

КОМП'ЮТЕРНІ ЗАСОБИ, МЕРЕЖІ ТА СИСТЕМИ

A.A. Barkalov, A.V. Matvienko,
S.A. Tsololo

OPTIMIZATION OF MOORE FSM LOGIC CIRCUIT WITH HOMOGENEOUS CPLD

The method of Moore FSM logic circuit optimization is proposed. The method is based on three different approaches for FSM state encoding. The common algorithm is proposed for choice of the best FSM model for given conditions.

Запропоновано метод оптимізації логічної схеми автомата Мура, заснований на використанні трьох різних способів кодування станів автомата. Наведено алгоритм вибору оптимальної структури для конкретного автомата.

Предлагается метод оптимизации логической схемы автомата Мура, основанный на использовании трех различных способов кодирования состояний автомата. Приведен алгоритм выбора оптимальной структуры для конкретного автомата.

© А.А. Баркалов, А.В. Матвиенко,
С.А. Цололо, 2009

УДК 681.324

А.А. БАРКАЛОВ, А.В. МАТВИЕНКО,
С.А. ЦОЛОЛО

ОПТИМИЗАЦИЯ СХЕМЫ АВТОМАТА МУРА НА ОДНОРОДНЫХ ПЛИС

Введение. При синтезе схем устройств управления часто используется модель микропрограммного автомата (МПА) Мура [1]. Одним из современных базисов, используемых для реализации схем МПА, являются программируемые логические интегральные схемы (ПЛИС) [2, 3]. Для уменьшения аппаратных затрат в схеме МПА необходимо учитывать как особенности его модели, так и особенности элементного базиса [4]. В настоящей работе рассматривается задача реализации схемы МПА Мура в базисе ПЛИС, в основе которых находятся макроячейки программируемой матричной логики (ПМЛ). Назовем такие ПЛИС однородными в отличие от микросхем, в состав которых входят и макроячейки ПМЛ, и встроенные блоки памяти [5]. Для оптимизации схемы МПА Мура предлагается использовать следующие особенности модели и базиса: наличие классов псевдоэквивалентных состояний МПА Мура; зависимость выходных сигналов только от состояний МПА; большой коэффициент объединения по входу макроячеек ПМЛ.

Характеристика 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 $a_m \in A$ разрядности $R = \lceil \log_2 M \rceil$; 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 $K(a_s)$; $h = 1, \dots, H$ – 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 является основой для формирования систем булевых функций:

$$\Phi = \Phi(T, X), \tag{1}$$

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

В системах (1)–(2) множество $T = \{T_1, \dots, T_R\}$ включает внутренние переменные, используемые для кодирования состояний МПА. Системы (1)–(2) определяют структурную схему МПА Мура U_1 (рис. 1).

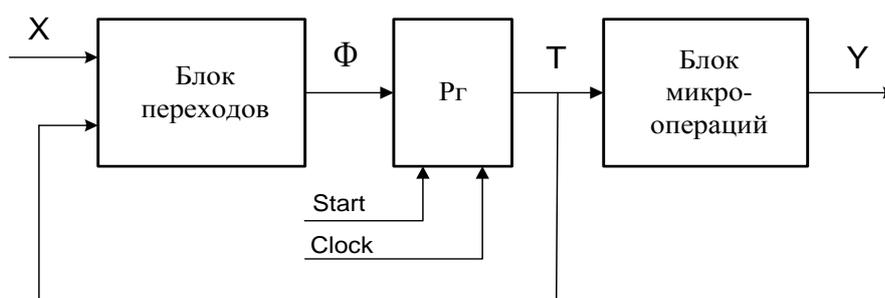


РИС. 1. Структурная схема автомата Мура U_1

В МПА U_1 блок переходов (БП) реализует систему (1), а блок микроопераций (БМО) – систему (2). Схемы обоих блоков реализуются на макроячейках ПМЛ. Память состояний МПА реализуется на регистре Рг, имеющем R триггеров. По сигналу Start регистр обнуляется, при этом МПА переходит в начальное состояние. По сигналу Clock в Рг записывается код состояния перехода.

