

УДК 519.713:681.3

Д.А. Кочанов

ПОСТРОЕНИЕ ПОСЛЕДОВАТЕЛЬНОСТЕЙ ДЛЯ ТЕСТИРОВАНИЯ D-ТРИГГЕРОВ СИНХРОННЫХ ПОСЛЕДОВАТЕЛЬНОСТНЫХ СХЕМ

Предлагается метод тестирования последовательностных схем, элементами памяти которых служат D-триггеры. Метод основывается на инициализации триггера и продвижении установленного в триггере значения на один из внешних полюсов схемы. Для выполнения инициализации предлагается оригинальный метод поиска установочной последовательности входных воздействий.

Введение

Тестирование синхронных последовательностных схем (элементами памяти таких схем служат триггеры) в настоящее время является актуальной задачей проектирования цифровых сверхбольших интегральных схем (СБИС). Существуют два подхода к решению этой проблемы: к первому относятся методы, основанные на построении автоматной модели последовательностной схемы [1], ко второму – структурные методы построения тестов [2, 3]. Развитием структурных методов являются генетические алгоритмы [2], которые используют псевдослучайный поиск для построения некоторого близкого к оптимальному решению.

В основе первых методов лежит проверка на эквивалентность конечных автоматов (в литературе показано, что конечный автомат является математической моделью последовательностной схемы) [5]. Напомним, что два конечных автомата эквивалентны, если их функции выходов совпадают на каждом из допустимых наборов входных воздействий. Существуют различные способы решения этой задачи, но все они не позволяют найти решение за приемлемое время для автоматов, число состояний которых превышает несколько сотен. Простейшим видом последовательностной схемы является схема, элементами памяти которой служат только D-триггеры. Автоматная модель такой схемы содержит от 2^n до 2^{2n} состояний, где n – число триггеров [2] (в работе рассматривается модель константных неисправностей выходных полюсов D-триггера). Поскольку современные промышленные схемы содержат до 1000 триггеров и выше, методы этого класса не получили широкого практического использования.

В основе структурных методов построения тестов лежит поиск проверяющей (тестовой) последовательности. Многие современные АГТП-алгоритмы (АГТП – автоматическая генерация тестовых последовательностей) для решения этой проблемы используют различные методы, основанные на моделировании неисправности в многозначных алфавитах и состоящие из следующих основных этапов [2, 3]:

1) активации (sensitization), при которой выполняется поиск тестовой последовательности, обеспечивающей возникновение требуемой неисправности на соответствующем выходе D-триггера тестируемой схемы;

2) D-распространения (D-propagation), при котором влияние заданной неисправности распространяется до одного из внешних выходов схемы, что обеспечивает наблюдаемость влияния неисправности;

3) доопределения (justification), при котором определяются значения внешних входов, обеспечивающие требуемые состояния внутренних линий и выходов тестируемой схемы;

4) импликации, которая используется для снятия неопределенностей на линиях схемы; последние могут возникать в результате присвоения на предыдущих этапах некоторым линиям определенных значений.

В статье предложен метод тестирования D-триггеров последовательностных схем. Метод состоит из двух этапов: инициализации триггера и продвижения установленного в триггере состояния на один из выходных полюсов схемы. В отличие от известных методов (алгоритма Рота, методов PODEM, FAN и др. [2, 3]), которые для активации неисправностей используют механизм моделирования состояний схемы в многозначных алфавитах (что позволяет обнаружить

неисправности const 0, const 1), предлагаемый метод для поиска инициирующей последовательности основывается на представлении функций возбуждения триггеров в виде дизъюнктивной нормальной формы (ДНФ) [5] и элиминации (исключения) переменных, соответствующих выходным полюсам тестируемого триггера.

В основе вторых методов (распространение неисправности на выходной полюс схемы) лежит моделирование неисправности в многозначном алфавите. Этот этап состоит из последовательного выполнения процедур D-распространения, доопределения и импликации. Для моделирования состояний неисправной схемы используется 16-значный алфавит V_{16} (нижний индекс обозначает число символов алфавита) [2].

