

А.А. Баркалов, Я.Е. Визор, А.В. Матвиенко, Л.А. Титаренко

Оптимизация схемы автомата Мура с двумя источниками кодов

Предложен метод уменьшения числа *LUT*-элементов в схеме автомата Мура, основанный на использовании двух источников кодов классов псевдоэквивалентных состояний. Приведен пример применения предложенного метода.

The method for reducing the number of *LUT* elements in the circuit of Moore finite state machine is proposed. The special multiplexer executes the choice of the input memory functions source. The system of the microoperations is implemented by embedded memory blocks. An example for application of the proposed method is given.

Запропоновано метод зменшення числа *LUT*-елементів у схемі автомата Мура, заснований на використанні двох джерел кодів класів псевдоеквівалентних станів. Наведено приклад застосування запропонованого методу.

Введение. При реализации схем устройств управления (УУ) часто используется модель микропрограммного автомата (МПА) Мура [1, 2]. Актуальной задачей, возникающей при синтезе УУ, есть уменьшение числа логических элементов в таких схемах [3, 4]. Решение этой задачи позволяет уменьшить число межблочных соединений в схеме и потребляемую мощность. В настоящее время для реализации схем цифровых систем используются микросхемы *FPGA* (*Field-Programmable Logic Arrays*) [5, 6]. В статье рассматривается метод уменьшения площади кристалла, занимаемой схемой МПА Мура, ориентированный на *FPGA*. При этом алгоритм управления представлен в виде граф-схемы алгоритма ГСА [1].

Как правило *FPGA* включают элементы табличного типа *LUT* (*Look-Up Table*) и встроенные блоки памяти *EMB* (*Embedded Memory Blocks*) [5, 6]. Число входов S логических элементов *LUT* ограничено ($S \leq 6$). Существует возможность связи выхода *LUT* с программируемым триггером. Блоки *EMB* обладают свойством реконфигурации, что позволяет менять число ячеек памяти (V) и их выходов (tF). При этом емкость блока $V0$ остается постоянной. Существуют следующие конфигурации *EMB*: $16K \times 1$, $8K \times 2$, $4K \times 4$, $2K \times 8$, $1K \times 16$, 512×32 [5, 6]. Это означает, что $SA \in \{14, 13, 12, 11, 10, 9\}$

и $tF \in \{1, 2, 4, 8, 16, 32\}$, где SA – число адресных разрядов блока.

При синтезе схем УУ необходимо учитывать как особенности модели МПА, так и элементного базиса. Предлагается использовать такие особенности МПА Мура, как наличие классов псевдоэквивалентных состояний (ПЭС) и регулярность системы микроопераций [1]. Первая особенность позволяет использовать более чем один источник кодов состояний [3, 4]. Вторая – использовать блоки *EMB* для реализации системы микроопераций. Небольшое число входов логических элементов *LUT* требует модификации структур МПА и методов их синтеза в сравнении с известными аналогами [8, 9].

Реализация МПА Мура на *FPGA*

Автомат Мура можно охарактеризовать множествами X , Y , A и двумя функциями – переходов и выходов [1]. Здесь $X = \{x_1, \dots, x_L\}$ – множество логических условий (ЛУ), $Y = \{y_1, \dots, y_N\}$ – множество микроопераций (МО) и $A = \{a_1, \dots, a_M\}$ – множество состояний. Как правило, в МПА выделяется начальное состояние $a_1 \in A$. Функции переходов и выходов представляются прямой структурной таблицей (ПСТ), имеющей столбцы: am – текущее состояние; $K(am)$ – код состояния $am \in A$; as – состояние перехода; $K(as)$ – код состояния $as \in A$; Xh – конъюнкция некоторых элементов

множества X (или их отрицаний), определяющая переход $\langle am, as \rangle$; Φh – набор функций возбуждения памяти МПА, принимающих единичное значение для переключения памяти из $K(am)$ в $K(as)$; $h = 1, \dots, H$; (Γ) – номер строки таблицы. В столбце am записывается набор микроопераций $Y(am) \subseteq Y$, формируемых в состоянии $as \in A$. Для кодирования состояний $as \in A$ используется множество внутренних переменных $T = \{T1, \dots, TR\}$, где $R = \lceil \log_2 M \rceil$. Функции возбуждения памяти МПА образуют множество $\Phi = \{D1, \dots, DR\}$, т.е. при синтезе используются триггеры типа D [10].

