

УДК 681:324

**Р.М. Бабаков (канд. техн. наук, доц.), И.В. Ярош**  
Донецкий национальный технический университет, г. Красноармейск  
кафедра компьютерных наук  
E-mail: cpld@mail.ru; yarosh.irina@inbox.ru

## ИСПОЛЬЗОВАНИЕ ТРАНЗИТНЫХ СОСТОЯНИЙ В МИКРОПРОГРАММНОМ АВТОМАТЕ С ОПЕРАЦИОННЫМ АВТОМАТОМ ПЕРЕХОДОВ

*Проанализированы возможные способы реализации автоматного перехода в микропрограммном автомате с операционным автоматом переходов при невозможности непосредственной реализации перехода с помощью одной из имеющихся операций переходов. Предложен подход к реализации таких переходов путем введения дополнительных состояний, называемых в работе транзитными. Характерной особенностью транзитных состояний является то, что их входящие и исходящие переходы реализуются посредством заданного множества операций переходов без необходимости реализации данных переходов каноническим способом. Таким образом, использование транзитных состояний в некоторых случаях может способствовать снижению аппаратных затрат в логической схеме автомата. Рассмотрен пример реализации автоматных переходов заданной граф-схемы алгоритма с использованием транзитных состояний, демонстрирующий возможность существования различных способов использования транзитных состояний для реализации переходов заданного микропрограммного автомата.*

**Ключевые слова:** микропрограммный автомат, операционный автомат переходов, операция переходов, транзитные состояния, аппаратные затраты.

**Введение.** В современных вычислительных системах функция управления узлами системы обычно отводится устройству управления, которое может быть реализовано в виде микропрограммного автомата (МПА) [1-3]. Среди множества известных структур МПА выделяется микропрограммный автомат с операционным автоматом переходов (МПА с ОАП) [4-7]. В основе функционирования МПА с ОАП лежит принцип операционного преобразования кодов состояний [4], в соответствии с которым преобразование кодов состояний при автоматных переходах осуществляется посредством арифметико-логических операций, называемых операциями переходов (ОП). Данный принцип определяет представление схемы формирования переходов МПА с ОАП в виде операционного автомата, называемого операционным автоматом переходов [8, 9].

Использование МПА с ОАП требует решения ряда задач, затрагивающих вопросы структурного синтеза и повышения эффективности данного класса микропрограммных автоматов. Одной из актуальных задач является задача снижения аппаратных затрат в логической схеме МПА с ОАП.

Целью данной работы является решение данной задачи, достигаемое за счет уменьшения сложности схемы в части реализации микропрограммных переходов, выполняемых каноническим способом.

Основная идея метода заключается в разбиении одного «канонического» перехода на несколько «операционных», что достигается за счет введения дополнительных состояний, называемых в данной работе транзитными.

Недостатком использования транзитных состояний является увеличение среднего числа тактов работы микропрограммного автомата, затрачиваемых на одно выполнение реализуемого алгоритма. В тех случаях, когда увеличение времени выполнения алгоритма,

вызванное транзитными состояниями, не является критичным, использование предлагаемого подхода может способствовать снижению аппаратных затрат в логической схеме микропрограммного автомата.

**Микропрограммный автомат с операционным автоматом переходов.** На рис. 1 изображена базовая структурная схема МПА с ОАП [10].

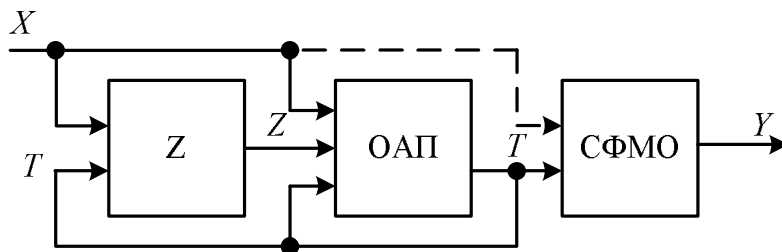


Рисунок 1 – Структура микропрограммного автомата с операционным автоматом переходов

Z-подсхема на основании кода текущего состояния  $T$  и сигналов логических условий  $X$  формирует код  $Z$  операции переходов, поступающий в операционный автомат переходов (ОАП). ОАП, внутренняя структура которого приведена на рис. 2, с помощью комбинационных схем  $KC_1$ - $KC_N$  выполняет одну из  $N$  операций переходов над сигналами  $T$  и  $X$ , причем выбор одного из результатов  $d_1$ - $d_N$  осуществляется путем мультиплексирования под управлением сигналов  $Z$ . Результат  $d$ , являющийся кодом состояния перехода, поступает с выхода мультиплексора  $MX$  в регистр памяти (РП).

