
КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ

УДК 658:512.011:681.326:519.713

ДЕДУКТИВНЫЙ АНАЛИЗ И ДИАГНОСТИРОВАНИЕ ЛОГИЧЕСКИХ X-ФУНКЦИЙ

*ХАХАНОВ В.И., ЕМЕЛЬЯНОВ И.В.,
ЛЮБАРСКИЙ М.М., ЧУМАЧЕНКО С.В.,
ЛИТВИНОВА Е.И., ЛАРЧЕНКО Л.В.*

Предлагается структурная модель взаимодействия X-функций и производных компонентов, ориентированных на синтез и анализ цифровых систем в целях уменьшения времени проектирования и тестирования вычислительных устройств. Вводится понятие простых X-функций от конечного числа переменных, которые характеризуются отсутствием минимизации и наличием свойств тестопригодности для решения задач синтеза тестов, моделирования и диагностирования. Формулируются метрические свойства X-функций от конечного числа переменных для оценки качества проверяющих тестов путем дедуктивного моделирования проверяемых константных неисправностей на кубитных структурах данных. Предлагается аналитическое выражение для синтеза кубитных покрытий X-функций от конечного числа переменных в целях последующего синтеза и анализа тестов проверки и диагностирования неисправностей. Синтезируются дедуктивные формулы транспортирования входных списков неисправностей на внешние выходы для X-функций от конечного числа переменных, для построения секвенсора моделирования дефектов, инвариантного к входным тестовым наборам. Предлагается кубитный и квантовый методы безусловного диагностирования дефектов, использующие параллельные логические операции для повышения быстродействия online поиска дефектов.

Ключевые слова: проектирование, тестирование, кубитное моделирование, кубитное диагностирование, X-функция, кубитные структуры данных, кубитное покрытие.

Key words: design, test, qubit simulation, qubit diagnosis, X-function, qubit data structure, qubit coverage.

1. Введение

Дедуктивный анализ цифровых систем и компонентов является самым эффективным аппаратом для синтеза тестов и оценки их качества, а также для диагностирования неисправностей в режиме реального времени. Сущность дедуктивного моделирования заключается в использовании синтези-

рованных цифровых подходов, которые являются дополнением к исходной функциональности, предназначенных для транспортирования к внешним выходам негативных реакций, возникающих в схеме неисправностей. Как правило, сложность дедуктивных подходов, реализуемых программно или аппаратно, в несколько раз выше, чем полезная функциональность. Поэтому эффективность применения дедуктивного анализа следует рассматривать с позиции уменьшения интегрального времени проектирования цифровой системы и выхода изделия на рынок электронных технологий.

Цель исследования – уменьшение времени проектирования цифровой системы на кристалле за счет синтеза дедуктивных функций логических компонентов и схем, включая X-функции, представленных кубитными формами описания поведения цифровых устройств.

Задачи: 1) Разработать структурную модель взаимодействия X-функций и производных компонентов, ориентированных на синтез и анализ цифровых систем. 2) Определить простые X-функции от конечного числа переменных, для синтеза тестов, моделирования и диагностирования цифровых устройств. 3) Сформулировать метрические свойства X-функций от конечного числа переменных для оценки качества проверяющих тестов на кубитных структурах данных. 4) Создать аналитическое выражение для синтеза кубитных покрытий X-функций от конечного числа переменных в целях анализа тестов проверки и диагностирования неисправностей. 5) Синтезировать дедуктивные формулы транспортирования входных списков неисправностей для X-функций от конечного числа переменных. 6) Разработать кубитный и квантовый методы безусловного диагностирования дефектов на основе использования параллельных логических операций.

2. Дедуктивные функции параллельного моделирования неисправностей

Дедуктивный анализ используется для определения качества теста относительно введенного класса неисправностей, как правило, одиночных константных. Существует развитая теория дедуктивного анализа [4], ориентированная на параллельную обработку списков неисправностей. Базовые понятия теории моделирования неисправностей представлены аппаратом транспортирования спис-

ков дефектов через примитивные функциональные логические элементы [13-19]. Далее определены дедуктивные функции параллельного моделирования неисправностей на исчерпывающем тесте для функциональных элементов and, or, not. Получение дедуктивного преобразователя для функции and:

$$\begin{aligned}
 L[T = (00,01,10,11), F = (X_1 \wedge X_2)] &= \\
 &= L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge [(X_1 \oplus T_{t1} \wedge X_2 \oplus T_{t2}) \oplus T_{t3}]\} = \\
 &= (\bar{x}_1\bar{x}_2)\{(X_1 \oplus 0) \wedge (X_2 \oplus 0)\} \oplus 0 \vee (\bar{x}_1x_2)\{(X_1 \oplus 0) \wedge (X_2 \oplus 1)\} \oplus 0 \vee \\
 &\vee (x_1\bar{x}_2)\{(X_1 \oplus 1) \wedge (X_2 \oplus 0)\} \oplus 0 \vee (x_1x_2)\{(X_1 \oplus 1) \wedge (X_2 \oplus 1)\} \oplus 1 = \\
 &= (\bar{x}_1\bar{x}_2)(X_1 \wedge X_2) \vee (\bar{x}_1x_2)(X_1 \wedge \bar{X}_2) \vee (x_1\bar{x}_2)(\bar{X}_1 \wedge X_2) \vee (x_1x_2)(X_1 \vee X_2)
 \end{aligned}$$

Аналогично выполняются вычисления для функции or:

$$\begin{aligned}
 L[T = (00,01,10,11), F = (X_1 \vee X_2)] &= \\
 &= L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge [(X_1 \oplus T_{t1} \vee X_2 \oplus T_{t2}) \oplus T_{t3}]\} = \\
 &= (\bar{x}_1\bar{x}_2)\{(X_1 \oplus 0) \vee (X_2 \oplus 0)\} \oplus 0 \vee (\bar{x}_1x_2)\{(X_1 \oplus 0) \vee (X_2 \oplus 1)\} \oplus 1 \vee \\
 &\vee (x_1\bar{x}_2)\{(X_1 \oplus 1) \vee (X_2 \oplus 0)\} \oplus 1 \vee (x_1x_2)\{(X_1 \oplus 1) \vee (X_2 \oplus 1)\} \oplus 1 = \\
 &= (\bar{x}_1\bar{x}_2)(X_1 \vee X_2) \vee (\bar{x}_1x_2)(\bar{X}_1 \wedge X_2) \vee (x_1\bar{x}_2)(X_1 \wedge \bar{X}_2) \vee (x_1x_2)(X_1 \wedge X_2)
 \end{aligned}$$

