



Math-Net.Ru

Общероссийский математический портал

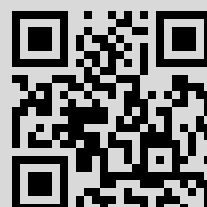
В. Левашенко, К. Морага, Г. Холовински, С. Н. Янушкевич, В. П. Шмерко,
Алгоритм тестирования комбинационных схем на многозначных элементах, *Автомат. и телемех.*, 2000, выпуск 5, 141–156

Использование Общероссийского математического портала Math-Net.Ru подразумевает, что вы прочитали и согласны с пользовательским соглашением
<http://www.mathnet.ru/rus/agreement>

Параметры загрузки:

IP: 194.44.128.2

23 октября 2014 г., 12:54:13



УДК 681.3.16.519.714.24

© 2000 г. В. ЛЕВАШЕНКО, канд. техн. наук
(Экономический университет, Минск, Беларусь),

К. МОРАГА, д-р техн. наук
(Дортмундский университет, Германия),
Г. ХОЛОВИНСКИ, канд. техн. наук,

С. ЯНУШКЕВИЧ, д-р техн. наук
(Технический университет, Щецин, Польша),

В. ШМЕРКО, д-р техн. наук
(Университет информатики и радиоэлектроники, Минск, Беларусь)

АЛГОРИТМ ТЕСТИРОВАНИЯ КОМБИНАЦИОННЫХ СХЕМ НА МНОГОЗНАЧНЫХ ЭЛЕМЕНТАХ¹

Предлагается новый алгоритм для тестирования комбинационных схем на m -значных (многозначных) элементах. Алгоритм позволяет обнаружить однократные неисправности, характерные для современных технологий производства. Предлагаемый алгоритм является обобщением D -алгоритма. Показано, что формирование D -кубов многозначных элементов эквивалентно вычислению направленных логических производных реализуемых ими функций. Алгоритм не требует модификации при изменении типов многозначных элементов в схеме или класса обнаруживаемых ошибок. Приводится классификация неисправностей в таких схемах. Излагаются результаты экспериментальных исследований.

1. Введение

Прогнозы 70-х годов о будущих проблемах бинарных технологий оправдались: современные технологии СБИС достигли физических пределов скорости распространения сигналов в кристалле и площади межконтактных соединений. Альтернативные технологии, например, биомолекулярные [1, 2], пока не достигли уровня промышленного освоения. Один из подходов в повышении степени интеграции состоит в более эффективном использовании линий передач информации в кристалле за счет применения m -уровневых сигналов и многозначных элементов (МЭ).

Более 40 лет (если брать за точку отсчета предпринятую в СССР первую попытку реализации ЭВМ “Сетунь” на 3-значных элементах [3]) потребовалось для создания реальных промышленных образцов МЭ. Успехи промышленности стимулировали формирование нового направления – логического проектирования устройств на МЭ (*Multiple-Valued Logic Design*) [4–8]. В настоящее время промышленностью освоено производство СБИС, в которых используются на одном кристалле бинарные и МЭ. Так, в 1981 г. INTEL начала серийное производство многоуровневой памяти. Далее, MOTOROLA и GENERAL INSTRUMENTS стали использовать 4-уровневые ПЗУ в своих процессорах. ИТАСНИ разработала 16-уровневые динамические ОЗУ, а корпорация NEC недавно заявила об освоении 4-Гбит 4-уровневого ОЗУ. В 1986 г. Тохоки университет и корпорация MATSUSHITA впервые изготовили 32×32 бит 4-уровневый умножитель, который потребляет на 50% мощности меньше, чем бинарные аналоги. TEXAS INSTRUMENTS и NIPPON TELEGRAPH & TELEPHONE заявили о создании семейства МЭ нового поколения. Следует отметить, что во всех случаях МЭ и схемы на их основе используются в составе традиционных схемных решений (на бинарных элементах). Функции и место этих элементов узко очерчены полезностью и эффективностью. В табл. 1 приведены фрагменты успешного промышленного освоения МЭ и устройств на их основе.

¹ Исследования частично финансировались Фондом фундаментальных исследований (Беларусь), Комитетом научных исследований (KBN) (Польша).

Таблица 1

Промышленные и экспериментальные схемы на МЭ

Технология	Схема на многозначных элементах	Характеристики и особенности схемы
КМОП	4-значное ПЗУ в процессоре <i>Intel</i> 8087 [9]	По сравнению с бинарным аналогом, уменьшение площади кристалла на 31%
КМОП	Процессор ввода-вывода <i>Intel</i> 432, <i>Motorola</i> MCM65256 [10]	
5 μ m ПЗС	4-значная ПЛИС <i>PLA-Testcel</i> 8535 [11]	Площадь кристалла 2,0 \times 2,0 мм ² , содержит 40 базовых элементов
КМОП	Сумматор и умножитель в поле <i>GF(4)</i> [12]	Однородная и регулярная структура
3 μ m КМОП	3-значное ПЗУ фирмы <i>Northern Telecom Silicon Canadian Microelectronics Corporation</i> [13]	Объем 2187 \times 5 трехзначных слов. 7 входных и 5 выходных трехзначных адресных полюсов. Площадь кристалла 4,51 \times 4,51 мм ²
5 μ m CMOS	12 \times 12 двухразрядный параллельный умножитель 4-значных сигналов с регулярной древовидной структурой [14]	Уменьшение времени вычисления в 1,2 раза; количества транзисторов в 1,65–1,77 раз; площади межсоединений в 5,2–6,36 раз

2. Неисправности в схемах на МЭ

При тестировании схем на МЭ требуется обнаруживать более широкий, чем в бинарных схемах, класс неисправностей. Так, возникающая на полюсе m -значного элемента однократная неисправность изменяет значение проходящего через полюс сигнала с a на b , где $a, b = 0, \dots, m-1$; $a \neq b$. Для полюса бинарной схемы существуют только два типа изменений проходящего сигнала: с 0 на 1 или с 1 на 0. Для полюса m -значной схемы число возможных изменений – $m(m-1)$ (рис. 1,а). Такое многообразие вариантов определяет более широкий класс возможных неисправностей. Заметим, что в ранних работах по тестированию схем на МЭ рассматривались только неисправности типа константа α ($\alpha = 0, \dots, m-1$) [15–19].

Определение 1. Под однократной неисправностью константа α (*stuck-at- α , constant variation*), возникающей на полюсе x m -значной схемы S , понимают состояние полюса x , при котором на нем устанавливается логический уровень α вне зависимости от состояния схемы и входных сигналов (рис. 1,б).

Последующие исследования неисправностей, возникающих в созданных по различным технологиям МЭ, показали, что для тестирования схем недостаточно обнаруживать только константные неисправности. Поэтому в [20] предложена модель неисправности Кой-Морага, формально записываемая в виде $T' = \nu_x T \vee \phi_x$, где $T = \{t_1, t_2, \dots\}$ – входная последовательность сигналов. В этой модели исходный сигнал t , проходя через полюс, может изменять параметры ν_x и ϕ_x . Модель позволяет описать неисправность окно $\beta_1 - \beta_2$.

Определение 2. Под однократной неисправностью окно $\beta_1 - \beta_2$ (*literal window shift*), $\beta_1, \beta_2 \in \{0, \dots, m-1\}$, возникающей на полюсе x схемы S , понимают состояние полюса x , при котором: а) фиксируется правильный сигнал при прохождении сигнала уровня t ($\beta_1 \leq t \leq \beta_2$), б) устанавливается логический уровень β_1 при прохождении сигнала $t < \beta_1$ или уровень β_2 при $t > \beta_2$ (рис. 1,в).

В [21] исследованы физические дефекты m -значных элементов, возникающие при производстве схем КМОП, управляемых током, и впервые введена модель неисправности r^\pm -кратное отклонение сигнала.

