

А.А. Баркалов, Л.А. Титаренко, А.С. Лаврик

## Уменьшение аппаратных затрат в устройстве управления с преобразователем адреса микрокоманды

Предложен метод уменьшения аппаратных затрат в логической схеме устройства управления, базирующийся на использовании большого коэффициента объединения по входу макроячеек ПМЛ, что позволяет использовать больше одного источника для адреса микрокоманды. Для оптимизации схемы блока адресации микрокоманд использован метод замены логических условий. Приведен пример применения предложенного метода.

A method of the hardware reduction in the logical circuit of the control unit is suggested. The method is based on control units and CPLD chips. The method is based on using a wide fan-in of PAL macrocells allowing the use of more than one source of the microinstruction address. The method of logical conditions replacement is used for the optimization of a microinstruction addressing block. An example of the suggested method application is given.

Запропоновано метод зменшення апаратних витрат у логічній схемі упорядження управління, який базується на використанні великого коефіцієнта об'єднання по входу макрокомірок ПМЛ, що дозволяє застосувати більше як одне джерело для адреси мікрокоманди. Для оптимізації схеми блока адресації мікрокоманд застосовано метод заміни логічних умов. Наведено приклад застосування цього методу.

**Введение.** В настоящее время для реализации схем устройств управления (УУ) [1] широко используются программируемые логические интегральные схемы (ПЛИС), включающие в свой состав макроячейки программируемой матричной логики (ПМЛ) [2, 3]. При синтезе УУ необходимо учитывать особенности как интерпретируемого алгоритма управления, так и используемых элементов. Так, при интерпретации линейных граф-схем алгоритмов (ГСА) целесообразно использовать модель композиционного микропрограммного устройства управления (КМУУ) [4]. Особенность ПЛИС – большой коэффициент объединения по входу [5, 6], используемый для увеличения числа источников кодов классов псевдоэквивалентных операторных линейных цепей (ОЛЦ) [7, 8]. В статье предлагается метод синтеза КМУУ, учитывающий особенности ПЛИС, а также использующий замену логических условий [1].

*Цель исследования* – уменьшение аппаратных затрат в схеме КМУУ за счет одновременного использования нескольких источников кодов классов псевдоэквивалентных ОЛЦ

**Ключевые слова:** устройство управления, ПЛИС, ПМЛ, операторная линейная цепь, граф-схема алгоритма.

и замены логических условий. *Задачей исследования* является разработка метода, позволяющего уменьшить аппаратные затраты в схемах адресации микрокоманд и преобразователя адреса микрокоманды.

### Особенности КМУУ с преобразователем адреса микрокоманды

Пусть ГСА  $\Gamma$  представлена множествами вершин  $B$  и дуг  $E$ , соединяющих эти вершины. При этом  $B = \{b_0, b_E\} \cup E_1 \cup E_2$ , где  $b_0$  – начальная вершина ГСА,  $b_E$  – его конечная вершина,  $E_1$  – множество операторных вершин, где  $|E_1| = M$ ,  $E_2$  – множество условных вершин. В вершинах  $b_q \in E_1$  записаны наборы микроопераций  $Y(b_q) \subseteq Y$ , где  $Y = \{y_1, \dots, y_N\}$  – множество микроопераций. В вершинах  $b_q \in E_2$  записаны элементы множества логических условий  $X = \{x_1, \dots, x_L\}$ . Если ГСА  $\Gamma$  является линейной, то она включает в себя более 75% операторных вершин [2].

Сформируем множество ОЛЦ  $C = \{\alpha_1, \dots, \alpha_G\}$  ГСА  $\Gamma$ , где каждая из ОЛЦ – последовательность операторных вершин и каждой паре ее

соседних компонент  $b_i, b_j$  соответствует дуга  $\langle b_i, b_j \rangle \in E$ . Каждая ОЛЦ имеет только один выход  $O_g$  и произвольное число входов  $g = (1, \dots, G)$ . Формальные определения ОЛЦ, их входов и выходов можно найти в [4]. Каждая вершина  $b_q \in E_1$  соответствует микрокоманде  $MI_q$ , хранимой в управляющей памяти (УП) КМУУ по адресу  $A(b_q)$ . Для адресации микрокоманд достаточно

$$R = \lceil \log_2 M \rceil \quad (1)$$

бит, представленных переменными  $T_r \in T$ , где  $|T| = R$ . Пусть ОЛЦ  $\alpha_g \in C$  включает  $F_g$  компонент, а адресация микрокоманд выполнена так, что

$$A(b_{g_{i+1}}) = A(b_{g_i}) + 1, \quad (2)$$

где  $b_{g_i}$  –  $i$ -я компонента ОЛЦ  $\alpha_g \in C$ ,  $i = 1, \dots, F_g - 1$ .