Здесь $T_t = (T_{t1}, T_{t2}, T_{t3})$, $(t = \overline{1,4})$ – тест-вектор, имеющий 3 координаты, где последняя из них определяет состояние выхода двухвходового элемента and (or); L – выходной список неисправностей; X – список неисправностей на конкретном входе примитива; $x = \{0,1\}$ – логическое значение на входе примитива. В следующем преобразовании $T_t = (T_{t1}, T_{t2})$, $(t = \overline{1,2})$ – тест-вектор, имеющий 2 координаты, где вторая – состояние выхода инвертора:

$$\begin{aligned}
 L[T = (0,1), F = \bar{X}_1] &= L\{(\bar{x}_1 \vee x_1)[(\bar{X}_1 \oplus T_{t1}) \oplus T_{t2}]\} = \\
 &= \bar{x}_1[(\bar{X}_1 \oplus 0) \oplus 1] \vee x_1[(\bar{X}_1 \oplus 1) \oplus 0] = \bar{x}_1\bar{X}_1 \vee x_1\bar{X}_1 = \bar{x}_1X_1 \vee x_1X_1.
 \end{aligned}$$

Последнее выражение иллюстрирует инвариантность инверсии ко входному набору для транспортирования дефектов. Она трансформируется в повторитель. Поэтому данная функция не фигурирует на выходах дедуктивных элементов. Совместная аппаратная реализация дедуктивных функций для оставшихся двухвходовых элементов and, or на исчерпывающем тесте представлена универсальной схемой (рис. 1) дедуктивно-параллельного анализа неисправностей.

В симуляторе представлены булевы (x1, x2) и регистровые (X1, X2) для кодирования неисправностей входы, переменная V выбора типа исправной функции (and, or), выходная регистровая переменная Y. Состояния двоичных входов x1, x2 и переменная выбора элемента определяют одну из че-

тырех дедуктивных функций для получения вектора Y проверяемых неисправностей.

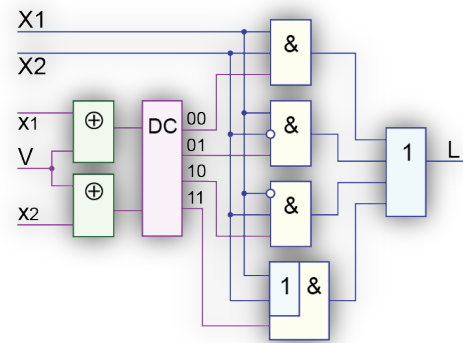


Рис. 1. Симулятор неисправностей примитивов
Логическая функция хог обладает уникальным свойством транспортирования входных списков неисправностей. Далее рассматривается синтез дедуктивной модели данной функции на входных тестовых последовательностях:

$$\begin{aligned}
 L[T = (00,01,10,11), F = X_1 \oplus X_2 = X_1\bar{X}_2 \vee \bar{X}_1X_2] &= \\
 &= L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge \\
 &\wedge [(X_1 \oplus T_{t1})(\bar{X}_2 \oplus T_{t2}) \vee (\bar{X}_1 \oplus T_{t1})(X_2 \oplus T_{t2})] \oplus T_{t3}\} :
 \end{aligned}$$

$$\begin{aligned}
 L(T_0 = 00) &= (\bar{x}_1\bar{x}_2) \wedge [(X_1 \oplus 0)(\bar{X}_2 \oplus 0) \vee (\bar{X}_1 \oplus 0)(X_2 \oplus 0)] \oplus 0 \vee \\
 &= (\bar{x}_1\bar{x}_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)];
 \end{aligned}$$

$$\begin{aligned}
 L(T_1 = 01) &= (\bar{x}_1x_2) \wedge [(X_1 \oplus 0)(\bar{X}_2 \oplus 1) \vee (\bar{X}_1 \oplus 0)(X_2 \oplus 1)] \oplus 1 \vee \\
 &= (\bar{x}_1x_2) \wedge [(X_1\bar{X}_2) \vee (\bar{X}_1\bar{X}_2)] \oplus 1 \vee \\
 &= (\bar{x}_1x_2) \wedge [(\bar{X}_1 \vee \bar{X}_2) \wedge (X_1 \vee X_2)] = (\bar{x}_1x_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)];
 \end{aligned}$$

$$\begin{aligned}
 L(T_2 = 10) &= (x_1\bar{x}_2) \wedge [(X_1 \oplus 1)(\bar{X}_2 \oplus 0) \vee (\bar{X}_1 \oplus 1)(X_2 \oplus 0)] \oplus 1 \vee \\
 &= (x_1\bar{x}_2) \wedge [(\bar{X}_1\bar{X}_2) \vee (X_1X_2)] \oplus 1 \vee \\
 &= (x_1\bar{x}_2) \wedge [(X_1 \vee X_2) \wedge (\bar{X}_1 \vee \bar{X}_2)] = (x_1\bar{x}_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)];
 \end{aligned}$$

$$\begin{aligned}
 L(T_3 = 11) &= (x_1x_2) \wedge [(X_1 \oplus 1)(\bar{X}_2 \oplus 1) \vee (\bar{X}_1 \oplus 1)(X_2 \oplus 1)] \oplus 0 \vee \\
 &= (x_1x_2) \wedge [(\bar{X}_1X_2) \vee (X_1\bar{X}_2)] \oplus 0 = (x_1x_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)].
 \end{aligned}$$

В результате синтеза оказалось, что все дедуктивные формулы на четырех входных тестовых наборах имеют одинаковый вид, что означает инвариантность хог-функции к входным тестовым наборам – любое входное слово имеет одну единственную дедуктивную формулу

$$\begin{aligned}
 L(\text{хог}) &= (\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)] = \\
 &= (xx) \wedge [(X_1\bar{X}_2 \vee \bar{X}_1X_2)].
 \end{aligned}$$

Это означает, что списки входных неисправностей всегда будут объединяться на выходе хог-элемента за исключением случая, когда данные списки будут идентичными. В этом случае на выходе логического элемента хог не будет проверяться ни одна неисправность.

