

# ПРОГРЕСИВНІ ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ

## ПРОГРЕССИВНЫЕ ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

### PROGRESSIVE INFORMATION TECHNOLOGIES

УДК 681.324

Бабаков Р. М.

Канд. техн. наук, доцент, доцент кафедры компьютерных наук Донецкого национального технического университета,  
г. Красноармейск, Украина

#### ПРОМЕЖУТОЧНАЯ АЛГЕБРА ПЕРЕХОДОВ В МИКРОПРОГРАММНОМ АВТОМАТЕ

Решена задача формализации задания микропрограммного автомата, в структуре которого часть автоматных переходов реализуется неканоническим способом. Предложен новый подход к организации функции переходов микропрограммного автомата, в соответствии с которым функция переходов представляется в виде семейства частичных функций, каждая из которых определена лишь на части области определения функции переходов автомата и соответствует некоторому подмножеству автоматных переходов.

С учетом предложенного подхода традиционное представление автомата в виде многоосновной алгебры претерпевает ряд изменений. Во-первых, взаимная независимость функций переходов и выходов, образующих сигнатуру алгебры, позволяет рассматривать их отдельно друг от друга, что приводит к представлению автомата в виде двух алгебр: алгебры переходов, сигнатура которой содержит только функцию переходов, и алгебры выходов, сигнатура которой содержит только функцию выходов. Во-вторых, представление функции переходов в виде множества частичных функций приводит к замене алгебры переходов множеством подалгебр переходов, в каждой из которых сигнатура образована частичной функцией переходов.

На примере микропрограммного автомата со счетчиком показано, что закон преобразования кодов состояний в рамках некоторого подмножества переходов может быть задан некоторой алгебраической функцией (операцией переходов), использующей скалярную интерпретацию кодов состояний структурного автомата. Скалярную интерпретацию кодов состояний совместно с операцией переходов предлагается представлять в виде т.н. промежуточной алгебры переходов, изоморфной соответствующим подалгебрам переходов абстрактного и эквивалентного ему структурного автоматов.

**Ключевые слова:** микропрограммный автомат, частичная функция переходов, промежуточная алгебра переходов, автомат на счетчике.

#### НОМЕНКЛАТУРА

МПА – микропрограммный автомат;  
 $S_A$  – абстрактный автомат;  
 $A$  – множество состояний абстрактного автомата;  
 $Z$  – входной алфавит абстрактного автомата;  
 $W$  – выходной алфавит абстрактного автомата;  
 $G_A$  – абстрактная алгебра, задающая абстрактный автомат;  
 $\mathcal{A}_A$  – носитель абстрактной алгебры;  
 $\mathcal{F}_A$  – сигнатура абстрактной алгебры;  
 $\delta$  – функция переходов абстрактного автомата;  
 $\lambda$  – функция выходов абстрактного автомата;  
 $G_\delta$  – абстрактная алгебра переходов;  
 $\mathcal{A}_\delta$  – носитель абстрактной алгебры переходов;  
 $\mathcal{F}_\delta$  – сигнатура абстрактной алгебры переходов;

$G_\lambda$  – абстрактная алгебра выходов;  
 $\mathcal{A}_\lambda$  – носитель абстрактной алгебры выходов;  
 $\mathcal{F}_\lambda$  – сигнатура абстрактной алгебры выходов;  
 $a_n$  – начальное состояние автомата;  
 $D_\delta$  – область определения функции переходов абстрактного автомата;  
 $\delta_i$  – частичная функция переходов абстрактного автомата;  
 $N_\delta$  – количество частичных функций переходов  $\delta_i$ ;  
 $B$  – количество переходов автомата;  
 $a_i$  – текущее состояние абстрактного автомата;  
 $z_j$  – входной сигнал абстрактного автомата;  
 $a_k$  – состояние перехода абстрактного автомата;  
 $D_{\delta_i}$  – область определения функции  $\delta_i$ ;  
 $G_{\delta_i}$  – частичная абстрактная подалгебра переходов;

© Бабаков Р. М., 2016

DOI 10.15588/1607-3274-2016-1-8

$\mathcal{A}_{\delta_i}$  – носитель частичной абстрактной подалгебры переходов  $G_{\delta_i}$ ;

$\mathcal{F}_{\delta_i}$  – сигнатура частичной абстрактной подалгебры переходов  $G_{\delta_i}$ ;

$A_{\delta_i}$  – подмножество состояний, встречающихся в кортежах функции  $\delta_i$ ;

$Z_{\delta_i}$  – подмножество входных сигналов, встречающихся в кортежах функции  $\delta_i$ ;

$S_S$  – структурный автомат;

$G_S$  – структурная алгебра переходов;

$\mathcal{A}_S$  – носитель структурной алгебры переходов;

$\mathcal{F}_S$  – сигнатура структурной алгебры переходов;

$d$  – структурная функция переходов;

$l$  – структурная функция выходов;

$X$  – множество структурных входных сигналов;

$Y$  – множество структурных выходных сигналов;

$G_d$  – структурная алгебра переходов;

$\mathcal{A}_d$  – носитель структурной алгебры переходов;

$\mathcal{F}_d$  – сигнатура структурной алгебры переходов;

$G_l$  – структурная алгебра выходов;

$\mathcal{A}_l$  – носитель структурной алгебры выходов;

$\mathcal{F}_l$  – сигнатура структурной алгебры выходов;

$D_d$  – область определения структурной функции переходов;

$d_i$  – частичная структурная функция переходов;

$N_d$  – количество частичных структурных функций переходов;

$D_{d_i}$  – область определения функции  $d_i$ ;

$F$  – мощность множества  $Z$ ;

$L$  – мощность множества  $X$ ;

$G$  – мощность множества  $W$ ;

$N$  – мощность множества  $Y$ ;

$K_S(a)$  – структурный код состояния автомата;

$K_S(z)$  – структурный код входного сигнала;

$T$  – структурный код текущего состояния автомата;

$R$  – разрядность кода  $T$ ;

$КС_{d_i}$  – комбинационная схема, реализующая структурную функцию переходов  $d_i$ ;

$MX$  – мультиплексор;

$КС_{Inc}$  – схема управления мультиплексором  $MX$ ;

$РП$  – регистр памяти;

$СФМО$  – схема формирования микроопераций;

$СТ$  – инкрементный счетчик состояний;

$K_I(a_i)$  – промежуточный код состояния  $a_i$ ;

$G_I$  – промежуточная алгебра переходов;

$\mathcal{A}_I$  – носитель промежуточной алгебры переходов;

$\mathcal{F}_I$  – сигнатура промежуточной алгебры переходов;

$O_{Inc}$  – операция инкремента;

$K_I(A_{\delta_i})$  – множество кодов  $K_I(a_i)$ , используемых в функции  $\delta_i$ .

## ВВЕДЕНИЕ

Важным компонентом современных вычислительных систем является устройство управления, одним из способов реализации которого является микропрограммный автомат [1]. Наблюдаемый сегодня рост сложности алгоритмов, интерпретируемых МПА, способствует увеличению аппаратных затрат в логической схеме автомата. В данном аспекте актуальной является задача снижения аппаратных затрат в схеме МПА, одно из решений которой состоит в разработке новых структур МПА и их формальных математических моделей. Классические математические модели абстрактного и структурного автоматов подходят для описания предложенной В. М. Глушковым канонической структуры МПА, однако при использовании иных структур данные модели в некоторых случаях не отражают внутренние информационные процессы, ограничивая возможности проектировщика цифровых устройств в вопросах оптимизации логической схемы синтезируемого устройства.

