

# Оптимизация аппаратурных затрат в схеме МПА Мура

А.А. Баркалов, С.А. Цололо  
Зеленогурский университет (Польша),  
Донецкий национальный технический университет  
A.Barkalov@iie.uz.zgora.pl

## **Abstract**

*Barkalov A.A., Tsololo S.A. Hardware costs's optimization of Moore FSM circuit. Method of Moore's circuit optimization is proposed. Method based on features of CPLD architecture and Moore's FSM model. The carried out researches have shown that the method reduces hardware expenses up to 10%.*

## **Введение**

Модель микропрограммного автомата (МПА) Мура [1] в настоящее время часто используется для реализации устройств управления (УУ) [2,3], которые координируют работу блоков цифровой системы. Для реализации как УУ, так и в целом цифровых систем все чаще используются «системы на кристалле» (SoC, system-on-chip) [4]. В этих системах используются функциональные элементы двух основных типов: макроячейки (PAL, programmable array logic) для реализации произвольной логики и блоки памяти (EMB, embedded memory blocks) для реализации табличных функций [5]. Использование этого базиса позволяет реализовать сложную цифровую систему на одном кристалле [6]. Актуальность задачи уменьшения аппаратурных затрат в схеме МПА обусловлена сложностью реализуемых систем [2]. При решении этой задачи необходимо учитывать особенности структурной схемы МПА и особенности элементного базиса SoC. Среди особенностей МПА Мура, которые можно использовать в этом случае, следует отметить наличие псевдоэквивалентных состояний [7], а также регулярный характер системы микроопераций, что позволяет реализовать ее на EMB [5]. Особенности PAL является большой коэффициент объединения по входу, который достигает нескольких десятков в реальных CPLD (complex programmable logic devices) [8], и ограниченное число элементарных конъюнкций (термов) в одной макроячейке (порядка восьми) [2].

Целью представленных в этой работе исследований является возможность оптимизации комбинационной схемы автомата Мура за счет использования нескольких источников кода текущего состояния автомата, что возможно благодаря особенностям PAL. Основной решаемой в работе задачей является разработка метода синтеза МПА Мура, позволяющего

оптимизировать число макроячеек PAL в схеме формирования функций возбуждения (СФФВ) триггеров памяти автомата.

### **Основные положения и особенности реализации МПА Мура**

Пусть алгоритм управления цифровой системы представлен ГСА  $\Gamma = \Gamma(V, E)$ , где  $V = \{b_0, b_E\} \cup E_1 \cup E_2$  – множество вершин,  $E = \{ \langle b_q, b_t \rangle \mid b_q, b_t \in V \}$  – множество дуг. Здесь  $b_0$  – начальная вершина ГСА,  $b_E$  – конечная вершина ГСА,  $E_1$  – множество операторных вершин,  $E_2$  – множество условных вершин. В вершинах  $b_q \in E_1$  записываются наборы микроопераций  $Y(b_q) \subseteq Y$ , где  $Y = \{y_1, \dots, y_N\}$  – множество микроопераций операционного автомата цифровой системы [9]. В вершинах  $b_q \in E_2$  записываются элементы множества логических условий  $X = \{x_1, \dots, x_L\}$ . Начальная и конечная вершины ГСА соответствуют состоянию  $a_1 \in A = \{a_1, \dots, a_M\}$ , где  $A$  – множество состояний автомата Мура, а каждая вершина  $b_q \in E_1$  соответствует одному из элементов множества  $A$  [1]. Логическая схема МПА Мура задается системой уравнений

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

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

где  $T = \{T_1, \dots, T_R\}$  – множество внутренних переменных, кодирующих состояния  $a_m \in A$ ,  $R = \lceil \log_2 M \rceil$ ;  $\Phi = \{D_1, \dots, D_R\}$  – множество функций возбуждения триггеров памяти состояний. Системы (1)-(2) формируются на основе прямой структурной таблицы (ПСТ) со столбцами:  $a_m$  – текущее состояние;  $K(a_m)$  – код состояния  $a_m \in A$ ;  $a_s$  – состояние перехода;  $K(a_s)$  – код состояния  $a_s \in A$ ;  $X_h$  – конъюнкция некоторых элементов множества  $X$  (или их отрицаний), определяющая переход  $\langle a_m, a_s \rangle$ ;  $\Phi_h$  – набор функций возбуждения памяти МПА, принимающих единичное значение для переключения памяти из  $K(a_m)$  в  $K(a_s)$ ;  $h = 1, \dots, H_1(\Gamma)$  – номер строки таблицы. В столбце  $a_m$  записывается набор микроопераций  $Y(a_m) \subseteq Y$ , формируемых в состоянии  $a_m \in A$ . Естественно, что  $Y(a_m) = Y(b_q)$ , где вершина  $b_q \in E_1$  отмечена состоянием  $a_m \in A$ .