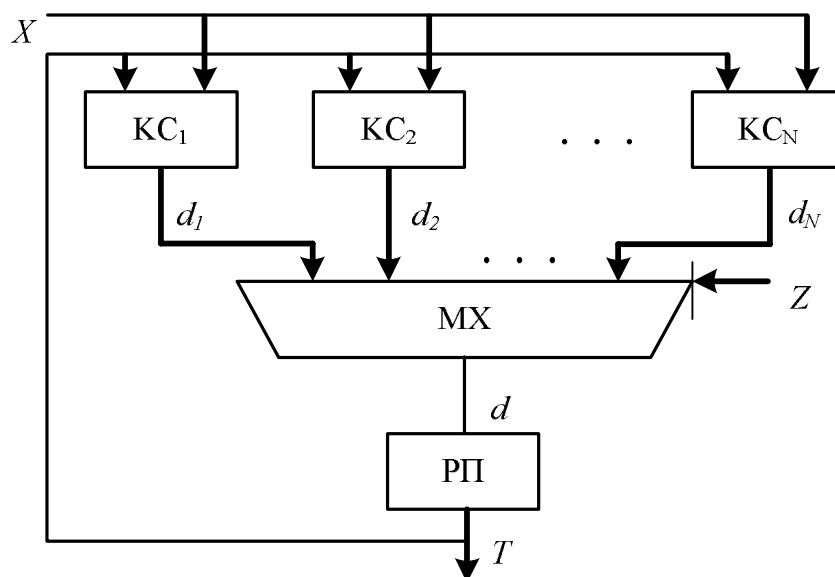


Рисунок 2 – Структура операционного автомата переходов

Код текущего состояния  $T$  поступает из РП в схему формирования микроопераций (СФМО) (рис. 1). Если МПА с ОАП реализует граф-схему алгоритма (ГСА) [2], отмеченную состояниями автомата Мили, в СФМО также поступают сигналы логических условий  $X$ . Поскольку схема СФМО реализует функцию выходов автомата и не имеет отношения к функции переходов, ее структура, элементный базис и метод синтеза в данном случае являются несущественными. Поскольку коды состояний формируются в процессе синтеза ОАП, их значения должны учитываться при синтезе СФМО.

**Принцип использования транзитных состояний.** Пусть в процессе синтеза МПА с ОАП необходимо реализовать переход из состояния  $a_m$  в состояние  $a_n$ , причем к моменту его реализации структурные коды  $K_S(a_m)$  и  $K_S(a_n)$  известны (например, вследствие предварительной реализации других переходов с участием данных состояний). Предположим также, что ни одна операция из множества операций переходов не подходит для реализации перехода из  $a_m$  в  $a_n$  при заданных значениях  $K_S(a_m)$  и  $K_S(a_n)$ .

В этом случае для решения задачи реализации перехода из  $a_m$  в  $a_n$  есть два наиболее очевидных способа:

1. Реализация перехода с помощью дополнительной ОП.
2. Реализация перехода каноническим способом.

Первый способ приводит к добавлению в схему ОАП дополнительной комбинационной схемы и подключению ее выхода к мультиплексору результата. Недостатком способа является увеличение аппаратных затрат в схеме ОАП. Достоинством способа является возможность использования добавляемой ОП для реализации других переходов автомата.

Второй способ приводит к модификации комбинационной схемы, отвечающей за реализацию части переходов автомата каноническим способом. Недостатком способа является увеличение аппаратных затрат в данной комбинационной схеме и необходимость ее повторного синтеза. Достоинством способа является отсутствие изменений в мультиплексоре результата.

