

УДК: 519.715

Голубева О. И.

к.т.н., доцент

Томский государственный университет

## ИНТЕРВАЛЬНЫЕ РАСШИРЕНИЯ БУЛЕВЫХ ФУНКЦИЙ И ТРОИЧНОЕ МОДЕЛИРОВАНИЕ ПОСЛЕДОВАТЕЛЬНОСТНЫХ СХЕМ

*В работе рассматривается понятие интервального расширения булевой функции, которое позволяет определить значения функции на интервалах булевых векторов. Задача точного троичного моделирования комбинационных и синхронных последовательностных схем может рассматриваться как задача вычисления значений интервальных расширений булевых функций и логических автоматов, соответственно. Представлены общие правила точного пошагового троичного моделирования синхронных последовательностных схем на основе функций переходов в состояния. На этих правилах основан ранее предложенный метод троичного моделирования по ДНФ представлению функций. Метод в дальнейшем был переложен на ROBDD представление функций. Кроме того, в данной работе ранее предложенный метод точного пошагового троичного моделирования по ROBDD представлению функций преобразуется к более универсальному виду.*

**Ключевые слова:** интервальное расширение булевой функции, функции переходов в состояния, точное троичное моделирование, пошаговое моделирование, синхронная последовательностная схема, ROBDD графы.

Golubeva O. I.

Ph.D., Associate Professor

Tomsk State University

## INTERVAL EXTENSIONS OF BOOLEAN FUNCTIONS AND THREE-VALUE SIMULATION OF SEQUENTIAL CIRCUITS

*In this paper, the notion of the interval extension of a Boolean function is considered, which allows determining the values of function on intervals of Boolean vectors. The task of exact three-value simulation of combinational and sequential circuits can be considered as a task of calculating the values of interval extensions of Boolean functions and logical automata, correspondingly. The general rules of exact step-by-step three-value simulation of synchronous sequential circuits based on the transition functions to states are presented. The previously proposed method of three-value simulation for DNF representation of functions is based on these rules. The method was later converted into the ROBDD representation of the functions. In addition, in this paper, the previously proposed method of exact step-by-step three-value simulation using the ROBDD representation of functions is transformed to a more universal form.*

**Key words:** interval extension of a Boolean function, the transition functions to states, exact three-value simulation, step-by-step simulation, sequential circuits, ROBDD.

Рассмотрим моделирование синхронных последовательностных схем на не полностью определенных входных последовательностях и не полностью определенном векторе начального состояния. Значением неопределенного входа (псевдовхода) схемы может быть как 1, так и 0; обозначим значение такого входа (псевдовхода) символом  $\times$ .

Будем называть псевдовыходами синхронной последовательностной схемы входы триггеров, а псевдовходами схемы – выходы триггеров.

Для формальной постановки задачи троичного моделирования в работе [1] вводятся понятия *интервального расширения булевой функции* и *интервального расширения логического автомата*.

В работе [1] рассматриваются как полностью определенные, так и частичные булевы функции и логические автоматы. Здесь для краткости приведем только понятие интервального расширения полностью определенной булевой функции.

Обозначим через  $E^f$  область определения функции  $f$ . Назовем функцию  $f(x_1, x_2, \dots, x_n)$  *полностью определенной булевой функцией*, если  $E^f = E_n^2$ , и *частичной булевой функцией*, если  $E^f \subset E_n^2$ , где  $E^2 = \{0, 1\}$  и  $E_n^2 = \underbrace{E^2 \times E^2 \times \dots \times E^2}_{n \text{ раз}}$ .

Далее рассмотрим полностью определенную булеву функцию  $f(x_1, x_2, \dots, x_n)$ .

Обозначим через  $M_1^f = \{\gamma \in E_n^2 : f(\gamma) = 1\}$  и  $M_0^f = \{\gamma \in E_n^2 : f(\gamma) = 0\}$ . Очевидно, что  $M_1^f \cup M_0^f = E_n^2$ . Будем множество  $M_1^f$  называть областью единичных значений функции  $f$ , а множество  $M_0^f$  – областью нулевых значений функции  $f$ .

Будем в дальнейшем  $\times$  рассматривать как символ, обозначающий любой из элементов множества  $\{0, 1, \times\}$ . Будем называть значения 1 и 0 *определенными*, а значение  $\times$  – *неопределенным*.

Вектор  $\alpha = (\alpha_0, \dots, \alpha_n)$ , компоненты которого принимают свои значения из множества  $\{0, 1, \times\}$ , будем называть *троичным вектором*.

Назовем *троичной функцией* – функцию  $f(x_1, x_2, \dots, x_n)$ , аргументы которой определены на множестве  $E^\times = \{0, 1, \times\}$ , и такую, что  $f(\alpha_1, \alpha_2, \dots, \alpha_n) \in E^\times$ , когда  $\alpha_i \in E^\times$ ,  $i = \overline{1, n}$ . Обозначим  $E_n^\times = \underbrace{E^\times \times E^\times \times \dots \times E^\times}_{n \text{ раз}}$ .

Введем на множестве  $E^\times = \{0, 1, \times\}$  отношение частичного порядка следующим образом:  $0 \leq \times$ ,  $1 \leq \times$ , 1 и 0 не сравнимы.

*Троичные векторы*  $\alpha = (\alpha_1, \dots, \alpha_n)$  и  $\beta = (\beta_1, \dots, \beta_n)$  находятся в отношении  $\alpha \leq \beta$ , если  $\alpha_i \leq \beta_i$ ,  $i = \overline{1, n}$ .

Множество булевых векторов размерности  $n$ , являющееся декартовым произведением  $N_1 \times N_2 \times \dots \times N_n$ , где  $N_i$ ,  $i = \overline{1, n}$ , – одно из трех множеств  $\{0\}$ ,  $\{1\}$  или  $\{0, 1\}$ , называется *интервалом булевых векторов*.

Интервал  $N'$  булевых векторов размерности  $n$  называется *минимальным покрывающим интервалом множества булевых векторов*  $M \subseteq E_n^2$ , если  $M \subseteq N'$ , и не существует интервала булевых векторов  $N''$  такого, что  $M \subseteq N'' \subset N'$ .

Троичный вектор  $\alpha = (\alpha_1, \dots, \alpha_n)$  представляет интервал булевых векторов  $N(\alpha)$ , такой что  $N(\alpha) = \{\gamma \in E_n^2 : \gamma \leq \alpha\}$ .