Как правило, число переходов  $H_1(\Gamma)$  больше числа переходов  $H_2(\Gamma)$  эквивалентного автомата Мили [1]. Это приводит к увеличению числа PAL в схеме МПА Мура по сравнению с этим показателем эквивалентного автомата Мили. Параметр  $H_1(\Gamma)$  можно уменьшить, благодаря наличию псевдоэквивалентных состояний (ПЭС) МПА Мура [10]. Состояния  $a_m, a_s \in A$  называется ПЭС, если выходы

соответствующих им вершин соединены с входом одной и той же вершины ГСА  $\Gamma$ . Пусть  $\Pi_A = \{B_1, \dots, B_I\}$  – разбиение множества  $A$  на классы ПЭС ( $I \leq M$ ). Поставим в соответствие классу  $B_i \in \Pi_A$  двоичный код  $K(B_i)$  разрядности  $R_i = \lceil \log_2 I \rceil$  и используем переменные  $\tau_i \in \tau$  для такого кодирования, где  $|\tau| = R_1$ . В этом случае МПА Мура представляется в виде структуры  $U_1$  (рис. 1).

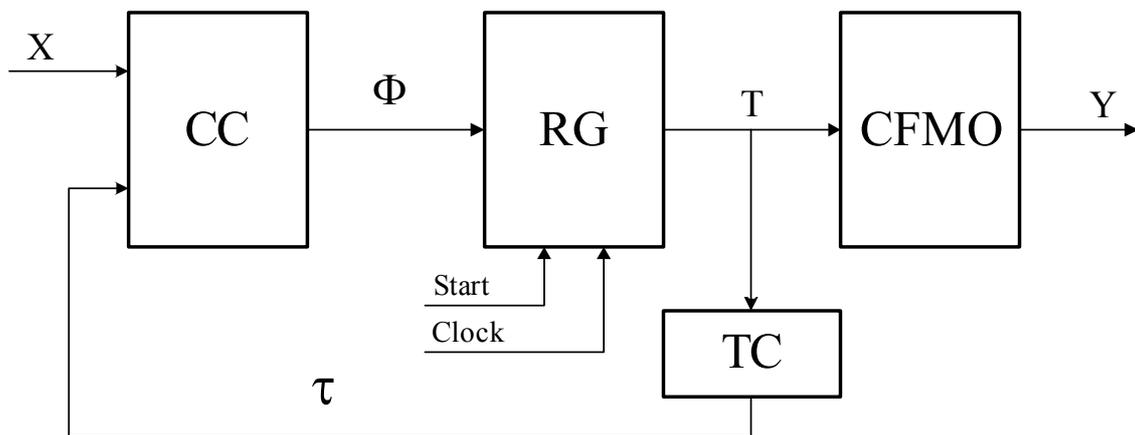


Рисунок 1 – Структурная схема МПА Мура  $U_1$

В МПА  $U_1$  схема  $CC$  формирует функции

$$\Phi = \Phi(\tau, X), \quad (3)$$

а схема формирования микроопераций  $CFMO$  реализует систему (2). Регистр  $RG$  представляет собой память состояний, по сигналу  $Start$  в  $RG$  заносится нулевой код начального состояния  $a_1 \in A$ , по сигналу  $Clock$  происходит смена кодов в регистре. Преобразователь кодов состояний  $TC$  реализует систему функций

$$\tau = \tau(\Gamma), \quad (4)$$

при этом код  $K(B_i)$  формируется на основе кода  $K(a_m)$ , где  $a_m \in B_i$ .

В [10] показано, что для МПА  $U_1$  число переходов уменьшается до  $H_2(\Gamma)$ . Недостатком МПА  $U_1$  является наличие схемы  $TC$ , которая требует дополнительных ресурсов. Отметим, что схема  $CC$  реализуется на  $PAL$ , а схемы  $TC$  и  $CFMO$  – на блоках памяти  $EMV$ . В настоящей работе предлагается метод синтеза МПА Мура, позволяющий уменьшить аппаратные затраты в схеме  $TC$  (при определенных условиях этот блок может не использоваться вообще). Предлагаемый метод основан на следующих особенностях  $SoC$ , основанных на технологии  $CPLD$  [2, 8]:

- коэффициент объединения по входу макроячейки  $PAL$  значительно превосходит максимально возможное число букв в термах системы (1), определяемое как  $L + R$ ;
- число выходов  $EMV$  может меняться в некотором диапазоне (как правило, 1, 2, 4, 8).

