

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ  
УНИВЕРСИТЕТ РАДИОЭЛЕКТРОНИКИ

ISSN 0135-1710

# **АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ И ПРИБОРЫ АВТОМАТИКИ**

**Всеукраинский межведомственный  
научно-технический сборник**

**Основан в 1965 г.**

**Выпуск 163**

Харьков  
2013

В сборнике представлены результаты исследований, касающихся компьютерной инженерии, управления, технической диагностики, автоматизации проектирования, оптимизированного использования компьютерных сетей и создания интеллектуальных экспертных систем. Предложены новые подходы, алгоритмы и их программная реализация в области автоматического управления сложными системами, оригинальные информационные технологии в науке, образовании, медицине.

Для преподавателей университетов, научных работников, специалистов, аспирантов.

У збірнику наведено результати досліджень, що стосуються комп'ютерної інженерії, управління, технічної діагностики, автоматизації проектування, оптимізованого використання комп'ютерних мереж і створення інтелектуальних експертних систем. Запропоновано нові підходи, алгоритми та їх програмна реалізація в області автоматичного управління складними системами, оригінальні інформаційні технології в науці, освіті, медицині.

Для викладачів університетів, науковців, фахівців, аспірантів.

**Редакционная коллегия:**

*В.В. Семенец*, д-р техн. наук, проф. (гл. ред.); *М.Ф. Бондаренко*, д-р техн. наук, проф.; *И.Д. Горбенко*, д-р техн. наук, проф.; *Е.П. Пуятин*, д-р техн. наук, проф.; *В.П. Тарасенко*, д-р техн. наук, проф.; *Г.И. Загарий*, д-р техн. наук, проф.; *Г.Ф. Кривуля*, д-р техн. наук, проф.; *Чумаченко С.В.*, д-р техн. наук, проф.; *В.А. Филатов*, д-р техн. наук, проф.; *Е.В. Бодянский*, д-р техн. наук, проф.; *Э.Г. Петров*, д-р техн. наук, проф.; *В.Ф. Шостак*, д-р техн. наук, проф.; *В.М. Левыкин*, д-р техн. наук, проф.; *Е.И. Литвинова*, д-р техн. наук, проф.; *В.И. Хаханов*, д-р техн. наук, проф. (отв. ред.).

Свидетельство о государственной регистрации  
печатного средства массовой информации

КВ № 12073-944ПР от 07.12.2006 г.

*Адрес редакционной коллегии:* Украина, 61166, Харьков, просп. Ленина, 14, Харьковский национальный университет радиоэлектроники, комн. 321, тел. 70-21-326

© Харківський національний університет  
радіоелектроніки, 2013

## СОДЕРЖАНИЕ

<i>В.И. ХАХАНОВ, BAGHDADI AMMAR AWNI ABBAS, С.В. ЧУМАЧЕНКО, А.С. ШКИЛЬ</i> КВАНТОВЫЕ СТРУКТУРЫ ДЛЯ ТЕСТИРОВАНИЯ ЦИФРОВЫХ УСТРОЙСТВ.....	4
<i>В.В. БАРАННИК, В.Н. КРИВОНОС, А.Э. МОСКВИН, Э.А. ПЛЕШКО</i> МЕТОД ВОССТАНОВЛЕНИЯ КОМПОНЕНТ ТРАНСФОРМАНА В ИНФОРМАЦИОННОЙ ТЕХНОЛОГИИ РЕКОНСТРУКЦИИ ВИДЕОДАНЫХ.....	18
<i>В.И. ХАХАНОВ, С.В. ЧУМАЧЕНКО, Е.И. ЛИТВИНОВА</i> СОСТОЯНИЕ ИТ-РЫНКА УКРАИНЫ (АНАЛИТИЧЕСКИЙ ОБЗОР).....	22
<i>В.В. БАРАННИК, Н.А. ХАРЧЕНКО, В.Н. КРИВОНОС</i> МЕТОД КОМПРЕССИИ ВИДЕОПОТОКА НА ОСНОВЕ ПОЛИАДИЧЕСКОГО КОДИРОВАНИЯ ПРЕДСКАЗЫВАЕМЫХ КАДРОВ.....	47
<i>А.П. ОКСАНИЧ, С.Э. ПРИТЧИН, М.Г. КОГДАСЬ, В.А. ТЕРБАН</i> УСОВЕРШЕНСТВОВАНИЕ МЕТОДА И АППАРАТУРЫ ИЗМЕРЕНИЯ ПЛОТНОСТИ ДИСЛОКАЦИЙ В ПОДЛОЖКАХ КРЕМНИЯ И АРСЕНИДА ГАЛЛИЯ.....	53
<i>Т.А. ЗАЙКО, А.О. ОЛІЙНИК, С.О. СУББОТІН</i> ІНДУКЦІЯ ЧИСЕЛЬНИХ АСОЦІАТИВНИХ ПРАВИЛ З ВРАХУВАННЯМ ІНДИВІДУАЛЬНОЇ ЗНАЧУЩОСТІ ОЗНАК.....	60
<i>И.В. ГУРИНА, А.Я. КУЗЕМИН</i> МИНИМИЗАЦИЯ РИСКОВ ДЛЯ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ РАСПРЕДЕЛЕННОЙ ЭНЕРГЕТИЧЕСКОЙ СЕТИ ПОТРЕБИТЕЛЕЙ.....	68
<i>А.А. ПЛУГИН, Л.Э. ЧАЛАЯ</i> МЕТОД АВТОМАТИЧЕСКОГО ГЕНЕРИРОВАНИЯ БИБЛИОГРАФИЧЕСКИХ СПИСКОВ.....	77
<i>Д.А. БУЛАВИН</i> АНАЛИЗ ОРГАНИЗАЦИИ, ОБУЧЕНИЯ И ПРИМЕНЕНИЯ КВАНТОВЫХ НЕЙРОННЫХ СЕТЕЙ.....	89
<i>К.С. ДЄЄВ</i> ВИВЧЕННЯ ХАРАКТЕРУ ВЗАЄМОДІЇ ТИПУ ТОЧКА-ТОЧКА ДЛЯ КЛАСИФІКАЦІЇ МЕРЕЖЕВОГО ТРАФІКУ.....	94
<b>РЕФЕРАТЫ.....</b>	<b>101</b>
<b>ПРАВИЛА</b> ОФОРМЛЕНИЯ СТАТЕЙ ДЛЯ АВТОРОВ НАУЧНО-ТЕХНИЧЕСКОГО СБОРНИКА.....	<b>105</b>

