УДК: 681.324

Р.М. Бабаков, А.А. Баркалов

## Реализация функции переходов микропрограммного автомата на базе операционного автомата

Предложен новый принцип организации схемы формирования переходов микропрограммного автомата в виде операционного автомата. В основе принципа лежит представление функции переходов в виде множества частичных функций наряду со специальным кодированием состояний. Приведен пример построения микропрограммного автомата с операционным автоматом переходов по граф-схеме алгоритма.

**Ключевые слова**: микропрограммный автомат, алгебраическое представление, частичная функция переходов, изоморфизм, операционный автомат переходов.

Запропоновано новий принцип організації схеми формування переходів мікропрограмного автомата у вигляді операційного автомата. В основу принципу покладено представлення функції переходів у вигляді набору часткових функцій разом із спеціальним кодуванням станів. Наведено приклад побудови мікропрограмного автомата з операційним автоматом переходів за граф-схемою алгоритму.

**Ключові слова**: мікропрограмний автомат, алгебраїчне представлення, часткова функція переходів, ізоморфізм, операційний автомат переходів.

**Введение.** Один из центральных узлов современных вычислительных систем – устройство управления, координирующее работу всех узлов системы. Реализация устройства управления в виде микропрограммного автомата (МПА) приводит к схемам, характеризующимся относительно высоким быстродействием при значительных аппаратурных затратах [1, 2]. В статье предлагается новая структура МПА, основанная на представлении функции переходов автомата в виде нескольких частичных функций. Специфика предлагаемой структуры позволяет провести аналогию в организации схемы формирования переходов с операционным автоматом.

**Постановка задачи**

Предложенный академиком Глушковым В.М. канонический метод структурного синтеза цифровых автоматов предполагает представление функции переходов МПА в виде системы канонических уравнений, сложность которой пропорциональна числу переходов автомата. Наблюдаемый сегодня рост сложности алгоритмов, интерпретируемых МПА, приводит к увеличению аппаратурных затрат в логической схеме автомата, что актуализирует задачу снижения аппаратурных затрат в логической схеме МПА. Одно из решений данной задачи – разработка новых структур МПА, обладающих меньшими аппаратурными затратами в сравнении с ранее известными структурами. Формализация их описания предполагает использование определенного математического аппарата, достаточного для определения информационных процессов, происходящих в логической схеме устройства. В данной статье в качестве математической основы использован аппарат теории универсальных алгебр, позволяющий выразить эквивалентность абстрактного автомата и предлагаемого структурного автомата в виде системы некоторых изоморфизмов.

**Алгебраическая интерпретация МПА**

На рис. 1 представлена каноническая структура микропрограммного автомата [1]. Схема формирования переходов (СФП) на основании сигналов логических условий $X$ и кода текущего состояния $T$ формирует код $d$ следующего состояния, поступающий в регистр памяти (РП). Схема формирования микроопераций (СФМО) генерирует множество сигналов микроопераций $Y$, поступающих в объект управления. При наличии связи, показанной пунктиром, схема соответствует автомату Мили, при отсутствии – автомату Мура.

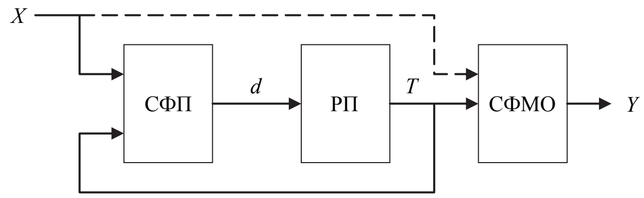


Рис. 1

Пусть заданы абстрактный автомат и имплементирующий его микропрограммный автомат. В соответствии с алгебраической теорией автоматов [3–5], в качестве математической модели абстрактного автомата может выступать алгебра

$$G_A = < \mathrm{A}_A, \mathrm{F}_A >, \tag{1}$$

где $\mathrm{A}_A$ – носитель, содержащий все алфавиты абстрактного автомата:

$$\mathrm{A}_A = \{A, Z, W\}, \tag{2}$$

$\mathrm{F}_A$ – сигнатура, включающая абстрактные функции переходов и выходов:

$$\mathrm{F}_A = \{\delta, \lambda\}. \tag{3}$$

Множества $Z = \{z_1,..,z_F\}$, $W = \{w_1,..,w_G\}$ и $A = \{a_1,..,a_M\}$ есть, соответственно, множества входных сигналов, выходных сигналов и состояний абстрактного автомата.