Объектом исследования в работе выступает микропрограммный автомат, предметом исследования – математическая модель МПА, предполагающая специальное представление функции переходов автомата. Цель работы состоит в разработке и формализации такой модели и достигается путем анализа и обобщения свойств одной из известных структур МПА – автомата на счетчике.

## 1 ПОСТАНОВКА ЗАДАЧИ

Основной научной задачей, решаемой в данной работе, является разработка формальных математических моделей абстрактного и структурного автоматов, использующих представление функции переходов автомата в виде множества частичных функций. Модели должны обеспечивать выражение закономерностей преобразования информации в части функции переходов структурного автомата, подчеркивая эквивалентность данной функции с функцией переходов тождественного абстрактного автомата. Подтверждение состоятельности предлагаемых моделей должно быть продемонстрировано на примере одной из известных структур МПА.

## 2 ОБЗОР ЛИТЕРАТУРЫ

Современная теория автоматов имеет тесную связь с теорией универсальных алгебр. Алгебраический подход находит применение в теории автоматов, способствуя установлению связи теории автоматов с другими областями математики. Математический аппарат универсальных алгебр лежит в основе алгебраической теории автоматов, изучающей дискретные автоматы с абстрактно-алгебраической точки зрения [2, 3].

Базовым объектом изучения в теории универсальных алгебр является алгебраическая система, определяемая как абстрактный объект

$$G = \langle \mathcal{A}, \mathcal{F}, \mathcal{P} \rangle, \quad (1)$$

состоящий из трех множеств: непустого множества  $\mathcal{A}$ , множества операций  $\mathcal{F} = \{\mathcal{F}_0, \dots, \mathcal{F}_\xi, \dots\}$  определенных на  $\mathcal{A}$ , и множества предикатов (отношений)  $\mathcal{P} = \{\mathcal{P}_0, \dots, \mathcal{P}_\eta, \dots\}$  заданных на  $\mathcal{A}$  [4]. Под операцией

здесь понимается функциональное отношение любой арности, заданное на множестве  $\mathcal{A}$ . Под предикатом понимается функция на множестве  $\mathcal{A}$ , значения которой, называемые модальностями или степенями истинности, лежат в особом множестве, состоящем из двух элементов – «истина» и «ложь» [4, 5].

Множество  $\mathcal{A}$  называют носителем системы  $G$ . Набор множеств  $\langle \mathcal{F}, \mathcal{P} \rangle$  образует сигнатуру или структуру алгебраической системы. В случае конечности носителя данные множества также конечны. Сигнатуру с пустым множеством отношений называют функциональной, с пустым множеством операций – предикатной [6].

Алгебраическая система (1) называется *универсальной алгеброй* или просто *алгеброй*, если ее сигнатура функциональна, и *моделью* (реляционной системой), если ее сигнатура предикатна [4]. Таким образом, алгеброй является следующий двухэлементный объект [4, 5]:

$$G = \langle \mathcal{A}, \mathcal{F} \rangle. \quad (2)$$

Иногда носитель алгебры представляется несколькими совокупностями разнородных объектов, объединение которых в единое множество является неестественным [7]. В этом случае оказывается удобным представлять носитель в виде семейства непересекающихся множеств. Пусть  $\mathcal{A} = \{\mathcal{A}_1, \dots, \mathcal{A}_k\}$  – множество основ (носителей),  $\mathcal{F} = \{f_1, \dots, f_m\}$  – сигнатура, причем  $f_i: \mathcal{A}_1^1 \times \mathcal{A}_1^2 \times \dots \times \mathcal{A}_1^{k_i} \rightarrow \mathcal{A}_j$ . Тогда система (2) называется *многоосновой* ( $k$ -основой) алгеброй [5, 7]. Многоосновная алгебра имеет несколько носителей, а каждая операция сигнатуры является соответствием, действующим из прямого произведения некоторых носителей в некоторый носитель [8].

В теории автоматов абстрактный и структурный автоматы иногда рассматриваются как алгебры, поскольку их структура как математических объектов подобна структуре алгебраической системы [9]. Представление автомата в виде алгебры дает возможность использовать алгебраические приемы при решении таких задач, как декомпозиция автоматов, их классификация, описание поведения и другие [3].

### 3 МАТЕРИАЛЫ И МЕТОДЫ

Дадим собственную алгебраическую интерпретацию конечных автоматов.

#### Абстрактный автомат.

Как известно, конечный абстрактный автомат  $S_A$  представляется совокупностью трех конечных множеств – состояний, входных и выходных слов, и двух детерминированных функций – переходов и выходов:  $S_A = \langle A, Z, W, \delta, \lambda \rangle$  [10]. В таком виде автомат может рассматриваться как трехосновная алгебра с двумя операциями [7]. Будем называть алгебру, отождествляемую с абстрактным автоматом, *абстрактной алгеброй*  $G_A$ :

$$G_A = \langle \mathcal{A}_A, \mathcal{F}_A \rangle, \quad (3)$$

$$\mathcal{A}_A = \{A, Z, W\}, \quad (4)$$

$$\mathcal{F}_A = \{\delta, \lambda\}. \quad (5)$$

Фактически, выражения (3)–(5) перечисляют объекты, которые должны быть определены для задания абстрактного автомата. При этом способ задания объектов не имеет значения и может быть различным [2].

Функции  $\delta$  и  $\lambda$  реализуют различные соответствия и могут рассматриваться отдельно друг от друга. Это оказывается полезным, например, в тех случаях, когда выходные реакции автомата неизвестны или не представляют интереса, а объектом исследования является доступная наблюдению эволюция его внутренних состояний под воздействием входных сигналов. В работах [3, 7, 11] упоминаются т.н. *автоматы без выходов* (полуавтоматы, X-автоматы, редукты) – объекты, задаваемые только с помощью множества состояний, входного алфавита и функции переходов.

Абстрактный автомат без функции выходов есть двухосновная алгебра:

$$G_\delta = \langle \mathcal{A}_\delta, \mathcal{F}_\delta \rangle = \langle \{A, Z\}, \{\delta\} \rangle, \quad (6)$$

без функции переходов – трехосновная алгебра

$$G_\lambda = \langle \mathcal{A}_\lambda, \mathcal{F}_\lambda \rangle = \langle \{A, Z, W\}, \{\lambda\} \rangle. \quad (7)$$

Выражение (7) справедливо для автомата Мили. Для автомата Мура необходимо исключить компонент  $Z$ :

$$G_\lambda = \langle \mathcal{A}_\lambda, \mathcal{F}_\lambda \rangle = \langle \{A, W\}, \{\lambda\} \rangle. \quad (8)$$

Будем называть алгебру (6) *абстрактной алгеброй переходов*, алгебры (7) и (8) – *абстрактной алгеброй выходов*. Тогда абстрактный автомат есть двойка

$$S_A = \langle G_\delta, G_\lambda \rangle. \quad (9)$$

В случае выделения на множестве состояний специального начального состояния (инициальный автомат) в выражение (9) может быть добавлен компонент  $a_n \in A$ :

$$S_A = \langle G_\delta, G_\lambda, a_n \rangle. \quad (10)$$

Как известно, в абстрактном автомате функция  $d$  есть соответствие

$$\delta: (D_\delta \subseteq A \times Z) \rightarrow A. \quad (11)$$

Будучи представленной в виде (11), функция  $\delta$  считается, в общем случае, частично определенной на множестве  $A \times Z$ . Исходя из математического определения соответствия, даваемого в [12], функция (11) есть множество кортежей

$$\langle a_i, z_j, a_k \rangle \in \delta, \quad (12)$$

каждый из которых соответствует одному автоматному переходу из состояния  $a_i \in A$  под воздействием входного сигнала  $z_j \in Z$  в состояние  $a_k \in A$ .