Логическая схема МПА задается системой уравнений

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

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

Системы (1) – (2) формируются на основе прямой структурной таблицы по правилам [1]. Системы (1) – (2) определяют модель PY автомата Мура (рис. 1).

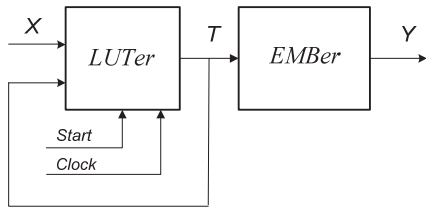


Рис. 1. Структурная схема PY автомата Мура

В этой схеме блок $LUTer$ состоит из LUT -элементов, реализующих (1). В состав блока входят R триггеров, обнуляемых по сигналу $Start$. Изменение кода $K(am)$ происходит по сигналу синхронизации $Clock$. Выходы блока $LUTer$ представляют собой внутренние переменные $Tr \in T$. Блок $EMBer$ состоит из встроенных блоков EMB , реализующих систему (2).

Постановка задачи

Как правило, число переходов $H1(\Gamma)$ больше числа переходов $H0(\Gamma)$ эквивалентного автомата Мили [1]. Это приводит к росту аппаратных затрат в схеме МПА Мура, в сравнении с этим показателем для эквивалентного автомата Мили. Параметр $H1(\Gamma)$ можно уменьшить благодаря наличию псевдоэквивалентных состояний (ПЭС) МПА Мура [7]. Состояния $am, as \in A$ называются ПЭС, если выходы соответ-

ствующих им вершин соединены с входом одной и той же вершины ГСА Γ . Пусть $PA = \{B1, \dots, BI\}$ – разбиение множества A на классы ПЭС ($I \leq M$). Построим систему функций

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

где булева переменная C_{mi} равна единице если и только если $am \in Bi$, A_m – конъюнкция внутренних переменных $Tr \in T$, соответствующая коду $K(am)$ состояния $am \in A$. Закодируем состояния $am \in A$ так, чтобы любая функция системы (3) представлялась одним конъюнктивным термом. Назовем такое кодирование оптимальным кодированием состояний.

Такой подход ведет к модели $P0Y$ автомата, структура которой совпадает со структурой PY автомата, но число термов совпадает с $H0(\Gamma)$. Однако такое кодирование не всегда возможно [7] из-за особенностей ГСА.

Для уменьшения числа строк ПСТ можно использовать преобразование кодов состояний $am \in A$ в коды классов псевдоэквивалентных состояний $K(Bi)$. Поставим в соответствие классу $Bi \in PA$ двоичный код $K(Bi)$ разрядности $R_b = \lceil \log_2 I \rceil$ и используем переменные $\tau \in \tau$ для такого кодирования, где $\tau = RB$. В этом случае для представления МПА используется модель PCY (рис. 2).

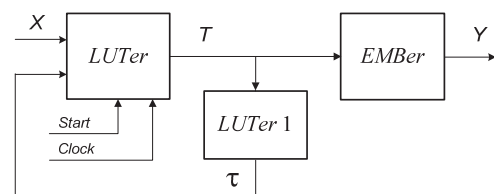


Рис. 2. Структурная схема PCY автомата Мура

Здесь блок $LUTer$ реализует систему функций

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

Блок $LUTer1$ – преобразователь кодов $K(am)$ в коды классов $K(Bi)$. $LUTer1$ реализует систему функций

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

В работе [7] показано, что $H3(\Gamma) = H0(\Gamma)$. Однако блок $LUTer1$ потребляет некоторые ресурсы $FPGA$. Предлагаемый метод, позволя-

ет сохранить положительные качества *PCY* автомата и удалить блок *LUTer1*.

Основная идея предлагаемого метода

Закодируем состояния $am \in A$ оптимальным образом. Пусть $T(Bi)$ – число термов в функции $Bi \in PA$. Представим множество PA в виде объединения множеств PB и PC . При этом распределение классов выполняется следующим образом:

$$\begin{aligned} (T(Bi) = 1) &\rightarrow Bi \in PB; \\ (T(Bi) > 1) &\rightarrow Bi \in PC. \end{aligned} \quad (6)$$

Очевидно, что преобразованию подлежат только коды состояний $am \in Bi$ для блоков $Bi \in PC$. Поставим в соответствие каждому классу $Bi \in PC$ двоичный код $K(Bi)$ разрядности

$$R_C = \lceil \log_2(IC) \rceil, \quad (7)$$

где $IC = PC$.