Существуют и другие функции, обладающие единственной универсальной дедуктивной формулой для всех входных тестовых воздействий. Ниже предлагается синтез дедуктивной формулы для операции эквивалентности:

$$L[T = (00, 01, 10, 11), F = X_1 \equiv X_2 = X_1 X_2 \vee \bar{X}_1 \bar{X}_2] = \\ = L\{(\bar{x}_1 \bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \bar{x}_2 \vee x_1 x_2) \wedge \\ \wedge [(X_1 \oplus T_{11})(X_2 \oplus T_{12}) \vee (\bar{X}_1 \oplus T_{11})(\bar{X}_2 \oplus T_{12})] \oplus T_{13}\};$$

$$L(T_0 = 00) (\bar{x}_1 \bar{x}_2) \wedge [(X_1 \oplus 0)(X_2 \oplus 0) \vee (\bar{X}_1 \oplus 0)(\bar{X}_2 \oplus 0)] \oplus 1\} = \\ = (\bar{x}_1 \bar{x}_2) \wedge [(X_1 \bar{X}_2) \vee (\bar{X}_1 X_2)] \oplus 1 = (\bar{x}_1 \bar{x}_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)];$$

$$L(T_1 = 01) (\bar{x}_1 x_2) \wedge [(X_1 \oplus 0)(X_2 \oplus 1) \vee (\bar{X}_1 \oplus 0)(\bar{X}_2 \oplus 1)] \oplus 0\} = \\ = (\bar{x}_1 x_2) \wedge [(X_1 \bar{X}_2) \vee (\bar{X}_1 X_2)] \oplus 0 = (\bar{x}_1 x_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)];$$

$$L(T_2 = 10) (x_1 \bar{x}_2) \wedge [(X_1 \oplus 1)(X_2 \oplus 0) \vee (\bar{X}_1 \oplus 1)(\bar{X}_2 \oplus 0)] \oplus 0\} = \\ = (x_1 \bar{x}_2) \wedge [(X_1 \bar{X}_2) \vee (\bar{X}_1 X_2)] \oplus 0 = (x_1 \bar{x}_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)];$$

$$L(T_3 = 11) (x_1 x_2) \wedge [(X_1 \oplus 1)(X_2 \oplus 1) \vee (\bar{X}_1 \oplus 1)(\bar{X}_2 \oplus 1)] \oplus 1\} = \\ = (x_1 x_2) \wedge [(X_1 \bar{X}_2) \vee (\bar{X}_1 X_2)] \oplus 1 = (x_1 x_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)].$$

Таким образом, функция эквивалентности (not-хог) также обладает уникальными способностями по транспортированию входных списков неисправностей на выход логической функции, которые не зависят от входного воздействия:

$$L(\text{not} - \text{хог}) = (\bar{x}_1 \bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \bar{x}_2 \vee x_1 x_2) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2) \\ = (xx) \wedge [(X_1 \bar{X}_2 \vee \bar{X}_1 X_2)].$$

В результате получилось, что дедуктивные формулы для двух различных логических функций хог и not-хог идентичны друг другу. Однако более существенным является факт, что любое тестовое воздействие для цифрового устройства, состоящего из упомянутых логических элементов, проверяет 50 процентов одиночных константных неисправностей, инверсных по отношению к исправному состоянию входных линий. Чтобы проверить все неисправности входных линий, необходимо подать на схему только два тестовых взаимно-инверсных набора.

Формула для определения производных входных булевых переменных использует хог-операцию между ячейками соседних частей кубита:

$$Q^k(X_k) = \{Q_i^L, Q_i^R\} = Q_i^L \bigoplus_{i=1, 2^{k-1}}^{k=1, n} Q_i^R.$$

Структура вычислителя-секвенсора кубитной производной представлена на рис. 2, где результаты выполнения хог-операции заносятся в обе части кубита.

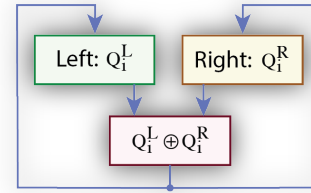


Рис. 2. Секвенсор кубитной производной

В общем случае можно сформулировать критерий существования дедуктивных функций, инвариантных по отношению к входным воздействиям: “Если кубитные производные по всем переменным равны единичному вектору, то дедуктивная формула инвариантна к входным тестовым наборам”. Для четырех логических функций от одной переменной существуют только два таких примитива, которые имеют одинаковые единичные векторы производных, получаемые путем выполнения хог-операции на встречном сдвиге частей кубита:

$$Y = X; Q = 01, Q' = 11;$$

$$Y = \bar{X}; Q = 10, Q' = 11.$$

Для шестнадцати логических функций от двух переменных существуют также два примитива, которые имеют одинаковые единичные векторы производных, получаемые путем выполнения хог-операции на встречном сдвиге частей кубита:

$$Y = X_1 \oplus X_2; Q = 0110; Q' = 1111;$$

$$Y = X_1 \oplus X_2; Q = 1001; Q' = 1111.$$

Для 256 логических функций от трех переменных существуют также два примитива, которые имеют одинаковые единичные векторы производных, получаемые путем выполнения хог-операции на встречном сдвиге частей кубита:

$$Y = X_1 \oplus X_2; Q = 01101001; Q' = 11111111;$$

$$Y = X_1 \oplus X_2; Q = 10010110; Q' = 11111111.$$

Здесь два кубита X-функций от любого конечного числа переменных представляют собой инверсии состояний соответствующих разрядов. Каждый из них может быть разделен на две равные по числу разрядов части, где левая является инверсией правой части:

$Q(n=1) = 01$
 (10);
 $Q(n=2) = 0110$
 (1001);
 $Q(n=3) = 01101001$
 (10010110);
 $Q(n=4) = 0110100110010110$
 (1001011001101001);
 $Q(n=5) = 0110100110010110 1001011001101001$
 (1001011001101001 0110100110010110).

Утверждение 1. Множество $|f(X)| = 2^{2^n}$ логических функций от конечного числа переменных n имеет только две функции, для которых кубитные производные определены как единичные векторы.

Утверждение 2. Количество логических функций от n переменных, где выполняется условие

$$\bigvee_{i=1}^n \left[\frac{df}{dX_i} = 1 \right],$$

всегда равно двум. Это означает, что не нужно никаких условий по другим входам для активизации выхода при изменении входной переменной X_i .

Определение. X-функция есть логическая простая функция от конечного числа переменных ($n=1,2,3, \dots$), которую невозможно минимизировать.

Утверждение 3. Шахматное представление нулевых и единичных состояний в карте Карно задает X-функцию. Следующие два примера логических функций, изображенных в форме карты Карно, иллюстрируют две взаимно инверсных булевых X-функции от четырех переменных:

$X_1 X_2 \setminus X_2 X_3$	00	01	11	10	$X_1 X_2 \setminus X_2 X_3$	00	01	11	10
00	0	1	0	1	00	1	0	1	0
01	1	0	1	0	01	0	1	0	1
11	0	1	0	1	11	1	0	1	0
10	1	0	1	0	10	0	1	0	1

$$Y(0110100110010110) = \bar{X}_1 \bar{X}_2 \bar{X}_3 X_4 \vee \bar{X}_1 \bar{X}_2 X_3 \bar{X}_4 \vee \bar{X}_1 X_2 \bar{X}_3 \bar{X}_4 \vee \bar{X}_1 X_2 X_3 X_4 \vee X_1 \bar{X}_2 \bar{X}_3 X_4 \vee X_1 X_2 X_3 \bar{X}_4 \vee X_1 \bar{X}_2 \bar{X}_3 X_4 \vee X_1 \bar{X}_2 X_3 X_4;$$

$$Y(1001011001101001) = \bar{X}_1 \bar{X}_2 \bar{X}_3 \bar{X}_4 \vee \bar{X}_1 \bar{X}_2 X_3 X_4 \vee \bar{X}_1 X_2 \bar{X}_3 X_4 \vee \bar{X}_1 X_2 X_3 \bar{X}_4 \vee X_1 \bar{X}_2 \bar{X}_3 \bar{X}_4 \vee X_1 \bar{X}_2 X_3 X_4 \vee X_1 \bar{X}_2 \bar{X}_3 X_4 \vee X_1 \bar{X}_2 X_3 \bar{X}_4.$$