Разобьем некоторым образом множество кортежей вида (12) на несколько непустых, попарно непересекающихся подмножеств, максимально возможное число которых равно числу переходов автомата. Каждое такое подмножество переходов автомата, будем называть *частичной функцией переходов*  $\delta_i$ , понимая здесь под термином «частичная» ее неполное определение на области определения  $D_\delta$  функции переходов  $\delta$ :

$$\delta_i: (D_{\delta_i} \subseteq D_\delta) \rightarrow A. \quad (13)$$

В этом случае функция переходов  $\delta$  есть множество подмножеств кортежей (12), то есть множество частичных функций переходов:

$$\delta = \{\delta_1, \dots, \delta_{N_\delta}\}, \quad (14)$$

где  $1 \leq N_\delta \leq B$ . При этом  $\bigcup_{i=1}^{N_\delta} \delta_i = \delta$ .

Подобное разбиение может быть выполнено и для функции выходов.

Свяжем с каждой частичной функцией переходов  $d_i$  собственную алгебру вида

$$G_{\delta_i} = \langle \mathcal{A}_{\delta_i}, \mathcal{F}_{\delta_i} \rangle = \langle \{A_{\delta_i}, Z_{\delta_i}\}, \{\delta_i\} \rangle. \quad (15)$$

В данном выражении  $A_{\delta_i} \subseteq A$  есть множество всех состояний, присутствующих в кортежах вида (12) функции  $\delta_i$ . Иными словами, в  $A_{\delta_i}$  входят те и только те состояния, переходы из которых или переходы в которые реализуются функцией  $\delta_i$ . Множество  $Z_{\delta_i} \subseteq Z$  есть множество входных сигналов, присутствующих в кортежах вида (12) функции  $\delta_i$ . Поскольку  $A_{\delta_i} \subseteq A$ ,  $Z_{\delta_i} \subseteq Z$  и для любых  $a \in A_{\delta_i}$  и  $z \in Z_{\delta_i}$  выполняется равенство  $\delta_i(a, z) = \delta(a, z)$ , выражение (15) есть *подалгебра* алгебры (6) [4]. Будем называть алгебру (15) *абстрактной подалгеброй переходов*. В алгебраической теории автоматов понятию подалгебры соответствует понятие *подавтомата* [2, 3].

Множество частичных функций переходов приводит ко множеству абстрактных подалгебр переходов, отождествляемому с абстрактной алгеброй переходов (6):

$$G_\delta = \{G_{\delta_1}, \dots, G_{\delta_{N_\delta}}\}. \quad (16)$$

Компоненты  $\mathcal{A}_\delta$  и  $\mathcal{F}_\delta$  алгебры  $G_\delta$  формируются путем объединения соответствующих компонентов всех подалгебр:

$$\mathcal{A}_\delta = \bigcup_{i=1}^{N_\delta} \mathcal{A}_{\delta_i} = \left\{ \bigcup_{i=1}^{N_\delta} A_{\delta_i}, \bigcup_{i=1}^{N_\delta} Z_{\delta_i} \right\}, \quad \mathcal{F}_\delta = \bigcup_{i=1}^{N_\delta} \mathcal{F}_{\delta_i} = \left\{ \bigcup_{i=1}^{N_\delta} \delta_i \right\}.$$

Отметим, что в то время как частичные функции  $\delta_i$  образуют *разбиение*, т.е.  $\forall(i, j \in [1, N_\delta], i \neq j) : \delta_i \cap \delta_j = \emptyset$ , семейства множеств  $A_{\delta_i}$  и  $Z_{\delta_i}$  образуют на множествах  $A$  и  $Z$  соответствующие *покрытия*, что в общем случае допускает  $\forall(i, j \in [1, N_\delta]) : A_{\delta_i} \cap A_{\delta_j} \neq \emptyset$  и  $Z_{\delta_i} \cap Z_{\delta_j} \neq \emptyset$ . Присутствие одного и того же состояния  $a \in A$  в носителях нескольких подалгебр означает, что переходы в данное состояние и/или из него выполняются с помощью разных частичных функций  $\delta_i$ .

По аналогии с (16), в виде множества подалгебр представляются абстрактная алгебра выходов (7) автомата Мили и абстрактная алгебра выходов (8) автомата Мура.

Структурный автомат.

Модель структурного автомата  $S_S = \langle X, Y, A, d, l \rangle$  также является трехосновной алгеброй, носитель  $\mathcal{A}_S$  которой образован конечными множествами структур-

ного автомата, а сигнатура  $\mathcal{F}_S$  – структурной функцией переходов  $d$  и структурной функцией выходов  $l$ :

$$\mathcal{A}_S = \{X, A, Y\}, \quad (17)$$

$$\mathcal{F}_S = \{d, l\}. \quad (18)$$

Отметим, что в выражении (17) множество состояний  $A$  отличается от одноименного множества в выражении (4) тем, что содержит не абстрактные, а структурные состояния, представленные своими структурными кодами [10].

Назовем алгебру, отождествляемую со структурным автоматом, *структурной алгеброй*  $G_S$ :

$$G_S = \langle \mathcal{A}_S, \mathcal{F}_S \rangle = \langle \{X, A, Y\}, \{d, l\} \rangle. \quad (19)$$

Разделение функций переходов и выходов дает две отдельные алгебры: *структурную алгебру переходов*:

$$G_d = \langle \mathcal{A}_d, \mathcal{F}_d \rangle = \langle \{A, X\}, \{d\} \rangle \quad (20)$$

и *структурную алгебру выходов* (выражение (21) – для автомата Мили, (22) – для автомата Мура):

$$G_l = \langle \mathcal{A}_l, \mathcal{F}_l \rangle = \langle \{A, X, Y\}, \{l\} \rangle, \quad (21)$$

$$G_l = \langle \mathcal{A}_l, \mathcal{F}_l \rangle = \langle \{A, Y\}, \{l\} \rangle. \quad (22)$$

При этом структурный автомат может рассматриваться как двойка

$$S_S = \langle G_d, G_l \rangle. \quad (23)$$

Согласно модели  $S_S = \langle X, Y, A, d, l \rangle$ , функция  $d$  есть соответствие

$$d : (D_d \subseteq X \times A) \rightarrow A. \quad (24)$$

В структурном автомате все элементы множеств абстрактного автомата представляются в виде слов в структурном (двоичном) алфавите. Предположение о том, что память структурного автомата строится в базисе триггеров D-типа, дает возможность отождествлять результат структурной функции переходов (структурный код состояния перехода) со множеством функций возбуждения элементов памяти структурного автомата.

При  $|X| = L$  и  $\lceil \log_2(|A|) \rceil = R$  выражение (24) принимает вид:

$$d : (D_d \subseteq x_1 \times x_2 \times \dots \times x_L \times e_1 \times e_2 \times \dots \times e_R) \rightarrow e_1 \times e_2 \times \dots \times e_R, \quad (25)$$

где каждый из элементов  $x$  и  $e$  есть множество букв структурного (двоичного) алфавита, т.е. множество  $\{0, 1\}$  [9]. Фактически, (25) есть соответствие:

$$d : (D_d \subseteq \{0, 1\}^{L+R}) \rightarrow \{0, 1\}^R, \quad (26)$$

элементами которого являются кортежи, состоящие из  $(L+2R)$  букв  $q \in \{0, 1\}$ :

$$\langle q_1, \dots, q_{L+2R} \rangle \in D_d. \quad (27)$$

Структурная функция переходов (24), являясь множеством кортежей (27), может быть представлена в виде множества *частичных структурных функций* переходов:

$$d = \{d_1, \dots, d_{N_d}\}, \quad (28)$$

$$d_i : (D_{d_i} \subseteq D_d) \rightarrow A. \quad (29)$$

Отметим, что в выражении (28) максимально возможное количество частичных функций  $N_d$  определяется числом кортежей (27). Если в случае абстрактного автомата имеем  $1 \leq N_d \leq B$ , то для значения  $N_d$  дело обстоит иначе.