Пусть следующее условие выполняется для блоков *EMBer*:

$$\left\lceil \frac{N}{t_F} \right\rceil = \left\lceil \frac{N + R_C + 1}{t_F} \right\rceil. \quad (8)$$

При этом функции (5) могут быть реализованы блоком *EMBer*, что устраняет необходимость в блоке *LUTer1*.

Представим множество Φ в виде объединения множеств Φ_1 и Φ_2 , где $\Phi_1 \cap \Phi_2 \neq 0$. Этим множествам соответствуют функции:

$$\Phi_1 = \Phi_1(T', X), \quad (9)$$

$$\Phi_2 = \Phi_2(\tau, X). \quad (10)$$

В (9) $T' \subseteq T$, что уменьшает число литералов в системе (9) в сравнении с (1).

На рис. 3 представлена структурная схема предлагаемого *P2Y* автомата Мура.

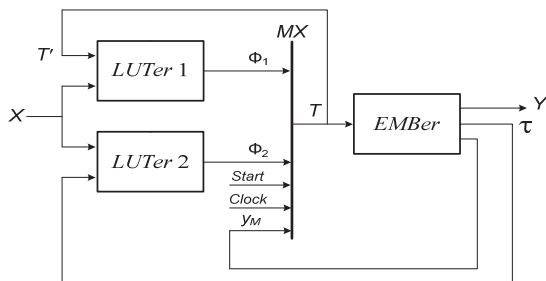


Рис.3. Структурная схема *P2Y* автомата Мура

В этой схеме блок *LUTer1* реализует систему (9), а блок *LUTer2* – систему (10). Блок *EMBer* реализует системы функций (2) и (5). Блок мультиплексора *MX* реализуется на *LUT*. Блок *MX* служит для выбора функций Φ_1 и Φ_2 в качестве кода состояния перехода. При этом выбор определяется переменной *yM*:

$$T = yM \Phi_1 \vee \bar{yM} \Phi_2. \quad (11)$$

Наличие единицы в (8) объясняется необходимостью резервирования выхода *EMBer* для переменной *yM*.

Далее приведен метод синтеза предложенного *P2Y* автомата Мура по исходной ГСА Г.

Предлагаемый метод синтеза

Данный метод синтеза включает следующие этапы:

- формирование множества состояний $A = \{a_1, \dots, a_M\}$;
- формирование разбиения $PA = \{B_1, \dots, B_I\}$ множества A на классы псевдоэквивалентных состояний;
- оптимальное кодирование состояний $am \in A$;
- формирование множеств PB и PC ;
- кодирование классов $Bi \in PC$;
- перекодирование состояний $am \in A$;
- формирование таблицы блока *LUTer1*;
- формирование таблицы блока *LUTer2*;
- формирование системы функций для блока *MX*;
- формирование таблицы блока *EMBer*;
- реализация схемы МПА в заданном элементном базисе.

Рассмотрим пример применения предложенного метода. В целях экономии используем задание автомата не в виде ГСА, а в виде системы обобщенных формул перехода (ОФП) [6, 7]. Пусть автомат Мура *S1* определяется следующей системой ОФП:

$$\begin{aligned} B_1 &\rightarrow x_1 a_2 \vee \bar{x}_1 a_3; \\ B_2 &\rightarrow x_2 a_4 \vee \bar{x}_2 x_3 a_5 \vee \bar{x}_2 \bar{x}_3 a_6; \\ B_3 &\rightarrow x_3 a_6 \vee \bar{x}_3 x_4 a_8 \vee \bar{x}_3 \bar{x}_4 a_1; \\ B_4 &\rightarrow x_5 a_8 \vee \bar{x}_5 x_6 a_{11} \vee \bar{x}_5 \bar{x}_6 a_{13}; \end{aligned}$$

$$B_5 = a_{10}; \quad B_6 \rightarrow x_1 a_{12} \vee \overline{x_1} a_1;$$

$$B_7 \rightarrow x_4 a_1 \vee \overline{x_4} x_5 a_7 \vee \overline{x_4} x_5 x_6 a_{11} \vee \overline{x_4} x_5 x_6 a_{13}$$