Аналогично, с помощью карт Карно, можно представить логические X-функции от трех переменных, которые имеют 8 состояний выходов:

$X_1 \setminus X_2 X_3$	00	01	11	10	$X_1 \setminus X_2 X_3$	00	01	11	10
0	0	1	0	1	0	1	0	1	0
1	1	0	1	0	1	0	1	0	1

Данным функциям можно поставить в соответствие их аналитические формы, которые облада-

ют полезным свойством – каждый терм СДНФ (Совершенная Дизъюнктивная Нормальная Форма) отличается от остальных по двум переменным:

$$Y(01101001) = \bar{X}_1 \bar{X}_2 X_3 \vee \bar{X}_1 X_2 \bar{X}_3 \vee X_1 \bar{X}_2 \bar{X}_3 \vee X_1 X_2 X_3;$$

$$Y(10010110) = \bar{X}_1 \bar{X}_2 \bar{X}_3 \vee \bar{X}_1 X_2 X_3 \vee X_1 \bar{X}_2 X_3 \vee X_1 X_2 \bar{X}_3.$$

Это означает, что любая одиночная константная неисправность может быть идентифицирована и исправлена на проверяющем входном наборе. Структурная реализация упомянутых выше булевых логических функций СДНФ представлена на рис. 3.

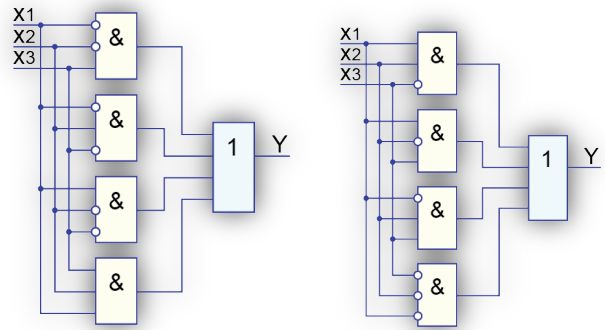


Рис. 3. Структурная реализация булевых X-функций

Моделирование исправного поведения всех входных наборов (таблица T) для первой X-схемы, анализ неисправностей (таблица D), минимизация тестовых последовательностей (таблица M) и кубитная форма минимального теста – таблица T(Q), представлены ниже:

T	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	0	0
1	0	0	1	1	0	0	0	1
2	0	1	0	0	1	0	0	1
3	0	1	1	0	0	0	0	0
4	1	0	0	0	0	1	0	1
5	1	0	1	0	0	0	0	0
6	1	1	0	0	0	0	0	0
7	1	1	1	0	0	0	1	1

D	1	2	3	4	5	6	7	8
0	1	1	1	1	1	1	1	1
1	1	1	0	0	0	
2	1	0	1	0	
3	1	0	0	1	1	1	1	1
4	0	1	1	0	
5	0	1	0	1	1	1	1	1
6	0	0	1	1	1	1	1	1
7	0	0	0	0	0

M	1	2	3	4	5	6	7	8
0	1	1	1	1	1	1	1	1
1	1	1	0	0	
2	1	0	1	0	
4	0	1	1	0	
7	0	0	0	0	0
C	x	x	x	x	x	x	x	x

T(Q)
1
1
1
0
1
0
0
1

Здесь каждый входной набор проверяет одиночные константные неисправности внешних входов и выходов, что зафиксировано в таблице D. Минимальный тест, покрывающий все одиночные кон-

стантные неисправности, представлен в таблице М. В последней строке представлен результат покрытия неисправностей в виде вектора С, координаты которого заданы символами $x=0\cup 1$. Столбец $T(Q) = 11101001$ своими единичными значениями создает минимальную кубитную форму теста (двоичные адреса единичных координат), который необходимо подать на внешние входы, чтобы проверить все одиночные константные неисправности внешних и внутренних линий цифровой логической схемы.

Моделирование исправного поведения всех входных наборов (таблица Т) для второй Х-схемы, анализ неисправностей (таблица D), минимизация тестовых последовательностей (таблица М) и кубитная форма минимального теста (таблица T(Q)), представлены ниже:

T	1	2	3	4	5	6	7	8
0	0	0	0	0	0	0	1	1
1	0	0	1	0	0	0	0	0
2	0	1	0	0	0	0	0	0
3	0	1	1	0	0	1	0	1
4	1	0	0	0	0	0	0	0
5	1	0	1	0	1	0	0	1
6	1	1	0	1	0	0	0	1
7	1	1	1	0	0	0	0	0

D	1	2	3	4	5	6	7	8
0	1	1	1	.	.	.	0	0
1	1	1	0	1	1	1	1	1
2	1	0	1	1	1	1	1	1
3	1	0	0	.	0	.	0	.
4	0	1	1	1	1	1	1	1
5	0	1	0	.	0	.	.	0
6	0	0	1	0	.	.	.	0
7	0	0	0	1	1	1	1	1

M	1	2	3	4	5	6	7	8
0	1	1	1	.	.	.	0	0
1	1	1	0	1	1	1	1	1
3	1	0	0	.	0	.	0	.
5	0	1	0	.	0	.	.	0
6	0	0	1	0	.	.	.	0
C	x	x	x	x	x	x	x	x

T(Q)
1
1
0
1
0
1
1
0

Здесь каждый входной набор проверяет одиночные константные неисправности внешних входов и выходов, что зафиксировано в таблице D. Минимальный тест, покрывающий все одиночные константные неисправности, представлен в таблице М. В последней строке представлен результат покрытия неисправностей в виде вектора С, координаты которого заданы символами $x=0\cup 1$. Столбец $T(Q) = 110101101$ своими единичными значениями создает минимальную кубитную форму теста (двоичные адреса единичных координат), который необходимо подать на внешние входы, чтобы проверить все одиночные константные неисправности внешних и внутренних линий цифровой логической схемы.