**Определение.** Назовем *интервальным расширением полностью определенной булевой функции*  $f(x_1, \dots, x_n)$  ( $E^f = E_n^2$ ) такую полностью определенную троичную функцию  $f^\times(x_1, \dots, x_n)$  ( $E^{f^\times} = E_n^\times$ ), что значение  $f^\times(X)$  на троичном векторе  $\alpha \in E_n^\times$  определяется следующим образом:

- 1)  $f^\times(\alpha) = 1$ , если для  $\forall \gamma \in N(\alpha): f(\gamma) = 1$  ( $N(\alpha) \subseteq M_1^f$ );
- 2)  $f^\times(\alpha) = 0$ , если для  $\forall \gamma \in N(\alpha): f(\gamma) = 0$  ( $N(\alpha) \subseteq M_0^f$ );

3)  $f^\times(\alpha) = \times$ , если  $\exists \gamma_i \in N(\alpha): f(\gamma_i) = 1$  и  $\exists \gamma_j \in N(\alpha): f(\gamma_j) = 0$  ( $N(\alpha) \cap M_1^f \neq \emptyset$  и  $N(\alpha) \cap M_0^f \neq \emptyset$ ).

Проиллюстрируем значение интервального расширения булевой функции  $f^\times(x_1, \dots, x_n)$  на троичном векторе  $\alpha = (\alpha_1, \dots, \alpha_n)$  на диаграмме.

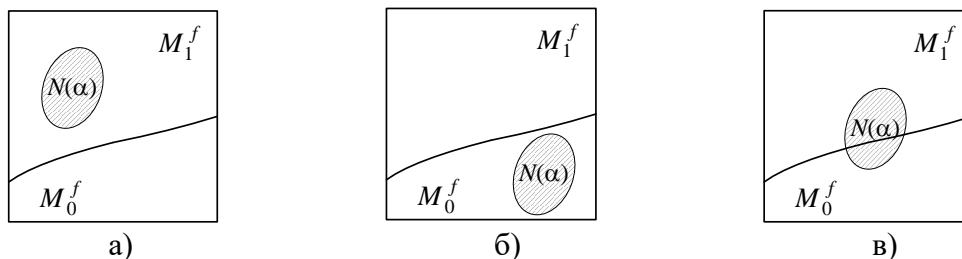


Рисунок 1. – а)  $f^\times(\alpha) = 1$ , б)  $f^\times(\alpha) = 0$ , в)  $f^\times(\alpha) = \times$

На рисунке 1 прямоугольником представлено булево пространство размерности  $n$ . Оно разделено на области  $M_1^f$  и  $M_0^f$ . Интервал  $N(\alpha)$  может полностью содержаться в области единичных (нулевых) значений, как на рисунке 1а (1б), в этом случае  $f^\times(\alpha) = 1(0)$  или же может пересекаться с обеими областями, как на рисунке 1в, тогда  $f^\times(\alpha) = \times$ .

В работе [1] задача точного троичного моделирования комбинационной схемы на троичном векторе рассматривается как задача вычисления значений интервальных расширений булевых функций, реализуемых выходами схемы. Задача точного троичного моделирования синхронной схемы на входной последовательности троичных векторов рассматривается как задача вычисления реакции интервального расширения логического автомата, являющегося функциональной моделью схемы. Такое представление задачи моделирования является формально корректным, но более сложным для изложения и понимания, особенно в случае последовательностных схем, поэтому в данной работе будем использовать менее формальные конструкции.

Далее будем говорить: “троичное значение булевой функции  $f(X)$  на троичном векторе  $\alpha$ ”, имея в виду троичное значение интервального расширения булевой функции  $f(X)$  –  $f^\times(X)$ , на троичном векторе  $\alpha$ . Будем иногда говорить: “троичное моделирование булевой функции  $f(X)$  на троичном векторе  $\alpha$ ”, имея в виду вычисление троичного значения  $f^\times(\alpha)$ .

Рассмотрим троичное моделирование синхронных последовательностных схем. Пусть  $f$  – булева функция, реализуемая выходом (псевдовыходом) комбинационного эквивалента длины  $k$  последовательностной схемы,  $\alpha$  – входная троичная последовательность длины  $k$ ,  $\delta$  – троичный вектор начального состояния. Будем называть метод троичного моделирования схемы *адекватным*, если он на всех  $\alpha$  и  $\delta$  дает истинное значение булевой функции  $f(\alpha, \delta)$  из  $\{0, 1, \times\}$  или значение  $\times$  вместо истинного значения 1 или 0. В дальнейшем будем рассматривать только адекватные методы троичного моделирования, поэтому слово “адекватный” будем опускать. *Троичное моделирование схемы является точным*, если при моделировании вычисляются точные значения  $f(\alpha, \delta)$  на всех  $\alpha$  и  $\delta$ . При *неточном (приближенном)* троичном моделировании схемы вместо определенного значения 1 или 0 на выходе (псевдовыходе) может быть получено неопределенное значение  $\times$ .

Неточные методы троичного моделирования позволяют получить значение  $\beta$  на выходе (псевдовыходе) такое что  $f(\alpha, \delta) \leq \beta$ .

На практике часто используются методы неточного троичного моделирования, так как они, как правило, более быстродействующие, требуют меньших вычислительных затрат и памяти. Тем не менее, неточные методы могут приводить к ухудшению качества решений для одних задач и не позволяют решить некоторые другие задачи.

Троичное моделирование логических схем может быть использовано при решении различных проблем диагностирования [1, 3-5].

В частности, троичное моделирование логических схем может применяться при построении тестовых последовательностей. Моделирование является одним из основных средств при построении тестов для схем с памятью. Моделирование может сочетаться с регулярными методами построения тестов, которые позволяют строить тесты для труднообнаружимых неисправностей схемы. Сочетание регулярных методов с моделированием дает возможность эффективно строить тесты для схем большой сложности [1]. Неточность троичного моделирования ведет к ухудшению качества тестовой последовательности, построенной на основе такого моделирования.

В работах [1, 2] предложен метод точного пошагового троичного моделирования синхронных последовательностных схем на входных последовательностях произвольной длины, основанный на использовании функций переходов в состояния схемы. В работах [1, 2] методы используют ДНФ представление функций переходов-выходов схемы. В работах [3, 4] предложенный метод был переложен на ROBDD представление функций. В работах [5, 6] выполнено дальнейшее исследование метода, рассматриваются решения задач с его применением и возможность использования неточных результатов при выполнении точного троичного моделирования. В данной работе представлены идеи, положенные в основу разработанного в работах [1, 2] метода для ДНФ представления функций, которые могут быть применены при разработке методов точного пошагового троичного моделирования синхронных последовательностных схем для любого представления функций переходов-выходов. Показано, что эти идеи реализованы в методе точного троичного моделирования, основанного на ROBDD представлении функций переходов-выходов. Кроме того, в данной работе процедуры троичного моделирования на основе ROBDD преобразованы к более универсальному виду по сравнению с ранее предложенными процедурами.