1. Выделение функций возбуждения элементов памяти

Рассмотрим схему EX (рис 1). Для упрощения на рисунке к триггерам V , W , Y не подведен сигнал синхронизации.

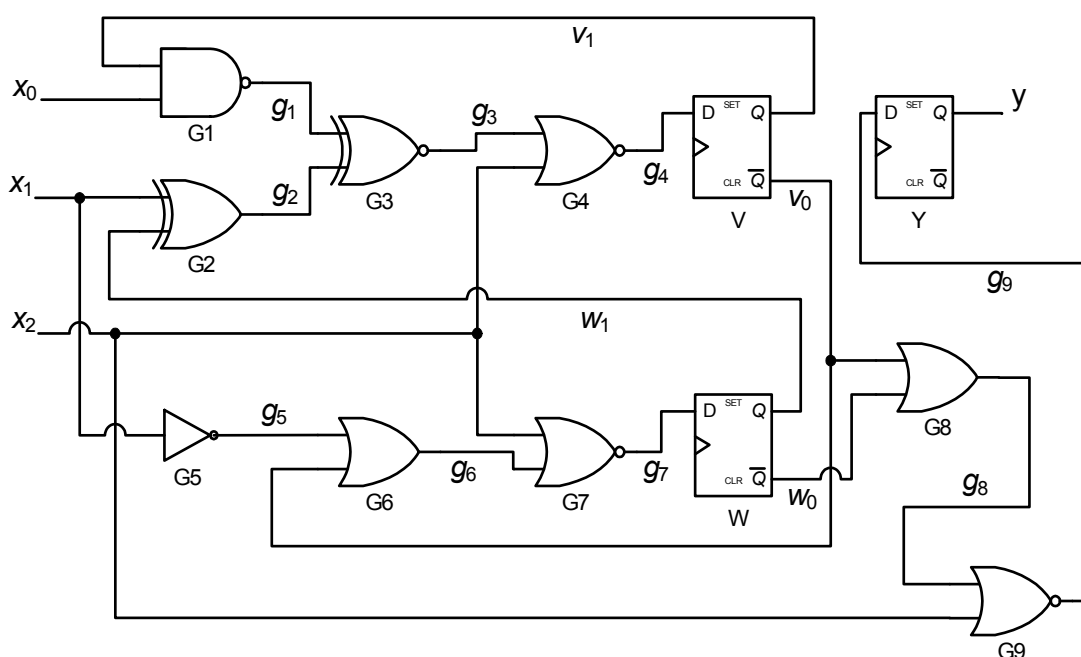
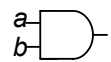
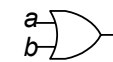
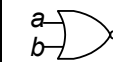
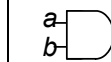
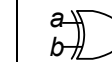
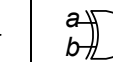
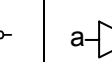


Рис. 1. Последовательностная схема EX

Схема EX синтезирована в библиотеке логических элементов базисных матричных кристаллов [6], которая в настоящее время используется при проектировании отечественных заказных цифровых СБИС. Описания библиотечных элементов, вошедших в состав схемы EX, приведены в табл. 1.

Таблица 1

Логические элементы и функции, которые они реализуют

Элемент							
Функция	$y = ab$	$y = a \vee b$	$y = \overline{a \vee b}$	$y = \overline{ab}$	$y = a \oplus b$	$y = a \sim b$	$y = \overline{a}$
Вентиль	AND	OR	NOR	NAND	XOR	XNOR	NOT

Под переменной состояния триггера будем понимать состояние его выходного информационного полюса на заданном такте. Каждой переменной состояния соответствует функция возбуждения элемента памяти, которая позволяет вычислить значение этой переменной на заданном такте работы схемы. Чтобы на основе структурного описания схемы получить функцию для вычисления значения переменной состояния некоторого триггера, необходимо выделить в

исходной последовательностной схеме комбинационную подсхему, которая оказывает влияние на входной информационный порт этого триггера на данном такте. Соответствующие подсхемы для триггеров схемы EX показаны на рис. 2.

