием блока преобразователя кодов. При этом выбор конкретной модели обуславливается видом разбиения (4). Кроме того, необходимо отметить, что в процессе синтеза автоматов $U_1 - U_3$ кодирование состояний $a_m \in A$ может быть выполнено различными способами. Например, могут быть использованы способы кодирования, подробно рассмотренные в работе [7].

Обобщенный метод синтеза автомата Мура

В общем случае метод синтеза автомата Мура в базе *FPGA* включает следующие основные этапы:

- формирование множества Π_A ;
- выполнение разбиения $\Pi_A = \Pi_B \cup \Pi_C$;
- формирование систем $B = B(A)$ и $Y = Y(A)$;
- выбор структуры автомата Мура:
 - $\Pi_C = \emptyset$ – модель U_1 ;
 - $\Pi_B = \emptyset$ – модель U_2 ;
 - $\Pi_B \neq \Pi_C \neq \emptyset$ – модель U_3 ;
- кодирование состояний автомата;
- формирование содержимого блока БМО;
- кодирование классов ПЭС $B_i \in \Pi_C$;
- формирование МПСТ автомата Мура;
- формирование содержимого блока БП;
- формирование таблицы блока БПК.

Пример использования предложенного метода

Рассмотрим пример использования предложенного метода. Пусть автомат Мура S_1 задан ГСА Γ_1 (рис. 4).

Сформируем множество Π_A : $\Pi_A = \{B_1, B_2, B_3, B_4\}$, где $B_2 = \{a_2, a_3, a_4, a_5\}$, $B_3 = \{a_6, a_7, a_8\}$, $B_4 = \{a_9, a_{10}\}$. Класс B_1 соответствует начальному и конечному состоянию автомата. Выполним разбиение $\Pi_A = \Pi_B \cup \Pi_C$: $\Pi_B = \emptyset$, $\Pi_C = \Pi_A = \{B_2, B_3, B_4\}$ – следовательно, для

синтеза необходимо использовать модель U_2 . Далее выполним оптимальное кодирование состояний автомата S_1 (рис. 5).

Сформируем систему (2) для автомата S_1 :

$$\begin{aligned} y_1 &= a_2 \vee a_3 \vee a_4 \vee a_{10}; \\ y_2 &= a_2 \vee a_5 \vee a_7 \vee a_8 \vee a_{10}; \\ y_3 &= a_3 \vee a_4 \vee a_6 \vee a_7 \vee a_8 \vee a_9. \end{aligned} \quad (11)$$

На основе (11) и рис. 5 получим систему формул для блока БМО:

$$\begin{aligned} y_1 &= \bar{T}_1 \bar{T}_3 \bar{T}_4 \vee T_1 \bar{T}_2; \\ y_2 &= \bar{T}_1 T_2 \bar{T}_3 \bar{T}_4 \vee \bar{T}_1 T_2 T_3 \bar{T}_4 \vee T_1 T_2 \bar{T}_3 T_4 \vee T_1 \bar{T}_2; \\ y_3 &= T_1 T_2 \bar{T}_3 \vee \bar{T}_1 T_2 T_4 \vee T_1 \bar{T}_2 \bar{T}_3 \bar{T}_4 \vee T_1 \bar{T}_2 T_3 \bar{T}_4. \end{aligned} \quad (12)$$

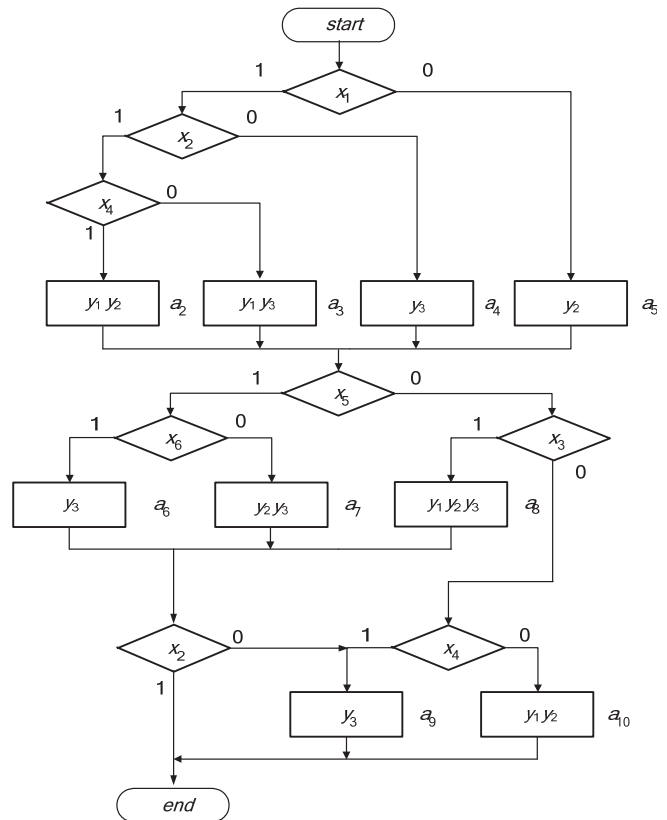


Рис. 4. ГСА Γ_1

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

Рис. 5. Кодирование состояний автомата S_1

Для автомата S_1 $I_C = 3$, $R_C = 2$, следовательно, для представления классов $B_i \in \Pi_A$ необходимо два разряда: $\tau = \{\tau_1, \tau_2\}$. Выполним кодирование классов следующим образом: $K(B_2) = 01$, $K(B_3) = 10$, $K(B_4) = 11$, классу B_1 соответствует код $K(B_1) = 00$. Сформируем таблицу блока БПК (табл. 1).