Рассмотрим синхронную последовательностную схему с  $n$  входами,  $m$  выходами и  $p$  линиями обратных связей.  $X = \{x_1, \dots, x_n\}$  – множество входных переменных схемы,  $Y = \{y_1, \dots, y_m\}$  – множество ее выходных переменных,  $Z = \{z_1, \dots, z_p\}$  – множество внутренних переменных схемы.

Система функций переходов-выходов схемы имеет вид:

$$\begin{aligned} y_i &= \Phi_i(x_1, \dots, x_n, z_1, \dots, z_p), & i &= \overline{1, m}, \\ z_j &= \Psi_j(x_1, \dots, x_n, z_1, \dots, z_p), & j &= \overline{1, p}. \end{aligned} \quad (1)$$

Необходимо выполнить точное троичное моделирование схемы при заданных входной троичной последовательности  $\alpha^0, \alpha^1, \dots, \alpha^{k-1}$  и троичном векторе начального состояния  $\delta^0$  и получить троичную последовательность значений внутренних переменных –  $\delta^1, \delta^2, \dots, \delta^k$  и троичную последовательность выходных значений –  $\beta^0, \beta^1, \dots, \beta^{k-1}$ .

Назовем объединение входного и внутреннего состояний схемы его *полным состоянием*. Будем представлять полное состояние схемы троичным вектором  $(\alpha, \delta)$ . Здесь  $\alpha$  – вектор значений входных переменных  $X$ , а  $\delta$  – вектор значений внутренних переменных  $Z$ .

Назовем *пошаговым вычислением* выходной троичной последовательности схемы  $\beta^{*0}, \beta^{*1}, \dots, \beta^{*k-1}$  и троичной последовательности значений внутренних переменных  $\delta^{*1}, \delta^{*2}, \dots, \delta^{*k}$  при заданной входной троичной последовательности  $\alpha^0, \alpha^1, \dots, \alpha^{k-1}$  и троичном векторе начального состояния  $\delta^0$  вычисление, выполняемое следующим образом. Значения  $\beta^{*0}$  и  $\delta^{*1}$  получаем, выполняя моделирование на троичном векторе  $(\alpha^0, \delta^0)$  по функциональному или структурному описанию комбинационного эквивалента схемы длины один, значения  $\beta^{*1}$  и  $\delta^{*2}$  получаем, выполняя моделирование на троичном векторе  $(\alpha^1, \delta^{*1})$ , и так далее. Пошаговое

моделирование, в общем случае, является неточным, то есть  $\beta^i \leq \beta^{*i}$  и  $\delta^j \leq \delta^{*j}$ ,  $i = \overline{0, k-1}$ ,  $j = \overline{1, k}$ .

Получая на каждом шаге точные значения функций переходов и выходов, тем не менее, возможна потеря точности на последовательности в целом. Покажем, каким образом происходит потеря точности.

Пусть выполнено точное троичное моделирование функций переходов системы (1) на векторе  $(\alpha^0, \delta^0)$ , представляющем полное состояние, и получен вектор значений внутренних переменных  $\delta^{*1}$ .  $\delta^{*1}$  представляет минимальный покрывающий интервал множества булевых векторов значений переменных  $Z$ , а не точное множество этих векторов. Таким образом, множество достижимых за один шаг из множества  $N(\alpha^0, \delta^0)$  состояний схемы может быть подмножеством множества  $N(\delta^{*1})$ . На следующем шаге моделирования будет получен вектор  $\delta^{*2}$ , представляющий состояния, достижимые из полных состояний  $N(\alpha^1, \delta^{*1})$ , то есть из множества, расширенного по отношению к множеству достижимых на предыдущем шаге состояний. Здесь уже вектор выходных значений  $\beta^{*2}$  также может оказаться неточным, т.е.,  $\beta^2 \leq \beta^{*2}$ . Таким образом, с каждым шагом моделирования количество неверно вычисленных неопределенных компонент может возрастать.

Для того чтобы на каждом шаге находить точное множество состояний, в которое может попасть схема из заданного множества полных состояний, в работах [1, 2] предлагается выполнять троичное моделирование с использованием *функций переходов в состоянии*. Рассмотрим эти функции, их построение и свойства.

Каждому двоичному вектору  $\tau^i = (\tau_1^i, \dots, \tau_p^i)$  размерности  $p$  значений внутренних переменных множества  $Z$  соответствует состояние  $q_i$  схемы. Будем вектор  $\tau^i = (\tau_1^i, \dots, \tau_p^i)$  называть *кодом состояния*  $q_i$  в переменных  $Z$ . Таким образом,  $Q = \{q_1, \dots, q_t\}$  – множество состояний схемы, где  $t = 2^p$  – мощность множества  $Q$ .

Сопоставим каждому состоянию  $q_i \in Q$  переменную  $z'_i$ . Получим множество переменных  $Z' = \{z'_1, \dots, z'_t\}$ .

Кодом состояния  $q_i$  в переменных множества  $Z'$  является булев вектор размерности  $t$  –  $\gamma^i = (\gamma_1^i, \dots, \gamma_t^i)$ , в котором  $i$ -ая компонента принимает значение 1, а все остальные компоненты – значение 0.

Пусть  $\delta'$  – троичный вектор значений переменных множества  $Z'$ , и  $r$  – число компонент вектора  $\delta'$  со значением  $\times$ . Троичный вектор  $\delta'$  представляет множество  $N(\delta') = \{\gamma^1, \dots, \gamma^r\}$  состоящее из  $r$  булевых векторов каждый из которых является кодом состояния  $q_{i_s}$  в переменных  $Z'$  и содержит единственную компоненту со значением 1. Таким образом, вектор  $\delta'$  представляет  $r$  состояний схемы, номера состояний соответствуют номерам компонент вектора  $\delta'$  со значением  $\times$ .

Заметим, что  $N(\alpha, \delta') = N(\alpha) \times N(\delta')$ , где  $N(\delta')$  определяется по специальному вышеописанному правилу.

### Пример.

Пусть схема имеет две внутренние переменные  $Z = \{z_1, z_2\}$ . Тогда  $p = 2$  и  $t = 2^p = 4$ .  $Z' = \{z'_1, z'_2, z'_3, z'_4\}$ .

Рассмотри состояние  $q_2$ , его код в переменных  $Z'$  –  $(0, 1, 0, 0)$ .

Рассмотрим троичный вектор  $\delta' = (\times, 0, \times, 0)$  значений переменных множества  $Z'$ .  $N(\delta') = \{\gamma^1, \gamma^3\} = \{(1, 0, 0, 0), (0, 0, 1, 0)\}$ .  $\delta' = (\times, 0, \times, 0)$  представляет два состояния:  $q_1$  и  $q_3$ .