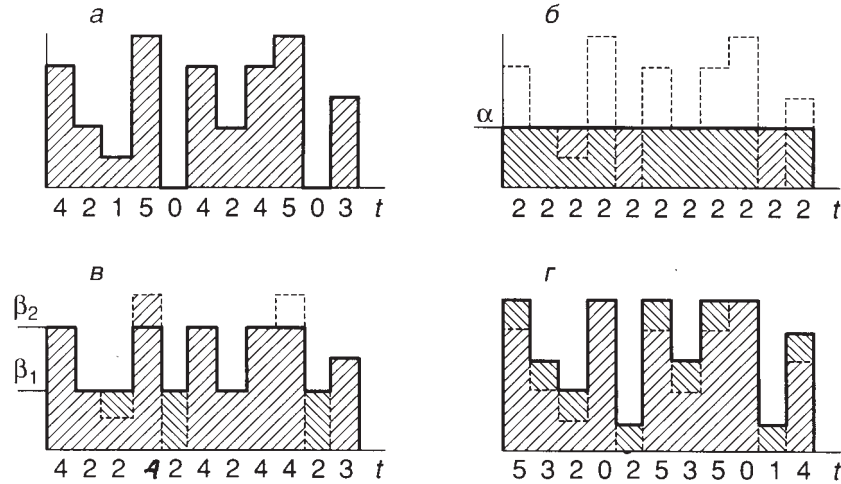


Рис. 1. Типовые неисправности в МЭ, характерные для современных технологий: а – исходная последовательность; б – неисправность константа α ; в – неисправность окно; г – неисправность 1^+ -кратное отклонение сигнала

Определение 3. Под однократной неисправностью r^+ -кратное отклонение сигнала (r^+ -order input signal variation) $r \in \{1, \dots, m-1\}$, возникающей на полюсе x схемы S , понимают состояние этого полюса, при котором значение проходящего через полюс сигнала увеличивается на r единиц: $t' = t + r$, где $t = \{0, \dots, m-1-r\}$. Аналогично, неисправность r^- -кратное отклонение сигнала преобразует значение сигнала как $t' = t - r$, где $t = \{r, \dots, m-1\}$ (рис. 1, г).

Основные типы физических неисправностей в схемах на МЭ, реализованных по различным технологиям, приведены в табл. 2.

Таблица 2

Типы физических и логических неисправностей в МЭ

Логический элемент	Физическая неисправность	Тип неисправности
<i>Технология: 3-значная nМОП и kМОП [22]</i>		
Литерал $k^a x^b (k=0, \dots, m-1)$	Разрыв ветви Короткое замыкание Изменение состояния ветви	константа 0 константа k окно
<i>Технология: 4-значная КМОП схема, управляемая током и напряжением [21]</i>		
Литерал $k^a x^b$ $k^a x^b = k$, если $a \leq x \leq b$; 0, в другом случае	Отражение тока Вариации источника тока Короткое замыкание	окно константа α , $\alpha = 0, \dots, m-1$ константа α константа k
$\min(x_1, x_2)$ и $\max(x_1, x_2)$	Отражение тока	отклонение сигнала
Сумматор $tsum(x_1, x_2)$ и умножитель $tprod(x_1, x_2)$	Отражение тока	константа α окно отклонение сигнала
Мультиплексор	Вариации источника тока	константа α
Сумматор $sum(x_1, x_2)$ и умножитель $prod(x_1, x_2)$ в поле $GF(4)$	Отражение тока	константа α отклонение сигнала

Таким образом, в схемах на МЭ могут возникать различные физические неисправности, которые существенно отличаются от неисправностей в схемах на бинарных элементах. В [23] показано, что для обнаружения этих неисправностей требуются специальные тесты, отличные от тестов для неисправностей *константа α* . В то же время не накоплена статистика физических неисправностей для различных технологий. Поэтому современные алгоритмы тестирования должны гибко реагировать на изменение типа технологии и состава элементов. Это требует создания технологии проектирования алгоритмов тестирования схем на многозначных элементах.

Данное обстоятельство является мотивацией настоящей работы. Кроме того, использование на одном кристалле схем на бинарных элементах и МЭ требует нетривиальных подходов к тестированию таких схем.

3. Алгоритмы тестирования комбинационных схем на МЭ

Анализ известных алгоритмов тестирования комбинационных схем на МЭ показывает, что в целом все они базируются на обобщении методов тестирования бинарных схем. Так, в [17], по-видимому, впервые предпринята попытка обобщения метода активизации многомерного пути Рота для комбинационных схем на МЭ. Предложенный алгоритм формирует тесты для обнаружения неисправностей типа *константа- α* в схемах, реализованных на элементах базиса [24]. Изменение базиса схемы, скажем на МЭ литерала и инвертора [25], для этого алгоритма недопустимо. Кроме того, алгоритм не может обнаружить некоторые неисправности, например, *окно* и *отклонение сигнала*. Развитие алгоритма [17] предложено в [32].

Иной подход предложен в [20]. Алгоритм обнаруживает неисправности типа *константа α* и *окно*. Идея его состоит в том, что для каждого полюса тестируемой схемы формируется модель неисправности, затем в результате решения уравнения находятся тесты. Предложенный алгоритм ориентирован на функциональный базис [25]. Кроме того, предполагается, что МЭ не имеют разветвления по выходу.

Ряд работ посвящен исследованию *чувствительности* схем на МЭ (реакции схемы на изменения значений сигналов на входных полюсах). В этом случае тип функционального базиса не имеет значения, поскольку схема задается не совокупностью МЭ и связей, а логическим описанием реализуемых схемой функций.

Так, в [18] задачу исследования чувствительности схемы предлагается решать с помощью направленных логических производных. При этом для реализуемой комбинационной схемой функции m -значной логики $f(X)$ вычисляется направленная логическая производная $\partial f(j \rightarrow k) / \partial x_i(a \rightarrow b)$. Эта производная определяет условия, при которых изменение входной переменной x_i с a на b вызывает изменение значений на выходе схемы с j на k . Из совокупности таких условий формируются тесты для обнаружения однократной неисправности, возникающей на полюсе x_i схемы. Основной недостаток этого алгоритма и алгоритмов [27, 28] заключается в их ориентации на анализ чувствительности только внешних полюсов схемы. Тот же недостаток присущ и алгоритмам [15, 19, 29].

В настоящей статье развивается идея тестирования схемы на основе принципа активизации многомерного пути. Для этого вводится аппарат обобщенной D -алгебры, который используется при описании схем на МЭ и возникающих на их полюсах неисправностей. Так называемые направленные D -кубы вычисляются с помощью направленных логических производных [30, 31]. Это, а также ряд других идей, позволяют достичь новых качественных характеристик алгоритма: независимости от базиса МЭ и возможности обнаруживать неисправности, характерные для современных технологий изготовления МЭ.

4. Обобщенная D -алгебра

Определение 4. Обобщенной D -алгеброй (GD -алгеброй) называется пара $\langle V, W \rangle$, где V – алфавит символов для описания изменений значений, а W – множество операций над этим алфавитом.

В GD -алгебре используются следующие символы алфавита V :

- а) символ $D_{p \rightarrow q}$ для обозначения изменения значения переменной (функции) с p на q ;
- б) целые числа $0, 1, \dots, m-1$ для определения одного значения переменной;
- в) символ U_Y для обозначения множества значений переменной, формируемого в результате операции разности множеств $M \setminus Y$, где $Y = \{0, 1, \dots, m-1\}$ и $Y \subseteq M$; символ U определяет любое значение из множества M .