Для двух логических Х-функций от двух переменных (рис. 4)

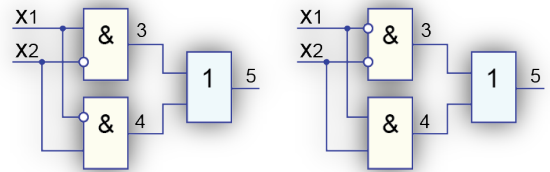


Рис. 4. Х-функции от двух переменных

ниже представлено моделирование исправного поведения всех входных наборов (таблица Т), анализ неисправностей (таблица D) и кубитная форма четырех вариантов минимальных тестов (таблица T(Q)), рассмотрены функциональные элементы (xor, not-xor – nxr):

T(xor)	1	2	3	4	5
0	0	0	0	0	0
1	0	1	0	1	1
2	1	0	1	0	1
3	1	1	0	0	0

D	1	2	3	4	5	
0	1	1	1	1	1	
1	1	1	0	.	0	0
2	0	1	0	.	0	
3	0	0	1	1	1	

T(Q)	1	2	3	4
0	0	1	1	1
1	1	0	1	1
2	1	1	0	1
3	1	1	1	0

Для обеих функций здесь получены минимальные тесты, состоящие из трех входных наборов. Это связано с тем, что противоположные входные векторы имеют одинаковые состояния выходной переменной.

Для двух логических Х-функций от одной переменной (рис. 5)

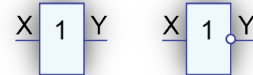


Рис. 5. Х-функции от одной переменной

ниже представлено моделирование исправного поведения всех входных наборов (таблица Т), анализ неисправностей (таблица D) и кубитная форма минимальных тестов (таблица T(Q)), рассмотрены функциональные элементы (повторитель – гер, инвертор – not):

T(rep)	X	Y
0	0	0
1	1	1

D	X	Y
0	1	1
1	0	0

T(Q)
1
1

T(not)	X	Y
0	0	1
1	1	0

D	X	Y
0	1	0
1	0	1

T(Q)
1
1

Практически полезными для синтеза и анализа цифровых схем могут быть следующие 10 свойств

X-функций, интегрированные в модель отношений, представленную на рис. 6.

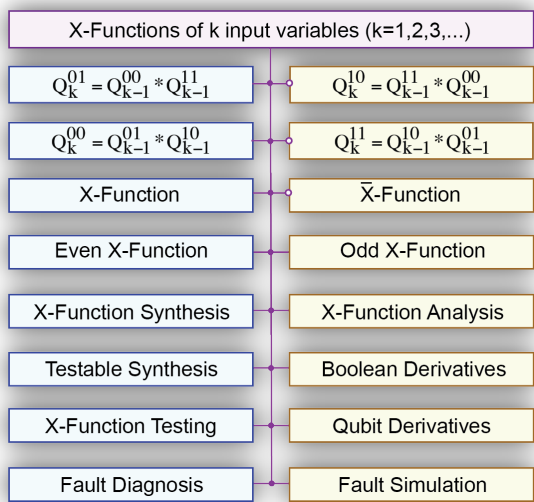


Рис. 6. Структурная модель взаимодействия X-функций

- 1) Кубитное покрытие X-функции имеет равное количество нулевых и единичных координат.
- 2) Количество X-функций от n булевых переменных всегда равно двум:

$$Q^{2^x}(n) = Q^x(n) \vee \bar{Q}^x(n).$$

Состояния координат кубитных покрытий обеих X-функций от n переменных являются взаимно инверсными.

- 3) X-функции от одной логической булевой переменной представлены повторителем и инвертором: $Y = X; Y = \bar{X}$.

- 4) X-функции от двух булевых переменных представлены известными логическими примитивами xor, not-xor: $Y = X_1 \bar{X}_2 \vee \bar{X}_1 X_2; Y = X_1 X_2 \vee \bar{X}_1 \bar{X}_2$.

- 5) Кубитная производная по любой переменной X-функции равна единичному вектору. Булева производная по любой переменной X-функции равна единице.

- 6) Для активизации входной переменной X-функции в целях изменения выхода не требуется никаких условий по состоянию других переменных.

- 7) Пара входных наборов, имеющая инверсные сигналы по всем координатам, всегда изменяет состояние выхода X-функции от нечетного числа переменных. Изменение состояния входа X-функции всегда приводит к изменению состояния выхода.

- 8) Синтез двух X-функций от n переменных реализуется путем конкатенации (*) кубитных векторов X-функций от n-1 переменной:

$$Q^{2^x}(n) = Q^x(n-1) * \bar{Q}^x(n-1) \vee \bar{Q}^x(n-1) * Q^x(n-1).$$

Другая интерпретация X-функций связана с их идентификацией по первому и последнему биту кубитного покрытия: 01, 10, 00, 11. Тогда синтез кубитных векторов X-функций от k переменных можно реализовать путем применения операции конкатенации к двум Q-покрытиям X-функций от k-1 переменных:

$$\begin{aligned} Q_k^{01} &= Q_{k-1}^{00} * Q_{k-1}^{11}; \\ Q_k^{10} &= Q_{k-1}^{11} * Q_{k-1}^{00}; \\ Q_k^{00} &= Q_{k-1}^{01} * Q_{k-1}^{10}; \\ Q_k^{11} &= Q_{k-1}^{10} * Q_{k-1}^{01}. \end{aligned}$$

Естественно, что между двумя кубитными покрытиями X-функций от k переменных существует взаимно-однозначное отношение инверсии:

$$\begin{aligned} Q_k^{01} &= \bar{Q}_k^{10}; \\ Q_k^{10} &= \bar{Q}_k^{01}; \\ Q_k^{00} &= \bar{Q}_k^{11}; \\ Q_k^{11} &= \bar{Q}_k^{00}. \end{aligned}$$

Это означает, если известна одна X-функция от k переменных, то легко можно получить вторую функцию, как двоичное дополнение к первой.

- 9) Любой входной набор для X-функции проверяет 50% неисправностей по внешним входам, которые являются инверсными по отношению к состояниям исправного поведения упомянутых линий. Два взаимно-инверсных тестовых набора проверяют все одиночные константные неисправности входных переменных в X-функции от нечетного числа переменных.

- 10) Дедуктивная формула X-функции транспортирует на выход симметрическую разность входных списков неисправностей. Это означает объединение входных списков проверяемых дефектов за исключением случая, когда списки неисправностей на всех входах идентичны.

Таким образом, логические X-функции, обладающие уникальными свойствами тестирования, могут быть использованы для синтеза тестопригодных и самовосстанавливаемых логических цифровых устройств, а также для транспортирования дефектов от внешних входов до выходов булевой структуры.

