

Р.М. Бабаков, А.А. Баркалов

Модификация микропрограммного автомата с операционным автоматом переходов и заменой входных переменных

Предложено новое структурное решение, способствующее снижению аппаратных затрат в логической схеме микропрограммного автомата с операционным автоматом переходов. Наряду с методом замены входных переменных используется метод сопоставления операций переходов состояниям автомата, что позволяет добиться одновременного снижения аппаратных затрат в операционном автомате переходов и схеме формирования кодов операций. Определено условие эффективности предложенной структуры автомата в сравнении со структурами-прототипами.

Ключевые слова: микропрограммный автомат, операционный автомат переходов, замена входных переменных, аппаратные затраты.

Запропоновано нове структурне рішення, яке сприяє зниженню апаратних витрат у логічній схемі мікропрограмного автомата з операційним автоматом переходів. Поряд з методом заміни вхідних змінних використовується метод зіставлення операцій переходів станам автомата, що дозволяє домогтися одночасного зниження апаратних витрат в операційному автоматі переходів і схемі формування кодів операцій. Визначено умову ефективності запропонованої структури автомата в порівнянні зі структурами-прототипами.

Ключові слова: мікропрограмний автомат, операційний автомат переходів, заміна вхідних змінних, апаратні витрати.

Введение. Существенный структурный элемент современных вычислительных систем – устройство управления, одной из интерпретаций которого является микропрограммный автомат (МПА) [1, 2]. При синтезе схемы МПА одной из актуальных научных проблем есть минимизация аппаратных затрат, решаемая путем разработки их новых структур и методов синтеза [3].

В МПА функция переходов может быть представлена в виде множества частичных функций, что приводит к структуре микропрограммного автомата с операционным автоматом переходов (МПА с ОАП) [4, 5]. При этом схема переходов автомата представляется в виде набора функциональных узлов, аппаратные затраты в каждом из которых не зависят или зависят незначительно от количества автоматных переходов, реализуемых этим узлом. Это делает схему формирования переходов МПА структурно схожей с операционным автоматом, в котором выбор операции переходов (ОП), выполняемой над кодом текущего состояния, может зависеть как от самого кода текущего состояния, так и от входных сигналов автомата.

В статье предлагается новая структурная модификация МПА с ОАП, в которой принцип сопоставления операций переходов состояниям автомата сочетается с методом замены входных переменных. Цель предлагаемого решения – дополнительное снижение аппаратных затрат в сравнении со структурами-прототипами.

Структурные модели микропрограммного автомата с операционным автоматом переходов

Базовая структура МПА с ОАП, обозначенная в рамках данной статьи символом U_1 , показана на рис. 1 [4].

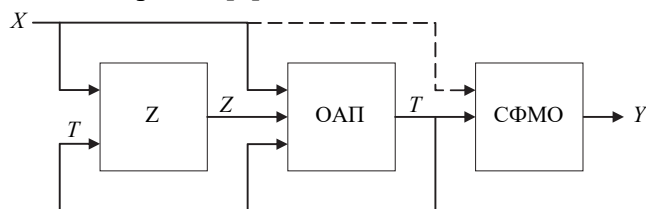


Рис. 1. Структура U_1

Блок ОАП в структуре представляет собой операционный автомат, выполняющий преобразование кода текущего состояния T и множества входных сигналов X под управлением кода Z , в соответствии с выражением (1).

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

где $T = \{T_1, \dots, T_R\}$, $X = \{x_1, \dots, x_L\}$, $Z = \{z_1, \dots, z_{R_z}\}$.