Таблица 1. Таблица блока БПК автомата S_1

a_m	$K(a_m)$	B_i	$K(B_i)$	τ_m	m
a_2, a_3, a_4, a_5	01**	B_2	01	τ_2	1
a_6, a_7, a_8	11**	B_3	10	τ_1	2
a_9, a_{10}	10**	B_4	11	$\tau_1\tau_2$	3

На основе табл. 1 с учетом минимизации сформируем систему (8) содержимого блока БПК:

$$\begin{aligned} \tau_1 &= T_1 T_2 \vee T_1 \bar{T}_2; \\ \tau_2 &= \bar{T}_1 T_2 \vee T_1 \bar{T}_2. \end{aligned} \quad (13)$$

Построим модифицированную ПСТ для автомата S_1 (табл. 2).

Таблица 2. МПСТ автомата S_1

B_i	$K(B_i)$		a_s	$K(a_s)$	X_h	Φ_h	h
	τ_1	τ_2					
B_1	0	0	a_5	0110	\bar{X}_1	$D_2 D_3$	1
			a_4	0111	$X_1 \bar{X}_2$	$D_2 D_3 D_4$	2
			a_3	0101	$X_1 X_2 \bar{X}_4$	$D_2 D_4$	3
			a_2	0100	$X_1 X_2 X_4$	D_2	4
B_2	0	1	a_6	1100	$X_5 X_6$	$D_1 D_2$	5
			a_7	1101	$X_5 \bar{X}_6$	$D_1 D_2 D_4$	6
			a_8	1000	$\bar{X}_5 X_3$	D_1	7
			a_9	1110	$\bar{X}_5 \bar{X}_3 X_4$	$D_1 D_2 D_3$	8
			a_{10}	1010	$\bar{X}_5 \bar{X}_3 \bar{X}_4$	$D_1 D_3$	9
B_3	1	0	a_1	0000	X_2	–	10
			a_9	1110	\bar{X}_2	$D_1 D_2 D_3$	
B_4	1	1	a_1	0000	1	–	11

На основе табл. 2 формируется система (7) блока БП:

$$\begin{aligned} D_1 &= \bar{\tau}_1 \tau_2 \vee \tau_1 \bar{\tau}_2 \bar{x}_2; \\ D_2 &= \bar{\tau}_1 \bar{\tau}_2 \vee \bar{\tau}_1 \tau_2 x_5 \vee \bar{\tau}_1 \tau_2 \bar{x}_3 \bar{x}_4 \vee \tau_1 \bar{\tau}_2 \bar{x}_2; \\ D_3 &= \bar{\tau}_1 \bar{\tau}_2 \bar{x}_1 \vee \bar{\tau}_1 \bar{\tau}_2 x_1 \bar{x}_2 \vee \bar{\tau}_1 \tau_2 \bar{x}_5 \bar{x}_3 \vee \tau_1 \bar{\tau}_2 \bar{x}_2; \\ D_4 &= \bar{\tau}_1 \bar{\tau}_2 x_1 \bar{x}_2 \vee \bar{\tau}_1 \bar{\tau}_2 x_1 x_2 \bar{x}_4 \vee \bar{\tau}_1 \tau_2 x_5 \bar{x}_6 \vee \tau_1 \bar{\tau}_2 \bar{x}_2. \end{aligned} \quad (14)$$

Далее реализация схемы автомата S_1 сводится к реализации систем (12), (13) и (14) на LUT-элементах в базисе *FPGA*.

Заключение. При синтезе схемы автомата Мура в базисе *FPGA* могут использоваться модели автомата как с одним, так и с несколькими источниками кодов состояний. Это необходимо для рационального использования малого числа входов в элементах *LUT*. В качестве источников кодов могут использоваться регистр памяти и блок преобразователя кодов. Сочетание нескольких типов моделей автомата Мура и способов кодирования состояний порождает различные варианты логической схемы, имеющей разное число уровней. Выбор лучшей модели не может быть осуществлен априорно. Поэтому предлагается использовать общий алгоритм, основанный на использовании всех трех методов кодирования состояний. При этом для каждого метода лучшая модель выбирается по формальным признакам, а затем определяется лучшая из моделей для каждого метода.

1. Baranov S. Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 p.
2. Maxfield C. The Design Warriors Guide to FPGAs. – Elsevier, 2004. – 541 p.
3. Грушницкий П.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем с использованием микросхем программируемой логики. – СПб.: БХВ-Петербург, 2002. – 608 с.
4. Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.
5. Баркалов А.А. Принципы оптимизации логической схемы микропрограммного автомата Мура // Кибернетика и системный анализ. – 1998. – № 1. – С. 65–72.
6. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. – Донецк: ДонГУ, 2002. – 262 с.
7. Баркалов А.А., Матвиенко А.В., Цололо С.А. Оптимизация схемы автомата Мура на однородных ПЛИС // Комп'ютерні засоби, мережі та системи: Зб. наук. праць. Ін-т кібернетики ім. В.М. Глушкова НАН України, 2009. – № 8. – С. 45–51.

Поступила 06.08.2012

Тел. для справок: +38 044 526-2504 (Киев)

© А.А. Баркалов, Я.Е. Визор, А.В. Матвиенко, 2013