Заметим, что троичный вектор  $\delta'$  в переменных множества  $Z'$ , содержащий одну,  $s$ -ую, компоненту со значением  $\times$ , а остальные компоненты со значением 0, задает тоже

множество состояний, что и троичный вектор  $\delta''$ , у которого  $s$ -ая компонента принимает значение 1, а остальные – значение 0. Будем в дальнейшем считать эти векторы эквивалентными и иногда не будем отдельно описывать случай троичного вектора с одной единицей. Тогда между множествами состояний и троичными векторами значений переменных  $Z'$  существует взаимно однозначное соответствие.

В дальнейшем будем полное состояние схемы представлять троичным вектором  $(\alpha, \delta)$ , где  $\delta$  – вектор в пространстве внутренних переменных  $Z$  или  $Z'$ .

Пусть  $\zeta(X, Z)$  – произвольная булева функция, зависящая от переменных множества  $X$  и  $Z$  (например, функция  $\varphi_i$  или  $\psi_j$ ). Определим зависимость функции  $\zeta(X, Z)$  от переменных множества  $Z'$  следующим образом:  $\zeta(\alpha, \gamma) = \zeta(\alpha, \tau)$ , где  $\alpha$  – булев вектор значений входных переменных,  $\gamma$  – булев вектор значений внутренних переменных множества  $Z'$ ,  $q$  – состояние схемы с кодом  $\gamma$  в переменных множества  $Z'$ , а  $\tau$  – код во внутренних переменных множества  $Z$  состояния  $q$ . Таким образом, функцию  $\zeta(X, Z)$  будем рассматривать также как частичную функцию  $\zeta(X, Z')$ , где переменные множества  $Z'$  определены на вышеописанных векторах значений. Переход к переменным множества  $Z'$  для функции  $\zeta(X, Z)$  в явном виде не выполняется. Будем для функции  $\zeta(X, Z)$  в качестве значений внутренних переменных использовать вектор  $\delta$  в переменных множества  $Z$  или вектор  $\delta'$  в переменных множества  $Z'$ . В первом случае рассматривается функция  $\zeta(X, Z)$ , во втором – функция  $\zeta(X, Z')$ .

Троичное значение функции  $\zeta(X, Z')$  на троичном векторе полного состояния  $(\alpha, \delta')$  определяется по общему вышеописанному правилу: если на любом булевом векторе из множества  $N(\alpha, \delta')$  функция  $\zeta(X, Z')$  принимает значение 1(0), то  $\zeta(X, Z') = 1(0)$ , иначе  $\zeta(X, Z') = \times$ .

Получим функции  $\psi'_j(x_1, \dots, x_n, z_1, \dots, z_p)$ ,  $j = \overline{1, t}$ , представляющие переходы в состояния схемы.

**Процедура 1** получения функции  $\psi'_j(x_1, \dots, x_n, z_1, \dots, z_p)$ ,  $j \in \{1, \dots, t\}$ , перехода в состояние  $q_j$  схемы.

Получим функцию  $\psi'_j$ , представляющую переходы в состояние  $q_j$ , перемножением  $p$  функций  $\psi_s$  или их инверсий в соответствии с кодом состояния в переменных множества  $Z$ . Пусть  $\tau^j = (\tau_1^j, \dots, \tau_p^j)$  – код состояния  $q_j$  в переменных  $Z$ . Если  $\tau_s^j = 1$ ,  $s \in \{1, \dots, p\}$ , то в произведение входит  $\psi_s$ , если  $\tau_s^j = 0$ , то в произведение входит  $\bar{\psi}_s$ .

В работах [1, 2] предложен метод точного пошагового троичного моделирования синхронных последовательностных схем для ДНФ представления функций. В работе [1] представлены формальные доказательства корректности метода. Здесь представим идеи, на которых основан предложенный в работах [1, 2] метод и его доказательства, а именно:

- 1) правила, обеспечивающие точное моделирование;
- 2) правила сокращения вычислений;
- 3) организацию процесса вычисления.

**Правила 1** точного пошагового троичного моделирования синхронных последовательностных схем.

1. При моделировании используются функции перехода в состояния  $\psi'_j$ ,  $j = \overline{1, t}$ , полученные по Процедуре 1, которые позволяют находить точное множество состояний, в которое может перейти схема за один шаг из заданного множества полных состояний.

2. Пусть  $\zeta(X, Z')$  – функция, зависящая от входных переменных множества  $X$  и внутренних переменных множества  $Z'$  (например, функция  $\varphi_i$  или  $\psi'_j$ ). Необходимо

вычислить значение функции на троичном векторе  $N(\alpha, \delta')$ .

$N(\alpha, \delta') = N(\alpha) \times N(\delta') = \{N(\alpha, \gamma^1), \dots, N(\alpha, \gamma^r)\}$ , поэтому вычислить значение  $\zeta(\alpha, \delta')$  можно, вычислив значение функции на каждом векторе из множества  $\{N(\alpha, \gamma^1), \dots, N(\alpha, \gamma^r)\}$ . Поскольку  $\gamma^i$  – это булев вектор, то  $\zeta(X, \gamma^i)$  зависит от переменных множества  $X$ . Пусть  $\zeta(X, \gamma^i) = \xi^i(X)$ . Вычислить значение  $\xi^i(X)$  на троичном векторе  $\alpha$  можно любым методом точного троичного моделирования полностью определенной булевой функции. Следовательно,  $\zeta(\alpha, \delta')$  можно вычислить по следующему правилу:

если  $\forall \gamma^i \in N(\delta') : \xi^i(\alpha) = 1$ , то  $\zeta(\alpha, \delta') = 1$ ; если  $\forall \gamma^i \in N(\delta') : \xi^i(\alpha) = 0$ , то  $\zeta(\alpha, \delta') = 0$ ; иначе  $\zeta(\alpha, \delta') = \times$ .

Заметим, что представленное правило позволяет остановить вычисление и присвоить  $\zeta(\alpha, \delta')$  значение  $\times$ , как только выяснится, что: 1)  $\exists \gamma^i \in N(\delta') : \xi^i(\alpha) = \times$  или 2)  $\exists \gamma^{i_1} \in N(\delta') : \xi^{i_1}(\alpha) = 1$  и  $\exists \gamma^{i_2} \in N(\delta') : \xi^{i_2}(\alpha) = 0$ .

### Правила 2 сокращения вычислений.

1. Для упрощения вычисления  $\xi^i(\alpha) = \zeta(\alpha, \gamma^i)$ , формула функции  $\zeta(X, Z')$  строится таким образом, чтобы для каждого состояния  $q_i$  функции  $\xi^i(X) = \zeta(X, \gamma^i)$  были представлены отдельной подформулой (подграфом при графовом представлении функций и т.д.). После выбора метода точного троичного моделирования полностью определенной булевой функции для вычисления троичного значения  $\xi^i(\alpha)$ , эти подформулы преобразуются к требуемому представлению (например, ОДНФ, сокращенная ДНФ и т.д. при работе с ДНФ представлением функций переходов-выходов).