Пусть при этом получено разбиение $PA = \{B1, \dots, B7\}$, где $B1 = \{a1\}$, $B2 = \{a2, a3\}$, $B3 = \{a4\}$, $B4 = \{a5, a6, a7\}$, $B5 = \{a8, a9\}$, $B6 = \{a10\}$, $B7 = \{a11, a12, a13\}$. Пусть система микроопераций $S1$ включает $N=13$ элементов. Очевидно, что для $S1$ имеем следующие множества и параметры: $A = \{a1, \dots, a13\}$, $M = 13$, $R=4$, $\Phi = \{D1, \dots, D4\}$, $T = \{T1, \dots, T4\}$, $I=7$. Сформируем систему (3), которая в случае автомата $S1$ имеет следующий вид:

$$B_1 = a_1; \quad B_2 = a_2 \vee a_3; \quad B_3 = a_4;$$

$$B_4 = a_5 \vee a_6 \vee a_7; \quad B_5 = a_8 \vee a_9;$$

$$B_6 = a_{10}; \quad B_7 = a_{11} \vee a_{12} \vee a_{13}.$$

Один из возможных вариантов оптимального кодирования состояний для автомата $S1$ показан картой Карно на рис. 4.

		$T_3 T_4$			
		00	01	11	10
$T_1 T_2$	00	a_1	a_2	\times	a_4
	01	a_5	\times	a_6	a_{10}
	11	a_{11}	a_3	a_7	a_8
	10	a_{13}	a_{12}	\times	a_9

Рис.4. Коды состояний автомата Мура $S1$

Анализ этой карты показывает, что классы $B1, B3, B5, B6 \in PB$ и $B2, B4, B7 \in PC$. Таким образом, $IC=3, RC=2, \tau = \{\tau_1, \tau_2\}$. Закодируем классы $Bi \in PC$ следующим образом: $K(B2)=00, K(B4)=01, K(B7)=10$. Из карты Карно имеем $K(B1)=0000, K(B3)=001 \times, K(B5)=1 \times 10, K(B6)=0110$.

Пусть для реализации схем используется микросхема $FPGA$, в состав которой входят EMB с конфигурацией 16×16 . В этом случае условие (8) выполняется и модель $P2Y$ автомата $S1$ может быть использована.

Поясним необходимость этапа 6. Из формулы (11) следует, что при $yM=1$ выходы блока $LUTer2$ игнорируются. Поэтому коды состояний $am \in Bi$, где $Bi \in PC$, можно рассматривать как несущественные наборы входных переменных. Коды $K(Bi)$ классов $Bi \in PC$ формируются блоком $EMBer$. При этом коды соот-

ветствующих состояний могут быть произвольными, а не такими, как на рис. 4. С учетом этих рассуждений перекодируем состояния $am \in A$ (рис. 5). Если в какой-либо клетке карты Карно (рис. 5) находится символ состояния и символ « \times », то это значит, что данный набор игнорируется при $yM=1$.

		$T_3 T_4$			
		00	01	11	10
$T_1 T_2$	00	a_1	a_8	a_9	a_4
	01	$a_2 \times$	$a_6 \times$	$a_{11} \times$	$a_{12} \times$
	11	$a_5 \times$	\times	\times	\times
	10	$a_3 \times$	$a_7 \times$	$a_{13} \times$	a_{10}

Рис.5. Коды состояний после перекодирования

Из рис. 5 можно получить следующие коды: $K(B1) = \times \times 10, K(B3) = 0 \times 10, K(B5) = \times \times \times 1, K(B6) = 1 \times \times \times$. Как следует из этих кодов, $T' = \{T1, T3, T4\}$.

Таблицы блоков $LUTer1$ и $LUTer2$ включают следующие столбцы: $Bi, K(Bi), aS, K(aS), Xh, \Phi_h^i, h$. В рассматриваемом примере таблица блока $LUTer1$ имеет $H1(S1) = 8$ строк (табл. 1). Эта таблица строится на базе системы ОФП (12) и кодов из рис. 5

Таблица 1. Таблица блока $LUTer1 P2Y$ автомата $S1$

Bi	$K(Bi)$ $T1T3T4$	aS	$K(aS)$	Xh	Φ_h^1	h
		$B1$	$\times 00$	$a2$	0100	$x1$
$a3$	1000			$x1$	D_1^1	2
$B3$	010	$a6$	0101	$x3$	$D_2^1 D_4^1$	3
		$a8$	0001	$x3 x4$	D_4^1	4
		$a1$	0000	$x3 x4$		5
$B5$	$\times \times 1$	a_{10}	1010	1	$D_1^1 D_3^1$	6
$B6$	$1 \times \times$	a_{12}	0110	$x1$	$D_2^1 D_3^1$	7
		a_1	0000	$x1$		8