В процессе кодирования каждому символу  $z_f$  входного алфавита абстрактного автомата ставится в соответствие вектор вида  $\langle x_1, \dots, x_L \rangle$ , образованный значениями переменных  $x_1, \dots, x_L$ , соответствующих одноименным входным структурным сигналам. Поскольку способ кодирования элементов множества  $Z$  может быть произвольным, возможны ситуации, когда в векторе  $\langle x_1, \dots, x_L \rangle$ , соответствующем входному символу  $z_f \in Z$ , некоторая переменная  $x_l$  может принимать произвольное значение [13]. В этом случае каждому переходу, выполняющемуся под воздействием входного сигнала  $z_f$ , в структурной функции  $d$  соответствуют два вектора вида (27), в первом из которых компонент, соответствующий  $x_l$ , равен нулю, во втором – единице.

В общем случае количество фиктивных переменных в коде символа  $z_f$  может быть больше единицы. Такие ситуации часто возникают в процессе структурного синтеза МПА, заданных в виде ГСА, когда переход из состояния в состояние зависит лишь от части логических условий, отождествляемых со структурными входными сигналами. В работе [13] предлагается каждый структурный входной сигнал условно представлять заданным на трехзначном множестве  $\{0, 1, -\}$ , где элемент «-» соответствует произвольному значению из  $\{0, 1\}$ . Будем полагать, что каждая из функций  $d_i$  есть множество кортежей вида (27) с трехзначным представлением структурных входных сигналов. Тогда, как и для абстрактных частичных функций, имеем  $1 \leq N_d \leq B$ . Понятно, что при  $N_d = B$  каждому переходу автомата будет сопоставлена отдельная функция вида (29), в то время как при  $N_d = 1$  все переходы автомата реализуются одной общей структурной функцией переходов.

Установим теперь формальную связь между абстрактной и структурной алгебрами переходов. Однотипность алгебраических систем характеризуется возможностью установления между ними различных отображений, таких как *гомоморфизм*, *изоморфизм* и др. [4]. Воспользуемся понятием *гомоморфизма автоматов*, которое В. М. Глушков определяет следующим образом [11]. Гомоморфизм  $\Psi$  автомата  $A(Z_A, A_A, W_A, \delta_A, \lambda_A)$  в автомат  $B(Z_B, A_B, W_B, \delta_B, \lambda_B)$  есть совокупность трех однозначных отображений:  $\psi_1 : Z_A \rightarrow Z_B$ ,  $\psi_2 : A_A \rightarrow A_B$ ,  $\psi_3 : W_A \rightarrow W_B$ , удовлетворяющих для любых элементов  $a \in A_A$  и  $z \in Z_A$  соотношениям:

$$\psi_1(\delta_A(a, z)) = \delta_B(\psi_1(a), \psi_2(z)), \quad (30)$$

$$\psi_3(\lambda_A(a, z)) = \lambda_B(\psi_1(a), \psi_2(z)). \quad (31)$$

Хотя МПА в части функции переходов может быть задан как абстрактной алгеброй (3), так и структурной алгеброй (19), установление гомоморфизма между данными алгебрами, с формальной точки зрения, невозможно. Причиной является несоответствие носителей (4) и

(17), а точнее – различные мощности множеств, образующих носители. Тот факт, что в общем случае  $(|Z| = F) \neq (|X| = L)$  и  $(|W| = G) \neq (|Y| = N)$ , не позволяет установить между элементами соответствующих множеств взаимно однозначное соответствие, обязательное для гомоморфизма.

Преобразуем структурную алгебру (19) следующим образом.

1. Пусть носитель  $\mathcal{A}_S$  содержит следующие три множества

– множество  $K_S(Z) = \{K_S(z_1), \dots, K_S(z_F)\}$  структурных кодов входных сигналов абстрактного автомата, где каждый элемент  $K_S(z_f)$  есть вектор  $\langle x_1, \dots, x_L \rangle$  значений структурных входных сигналов, определенных на трехэлементном алфавите  $\{0, 1, -\}$ ;

– множество  $K_S(A) = \{K_S(a_1), \dots, K_S(a_M)\}$  структурных кодов состояний абстрактного автомата, заданных векторами  $\langle e_1, \dots, e_R \rangle$ ,  $e_r \in \{0, 1\}$ ;

– множество  $K_S(W) = \{K_S(w_1), \dots, K_S(w_G)\}$  структурных кодов выходных сигналов абстрактного автомата, заданных векторами  $\langle y_1, \dots, y_N \rangle$ ,  $y_n \in \{0, 1\}$ .

2. Структурные функции переходов  $d$  и выходов  $l$ , образующие сигнатуру  $\mathcal{F}_S$ , определим выражениями (32) и (33), (34) соответственно:

$$d : (D_d \subseteq K_S(Z) \times K_S(A)) \rightarrow K_S(A). \quad (32)$$

$$l : (D_l \subseteq K_S(Z) \times K_S(A)) \rightarrow K_S(W). \quad (33)$$

$$l : (D_l \subseteq K_S(A)) \rightarrow K_S(W). \quad (34)$$

Выражение (33) справедливо для автомата Мили, (34) – для автомата Мура.

Тогда структурная алгебра (19) принимает следующий вид:

$$G_S = \langle \mathcal{A}_S, \mathcal{F}_S \rangle = \langle \{K_S(Z), K_S(A), K_S(W)\}, \{d, l\} \rangle. \quad (35)$$

Структурные алгебры переходов и выходов определяются аналогично (20)–(22) следующим образом:

$$G_d = \langle \mathcal{A}_d, \mathcal{F}_d \rangle = \langle \{K_S(A), K_S(Z)\}, \{d\} \rangle, \quad (36)$$

$$G_l = \langle \mathcal{A}_l, \mathcal{F}_l \rangle = \langle \{K_S(A), K_S(Z), K_S(W)\}, \{l\} \rangle, \quad (37)$$

$$G_l = \langle \mathcal{A}_l, \mathcal{F}_l \rangle = \langle \{K_S(A), K_S(W)\}, \{l\} \rangle. \quad (38)$$

Структурная алгебра (35) отличается от (19) тем, что формальными аргументами и значениями ее операций являются не отдельные структурные сигналы, а элементы носителей абстрактной алгебры, представленные в виде векторов элементарных структурных сигналов. Это позволяет установить гомоморфизм абстрактной алгебры (3) в структурную алгебру (35). Поскольку между носителями структурной алгебры (35) и носителями абстрактной алгебры (3) могут быть установлены однозначные соответствия  $K_S(A) \rightarrow A$ ,  $K_S(Z) \rightarrow Z$ ,  $K_S(W) \rightarrow W$ , гомоморфизм алгебр (3) и (35) является *изоморфизмом* (взаимно однозначным гомоморфизмом, [11]). Очевидно, что существуют также изоморфизмы между алгебрами переходов (6) и (36), а также между алгебрами выходов: (7) и (37) для автомата Мили, (8) и (38) для автомата Мура.