Код операции Z формируется Z -подсхемой в соответствии с выражением (2) и определяет операцию переходов, выполняемую ОАП в каждом такте работы устройства.

$$Z = Z(T, X). \quad (2)$$

Схема формирования микроопераций (СФМО) реализует функцию выходов автомата, формируя множество микроопераций Y в соответствии с выражением (3) в случае автомата Мили и (4) в случае автомата Мура. Таким образом, наличие в структуре МПА с ОАП связи, показанной на рис. 1 пунктиром, позволяет считать данную структуру автоматом Мили, отсутствие связи – автоматом Мура.

$$Y = Y(T, X). \quad (3)$$

$$Y = Y(T). \quad (4)$$

Принцип функционирования МПА с ОАП и его математическая модель изложены в работах [4] и [5] соответственно.

В работе [6] предложена модификация структуры U_1 , позволяющая сопоставлять операцию переходов отдельному состоянию автомата (рис. 2). Особенность данной структуры (обозначаемой в статье символом U_2) – отсутствие входных сигналов автомата на входе Z -подсхемы, что позволяет снизить аппаратные затраты в Z -подсхеме в сравнении со структурой U_1 .

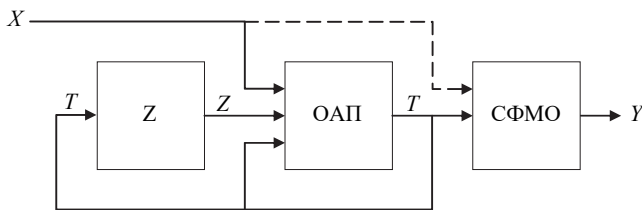


Рис. 2. Структура U_2

В структуре U_2 Z -подсхема реализует функцию

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

в соответствии с которой каждому состоянию автомата подходит только одна ОП, реализующая все переходы из данного состояния. Поскольку сигналы X отсутствуют в списке аргументов функции (5), возможность сопос-

тавить операцию отдельному автоматному переходу отсутствует.

В [7] предложена модификация структуры U_1 , также позволяющая снизить сложность Z -подсхемы путем уменьшения числа входов Z -подсхемы. Данная структура показана на рис. 3 и обозначена символом U_3 .

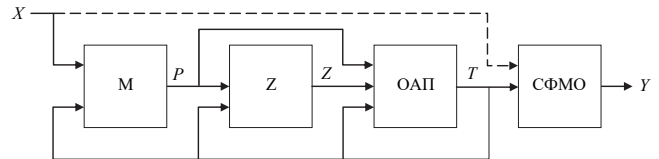


Рис. 3. Структура U_3

В структуре U_3 использован известный метод замены входных переменных, заключающийся в следующем [3]. Если любой переход зависит не более чем от G входных сигналов, множество входных сигналов X может быть заменено множеством $P = \{p_1, \dots, p_G\}$, а в структуру автомата добавляется специальная M -подсхема, синтезируемая в базе мультиплексов и выполняющая преобразование

$$P = P(X, T). \quad (6)$$

При этом выигрыш в аппаратных затратах достигается при $G \ll L$ [3, 7].

Модификация МПА с ОАП и заменой входных переменных

Объединим особенности структур U_2 и U_3 в одной структуре. В результате получим структуру МПА с ОАП с заменой входных переменных и сопоставлением операций переходов состояниям автомата (рис. 4). Обозначим данную структуру символом U_4 .

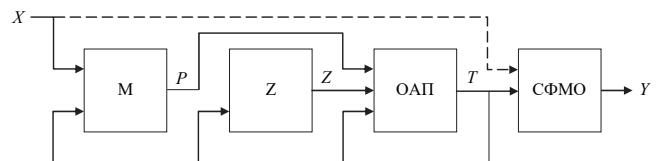


Рис. 4. Структура U_4

В структуре U_4 на вход Z -подсхемы подается только код текущего состояния автомата T , в результате чего код операции переходов Z формируется в соответствии с выражением (5) по аналогии со структурой U_2 . Поскольку входные сигналы отсутствуют на входе Z -подсхемы,

использование M -подсхемы в данной структуре не позволяет уменьшить сложность Z -подсхемы, как это было в структуре U_3 . В то же время сигналы P , формируемые M -подсхемой, поступают на вход ОАП, приводя к следующему.