Назовем алгебру (1) *абстрактной алгеброй*. Рассматривая функции переходов δ и выходов λ отдельно друг от друга, представим *абстрактный* автомат системой двух таких алгебр: переходов (4) и выходов (5):

$$G_\delta = < \mathsf{A}_\delta, \mathsf{F}_\delta > \; = \; < \{A, Z\}, \{\delta\} >, \tag{4}$$

$$G_\lambda = < \mathsf{A}_\lambda, \mathsf{F}_\lambda > \; = \; < \{A, Z, W\}, \{\lambda\} >. \tag{5}$$

Выражение (5) справедливо для автомата Мили. Для автомата Мура необходимо из носителя исключить компонент $Z$:

$$G_\lambda = < \mathsf{A}_\lambda, \mathsf{F}_\lambda > \; = \; < \{A, W\}, \{\lambda\} >. \tag{6}$$

Как известно, в абстрактном автомате функция переходов δ есть соответствие

$$\delta : (D_\delta \subseteq A \times Z) \to A \tag{7}$$

и, в общем случае, частично определенная на множестве $A \times Z$. Формально функция (7) есть множество кортежей

$$< a_i, z_j, a_k > \in D_\delta\,, \tag{8}$$

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 множество кортежей (8) некоторым способом на попарно непересекающиеся подмножества, максимально возможное число которых равно числу переходов автомата. Каждое подмножество назовем *частичной абстрактной функцией переходов* $\delta_i$, понимая здесь под термином *частичная* ее неполное определение на области определения $D_\delta$ функции переходов δ:

$$\delta_i : (D_{\delta_i} \subseteq D_\delta) \to A\,. \tag{9}$$

При этом функция переходов δ есть множество подмножеств кортежей (8), т.е. множество частичных функций переходов:

$$\delta = \{\delta_1, \ ..., \ \delta_{N_\delta}\}, \tag{10}$$

где $1 \le N_\delta \le B$, $B$ – число переходов автомата, равное числу непустых клеток таблицы переходов. При этом $\bigcup_{i=1}^{N_\delta} \delta_i = \delta$.

Свяжем с каждой частичной функцией переходов $\delta_i$ алгебру следующего вида:

$$G_{\delta_i} = < \mathsf{A}_{\delta_i}, \mathsf{F}_{\delta_i} > \; = \; < \{A_{\delta_i}, Z_{\delta_i}\}, \{\delta_i\} >. \tag{11}$$

В данном выражении $A_{\delta_i} \subseteq A$ есть множество всех состояний, присутствующих в кортежах (8) частичной абстрактной 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 (2.12) 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 (11) есть *подалгебра* алгебры (4) [6]. Будем называть алгебру (11) *абстрактной подалгеброй переходов*. В алгебраической теории автоматов понятию подалгебры соответствует понятие *подавтомата* [7].

Множеству частичных функций переходов соответствует множество абстрактных подалгебр переходов, которое условно можно отождествлять с абстрактной алгеброй (4):

$$G_\delta = \{G_{\delta_1}, \ ..., \ G_{\delta_{N_\delta}}\}. \qquad (12)$$

При этом компоненты $A_\delta$ и $F_\delta$ алгебры $G_\delta$ формируются объединением соответствующих компонентов всех подалгебр: $\mathsf{A}_\delta = \bigcup_{i=1}^{N_\delta} \mathsf{A}_{\delta_i} = \left\{ \bigcup_{i=1}^{N_\delta} A_{\delta_i}, \bigcup_{i=1}^{N_\delta} Z_{\delta_i} \right\}$, $\mathsf{F}_\delta = \bigcup_{i=1}^{N_\delta} \mathsf{F}_{\delta_i} = \left\{ \bigcup_{i=1}^{N_\delta} \delta_i \right\}$.