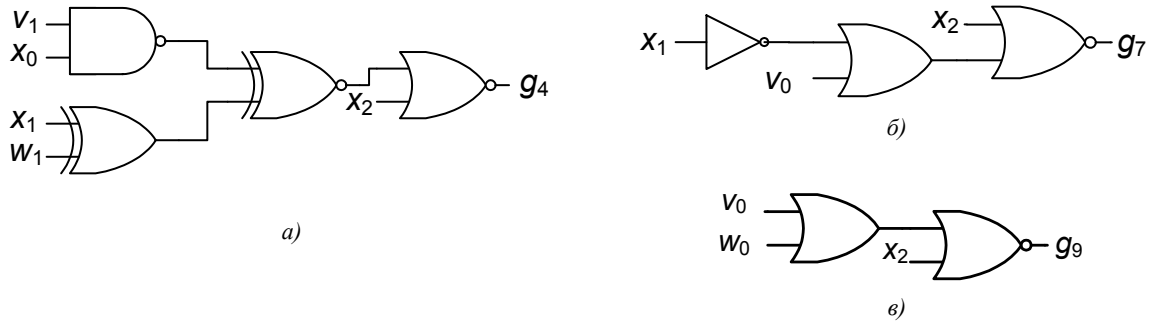


Рис. 2. Комбинационные подсхемы, соответствующие функциям возбуждения триггеров V, W и Y схемы EX: а) LOG1; б) LOG2; в) LOG3

Подсхемы LOG1, LOG2, LOG3 реализуют булевы функции g_4 , g_7 , g_9 возбуждения триггеров V, W, Y схемы EX:

$$g_4 = \overline{\overline{x_0 v_1} \sim (x_1 \oplus w_1)} \vee x_2; \quad (1)$$

$$g_7 = \overline{x_1 \vee v_0 \vee x_2}; \quad (2)$$

$$g_9 = \overline{v_0 \vee w_0 \vee x_2}. \quad (3)$$

2. Поиск инициализирующей последовательности

Инициализация триггера заключается в поиске последовательности входных воздействий, обеспечивающей одно и то же значение функции возбуждения тестируемого триггера на заданном такте как в исправной, так и в неисправной схемах. Чтобы найти входную последовательность, обеспечивающую инициализацию в состояние логической единицы триггера Z_i произвольной последовательностной схемы, необходимо решить систему уравнений

$$f_i^j(x_n^j, \dots, x_m^j, f_p^{j-1}, \dots, f_q^{j-1}) = 1, \quad (4)$$

где f_i^j – функция активации триггера Z_i . Функция f_i^j зависит от входных сигналов x_n^j, \dots, x_m^j на j -м такте и функций $f_p^{j-1}, \dots, f_q^{j-1}$ активации триггеров Z_p, \dots, Z_q на $(j-1)$ -м такте работы схемы.

Для решения системы (4) относительно x_n^j, \dots, x_m^j достаточно представить ее левую часть в виде ДНФ и в полученном представлении выполнить элиминацию внутренних переменных $f_p^{j-1}, \dots, f_q^{j-1}$ для $j=1, 2, \dots$, пока не будут найдены элементарные конъюнкции [5], не зависящие от $f_p^{j-1}, \dots, f_q^{j-1}$. Найденные элементарные конъюнкции будут определять интервалы, содержащие искомые последовательности входных воздействий. Эти последовательности обеспечивают инициализацию $f_i^j = 1$.