Пусть некоторая граф-схема алгоритма (ГСА) Γ , отмеченная состояниями автомата Мура, содержит фрагмент, изображенный на рис. 5.

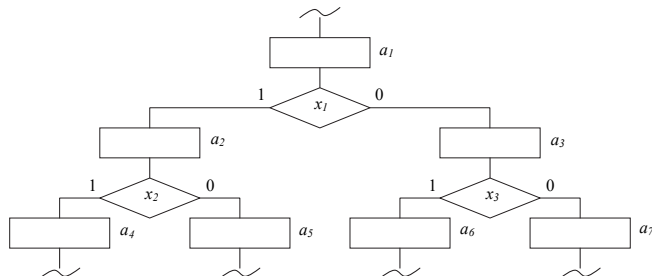


Рис. 5. Фрагмент ГСА Γ

Предположим, что общее количество состояний ГСА Γ таково, что для их кодирования требуется минимум $R = 4$ двоичных разряда. Закодируем состояния так, как показано в таблице.

a_i	a_1	a_2	a_3	a_4	a_5	a_6	a_7
$K_I(a_i)$	1	6	5	11	8	10	7
$K_S(a_i)$	0001	0110	0101	1011	1000	1010	0111

Из таблицы следует, что $K_I(a_i)$ – промежуточный код состояния a_i , являющийся скалярной величиной, $K_S(a_i)$ – структурный код состояния a_i , являющийся двоичным представлением кода $K_I(a_i)$ в формате целого числа без знака [8].

Отметим, что если бы во всех условных вершинах заданного фрагмента ГСА было записано одно и то же логическое условие (ЛУ) (например, x_1), то все переходы в рамках данного фрагмента могли быть реализованы путем сопоставления состояниям $a_1 - a_3$ следующей операции переходов:

$$O_1: K_I(a^{t+1}) = \begin{cases} K_I(a^t) + 5, & \text{если } x_1 = 0; \\ K_I(a^t) / 2 + 5, & \text{если } x_1 = 1, \end{cases} \quad (7)$$

схемная реализация которой соответствует рис. 6. Графически данная возможность пока-

зана на рис. 6, где блок R_1 выполняет операцию логического сдвига на 1 разряд вправо, что для беззнаковых чисел эквивалентно операции деления на 2.

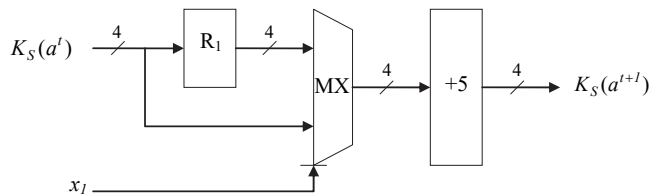


Рис. 6. Функциональная схема для операции (7)

Тот факт, что на рис. 5 в каждой условной вершине записаны разные ЛУ, в случае МПА со структурой U_2 не позволяет сопоставить операцию O_1 состояниям a_2 и a_3 . Данная ОП может быть сопоставлена только состоянию a_1 , тогда как для состояния a_2 должна быть задана отдельная ОП O_2 , использующая в своей формуле логическое условие x_2 , для a_3 – ОП O_3 , использующая ЛУ x_3 . Данные ОП могут быть заданы выражениями, подобными (7):

$$O_2: K_I(a^{t+1}) = \begin{cases} K_I(a^t) + 5, & \text{если } x_2 = 0; \\ K_I(a^t) / 2 + 5, & \text{если } x_2 = 1, \end{cases} \quad (8)$$

$$O_3: K_I(a^{t+1}) = \begin{cases} K_I(a^t) + 5, & \text{если } x_3 = 0; \\ K_I(a^t) / 2 + 5, & \text{если } x_3 = 1. \end{cases} \quad (9)$$