Отметим, что в то время как частичные функции $\delta_i$ образуют разбиение, т.е. $\forall\,(i, j \in [1, N_\delta], i \neq j): \delta_i \cap \delta_j = \varnothing$, семейства множеств $A_{\delta_i}$ и $Z_{\delta_i}$ образуют на множествах $A$ и $Z$ соответствующие *покрытия*, что в общем случае допускает $\forall\,(i, j \in [1, N_\delta]): A_{\delta_i} \cap A_{\delta j} \neq \varnothing$ и $Z_{\delta_i} \cap Z_{\delta_j} \neq \varnothing$. 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. Присутствие одного и того же входного сигнала $z \in Z$ в носителях нескольких подалгебр говорит о том, что данный сигнал используется несколькими частичными функциями $\delta_i$.

Дадим алгебраическую интерпретацию структурного автомата. Пусть функция переходов $d$ структурного автомата образует сигнатуру следующей алгебры:

$$G_d = \langle A_d, F_d \rangle = \langle \{K_S(A), K_S(Z)\}, \{d\}. \qquad (13)$$

Здесь $K_S(A)$ – множество структурных кодов состояний, представляемых векторами переменных $\langle T_1, \ ..., \ T_R \rangle$, $T_r \in \{0,1\}$. $K_S(Z)$ – множество структурных кодов входных сигналов, образованных векторами $\langle x_1, \ ..., \ x_L \rangle$ переменных, отождествляемых с одноименными входными структурными сигналами. Функция переходов $d$ есть отображение

$$d : (D_d \subseteq K_S(Z) \times K_S(A)) \to K_S(A), \qquad (14)$$

формально представляемое множеством кортежей вида

$$\langle K_S(a_i), \ K_S(z_j), \ K_S(a_k) \rangle \in D_d. \qquad (15)$$

Назовем алгебру (13) *структурной алгеброй переходов*.

В общем случае некоторые коды $K_S(z_i)$ могут не зависеть от ряда компонентов вектора $\langle x_1, \ ..., \ x_L \rangle$. С учетом этого будем условно представлять каждую переменную $x_i$ заданной на трехэлементном множестве {0, 1, –}, где элемент «–» соответствует произвольному значению из {0, 1} [8]. При этом во множество $Z$ часто вводят специальный сигнал $z_0$, в структурном коде которого все компоненты вектора $\langle x_1, \ ..., \ x_L \rangle$ принимают значения, равные «–». Данный сигнал сопоставляется безусловным переходам автомата.

При трехзначном представлении переменных $x_i$ каждому кортежу (8) абстрактной функции переходов $\delta$ может быть поставлен в соответствие единственный кортеж вида (15) в структурной функции переходов $d$. Поскольку между элементами носителей абстрактной алгебры переходов (множеств $A$ и $Z$) и элементами соответствующих носителей структурной алгебры переходов (множеств $K_S(A)$ и $K_S(Z)$) существуют биекции, можно утверждать, что существует *изоморфизм* алгебры (4) в алгебру (13). Обозначим данный изоморфизм следующим выражением:

$$G_\delta \leftrightarrow G_d. \qquad (16)$$

Как и абстрактная функция переходов, структурная функция переходов $d$ может быть представлена в виде семейства частичных структурных функций переходов:

$$d = \{d_1, \ ..., \ d_{N_d}\}, \qquad (17)$$

где

$$d_i : (D_{d_i} \subseteq D_d) \to K_S(A). \qquad (18)$$

По аналогии с (11), с*труктурной подалгеброй переходов* будем называть алгебру вида

$$G_{d_i} = \langle \mathsf{A}_{d_i}, \mathsf{F}_{d_i} \rangle = \langle \{K_S(\mathsf{A}_{d_i}), K_S(Z_{d_i})\}, \{d_i\} \rangle, \qquad (19)$$