Проиллюстрируем данный подход на примере схемы EX. Предположим, что неисправен триггер V и выполним поиск последовательности, обеспечивающей инициализацию $g_4 = 1$. Для этого запишем функции возбуждения триггеров (1), (2) с учетом такта n и влияния вносимой триггером V неисправности:

$$g_4^n = \overline{\overline{x_0^n v_1^n} \sim (x_1^n \oplus g_7^{n-1})} \vee x_2^n; \quad (5)$$

Выражение (10) содержит две элементарные конъюнкции, не содержащие переменных состояния v_0^1, v_1^2 , – это $\overline{x_0^2 x_1^1 x_1^2 x_2^2}$ и $\overline{x_0^2 x_1^2 x_2^1 x_2^2}$. Они определяют два интервала ($x_0^1=0, x_0^2=0, x_1^1=0, x_1^2=0, x_2^1=0, x_2^2=0$) и ($x_0^1=0, x_0^2=0, x_1^1=0, x_1^2=0, x_2^1=1, x_2^2=0$), которые задают входные последовательности, обеспечивающие инициализацию $g_4=1$ за два такта. Для поиска последовательности, осуществляющей инициализацию $g_4=0$, достаточно выполнить аналогичные действия для $\overline{g_4}$.

3. Многозначный алфавит B_{16}

При моделировании схемы, содержащей неисправность, приходится отдельно различать состояния исправной и неисправной схемы. Чтобы этого избежать, применяют многозначные алфавиты, которые позволяют на одной модели оценить поведение как исправной, так и неисправной схем. Одним из простейших многозначных алфавитов является алфавит B_4 , или алфавит Рота, который представляет собой множество из четырех символов $B_4=\{0, 1, D, D'\}$. Это значит, что при моделировании в алфавите B_4 сигнал на линии может принимать одно из этих значений. Символы 0 и 1 обозначают, что значения сигнала в исправной и неисправной схемах совпадают и равны 0 и 1 соответственно. Символы D (D') служат для обозначения рассогласования сигналов: если в исправной схеме сигнал равен 1, а в неисправной 0, то значение этого сигнала в алфавите B_4 равно D , иначе D' .

На практике возникают ситуации, когда моделирование в алфавите B_4 не позволяет найти тестовую последовательность для конкретной неисправности, и в то же время тестовая последовательность существует. Примеры таких схем можно найти в работах [2, 3, 7]. Использование при моделировании алфавитов большей значности, как правило, позволяет обнаружить более широкий класс неисправностей.

В настоящей статье предлагается использовать 16-значный алфавит B_{16} [2]. Каждый символ этого алфавита является подмножеством алфавита B_4 . При моделировании неисправности в алфавите B_{16} сигнал на линии может принимать одно из тех значений алфавита B_4 , которые входят в присвоенный данной линии символ алфавита B_{16} (табл. 2). Для кодирования символов алфавита использованы четыре двоичные переменные $x^0, x^{D'}, x^D, x^1$. Каждой из переменных соответствует один из символов алфавита B_4 . Если двоичной переменной присвоено значение 1, то соответствующий данной переменной символ алфавита B_4 входит в кодируемый символ алфавита B_{16} .

Таблица 2

Универсальный алфавит тестирования B_{16}

Элемент алфавита B_{16}	Подмножество B_4	Кодирование			
		x^0	$x^{D'}$	x^D	x^1
\emptyset	$\{\emptyset\}$	0	0	0	0
1	$\{1\}$	0	0	0	1
D	$\{D\}$	0	0	1	0
$G1$	$\{D, 1\}$	0	0	1	1
D'	$\{D'\}$	0	1	0	0
$F1$	$\{D', 1\}$	0	1	0	1
D^*	$\{D', D\}$	0	1	1	0
$D1$	$\{D', D, 1\}$	0	1	1	1
0	$\{0\}$	1	0	0	0
C	$\{0, 1\}$	1	0	0	1
$F0$	$\{0, D\}$	1	0	1	0
H	$\{0, D, 1\}$	1	0	1	1
$G0$	$\{0, D'\}$	1	1	0	0
E	$\{0, D', 1\}$	1	1	0	1
$D0$	$\{0, D', D\}$	1	1	1	0
U	$\{0, D', D, 1\}$	1	1	1	1

4. Импликация и доопределение

После этапа инициализации осуществляется продвижение установленного в триггере состояния на один из внешних полюсов схемы. Для этого вначале выбирается путь активации, состоящий из цепочки логических элементов схемы, в которой первым элементом является тестируемый триггер, а выход последнего служит выходным полюсом схемы. Затем выполняется процедура D-распространения, которая сводится к последовательному присвоению символа D^* каждому из выходных полюсов тех логических элементов, которые входят в выбранный путь активации.