Такое представление получаем для функций выходов, назовем их –  $\phi'_i$ , и функций переходов в состояния –  $\psi'_j$ .

2. При моделировании функций переходов в состояния принимается во внимание свойство: компонента  $\delta''_s$  вектора значений внутренних переменных множества  $Z' - \delta''$ , принимает значение 1 тогда и только тогда, когда все остальные компоненты вектора  $\delta''$ , кроме  $s$ -ой, принимают значение 0.

Это свойство позволяет при вычислении вектора  $\delta''$  значений функций переходов в состояния для каждой функции переходов определять не точное значение, а лишь то, равно оно 0 или отлично от 0. Если компонента вектора  $\delta''$  не равна 0, то присваиваем ей значение  $\times$ . После вычисления всех компонент вектора  $\delta''$  корректируем результат следующим образом: если в векторе  $\delta''$   $s$ -ая компонента принимает значение  $\times$ , а все остальные компоненты – значение 0, тогда компоненте  $\delta''_s$  присваиваем значение 1.

Представленное свойство позволяет остановить вычисление и присвоить  $\psi'_j(\alpha, \delta')$  значение  $\times$ , как только выяснится, что  $\exists \gamma^{i_1} \in N(\delta') : \psi'(\alpha, \gamma^{i_1}) = \times$  или  $\exists \gamma^{i_2} \in N(\delta') : \psi'(\alpha, \gamma^{i_2}) = 1$ .

**Процесс вычисления 1** с использованием функций выходов  $\phi'_i$ ,  $i = \overline{1, m}$ , и функций переходов в состояния  $\psi'_j$ ,  $j = \overline{1, t}$ .

1. Начальное состояние схемы представлено троичным вектором  $\delta^0$  в переменных  $Z$  (или подмножеством состояний  $Q^0 \subseteq Q$ ). Чтобы выполнить точное троичное моделирование схемы, используя функции переходов в состояния, необходимо перейти от троичного вектора  $\delta^0$  в переменных  $Z$  (от  $Q^0$ ) к троичному вектору  $\delta'^0$  в переменных  $Z'$ . Процедура такого перехода предложена в [1] (а также представлена в [4]).

2. Для точного троичного моделирования функций выходов  $\varphi'_i$ ,  $i = \overline{1, m}$ , и переходов в состояния схемы  $\psi'_j$ ,  $j = \overline{1, t}$ , на входной троичной последовательности  $\alpha^0, \alpha^1, \dots, \alpha^{k-1}$  длины  $k$  и векторе начального состояния  $\delta^0$  выполняется пошаговое моделирование. На каждом шаге вычисляется значение каждой функции  $\varphi'_i$  и  $\psi'_j$  по процедурам, реализующим п. 2 Правил 1 моделирования и Правила 2 сокращения вычислений. В результате получим последовательность векторов значений на выходах  $\beta^0, \beta^1, \dots, \beta^{k-1}$  и внутренних переменных множества  $Z'$ :  $\delta^1, \delta^2, \dots, \delta^k$ .

3. Получив последовательность  $\delta^1, \delta^2, \dots, \delta^k$  троичных векторов значений внутренних переменных множества  $Z'$ , можно перейти к последовательности  $\delta^1, \delta^2, \dots, \delta^k$  троичных векторов значений внутренних переменных множества  $Z$ . Процедура такого перехода предложена в [1] (а также представлена в [4]).

Представленные идеи можно применять при разработке методов точного пошагового троичного моделирования синхронных последовательностных схем при использовании любого представления функций переходов-выходов. При этом разные функциональные представления несколько по-своему реализуют представленные идеи и могут позволить добавить свои особые дополнительные возможности сокращения вычислений. Кроме того, на базе представленных правил можно разрабатывать модификации.

Разработанный в работах [1, 2] на основе описанных идей метод точного пошагового троичного моделирования последовательностных схем для ДНФ представления функций, в работах [3–6] был переложен на ROBDD представление функций. В представленных в работах [3–6] методах для вычисления троичного значения булевой функции, зависящей от переменных множества  $X$ , использовался вероятностный подход. В данной работе представим более универсальные процедуры моделирования без привязки к конкретному методу троичного моделирования булевой функции.

Обозначать через  $G(f)$  ROBDD представление функции  $f$ .

Получим функции  $\varphi'_i(x_1, \dots, x_n, z_1, \dots, z_p)$  и  $\psi'_j(x_1, \dots, x_n, z_1, \dots, z_p)$ ,  $j = \overline{1, t}$ , представляющие переходы в состояния схемы, в виде ROBDD графов.

Система ROBDD функций выходов –  $G(\varphi'_i)$ , и переходов в состояния схемы –  $G(\psi'_j)$ , имеет вид:

$$y_i = G(\varphi'_i(x_1, \dots, x_n, z_1, \dots, z_p)), \quad i = \overline{1, m}, \quad (2)$$

$$z'_j = G(\psi'_j(x_1, \dots, x_n, z_1, \dots, z_p)), \quad j = \overline{1, t}. \quad (3)$$

**Процедура 2** получения систем (2) и (3).

1. Получим каждую функцию переходов-выходов в виде ROBDD. При получении ROBDD для функций  $\varphi_i$ ,  $i = \overline{1, m}$ , и  $\psi_j$ ,  $j = \overline{1, p}$ , сначала выполним декомпозицию Шеннона по внутренним переменным множества  $Z$ , а затем по входным переменным множества  $X$ . Полученные таким образом ROBDD функций  $\varphi_i$ ,  $i = \overline{1, m}$ , обозначим через  $G(\varphi'_i)$ .

2. Для каждого состояния  $q_j$  схемы получим ROBDD  $G(\psi'_j)$ , представляющий переходы в состояние, перемножением  $p$  ROBDD функций  $\psi_s$  или их инверсий в соответствии с кодом состояния в переменных множества  $Z$  как показано в Процедуре 1.

3. ROBDD  $G(\varphi'_i(X, Z))$  и  $G(\psi'_j(X, Z))$  зависят от внутренних переменных множества  $Z$ , а состояниям в системе (3) сопоставлены переменные множества  $Z'$ .