## Основная идея предлагаемого метода

Применим идею оптимального кодирования состояний МПА Мура [10], смысл которой заключается в таком кодировании ПЭС, чтобы максимально возможное число классов  $V_i \in \Pi_A$  соответствовали одному обобщенному интервалу  $R$ -мерного булева пространства. Представим множество  $\Pi_A$  в виде  $\Pi_A = \Pi_B \cup \Pi_C$ , где  $V_i \in \Pi_B$ , если

$$|V_i| > 1, \quad (5)$$

и  $V_i \in \Pi_C$  в противном случае. Очевидно, что схема ТС должна формировать только коды классов  $V_i \in \Pi_B$ .

Пусть  $t_F$  – фиксированное число выходов блока ЕМВ и пусть  $q$  – число слов в блоке при  $t_F = 1$ . При реализации схемы CFMO автомата  $U_1$  параметр  $t_F$  определяется следующим образом:

$$t_F = \lfloor q / M \rfloor. \quad (6)$$

При этом интегрально блоки ЕМВ схемы CFMO имеют

$$t_S = \lfloor N / t_F \rfloor \cdot t_F. \quad (7)$$

выходов. Очевидно, что  $\Delta_t$  выходов могут не использоваться для представления микроопераций, где

$$\Delta_t = t_S - N. \quad (8)$$

Эти выходы можно использовать для представления переменных  $\tau_r \in \tau$ . При выполнении условия

$$\Delta_t \geq R_1. \quad (9)$$

преобразователь кодов ТС можно реализовать в составе схемы СМОС.

В этом случае для интерпретации ГСА  $\Gamma$  предлагается использовать МПА Мура  $U_2$  (рис.2).

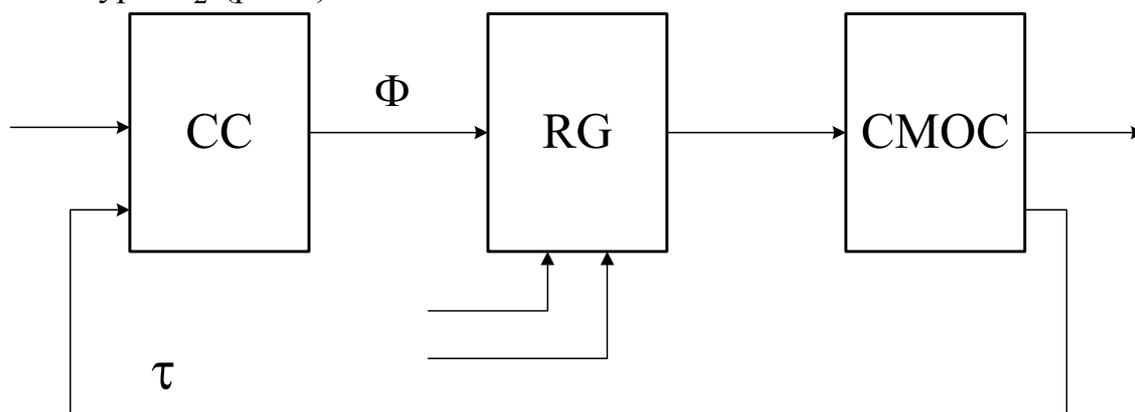


Рисунок 2 – Структурная схема МПА Мура  $U_2$

Автомат  $U_2$  имеет ряд отличий от автомата  $U_1$  :  
– схема СС формирует систему функций

$$\Phi = \Phi(\tau, X); \quad (10)$$

– вместо схемы CFMO используется схема СМОС, реализующая систему функций (2) и систему функций

$$\tau = \tau(T) \quad (11)$$

для представления кодов классов  $V_i \in \Pi_B$ ;

– преобразователь кодов ТС реализуется в составе СМОС.

Предлагаемый метод синтеза схемы МПА  $U_2$  включает следующие этапы:

1. Формирование отмеченной ГСА Г.
2. Формирование разбиения  $\Pi_A = \Pi_B \cup \Pi_C$ .
3. Кодирование классов  $V_i \in \Pi_B$ .
4. Формирование таблицы схемы СМОС.
5. Формирование модифицированной прямой структурной таблицы автомата  $U_2$ .
6. Реализация схемы автомата в заданном элементном базисе.