Если выходы  $O_i, O_j$  соединены с входом одной и той же вершины ГСА  $\Gamma$ , то ОЛЦ  $\alpha_i, \alpha_j \in C$  являются псевдоэквивалентными ОЛЦ (ПОЛЦ) [2]. Найдем разбиение  $\Pi_C = \{B_1, \dots, B_l\}$  множества  $C_1 \subseteq C$  на классы ПОЛЦ. При этом ОЛЦ  $\alpha_g \in C_1$ , если ее вход не связан с вершиной  $b_E$ , т.е.  $\langle O_g, b_E \rangle \notin E$ . Закодируем классы  $B_i \in \Pi_C$  двоичными кодами  $K(B_i)$  разрядности

$$R_1 = \lceil \log_2 I \rceil \quad (3)$$

и используем для кодирования элементы множества  $\tau$ , где  $|\tau| = R_1$ . В этом случае для интерпретации ГСА  $\Gamma$  может быть использовано КМУУ с преобразователем адреса (рис. 1), обозначаемое в дальнейшем символом  $U_1$  [4].

По сигналу *Start* в счетчик (СТ) записывается начальный адрес микропрограммы, а триггер выборки (ТВ) устанавливается в единичное состояние. При этом *Fetch* = 1, что разрешает выборку микрокоманд из УП. Если считанная микрокоманда  $MI_q$  не соответствует выходу ОЛЦ  $\alpha_g \in C_1$ , то одновременно с микроопера-

циями  $Y(b_q)$  формируется переменная  $y_0$ . Если  $y_0 = 1$ , то содержимое СТ увеличивается на 1, что соответствует режиму безусловного перехода (2) в пределах ОЛЦ. В противном случае  $y_0 = 0$  и блок адресации микрокоманд (БАМ) формирует функции

$$\Phi = \Phi(\tau, X) \quad (4)$$

для записи в СТ адреса входа очередной ОЛЦ.

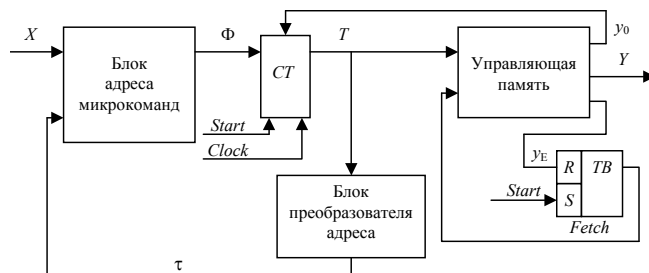


Рис. 1. Структурная схема КМУУ  $U_1$

При этом блок преобразователя адреса (БПА) формирует функции

$$\tau = \tau(T), \quad (5)$$

равные единице в коде  $K(B_i)$ , где  $\alpha_g \in B_i$ . Если достигнут выход ОЛЦ  $\alpha_g \notin C_1$ , то формируется  $y_E = 1$ , триггер ТВ обнуляется и выборка микрокоманд прекращается.

Такая организация КМУУ позволяет уменьшить число термов в системе функций  $\Phi$  от  $H_1$  до  $H_0$ , где  $H_1$  – число строк таблицы переходов автомата Мура, а  $H_0$  – число строк таблицы переходов эквивалентного автомата Мили. Однако, схема БПА потребляет некоторые ресурсы ПЛИС или программируемого пассивного запоминающего устройства (ППЗУ), из которых строится УП. В статье предлагается метод синтеза КМУУ  $U_2$ , в котором  $H_2 = H_0$ , а схема БПА требует меньше аппаратных затрат, чем в КМУУ  $U_1$ . При определенных условиях этот блок может вообще исчезнуть.