Зависимость ROBDD  $G(\varphi'_i)$  и  $G(\psi'_j)$  от переменных множества  $Z'$  определяется вышеописанным образом. Таким образом, ROBDD  $G(\varphi'_i(x_1, \dots, x_n, z_1, \dots, z_p))$  и



$G(\psi'_j(x_1, \dots, x_n, z_1, \dots, z_p))$  будем рассматривать также как ROBDD  $G(\phi'_i(x_1, \dots, x_n, z'_1, \dots, z'_i))$  и  $G(\psi'_j(x_1, \dots, x_n, z'_1, \dots, z'_i))$  частичных функций  $\phi'_i(X, Z')$  и  $\psi'_j(X, Z')$ . Переход к переменным множества  $Z'$  в ROBDD  $G(\phi'_i)$  и  $G(\psi'_j)$  в явном виде не выполняется. Будем для функций  $G(\phi'_i)$  и  $G(\psi'_j)$  в качестве значений внутренних переменных использовать вектор  $\delta$  в переменных множества  $Z$  или вектор  $\delta'$  в переменных множества  $Z'$ . В первом случае рассматриваются функции  $G(\phi'_i(X, Z))$  и  $G(\psi'_j(X, Z))$ , во втором – функции  $G(\phi'_i(X, Z'))$  и  $G(\psi'_j(X, Z'))$ .

При выполнении п.2. Процедуры 2 будем иметь ввиду, что ROBDD инверсии функции можно получить из ROBDD функции заменой значений терминальных вершин на противоположные.

Здесь будем использовать те же сокращения выражений, что и в работе [5], введенные для краткости изложения.

Пусть при получении ROBDD  $G(\zeta(X, Z))$  функции  $\zeta(X, Z)$  сначала выполнена декомпозиция Шеннона по внутренним переменным множества  $Z$ , а затем по входным переменным множества  $X$ .

Каждой цепи  $\pi$  ROBDD  $G(\zeta(X, Z))$  из внутренних переменных множества  $Z$ , исходящей из корня, соответствует троичный вектор  $\delta^\pi$  значений переменных множества  $Z$  такой что: если цепь проходит через вершину, сопоставленную переменной  $z_s$ , по значению  $1(0)$ , то  $\delta_s^\pi = 1(0)$ , если цепь не проходит через вершину, сопоставленную  $z_s$ , то  $\delta_s^\pi = \times$ .

Будем говорить, что цепь  $\pi$  ROBDD  $G(\zeta(X, Z))$  из переменных множества  $Z$ , исходящая из корня и приводящая в корень подграфа, состоящего из переменных множества  $X$  или в терминальную вершину, соответствует двоичному вектору  $\tau$  значений переменных  $Z$ , если  $\tau \in N(\delta^\pi)$ . Любому двоичному вектору  $\tau$  значений переменных  $Z$  соответствует только одна такая цепь. Будем эту цепь обозначать как  $\pi^\tau$ .

Будем говорить, что цепь ROBDD  $G(\zeta(X, Z))$  из переменных множества  $Z$ , исходящая из корня, соответствует состоянию  $q$  (представляет состояние  $q$ ), если она соответствует коду состояния  $q$  в переменных  $Z$ . Будем такую цепь обозначать как  $\pi^q$ .

Заметим, что при разложении сначала по переменным множества  $Z$  функции  $\zeta(X, Z)$  при получении ROBDD,  $G(\zeta(X, \tau^i))$  – это подграф  $G(\zeta(X, Z))$  из входных переменных, в корень которого приводит цепь  $\pi^{\tau^i}$ . Как было показано выше,  $G(\zeta(X, \tau^i)) = G(\zeta(X, \gamma^i))$ . То есть, для полученных по Процедуре 2 ROBDD  $G(\phi'_i)$  и  $G(\psi'_j)$  выполняется п.1 Правил 2 сокращения вычислений.

Обозначим через  $G_{ji}^z(X)$  ( $G_{ji}^y(X)$ ) ROBDD из входных переменных, в который приводит цепь  $\pi^{q_i}$ , исходящая из корня  $G(\psi'_j)$  ( $G(\phi'_j)$ ) и соответствующая состоянию  $q_i$ . Отметим, что  $G_{ji}^z(X) = G(\psi'_j(X, \tau^i))$  ( $G_{ji}^y(X) = G(\phi'_j(X, \tau^i))$ ), где  $\tau^i$  – код состояния  $q_i$ . Если цепь приводит в терминальную вершину 0, то  $G_{ji}^z(X) \equiv 0$  ( $G_{ji}^y(X) \equiv 0$ ). Если цепь приводит в терминальную вершину 1, то  $G_{ji}^z(X) \equiv 1$  ( $G_{ji}^y(X) \equiv 1$ ). Переход в состояние  $q_j$  схемы возможен из состояния  $q_i$  при каких-либо значениях входных переменных за один шаг, если  $G_{ji}^z(X) \neq 0$ .  $G_{ji}^z(X)$  представляет все векторы значений входных переменных при которых такой переход возможен. На выходе  $j$  схемы возможно получить значение 1 из состояния  $q_i$  схемы при каких-либо значениях входных переменных за один шаг, если  $G_{ji}^y(X) \neq 0$ ;  $G_{ji}^y(X)$  представляет все векторы значений входных переменных при которых будет получено

значение 1 из состояния  $q_i$ . Для двух различных состояний  $q_{i_1}$  и  $q_{i_2}$   $G_{j_1}^z(X) (G_{j_1}^y(X))$  может совпадать с  $G_{j_2}^z(X) (G_{j_2}^y(X))$ .

Будем в дальнейшем обозначать через  $G_j^z(X) (G_j^y(X))$  множество  $G_{j_i}^z(X) (G_{j_i}^y(X))$  для всех  $q_i \in Q$ . Через  $G_{ji}(X)$  будем обозначать множество  $\{G_{ji}^z(X), G_{ji}^y(X)\}$ .

Рассмотрим процедуры вычисления троичного вектора точных значений на выходах  $\beta$  и троичного вектора точных значений внутренних переменных  $\delta''$  на троичном векторе  $(\alpha, \delta')$ , где  $\delta'$  и  $\delta''$  – векторы в переменных  $Z'$ . Процедуры вычисляют точные троичные значения для отдельных компонент векторов.

Опишем процедуры вычисления без привязки к конкретному методу точного троичного моделирования полностью определенных булевых функций  $G_{ji}(X)$ , представленных в виде ROBDD.

Каждой вершине  $w$  ROBDD  $G(\varphi'_i)$  и  $G(\psi'_j)$ , соответствующей переменной из множества  $X$ , сопоставим значение  $\times(w)$ , в котором при вычислении троичного значения функции  $G(\varphi'_i)$  или  $G(\psi'_j)$  на векторе  $(\alpha, \delta')$  будет содержаться троичное значение булевой функции, сопоставленной вершине  $w$ , на векторе  $\alpha$ , если оно уже вычислено, и неопределенное значение, если – нет.