где $K_S(A_{d_i}) \subseteq K_S(A)$ – множество структурных кодов состояний, присутствующих в кортежах (15) функции $d_i$; $K_S(Z_{d_i}) \subseteq K_S(Z)$ – множество структурных кодов входных сигналов, используемых в кортежах (15) функции $d_i$. Структурную алгебру переходов (13), по аналогии с (12), представим в виде множества структурных подалгебр переходов:

$$G_d = \{G_{d_1}, ..., G_{d_{N_d}}\}. \qquad (20)$$

Будем полагать, что для эквивалентных абстрактного и структурного автоматов $N_d = N_\delta$. Тот факт, что количество кортежей (8) в функции δ равно количеству кортежей (15) в функции $d$ и равно числу $B$ переходов автомата, позволяет каждую частичную функцию переходов $\delta_i$ представить тождественной структурной функцией переходов $d_i$. Это дает возможность, при условии существования изоморфизма (16), задать подалгебры $G_{\delta_i}$ и $G_{d_i}$ таким образом, чтобы для любого $i \in [1, N_d]$ существовали изоморфизмы

$$G_{\delta_i} \leftrightarrow G_{d_i}. \qquad (21)$$

При $N_\delta > 1$ возможна ситуация, когда некоторое состояние $a_i$ имеет в подалгебре $G_{d_m}$ структурный код $K_S^m(a_i)$, а в подалгебре $G_{d_n}$ – структурный код $K_S^n(a_i)$, причем $K_S^m(a_i) \neq$ $\neq K_S^n(a_i)$. Аналогичная ситуация возможна и для входных сигналов. Использование для кодирования состояния или входного сигнала различных структурных кодов делает невозможным структурный синтез автомата. Требование единственности и уникальности структурных кодов состояний и входных сигналов может быть выражено в существовании изоморфизма (16), устанавливающего, в числе прочего, биекции $A \leftrightarrow K_S(A)$ и $Z \leftrightarrow K_S(Z)$. Для обеспечения данного требования изоморфизмы (21) для всех $i \in [1, N_\delta]$ должны задаваться не независимо один от другого, а совместно.

Объединим изоморфизм (16) и множество изоморфизмов (21) в систему:

$$\begin{cases} G_\delta \leftrightarrow G_d; \\ G_{\delta_1} \leftrightarrow G_{d_1}; \\ ... \\ G_{\delta_{N_\delta}} \leftrightarrow G_{d_{N_d}}. \end{cases} \qquad (22)$$

Данная система представляет собой условие эквивалентности абстрактного и структурного автоматов в части функции переходов при условии ее представления в виде нескольких частичных функций.

**Структурная организация микропрограммного автомата, использующего представление функции переходов в виде множества частичных функций**

Рассмотрим структурную организацию МПА, отвечающего системе (22). Сопоставим каждой частичной структурной функции переходов $d_i$ отдельную комбинационную схему $\text{КС}_{d_i}$. В общем случае на вход каждой $\text{КС}_{d_i}$ поступают $R$-разрядный структурный код $T$ текущего состояния автомата и множество структурных входных сигналов $X$. На выходе каждой схемы $\text{КС}_{d_i}$ формируется значение частичной структурной функции переходов $d_i$, являющееся $R$-разрядным структурным кодом состояния перехода.

Предположим, что в каждом такте работы структурная функция $d$ принимает значение одной из частичных функций $d_i$, $i \in [1, N_d]$. Организуем выбор частичной функции переходов путем мультиплексирования, для чего закодируем функции $d_i$ кодами $K(d_i)$, представляемыми структурными сигналами $z_1$, … …, $z_{R_Z}$, где $R_Z = \lceil \log_2(N_d) \rceil$. Вектор $Z = < z_1,...$ $..., z_{R_Z} >$ используем для мультиплексирования множества $R$-разрядных значений частичных структурных функций переходов. Получаемая в результате структурная схема изображена на рис. 2. Поскольку каждая частичная функция $d_i$ рассматривается нами как операция некоторой алгебры, назовем данную схему *операционной частью* (ОЧ).

В соответствии с данной схемой структурная функция переходов $d$ определяется следующим образом:

$$d = d(d_1, \ ..., \ d_{N_d}, \ Z)\,. \qquad (23)$$

Представим операционную часть в виде отдельного блока. Структурный код состояния перехода, формируемый ОЧ, подадим на вход регистра памяти РП. Выход РП подадим в качестве обратной связи на вход ОЧ (рис. 3).