Существует еще один подход к решению данной задачи, позволяющий, в некоторых случаях, реализовать данный переход посредством имеющихся операций переходов. Выполним следующее:

1. Добавим в заданную ГСА, содержащую  $M$  состояний  $a_0 - a_{M-1}$ , последовательность из  $k$  операторных вершин, в которых не формируются никакие микрооперации. Последовательность включим в разрыв ветви, соответствующей переходу из состояния  $a_m$  в состояние  $a_n$  и отметим состояниями  $a_M, \dots, a_{M+k-1}$ , где  $k \geq 1$ .

2. Закодируем состояния  $a_M, \dots, a_{M+k-1}$  структурными кодами  $K_S(a_M), \dots, K_S(a_{M+k-1})$  разрядности  $R = \lceil \log_2 M \rceil$  таким образом, чтобы:

– коды  $K_S(a_M), \dots, K_S(a_{M+k-1})$  были уникальны и свободны (не использовались для кодирования других состояний заданной ГСА);

– переходы  $a_m \rightarrow a_M, a_M \rightarrow a_{M+1}, \dots, a_{M+k-2} \rightarrow a_{M+k-1}, a_{M+k-1} \rightarrow a_n$  могли быть реализованы посредством операций переходов из заданного множества ОП.

Назовем состояния, добавляемые в исходный автомат с целью реализации переходов автомата посредством имеющегося множества операций переходов, *транзитными состояниями*. Смысл данного термина определяется вспомогательной ролью этих состояний в процессе синтеза автомата.

Пусть, например,  $K_S(a_m)=0101_2$ ,  $K_S(a_n)=1111_2$ . Ограничим множество операций переходов образовано следующими ОП:

$$O_1: K_{I_1}(a^{t+1}) = (K_{I_1}(a^t) + 13) \bmod 16, \quad (1)$$

$$O_2: K_{I_2}(a^{t+1}) = K_{I_2}(a^t) \div 4, \quad (2)$$

$$O_3: K_{I_3}(a^{t+1}) = \overline{K_{I_3}(a^t)}, \quad (3)$$

где  $a^t$  – текущее состояние,  $a^{t+1}$  – состояние перехода.

Для операций  $O_1$  и  $O_2$  будем интерпретировать структурные коды состояний как беззнаковые целые в диапазоне  $[0; 15]$ . Это позволяет задать т.н. промежуточные коды состояний  $K_{I_i}(a_j)$  следующим образом:  $K_{I_1}(a_m) = K_{I_2}(a_m) = 5_{10}$ ,  $K_{I_1}(a_n) = K_{I_2}(a_n) = 15_{10}$ . Для операции  $O_3$  будем рассматривать промежуточные коды состояний как четырехразрядные двоичные векторы, совпадающие с соответствующими структурными кодами:  $K_{I_3}(a_m) = K_S(a_m) = 0101_2$ ,  $K_{I_3}(a_n) = K_S(a_n) = 1111_2$ .

При данных условиях реализация перехода из состояния  $a_m$  в состояние  $a_n$  с помощью транзитных состояний возможна различными способами, некоторые из которых показаны на рис. 3.

На приведенном рисунке в операторных вершинах, отмеченных состояниями автомата Мура, записаны соответствующие состояниям промежуточные коды  $K_{I_1}(a)$  и  $K_{I_2}(a)$ , причем код  $K_{I_2}(a)$  совпадает с соответствующим структурным кодом.

Ребра обозначены операциями переходов, выполняющимися над одним из промежуточных кодов текущего состояния: обозначение «+13» соответствует операции  $O_1$ , «÷4» – операции  $O_2$ , «not» – операции  $O_3$ . Добавленные вершины обозначены затемненным фоном.