### Основная идея предлагаемого метода

Отметим, что схемы блоков БАМ, БПА, СТ и ТВ реализуются в составе ПЛИС, а для реализации УП требуются ППЗУ, имеющие  $t$  выходов ( $t = 1, 2, 4, 8, 16$ ). Выполним адресацию

ОЛЦ  $\alpha_g \in C_1$  таким образом, чтобы выполнялось (2) и максимально возможное число классов  $B_i \in \Pi_C$  выражалось одним обобщенным интервалом  $R$ -мерного булева пространства. Для такой адресации необходимо разработать алгоритм, который нам пока не известен.

Пусть  $\Pi_C = \Pi_A \cup \Pi_B$ , где  $B_i \in \Pi_A$ , если этому классу соответствует один интервал, и  $B_i \in \Pi_B$  в противном случае. Источником кодов для классов  $B_i \in \Pi_A$  является счетчик СТ. Если выполняется условие

$$\Pi_B = \emptyset, \quad (6)$$

то блок БПА отсутствует. В противном случае преобразованию подлежат только адреса выходов ОЛЦ, входящих в классы  $B_i \in \Pi_B$ . Для кодирования этих классов достаточно

$$R_2 = \lceil \log_2(I_B + 1) \rceil \quad (7)$$

переменных, где  $I_B = |\Pi_B|$ , единица прибавляется для кодирования ситуации  $B_i \in \Pi_A$ . Отметим, что часть кодов может быть реализована на свободных выходах ППЗУ. Пусть для кодирования микроопераций используется стратегия унитарного кодирования [2], тогда слово УП имеет  $N + 2$  разрядов. Для реализации УП требуется

$$R_0 = \left\lceil \frac{N + 2}{t} \right\rceil \quad (8)$$

микросхем с числом ячеек, не меньшим  $M$ . При этом

$$R_3 = R_0 * t - N - 2 \quad (9)$$

выходов ППЗУ – свободны. Если

$$R_3 \geq R_2, \quad (10)$$

то источниками кодов классов  $B_i \in \Pi_B$  являются УП, а блок БПА отсутствует. Такой подход позволяет уменьшить число макроячеек в схеме блока БАМ и число ППЗУ, используемых для преобразователя адреса.

Дальнейшая оптимизация схемы БАМ возможна за счет замены логических условий [1], когда множество  $X$  заменяется множеством

$P = \{p_1, \dots, p_Q\}$ , где  $Q \ll L$ . В этом случае для интерпретации ГСА  $\Gamma$  предлагается КМУУ  $U_2$  (рис. 2).

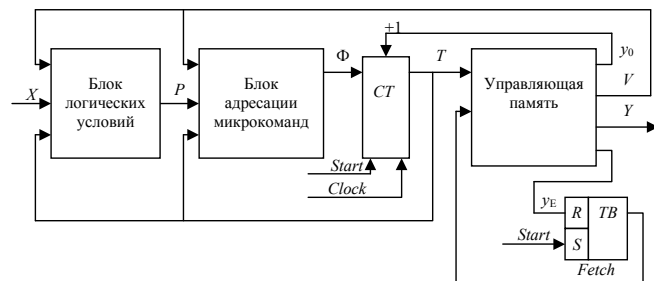


Рис. 2. Структурная схема КМУУ  $U_2$

В КМУУ  $U_2$  коды  $K_A(B_i)$  классов  $B_i \in \Pi_A$  представляются переменными  $T_r \in T$ , коды  $K_B(B_i)$  классов  $B_i \in \Pi_B$  – переменными  $v_r \in V$ , где  $|V| = R_2$ . В отличие от КМУУ  $U_1$  здесь блок БПА отсутствует, а блок БАМ реализует функции

$$\Phi = \Phi(T, V, P). \quad (11)$$

Переменные  $p_q \in P$  формируются блоком логических условий (БЛУ) в виде системы

$$P = P(T, V, X). \quad (12)$$

Пусть символ  $U_i(\Gamma_j)$  означает, что КМУУ  $U_i$  интерпретирует ГСА  $\Gamma_j$ , а  $Q_i(\Gamma_j)$  – число макроячеек в схеме БАМ КМУУ  $U_i(\Gamma_j)$ , где  $i = 1, 2$ . Пусть каждая макроячейка ПМЛ имеет  $S$  входов, причем на входы  $q$ -й макроячейки КМУУ  $U_i(\Gamma_j)$  поступает  $Q_q$  переменных  $p_q$ . Применение предложенного метода целесообразно при выполнении условия

$$Q_q + R + R_2 \leq S, \quad (13)$$

где  $q = 1, \dots, Q_1(\Gamma_j)$ . В противном случае число  $Q_2(\Gamma_j)$  значительно увеличивается по сравнению с  $Q_1(\Gamma_j)$ .