После применения к очередному логическому элементу процедуры D-распространения к нему последовательно применяются процедуры доопределения и импликации. Для выполнения воспользуемся моделью кодирования алфавита B_{16} при помощи четырех характеристических функций $f^0(a, b), f^{D'}(a, b), f^D(a, b), f^1(a, b)$, $a, b \in B_{16}$ [8]. Каждая из функций является двоичной и зависит от переменных $a, b \in B_{16}$, содержащих значения входных полюсов того логического элемента, к которому применяется соответствующая процедура. Характеристические функции позволяют вычислять значения двоичных переменных $x^0, x^{D'}, x^D, x^1$, последние в совокупности определяют код символа выходного полюса логического элемента в алфавите B_{16} . Правила для вычисления характеристических функций элементов AND, OR, NOT, NAND, NOR, XOR и XNOR приведены в табл. 3 [2].

Таблица 3

Характеристические функции для базовых логических элементов

Вентиль	Характеристическая функция	Правило для вычисления характеристической функции		Характеристическая функция	Вентиль
AND	$f^0(a, b)$	$a^0 \vee b^0 \vee a^{D'}b^D \vee a^Db^{D'}$		$f^1(a, b)$	NAND
	$f^{D'}(a, b)$	$a^{D'}b^1 \vee a^1b^{D'} \vee a^{D'}b^{D'}$		$f^D(a, b)$	
	$f^D(a, b)$	$a^Db^1 \vee a^1b^D \vee a^Db^D$		$f^{D'}(a, b)$	
	$f^1(a, b)$	a^1b^1		$f^0(a, b)$	
OR	$f^0(a, b)$	a^0b^0		$f^1(a, b)$	NOR
	$f^{D'}(a, b)$	$a^{D'}b^0 \vee a^0b^{D'} \vee a^{D'}b^{D'}$		$f^D(a, b)$	
	$f^D(a, b)$	$a^Db^0 \vee a^0b^D \vee a^Db^D$		$f^{D'}(a, b)$	
	$f^1(a, b)$	$a^1 \vee b^1 \vee a^{D'}b^D \vee a^Db^{D'}$		$f^0(a, b)$	
XOR	$f^0(a, b)$	$a^0b^0 \vee a^1b^1 \vee a^{D'}b^D \vee a^Db^{D'}$		$f^1(a, b)$	XNOR
	$f^{D'}(a, b)$	$a^{D'}b^1 \vee a^1b^{D'} \vee a^{D'}b^0 \vee a^0b^{D'}$		$f^D(a, b)$	
	$f^D(a, b)$	$a^Db^1 \vee a^1b^D \vee a^Db^0 \vee a^0b^D$		$f^{D'}(a, b)$	
	$f^1(a, b)$	$a^1b^0 \vee a^0b^1 \vee a^{D'}b^{D'} \vee a^Db^D$		$f^0(a, b)$	
NOT	$f^0(a, b)$	a^1	a^0	$f^1(a, b)$	NOT
	$f^{D'}(a, b)$	a^D	$a^{D'}$	$f^D(a, b)$	