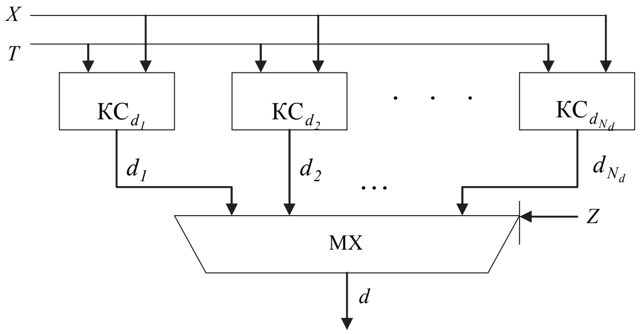


Рис. 2

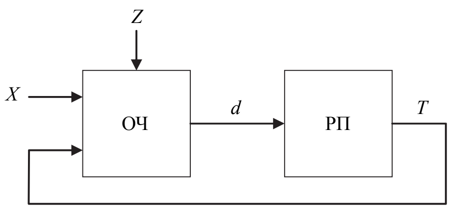


Рис. 3

Заметим, что композиция блоков ОЧ и РП может рассматриваться как операционный автомат (ОА) [2]. В сравнении с традиционной структурой данный ОА имеет следующие отличительные особенности.

• Множество функций операционного автомата представлено множеством структурных функций переходов $d_1 - d_{N_d}$.

• Память автомата представлена единственным регистром РП, выступающим в каждой выполняемой операции в роли регистра исходных данных и регистра результата.

• Роль управляющих сигналов выполняют сигналы кода операции $Z$, применяемые к мультиплексору результата MX.

• В традиционном ОА процесс обработки данных допускает возникновение различных ошибочных ситуаций (переполнение, деление на ноль и т.п.) Подобные ситуации находят отражение во множестве сигналов логических условий (ЛУ), формируемых специальным блоком и анализируемых в дальнейшем логической схемой управляющего автомата. В отличие от традиционного ОА, характеризующегося определенной универсальностью, блок ОЧ всегда проектируется для автомата с известным законом функционирования. Это позволяет построить блок ОЧ таким образом, чтобы исключить возникновение в процессе обработки кодов состояний каких-либо ситуаций, требующих анализа. Как следствие, блок ОЧ не формирует никаких признаков результата выполнения операций, и в его структуре на рис. 2 схема формирования логических условий отсутствует.

• В качестве внешних данных выступают сигналы логических условий $X$.

Поскольку роль блока ОЧ состоит в преобразовании структурных кодов состояний, то есть в реализации переходов микропрограммного автомата, назовем ОА, изображенный на рис. 3, *операционным автоматом переходов* (ОАП).

Для формирования $R_Z$-разрядного кода $Z$ будем использовать специальную $Z$-подсхему, реализующую функцию

$$Z = Z(T, X) \qquad (24)$$

и строящуюся каноническим способом по системе булевых уравнений.

Добавление в структурную схему, изображенную на рис. 3, $Z$-подсхемы и СФМО приводит к структуре МПА (рис. 4).

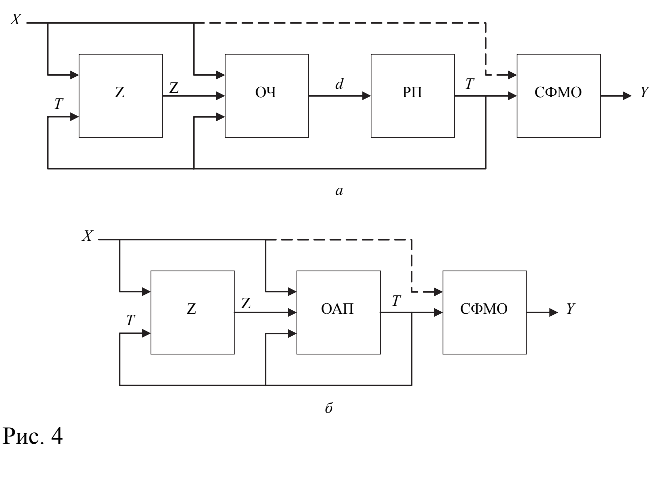


Рис. 4

На рис. 4, *а* операционный автомат переходов показан в виде композиции блоков ОЧ и РП, на рис. 4, *б* – в виде единого блока. Назовем данную структуру *микропрограммным автоматом с операционным автоматом переходов* (МПА с ОАП). Наличие в структурах на рис. 4 связи, показанной пунктиром, позволяет классифицировать их как автомат Мили, отсутствие связи – как автомат Мура. Будем считать, что МПА с ОАП задан, если задана система изоморфизмов (22), в которой $N_d > 1$.