В статье предлагается метод синтеза КМУУ  $U_2$ , включающий следующие этапы:

- формирование множеств  $C$ ,  $C_1$ ,  $\Pi_C$  для ГСА  $\Gamma$ ;
- адресация микрокоманд;
- формирование множеств  $\Pi_A$ ,  $\Pi_B$ ;

- кодирование классов  $B_i \in \Pi_B$ ;
- формирование содержимого управляющей памяти;
- замена логических условий;
- формирование таблицы переходов КМУУ;
- формирование таблицы блока БЛУ;
- синтез логической схемы КМУУ.

### Пример применения предложенного метода

Пусть для ГСА  $\Gamma_1$  построены множества  $C = \{\alpha_1, \dots, \alpha_9\}$ ,  $C_1 = \{\alpha_1, \dots, \alpha_8\}$  и  $\Pi_C = \{B_1, \dots, B_5\}$ , где  $\alpha_1 = \langle b_1, b_2 \rangle$ ,  $\alpha_2 = \langle b_3, \dots, b_6 \rangle$ ,  $\alpha_3 = \langle b_7, b_8 \rangle$ ,  $\alpha_4 = \langle b_5, \dots, b_{13} \rangle$ ,  $\alpha_5 = \langle b_{14}, \dots, b_{17} \rangle$ ,  $\alpha_6 = \langle b_{18}, \dots, b_{21} \rangle$ ,  $\alpha_7 = \langle b_{22}, \dots, b_{25} \rangle$ ,  $\alpha_8 = \langle b_{26}, \dots, b_{28} \rangle$ ,  $\alpha_9 = \langle b_{29}, \dots, b_{31} \rangle$ ,  $B_1 = \{\alpha_1\}$ ,  $B_2 = \{\alpha_2, \alpha_3\}$ ,  $B_3 = \{\alpha_4, \alpha_5\}$ ,  $B_4 = \{\alpha_6, \alpha_7\}$ ,  $B_5 = \{\alpha_8\}$ . Итак  $I = 5$ ,  $R_1 = 3$ ,  $\tau = \{\tau_1, \tau_2, \tau_3\}$ ,  $M = 31$ ,  $R = 5$ .

Выполним адресацию микрокоманд, модифицировав алгоритм [4]. При этом  $A(b_1) = 00000$ , ...,  $A(b_{25}) = 110000$ ,  $A(b_{26}) = 11100$ , ...,  $A(b_{28}) = 11110$ ,  $A(b_{29}) = 11001$ , ...,  $A(b_{31}) = 11011$ .

Построим карту Карно, отмеченную переменными  $T_r \in T = \{T_1, \dots, T_5\}$ , в которой показаны выходы  $O_g$  ОЛЦ  $\alpha_g \in C$  и интервалы кодового пространства, соответствующие классам  $B_i \in \Pi_C$  (рис. 3).

|           |               |       |       |       |       |       |       |       |
|-----------|---------------|-------|-------|-------|-------|-------|-------|-------|
|           | $T_1 T_2 T_3$ | $B_1$ |       | $B_4$ | $B_5$ |       | $B_3$ |       |
| $T_4 T_5$ | 000           | 001   | 011   | 010   | 110   | 111   | 101   | 100   |
| 00        | *             | *     | $O_4$ | *     | $O_7$ | *     | $O_6$ | $O_5$ |
| 01        | $O_1$         | $O_2$ | *     | *     | *     | *     | *     | *     |
| 11        | *             | $O_3$ | *     | *     | *     | *     | *     | *     |
| 10        | *             | *     | *     | *     | $O_8$ | $O_8$ | *     | *     |
|           | $B_2$         | $B_3$ |       | $B_6$ |       |       |       |       |

Рис. 3. Карта Карно для выходов ОЛЦ КМУУ  $U_2(\Gamma_1)$

В этой карте символом \* обозначена ситуация, когда вершина  $b_q \in E_1$  с адресом  $A(b_q)$  не является выходом ОЛЦ  $\alpha_g \in C$ . Из рис. 3 можно найти кодовые интервалы, соответствующие выходам ОЛЦ для классов  $B_i \in \Pi_C$ : классу  $B_1$  соответствует интервал  $0000*$ , классу

$B_2$  – интервал  $001**$ , классу  $B_3$  – интервалы  $01***$  и  $10000$ , классу  $B_4$  – интервалы  $101**$  и  $11000$ , классу  $B_5$  – интервал  $111**$ . Отметим, что класс  $B_6 = \{\alpha_9\}$  не рассматривается, так как ОЛЦ  $\alpha_9 \notin C_1$ .