Функция переходов (32) есть множества кортежей следующего вида:

$$\langle K_S(a_i), K_S(z_j), K_S(a_k) \rangle \in D_d \quad (39)$$

и может быть представлена в виде семейства частичных функций переходов (28), причем

$$d_i : (D_{d_i} \subseteq D_d) \rightarrow K_S(A). \quad (40)$$

Изоморфизм алгебр (3) и (35) устанавливает, в том числе, отображение между элементами соответствующих носителей данных алгебр. Сопоставляя каждому компоненту любого кортежа (12) соответствующий ему элемент из носителей структурной алгебры, можно получить тождественный кортеж (39). Следовательно, каждой частичной функции переходов  $\delta_i$  абстрактного автомата, определяемой выражением (13) и являющейся некоторым подмножеством множества кортежей (12), может быть поставлена в соответствие частичная функция переходов  $d_i$ , определяемая выражением (40) и являющаяся подмножеством множества кортежей (39). При этом изоморфизм абстрактной и структурной алгебр возможен при любом разбиении функций абстрактного автомата на множества частичных функций. Данное утверждение также справедливо как алгебр переходов (6) и (36) и для алгебр выходов (7)/(8) и (37)/(38).

По аналогии с (15), *структурной подалгеброй переходов* будем называть алгебру вида:

$$G_{d_i} = \langle \mathcal{A}_{d_i}, \mathcal{F}_{d_i} \rangle = \langle \{K_S(A_{d_i}), K_S(Z_{d_i})\}, \{d_i\} \rangle. \quad (41)$$

Здесь  $K_S(A_{d_i}) \subseteq K_S(A)$  – множество структурных кодов состояний, присутствующих в кортежах (39) частичной структурной функции  $d_i$ ;  $K_S(Z_{d_i}) \subseteq K_S(Z)$  – множество структурных кодов входных сигналов, используемых для кодирования входных сигналов в кортежах (39) функции  $d_i$ . Структурную алгебру переходов (35), по аналогии с (16), представим в виде множества структурных подалгебр переходов:

$$G_d = \{G_{d_1}, \dots, G_{d_{N_d}}\}. \quad (42)$$

Как и в случае абстрактного автомата, компоненты структурной алгебры переходов (42) формируются объединением элементов соответствующих компонентов своих подалгебр. Аналогичным образом в виде множества своих подалгебр могут быть представлены структурные алгебры выходов (37) и (38).

#### 4 ЭКСПЕРИМЕНТЫ

Рассмотрим отдельно функцию переходов автомата. В соответствии с предложенным В.М. Глушковым каноническим методом структурного синтеза функция переходов представляется в векторном виде и задается системой канонических скалярных уравнений [10]. Система скалярных уравнений строится известным способом [10, 13], при котором для каждой пары  $\langle K_S(a_i), K_S(z_j) \rangle$  кортежа (39), соответствующего одному переходу автомата, формируется терм, присутствие которого в том или ином уравнении системы определяется элементом

В структурной теории автоматов известен класс устройств, называемых автоматами на счетчике или С-автоматами [14]. В их основе лежит принцип специального кодирования состояний, при котором на множестве состояний заданной ГСА выделяются т.н. линейные последовательности состояний, внутри которых коды состояний задаются в естественном (последовательном) порядке.

Применим принцип представления функции переходов автомата в виде множества частичных функций к автомату на счетчике. Пусть произвольный абстрактный автомат задан алгеброй (3), а реализующий его автомат на счетчике – алгеброй (35). Функции переходов данных автоматов образуют сигнатуры абстрактной алгебры переходов (6) и структурной алгебры переходов (36) соответственно. В силу изоморфизма алгебр (3) и (35), алгебры переходов (6) и (36) также изоморфны:

$$G_\delta \leftrightarrow G_S. \quad (43)$$

Тот факт, что в С-автомате часть переходов реализуется с использованием инкрементного счетчика, часть – в виде системы канонических уравнений, позволяет представить функцию  $d$  в виде двух частичных функций  $d_1$  и  $d_2$ , образующих сигнатуры двух подалгебр переходов:

$$G_{d_1} = \langle \mathcal{A}_{d_1}, \mathcal{F}_{d_1} \rangle = \langle \{K_S(A_{d_1}), K_S(Z_{d_1})\}, \{d_1\} \rangle; \quad (44)$$

$$G_{d_2} = \langle \mathcal{A}_{d_2}, \mathcal{F}_{d_2} \rangle = \langle \{K_S(A_{d_2}), K_S(Z_{d_2})\}, \{d_2\} \rangle. \quad (45)$$

Представление функции  $d$  в виде двух частичных функций предполагает аналогичное представление абстрактной функции переходов  $\delta$ . Частичные абстрактные функции переходов  $\delta_1$  и  $\delta_2$  образуют сигнатуры двух абстрактных подалгебр переходов:

$$G_{\delta_1} = \langle \mathcal{A}_{\delta_1}, \mathcal{F}_{\delta_1} \rangle = \langle \{A_{\delta_1}, Z_{\delta_1}\}, \{\delta_1\} \rangle; \quad (46)$$

$$G_{\delta_2} = \langle \mathcal{A}_{\delta_2}, \mathcal{F}_{\delta_2} \rangle = \langle \{A_{\delta_2}, Z_{\delta_2}\}, \{\delta_2\} \rangle. \quad (47)$$

Из изоморфизма (43) вытекают изоморфизмы соответствующих подалгебр:

$$G_{\delta_1} \leftrightarrow G_{d_1}; \quad (48)$$

$$G_{\delta_2} \leftrightarrow G_{d_2}. \quad (49)$$

Структурная схема С-автомата, отражающая представление структурной функции переходов в виде двух частичных функций, приведена на рис. 1.

Здесь схема  $KC_{d_1}$  реализует частичную структурную функцию переходов  $d_1$  и представляет собой  $R$ -разрядный инкрементор. Структурный код  $T$  текущего состояния, представленный набором структурных переменных  $\langle T_1, \dots, T_R \rangle$ , в котором  $T_r \in \{0, 1\}$ , интерпретируется на входе  $KC_{d_1}$  как  $R$ -разрядное беззнаковое целое, представленное в структурном (двоичном) алфавите. На выходе инкрементора формируется структурный код состояния перехода, обозначенный символом  $d_1$  и отождествляемый со значением одноименной функции переходов.