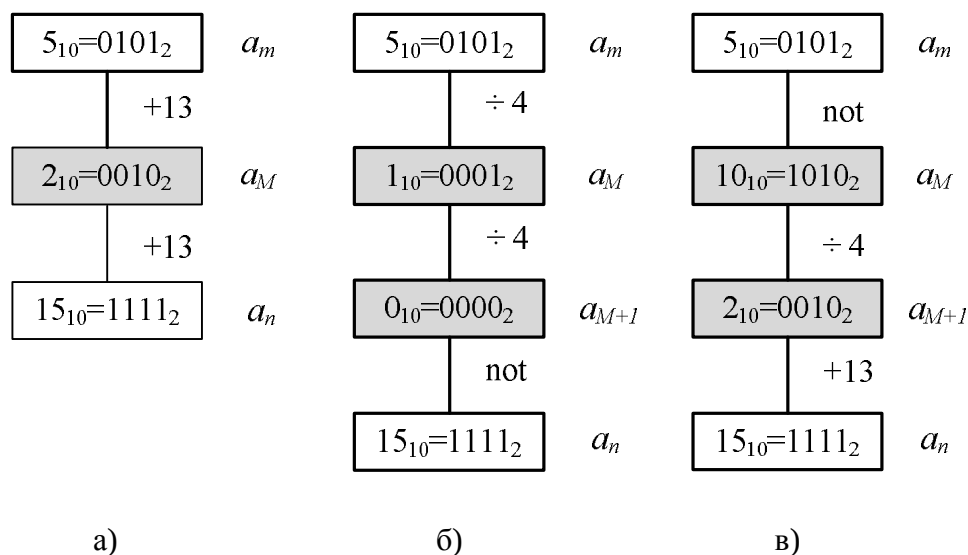


Рисунок 3 – Примеры реализации автоматного перехода с использованием дополнительных состояний

На рис. 3, а для перехода из состояния  $a_m$  в состояние  $a_n$  использовано одно транзитное состояние  $a_M$ . Выбранный для него структурный код  $K_S(a_M) = 0010_2$  позволяет реализовать переходы  $a_m \rightarrow a_M$  и  $a_M \rightarrow a_n$  с помощью одной и той же операции  $O_1$ . Отметим, что действие «mod 16» в выражении (1) на уровне логической схемы реализуется путем отбрасывания переноса из старшего разряда суммы.

На рис. 3, б и в для перехода из  $a_m$  в  $a_n$  использованы два транзитных состояния. Их выполнение занимает два такта работы схемы автомата вместо одного такта в случае рис. 3, а. Таким образом, при реализации перехода с помощью транзитных состояний следует стремиться к уменьшению их количества в добавляемой последовательности.

**Пример использования транзитных состояний.** В некоторых случаях использование транзитных состояний позволяет добиться реализации всех переходов автомата посредством заданного множества операций переходов.

Пусть МПА задан ГСА  $\Gamma_1$  (см. рис. 4). Отметка ГСА состояниями автомата Мура позволяет выделить  $M=9$  состояний  $a_0-a_8$ , для которых разрядность кода состояния  $R=4$ . Пусть множество операций переходов образовано операциями  $O_1-O_3$ , задаваемыми выражениями (1)-(3) соответственно.