Полученные интервалы определяют множества  $\Pi_A = \{B_1, B_2, B_5\}$  и  $\Pi_B = \{B_3, B_4\}$ . Пусть для ГСА  $\Gamma_1$   $N = 12$ , а для реализации схемы УП используются ППЗУ с  $t = 4$ . В этом случае имеем  $R_3 = 2$  и условие (10) выполняется, так как  $R_2 = \lceil \log_2(2+1) \rceil = R_3$ . Итак, применение КМУУ  $U_2(\Gamma_j)$  имеет смысл, и блок БПА отсутствует.

Закодируем классы  $B_i \in \Pi_B$  тривиальным образом:  $K_B(B_3) = 01$ ,  $K_B(B_4) = 10$ . При этом код 00 соответствует ситуации  $B_i \in \Pi_A$ , а код 11 может использоваться для оптимизации кодов классов. Окончательно имеем  $K_B(B_3) = *1$ ,  $K_B(B_4) = 1*$ . Кроме того, из карты Карно на рис. 3 можно определить следующие коды:  $K_A(B_1) = 0000*$ ,  $K_A(B_2) = 001**$ ,  $K_A(B_5) = 111**$ .

Для формирования содержимого УП предлагается следующая процедура, являющаяся модификацией метода [4].

1.  $q = 1$ .
2. Если  $b_q \in E_1$ , то по адресу  $A(b_q)$  записать  $Y(b_q)$ . Иначе – перейти к п.6.
3. Если  $b_q$  не является выходом ОЛЦ  $\alpha_g \in C$ , то по адресу  $A(b_g)$  записывается  $y_o$ .
4. Если  $b_q$  – выход ОЛЦ  $\alpha_g \notin C_1$ , то по адресу  $A(b_q)$  записывается  $y_E$ .
5. Если  $b_q$  – выход ОЛЦ  $\alpha_g \in B_i$ , где  $B_i \in \Pi_B$ , то по адресу  $A(b_q)$  записывается код  $K_B(B_i)$ .
6. Если все вершины ГСА  $\Gamma$  проанализированы, то перейти к п.7, иначе  $q := q + 1$  и перейти к п.2.
7. Конец.

Применение этой процедуры не вызывает трудностей, и в нашем примере результат этого этапа не показан.

Пусть переходы для классов  $B_i \in \Pi_C$  задаются следующей системой обобщённых формул перехода (ОФП) [7]:

$$\begin{aligned} B_1 &\rightarrow x_1 b_3 \vee \overline{x_1 x_4 b_5} \vee \overline{x_1 x_4 b_7}; \\ B_2 &\rightarrow x_3 b_9 \vee \overline{x_3 b_{26}}; \\ B_3 &\rightarrow x_1 b_{18} \vee \overline{x_1 x_2 b_{20}} \vee \overline{x_1 x_2 b_{26}}; \\ B_4 &\rightarrow x_5 b_{27} \vee \overline{x_5 b_5}; \\ B_5 &\rightarrow x_6 b_{24} \vee \overline{x_6 x_7 b_{29}} \vee \overline{x_6 x_7 b_{19}}. \end{aligned} \quad (14)$$

Формулы вида (14) являются модификацией формул перехода для микропрограммных автоматов (МПА) [1]. При этом учитывается тот факт, что ОЛЦ в КМУУ являются аналогами состояний в МПА. Поскольку переходы из выходов всех ОЛЦ  $\alpha_g \in B_i$  абсолютно одинаковы, то вместо выходов в ОФП записываются классы  $B_i \in \Pi_C$ . Термин ОФП подчеркивает факт замены выходов ОЛЦ  $\alpha_g \in B_i$  соответствующим классом  $\Pi_C$ . Переходы из класса  $B_i \notin \Pi_C$  не рассматриваются, так как для подобных классов функционирование КМУУ прекращается после формирования сигнала  $y_E$ .