### **Исследование эффективности предложенного метода**

В данной работе в основе исследований лежит принцип перехода от отдельных ГСА к классам ГСА, который основан на вероятностном подходе. Данный принцип был предложен Г.И. Новиковым [11] и развит в работе [7]. В качестве принципа оценки аппаратных затрат был выбран принцип перехода от схемы на макроячейках PAL и блоках EMB к матричным схемам предложенным С.И. Барановым [1]. Кроме того, для определения общих тенденций, а не отдельных точек в пространстве решений, был выполнен переход от абсолютных оценок аппаратных затрат к относительным оценкам для разных структур схемы автомата.

Матричная реализация МПА  $U_1$  представлена на рис. 3. Символ «&» означает конъюнктивную матрицу, а символ «v» – дизъюнктивную матрицу.

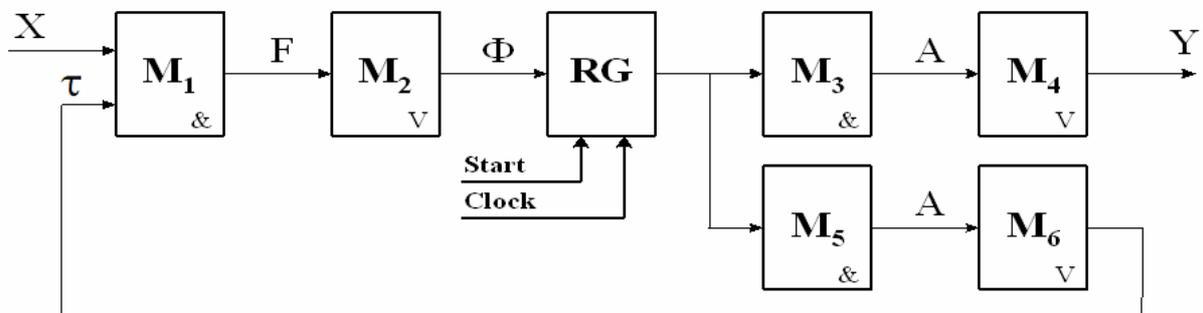


Рисунок 3 – Матричная реализация МПА Мура  $U_1$

Здесь матрицы  $M_1$  и  $M_2$  образуют схему СС, в которой число термов и переменных обратной связи совпадают с характеристиками эквивалентного автомата Мили. Матрицы  $M_3$  и  $M_4$  образуют схему

CFMO, выходы схемы  $M_3$  соответствуют конъюнкциям  $A_m$  ( $m = 1, \dots, M$ ), соответствующих кодам состояний автомата Мура. Матрицы  $M_5$  и  $M_6$  образуют схему ТС. Площади  $S(M_i)$  матриц  $M_i$  ( $i = 1, \dots, 6$ ) могут быть определены следующим образом:

$$\begin{aligned} S(M_1) &= 2(L + R_1)H_0; & S(M_2) &= H_0R; \\ S(M_3) &= S(M_5) = 2R \cdot 2^R; & S(M_4) &= 2^R \cdot N; \\ S(M_6) &= 2^R \cdot R_1. \end{aligned} \quad (12)$$

В нашем случае эти величины определяются в условных единицах площади.

Оценим классы ГСА параметром  $P_1$ , равным доле операторных вершин ГСА, как это предложено в работе [11]. Используя этот параметр и результаты работы [7] можно найти следующие выражения для аргументов систем (12):

$$\begin{aligned} L &\approx 0.75 \cdot (1 - P_1)K; \\ R_1 &= \lceil \log_2(3.55 + 0.3 \cdot P_1 \cdot K) \rceil; \\ H_0 &= 4.44 + P_1 \cdot K; \\ R &= \lceil \log_2 P_1 \cdot K \rceil, \end{aligned} \quad (13)$$

где  $K$  – число вершин ГСА. Введем коэффициент  $k_p$ , определяющий отношение одной условной единицы площади макроячейки PAL к соответствующему параметру ЕМВ. Теперь площадь матричной реализации схемы МПА  $U_1$  может быть выражена формулой

$$S(U_1) = k_p H_0 (2L + 2R_1 + R) + 2^R (4R + N + R_1) \quad (14)$$

С учетом формул (13) выражение  $S(U_1)$  может быть представлено как функции от аргументов  $K, P_1, k_p, N$ .

По аналогичному принципу может быть построена матричная схема МПА  $U_2$  (рис. 4).