**Пример синтеза МПА с ОАП**

В соответствии с каноническим методом структурного синтеза цифровых автоматов функция переходов задается в векторном виде системой канонических скалярных уравнений [9]. Данная система строится известным способом [8, 9], при котором для каждой пары $< K_S(a_i), K_S(z_j) >$ кортежа (15), соответствующего одному переходу автомата, формируется терм, присутствие которого в том или ином уравнении системы определяется элементом $K_S(a_k)$ данного кортежа.

Максимально возможное количество различных термов в системе уравнений функции переходов равно числу *B* переходов автомата и возрастает с увеличением *B*. Как следствие, возрастают и затраты аппаратуры в комбинационной схеме, реализующей каноническую систему уравнений функции переходов. Таким образом, если под автоматным переходом понимать некоторое преобразование информации, сложность схемы формирования переходов (рис. 1) зависит от количества выполняемых ею таких преобразований.

В то же время существуют преобразователи информации, сложность схем которых не зависит или зависит несущественно от числа выполняемых преобразований, а преобразование информации выполняется по некоторому известному закону. Примером таких устройств есть инкрементор, схема которого при *R* входных разрядах способна выполнять $2^R$ различных преобразований. Помимо инкрементора, подобными свойствами обладают функциональные блоки, выполняющие операции декремента, суммирования, умножения, сдвига, инверсии, различных логических операций, а также блоки, реализующие сочетания подобных операций.

Применение подобных блоков для реализации автоматных переходов возможно при использовании специальных методов кодирования состояний. Так, в структурной теории автоматов известен класс автоматов на счетчике или *С*-автоматов [10]. В автомате на счетчике для реализации части переходов используется схема инкрементора, для остальной части – комбинационная схема, строящаяся каноническим способом. Уменьшение аппаратурных затрат в автомате на счетчике в сравнении с автоматом с канонической структурой достигается путем того, что затраты аппаратуры в схеме инкрементора не зависят от того, какое количество переходов реализуется данной схемой.

Обобщим принцип, используемый в *С*-автомате. Введем следующие утверждения:

- Для реализации переходов могут быть использованы операции, отличные от операции инкремента.
- В одном автомате может быть использовано несколько различных операций.

Подтвердим состоятельность данных утверждений примером. Пусть микропрограммный автомат задан граф-схемой алгоритма (ГСА) Г, изображенной на рис. 5.

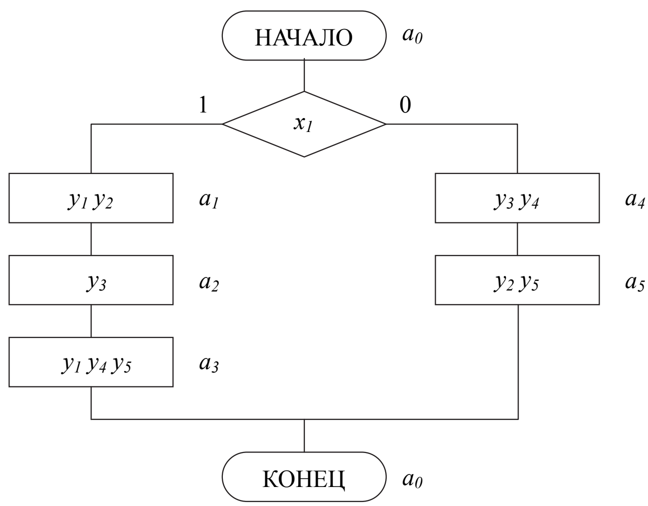


Рис. 5

Поскольку функция выходов автомата в статье не рассматривается, содержимое операторных вершин ГСА не существенно и может быть произвольным. Построим автомат, в котором переходы внутри первой последовательности реализуются схемой инкрементора, внутри второй последовательности – схемой декрементора, остальные переходы – каноническим способом.