Пусть  $X(B_i)$  – множество логических условий, определяющих переходы из класса  $B_i \in \Pi_C$ , причём  $|X(B_i)| = Q_i$ . Пусть  $Q = \max(Q_i | B_i \in \Pi_C)$ , тогда для замены логических условий  $x_l \in X$  достаточно  $Q$  элементов множества  $P$ .

Для КМУУ  $U_2(\Gamma_1)$  из системы (14) имеем:  $X(B_1) = \{x_1, x_4\}$ ,  $X(B_2) = \{x_3\}$ ,  $X(B_3) = \{x_1, x_2\}$ ,  $X(B_4) = \{x_5\}$ ,  $X(B_5) = \{x_6, x_7\}$ ,  $Q_1 = Q_3 = Q_5 = 2$ ,  $Q_2 = Q_4 = 1$  и  $Q = 2$ . Таким образом,  $P = \{p_1, p_2\}$ . Построим таблицу замены логических условий со столбцами  $B_i \in \Pi_C$  и строками  $p_q \in P$ . При этом на пересечении строки  $p_q$  и столбца  $B_i$  записывается условие  $x_l \in X$ , заменяемое для класса  $B_i \in \Pi$  переменной  $p_q \in P$ . Замена производится так, чтобы одна и та же переменная  $x_l \in X$  не появлялась в разных строках таблицы (табл. 1).

Таблица 1. Замена логических условий КМУУ  $U_2(\Gamma_1)$

| $B_i$ | $B_1$ | $B_2$ | $B_3$ | $B_4$ | $B_5$ |
|-------|-------|-------|-------|-------|-------|
| $P_1$ | $x_1$ | $x_3$ | $x_1$ | –     | $x_6$ |
| $P_2$ | $x_4$ | –     | $x_2$ | $x_5$ | $x_7$ |

Построим преобразованную систему ОФП, заменив условия  $x_l \in X$  переменными  $p_q \in P$ :

$$\begin{aligned} B_1 &\rightarrow p_1 b_3 \vee \overline{p_1 p_2 b_5} \vee \overline{p_1 p_2 b_7}; \\ B_2 &\rightarrow p_1 b_9 \vee \overline{p_1 b_{26}}; \\ B_3 &\rightarrow p_1 b_{18} \vee \overline{p_1 p_2 b_{20}} \vee \overline{p_1 p_2 b_{26}}; \\ B_4 &\rightarrow p_2 b_{27} \vee \overline{p_2 b_5}; \\ B_5 &\rightarrow p_1 b_{24} \vee \overline{p_1 p_2 b_{29}} \vee \overline{p_1 p_2 b_{19}}. \end{aligned} \quad (15)$$

Система ОФП вида (15) – основа для формирования таблицы переходов КМУУ  $U_2$  (табл. 2), имеющей столбцы  $B_i, K(B_i), b_q, A(b_q), P_h, \Phi_h, h$ . Для КМУУ  $U_2(\Gamma_1)$  эта таблица строится по системе (15) и имеет  $H = 13$  строк. Число строк определяется числом термов в системе (15). Назначение столбцов ясно из самой системы (15). При этом код  $K(B_i)$  состоит из двух частей  $K_A(B_i)$  и  $K_B(B_i)$ , соответствующих классам  $\Pi_A$  и  $\Pi_B$ .

Эта таблица является основой для формирования системы (11), термы  $F_h$  которой представляются в виде:

$$F_h = \left( \begin{matrix} R \\ \Lambda \\ r=1 \end{matrix} T_r^{l_{rh}} \right) * \left( \begin{matrix} R_2 \\ \Lambda \\ r=1 \end{matrix} V_r^{E_{rh}} \right) * P_h \quad (16)$$

где  $l_{rh} \in \{0, 1, *\}$  – значение  $r$ -го разряда кода  $K_A(B_i)$  из  $h$ -й строки таблицы переходов;  $E_{rh} \in \{0, 1, *\}$  – значение  $r$ -го разряда кода  $K_B(B_i)$  из  $h$ -й строки таблицы переходов;  $T_r^0 = \overline{T_r}, T_r^1 = T_r, T_r^* = 1 (r = 1, \dots, R)$ ;  $V_r^0 = \overline{V_r}, V_r^1 = V_r, V_r^* = 1 (r = 1, \dots, R_2)$ ;  $h = 1, \dots, H$ . Например, из табл. 2 строится дизъюнктивная нормальная форма (ДНФ) следующей функции:  $D_1 = F_5 \vee \dots \vee F_9 \vee F_{11} \vee F_{12} \vee F_{13} = \overline{T_1} \overline{T_2} T_3 \overline{V_1} \overline{V_2} \overline{P_1} \vee \vee V_2 \vee V_1 P_2 \vee T_1 T_2 T_3 \overline{V_1} \overline{V_2}$  (с учетом минимизации).