Основной недостаток модели U_1 – число строк ПСТ (H_1) превышает число H_0 строк ПСТ эквивалентного автомата Мили [1]. При этом для реализации системы (1) требуется больше макроячеек, чем в эквивалентном МПА Мили. Для устранения этого недостатка можно использовать оптимальное кодирование состояний либо преобразование кодов состояний в коды классов псевдоэквивалентных состояний [6]. Оба метода основаны на наличии классов псевдоэквивалентных состояний $B_i \in \Pi_A$ с мощностью, большей двух [7, 8]. Состояния $a_m, a_p \in A$ называются псевдоэквивалентными, если ими отмечены операторные вершины алгоритма управления [1], выходы которых соединены со входом од-

ной и той же вершины алгоритма. Такие состояния относятся к одному классу разбиения $\Pi_A = \{B_1, \dots, B_l\}$ множества состояний, основанному на отношении псевдоэквивалентности [6]. При оптимальном кодировании каждый класс $B_i \in \Pi_A$ представляется минимально возможным числом обобщенных интервалов R -мерного булева пространства, а в пределе – одним. При этом длина ПСТ (число строк в ней) может быть уменьшена до величины H_0 .

Следует отметить, что такой подход не гарантирует уменьшения числа термов в системе (2) и число макроячеек в блоке БМО может быть далеко от оптимального. Для уменьшения аппаратных затрат в блоке БМО можно использовать метод специального кодирования состояний, ориентированный на уменьшение числа термов $Q(y_n)$ в каждой из функций $y_n \in Y$ [9]. При этом возможно уменьшение числа уровней в комбинационной схеме блока БМО, что уменьшает длительность такта работы МПА. В предельном случае общее число макроячеек в блоке БМО равняется числу микроопераций N .

Этот подход не гарантирует уменьшения числа термов в системе (1). При этом увеличивается число макроячеек в блоке БП по сравнению с этой характеристикой для метода оптимального кодирования. В настоящей работе для решения этой задачи предлагается использовать большой коэффициент объединения по входу макроячеек ПМЛ. Вначале рассматривается решение этой задачи при оптимальном и специальном кодировании состояний, а затем предлагается метод комбинированного кодирования состояний.

Уменьшение аппаратных затрат при оптимальном кодировании состояний. Закодируем состояния $a_m \in A$ оптимальным образом, для чего может быть использован, например, алгоритм *ESPRESSO* [10]. При этом множество Π_A разбивается на множества Π_B и Π_C . Пусть $I(B_i)$ – число обобщенных интервалов булева пространства, представляющее код $K(B_i)$ класса $B_i \in \Pi_A$. Множества Π_B и Π_C формируются следующим образом:

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

Если $\Pi_C = \emptyset$, то автомат Мура реализуется в виде модели U_1 . Однако в этой модели число строк ПСТ равно H_0 . Этим параметром определяется суммарное число уникальных термов в системе (1). Если $\Pi_B = \emptyset$, то автомат реализуется в виде модели U_2 (рис. 2).

В МПА U_2 используется кодирование классов $B_i \in \Pi$ кодами $K(B_i)$ разрядности $R_1 = \lceil \log_2 I_C \rceil$, где $I_C = |\Pi_C|$. Для автомата U_2 $I_C = I$, так как $\Pi_A = \Pi_C$. Для кодирования классов используются переменные $\tau_r \in \tau$, где $|\tau| = R_1$.

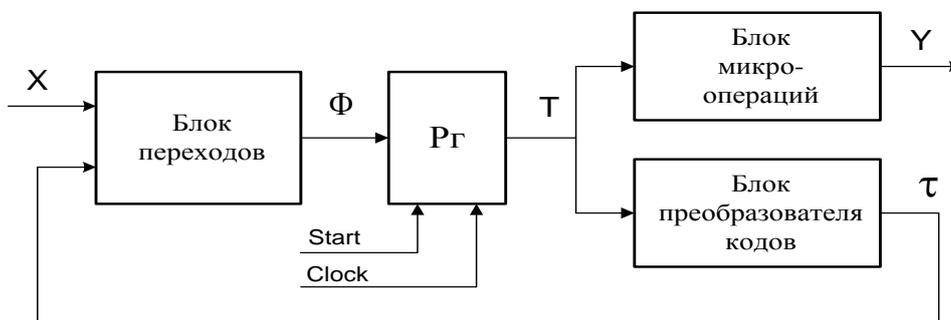


РИС. 2. Структурная схема автомата Мура U_2

Блок преобразователя кодов (БПК) используется для преобразования кодов $K(a_m)$ в коды $K(B_i)$. Блок БП реализует систему функций $\Phi = \Phi(\tau, X)$, блок БПК – систему функций $\tau = \tau(T)$, а блок БМО – систему (2). Такой подход гарантирует уменьшение параметра N_1 до N_0 , но связан с введением блока БПК, который потребляет ресурсы ПЛИС.