3. Квантовый метод безусловного поиска дефектов

Для второй X-схемы выполним диагностический эксперимент по построенной таблице неисправностей в целях определения константных дефектов, если реакция устройства на тест зафиксирована в виде: R= (11001100):

Q _{ij}	1	2	3	4	5	6	7	8	R
0	1	1	1	.	.	.	0	0	1
1	1	1	0	1	1	1	1	1	1
2	1	0	1	1	1	1	1	1	0
3	1	0	0	.	.	0	.	0	0
4	0	1	1	1	1	1	1	1	1
5	0	1	0	.	0	.	.	0	1
6	0	0	1	0	.	.	.	0	0
7	0	0	0	1	1	1	1	1	0
UF(1)	x	1	x	1	x	1	x	x	1
UF(0)	x	0	x	x	1	x	1	x	0
∩F(1)	.	1	1s
UF(1)∖UF(0)	.	1	.	.	0	.	0	.	m
∩F(1)∖UF(0)	.	1	s

Здесь реализован квантовый метод поиска константных дефектов, основная идея или сущность которого определяется следующими выражениями для диагностирования кратных (m) и одиночных (s) неисправностей соответственно:

$$F = \left(\bigcup_{\forall R_i=1} Q_{ij} \right) \setminus \left(\bigcup_{\forall R_i=0} Q_{ij} \right) = \left(\bigvee_{\exists R_i=1} Q_{ij} \right) \wedge \overline{\left(\bigvee_{\exists R_i=0} Q_{ij} \right)}$$

$$F = \left(\bigcap_{\forall R_i=1} Q_{ij} \right) \setminus \left(\bigcup_{\forall R_i=0} Q_{ij} \right) = \left(\bigwedge_{\exists R_i=1} Q_{ij} \right) \wedge \overline{\left(\bigvee_{\exists R_i=0} Q_{ij} \right)}$$

Диагностирование по второму варианту дает один единственный дефект в схеме, равный константе 1 на втором входе:

$$F^s = \cap F(1) \setminus UF(0) = \boxed{.1.} = \{2^1\}.$$

В предположении существования кратных дефектов выполнение диагностической процедуры по первому уравнению дает результат:

$$F^m = UF(1) \setminus UF(0) = \boxed{.1. . . 0. . 0.} = \{2^1, 5^0, 7^0\}.$$

Второй вариант диагностического эксперимента зафиксировал следующий вектор экспериментальной проверки по внешнему выходу R=(011001001). Вычислительные процедуры, выполненные в соответствии с двумя приведенными выше уравнениями, дали следующие результаты:

Q _{ij}	1	2	3	4	5	6	7	8	R
0	1	1	1	.	.	.	0	0	0
1	1	1	0	1	1	1	1	1	1
2	1	0	1	1	1	1	1	1	1
3	1	0	0	.	.	0	.	0	0
4	0	1	1	1	1	1	1	1	1
5	0	1	0	.	0	.	.	0	0
6	0	0	1	0	.	.	.	0	0
7	0	0	0	1	1	1	1	1	1
UF(1)	x	x	x	1	1	1	1	1	1
UF(0)	x	x	x	0	0	0	0	0	0
∩F(1)	.	.	.	1	1	1	1	1	1s
UF(1)∖UF(0)	.	.	.	1	1	1	1	1	m
∩F(1)∖UF(0)	.	.	.	1	1	1	1	1	s

Это означает, что итоговый результат, полученный по применению обоих уравнений, получился одинаковым:

$$F^m = UF(1) \setminus UF(0) = \boxed{. . . 1 1 1 1 1 1} = \{4^1, 5^1, 6^1, 7^1, 8^1\}.$$

$$F^s = \cap F(1) \setminus UF(0) = \boxed{. . . 1 1 1 1 1 1} = \{4^1, 5^1, 6^1, 7^1, 8^1\}.$$

Данный факт означает, что полученные неисправности являются эквивалентными и неразличимыми на полном проверяющем исчерпывающем тесте, содержащем 6 наборов. Дальнейшее уменьшение множества дефектов связано с применением условного алгоритма поиска неисправностей в схеме цифрового устройства на основе использования технологий зондового диагностирования.

4. Кубитный метод диагностирования дефектов

Рассмотрим кубитный метод поиска дефектов путем применения теоретико-множественной разности двух векторов, соответствующих единичному и нулевому значению состояний выходов, как реакций наблюдаемых выходов на входной тест проверки неисправностей:

$$F = \left(\bigcup_{\forall R_i=1} Q_{ij} \right) \setminus \left(\bigcup_{\forall R_i=0} Q_{ij} \right) = \left(\bigvee_{\exists R_i=1} Q_{ij} \right) \wedge \overline{\left(\bigvee_{\exists R_i=0} Q_{ij} \right)}$$

Структуры данных представлены таблицей неисправностей на декартовом произведении тестовых наборов и множестве линий объекта диагностирования, где каждая ячейка представляет собой два бита: первый из них идентифицирует проверяемую константную неисправность нуля (10), а второй – константную неисправность единицы (01):

$$Q = \{F, T, L\},$$

$$Q = Q_{ij}, i = \overline{1, m}; j = \overline{1, n};$$

$$F = (F_1, F_2, \dots, F_j, \dots, F_n),$$

$$F_j = \{10 \equiv 0; 01 \equiv 1; 11 = \{0, 1\}; 00 = \emptyset\};$$

$$T = (T_1, T_2, \dots, T_i, \dots, T_m);$$

$$L = (L_1, L_2, \dots, L_i, \dots, L_n).$$

Суперпозиция неисправностей (две единицы на одной линии-ячейке) дает возможность существенно минимизировать структуры данных для хранения информации в целях последующего поиска дефектов при выполнении диагностического эксперимента в режиме online.

Для проверки метода поиска дефектов далее предлагается логическая схема, представленная на рис. 7, которая имеет 6 элементов and-not, 11 линий, 5 входов и два выхода.