Доопределение заключается в получении значения выходного полюса логического элемента при заданных значениях его входных полюсов. Рассмотрим процедуру доопределения на примере вентиля $G1$ схемы EX. Вентиль реализует функцию NAND. Пусть его входные полюсы v_1 и x_0 приняли значения $v_1=D, x_0=C$. Для выполнения доопределения определим (см. табл. 2) коды символов D и C : $D = (0, 0, 1, 0)$, $C = (1, 0, 0, 1)$. Отсюда запишем двоичные векторы $(a^0, a^{D'}, a^D, a^1)=(0, 0, 1, 0)$ и $(b^0, b^{D'}, b^D, b^1)=(1, 0, 0, 1)$, определяющие значения переменных характеристических функций. Подставив значения переменных в правила для вычисления характеристических функций элемента NAND (см. табл. 3), найдем двоичный код, определяющий значение в алфавите B_{16} выходного полюса g_1 вентиля $G1$ при заданных значениях входных воздействий:

$$\begin{aligned}
 f^0 &= a^1b^1 = (0 \wedge 1) = 0, \\
 f^{D'} &= a^{D'}b^1 \vee a^1b^{D'} \vee a^{D'}b^{D'} = (1 \wedge 1) \vee (0 \wedge 0) \vee (1 \wedge 0) = 1, \\
 f^D &= a^Db^1 \vee a^1b^D \vee a^Db^D = (0 \wedge 1) \vee (0 \wedge 0) \vee (0 \wedge 0) = 0, \\
 f^1 &= a^0 \vee b^0 \vee a^{D'}b^D \vee a^Db^{D'} = 0 \vee 1 \vee (0 \wedge 0) \vee (1 \wedge 0) = 1.
 \end{aligned}$$

Характеристические функции определили двоичный код (0, 1, 0, 1), которому соответствует символ $F1$ (см. табл. 2) алфавита B_{16} . Таким образом, процедура доопределения нашла значение выходного полюса вентиля $G1$ при заданных значениях входных воздействий.

Импликация заключается в снятии возможных противоречий на внутренних сигналах схемы, последние могут возникать вследствие выполнения процедур доопределения и D-распространения. Рассмотрим процедуру импликации на примере вентиля $G8$ схемы ЕХ. Пусть входные полюсы этого вентиля приняли значения $v_0=D'$, $w_0=0$. Выполняя D-распространение, положим $g_8=D^*$. Для снятия возможной неопределенности на линии g_8 необходимо выполнить импликацию. Для этого вначале выполним доопределение, для вентиля $G8$ получим:

$$f^0 = 0, f^{D'} = 1, f^D = 0, f^1 = 0.$$

Импликация сводится к выполнению покомпонентной конъюнкции кода (0, 1, 0, 0) с кодом символа D^* (0, 1, 1, 0). После этого получим: $(0, 1, 0, 0) \& (0, 1, 1, 0) = (0, 1, 0, 0) = D'$. Таким образом, значение полюса g_8 приняло значение D' , что удовлетворяет условию D-распространения. Последнее заключается в том, что значение выходного полюса каждого из элементов в пути активации должно равняться D , D' либо D^* . Если результат импликации не удовлетворяет требованиям D-распространения, то распространение неисправности на выход схемы на данном такте выполнить невозможно.

5. Распространение неисправности на внешний полюс схемы

Рассмотрим триггер V схемы ЕХ и найдем входную последовательность, обеспечивающую проверку неисправности $v_1 \equiv 0$, $v_0 \equiv 1$. Вначале необходимо выполнить инициализацию триггера V в состояние логической единицы ($g_4=1$). Для инициализации последовательно подадим на входы схемы два набора из интервалов ($x_0^1=-, x_1^1=-, x_2^1=1$) и ($x_0^2=0, x_1^2=0, x_2^2=0$), где черточка обозначает, что соответствующая переменная принимает значение из множества $\{0, 1\}$.

Для активации неисправности обозначим состояния выходных полюсов v_1 и v_0 триггера V символами D и D' соответственно и выполним доопределение, т.е. вычислим значения остальных линий схемы ЕХ в алфавите B_{16} на данном такте (табл. 4).