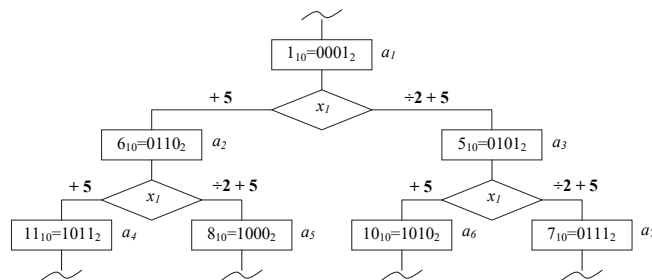


Рис. 7. Графическое представление операционной реализации переходов во фрагменте ГСА Γ с помощью операции (6) при использовании во всех условных вершинах логического условия x_1

Несмотря на общую схожесть выражений (8) и (9) с выражением (7), в схеме ОАП со структурой U_2 данным ОП будут соответствовать отдельные комбинационные схемы, подобные схеме ОП O_1 (рис. 6), но использующие различные сигналы ЛУ для управления мультиплексором (MX). При этом каждая ОП будет иметь уникальный код, учитывающийся

при построении мультиплексора в операционной части ОАП [4]. Фрагмент функциональной схемы ОАП, соответствующий операциям переходов $O_1 - O_3$, приведен на рис. 8.

В случае МПА со структурой U_4 переменные $x_1 - x_3$ в рамках фрагмента ГСА на рис. 5 заменяются переменной p_1 . Это позволяет получить результаты синтеза МПА с ОАП, приведенные на рис. 7, при использовании единственной ОП вида

$$O_4: K_I(a^{t+1}) = \begin{cases} K_I(a^t) + 5, & \text{если } p_1 = 0; \\ K_I(a^t) / 2 + 5, & \text{если } p_1 = 1. \end{cases} \quad (10)$$

Схемная реализация данной ОП соответствует рис. 6, но вместо сигнала x_1 на управляющий вход МХ подается сигнал p_1 . Поскольку данное отличие не влияет на аппаратные затраты в комбинационной схеме для операции O_4 , сравнение рис. 6 и 8 позволяет утверждать, что для рассматриваемого фрагмента ГСА затраты аппаратуры в МПА ОАП со структурой U_4 будут ниже, чем в случае МПА со структурой U_2 .

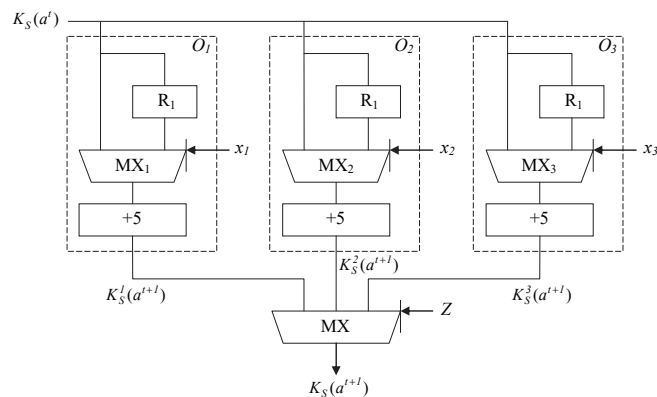


Рис. 8. Фрагмент функциональной схемы ОАП МПА со структурой U_2 , соответствующий операциям переходов (7) – (9)

Отметим, что если в процессе синтеза МПА по ГСА G операция O_2 была сопоставлена только состоянию a_2 и никакому другому, а операция O_3 – только состоянию a_3 , то замена каждой из ОП O_2 и O_3 операцией O_4 приведет к уменьшению общего количества операций переходов в ОАП. Если уменьшение количества ОП сопровождается уменьшением разрядности R_Z кода операции, то в Z -подсхеме и ОАП имеют место такие изменения:

- уменьшается число выходов Z -подсхемы, что приводит к снижению аппаратных затрат в Z -подсхеме.

- уменьшается число входов мультиплексора результата ОАП, что приводит к дополнительному снижению аппаратных затрат в схеме ОАП.