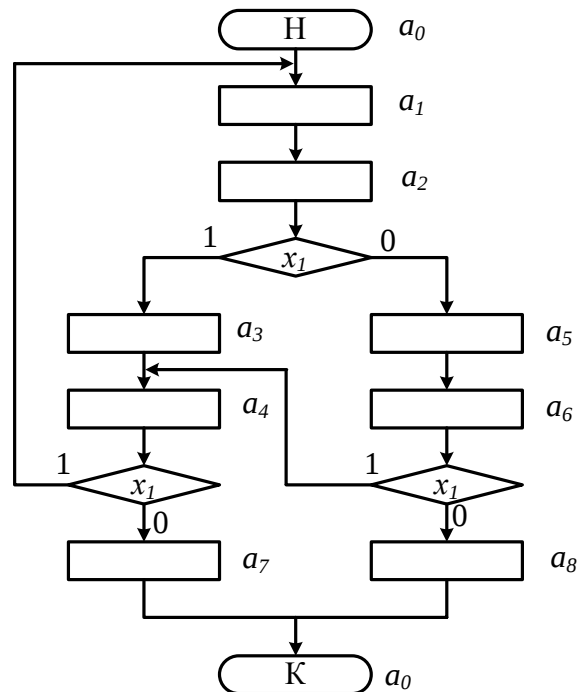


Рисунок 4 – Граф-схема алгоритма  $\Gamma_1$

Использование транзитных состояний приводит к изменению исходной ГСА. На рис. 5 представлены два различных варианта синтеза МПА с ОАП с помощью транзитных состояний. Можно заметить, что в некоторых случаях при выбранных структурных кодах состояний один и тот же переход может быть реализован различными ОП. Например, в ГСА  $\Gamma_{1-1}$  и  $\Gamma_{1-2}$  переход  $a_6 \rightarrow a_8$  может быть реализован как операцией  $O_1$ , так и операцией  $O_3$ . Также в ГСА  $\Gamma_{1-2}$  переход  $a_3 \rightarrow a_4$  может быть реализован операцией  $O_2$  либо операцией  $O_3$ . Выбор той или иной ОП обычно не имеет принципиального значения, поскольку аппаратные затраты в комбинационных схемах, соответствующих ОП  $O_1-O_3$ , не зависят от количества автоматных переходов, реализуемых с помощью данной ОП. Выбор ОП для реализации конкретных переходов учитывается лишь при синтезе Z-подсхемы.

Пример на рис. 5, б показывает, что в одной ГСА допустимо использование нескольких последовательностей транзитных состояний, длины которых в общем случае могут быть различными. Поиск наилучшего варианта использования транзитных состояний требует применения специальных методов, разработка которых выходит за рамки данной работы.

Использование транзитных состояний в процессе синтеза МПА с ОАП возможно при условии допустимости преобразования исходного автомата и приводит, в общем случае, к следующим результатам:

1. Среднее время выполнения алгоритма, интерпретируемого МПА, увеличивается за счет тактов, затрачиваемых на переходы в транзитные состояния.
2. Изменения в логической схеме ОАП отсутствуют, поскольку не изменяются ни разрядность структурного кода состояния, ни множество операций переходов.
3. Z-подсхема модифицируется таким образом, чтобы формировать нужные коды ОП в пределах последовательностей транзитных состояний.

4. Схема СФМО модифицируется таким образом, чтобы при нахождении автомата в транзитных состояниях исключить формирование любых микроопераций. При этом нахождение МПА в транзитных состояниях не должно сказываться на функционировании объекта, управляемого микропрограммным автоматом.