Например, запись $D_{1 \rightarrow 2}$ фиксирует факт изменения значения сигнала (переменной) с 1 на 2; запись $U_{0,2}$ означает, что сигнал принимает значения из множества $M \setminus \{0, 2\} = \{1, 3, \dots, m-1\}$ ($m > 4$).

Над символами алфавита V в GD -алгебре можно выполнять операции пересечения и объединения двух символов U_P и U_Q

$$U_P \cap U_Q = \begin{cases} U_{P \cup Q}, & P \cup Q \neq M, \\ \emptyset, & P \cup Q = M. \end{cases} \quad U_P \cup U_Q = \begin{cases} U_{P \cap Q}, & P \cap Q = \emptyset, \\ M, & P \cap Q \neq \emptyset. \end{cases}$$

Аналогами символов алфавита V в бинарной D -алгебре являются символы D и \bar{D} (описывают изменение значения переменной с 1 на 0 и наоборот), символы 0, 1 и X .

Для задания изменения значения функции относительно значения переменной используются *направленные логические производные* функций m -значной логики.

Определение 5. Вектор значений направленной логической производной $\partial \mathbf{X}(j \rightarrow k) / \partial x_i(a \rightarrow b)$ функции m -значной логики $f(x_1, \dots, x_i, \dots, x_n)$, заданной вектором значений \mathbf{X} , определяет условия (наборы $t_1, \dots, t_{i-1}, t_{i+1}, \dots, t_n$ переменных $x_1, \dots, x_{i-1}, x_{i+1}, \dots, x_n$), на которых в результате изменения значения переменной x_i с a на b происходит изменение значения функции с j на k (см. Приложение).

Отождествим переменные функции m -значной логики с входными полюсами схемы, а значения функции с выходным полюсом.

Определение 6. Сингулярным кубом $M\mathcal{E}$ называется способ описания его функции в символах U_Y GD -алгебры.

Сингулярный куб традиционно формируется в результате операции объединения наборов переменных, соответствующих одинаковым значениям функции.

С целью упрощения изложения здесь и далее будем рассматривать $M\mathcal{E}$ с $n \geq 1$ входами и одним выходом

Входные полюсы $M\mathcal{E}$					Выходной полюс $M\mathcal{E}$
x_1	\dots	x_i	\dots	x_n	$f(x_1, \dots, x_i, \dots, x_n)$
U_{Y_1}	\dots	U_{Y_i}	\dots	U_{Y_n}	$f(U_{Y_1}, \dots, U_{Y_i}, \dots, U_{Y_n})$

Символы U_{Y_1}, \dots, U_{Y_n} эквивалентны записи U_Y для полюсов x_1, \dots, x_n , соответственно.

Например, в результате объединения четырехзначных наборов $\langle 1U_000 \rangle$ и $\langle 1U_020 \rangle$ формируется сингулярный куб $\langle 1U_0U_{13}0 \rangle$ (последняя координата записи соответствует выходу $M\mathcal{E}$). В табл. 3 приведены сингулярные кубы $M\mathcal{E}$ для $m = 4$.

Сингулярный куб описывает условия (наборы входных сигналов) постоянства значения на выходе схемы. Для определения условий изменения значений на выходе схемы при изменении входных сигналов введем понятие направленного D -куба.

Таблица 3

Сингулярные кубы типовых 4-значных элементов ($m = 4$)

$\min(x_1, x_2)$			$\max(x_1, x_2)$			$(x_1 \wedge x_2) + 1 \pmod{m}$			$(x_1 \vee x_2) + 1 \pmod{m}$		
0	U	0	0	0	0	0	U	1	0	0	1
U	0	0	1	U_{23}	1	U	0	1	1	U_{23}	2
1	U_0	1	U_{23}	1	1	1	U_0	2	U_{23}	1	2
U_0	1	1	2	U_3	2	U_0	1	2	2	U_3	3
2	U_{01}	2	U_3	2	2	2	U_{01}	3	U_3	2	3
U_{01}	2	2	3	U	3	U_{01}	2	3	3	U	0
3	3	3	U	3	3	3	3	0	U	3	0

Определение 7. Направленным D -кубом (DD -кубом) для МЭ называется способ описания его функции в символах $D_{p \rightarrow q}$ и U_Y множества V GD -алгебры.

Другими словами, DD -куб является одной из форм записи функции МЭ, причем записи только одного из возможных изменений в МЭ. Для упрощения изложения далее, если специально не оговорено, рассматриваются DD -кубы для МЭ с изменением на одном входе x_i .

В общем случае DD -куб для МЭ с n входами $x_1, \dots, x_i, \dots, x_n$, реализующего логическую функцию $f(x_1, \dots, x_i, \dots, x_n)$, представляется следующим образом:

Входные полюсы МЭ					Выходной полюс МЭ
x_1	\dots	x_i	\dots	x_n	$f(x_1, \dots, x_i, \dots, x_n)$
U_{Y_1}	\dots	$D_{a \rightarrow b}$	\dots	U_{Y_n}	$D_{j \rightarrow k}$

Скажем, запись DD -куба в виде $\langle D_{0 \rightarrow 1} U_0 2 D_{1 \rightarrow 2} \rangle$ означает: а) МЭ имеет три входа и один выход; б) изменение сигнала на первом входе с 0 на 1 приводит к изменению выходного сигнала с 1 на 2 при условиях: сигнал на втором входе отличен от нуля, а сигнал на третьем входе соответствует уровню логической 2.

DD -кубы можно формировать с помощью направленных логических производных функций m -значной логики или операции пересечения сингулярных кубов.

В работе используется первый способ. Суть его состоит в том, что между направленной логической производной функции m -значной логики и DD -кубом для МЭ, реализующего эту функцию, существует взаимно однозначная связь.

Утверждение 1. Направленная логическая производная $\partial \mathbf{X}(j \rightarrow k) / \partial x_i(a \rightarrow b)$ функции m -значной логики определяет DD -куб.

Доказательство следует из определения производной $\partial \mathbf{X}(j \rightarrow k) / \partial x_i(a \rightarrow b)$ и DD -куба МЭ.

Например, для 3-значной функции $f(X)$ 4-х переменных производная, заданная вектором $\partial \mathbf{X}(0 \rightarrow 2) / \partial x_1(0 \rightarrow 1) = [222000000 \ 222000000 \ 000000000 \ 222000000 \ 222000000 \ 000000000 \ 222000000 \ 222000000 \ 000000000]^T$, имеет вид $\partial f(0 \rightarrow 2) / \partial x_1(0 \rightarrow 1) = (x_2^0 \vee x_2^1) x_3^0$. Изменение значения функции с 0 на 2 ($D_{0 \rightarrow 2}$) при изменении значения переменной x_1 с 0 на 1 ($D_{0 \rightarrow 1}$) происходит: а) при значении $x_2 = 0$ или 1 (U_2); б) при значении $x_3 = 0$ (0) и любом значении x_4 (U). МЭ, реализующему эту функцию, соответствует DD -куб:

Входные полюсы элемента				Выходной полюс элемента
x_1	x_2	x_3	x_4	$f(x_1, x_2, x_3, x_4)$
$D_{0 \rightarrow 1}$	U_2	0	U	$D_{0 \rightarrow 2}$

Теорема 1. Для МЭ с n входами и 1 выходом существует не более $m(m-1)^2 n m^{n-2}$ DD -кубов.