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

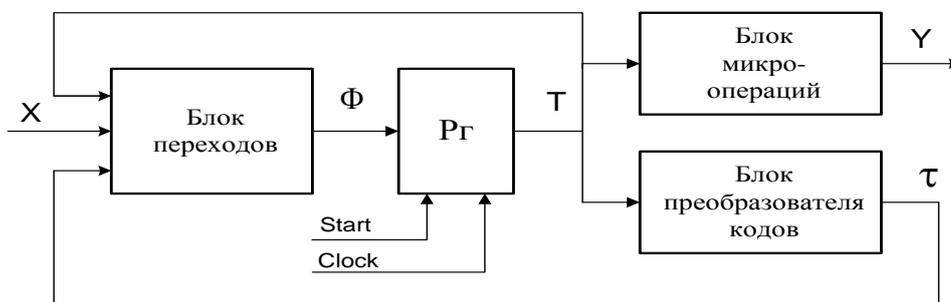


РИС. 3. Структурная схема автомата Мура U_3

В автомате U_3 используется два источника кодов классов $B_i \in \Pi_A$, что возможно благодаря большому коэффициенту объединения по входу макроячеек ПМЛ [11, 12]. 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_1 определяется как $R_1 = \lceil \log_2(I_C + 1) \rceil$. Здесь единица добавляется для учета необходимости кодирования ситуации $B_i \notin \Pi$. При этом блок БП реализует функции $\Phi = \Phi(T, \tau, X)$.

Такой подход гарантирует уменьшение параметра H_1 до H_0 , при этом число макроячеек в схеме БПК будет меньше, чем для модели U_2 . Однако данный подход не гарантирует уменьшение сложности блока БМО.

Уменьшение аппаратурных затрат при специальном кодировании состояний. Как уже отмечалось, специальное кодирование состояний ориентировано на оптимизацию блока БМО. Возможны три ситуации для классов $B_i \in \Pi_A$:

– $\Pi_C = \emptyset$, $\Pi_A = \Pi_B$. Автомат представляется моделью U_1 , в которой $H_1 = H_0$. Такой случай соответствует схеме с минимальными аппаратурными затратами, но его вероятность очень мала;

– $\Pi_B = \emptyset$, $\Pi_A = \Pi_C$. Используется модель U_2 , где блок БПК потребляет максимальное количество ресурсов. Этот случай также маловероятен;

– $\Pi_B \neq \emptyset$, $\Pi_C \neq \emptyset$. Используется модель U_3 , где аппаратурные затраты в схеме БМО минимальны. Общие затраты зависят от того, какая часть классов $B_i \in \Pi_A$ представлена блоком БПК. Это наиболее общий случай.

Отметим, что для специального кодирования состояний можно использовать метод *ESPRESSO* [10].

Комбинированное кодирование состояний. Оптимальное кодирование состояний ориентировано на оптимизацию только блока БП. Специальное кодирование состояний ориентировано на оптимизацию только блока БМО. Назовем комбинированным кодированием состояний подход, ориентированный на оптимизацию схемы автомата Мура в целом.

Построим систему функций $B = B(A)$, которую можно представить в виде

$$B_i = \bigvee_{m=1}^M C_{mi} a_m \quad (i=1, \dots, I). \quad (3)$$

Здесь булевская переменная $C_{mi} = 1$, если и только если $a_m \in B_i$. Аналогично может быть представлена и система микроопераций:

$$y_n = \bigvee_{m=1}^M C_{mn} a_m \quad (n=1, \dots, N), \quad (4)$$

где булевская переменная $C_{mn} = 1$, если и только если микрооперация $y_n \in Y$ формируется в состоянии $a_m \in A$. Закодируем состояния $a_m \in A$ таким образом, чтобы системы (3) и (4) имели минимально возможное число термов. Для этой цели может быть использован алгоритм *ESPRESSO* [10].

Пусть q – число термов, реализуемых одной макроячейкой ПМЛ. Очевидно, что число $Q(y_n) = q$ будет оптимальным. Нет смысла в дальнейшем уменьшении $Q(y_n)$, так как это не ведет к уменьшению аппаратурных затрат. Этот параметр (q) можно использовать в качестве ограничения для системы (4). Оптимальным результатом для системы (3) является такой, когда все ее функции представляются в виде одной конъюнкции внутренних переменных $T_r \in T$. Отметим, что в настоящее время такой алгоритм нам неизвестен.