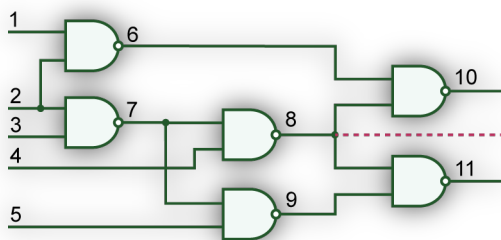


Рис. 7. ISCAS-схема для верификации

Следующая таблица иллюстрирует выполнение диагностического эксперимента в целях объединения множества дефектов, которые формируют некорректные состояния выходов на тестовых наборах {T1-R10; T5-R11; T6-R10, R11; T8-R11}:

Q = Q _{ij}	1	2	3	4	5	6	7	8	9	10	11	R ₁₀	R ₁₁
T ₁	01	10	01	00	10	00	10	10	00	10	01	1	0
T ₂	10	00	10	00	01	10	00	00	10	01	10	0	0
T ₃	00	01	01	00	00	01	10	01	01	10	10	0	0
T ₄	10	00	01	00	10	00	01	00	10	01	01	0	0
T ₅	00	10	00	01	00	01	00	10	00	10	10	0	1
T ₆	01	10	00	00	10	00	00	01	10	01	10	1	1
T ₇	01	00	00	10	00	00	01	00	10	01	01	0	0
T ₈	00	10	10	01	01	10	00	00	00	01	10	0	1
Q ₁	01	11	11	01	11	11	10	11	10	11	11	1	1
Q ₀	11	01	11	10	11	11	11	01	11	11	11	0	0
F	00	10	00	01	00	00	00	10	00	00	00	1/0	1/0

Здесь дизъюнкция строк T₁, T₅, T₆, T₈ формирует вектор Q₁, который собирает все возможные дефекты, проверяемые на тестовых наборах. Вектор Q₀ с помощью строк T₂, T₃, T₄, T₇ объединяет все невозможные, непроверяемые на тестовых наборах дефекты. Вычитание всех невозможных из всех возможных дефектов дает искомый результат в виде трех неисправностей, закодированных как F₂=10; F₄=01; F₈=10. Таким образом, параллельное выполнение двух регистровых og-операций на основе результатов проведенного диагностического эксперимента позволило определить три возмож-

ных неисправности, каждая из которых может иметь место в логической схеме:

$$F = \{2^0, 4^1, 8^0\}.$$

Более жестким является ограничительное условие существования в логической схеме одиночного константного дефекта, использование которого приводит к вычислению дефектов на основе следующего выражения:

$$F = \left(\bigcap_{\forall R_i=1} Q_{ij} \right) \wedge \left(\bigcup_{\forall R_i=0} Q_{ij} \right) = \left(\bigwedge_{\exists R_i=1} Q_{ij} \right) \wedge \left(\bigvee_{\exists R_i=0} \overline{Q_{ij}} \right).$$

Применение данной формулы существенно уточняет результат диагностирования и приводит его к

виду: $F = \{2^0\}$ за счет противоречивости кодов дефектов по and-операции в столбцах 4 и 8. Условие наличия в логической схеме одиночной константной неисправности ставит во главу угла следующее утверждение.

Утверждение: Если в столбце таблицы неисправностей существует координата 00 или 01, которая создает на наблюдаемых выходах некорректность R=1, связанную с неисправностью 10 на остальных координатах столбца, то такой одиночный дефект (10) в логической схеме невозможен.

Доказательство. Пусть на n тестовых наборах зафиксировано несовпадение на внешних выходах эталонных и реальных значений сигналов. При этом n-1 координата в рассматриваемом столбце имеет значение 10 (01) и лишь одна n-координата имеет значение 01 (10). Если предположить, что в логической схеме имеется дефект 10, то на n-координате также должен присутствовать дефект 10, который создает некорректное состояние выходов. Но по условиям моделирования такой дефект там отсутствует. Следовательно, невозможно считать, что в схеме присутствует дефект 10. Это подтверждается также формальным результатом – пустым пересечением всех координат столбца, связанных с некорректными состояниями выходов схемы:

$$F = \left(\bigwedge_{\exists R_i=1} Q_{ij} \right) = \begin{cases} 10 \wedge 10 \wedge \dots \wedge 10 \wedge 01 = 00; \\ 10 \wedge 10 \wedge \dots \wedge 10 \wedge 00 = 00. \end{cases}$$

Все сказанное относится и к состоянию n-координаты, которое идентифицируется сигналом пустого множества 00, взаимодействие с которым также делает невозможным присутствие в логической схеме одиночной константной 0-неисправности (10).

5. Выводы

- 1) Разработана структурная модель взаимодействия X-функций и производных продуктов, ориентированных на синтез и анализ цифровых систем в целях получения тестопригодных решений, связанных с уменьшением времени проектирования и тестирования вычислительных устройств.
- 2) Впервые введено понятие простых X-функций от конечного числа переменных, которые характеризуются отсутствием минимизации и наличием свойств тестопригодности, что дает возможность синтезировать цифровые устройства, технологичные для решения задач тестирования, моделирования и диагностирования.
- 3) Сформулированы метрические свойства X-функций от конечного числа переменных, которые дают возможность использовать их в практике разработки тестопригодных цифровых устройств, генерирования проверяющих тестов и оценки их качества путем дедуктивного моделирования проверяемых константных неисправностей на кубитных структурах данных.
- 4) Предложено аналитическое выражение для синтеза кубитных покрытий X-функций от конечного числа переменных, что дает возможность создавать тестопригодные логические схемы, не требующие экспоненциальных затрат на синтез и анализ тестов проверки и диагностирования неисправностей.
- 5) Получены дедуктивные формулы транспортирования входных списков неисправностей на внешние выходы для X-функций от конечного числа переменных, которые характеризуются единичными векторами производных по всем переменным, что дает возможность построить секвенсор моделирования дефектов, инвариантный к входным тестовым наборам.
- 6) На основе таблицы неисправностей, полученной путем дедуктивного моделирования, предложены кубитный и квантовый методы безусловного диагностирования дефектов, которые используют параллельные логические операции, существенно повышающие быстродействие поиска дефектов в реальном масштабе времени.

Литература: 1. *Abramovici M., Breuer M.A., Friedman A.D.* Digital System Testing and Testable Design. Comp. Sc. Press, 1998. 652 p. 2. *Fujiwara H.* Fault Simulation. Logic Testing and Design for Testability. 1. MIT Press, 1985. P. 84-108. 3. *Карибский В.В., Пархоменко П.П.,*

Согомонян Е.С., Халчев В.Ф. Основы технической диагностики. Кн. 1. М.: «Энергия», 1976. 346 с. 4. *Пархоменко П.П., Согомонян Е.С.* Основы технической диагностики. Оптимизация алгоритмов диагностирования, аппаратные средства / Под ред. П.П. Пархоменко. М.: Энергия, 1981. 320 с. 5. *Малышенко Ю.В. и др.* Автоматизация диагностирования электронных устройств / Под ред. В.П. Чипулиса. М.: Энергоатомиздат, 1986. 6. *Хананов В. И., Литвинова Е. И., Хаханова И. В., Гузь О. А.* Проектирование и тестирование цифровых систем на кристаллах. Харьков: ХНУРЭ, 2009. 484 с. 7. *Lee S. M., Lee S. E.* Static fault analysis for resilient System-on-Chip design // 2015 International SoC Design Conference (ISOCC). Gyungju. 2015. P. 5-6. 8. *Yildiz A., Gürsoy C. C., Gören S.* Fault emulation on heterogeneous architectures // 2017 International Conference on Computer Science and Engineering (UBMK). Antalya. Turkey. 2017. P. 905-910. 9. *Hosokawa T., Takano H., Yamazaki H., Yamazaki K.* A Diagnostic Fault Simulation Method for a Single Universal Logical Fault Model // 2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC). Christchurch. 2017. P. 217-218. 10. *Dhiliban C., Govindaraju S.* Fault simulation and analysis of VLSI circuit using n-detect test sets // 2016 Online Int. Conference on Green Engineering and Technologies (IC-GET). Coimbatore. 2016. P. 1-5. 11. *Jinling D., Aiqiang X.* A fault simulation method based on mutated truth table of logic gates // 2016 International Conference on Integrated Circuits and Microsystems (ICICM). Chengdu. 2016. P. 28-32. 12. *Hadjitheophanous S., Neophytou S. N., Michael M. K.* Scalable parallel fault simulation for shared-memory multiprocessor systems // 2016 IEEE 34th VLSI Test Symposium (VTS). Las Vegas, NV. 2016. P. 1-6.