Доказательство. Поскольку значение сигнала на полюсе x_i МЭ может изменяться с a на b ($a, b = 0, \dots, m-1$, $a \neq b$), то для этого полюса возможно $A_m^2 = m(m-1)$ различных изменений $D_{a \rightarrow b}$ (A_m^2 – число размещений из m по 2). Для каждого входного полюса МЭ имеем $m(m-1)n$ возможных изменений. При одном изменении входного сигнала возможно $(m-1)t$ изменений выходного сигнала. Общее количество возможных изменений входных и выходных сигналов МЭ равно $m(m-1)^2n$. Эта величина определяет *верхнюю границу* числа отличных от нуля направленных логических производных. Однако последняя зависит от $n-1$ переменной. Известно, что верхняя граница числа слагаемых в минимальных и тупиковых формах представления функций $n-1$ переменных в базисе Россера–Тьюкета не превышает числа m^{n-2} . Следовательно, по одной направленной производной формируются не более m^{n-2} DD -кубов, а для МЭ с n входами и одним выходом существует не более $m(m-1)^2nm^{n-2}$ DD -кубов.

Следствие 1. Для МЭ с двумя входами и одним выходом существует не более $2m(m-1)^2$ DD -кубов.

Утверждение 2. Для МЭ с 2 входами и одним выходом существует в точности q DD -кубов, q – число отличных от нуля направленных логических производных.

Доказательство следует из факта непорождения DD -кубов нулевыми производными.

Теорема 2. DD -куб, полученный при одновременной замене компонент $D_{a \rightarrow b}$ на $D_{b \rightarrow a}$ и $D_{j \rightarrow k}$ на $D_{k \rightarrow j}$, эквивалентен исходному DD -кубу.

Доказательство следует из свойства симметрии направленной логической производной $\partial \mathbf{X}(j \rightarrow k) / \partial x_i(a \rightarrow b) = \partial \mathbf{X}(k \rightarrow j) / \partial x_i(b \rightarrow a)$.

Изложенное является основой для построения алгоритма формирования DD -куба по результату вычисления направленной логической производной.

Алгоритм формирования DD -куба для МЭ

<i>Вход:</i>	Логическая функция МЭ в виде вектора значений \mathbf{X}
<i>Выход:</i>	DD -куб для m -уровневого элемента.
<i>Шаг 1</i>	Вычислить отличную от нуля направленную логическую производную $\partial \mathbf{X}(j \rightarrow k) / \partial x_i(a \rightarrow b)$ для $j, k, a, b = 0, \dots, m-1$; $a \neq b$; $i = 1, \dots, n$.
<i>Шаг 2</i>	Определить номера t элементов вектора $\partial \mathbf{X}(j \rightarrow k) / \partial x_i(a \rightarrow b)$, равных $(m-1)$.
<i>Шаг 3</i>	Записать DD -куб, используя параметры изменения j, k, a, b и m -значные представления номеров $t = (t_1, \dots, t_{i-1}, t_{i+1}, \dots, t_n)_m$ по правилу: а) i -му компоненту DD -куба соответствует запись $D_{a \rightarrow b}$, а $(n+1)$ -му – $D_{j \rightarrow k}$; б) поставить в соответствие ℓ -му ($\ell = 1, \dots, n$, $\ell \neq i$) компоненту DD -куба множество $Y_\ell = M \setminus t_\ell$; в) представить ℓ -й компонент DD -куба символом U_{Y_ℓ} .

Конец алгоритма

Приведенный алгоритм формирует одномерные DD -кубы. Для формирования многомерных DD -кубов на шаге 1 следует вычислять направленную производную по вектору переменных [32]. С учетом этого шаг 3 алгоритма будет несколько модифицирован.

Пример 1. Вычисленные в соответствии с приведенным алгоритмом DD -кубы для некоторых 4-значных элементов приведены в табл. 4.

Определение 8. *Пересечением DD -кубов называется операция над DD -кубами соединенных между собой МЭ, при условии совпадения изменения значения сигнала на выходе одного из них и на входе другого.*

Таблица 4

Направленные логические производные и соответствующие им DD -кубы для 4-значных элементов, реализующие функции $f_1(X)$, $f_2(X)$ и литерала

Отличная от нуля направленная логическая производная	DD -куб элемента	
	Входы	Выход
$f_1(\mathbf{X}) = (x_1 \wedge x_2) + 1(\bmod 4)$, $\mathbf{X} = [11111222 \ 12331230]$		
$\partial X(1 \rightarrow 2)/\partial x_1(0 \rightarrow 1) = 0333033303330333$	$D_{0 \rightarrow 1} \ U_0$	$D_{1 \rightarrow 2}$
$\partial X(1 \rightarrow 2)/\partial x_1(0 \rightarrow 2) = 0300030003000300$	$D_{0 \rightarrow 2} \ 1$	$D_{1 \rightarrow 2}$
$\partial X(1 \rightarrow 3)/\partial x_1(0 \rightarrow 2) = 0033003300330033$	$D_{0 \rightarrow 2} \ U_{01}$	$D_{1 \rightarrow 3}$
$\partial X(1 \rightarrow 2)/\partial x_1(0 \rightarrow 3) = 0300030003000300$	$D_{0 \rightarrow 3} \ 1$	$D_{1 \rightarrow 2}$
$\partial X(1 \rightarrow 3)/\partial x_1(0 \rightarrow 3) = 0030003000300030$	$D_{0 \rightarrow 3} \ 2$	$D_{1 \rightarrow 3}$
$\partial X(1 \rightarrow 0)/\partial x_1(0 \rightarrow 3) = 0003000300030003$	$D_{0 \rightarrow 3} \ 3$	$D_{1 \rightarrow 0}$
$\partial X(2 \rightarrow 3)/\partial x_1(1 \rightarrow 2) = 0033003300330033$	$D_{1 \rightarrow 2} \ U_{01}$	$D_{2 \rightarrow 3}$
$\partial X(1 \rightarrow 3)/\partial x_1(1 \rightarrow 3) = 0030003000300030$	$D_{1 \rightarrow 3} \ 2$	$D_{2 \rightarrow 3}$
$\partial X(2 \rightarrow 0)/\partial x_1(1 \rightarrow 3) = 0003000300030003$	$D_{1 \rightarrow 3} \ 3$	$D_{2 \rightarrow 0}$
$\partial X(3 \rightarrow 0)/\partial x_1(2 \rightarrow 3) = 0003000300030003$	$D_{2 \rightarrow 3} \ 3$	$D_{3 \rightarrow 0}$
$f_2(X) = (x_1 \vee x_2) + 1(\bmod 4)$, $\mathbf{X} = [12302230 \ 33300000]$		
$\partial X(1 \rightarrow 2)/\partial x_1(0 \rightarrow 1) = 3000300030003000$	$D_{0 \rightarrow 1} \ 0$	$D_{1 \rightarrow 2}$
$\partial X(1 \rightarrow 3)/\partial x_1(0 \rightarrow 2) = 3000300030003000$	$D_{0 \rightarrow 2} \ 0$	$D_{1 \rightarrow 3}$
$\partial X(2 \rightarrow 3)/\partial x_1(0 \rightarrow 2) = 0300030003000300$	$D_{0 \rightarrow 2} \ 1$	$D_{2 \rightarrow 3}$
$\partial X(1 \rightarrow 0)/\partial x_1(0 \rightarrow 3) = 3000300030003000$	$D_{0 \rightarrow 3} \ 0$	$D_{1 \rightarrow 0}$
$\partial X(2 \rightarrow 0)/\partial x_1(0 \rightarrow 3) = 0300030003000300$	$D_{0 \rightarrow 3} \ 1$	$D_{2 \rightarrow 0}$
$\partial X(3 \rightarrow 0)/\partial x_1(0 \rightarrow 3) = 0030003000300030$	$D_{0 \rightarrow 3} \ 2$	$D_{3 \rightarrow 0}$
$\partial X(2 \rightarrow 3)/\partial x_1(1 \rightarrow 2) = 3300330033003300$	$D_{1 \rightarrow 2} \ U_{23}$	$D_{2 \rightarrow 3}$
$\partial X(2 \rightarrow 0)/\partial x_1(1 \rightarrow 3) = 3300330033003300$	$D_{1 \rightarrow 3} \ U_{23}$	$D_{2 \rightarrow 0}$
$\partial X(3 \rightarrow 0)/\partial x_1(1 \rightarrow 3) = 0030003000300030$	$D_{1 \rightarrow 3} \ 2$	$D_{3 \rightarrow 0}$
$\partial X(3 \rightarrow 0)/\partial x_1(2 \rightarrow 3) = 3330333033303330$	$D_{2 \rightarrow 3} \ U_3$	$D_{3 \rightarrow 0}$
Литерал $- 2^0 x^2$, $\mathbf{X} = [2220]$		
$\partial X(0 \rightarrow 1)/\partial x(0 \rightarrow 1) = [3333]$	$D_{0 \rightarrow 3}$	$D_{2 \rightarrow 0}$
$\partial X(0 \rightarrow 1)/\partial x(0 \rightarrow 2) = [3333]$	$D_{1 \rightarrow 3}$	$D_{2 \rightarrow 0}$
$\partial X(0 \rightarrow 1)/\partial x(3 \rightarrow 1) = [3333]$	$D_{2 \rightarrow 3}$	$D_{2 \rightarrow 0}$