**Процедура 1\*** вычисления точного значения компоненты  $\beta_i$ ,  $i \in \{1, \dots, m\}$ , вектора выходных значений  $\beta$  на векторе  $(\alpha, \delta')$  по ROBDD  $G(\varphi'_i)$  системы (2) ( $\beta_i = \varphi'_i(\alpha, \delta')$ ).

1. Пусть  $\delta'$  представляет множество состояний схемы  $Q' = \{q_{i_1}, \dots, q_{i_r}\}$ . Пусть  $\tau' = \{\tau^{i_1}, \dots, \tau^{i_r}\}$  – коды состояний из множества  $Q'$  в переменных  $Z$ .
2. Поочередно просматриваем двоичные векторы  $\tau^{i_s}$ ,  $s = \overline{1, r}$ .

Проходим по соответствующей вектору  $\tau^{i_s}$  цепи ROBDD  $G(\varphi'_i)$  из переменных  $Z$ , исходящей из корня. Цепь соответствующая  $\tau^{i_s}$  приводит в корень подграфа, состоящего из переменных множества  $X$ ,  $G_{j_s}^y(X)$ . Если значение  $G_{j_s}^y(\alpha)$  не было вычислено при просмотре предыдущих цепей, то вычисляем это значение. Если  $G_{j_s}^y(\alpha) = \times$ , то  $\beta_i$  присвоим значение  $\times$  ( $\varphi'_i(\alpha, \delta') = \times$ ), процедура завершена.

Если уже пройдена цепь  $\tau^{i_{s_1}}$ , такая, что  $G_{j_{s_1}}^y(\alpha) = 1$  и цепь  $\tau^{i_{s_2}}$ , такая, что  $G_{j_{s_2}}^y(\alpha) = 0$ , то  $\beta_i$  присвоим значение  $\times$  ( $\varphi'_i(\alpha, \delta') = \times$ ), процедура завершена.

3. Если  $\forall \tau^{i_s} \in \tau': G_{j_s}^y(\alpha) = 0$ , то  $\beta_i = 0$  ( $\varphi'_i(\alpha, \delta') = 0$ ); если  $\forall \tau^{i_s} \in \tau': G_{j_s}^y(\alpha) = 1$ , то  $\beta_i = 1$  ( $\varphi'_i(\alpha, \delta') = 1$ ).

**Процедура 2\*** вычисления точного значения компоненты  $\delta''_j$ ,  $j \in \{1, \dots, t\}$ , вектора следующего состояния  $\delta''$  на троичном векторе  $(\alpha, \delta')$  по ROBDD  $G(\psi'_j)$  системы (3) ( $\delta''_j = \psi'_j(\alpha, \delta')$ ).

1. Пусть  $\delta'$  представляет множество состояний схемы  $Q' = \{q_{i_1}, \dots, q_{i_r}\}$ . Пусть  $\tau' = \{\tau^{i_1}, \dots, \tau^{i_r}\}$  – коды состояний из множества  $Q'$  в переменных  $Z$ .
2. Поочередно просматриваем двоичные векторы  $\tau^{i_s}$ ,  $s = \overline{1, r}$ .

Проходим по соответствующей вектору  $\tau^{i_s}$  цепи ROBDD  $G(\psi'_j)$  из переменных  $Z$ , исходящей из корня. Цепь соответствующая  $\tau^{i_s}$  приводит в корень подграфа, состоящего из переменных множества  $X$ ,  $G_{j_i_s}^z(X)$ . Если значение  $G_{j_i_s}^z(\alpha)$  не было вычислено при просмотре предыдущих цепей, то вычисляем это значение. Если  $G_{j_i_s}^z(\alpha) = \times$  или  $G_{j_i_s}^z(\alpha) = 1$ , то  $\delta''_j$  присвоим значение  $\times$  ( $\psi'_j(\alpha, \delta') = \times$ ), процедура завершена.

3.  $\delta''_j$  присвоим значение 0 ( $\psi'_j(\alpha, \delta') = 0$ ).

Если после вычисления всех компонент вектора  $\delta''$  Процедурой 2\* только одна компонента принимает значение  $\times$ , то этой компоненте присвоим значение 1.

**Заметим**, что п. 2 в Процедурах 1\* и 2\* не подразумевает нескольких обходов графа для просмотра различных цепей. Поочередный просмотр цепей выполняется за один обход графа.

В Процедурах 1\* и 2\* троичное значение  $G_{ji}(X)$  на троичном векторе  $\alpha$  можно вычислить любым методом точного троичного моделирования полностью определенной булевой функции, представленной ROBDD графом, например, методом, использующим вероятностный подход.

Троичный вектор  $\delta''$  представляет множество состояний схемы, в которое схема может перейти за один шаг, из множества состояний, представленного троичным вектором  $\delta'$ , подавая на вход схемы булевы векторы из множества, представленного входным троичным вектором  $\alpha$ .

Как отмечается в работах [4, 5], если при моделировании Процедурами 1\* и 2\* в качестве вектора входных значений  $\alpha$  взять вектор из значений  $\times$ , то троичный вектор  $\delta''$  представляет максимальное множество состояний схемы  $Q'' = \{q_{i_1}, \dots, q_{i_s}\}$ , в которое схема может перейти за один шаг из множества состояний  $Q' = \{q_{i_1}, \dots, q_{i_r}\}$ , представленного троичным вектором  $\delta'$ , при любом векторе входных значений.

В работах [4, 5] представлены процедуры вычисления троичного вектора значений на выходах  $\beta$  и троичного вектора значений внутренних переменных  $\delta''$  на троичном векторе полного состояния  $(\alpha, \delta')$ , где все компоненты вектора  $\alpha$  принимают значение  $\times$ . Ограничение на значения вектора  $\alpha$  позволяют сократить объем вычислений, основываясь на следующих наблюдениях: 1) если  $G_{ji}(X) \neq 0(1)$ , то  $G_{ji}(\alpha) \neq 0(1)$ , 2) если  $G_{ji}(X) \neq 0$  и  $G_{ji}(X) \neq 1$ , то  $G_{ji}(\alpha) = \times$ .

**Процедура 1'** вычисления точного значения компоненты  $\beta_i$ ,  $i \in \{1, \dots, m\}$ , вектора выходных значений  $\beta$  на векторе  $(\alpha, \delta')$  по ROBDD  $G(\phi'_i)$  системы (2) ( $\beta_i = \phi'_i(\alpha, \delta')$ ), где все компоненты вектора  $\alpha$  принимают значение  $\times$

1. Пусть  $\delta'$  представляет множество состояний схемы  $Q' = \{q_{i_1}, \dots, q_{i_r}\}$ . Пусть  $\tau' = \{\tau^{i_1}, \dots, \tau^{i_r}\}$  – коды состояний из множества  $Q'$  в переменных  $Z$ .