Транслитерированный список литературы:

1. *Abramovici M., Breuer M.A., Friedman A.D.* Digital System Testing and Testable Design. Comp. Sc. Press, 1998. 652 p.
2. *Fujiwara H.* Fault Simulation. Logic Testing and Design for Testability. 1. MIT Press, 1985. P. 84-108.
3. *Karibskij V.V., Parhomenko P.P., Sogomonjan E.S., Halchev V.F.* Osnovy tehnikeskoj diagnostiki. Kn. 1. M.: «Jenergija», 1976. 346s.
4. *Parhomenko P.P., Sogomonjan E.S.* Osnovy tehnikeskoj diagnostiki. Optimizacija algoritmov diagnostirovanija, apparaturnye sredstva / Pod red. P.P. Parhomenko. M.: Jenergija, 1981. 320 c.
5. *Malyshenko Ju.V. i dr.* Avtomatizacija diagnostirovanija jelektronnyh ustrojstv / Pod red. V.P. Chipulisa. M.: Jenergoatomizdat, 1986.
6. *Hahanov V. I., Litvinova E. I., Hahanova I. V., Guz' O. A.* Proektirovanie i testirovanie cifrovyyh sistem na kristallah. Har'kov: HNURJe, 2009. 484 s.
7. *Lee S. M., Lee S. E.* Static fault analysis for resilient System-on-Chip design // 2015 International SoC Design Conference (ISOCC). Gyungju. 2015. P. 5-6.
8. *Yildiz A., Gürsoy C. C., Gören S.* Fault emulation on heterogeneous architectures // 2017 International Confer-

ence on Computer Science and Engineering (UBMK). Antalya, Turkey. 2017. P. 905-910.

9. *Hosokawa T., Takano H., Yamazaki H., Yamazaki K.* A Diagnostic Fault Simulation Method for a Single Universal Logical Fault Model // 2017 IEEE 22nd Pacific Rim International Symposium on Dependable Computing (PRDC). Christchurch. 2017. P. 217-218.

10. *Dhiliban C., Govindaraju S.* Fault simulation and analysis of VLSI circuit using n-detect test sets // 2016 Online Int. Conference on Green Engineering and Technologies (IC-GET). Coimbatore. 2016. P. 1-5.

11. *Jinling D., Aiqiang X.* A fault simulation method based on mutated truth table of logic gates // 2016 International Conference on Integrated Circuits and Microsystems (ICICM). Chengdu. 2016. P. 28-32.

12. *Hadjitheophanous S., Neophytou S. N., Michael M. K.* Scalable parallel fault simulation for shared-memory multi-processor systems // 2016 IEEE 34th VLSI Test Symposium (VTS). Las Vegas, NV. 2016. P. 1-6.

Поступила в редколлегию 11.02.2018

Рецензент: д-р техн. наук, проф. Меликян В.

Хаханов Владимир Иванович, д-р техн. наук, проф., главный научный сотрудник кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Науки, 14, e-mail: hahanov@icloud.com.

Емельянов Игорь Валериевич, научный сотрудник кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: путешествия. Адрес: Украина, 61166, Харьков, пр. Науки, 14, e-mail: iyemelyanov@itdelight.com.

Любарский Михаил Михайлович, соискатель кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: путешествия. Адрес: Украина, 61166, Харьков, пр. Науки, 14.

Чумаченко Светлана Викторовна, д-р техн. наук, профессор, зав. кафедрой АПВТ ХНУРЭ. Научные интересы: математическое моделирование вычислительных процессов, теория рядов, методы дискретной оптимизации, инновационные формы обучения. Адрес: Украина, 61166, Харьков, пр. Науки, 14, тел. +380577021326, e-mail: svetlana.chumachenko@nure.ua

Литвинова Евгения Ивановна, д-р техн. наук, проф. кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование цифровых систем. Хобби: музыка. Адрес: Украина, 61166, Харьков, пр. Науки, 14, тел. +380577021326, e-mail: litvinova_eugenia@icloud.com.

Ларченко Лина Викторовна, канд. техн. наук, доцент кафедры АПВТ ХНУРЭ. Научные интересы: моделирование специализированных архитектур. Адрес: Украина, 61166, Харьков, пр. Науки, 14, тел. +380577021326, e-mail: lina.larchenko@nure.ua

Hahanov Vladimir Ivanovich, Dr., Prof., Chief Scientific Officer, Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobby: football, downhill skiing. Address: Ukraine, 61166, Kharkov, Science, 14, e-mail: hahanov@icloud.com.

Iemelianov Igor Valerievich, Research Associate, Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobbies: traveling. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14, e-mail: iyemelyanov@itdelight.com.

Lyubarsky Mikhail Mikhailovich, PhD student, Design Automation Department, NURE. Scientific interests: project-bathing and testing digital systems. Scientific interests: design and testing of digital systems. Hobbies: traveling. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14.

Chumachenko Svetlana Victorovna, Dr., Prof., Head of Design Automation Department, NURE. Scientific interests: mathematical modeling of computational processes, theory of series, methods of discrete optimization, educational innovations. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14, phone + 3805770-21-326, e-mail: svetlana.chumachenko@nure.ua

Litvinova Evgenia Ivanovna, Dr., Prof., Design Automation Department, NURE. Scientific interests: design and testing of digital systems. Hobbies: music. Address: Ukraine, 61166, Kharkov, Nauki Ave, 14, e-mail: litvinova_eugenia@icloud.com.

Larchenko Lina, Cand. tech. Sci., Associate Professor of Design Automation Department, NURE. Scientific interests: modeling of specialized architectures. Address: Ukraine, 61166, Kharkov, Nauka Ave., 14, phone +380577021326, e-mail: lina.larchenko@nure.ua