Например, результатом операции пересечения DD -кубов для МЭ $\min(x_1, x_2)$ (A) и $\max(x_1, x_2)$ (B) является DD -куб составного МЭ (C), реализующего функцию $\max(\min(x_1, x_2), x_3)$ (рис. 2). Результирующий DD -куб определяет условия (значения сигналов на 2-м и 3-м полюсах МЭ C), при которых изменение сигнала $D_{0 \rightarrow 1}$ на его первом входе передается на выход элемента ($D_{0 \rightarrow 1}$).

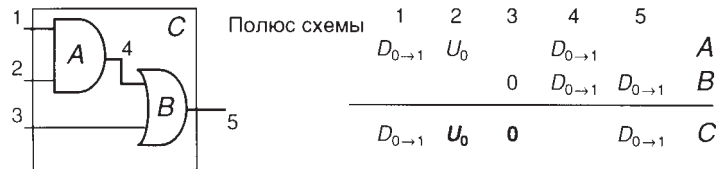


Рис. 2. Интерпретация операции пересечения DD -кубов на примере МЭ $\min(x_1, x_2)$ и $\max(x_1, x_2)$

5. Взаимосвязь неисправностей МЭ в терминах GD -алгебры

Соответствие между DD -кубами и моделями типовых неисправностей, присущих некоторым технологиям изготовления МЭ, устанавливается на основе табл. 5 [23, 33, 34].

В общем случае однократная неисправность комбинационной схемы на МЭ определяется через ее влияние на изменение значения проходящего через полюс сигнала. Если на полюсе схемы присутствует неисправность, то значение проходящего сигнала изменяется с уровня a на уровень b . Поэтому для обнаружения неисправности необходимо: а) определить исходное значение сигнала, которое изменяет неисправность, – параметр a ; б) задать изменение этого сигнала, вызванное этой неисправностью $D_{a \rightarrow b}$. Эти изменения являются основой для описания примитивных DD -кубов неисправности. Последние состоят из входных наборов, позволяющих наблюдать неисправность на выходе элемента, и символа $D_{j \rightarrow k}$ в выходной координате.

В ряде работ по тестированию схем на бинарных элементах показана достаточность построения тестов, обнаруживающих неисправности *константа 0* и *константа 1*. Это объясняется тем, что обе указанные неисправности обнаруживаются всеми возможными изменениями сигнала на этом полюсе ($D_{1 \rightarrow 0}$ и $D_{0 \rightarrow 1}$). Однако для МЭ число возможных изменений равно $m(m-1)$. Поэтому важным является установление взаимосвязей между различными типами неисправностей через соответствующие им изменения.

Анализ отношений между изменениями позволяет сформулировать следующие правила

Правило 1. Множество всех изменений, вызываемых двумя неисправностями r^+ - и $(m-r)$ -кратное отклонение сигнала ($r = 1, \dots, m-1$), принадлежит множеству изменений, вызываемых всеми неисправностями *константа α* ($\alpha = 0, \dots, m-1$).

Правило 2. Множество всех изменений, вызываемых одной неисправностью *константа $(m-1)$* , принадлежит множеству изменений, вызываемых всеми неисправностями r^+ -кратное отклонение сигнала ($r = 1, \dots, m-1$). Аналогично, множество всех изменений, вызываемых неисправностью *константа 0*, принадлежит множеству изменений, вызываемых неисправностями r^- -кратное отклонение сигнала ($r = 1, \dots, m-1$).

Правило 3. Множество изменений, вызываемых неисправностью *окно $\beta_1 - \beta_2$* ($\beta_2 = m-1$), является подмножеством изменений, вызываемых неисправностью *константа β_1* . Аналогично, множество изменений, вызываемых неисправностью *окно $\beta_1 - \beta_2$* ($\beta_1 = 0$), является подмножеством изменений, вызываемых неисправностью *константа β_2* .

Таблица 5

Неисправности в комбинационных схемах на МЭ и их описание в GD -алгебре

Тип неисправностей	Логическая модель неисправности	Изменения в терминах GD -алгебры
константа α	$t' = \alpha$	$D_{p \rightarrow \alpha}, p \in \{0, \dots, m-1\}, p \neq \alpha$
r^\pm -кратное отклонение сигнала	$t' = t + r$ $t' = t - r$	$D_{p \rightarrow p+r}, p \in \{0, \dots, (m-1)-r\}$ $D_{p \rightarrow p-r}, p \in \{r, \dots, m-1\}$
окно $\beta_1 - \beta_2$	$t' = t, \beta_1 \leq t \leq \beta_2$ $t' = \beta_1, t < \beta_1$ $t' = \beta_2, t > \beta_2$	$D_{p \rightarrow \beta_1}, p \in \{0, \dots, \beta_1\}$ $D_{p \rightarrow \beta_2}, p \in \{\beta_2, \dots, m-1\}$

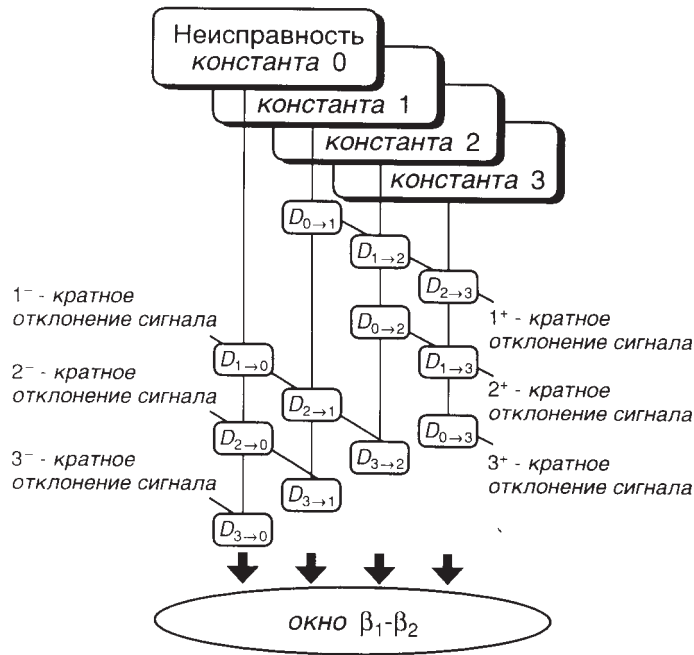


Рис. 3. Взаимосвязь между изменениями, вызванными неисправностями константа α r -кратное отклонение сигнала и окно $\beta_1 - \beta_2$ схем на 4-значных элементах ($m = 4$)

Правило 4. Изменение, вызываемое неисправностью $(m - 1)^-$ -кратное отклонение сигнала, принадлежат множеству изменений, вызываемых неисправностью константа 0. Аналогично, изменение, вызываемое неисправностью $(m - 1)^+$ -кратное отклонение сигнала, принадлежит множеству изменений, вызываемых неисправностью константа $(m - 1)$.