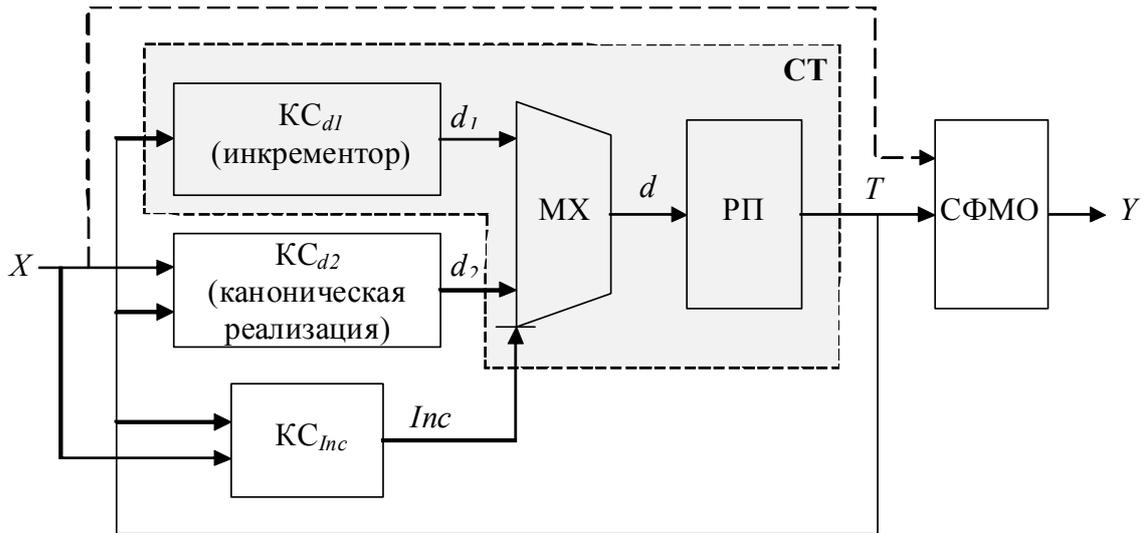


Рисунок 1 – Структурная схема С-автомата

Частичная функция  $d_2$  представлена комбинационной схемой  $KS_{d_2}$ . Данная схема строится по известной методике [14] и реализует систему канонических уравнений, соответствующую функции  $d_2$ . На вход  $KS_{d_2}$  поступают структурный код текущего состояния  $T = \langle T_1, \dots, T_R \rangle$  и структурный код входного слова, представленный набором  $X = \langle x_1, \dots, x_L \rangle$ , в котором  $x_l \in \{0, 1, -\}$ . Код, формируемый на выходе  $KS_{d_2}$ , обозначен символом  $d_2$  и отождествляется с значением одноименной функции переходов.

Поскольку каждая функция  $d_i$ , представленная в виде логической схемы, определена на всех наборах аргументов, но при этом отдельный автоматный переход реализуется всегда какой-то одной частичной функцией, на этапе структурного синтеза возникает задача организации выбора нужной частичной функции при каждом переходе автомата. В С-автомате данная задача решается мультиплексированием одновременно формируемых значений двух частичных функций. На рис. 1 схема  $KS_{Inc}$  вырабатывает сигнал  $Inc$ , управляющий мультиплексором  $MX$ . Формируемый на выходе мультиплексора структурный код  $d$  состояния перехода отождествляется со значением структурной функции переходов автомата и поступает в регистр памяти  $РП$ . Отметим, что в структуре С-автомата, приведенной в [14], блокам  $KS_{d_1}$ ,  $MX$  и  $РП$  соответствует инкрементный счетчик  $СТ$ .

Увеличение числа переходов, реализуемых функцией  $d_1$ , приводит, в общем случае, к снижению аппаратных затрат в схеме формирования переходов С-автомата по сравнению с канонической структурой МПА. В связи с этим С-автоматы рекомендуется использовать для линейных ГСА, в которых не менее 75% вершин являются операторными [14].

Не смотря на то, что каждая частичная функция реализует собственное подмножество переходов, одно и то же состояние автомата может встречаться в кортежах нескольких частичных функций. Например, переход в

состояние  $a_k$  может выполняться с помощью частичной функции  $\delta_1$ , переход из этого состояния – с помощью частичной функции  $\delta_2$ . Если изоморфизмы (48) и (49) задавать независимо друг от друга, возможны ситуации, когда один и тот же структурный код присвоен двум различным состояниям, либо одному и тому же состоянию присвоено два различных структурных кода. Сказанное относится также и к структурным кодам входных сигналов. Подобные неоднозначности делают невозможным последующий синтез логической схемы автомата.

Решение данной проблемы заключается во *взаимосвязанном задании изоморфизмов* (48) и (49), при котором:

- каждое состояние  $a_i \in A$  получает единственный и уникальный структурный код  $K_S(a_i)$  в соответствующих носителях всех структурных подалгебр, в которых оно встречается;

- каждый входной сигнал  $z_j \in Z$  получает единственный и уникальный структурный код  $K_S(z_j)$  в соответствующих носителях всех структурных подалгебр, в которых он встречается.

В алгебраической форме данные требования могут быть выражены требованием существования изоморфизма (43). Данный изоморфизм устанавливает, в том числе, взаимно однозначные соответствия между каждым состоянием  $a_i \in A$  и его структурным кодом  $K_S(a_i) \in K_S(A)$ , а также между каждым входным сигналом  $z_j \in Z$  и его структурным кодом  $K_S(z_j) \in K_S(Z)$ . В результате за каждым состоянием и входным сигналом закрепляется единственный структурный код независимо от количества используемых подалгебр.

Применяемая в С-автомате операция инкремента является арифметической операцией и определена для скалярных величин. В процессе синтеза С-автомата на этапе формирования линейных последовательностей состояний каждое состояние  $a_i \in A_{\delta_1}$  получает уникальный «скалярный» код  $K_I(a_i)$  из множества натуральных чисел. «Скалярные» коды назначаются всем состояниям, переходы в которые или переходы из которых

выполняются с помощью операции инкремента. Таким образом, результатом кодирования всех состояний множества  $A_{\delta_1}$  является множество скалярных кодов  $K_I(A_{\delta_1})$ . В то же время состояния, отсутствующие в кортежах функции  $\delta_1$ , «скалярных» кодов не получают.

Обозначим операцию инкремента символом  $O_{Inc}$ . Заметим, что множество  $K_I(A_{\delta_1})$  совместно с операцией  $O_{Inc}$  можно рассматривать как алгебру

$$G_I = \langle \mathcal{A}_I, \mathcal{F}_I \rangle = \langle K_I(A_{\delta_1}), O_{Inc} \rangle. \quad (50)$$

Здесь операция  $O_{Inc}$  формально представляется множеством кортежей вида  $\langle K_I(a_i), K_I(a_j) \rangle \in O_{Inc}$ . Компонент  $K_I(a_i) \in K_I(A_{\delta_1})$  есть «скалярный» код исходного состояния, компонент  $K_I(a_j) \in K_I(A_{\delta_1})$  – «скалярный» код состояния перехода. Очевидно, что подобный кортеж не содержит информации о входном сигнале, под воздействием которого выполняется переход, и в таком виде не может быть поставлен в соответствие кортежам частичных функций  $\delta_1$  и  $d_1$ .

Введем в алгебру (50) дополнительный носитель  $Z_{\delta_1}$ , идентичный одноименному носителю из абстрактной подалгебры  $G_{\delta_1}$ . В результате получим двухосновную алгебру:

$$G_I = \langle \mathcal{A}_I, \mathcal{F}_I \rangle = \langle \{K_I(A_{\delta_1}), Z_{\delta_1}\}, \{O_{Inc}\} \rangle, \quad (51)$$

в которой операция  $O_{Inc}$  задается множеством кортежей вида:

$$\langle K_I(a_i), z_j, K_I(a_k) \rangle \in O_{Inc}, \quad (52)$$

где  $K_I(a_i), K_I(a_j) \in K_I(A_{\delta_1})$ ,  $z_j \in Z_{\delta_1}$ .

Теперь, учитывая возможность установления попарных взаимно однозначных соответствий между носителями  $A_{\delta_1}$ ,  $K_I(A_{\delta_1})$  и  $K_S(A_{d_1})$ , а также между носителями  $Z_{\delta_1}$  и  $K_S(Z_{d_1})$ , каждому кортежу вида (12) в функции  $\delta_1$  и соответствующему ему кортежу вида (39) в функции  $d_1$  может быть поставлен в соответствие единственный кортеж вида (52). Это позволяет задать операцию  $O_{Inc}$  таким множеством кортежей (52), чтобы в рамках алгебры (51) она выполняла преобразования «скалярных» кодов состояний по тому же закону, что и функции  $\delta_1$  и  $d_1$  в своих подалгебрах, реализуя, таким образом, те же автоматные переходы, что и функции  $\delta_1$  и  $d_1$ .