Из табл. 1 следует, что $\Phi_1 = \{D_1^1, \dots, D_4^1\}$. И тогда можно получить систему (9). Например, с учетом минимизации можно получить следующее уравнение: $D_4^1 = \overline{T_1} T_3 \overline{T_4} x_3 \vee \overline{T_1} T_3 \overline{T_4} x_4$. Эта функция может быть реализована на одном элементе LUT , имеющем $S=5$ входов. При этом любая функция D_r^1 ($r=1, 4$) из табл. 1 реализуется на одном LUT с $S=5$. Таким обра-

зом, для реализации системы Φ_1 требуется четыре элемента LUT с $S=5$. Можно показать, что при использовании кодов состояний из рис. 4 для реализации системы Φ_1 требуется 10 элементов LUT с $S=5$. Кроме того, эта схема – двухуровневая. Таким образом, для данного примера перекодирование состояний позволяет в два с половиной раза уменьшить аппаратные затраты и в два раза увеличить быстродействие для $LUTer1$. Естественно, это верно только для рассматриваемого примера и при $S=5$.

Таблица блока $LUTer2$ для $P2Y$ автомата $S1$ имеет 10 строк. Ее фрагмент показан в табл. 2.

Таблица 2. Фрагмент таблицы блока $LUTer2$ $P2Y$ автомата $S1$

Bi	$K(Bi)$		aS	$K(aS)$	Xh	Φ_h^2	h
	τ_1	τ_2					
$B2$	00		$a4$	0010	$x2$	D_3^2	1
			$a5$	1100	$x2\ x3$	$D_1^2\ D_2^2$	2
			$a6$	0101	$x2\ x3$	$D_2^2\ D_4^2$	3
$B4$	01		$a8$	0001	$x5$	D_4^2	4
			$a11$	0111	$x5\ x6$	$D_2^2\ D_3^2\ D_4^2$	5
			$a13$	1011	$x5\ x6$	$D_1^2\ D_3^2\ D_4^2$	6

Из этой таблицы можно получить систему функций (10), а также получить формулу $D_4^2 = \tau_1 \tau_2 x_2 x_3 \vee \tau_2$. Отметим, что $\Phi_2 = \{D_1^2, \dots, D_4^2\}$, коды состояний $am \in A$ взяты из карты Карно (рис. 5). Код 11 использован для оптимизации кодов $K(Bi)$, где $Bi \in PC$. С учетом кода 11 имеем $K(B4) = \times 1$ и $K(B7) = 1 \times$.

Формирование содержимого $EMBer$ сводится к формированию таблицы со столбцами am , $K(am)$, $Y(am)$, τm , yM , m . Здесь $Y(am) \subseteq Y$ – набор микроопераций, формируемый в состоянии $am \in A$. Эта информация содержится в операторных вершинах ГСА Г. Столбец τm содержит переменные $\tau r \in \tau$. В столбце yM записывается единица, если $am \in Bi$ и $Bi \in PB$. Эта таблица строится элементарным образом, но мы ее не рассматриваем в статье.

Система уравнений блока MX содержит R уравнений, имеющих следующий вид:

$$D_r = y_M D_r^1 \vee \overline{y_M} D_r^2 \quad (\overline{r=1, R}). \quad (12)$$

В рассматриваемом примере $R=4$. Следовательно, для реализации блока MX достаточно четырех элементов LUT , имеющих три входа.

Последний этап предлагаемого метода сводится к использованию стандартных пакетов [6] и не рассматривается в статье.

Заключение. Особенности технологии $FPGA$ требуют адаптации к ним известных методов синтеза МПА. Для уменьшения числа LUT -элементов в схеме нужно использовать характерные черты как модели МПА, так и элементного базиса. Это уменьшает как число LUT -элементов, так и число межсоединений в схеме автомата.