Рис. 3 иллюстрирует приведенные правила. Эти правила позволяют определить примитивные DD -кубы неисправности, описывающие неисправность полюса схемы в зависимости от типа МЭ и технологии его изготовления.

6. Обобщенный D -алгоритм

Поясним процесс использования предложенных DD -кубов для транспортировки неисправности.

Определение 9. Направленной DD -последовательностью будем называть способ описания схемы в терминах понятия "изменение" с заданной на s -м полюсе неисправностью, а алгоритм реализации этой последовательности – обобщенным D -алгоритмом или GD -алгоритмом.

Последовательность шагов этого алгоритма соответствует классической схеме: задание неисправности, активизация возможных путей транспортировки неисправности от места возникновения к выходу схемы, анализ противоречий и выбор альтернативных путей транспортировки неисправности, определение условий активизации и транспортировки неисправности [35].

Задание неисправности состоит в построении для полюса комбинационной схемы примитивного DD -куба неисправности. Активизация пути от МЭ к выходу схемы

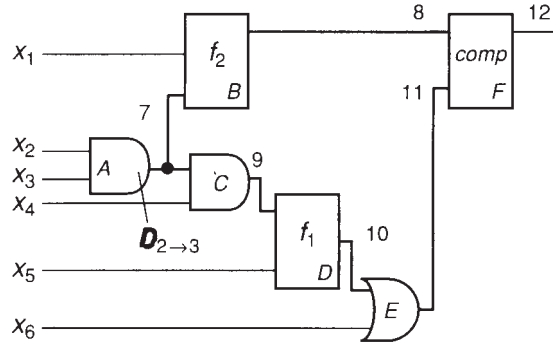


Рис. 4. Многомерный путь транспортировки неисправности в результате тестирования комбинационной схемы на МЭ

означает последовательное выполнение операции пересечения DD -последовательности для заданной неисправности с DD -кубами МЭ на этом пути. Результатом транспортировки является построение промежуточной DD -последовательности, в которой выходной полюс схемы описывается символом $D_{p \rightarrow q}$. Активизация возможных путей транспортировки неисправности, анализ противоречий и выбор альтернативных путей транспортировки эквивалентны предложенным в [33, 35].

GD -алгоритм для обнаружения однократных неисправностей в комбинационной схеме на МЭ

<i>Вход:</i>	Структура комбинационной схемы на МЭ; номер тестируемого полюса; тип тестируемой неисправности.
<i>Выход:</i>	Тест для заданной неисправности схемы на МЭ.
<i>Шаг 1</i>	Определить для тестируемого полюса схемы изменения значений сигналов, вызванные неисправностью (DD -куб неисправности).
<i>Шаг 2</i>	Определить МЭ схемы, находящиеся на пути транспортировки неисправности от места ее возникновения до выхода схемы.
<i>Шаг 3</i>	Сформировать DD -кубы для этих МЭ.
<i>Шаг 4</i>	Выполнить операции пересечения DD -куба неисправности с DD -кубами для МЭ, находящимися на пути транспортировки неисправности (прямая фаза). Результат – DD -последовательность.
<i>Шаг 5</i>	Выполнить доопределение DD -последовательности входными значениями МЭ, не лежащих на пути активизации (обратная фаза).
Конец алгоритма	

Пример 2. Построить тест для полюса 7 комбинационной схемы на рис. 4. Схема реализована на 4-значных элементах КМОП, управляемых током, а также током и напряжением. Элементы A и C реализуют функцию $\min(x_1, x_2)$, элемент $B - f_2(X) = (x_1 \vee x_2) + 1(\text{mod } 4)$, элемент $D - f_1(X) = (x_1 \wedge x_2) + 1(\text{mod } 4)$, элемент $E - \max(x_1, x_2)$ и элемент $F - \text{компаратор } \text{comp}(x_1, x_2)$. Тест реализуется следующим образом.

1°. На полюсе 7 возможно возникновение неисправностей *отклонение сигнала* (табл. 2): 1^+ -кратное *отклонение сигнала* увеличивает значения сигналов на 1, например, 2 на 3 ($D_{2 \rightarrow 3}$) (табл. 5).

2°. Определить пути транспортировки этого изменения: а) $f_2(X) - \text{comp}(x_1, x_2) - \text{выход схемы}$; б) $\min(x_1, x_2) - f_1(X) - \max(x_1, x_2) - \text{comp}(x_1, x_2) - \text{выход схемы}$.

Таблица 6

Построение DD -последовательности для обнаружения на полюсе 7
схемы рис. 5 изменения $D_{2 \rightarrow 3}$

Номер МЭ	Номер полюса в комбинационной схеме											
	Входные полюсы схемы						Промежуточные полюсы					Выход
	1	2	3	4	5	6	7	8	9	10	11	12
A	U_{01} 2						$D_{2 \rightarrow 3}$					
C	3						$D_{2 \rightarrow 3}$ $D_{2 \rightarrow 3}$					
D	3						$D_{2 \rightarrow 3}$ $D_{3 \rightarrow 0}$					
E	2						$D_{3 \rightarrow 0}$ $D_{3 \rightarrow 2}$					
B	U_3						$D_{2 \rightarrow 3}$ $D_{3 \rightarrow 0}$					
F							$D_{3 \rightarrow 0}$ $D_{3 \rightarrow 2}$					
U_3 U_{01} 2 3 3 2							$D_{2 \rightarrow 3}$	$D_{3 \rightarrow 0}$	$D_{2 \rightarrow 3}$	$D_{3 \rightarrow 0}$	$D_{3 \rightarrow 2}$	$D_{1 \rightarrow 2}$

Примечание: а) поскольку транспортировка осуществляется по двумерному пути, то для элемента F использован DD -куб, полученный при условии изменений сигналов на обоих входных полюсах; б) DD -куб МЭ F формируется по результату направленной логической производной по вектору переменных [32, 34].

3°. Сформировать сингулярные кубы и DD -кубы для элементов, находящихся на пути транспортировки (табл. 3, 4). Двумерный DD -куб элемента F равен $D_{3 \rightarrow 0} D_{3 \rightarrow 2} D_{1 \rightarrow 2}$.

4°. Активизировать путь транспортировки неисправности (табл. 6). Результат: DD -последовательность, полученная путем покоординатного пересечения DD -кубов для соответствующих МЭ. Искомый тест для обнаружения изменения $D_{2 \rightarrow 3}$, вызванного неисправностью, имеет вид: $U_3 U_{01} 2 3 3 2$.

5°. Аналогично обнаруживаются остальные возможные неисправности.

7. Экспериментальные результаты

Исследования проведены на персональном компьютере Сyrix 200+, 32 Mb ОЗУ в среде Linux-UNIX OS. Алгоритм реализован на языке C++. Экспериментальные исследования проводились на типовых схемах (*benchmarks*, в формате *EDIF*) с интерпретацией элементов в 4-значной логике. Преобразование заключалось в изменении значности реализуемых элементами логических функций с бинарной на m -значную. Такой подход применяется, в частности, в [36]. Алгоритм тестировался на схемах в функциональных базисах: элементы $\min(x_1, x_2)$, $\max(x_1, x_2)$, литерала (табл. 7) и $\text{sum}(x_1, x_2)$, $\text{prod}(x_1, x_2)$ в $GF(4)$ [12].

Эксперимент предусматривал: а) определение для каждого полюса схемы неисправностей, присущих схемам КМОП, управляемым током; б) определение множества изменений, вызываемых этими неисправностями, упорядочение этих изменений по частоте встречаемости; в) применение GD -алгоритма для определения тестов, обнаруживающих эти неисправности.