Все это позволяет говорить о взаимном изоморфизме трех алгебр: алгебры (51), абстрактной подалгебры  $G_{\delta_1}$  и структурной подалгебры  $G_{d_1}$ :

$$G_{\delta_1} \leftrightarrow G_I \leftrightarrow G_{d_1}. \quad (53)$$

Данное выражение содержит три изоморфизма:  $G_{\delta_1} \leftrightarrow G_{d_1}$ ,  $G_{\delta_1} \leftrightarrow G_I$  и  $G_I \leftrightarrow G_{d_1}$ . Учитывая требование их одновременного существования, будем для краткости рассматривать данные изоморфизмы как единый изоморфизм (53).

Если структурные коды состояний, встречающиеся в кортежах частичной функции  $d_2$ , могут задаваться произвольно (из допустимого множества структурных кодов), то в случае функции  $d_1$  структурные коды должны задаваться таким образом, чтобы сделать возможным их преобразование с помощью операции инкремента. Задание «скалярных» кодов состояний в естественном (инкрементном) порядке есть задание транзитивного замыкания для отношения инкремента на некотором множестве скалярных чисел. На произвольном множестве структурных кодов, для которых неизвестна их «скалярная» интерпретация, отношение инкремента не может быть определено. По этой причине выбор «скалярных» кодов состояний множества  $A_{\delta_1}$  должен предшествовать выбору их структурных кодов. В свою очередь, задание структурных кодов состояний множества  $A_{\delta_2}$  выполняется после задания структурных кодов состояний множества  $A_{\delta_1}$ , а синтез схем  $KC_{d_2}$  и  $KC_{Inc}$  возможен лишь после задания структурных кодов всех состояний автомата.

## 5 РЕЗУЛЬТАТЫ

Применение принципа представления функции переходов автомата в виде множества частичных функций позволяет, в случае автомата на счетчике, сформировать следующий порядок задания алгебр:

1. Задание абстрактных подалгебр переходов (46) и (47) в соответствии со сформированными линейными последовательностями состояний.
2. Задание алгебры (51), изоморфной подалгебре (46).
3. Задание структурной подалгебры (44), изоморфной алгебре (51).
4. Задание структурной подалгебры (45), изоморфной подалгебре (44).

Учитывая то, что алгебра (51) играет в С-автомате вспомогательную роль при построении структурной алгебры переходов по заданной абстрактной алгебре переходов, назовем алгебру  $G_I$  *промежуточной алгеброй переходов*, понимая под нижним индексом «I» термин «intermediate» – «промежуточный». «Скалярные» коды состояний, образующие носитель промежуточной алгебры, назовем *промежуточными кодами состояний*. Операцию, образующую сигнатуру промежуточной алгебры, назовем *операцией переходов*.

В алгебраической форме эквивалентность абстрактного автомата и реализующего его структурного С-автомата выражается существованием следующих изоморфизмов:

1. Изоморфизм (53) промежуточной алгебры  $G_I$  с соответствующими ей абстрактной и структурной подалгебрами  $G_{\delta_1}$  и  $G_{d_1}$ . Данный изоморфизм выражает реализацию части переходов автомата с помощью операции инкремента.
2. Изоморфизм (43) абстрактной и структурной алгебр переходов. Данный изоморфизм обеспечивается совместным заданием изоморфизмов (48) и (49), при котором сохраняются уникальность и однозначность структурных кодов состояний и входных сигналов независимо от способа разбиения функции переходов на частичные функции.

Если хотя бы один из данных изоморфизмов не может быть задан, то не может быть построен С-автомат, эквивалентный заданному абстрактному автомату. Таким образом, каждый из изоморфизмов является необходимым условием существования С-автомата, а достаточным условием является одновременное существование обоих изоморфизмов. Сведем изоморфизмы (53) и (43) в систему (54), которую будем рассматривать в качестве *математической модели автомата на счетчике*:

$$\begin{cases} G_{\delta_1} \leftrightarrow G_I \leftrightarrow G_{d_1}; \\ G_{\delta} \leftrightarrow G_d. \end{cases} \quad (54)$$

Процесс построения системы (54) включает выполнение следующих этапов структурного синтеза С-автомата: формирование линейных последовательностей состояний, кодирование состояний, формирование таблицы переходов [14].

После задания системы (54) следует завершающий этап структурного синтеза – реализация в заданном элементном базисе логической схемы автомата (комбинационных схем  $KC_{d_2}$ ,  $KC_{Inc}$  и схемы формирования микроопераций) в соответствии со сформированными множеством структурных кодов состояний и разбиением функции переходов на частичные функции. Таким образом, метод структурного синтеза С-автомата может быть сокращенно представлен двумя этапами:

1. Задание системы изоморфизмов (54).

2. Реализация логической схемы автомата в заданном элементном базисе.

## 6 ОБСУЖДЕНИЕ

Представление функции переходов С-автомата в виде двух частичных функций наряду с использованием промежуточной алгебры переходов отражает совмещение в одной структуре двух различных способов реализации переходов автомата – канонического и инкрементного. Разбиение функции переходов на частичные функции в случае канонической структуры МПА не имеет смысла, поскольку в этом случае каждая частичная функция будет лишь реализовывать некоторый фрагмент единой системы канонических уравнений, а требуемая при этом реализация выбора частичной функции путем мультиплексирования приведет к увеличению аппаратных затрат в схеме устройства.

Приведенный пример использования промежуточной алгебры на базе операции инкремента позволяет предположить возможность формирования сигнатуры промежуточной алгебры из других операций, например, операции декремента. Подобный подход может способствовать снижению аппаратных затрат в схеме проектируемого микропрограммного автомата за счет использования в схеме формирования переходов таких комбинационных схем, аппаратные затраты в которых не зависят или зависят незначительно от количества реализуемых автоматных переходов.

## ВЫВОДЫ

В настоящей работе предложены новые математические модели абстрактного и структурного автоматов, в основе которых лежит представление функции переходов в виде множества частичных функций. Результатом исследований является представление процессов преобразования кодов состояний в структурном автомате в виде промежуточной алгебры переходов, описывающей закон преобразования кодов состояний для некоторого подмножества автоматных переходов.

Использование различных операций переходов позволяет выделять на функции переходов структурного автомата различные законы преобразования кодов состояний, что может способствовать снижению аппаратных затрат в логической схеме автомата в сравнении с другими известными структурами МПА. Вопрос целесообразности применения предложенного подхода для оптимизации аппаратных затрат в логической схеме микропрограммного автомата не является очевидным из предложенных моделей и требует самостоятельного исследования.