Как и для предыдущих методов кодирования, возможны три ситуации ($\Pi_B = \emptyset$, $\Pi_C = \emptyset$, $\Pi_B \neq \Pi_C \neq \emptyset$). При этом для реализации схемы используются модели $U_1 - U_3$, в которых все блоки имеют некоторые усредненные затраты.

Общий метод синтеза автомата Мура. Поскольку заранее неизвестно, какой из методов кодирования состояний даст оптимальные результаты для конкретного автомата S_1 , необходимо использовать все методы и выбрать из них лучший. Для каждого из методов необходимо выбрать лучшую модель, для чего используется следующая процедура P:

1. Кодирование состояний $a_m \in A$.
2. Формирование множеств Π_B и Π_C .
3. Если $\Pi_C = \emptyset$, то синтезируется модель U_1 .
4. Если $\Pi_B = \emptyset$, то синтезируется модель U_2 .
5. Если $\Pi_B \neq \Pi_C \neq \emptyset$, то синтезируется модель U_3 .

Синтез схемы для модели U_j ($j=1,2,3$) предполагает построение таблиц, задающих отдельные блоки модели, формирование систем булевых функций для этих блоков и реализация схемы с использованием стандартных промышленных пакетов типа *WebPack* [11].

Заключение. Предлагаемый метод реализации автомата Мура в базе ПЛИС с ячейками ПМЛ основан на использовании трех методов кодирования состояний. Метод оптимального кодирования ориентирован на уменьшение числа макроячеек в схеме, формирующей функции возбуждения памяти автомата. Метод специального кодирования состояний ориентирован на уменьшение аппаратных затрат в схеме формирования микроопераций. При комбинированном кодировании состояний упрощаются схемы обоих блоков.

В зависимости от характеристик автомата и ПЛИС в каждом случае возможно применение одной из трех моделей автомата. Эти модели отличаются источниками кодов классов псевдоэквивалентных состояний, в качестве которых может использоваться регистр памяти или преобразователь кодов состояний. Большой коэффициент объединения по входу макроячеек ПМЛ позволяет использовать оба этих источника одновременно.

Необходимо отметить, что лучшая модель не может быть выбрана априорно. Поэтому в работе предлагается общий алгоритм, основанный на использовании всех трех методов кодирования состояний. Отметим, что разные модели автоматов могут иметь разное число уровней в их логических схемах. При этом чем меньше макроячеек используется в схеме, тем меньше уровней она включает. Таким образом, лучшая из моделей обладает и минимальным временем такта для всех существующих девяти вариантов схемы.

Дальнейшие направления наших исследований связаны с адаптацией алгоритма *ESPRESSO* к условиям базиса однородных ПЛИС, а также с возможностью применения этих методов для реализации схем автомата в базисах, отличных от ПЛИС [13].

1. *Baranov S.* Logic Synthesis for Control Automata. – Dordrecht: Kluwer Academic Publishers, 1994. – 312 p.
2. *Соловьев В.В.* Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.
3. *Грушницкий Р.И., Мурсаев А.Х., Угрюмов Е.П.* Проектирование систем с использованием микросхем программируемой логики. – СПб: БХВ. – Петербург, 2002. – 608 с.
4. *Баркалов А.А., Цололо С.А.* Оптимизация автомата Мура, реализуемого в базе CPLD // Управляющие системы и машины. – 2008. – № 4. – С. 43–48.
5. *Cypress Semiconductor* – <http://www.cypress.com>.
6. *Баркалов А.А.* Принципы оптимизации логической схемы микропрограммного автомата Мура // Кибернетика и системный анализ. – 1998. – № 1. – С. 65–72.
7. *Баркалов А.А., Цололо С.А.* Оптимизация схемы автомата Мура в составе системы на кристалле // Радиоэлектроника и информатика. – 2007. – № 1. – С. 35–39.
8. *Баркалов А.А., Матвиенко А.В., Цололо С.А.* Оптимизация логической схемы автомата Мура на CPLD // Комп'ютерні системи, засоби та мережі. – 2007. – № 6. – С. 46–51.
9. *Barkalov A., Wegrzyn M.* Design of Control Units with Programmable Logic – Zielona Gora: University of Zielona Gora Press, 2006. – 150 p.
10. *DeMicheli G.* Synthesis and Optimization of Digital Circuits. – New York: McGraw-Hill, 1994. – 636 p.
11. *Xilinx* – www.xilinx.com.
12. *Altera* – www.altera.com.
13. *Maxfield C.* The Design Warrior's Guide to FPGAs. – Amsterdam: Elsevier, 2004. – 541 p.

Получено 07.09.2009