**Таблица 2.** Таблица переходов КМУУ  $U_2(\Gamma_1)$

| $B_i$ | $K_A(B_i)$ | $K_B(B_i)$ | $b_q$    | $A(b_q)$ | $P_h$                | $\Phi_h$          | $h$ |
|-------|------------|------------|----------|----------|----------------------|-------------------|-----|
| $B_1$ | 0000*      | 00         | $b_3$    | 00010    | $p_1$                | $D_4$             | 1   |
|       |            |            | $b_5$    | 00100    | $\overline{p_1 p_2}$ | $D_3$             | 2   |
|       |            |            | $b_7$    | 00110    | $\overline{p_1 p_2}$ | $D_3 D_4$         | 3   |
| $B_2$ | 001**      | 00         | $b_9$    | 01000    | $p_1$                | $D_2$             | 4   |
|       |            |            | $b_{26}$ | 11100    | $\overline{p_1}$     | $D_1 D_2 D_3$     | 5   |
| $B_3$ | *****      | *0         | $b_{18}$ | 10001    | $p_1$                | $D_1 D_5$         | 6   |
|       |            |            | $b_{20}$ | 10011    | $\overline{p_1 p_2}$ | $D_1 D_4 D_5$     | 7   |
|       |            |            | $b_{26}$ | 11100    | $\overline{p_1 p_2}$ | $D_1 D_2 D_3$     | 8   |
| $B_4$ | *****      | 1*         | $b_{27}$ | 11101    | $p_2$                | $D_1 D_2 D_3 D_5$ | 9   |
|       |            |            | $b_5$    | 00100    | $\overline{p_2}$     | $D_3$             | 10  |
| $B_5$ | 111**      | 00         | $b_{24}$ | 10111    | $p_1$                | $D_1 D_3 D_4 D_5$ | 11  |
|       |            |            | $b_{29}$ | 11001    | $\overline{p_1 p_2}$ | $D_1 D_2 D_5$     | 12  |
|       |            |            | $b_{19}$ | 10010    | $\overline{p_1 p_2}$ | $D_1 D_4$         | 13  |

Таблица блока БЛУ формируется на основе таблицы замены логических условий и использует коды классов  $B_i \in \Pi_C$ . Она состоит из столбцов  $B_i, K_A(B_i), K_B(B_i), P_1, \dots, P_Q, i$  (табл. 3).

**Таблица 3.** Таблица блока замены логических условий КМУУ  $U_2(\Gamma_1)$

| $B_i$ | $K_A(B_i)$ | $K_B(B_i)$ | $p_1$ | $p_2$ | $i$ |
|-------|------------|------------|-------|-------|-----|
| $B_1$ | 0000*      | 00         | $x_1$ | $x_4$ | 1   |
| $B_2$ | 001**      | 00         | $x_3$ | –     | 2   |
| $B_3$ | *****      | *1         | $x_1$ | $x_2$ | 3   |
| $B_4$ | *****      | 1*         | –     | $x_5$ | 4   |
| $B_5$ | 111**      | 00         | $x_6$ | $x_7$ | 5   |

Из этой таблицы формируется система (12), термы которой включают те же конъюнкции, что и термы  $F_h$  (кроме конъюнкции  $P_h$ ). Так, из табл. 3 следует  $P_1 = \overline{T_1 T_2 T_3 T_4} \overline{V_1 V_2} x_1 \vee \overline{T_1 T_2 T_3} \overline{V_1 V_2} x_3 \vee \overline{V_2} x_1 \vee \overline{T_1 T_2 T_3} \overline{V_1 V_2} x_6$ .

Синтез логической схемы КМУУ  $U_2$  сводится к реализации систем (12)–(13) на ПМЛ и к реализации управляющей памяти на ППЗУ. Эти задачи достаточно изучены в [3] и в нашей статье не рассматриваются.