## СПИСОК ЛИТЕРАТУРЫ

1. Баранов С. И. Синтез микропрограммных автоматов / С. И. Баранов. – Л. : Энергия, 1979. – 232 с.
2. Алгебраическая теория автоматов, языков и полугрупп / под ред. М. Арбиба ; пер. с англ. – М. : Статистика, 1975. – 335 с.
3. Плоткин Б. И. Элементы алгебраической теории автоматов: учеб. пособие для вузов / Б. И. Плоткин, Л. Я. Гринглаз, А. А. Гварамия. – М. : Высшая школа, 1994. – 191 с.
4. Мальцев А. И. Алгебраические системы / А. И. Мальцев. – М. : Наука, 1970. – 392 с.
5. Плоткин Б. И. Универсальная алгебра, алгебраическая логика и базы данных / Б. И. Плоткин. – М. : Наука, гл. ред. физ.-мат. лит., 1991. – 448 с.
6. Судоплатов С. В. Элементы дискретной математики : учебник / С. В. Судоплатов, Е. В. Овчинникова. – М. : ИНФРА-М, Новосибирск : Изд-во НГТУ, 2002. – 280 с.
7. Богомолов А. М. Алгебраические основы теории дискретных систем / А. М. Богомолов, В. Н. Салий. – М. : Наука-Физматлит, 1997. – 368 с.
8. Новиков Ф. А. Дискретная математика для программистов / Ф. А. Новиков. – СПб. : Питер, 2000. – 304 с.
9. Кудрявцев В. Б. Введение в теорию автоматов / В. Б. Кудрявцев, С. В. Алешин, А. С. Подколзин. – М. : Наука, 1985. – 320 с.
10. Глушков В. М. Синтез цифровых автоматов / В. М. Глушков. – М. : Физматгиз, 1962. – 476 с.
11. Глушков В. М. Абстрактная теория автоматов / В. М. Глушков // Успехи математических наук. – 1961. – Т. XVI, Вып. 5. – С. 3–62.
12. Трахтенброт Б.А. Конечные автоматы (поведение и синтез) / Б. А. Трахтенброт, Я.М. Бардзинь. – М. : Наука, 1970. – 400 с.
13. Шиханович Ю. А. Введение в современную математику (начальные понятия) / Ю. А. Шиханович. – М. : Наука, 1965. – 376 с.
14. Баркалов А. А. Синтез устройств управления на программируемых логических устройствах / А. А. Баркалов. – Донецк : ДонНТУ, 2002. – 262 с.

Статья поступила в редакцию 24.09.2015.  
После доработки 05.10.2015.

Бабаков Р. М.

Канд. техн. наук, доцент, доцент кафедри комп'ютерних наук Донецького національного технічного університету, м. Красноармійськ, Україна

#### ПРОМІЖНА АЛГЕБРА ПЕРЕХОДІВ В МІКРОПРОГРАМНОМУ АВТОМАТІ

Вирішено задачу формалізації завдання мікропрограмного автомата, в структурі якого частина автоматних переходів реалізується неканонічним шляхом. Запропоновано новий підхід до організації функції переходів мікропрограмного автомата, відповідно до якого функція переходів представляється у вигляді сімейства часткових функцій, кожна з яких визначена лише на частині області визначення функції переходів автомата і відповідає певній підмножині автоматних переходів.

З урахуванням запропонованого підходу традиційне представлення автомата у вигляді багатоосновної алгебри матиме певні зміни. По-перше, взаємна незалежність функцій переходів і виходів, що утворюють сигнатуру алгебри, дозволяє розглядати їх окремо одна від іншої, що призводить до представлення автомата у вигляді двох алгебр: алгебри переходів, сигнатура якої містить лише функцію переходів, і алгебри виходів, сигнатура якої містить лише функцію виходів. По-друге, представлення функції переходів у вигляді множини часткових функцій виводить до заміни алгебри переходів множиною підалгебр переходів, в кожній з яких сигнатура утворена частковою функцією переходів.

На прикладі мікропрограмного автомата з лічильником показано, що закон перетворення кодів станів в рамках певної підмножини переходів може бути заданий алгебраїчною функцією (операцією переходів), що використовує скалярну інтерпретацію кодів станів структурного автомата. Скалярну інтерпретацію кодів станів разом з операцією переходів пропонується представляти у вигляді так званої проміжної алгебри переходів, яка ізоморфна відповідним підалгебрам переходів абстрактного і еквівалентного йому структурного автоматів.

**Ключові слова:** мікропрограмний автомат, часткова функція переходів, проміжна алгебра переходів, автомат із лічильником.

Babakov R. M.

PhD, Associate Professor of the Department of Computer Sciences of the Donetsk National Technical University, Krasnoarmiysk, Ukraine

#### INTERMEDIATE ALGEBRA OF TRANSITIONS IN MICROPROGRAM FINAL-STATE MACHINE

The problem of formalization of representation of final-state machine, where the part of automaton transition is realized in non-canonical way, is solved. A new approach for organization of the function of transitions of the final-state machine is proposed. According to it the function of transitions is represented as a family of partial functions, each of which is defined only on the part of the domain of the function of transitions, and corresponds to a subset of the automaton transitions.

According to the proposed approach the traditional representation of the final-state machine as a polybasic algebra is changed. First, the mutual independence of the function of transitions and function of outputs that form the signature of algebra, allows us to consider them separately from each other. This way leads to presentation of the final-state machine as two algebras: the algebra of transitions whose signature contains only the function of transitions and algebra of outputs whose signature contains only the function of outputs. Second, the representation of the function of transitions in the form of a set of partial functions leads to the replacement of the algebra of transitions by the set of subalgebras of transitions, a signature of each of which is formed by partial function of transitions.

The example of the final-state machine with a counter shows that the law of transformation of codes of states within a certain subset of transitions can be set by an algebraic function (operation of transitions) using scalar interpretation of codes of states of the structural final-state machine. The representation of scalar interpretation of codes of states and the operation of transitions as so-called intermediate algebra of transitions isomorphic to both according subalgebras of transitions of the abstract and equivalent structural final-state machines is proposed.

**Keywords:** microprogram final-state machine, partial function of transitions, intermediate algebra of transitions, final-state machine with counter.

#### REFERENCES

1. Baranov S. I. Sintez mikroprogramnih avtomatov. Leningrad, Energiya, 1979, 232 p.
2. Algebraicheskaya teoriya avtomatov, yazikov i polugrupp. Pod red. M. Arbiba. Per. s angl. Moscow, Statistika, 1975, 335 p.
3. Plotkin B. I., Gringlaz L. Y., Gvaramiya A. A. Elementi algebraicheskoy teorii avtomatov: ucheb. posobie dlya vuzov. Moscow, Vish. shk., 1994, 191 p.
4. Maltsev A. I. Algebraicheskie sistemi. Moscow, Nauka, 1970, 392 p.
5. Plotkin B. I. Universalnaya algebra, algebraicheskaya logika i bazi danih. Moscow, Nauka. Gl. red. fiz.-mat. lit., 1991, 448 p.
6. Sudoplatov S. V., Ovchinnikova E. V. Elementi diskretnoy matematiki: uchebnik. Moscow, INFRA-M, Novosibirsk, Izdatelstvo NGTU, 2002, 280 p.
7. Bogomolov A. M., Saliy V. N. Algebraicheskie osnovi teorii diskretnih sistem. Moscow, Nauka, Fizmatlit, 1997, 368 p.
8. Novikov F. A. Diskretnaya matematika dlya programmistov. SPb, Piter, 2000, 304 p.
9. Kudryavtsev V. B., Alyoshin S. V., Podkolzin A. S. Vvedenie v teoriyu avtomatov. Moscow, Nauka, 1985, 320 p.
10. Glushkov V. M. Sintez tsifrovih avtomatov. Moscow, Fizmatgiz, 1962, 476 p.
11. Glushkov V. M. Abstraktnaya teoriya avtomatov, *Uspehi matematicheskikh nauk*, 1961, Vol. XVI, Issue 5, pp. 3–62.
12. Trahtenbrot B. A., Bardzin Y. M. Konechnie avtomati (povedenie i sintez). Moscow, Nauka, 1970, 400 p.
13. Shihanovich U. A. Vvedenie v sovremennuyu matematiku (nachalnie ponyatiya). Moscow, Nauka, 1965, 376 p.
14. Barkalov A. A. Sintez ustroystv upravleniya na programmiruemykh logicheskikh ustroystvah. Donetsk, DonNTU, 2002, 262 p.