Таким образом, в структуре U_4 в сравнении со структурами U_2 и U_3 возможны как прирост аппаратных затрат из-за M -подсхемы, так и их уменьшение в Z -подсхеме и ОАП. Составив, по аналогии с [7], неравенство, при выполнении которого логическая схема МПА со структурой U_4 будет иметь меньшие аппаратные затраты в сравнении со схемой эквивалентного МПА со структурами U_2 и U_3 :

$$H^{U_4} < \min(H^{U_2}, H^{U_3}), \quad (11)$$

где H^{U_2} , H^{U_3} и H^{U_4} – численно выраженные аппаратные затраты в соответствующих структурах МПА с ОАП, определяемые следующими выражениями:

$$H^{U_2} = H_Z^{U_2} + H_{\text{ОАП}}^{U_2} + H_{\text{СФМО}}^{U_2}, \quad (12)$$

$$H^{U_3} = H_M^{U_3} + H_Z^{U_3} + H_{\text{ОАП}}^{U_3} + H_{\text{СФМО}}^{U_3}, \quad (13)$$

$$H^{U_4} = H_M^{U_4} + H_Z^{U_4} + H_{\text{ОАП}}^{U_4} + H_{\text{СФМО}}^{U_4}. \quad (14)$$

Каждое слагаемое в правой части выражений (12) – (14) – численно выраженные аппаратные затраты в блоках, определяемых нижними индексами, принадлежащих структурам МПА с ОАП, определяемым верхним индексом. При использовании базиса ПЛИС типа *FPGA* единицей измерения аппаратных затрат могут быть *LUT*-элементы, являющиеся регулярными функциональными узлами данных ПЛИС [9, 10].

Заключение. В статье предложена новая структура МПА с ОАП. В ней совмещены два известных подхода к оптимизации аппаратных затрат в логической схеме автомата: сопоставление операций переходов отдельным состояниям и замена входных переменных. В общем виде эффективность предлагаемой структуры по критерию аппаратных затрат в сравнении со структурами-прототипами определяется неравенством (11). При этом условия вы-

полнения данного неравенства не очевидны из структур $U_2 - U_4$ и предполагают проведение отдельных исследований. Практическое использование предложенной структуры требует разработки формализованных методов структурного синтеза данного типа автоматов, ориентированных на использование современного элементного базиса программируемых логических устройств.

1. Глушков В.М. Синтез цифровых автоматов. – М.: Физматгиз, 1962. – 476 с.
2. Баранов С.И. Синтез микропрограммных автоматов. – Л.: Энергия, 1979. – 232 с.
3. Баркалов А.А. Синтез устройств управления на программируемых логических устройствах. – Донецк, ДонНТУ, 2002. – 262 с.
4. Бабаков Р.М., Баркалов А.А. Реализация функции переходов микропрограммного автомата на базе операционного автомата // УСиМ. – 2015. – № 5. – С. 22–29.
5. Бабаков Р.М. Математическая модель микропрограммного автомата с операционным автоматом переходов. – Информатика, кибернетика и вычис-

- лительная техника: Сб. научн. трудов ДонНТУ. – 2016. – 1(22). – С. 54–57.
6. Бабаков Р.М., Ярош И.В. Формирование кодов операций переходов в микропрограммном автомате с операционным автоматом переходов. – Там же. – 2015. – 1(20). – С. 11–16.
 7. Бабаков Р.М. Using of method of replacement of input variables in microprogram finite-state machine with datapath of transitions // Технологический аудит и резервы производства. – 2017. – 36, № 4/2. – С. 18–23.
 8. Бабаков Р.М. Промежуточная алгебра переходов в микропрограммном автомате // Радиотехника, информатика, управление. – 2016. – № 1. – С. 64–73.
 9. Грушвицкий Р.И., Мурсаев А.Х., Узрюмов Е.П. Проектирование систем на микросхемах программируемой логики. – БХВ-Петербург, 2002. – 608 с.
 10. Максфилд К. Проектирование на ПЛИС. Курс молодого бойца. – М.: Издательский дом «Додэка-XXI», 2007. – 408 с.