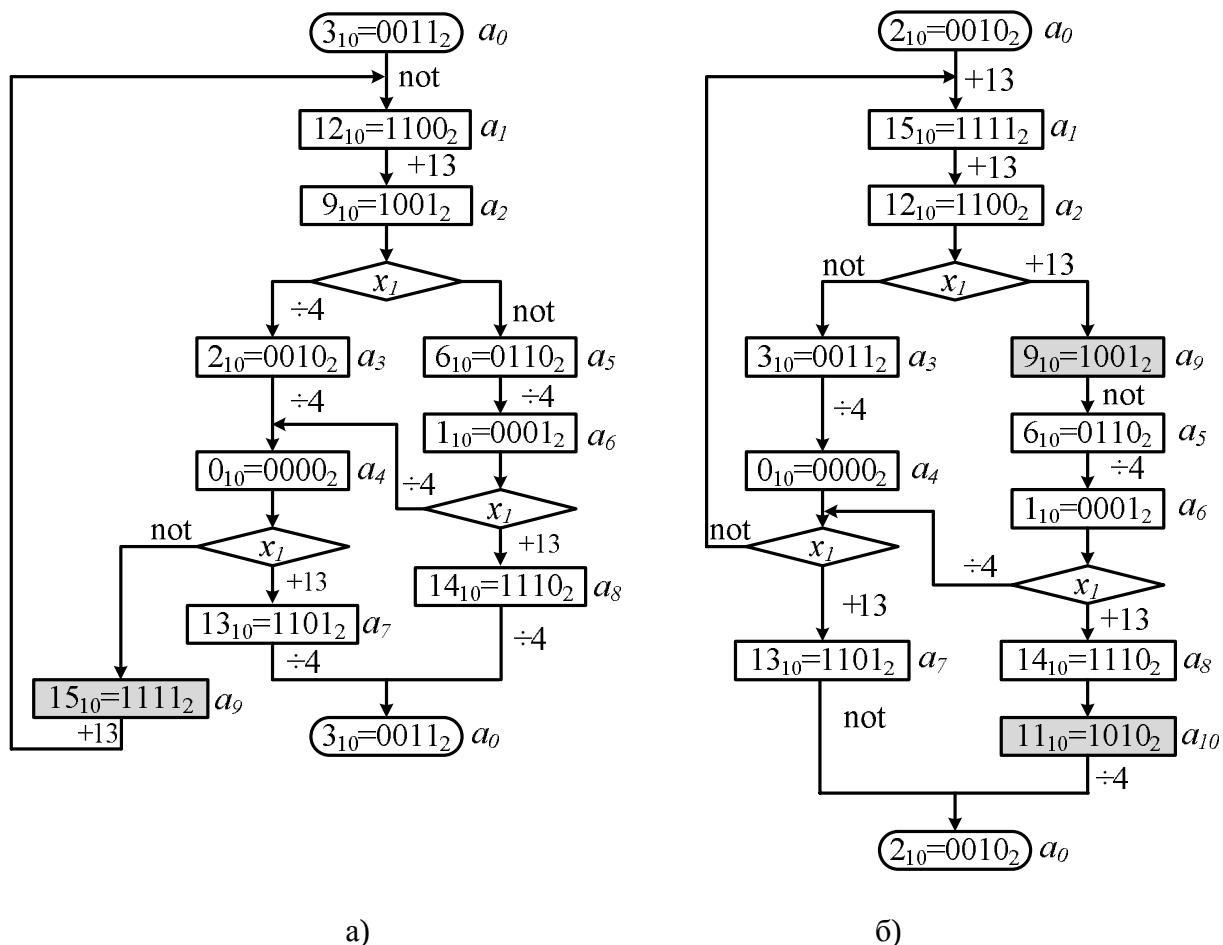


Рисунок 5 – Результаты синтеза МПА с ОАП по ГСА  $\Gamma_1$ : ГСА  $\Gamma_{1-1}$  с одним транзитным состоянием (а) и ГСА  $\Gamma_{1-2}$  с двумя транзитными состояниями (б)

### Выводы.

Как показали исследования, решение задачи минимизации аппаратных затрат в логической схеме микропрограммного автомата, являющееся целью данной работы, возможно за счет реализации части автоматных переходов с использованием транзитных состояний. Необходимо учитывать, что предложенный в работе подход имеет свои достоинства и недостатки. Основным достоинством является то, что автоматный переход, который не может быть реализован ни одной из имеющихся операций переходов, преобразовывается в несколько переходов, каждый из которых может быть реализован посредством одной из заданных операций переходов. В результате уменьшается число автоматных переходов, реализуемых каноническим способом, что приводит к снижению аппаратных затрат в том блоке КС<sub>i</sub> операционного автомата переходов, который отвечает за реализацию части переходов каноническим способом.

Вместе с тем при использовании транзитных состояний следует учитывать следующие моменты:

1. Введение в интерпретируемую ГСА дополнительных состояний приводит к увеличению среднего числа переходов, выполняемых за время работы микропрограммного

автомата. Увеличение временных затрат в микропрограммном автомате, выступающем в качестве устройства управления вычислительной системы, негативно сказывается на временных характеристиках системы в целом. Если введение транзитных состояний приводит к нарушению допустимых временных критериев проектирования вычислительной системы, всегда имеется возможность отказаться от реализации части переходов с помощью транзитных состояний, обеспечив компромисс по критериям быстродействия и аппаратурных затрат.

2. Как было отмечено, уменьшение числа переходов, реализуемых каноническим способом, способствует уменьшению аппаратурных затрат в логической схеме ОАП. Вместе с тем увеличение общего числа состояний может приводить к увеличению аппаратурных затрат в Z-подсхеме и схеме СФМО, что может нивелировать достигаемый выигрыш с схеме ОАП. Таким образом, критерием целесообразности использования предлагаемого подхода являются результирующие аппаратурные затраты в логической схеме МПА с ОАП, которые могут быть определены, например, путем моделирования схемы автомата с использованием специализированных САПР цифровых устройств.