В статье предлагается метод снижения аппаратных затрат в схеме МПА Мура. Метод основан на использовании двух источников кодов классов ПЭС. Такой подход позволяет уменьшить число термов в системе функций возбуждения памяти. Часть термов включает в себя конъюнкции кодов состояний, имеющих меньше R переменных, где R – разрядность кодов состояний. Это ведет к дополнительному уменьшению числа LUT -элементов в схеме автомата. При выполнении условия (8) система микроопераций реализуется на одном блоке EMB .

Предложенный метод перекодирования состояний позволяет уменьшить число переменных обратной связи в блоке $LUTer1$. Это приводит как к уменьшению числа межсоединений, так и к уменьшению числа литералов в функциях возбуждения памяти.

Недостаток предложенного метода – наличие мультиплексора MX . Этот блок вносит дополнительную задержку к времени распространения сигнала, в сравнении с PY -автоматом. Однако уменьшение числа уровней в схеме формирования функций Φ_1 и Φ_2 (в сравнении с Φ) может компенсировать этот недостаток.

Для сравнения различных методов синтеза МПА используется библиотека стандартных ГСА [10]. Анализ этой библиотеки показал, что условие (8) выполняется для 95 процентов автоматов, т.е. для 95 процентов стандартных примеров система микроопераций реализуется на одном блоке встроенной памяти. Дальнейшее

направление наших исследований заключается в разработке САПР для синтеза *P2Y*-автоматов.

Научная новизна предложенного метода заключается в учете особенностей автомата Мура и элементного базиса *FPGA* для уменьшения аппаратных затрат в схеме автомата.

Практическая значимость заключается в уменьшении стоимости схемы автомата Мура на *FPGA* в сравнении с известными из литературы аналогами.

1. *Baranov S.* Logic Synthesis for Control Automata. – Dordrecht: Kluwer Acad. Publ. – 1994. – 312 p.
2. *DeMicheli G.* Synthesis and Optimization of Digital Circuits. – New York: McGraw-Hill, 1994. – 636 p.
3. *Соловьев В.В.* Проектирование цифровых схем на основе программируемых логических интегральных схем. – М.: Горячая линия – ТЕЛЕКОМ, 2001. – 636 с.
4. *Skliarova I., Sklyarov V., Sudnitson A.* Design of FPGA-based circuits using Hierarchical Finite State Machines. – Tallinn: TUT Press, 2012. – 240 p.
5. *Грушницкий Р.И., Мурсаев А.Х., Угрюмов Е.П.* Проектирование систем с использованием микросхем

программируемой логики. – СПб.: БХВ. – Петербург, 2002. – 608 с.

6. *Synthesis and Optimization of FPGA-based Systems / V. Sklyarov, I. Skliarova, A. Barkalov et al.* – Berlin: Springer, 2014. – 432 p.
7. *Баркалов А.А.* Принципы оптимизации логической схемы микропрограммного автомата Мура // Кибернетика и системный анализ. – 1998, №1. – С. 65–72.
8. *Баркалов А.А., Цололо С.А.* Оптимизация схемы автомата Мура в составе системы на кристалле // Радиоэлектроника и информатика. – 2007. – № 1. – С. 35–39.
9. *Баркалов А.А., Цололо С.А.* Оптимизация числа макроячеек *PAL* в схеме автомата Мура // УСиМ. – 2008. – № 2. – С. 54–59.
10. *Yang S.* Logic Synthesis and optimization benchmarks user guide. Version 3.0 Techn. Report. – Microelectronics Center of North Carolina. – 1991, Jan. 15. – 43 p.

Поступила 03.02.2015

Тел. для справок: +38 044 526-2504, 526-3069 (Киев)

E-mail: A.Barkalov@iie.uz.zgora.pl, yaviz@ukr.net,

matv@online.ua

©А.А. Баркалов, Я.Е. Визор, А.В. Матвиенко,

Л.А. Титаренко, 2015

Внимание!

Доступен сайт журнала: usim.irtc.org.ua, на котором размещен архив журнала с 2009 года.

На сайте Национальной библиотеки Украины имени В.И. Вернадского в рубрике «Наукова періодика України» также доступен архив журнала с 2009 года. Все научные издания, представленные на этом ресурсе на новой платформе, будут корректно индексироваться поисковой системой *Google Scholar*.

Журнал представлен в научно-метрической базе (<http://www.elibrary.ru>). Научная электронная библиотека содержит Российский индекс научного цитирования (РИНЦ), электронные научные публикации, информационные базы данных, а также сервис индивидуальной подписки на электронные версии научных изданий.