Поступила 02.11.2017
Тел. для справок: +48 68 328-2693 (Винница, Зеленая Гура, Польша)
E-mail: r.babakov@donnu.edu.ua, cpld@mail.ru, a.barkalov@imei.uz.zgora.pl
© Р.М. Бабаков, А.А. Баркалов, 2017

UDC 004.2

Roman Babakov¹, Alexander Barkalov²

¹ PhD in Techn.Sciences, Associate Professor, Vasyly' Stus Donetsk National University, 600-richa str., 21, Vinnitsya, 21021, Ukraine, r.babakov@donnu.edu.ua

² Doctor in Techn. Sciences, Professor, University of Zielona Gora, Podgorna str., 50, Zielona Gora, 65246, Poland, a.barkalov@imei.uz.zgora.pl

Microprogram Finite State Machine Modification with the Datapath of Transitions and Replacement of the Input Variables

Keywords: microprogram finite state machine, datapath of transitions, replacement of input variables, hardware expenses.

Introduction. The object of research is the microprogram finite state machine with datapath of transitions. In digital devices, microprogram finite state machine performs the functions of control unit and coordinates the work of other units of the system. One of the topical scientific and practical problems is the reduction of hardware expenses in the logical circuit of the microprogram final-state machine. One way to solve this problem is to develop new microprogram final-state machine structures and methods for their synthesis. One of such structures is a microprogram final-state machine with datapath of transitions. In this structure, the transition formation circuit is implemented in the form of an datapath consisting of the separate operational blocks. Each operational block implements its own law of converting the state codes and input signals, realizing a disjoint subset of microprogram transitions. Herewith, the hardware expenses in the operational block do not depend or depend insignificantly on the number of microprogram transitions it implements. This allows under certain conditions to achieve reducing of hardware expenses in comparison with the implementation of transition function of the finite state machine by the canonical method using the system of Boolean equations. The approach, consisting in the representation of the circuit for forming the microprogram final-state machine transitions in the form of an datapath, is called the operational realization of the transition function of the finite state machine.

Purpose. The purpose of this article is to research a new way for hardware expenses optimization in logical circuit of finite state machine with datapath of transitions.

Methods. It is proposed to use in one structure of finite state machine simultaneously two known methods for hardware expenses optimization. The method of comparing the codes of transition operations to the FSM states makes it possible to reduce the complexity of the circuit of the operation codes formation by reducing the number of input signals. The method of replacement of input variables makes it possible to reduce the complexity of the datapath of transitions by reducing the number of functional blocks.

Results. A new modification of the microprogram finite state machine structure with datapath of transitions and replacement of input variables is proposed. The peculiarity of this structure is the lower hardware expenses in the logical circuit of finite state machine in comparison with the prototype structures. The general condition of the effectiveness of this structure by the criterion of hardware expenses is determined.

Conclusion. Reduction of hardware expenses in the logical circuit of a microprogram finite state machine with datapath of transitions is possible due to the joint use of the several known methods of optimization of hardware expenses. For practical application of the researched structure, it is necessary to develop a synthesis method, which is a combination of synthesis methods of prototype structures. To determine the appropriateness of using the proposed structure, it is required to determine the area of its effective using, which is expressed as a set of values and ranges of the parameters of the finite state machine.