Таблица 4

Результат доопределения для схемы ЕХ на втором и третьем тактах

Такт	x_0	x_1	x_2	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	v_0	v_1	w_0	w_1
2	0	0	0	1	0	0	1	1	1	0	1	0	U	U	1	0
3	C	C	C	$F1$	C	U	U	C	$F1$	$F0$	1	C	D'	D	1	0

Далее необходимо выбрать путь для D-распространения (путь выбирается таким образом, чтобы он вел от одного из выходов тестируемого триггера к внешнему полюсу схемы и не содержал циклов). В случае триггера V существует два варианта выбора пути: $v_0-g_8-g_9-y$ и $v_0-g_6-g_7-w_0-g_8-g_9-y$, причем D-распространение по первому из путей можно попытаться выполнить за один, а по второму – за два и более тактов, так как минимальное число тактов определяется числом триггеров в пути активации.

Рассмотрим путь $v_0-g_8-g_9-y$. Для D-распространения по нему надо положить $g_8^3=D^*$, но это невозможно, так как результат доопределения $g_8^3=1$ (см. табл. 4). Полученное противоречие показывает, что выполнить D-распространение по первому пути за один такт невозможно. Проверим выполнимость D-распространения по этому пути за два такта. Для этого необходимо на третьем такте выполнить активацию неисправности. Воспользовавшись формулой (2) для активации неисправности, найдем входные наборы, обеспечивающие выполнение одного из условий:

$$g_4^3 = 1; \quad (12)$$

$$g_4^3 = D. \quad (13)$$

Подставим в (12) и (13) выражение (5) и решим полученные уравнения:

$$\overline{(x_0^3 D \oplus x_1^3)} x_2^3 = 1 \Rightarrow x_0^3 = 0, x_1^3 = 0, x_2^3 = 0; \quad (14)$$

$$\overline{(x_0^3 D \oplus x_1^3)} x_2^3 = D \Rightarrow x_0^3 = 1, x_1^3 = 1, x_2^3 = 0. \quad (15)$$

В результате найдены наборы $x_0^3=0, x_1^3=0, x_2^3=0$ и $x_0^3=1, x_1^3=1, x_2^3=0$, обеспечивающие активацию требуемой неисправности. Проверим возможность распространения неисправности на наборе (14). Для этого положим $v_1^3=D$ и $v_0^3=D'$ и выполним доопределение (табл. 5).

Таблица 5

Результат доопределения на третьем и четвертом тактах для набора (16)

Такт	x_0	x_1	x_2	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	v_0	v_1	w_0	w_1
2	0	0	0	1	0	0	1	1	1	0	1	0	D'	D	1	0
3	C	C	C	$F1$	C	U	U	C	E	H	1	0	D'	D	1	0

Для D-распространения по пути $v_0-g_8-g_9$ -у положим $g_8^3=D^*$, сделаем требуемое присвоение, после чего выполним импликацию для символов D^* и 1 (результат доопределения для полюса g_8^3), получим $(0, 1, 1, 0) \& (0, 0, 0, 1) = (0, 0, 0, 0) = \emptyset$. Результат импликации показывает невозможность выполнения D-распространения по выбранному пути на наборе $(x_0^3=0, x_1^3=0, x_2^3=0)$. Рассмотрим набор (15). Результат D-распространения и доопределения для него приведен в табл. 6.

Таблица 6

Результат доопределения на третьем и четвертом тактах для набора (17)

Такт	x_0	x_1	x_2	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	v_0	v_1	w_0	w_1
2	0	0	0	D'	1	D'	D	0	D'	D	1	0	D'	D	1	0
3	C	C	C	$F1$	D^*	u	u	C	$D1$	u	D'	$F0$	D'	D	D'	D

Для D-распространения положим $g_8^3=D^*$ и выполним импликацию для полюса w_0 , получим $w_0^3=G0$. Применим к элементу $G8$ импликацию, после чего выполним D-распространение, доопределение и импликацию для элемента $G9$ (табл. 7).