**Заключение.** Предлагаемый в статье метод оптимизации схемы КМУУ с преобразователем адреса микрокоманды ориентирован на уменьшение числа макроячеек ПМЛ в схеме блока БПА. Решение этой задачи достигается за счет использования нескольких источников кодов

классов псевдоэквивалентных ОЛЦ (в пределах трех). Использование метода замены логических условий позволяет уменьшить число входов макроячеек, на которые подаются логические условия. Тем самым появляется возможность использования дополнительных переменных, кодирующих классы ПОЛЦ, что ведет к уменьшению числа макроячеек в блоке адресации микрокоманд БАМ.

Однако этот выигрыш в аппаратуре сопровождается уменьшением быстродействия за счет введения блока замены логических условий БЛУ. Кроме того, этот блок потребляет некоторые ресурсы кристалла. Таким образом, использование предложенного метода имеет смысл, если суммарные затраты в блоках БАМ и БЛУ меньше аппаратурных затрат в блоке БАМ эквивалентного КМУУ  $U_1$ .

*Научная новизна* предложенного метода заключается в одновременном использовании особенностей базиса ПЛИС, а именно большого коэффициента объединения по входу, для уменьшения аппаратурных затрат в схеме БПА и метода замены логических условий для оптимизации схемы БАМ. Отметим, что при выполнении условия (10) блок БПА вообще отсутствует. *Практическая значимость* этого метода заключается в уменьшении числа микросхем при реализации схемы КМУУ, что позволяет получить схемы, обладающие меньшей стоимостью по сравнению с известными из литературы аналогами. Рассмотренные нами примеры показали, что число макроячеек в блоке БПА уменьшается на 60–70%. При этом общее число макроячеек в схеме КМУУ  $U_2(\Gamma_1)$  до 10% меньше, чем в КМУУ  $U_1(\Gamma_1)$ .

Для повышения эффективности метода необходимо разработать алгоритм адресации микрокоманд КМУУ, уменьшающий число ОЛЦ, адреса выходов которых должны преобразовываться. Это относится к дальнейшему направлению наших исследований, как и проверка возможности использования этого метода для случая ПЛИС, использующих технологию программируемых логических матриц [9], а также в «системах–на–кристалле» [10].

Окончание на стр. 88

1. *Baranov S.* Logic Synthesis for Control Automata. – Kluwer Academic Publishers, 1994. – 312 p.
2. *Грушвицкий Р.И., Мурсаев А.Х., Угрюмов Е.П.* Проектирование систем с использованием микросхем программируемой логики. – СПб: БХВ. – Петербург, 2002. – 608 с.
3. *Соловьев В.В.* Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия–ТЕЛЕКОМ, 2001. – 636 с.
4. *Barkalov A., Titarenko L.* Logic Synthesis for Compositional Microprogram Control Units. – Berlin: Springer, 2008. – 272 p.
5. *Altera devices overview.* – [http://www.altera.com/products/devices/common/dev-family\\_overview.html](http://www.altera.com/products/devices/common/dev-family_overview.html)
6. *Xilinx CPLDs.* – [http://www.ilinx.com/products/silicon\\_solutions/cplds/index.htm](http://www.ilinx.com/products/silicon_solutions/cplds/index.htm)
7. *Баркалов А.А., Зеленёва И.Я., Лаврик А.С.* Использование особенностей ПЛИС для оптимизации схемы устройства управления / Наук. пр. Донецького нац. техн. ун-ту. Серія «Інформатика, кібернетика і обчислювальна техніка» (ІКОТ–2008). Вип. 9 (132) – Донецьк: ДонНТУ. – 2008. – С. 178–182.
8. *Оптимизация устройства управления с преобразователем адреса микрокоманд / А.А. Баркалов, С.А. Ковалев, А.А. Красичков и др.* // Материалы Девятого Междунар. науч.-практ. семинара. В 3-х кн. – Таганрог. Кн. 3. – 2008. – С. 12–20.
9. *CoolRunner CPLD Datasheet.* – <http://www.xilinx.com/support/documentation/coolrunner-ii.htm>
10. *Maxfield C.* The Design Warrior's Guide to FPGAs. – Amsterdam: Elsevier, 2004. – 541 p.

Поступила 31.03.2009

Тел. для справок: (XXX) 301-0723 (Донецк)

E-mail: [A.Barkalov@iie.uz.zgora.pl](mailto:A.Barkalov@iie.uz.zgora.pl)

© А.А. Баркалов, Л.А. Титаренко, А.С. Лаврик, 2009