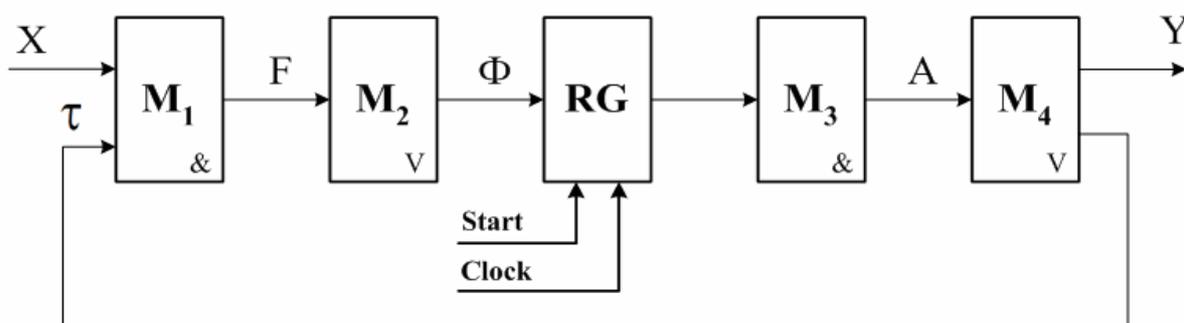


Рисунок 4 – Матричная реализация МПА Мура  $U_2$

Так как переменные  $\tau_i \in T$  формируются на свободных выходах схемы CMOS, то площади схем CMOS и CFMO также будем считать

совпадающими. Таким образом, площадь матричной реализации схемы МПА  $U_2$  может быть выражена формулой

$$S(U_2) = k_p H_0 (2L + 2R_1 + R) + 2^R (4R + N). \quad (15)$$

С учетом формул (13) выражение  $S(U_2)$  может быть представлено как функция от аргументов  $k_1$ ,  $P_1$ ,  $k_p$ ,  $N$ .

Для определения эффективности предложенного метода, необходимо найти область, где выполняется условие

$$f_1 = \frac{S(U_2)}{S(U_1)} < 1. \quad (16)$$

На рис. 5 и рис. 6 представлены некоторые результаты исследований, при этом параметр  $K$  менялся в диапазоне от 100 до 1000, параметр  $k_p = 0.2$ , число микроопераций  $N \in \{10, 50, 100\}$ , вероятность  $P_1 \in \{0.3; 0.7\}$ .