Выделим в ГСА Г две последовательности состояний: $<a_0, a_1, a_2, a_3>$ и $<a_0, a_4, a_5>$. Закодируем состояния автомата следующим образом: $K_S(a_0) = 010_2 = 2_{10}$, $K_S(a_1) = 011_2 = 3_{10}$, $K_S(a_2) = 100_2 = 4_{10}$, $K_S(a_3) = 101_2 = 5_{10}$, $K_S(a_4) = 001_2 = 1_{10}$, $K_S(a_5) = 000_2 = 0_{10}$. Очевидно, что при данном способе кодирования все переходы внутри последовательности $<a_0, a_1, a_2, a_3>$ могут быть выполнены с помощью операции инкремента, все переходы внутри последовательности $<a_0, a_4, a_5>$ – с помощью операции декремента. Переходы из $a_3$ в $a_0$ и из $a_5$ в $a_0$ могут быть реализованы каноническим способом или какой-то иной подходящей операцией.

Таким образом, в синтезируемом автомате можно выделить три частичных структурных функции переходов: $d_1$, реализуемая операцией инкремента; $d_2$, реализуемая операцией декремента; $d_3$, реализуемая каноническим способом. Учитывая это, представим структурную схему автомата в виде (рис. 6), на котором блоки $\text{КС}_{d_1} - \text{КС}_{d_3}$ и мультиплексор MX образуют операционную часть, которая вместе с РП образует операционный автомат переходов.

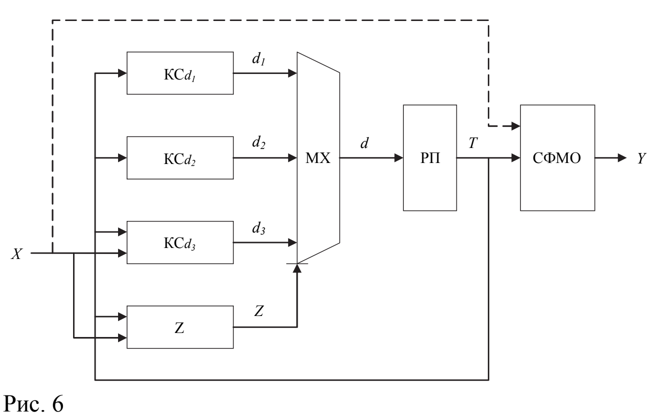

Рис. 6

Закодируем функции $d_1 - d_3$ двоичными кодами $K(d_1) = 00$, $K(d_2) = 01$, $K(d_3) = 10$. Построим таблицу истинности $Z$-подсхемы. Она включает следующие столбцы: $a_i$ – текущее состояние автомата, $T$ – структурный код $K_S(a_i)$ текущего состояния, $X$ – сигналы логических условий, анализируемые при переходе из состояния $a_i$, $Z$ – сигналы кода частичной функции переходов, реализующей переход из состояния $a_i$ по условию $X$.

**Таблица**

| $a_i$ | $a_0$ | $a_0$ | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ |
|---|---|---|---|---|---|---|---|
| $T = <T_1, T_2, T_3>$ | 0 1 0 | 0 1 0 | 0 1 1 | 1 0 0 | 1 0 1 | 0 0 1 | 0 0 0 |
| $X = \{x_1, \bar{x}_1, 1\}$ | $x_1$ | $\bar{x}_1$ | 1 | 1 | 1 | 1 | 1 |
| $Z = <Z_1, Z_2>$ | 0 0 | 0 1 | 0 0 | 0 0 | 1 0 | 0 1 | 1 0 |

Рассмотренный пример подтверждает состоятельность обобщений, сделанных выше для автомата на счетчике.

**Заключение.** Предлагаемый в статье способ представления функции переходов микропрограммного автомата в виде множества частичных функций дает потенциальную возможность снижения аппаратурных затрат в схеме МПА в сравнении с канонической реализацией. Данный эффект достигается путем использования функциональных блоков, сложность которых не зависит от количества реализуемых ими автоматных переходов. При этом схема формирования переходов МПА может быть представлена в виде операционного автомата, использование которого приводит к новой структуре микропрограммного автомата с операционным автоматом переходов. Формализация процесса структурного синтеза МПА с ОАП требует разработки соответствующих методов, использующих в своей основе специальные способы кодирования состояний. Реализация схемы МПА с ОАП в базисе современных ПЛИС позволяет задействовать встроенные функциональные блоки, характеризующиеся минимальными аппаратурными затратами при максимальном быстродействии. Решение этих и других вопросов, вызванных использованием операционного автомата переходов, не является очевидным и требует отдельных исследований.