Целью проведения эксперимента было определение временных и количественных характеристик работы предлагаемого алгоритма. Поэтому при проведении эксперимента предварительно не были применены другие методы тестирования, например, псевдослучайное тестирование.

Полученные результаты (табл. 7, 8) свидетельствуют о достаточно высокой эффективности предложенного алгоритма и возможности его практического использования.

Таблица 7

Результаты тестирования схем, содержащих элементы $\min(x_1, x_2)$, $\max(x_1, x_2)$ и литералы

Схема			Время [с]	Число тестов	Обнаруженные неисправности (в %)							
Имя	Вх/Вых	МЭ			$r+2$	$r+3$	$r-1$	$r-2$	$r-3$	$s-a-0$	$s-a-1$	$s-a-2$
$b12_f2$	10/1	59	13,07	159	100	96	100	98	98	93	100	93
$b12_f6$	9/1	57	2,90	57	100	96	100	98	98	100	100	100
sao_f0	10/1	109	18,38	109	100	98	100	98	98	100	100	100
$b1$	3/4	19	0,07	101	96	96	98	98	98	93	100	93
$c17$	5/2	12	0,02	51	98	96	98	98	98	100	100	100
$cm151a$	12/2	56	2,92	249	96	93	96	98	96	77	89	77
$cm162a$	14/5	80	6,87	308	96	94	96	98	96	79	89	79
$cm42a$	4/10	29	0,09	124	96	94	96	98	96	94	100	92
$cm82a$	5/3	30	0,09	150	98	94	94	100	94	94	100	94
$cm85a$	11/3	76	24,20	273	98	98	98	98	98	100	100	100

Примечание: символам $r+2$, $r-2$ и пр. соответствуют неисправности 2^+ -кратное отклонение сигнала, 2^- -кратное отклонение сигнала и пр. Запись $s-a-0$ и пр. определяет неисправность типа константа 0 и пр. Вх/Вых – число входов и выходов (полюсов) схемы; МЭ – число элементов в схеме. Обнаружено 100% неисправностей $r+1$, константа 3.

Таблица 8

Результаты тестирования схем, содержащих элементы сумматора и умножителя в поле $GF(4)$

Схема			Время [с]	Число тестов
Имя	Вх/Вых	МЭ		
bw_f4	5/1	45	0,05	60
$ex5_f2$	5/1	47	0,06	60
inc_f6	6/1	43	0,14	57
$rd84_f0$	8/1	55	0,10	84
$root_f0$	6/1	91	0,15	126

Примечание: обнаружено 100% неисправностей $r \pm 1$, $r \pm 2$, $r \pm 3$, константа 0, 1, 2, 3.

8. Заключение

Предложенный GD алгоритм тестирования схем на МЭ позволяет строить тесты для обнаружения неисправности заданного типа на любом полюсе схемы, если данная неисправность обнаруживаема. Алгоритм обладает рядом новых функциональных возможностей и позволяет:

а) обнаруживать класс неисправностей, характерных для различных современных технологий изготовления МЭ и схем на их основе;

б) гибко реагировать на потребности технологии, т.е. настраиваться на логические ошибки, характерные для конкретных МЭ и технологических особенностей их изготовления.

Мы проводим дальнейшие исследования по двум направлениям. Первое из них имеет целью улучшение временных характеристик алгоритма и, как следствие, тестирование схем с большим числом МЭ (в данной работе мы ограничились схемами с числом МЭ порядка 100). Так, например, весьма привлекательны для обобщения идеи, положенные в основу алгоритмов PODEM и CONT. Второе направление связано с обобщением подхода, реализуемого в хорошо известных системах FAN и SOCRATES: на первом этапе проводить тестирование схемы с помощью псевдослучайных, вероятностных алгоритмов, а на втором (в случае необходимости) те-

стировать схему с помощью предлагаемого алгоритма. В этом плане представляют интерес результаты [36, 37], в которых показано, что случайное тестирование на основе эволюционных принципов очень эффективно.

Авторы признательны за критические замечания и дискуссии профессору В. Н. Ярмолику (Беларусь), профессору В. Д. Малюгину (Россия) и профессору G. Dueck (Канада).

ПРИЛОЖЕНИЕ

Основы булева дифференциального исчисления изложены в [30, 37]. Обобщения для функций многозначной логики приведены в [18, 34]. Техника вычислений логических дифференциальных операторов представлена в [31, 34, 38], а некоторые прикладные аспекты использования этих операторов в [15, 19, 27, 34]. В основу вычисления логических дифференциальных операторов положены регулярные преобразования с использованием векторно-матричных преобразований.

Пусть задана m -значная логическая функция $f(X)$, зависящая от n переменных

$$f(X) = \bigvee_{\ell=0}^{m-1} \ell \wedge P_{\ell}(X), \quad \text{где} \quad P_{\ell}(X) = \bigvee_{f(r_1, r_2, \dots, r_n) = \ell} x_1^{r_1} x_2^{r_2} \dots x_n^{r_n},$$

где $x_i^{r_i}$ означает операцию литерал: $x_i^{r_i} = 0$ при $x_i \neq r_i$ и $x_i^{r_i} = m - 1$ при $x_i = r_i$.

Направленная логическая производная этой функции определяется выражением

$$\partial f(j \rightarrow k) / \partial x_i(a \rightarrow b) = P_j(X)|_{x_i=a} \cdot P_k(X)|_{x_i=b},$$

где $j, k, a, b = 0, \dots, m - 1$, символ “ \cdot ” означает логический минимум (конъюнкцию) [18].

Если функция задана вектором значений \mathbf{X} , то вектор значений направленной логической производной вычисляется по соотношению

$$(П.1) \quad \partial \mathbf{X}(j \rightarrow k) / \partial x_i(a \rightarrow b) = \left(P^{(i,a)} \cdot \varphi_j(\mathbf{X}) \right) \cdot \left(P^{(i,b)} \cdot \varphi_k(\mathbf{X}) \right),$$

где $\varphi_j(\mathbf{X})$ и $\varphi_k(\mathbf{X})$ – операции вычисления векторного литерала. Матрицы $P^{(i,a)}$ и $P^{(i,b)}$ размерности $m^n \times m^n$ формируются посредством кронекеровского произведения и имеют регулярную блочную структуру:

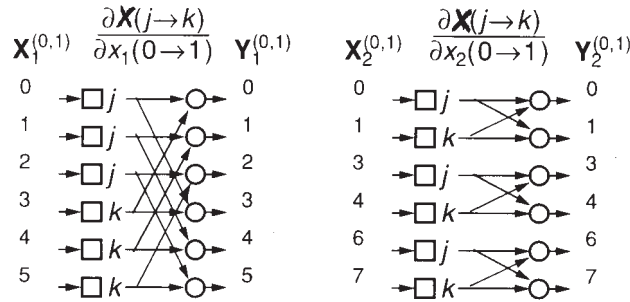


Рис. 5. Операционные графы процесса вычислений направленных логических производных (П.1) для $m = 3$; $n = 2$; $a = 0$, $b = 1$; $i = 1, 2$

$$P^{(i,\ell)} = M_m i - 1 \otimes P_m^{(\ell)} \otimes M_m n - i,$$

где $M_m i - 1$ и $M_m n - i$ – диагональные матрицы размерности $m^{i-1} \times m^{i-1}$ и $m^{n-i} \times m^{n-i}$ соответственно, на главных диагоналях которых записаны значения $(m-1)$; \otimes означает кронекеровское произведение; матрица $P_m^{(\ell)}$ имеет структуру вида:

$$P_m^{(\ell)} = [\varphi_\ell(0)\varphi_\ell(1) \dots \varphi_\ell(m-1)] \otimes \left[\begin{array}{c} m-1 \\ \dots \\ m-1 \end{array} \right] \Bigg\} m \text{ строк.}$$

Рис. 5 поясняет выражение (П.1).