1. *Glushkov V.M.* Sintez tsifrovih avtomatov, Moscow, 1962, 476 p.
2. *Baranov S.I.* Sintez mikroprogrammni avtomatov, Leningrad, 1979, 232 p.
3. *Barkalov A.A.* Sintez ustroystv upravleniya na programmiruemih logicheskikh ustroystvah, Donetsk, 2002, 262 p.
4. *Barkalov A.A., Babakov R.M.* Upravljajushie sistemi i mashini, no 5, 2015, P. 22–29.
5. *Babakov R.M.* Sbornik nauchnih trudov DonNTU. Seriya: «Informatika, kibernetika I vichislitelnaya tehnika», n 1 (22), 2016, P. 54–57.
6. *Babakov R.M., Yarosh I.V.* Sbornik nauchnih trudov DonNTU. Seriya: «Informatika, kibernetika I vichislitelnaya tehnika», n 1 (20), 2015, P. 11–16.
7. *Babakov R.M.* Tehnologicheskij audit i rezervi proizvodstva. n. 4/2 (36), 2017, P. 18–23.
8. *Babakov R.M.* Radiotekhnika, informatika, upravlenie. n. 1, 2016, P. 64–73.
9. *Grushvitskiy R.I.* Proektirovanie sistem na mikroshemah programmiruemoj logiki, St. Peterburg, 2002, 608 p.
10. *Maksfild K.* Proektirovanie na PLIS. Kurs molodogo boytsa, Moscow, 2007, 408 p.

Окончание статьи В.М. Куйко

1. https://en.wikipedia.org/wiki/Automatic_number_plate_recognition.
2. *Dhawal Wazalwar, Erdal Oruklu, Jafar Saniee*, A Design Flow for Robust License Plate Localization and Recognition in Complex Scenes, J. of Transportation Technologies, 2012, 2, P. 13–21.
3. *Martinsky O.* Algorithmic And Mathematical Principles Of Automatic Number Plate Recognition Systems, Bachelor's thesis, Department of Intelligent Systems, Faculty of Information Technology, Brno University of Technology, 2007.
4. *Shih-Hao Yu, Jun-Wei Hsieh and Yung-Sheng Chen*, "Morphology-based License Plate Detection from Complex Scenes," IEEE Proc. Of Int. Conf, on Pattern Recognition, Quebec, Canada (August 11–15, 2002), III, P. 176–179, 2002.
5. *Amr Badr, Mohamed M. Abdelwahab, Ahmed M. Thabet, and Ahmed M. Abdelsadek*, Automatic Number Plate Recognition System, Annals of the University of Craiova, Mathematics and Computer Science Series, 38(1), 2011, P. 62–71.
6. *Matas J. and Zimmerman K.*, Unconstrained Licence Plate and Text Localization and Recognition. In: IEEE Int. Conf.on Intell. Transp. Systems, New York, 2005, P. 225–230.
7. *Duda R.O. and P.E. Hart*, Use of the Hough Transformation to Detect Lines and Curves in Pictures, Comm. ACM, 15, 1972, P. 11–15.
8. *Kiryati N., Eldar Y. and Bruckstein A.M.*, A Probabilistic Hough transform, Pattern Recognition, 1991, 24, (4), P. 303–316.
9. *Schlezinger M.I.* Fast implementation of one class of linear convolutions. In: Theoretical and applied questions of image recognition, Kiev: IK AN USSR, 1991, P. 61–69. (In Russian).
10. *Nobuyuki Otsu*. A threshold selection method from gray-level histograms. IEEE Trans. Sys., Man., Cyber. 9 (1), 1979, P. 62–66.
11. *Levenshtejn V.I.* Binary codes with correction of fallouts, insertions and symbol substitutions. Reports AN SSSR, M., 1965, 163 (4), P. 845–848. (In Russian).
12. *Shlezinger M.I., Vodolazskiy E.V., Yakovenko V.M.* Recognition of the similarity of polygons in the strengthened Hausdorff metric. Cybernetics and Systems Analysis, 2014, 3, P. 174–187. (In Russian).
13. *Murygin K.V.* Detection of automobile numbers on the basis of the mixed cascade of classifiers, Artificial Intelligence, 2010, 2, P. 47–152. (In Russian).
14. *Murygin K.V.* Preliminary processing of candidates when car numbers on images are detected, Artificial Intelligence, 2013, 2, P. 32–37. (In Russian).
15. <http://viatec.ua/product/nomerok-4>.
16. <http://isscctv.com.ua/?page=SecurOSAuto>.