Таблица 7

Результат D-распространения, доопределения и импликации для элементов $G7$ и $G8$

Такт	x_0	x_1	x_2	g_1	g_2	g_3	g_4	g_5	g_6	g_7	g_8	g_9	v_0	v_1	w_0	w_1
2	1	1	0	D'	1	D'	D	0	D'	D	1	0	D'	D	1	0
3	C	C	0	$F1$	D^*	U	U	C	$D1$	$D0$	D'	D	D'	D	D'	D

Выполним D-распространение для триггера Y . Для этого на четвертом такте подадим на входные порты схемы набор из интервала, удовлетворяющего результату последней импликации – $(x_0^4=-, x_1^4=-, x_2^4=0)$. Поскольку выход триггера Y является одновременно и выходным полюсом схемы, то работа алгоритма заканчивается. Таким образом, найдена входная последовательность $(x_0^1=-, x_1^1=-, x_2^1=1), (x_0^2=0, x_1^2=0, x_2^2=0), (x_0^3=1, x_1^3=1, x_2^3=0), (x_0^4=-, x_1^4=-, x_2^4=0)$, обеспечивающая инициализацию и распространение состояний триггера V на внешний полюс схемы EX .

Заключение

В работе предложен метод тестирования D-триггеров последовательностных схем. Условно в методе можно выделить два основных этапа: инициализацию триггера и распространение состояния, установленного в триггере, на один из внешних полюсов схемы.

Отличительной особенностью предлагаемого метода является реализация этапа инициализации триггера. Цель этапа заключается в поиске последовательности входных воздействий, обеспечивающей установку выбранного триггера в требуемое состояние независимо от значений выходных полюсов этого триггера. В работе предложен оригинальный способ осуществления такого поиска. Способ основан на представлении функций возбуждения триггеров схемы в виде совершенной ДНФ и элиминации в полученных представлениях переменных, соответствующих выходным полюсам тестируемого триггера.

Распространение состояния триггера на один из внешних полюсов схемы предлагается выполнять при помощи моделирования поведения неисправной схемы в многозначном алфавите V_{16} . Для этого необходимо последовательно выполнять процедуры D-распространения, доопределения и импликации.

Список литературы

1. Kuehlmann, A. Combinational and sequential equivalence checking / A. Kuehlmann, A.J. Cornelis // Logic synthesis and verification ; ed. by S. Massau, T. Sesao. – N.-Y. : Kluwer Academic Publishers, 2002. – P. 125–143.
2. Скобцов, Ю.А. Логическое моделирование и тестирование цифровых устройств / Ю.А. Скобцов, В.Ю. Скобцов. – Донецк : ДонНТУ, 2005. – 436 с.
3. Bushnell, M.L. Essentials of electronic testing for digital, memory and mixed-signal VLSI circuits / M.L. Bushnell, D.A. Vishwani. – N.-Y. : Kluwer Academic Publishers, 2002. – 713 p.
4. Закревский, А.Д. Логические основы проектирования дискретных устройств / А.Д. Закревский, Ю.В. Поттосин, Л.Д. Черемисинова. – М. : Физматлит, 2007. – 592 с.
5. Бибило, П.Н. VHDL. Эффективное использование при проектировании цифровых систем / П.Н. Бибило, Н.А. Авдеев. – М. : СОЛОН-ПРЕСС, 2006. – 344 с.
6. Киносита, К. Логическое проектирование СБИС ; пер. с япон. / К. Киносита, К. Асада, О. Карацу. – М. : Мир, 1988. – 309 с.
7. Скобцов, Ю.А. Многозначные алфавиты в моделировании и построении тестов цифровых устройств / Ю.А. Скобцов // Электронное моделирование. – 1989. – № 3. – С. 54–60.

Поступила 23.06.08

*Объединенный институт проблем
информатики НАН Беларуси,
Минск, Сурганова, 6
e-mail: kochanovda@newman.bas-net.by*

D.A. Kochanov

CONSTRUCTING TEST PATTERNS FOR TESTING D-TRIGGERS OF SYNCHRONOUS SEQUENTIAL CIRCUITS

A method for sequential circuits testing is proposed where D-triggers are used as memory elements. The method allows finding the test pattern that guarantees the initialization of given D-trigger and propagation of its value to one of the circuit outputs.