3. Рассмотренный выше пример показывает, что для автомата, интерпретирующего заданную ГСА, существует в общем случае множество вариантов использования транзитных состояний. Это ставит задачу поиска оптимального варианта, при котором достигается максимальное уменьшение затрат аппаратуры в схеме автомата при приемлемом увеличении среднего времени работы. Поиск таких вариантов требует разработки специальных методов синтеза МПА с ОАП, формализующих предлагаемый в данной работе подход к реализации переходов автомата с помощью транзитных состояний.

#### Список использованной литературы

1. Глушков, В.М. Синтез цифровых автоматов / В.М. Глушков. – М.: Физматгиз, 1962. – 476 с.
2. Баранов, С.И. Синтез микропрограммных автоматов / С.И. Баранов. – Л.: Энергия, 1979. – 232 с.
3. Майоров С.А. Структура электронных вычислительных машин / С.А. Майоров, Г.И. Новиков. – Л.: Машиностроение, 1979. – 384 с.
4. Баркалов, А.А. Операционное формирование кодов состояний в микропрограммных автоматах / А.А. Баркалов, Р.М. Бабаков. – Кибернетика и системный анализ. – 2011. – №2. – С. 21 – 26.
5. Баркалов, А.А. Организация устройств управления с операционной адресацией / А.А. Баркалов, Р.М. Бабаков // Управляющие системы и машины. – 2008. – №6. – С. 34 – 39.
6. Баркалов, А.А. Организация операционной части в управляющем автомате с операционным автоматом переходов / А.А. Баркалов, Р.М. Бабаков // Наукові праці Донецького національного технічного університету. Серія «Обчислювальна техніка та автоматизація». – Донецьк: ДонНТУ, 2011. – Випуск 21 (183). – С. 149 – 156.
7. Баркалов, А.А. Структурное представление процесса синтеза управляющих автоматов с операционным автоматом переходов / А.А. Баркалов, Р.М. Бабаков // Управляющие системы и машины. – 2011. – № 3. – С. 47 – 53.
8. Бабаков, Р.М. Реализация функций микропрограммного автомата на базе операционного автомата / Р.М. Бабаков, А.А. Баркалов // Управляющие системы и машины. – 2015. – №5. – С. 22-29.
9. Бабаков, Р.М. Алгебраическая интерпретация микропрограммного автомата с операционным автоматом переходов. / Р.М. Бабаков, А.А. Баркалов // Кибернетика и системный анализ. – 2016. – №2. – С. 22 – 29.
10. Бабаков Р.М. Операционный автомат переходов. / Р.М. Бабаков, И.В. Ярош. // Наукові

праці Донецького національного технічного університету. Серія «Обчислювальна техніка та автоматизація». – Красноармейск: ДонНТУ, 2015. – Випуск 1 (28). – С. 33 – 40.

### References

1. Glushkov, V.M. (1962), *Sintez tsifrovih avtomatov* [Synthesis of digital automata], Fizmatgiz, Moscow, Russia.
2. Baranov, S.I. (1979), *Sintez mikroprogramnih avtomatov* [Synthesis of microprogram automata], Energiya, Leningrad, Russia.
3. Majorov, S.A. and Novikov, G.I. (1979). *Struktura elektronnih vichislitel'nykh mashin* [Structure of computers], Mashinostroenie, Leningrad, Russia.
4. Barkalov, A.A. and Babakov R.M. (2011), "Operational forming of state codes in microprogram automata", *Kibernetika i sistemny analiz*, vol. 2, pp. 21 – 26.
5. Barkalov, A.A. and Babakov R.M. (2008), "Organization of control unit with operational addressing", *Upravljajushie sistemy i mashiny*, vol. 6, pp. 34 – 39.
6. Barkalov, A.A. and Babakov R.M. (2008), "Organization of operational part in control unit with datapath of transitions", *Sbornik naukovih prats Donetskogo natsionalnogo tehnicnogo universitetu, seriya "Obchislualna tehnika ta avtomatizatsiya"*, vol. 21, no. 183, pp. 149 – 156.
7. Barkalov, A.A. and Babakov R.M. (2011), "Structural representation of process of synthesis of control units with datapath of transitions", *Upravljajushie sistemy i mashiny*, vol. 3, pp. 47 – 53.
8. Babakov, R.M. and Barkalov A.A. (2015), "Realization of functions of microprogram final-state machine on the base of datapath", *Upravljajushie sistemy i mashiny*, vol. 5, pp. 22 – 29.
9. Babakov, R.M. and Barkalov A.A. (2016), "Algebraic interpretation of microprogram final-state machine with datapath of transitions", *Kibernetika i sistemny analiz*, vol. 2, pp. 22 – 29.
10. Babakov, R.M. and Yarosh I.V. (2015), "Datapath of transitions", *Sbornik naukovih prats Donetskogo natsionalnogo tehnicnogo universitetu, seriya "Obchislualna tehnika ta avtomatizatsiya"*, vol. 1, no. 28, pp. 33 – 40.