2. Поочередно просматриваем двоичные векторы  $\tau^{i_s}$ ,  $s = \overline{1, r}$ .

Проходим по соответствующей вектору  $\tau^{i_s}$  цепи ROBDD  $G(\phi'_i)$  из переменных  $Z$ , исходящей из корня. Цепь соответствующая  $\tau^{i_s}$  приводит в корень подграфа, состоящего из переменных множества  $X$ ,  $G_{j_i_s}^y(X)$ . Если  $G_{j_i_s}^y(X) \neq 1$  ( $G_{j_i_s}^y(X)$  – не терминальная вершина 1) и  $G_{j_i_s}^y(X) \neq 0$  ( $G_{j_i_s}^y(X)$  – не терминальная вершина 0), то  $\beta_i$  присвоим значение  $\times$  ( $\phi'_i(\alpha, \delta') = \times$ ), процедура завершена.

Если уже пройдена цепь  $\tau^{i_{s1}}$ , такая, что  $G_{j_{i_{s1}}}^y(X) \equiv 1$  и цепь  $\tau^{i_{s2}}$ , такая, что  $G_{j_{i_{s2}}}^y(X) \equiv 0$ , то  $\beta_i$  присвоим значение  $\times (\varphi'_i(\alpha, \delta') = \times)$ , процедура завершена.

3. Если  $\forall \tau^{i_s} \in \tau': G_{j_{i_s}}^y(X) \equiv 0$ , то  $\beta_i = 0 (\varphi'_i(\alpha, \delta') = 0)$ ; если  $\forall \tau^{i_s} \in \tau': G_{j_{i_s}}^y(\alpha) \equiv 1$ , то  $\beta_i = 1 (\varphi'_i(\alpha, \delta') = 1)$ .

**Процедура 2'** вычисления значения компоненты  $\delta''_j$ ,  $j \in \{1, \dots, t\}$ , вектора  $\delta''$  на троичном векторе  $(\alpha, \delta')$  по ROBDD  $G(\psi'_j)$  системы (3) ( $\delta''_j = \psi'_j(\alpha, \delta')$ ), где все компоненты вектора  $\alpha$  принимают значение  $\times$ .

1. Пусть  $\delta'$  представляет множество состояний схемы  $Q' = \{q_{i_1}, \dots, q_{i_r}\}$ . Пусть  $\tau' = \{\tau^{i_1}, \dots, \tau^{i_r}\}$  – коды состояний из множества  $Q'$  в переменных  $Z$ .

2. Поочередно просматриваем двоичные векторы  $\tau^{i_s}$ ,  $s = \overline{1, r}$ .

Проходим по соответствующей вектору  $\tau^{i_s}$  цепи ROBDD  $G(\psi'_j)$  из переменных множества  $Z$ , исходящей из корня. Цепь соответствующая  $\tau^{i_s}$  приводит в корень подграфа, состоящего из переменных множества  $X$ ,  $G_{j_{i_s}}^z(X)$ . Если  $G_{j_{i_s}}^z(X) \neq 0$  ( $G_{j_{i_s}}^z(X)$  – не терминальная вершина 0), то  $\delta''_j$  присвоим значение  $\times (\psi'_j(\alpha, \delta') = \times)$ , процедура завершена (при необходимости запоминаем состояние  $q_{i_s}$  в качестве одного из состояний множества  $Q'$ , из которого есть переход в состояние  $q_j$  за один шаг).

3.  $\delta''_j$  присвоим значение 0 ( $\psi'_j(\alpha, \delta') = 0$ ).

Точное пошаговое троичное моделирование синхронной последовательностной схемы на троичном векторе  $(\alpha, \delta')$ , где все компоненты вектора  $\alpha$  принимают значение  $\times$ , может применяться при решении различных задач. Например, с использованием Процедуры 2', в работе [5] решается задача построения последовательности входных векторов минимальной длины, обеспечивающей переход из некоторого состояния одного множества состояний схемы в одно из состояний другого множества, в работах [4, 5] также приводятся примеры других задач, решаемых с применением Процедуры 2'.

В работе [6] рассматривается использование результатов неточного метода троичного моделирования схемы с целью сокращения вычислительных затрат связанных с реализацией точного метода троичного моделирования. Целесообразность использования неточных результатов может зависеть от используемого метода точного троичного моделирования для вычисления значений  $G_{ji}(X)$  на троичном векторе  $\alpha$ .

Итак, в представленных процедурах вычисления троичных значений функций выходов  $\varphi'_i$  и переходов в состояния  $\psi'_j$ , представленных в виде ROBDD, реализованы п. 2 Правил 1 точного пошагового троичного моделирования последовательностных схем и Правила 2 сокращения вычислений ранее предложенные для ДНФ представления функций.

Точное пошаговое троичное моделирование последовательностной схемы на входной троичной последовательности  $\alpha^0, \alpha^1, \dots, \alpha^{k-1}$  и векторе начального состояния  $\delta^0$  с использованием Процедур 1\* и 2\* (1' и 2') выполняется по Процессу вычисления 1.

### Литература

1. Голубева, О. И. Разработка и исследование методов моделирования и оценки мер тестопригодности логических схем: дис. ... канд. техн. наук: 05.13.01. – Томск, 2000. – 112 с.
2. Голубева, О. И. Коррекция результатов троичного моделирования на последовательностях произвольной длины // Докл. 2-й всероссийской конф. “Новые

информационные технологии в исследовании дискретных структур”. Екатеринбург. 2-5 нояб. 1998. – Екатеринбург : ИМАШ УрО РАН, 1998. – С. 62-68.

3. Голубева, О. И. Троичное моделирование синхронных последовательностных схем, основанное на ROBDD представлении функций // Кулагинские чтения: техника и технологии производственных процессов: XVI Междунар. науч.-практ. конф.: сб. ст. [в 3 ч.] / Забайкал. гос. ун-т. Чита. 28-30 ноября 2016. – Чита: ЗабГУ, 2016. – Ч. 2. – С. 131-136.

4. Голубева, О. И. Троичное моделирование синхронных схем с памятью на последовательностях произвольной длины, основанное на ROBDD представлении функций // Таврический научный обозреватель. — 2017. — №3 (20). – С. 129 – 140.

5. Голубева, О. И. Троичное моделирование последовательностной схемы и построение на его основе входных последовательностей, обеспечивающих заданные переходы схемы // Таврический научный обозреватель. — 2017. — №4 (21).

6. Голубева, О. И. Использование неточных результатов для сокращения вычислительных затрат точного троичного моделирования последовательностных схем // Таврический научный обозреватель. — 2017. — №4 (21).