СПИСОК ЛИТЕРАТУРЫ

1. Aoki T., Kameyama M., Higuchi T. Design of interconnection-free biomolecular computing system // Proc. 21th IEEE Int. Symp. on Multiple-Valued Logic. 1992. P. 173–180.
2. Roof D. Recent development in DNA-computing // Proc. 27th IEEE Int. Symp. on Multiple-Valued Logic. 1997. P. 3–9.
3. Бруснецов Н. П. Микро-ЭВМ СЕТУНЬ // Вестник МГУ. 1962. № 4. С. 3–12.
4. Hurst S. L. Multiple-valued logic – its status and its future // IEEE Trans. on Computers. 1984. V. C-33. No. 12. P. 1160–1179.
5. Jain A. K., Bolton R. J., Abd-El-Barr M. H. CMOS multiple-valued logic design. Part I: Circuit implementation. Part II: Function realization // IEEE Trans. on Circuits and Systems. 1993. V. 40. No. 8. P. 503–522.
6. Miller D. M. Multiple-valued logic design tools // Proc. 23th IEEE Symp. on Multiple-Valued Logic. 1993. P. 2–11.
7. Muzio J. C., Wesselkamper T. S. Multiple-valued switching theory. Adam Higler. 1986.
8. Sasao T. On the optimal design of multiple-valued PLA's // IEEE Trans. Computers. 1989. V. C-38. No. 4. P. 582–592.
9. Stark M. Two bits per cell ROM // Proc. of IEEE COMPCON. 1981. P. 209–216.
10. Agarwal V. K., Pugsley J. H., Silio C. B. Multiple-valued output ROM circuits // Proc. 14th IEEE Int. Symp. on Multiple-Valued Logic. 1984. P. 224–231.
11. Kerkhoff H. G., Butlar J. T. Design of a high-radix programmable logic array using profiled peristaltic charge coupled devices // IEEE Proc. 16th Int. Symp. on Multiple-Valued Logic. 1986. P. 128–136.
12. Zilic Z., Vranesic Z. G. Current-mode CMOS Galois field circuits // IEEE Proc. 20th Int. Symp. on Multiple-Valued Logic. 1990. P. 245–250.
13. Cho Y. H., Mouftan H. T. A CMOS ternary ROM chip // Proc. 18th IEEE Int. Symp. on Multiple-Valued Logic. 1988. P. 358–363.
14. Kawahito S., Ishida M., Nakamura T., Kameyama M. High-speed area-efficient multiplier design using multiple-valued current-mode circuits // IEEE Trans. Computers. 1994. V. C-28. No. 1. P. 34–41.
15. Lo H., Lee S. C. Fault detection in m -logic circuits using the m -difference // Proc. 14th IEEE Int. Symp. on Multiple-Valued Logic. 1984. P. 62–70.
16. Miller P. J., Taylor G. E. Fault modelling and automatic test generation for three-level system // Proc. 14th IEEE Int. Symp. on Multiple-Valued Logic. 1984. P. 71–77.
17. Spillman R. J., Su S. Y. H. Detection of single, stuck-type failures in multivalued combinational networks // IEEE Trans. Computers. 1977. V. C-26. No. 12. P. 1242–1251.
18. Tapia M. A., Guima T. A., Katbab A. Calculus for a multivalued logic algebraic system // Applied Math. and Computation. 1991. P. 225–285.
19. Whitney M., Muzio J. Decisive differences and partial differences for stuck-at-fault detection in MVL circuits // IEEE Proc. 18th Int. Symp. on Multiple-Valued Logic. 1988. P. 321–328.

20. *Coy W., Moraga C.* Description and detection of faults in multiple-valued logic circuits // Proc. 9th IEEE Int. Symp. on Multiple-Valued Logic. 1979. P. 74–81.
21. *Chang Y. J., Lee C. L., Chen J. E.* Fault models for the multi-valued current mode CMOS circuit // Presented at Proc. 26th IEEE Int. Symp. on Multiple-Valued Logic. 1996.
22. *Chiang K. W., Vranesic Z. G.* Fault detection in ternary nMOS and CMOS circuits // Proc. 12th IEEE Int. Symp. on Multiple-Valued Logic. 1982. P. 129–138.
23. *Yanushkevich S., Levashenko V., Moraga C.* Fault models and their classification for MVL gates and circuits // Proc. 4th Int. Conf. Applications on Computer Systems, Szczecin, Poland. 1997. P. 309–314.
24. *Vranesic Z., Lee S., Smith K.* A many-valued algebra for switching systems // IEEE Trans. Computer. 1970. V. C-19. No. 10. P. 964–971.
25. *Allen C. M., Givone D. D.* The Allen-Givone implementation oriented algebra // Computer Science and Multiple-Valued Logic (Ed. Rine D.C.), North Holland, Amsterdam, 1977.
26. *Tabakow I.* Using *D*-algebra to generate tests for *m*-logic combinational circuits // Int. J. Electronics. 1993. V. 75. No. 5. P. 897–906.
27. *Shmerko V., Yanushkevich S.* Fault detection in MVL networks by new type of derivatives of MVL functions // Proc. European Conf. on Circuit Theory and Design, Switzerland. 1993. P. 643–646.
28. *Wang H. M., Lee C. L., Chen J. E.* Complete test set generation for multiple-valued logic networks // Int. J. on Multiple-Valued Logic. 1996. V. 1. P. 185–218.
29. *Damarla T.* Generalized transforms for multiple-valued logic circuits and their fault detection // IEEE Trans. Computers. 1992. V. C-41. No. 9. P. 1101–1109.
30. *Бохманн Д., Постхофф Х.* Двоичные динамические системы. М.: Энергоатомиздат, 1986.
31. *Shmerko V., Yanushkevich S., Levashenko V.* Test pattern generation for combinational MVL networks based on generalized *D*-algorithm // Proc. 27th IEEE Int. Sym. on Multiple-Valued Logic. 1997. P. 139–144.
32. *Levashenko V. G., Yanushkevich S. N., Zaitseva E. N.* Synthesis of parallel algorithms to compute derivatives of MVL-functions // Proc. of Workshop on Sampling Theory and Applications. Riga. Latvia. 1995. P. 319–322.
33. *Shmerko V., Yanushkevich S., Levashenko V., Bondar I.* Techniques of computing logical derivatives for MVL-functions // Proc. 26th IEEE Int. Symp. on Multiple-Valued Logic. 1996. P. 267–272.
34. *Yanushkevich S.* Logic differential calculus in multi-valued logic design. Poland: Technical Univ. of Szczecin. Academic Publishers, 1998.
35. *Stephen Y. H. Su, Yon-Chung Cho.* A new approach to the fault location of combinational circuits // IEEE Trans. Computer. 1972. V. C-21. No. 1. P. 21–30.
36. *Keim M., Goeckel N., Drechler R., Becker B.* Test generation for (sequential) multi-valued logic networks based on genetic algorithm // Proc. 28th IEEE Int. Symp. on Multiple-Valued Logic. 1998. P. 266–270.
37. *Davio M., Deschamps J. P., Thayse A.* Discrete and switching functions. McGraw-Hill Int. Book Co. 1978.
38. *Левашенко В. Г., Шмерко В. П., Янушкевич С. Н.* Параллельные алгоритмы вычисления направленных логических производных многозначных ФАЛ // Кибернетика и системный анализ. 1996. № 6. С. 41–58.

Статья представлена к публикации членом редколлегии П. П. Пархоменко.

Поступила в редакцию 20.07.98