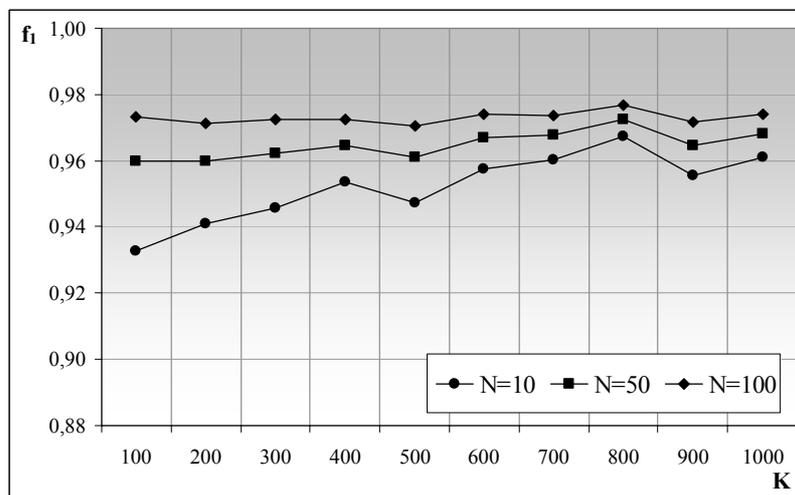


Рисунок 5 – Функция  $f_1$  при  $P_1 = 0.3, k_p = 0.2$

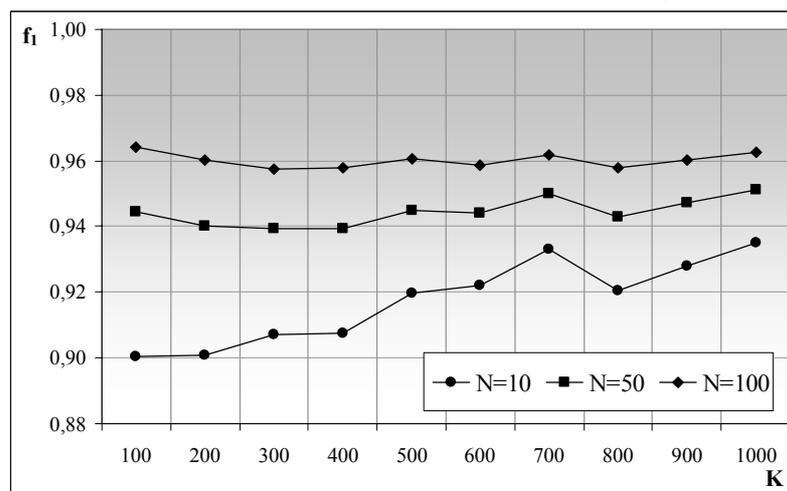


Рисунок 6 – Функция  $f_1$  при  $P_1 = 0.7, k_p = 0.2$

На основе анализа графиков можно сделать вывод, что предлагаемый метод позволяет получить устройства с лучшими

характеристиками, чем у автомата  $U_1$ . Выигрыш в аппаратных затратах увеличивается как при уменьшении числа микроопераций и числа вершин, так и при росте доли операторных вершин ГСА (рост  $P_1$ ). Максимальный выигрыш достигается для ГСА, имеющих около 100 вершин, и составлял 10% при  $N = 10$  и  $P_1 = 0.7$ .

Выигрыш также увеличивается по мере уменьшения параметра  $k_p$ , однако эти графики не показаны.

### **Заключение**

Предложенный метод позволяет уменьшить аппаратные затраты в схеме автомата Мура, реализуемого в базисе CPLD, по сравнению с МПА  $U_1$ , включающим преобразователь ТС кодов псевдоэквивалентных состояний в коды классов псевдоэквивалентных состояний. При выполнении условия (9) появляется возможность реализовать ТС в составе схемы СМОС, что уменьшает число блоков ЕМВ по сравнению с МПА  $U_1$ . Уменьшение пропорционально коэффициенту

$$\eta = N_1(\Gamma) / N_2(\Gamma). \quad (17)$$

Проведенные исследования показали, что при выполнении условия (9) автомат  $U_2(\Gamma)$  всегда имеет меньшие аппаратные затраты, чем эквивалентный МПА  $U_1(\Gamma)$ . Максимальный выигрыш может достигать 10%. Кроме того, необходимо отметить, что автоматы  $U_1(\Gamma)$  и  $U_2(\Gamma)$  имеют одинаковое быстродействие, то есть выигрыш по аппаратуре не приводит к потере производительности.

Научная новизна предложенного метода заключается в использовании особенностей автомата Мура (наличие классов псевдоэквивалентных состояний) и элементного базиса (большой коэффициент объединения по входу) для оптимизации числа макроячеек PAL в логической схеме автомата. Практическая значимость метода заключается в уменьшении площади кристалла SoC, занимаемой комбинационной схемой МПА, что позволяет получить схемы, которые обладают меньшей стоимостью, чем известные из литературы

### **Литература**

1. Baranov S. Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 pp.
2. Соловьев В.В. Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия-ТЕЛЕКОМ, 2001. – 636 с.
3. Баркалов А.А., Палагин А.В. Синтез микропрограммных устройств управления. – Киев: Институт кибернетики НАН Украины, 1997. – 136с.

4. Barkalov A., Wegrzyn W. Design of Control Units with Programmable Logic. – Zielona Gora: University of Zielona Gora Press, 2006. – 150 pp.
5. Maxfield C. The Design Warriors Guide to FPGAs. – Elsevier, 2004. – 541p.
6. Грушницкий Р.И., Мурсаев А.Х., Угрюмов Е.П. Проектирование систем с использованием микросхем программируемой логики. – СПб: БХВ. – Петербург, 2002. – 608с.
7. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. – Донецк: ДНТУ, 2002. – 262 с.
8. Kania D. Synteza logiczna przeznaczona dla matrycowych struktur programowalnych typuPAL. – Zeszyty naukowe Politechniki Śląskiej, Gliwice, 2004. – 240 pp.
9. DeMicheli G. Synthesis and Optimization of Digital Circuits. – McGraw-Hill, 1994. – 636p.
10. Баркалов А. А. Принципы оптимизации логической схемы микропрограммного автомата Мура. // Кибернетика и системный анализ. – 1998, №1.
11. Новиков Г.И. оценка эффективности параллельной выборки микрокоманд // Известия вузов СССР. Приборостроение. – 1974, №2. – с.71-73

Дата надходження до редакції 01.11.2007 р.