Поступила в редакцію:  
10.04.2016 г.

Рецензент:  
д-р.техн. наук, проф. Зори А.А.

**Р.М. Бабаков, І.В. Ярош**

**ДВНЗ «Донецький національний технічний університет»**

**Використання транзитних станів в мікропрограмному автоматі з операційним автоматом переходів.** Проаналізовані можливі засоби реалізації автоматного переходу в мікропрограмному автоматі з операційним автоматом переходів при неможливості безпосередньої реалізації переходу за допомогою однієї із доступних операцій переходів. Запропоновано підхід до реалізації таких переходів шляхом введення додаткових станів, що названі в роботі транзитними. Характерною особливістю транзитних станів є те, що їхні вхідні і вихідні переходи реалізуються за допомогою заданої множини операцій переходів без необхідності реалізації даних переходів канонічним способом. Таким чином, використання транзитних станів в деяких випадках може сприяти зниженню апаратних витрат в логічній схемі автомату. Розглянуто приклад реалізації автоматних переходів заданої граф-схеми алгоритму із використанням транзитних станів. Приклад демонструє можливість існування різних способів використання транзитних станів для реалізації переходів заданого мікропрограмного автомата.

**Ключові слова:** мікропрограмний автомат, операційний автомат переходів, операція переходів, транзитні стани, апаратні витрати.

**R.M. Babakov, I.V. Yarosh****Donetsk National Technical University**

**Using of transit states in microprogram final-state machine with datapath of transitions.** In this paper, the object of study is the microprogram final-state machine with datapath of transitions. This type of final-state machines is one way of implementing of the control unit, which is part of all modern digital systems, and coordinates the work of other units of the system. In the microprogram final-state machine with the datapath of transitions the basic way of realization of automaton transitions are operations of transitions. Usually always there are transitions which cannot be realized by operations of the given set of operations of transitions. Such transitions should be realized by additional operations of transitions, or by canonic way. Both these ways lead to increasing of hardware expenses in the logic circuit of the microprogram final-state machine. In article the approach consisting in replacement of such transition by two transitions, connected with an additional state is offered. Such state is named by a transit state as its role consists in the organization of transitions between two base states of the automatic machine. The main feature of a transit state is that input and output transitions are realized by operations of transitions from the given set of operations of transitions. As a result, use of additional operation of transitions or transition realization by canonic way is not required. It leads to reduction of hardware expenses in the logic circuit of the final-state machine. A lack of use of transit states is the increase in an average operating time of work of the final-state machine. For same flow-chart realization of transitions with use of transit states is possible, generally, in the various ways. In article the example showing two ways of realization of transitions of flow-chart with use of transit states is considered. Thus, working out of the formalized methods allowing optimum of using of transit states in each specific case is important. The given problem is a part of the subsequent researches in the field of the theory of digital final-state machines.

**Keywords:** microprogram final-state machine, datapath of transitions, operation of transitions, transit states, hardware expenses.



**Бабаков Роман Маркович**, Украина, канд. тех. наук, доцент, доцент кафедры компьютерных наук. ГВУЗ «Донецкий национальный технический университет» (пл. Шибанкова, 2, г. Красноармейск, 85300, Украина). Основное направление научной деятельности – разработка и исследование методов синтеза цифровых устройств управления.



**Ярош Ирина Викторовна**, Украина, ассистент кафедры компьютерных наук. ГВУЗ «Донецкий национальный технический университет» (пл. Шибанкова, 2, г. Красноармейск, 85300, Украина). Основное направление научной деятельности – разработка и исследование систем с элементами искусственного интеллекта.