1. *Баранов С.И.* Синтез микропрограммных автоматов. – Л.: Энергия, 1979. –232 с.
2. *Майоров С.А.*, *Новиков Г.И.* Структура электронных вычислительных машин. – Л.: Машиностроение, 1979. – 384 с.
3. *Глушков В.М.* Абстрактная теория автоматов // Успехи математических наук. – 1961. – Т. XVI. – **5**. – С. 3–59.
4. *Плоткин Б.И.*, *Гринглаз Л.Я.*, *Гварамия А.А.* Элементы алгебраической теории автоматов: Учеб. пособие для вузов. – М.: Высш. шк., 1994. – 191 с.
5. *Алгебраическая* теория автоматов, языков и полугрупп / Под ред. М. Арбиба. – М.: Статистика, 1975. – 335 с.
6. *Мальцев А.И.* Алгебраические системы. – М.: Наука, 1970. – 392 с.
7. *Энциклопедия* кибернетики: В 2 т. / Под ред. В.М. Глушкова. – Киев: Главная редакция Украинской советской энциклопедии, 1973. – Т. 2. – 608 с.
8. *Баранов С.И.*, *Скляров В.А.* Цифровые устройства на программируемых БИС с матричной структурой. – М.: Радио и связь, 1986. – 272 с.
9. *Глушков В.М.* Синтез цифровых автоматов. – М.: Физматгиз, 1962. – 476 с.
10. *Баркалов А.А.* Синтез устройств управления на программируемых логических устройствах. – Донецк: ДонНТУ, 2002. – 262 с.

Поступила 27.04.2015
Тел. для справок: +38 044 526-3069 (Киев)
*E-mail*: *cpld@mail.ru*, *a.barkalov@iie.uz.zgora.pl*


UDC 681.324

R.M. Babakov, A.A. Barkalov

## Realization of Function of Microprogram Final State Machine Transitions on the Base of Operational Final-State Machine

**Keywords:** microprogram final state machine, algebraic representation, partial function of transitions, isomorphism, operational final state machine of transitions.

The object of research is the microprogram final-state machine with data path of transitions. In digital devices, the microprogram final-state machine performs the functions of control unit and coordinates the work of the other units of the system.

The microprogram final-state machine can be formally represented as a polybasic algebra whose carrier set is formed by the sets of states, input and output signals, and the signature – by function of transitions and function of outputs. Mutual independence of these functions allows to consider them separately. This leads to two algebras – algebra of transitions and algebra of outputs, which differ for the abstract and structural final-state machines. Equivalence of abstract and structural final-state machines expressed in isomorphisms of abstract and structural algebras of transitions and abstract and structural algebras of outputs.

Split of the set of transitions into several disjoint subsets is equal to representation of the function of transitions as a set of partial functions of transitions. Each partial function of transitions may form signature of a separate subalgebra of transitions. Thus, it may be obtained a collection of abstract subalgebras of transitions and isomorphic structural subalgebras of transitions.

The mathematical model, in which equivalence of the abstract and structural final state machines is presented as isomorphisms of abstract and structural subalgebras of transitions, can be interpreted by microprogram final-state machine with datapath of transitions. The basic structure of transitions datapath is a collection of combinational circuits, each of them implements a different transitions subset. Selecting one of the combinational circuits results is performed via the multiplexer controlled by signals, generated by a special circuit.

The logic circuit of microprogram final-state machine with datapath of transitions may have fewer hardware expenses compared with the canonical structure of microprogram final-state machine. A prerequisite is the use for the realization of transitions partial functions of such functional blocks whose complexity does not depend on the number of realizing transitions. Such circuit is, for example, incrementor used in the known structures of microprogram final-state machine with counter. In this respect, the structure of microprogram final-state machine with datapath of transitions is a generalization pof final-state machine with counter. The mathematical model proposed in this paper is the basis for development of the final-state machine microprogram methods synthesis with the transitions datapath.

●