## **КВАНТОВЫЕ СТРУКТУРЫ ДЛЯ ТЕСТИРОВАНИЯ ЦИФРОВЫХ УСТРОЙСТВ**

---

Предлагаются теория и примеры реализации квантовых моделей, методов и алгоритмов для повышения быстродействия существующих программных и аппаратных средств анализа цифровых вычислительных устройств за счет увеличения размерности структур данных и памяти. Вводятся понятия, термины и определения, необходимые для понимания теории и практики квантовых вычислений. Описываются результаты исследований, касающихся моделей и методов квантового диагностирования цифровых систем, кубитного моделирования исправного поведения, восстановления работоспособности отказавших примитивов.

### **1. Введение**

Эволюция киберпространства планеты условно делится на следующие периоды: 1) 1980-е годы – формирование парка персональных компьютеров (Computation); 2) 1990-е годы – внедрение Интернет-технологий в производственные процессы и быт человека (Internet); 3) 2000-е годы – повышение качества жизни за счет внедрения мобильных устройств и облачных сервисов (Mobility); 4) 2010-е годы – создание цифровой инфраструктуры мониторинга, управления и взаимодействия между собой (Internet of Things) стационарных и движущихся объектов (воздушный, морской, наземный транспорт и роботы); 5) 2015-е годы – создание глобальной цифровой инфраструктуры киберпространства, где все процессы, явления идентифицируются во времени и в трехмерном пространстве и становятся интеллектуальными (умными) Smart Everything. В связи с необходимостью развития зеленых технологий квантовые вычисления в последние годы становятся все более значимыми для анализа эволюционирующего кибернетического пространства, создания облачных Интернет сервисов благодаря их позитивной альтернативности существующим затратным по материалам, времени и энергии классическим моделям вычислительных процессов [1].

Рыночная привлекательность квантовых методов и кубитных моделей основывается на высоком параллелизме решения практически всех задач дискретной оптимизации, факторизации, минимизации булевых функций, эффективного сжатия, компактного представления и телепортации данных, отказоустойчивого проектирования за счет существенного повышения аппаратных затрат. Однако такая плата в настоящее время вполне допустима, поскольку рынок nano-электронных технологий предоставляет разработчикам цифровых систем до 1 миллиарда вентилях на кристалле размерностью 2x2 см при толщине пластины в 5 микрон. При этом современные технологии допускают создание пакета или «сэндвича», содержащего до 7 кристаллов, что соизмеримо с объемом нейронов головного мозга человека. Практически «беспроводное» соединение таких пластин основывается на технологической возможности сверления порядка 10 тысяч сквозных отверстий (vias) на 1 квадратном сантиметре.

Кроме того, появление трехмерных FinFETs транзисторов и основанных на них 3D-технологий реализации объемных цифровых систем предоставляет практически неограниченные аппаратные возможности для исследователей в части создания новых параллельных вычислительных устройств [2-7]. Поэтому можно и нужно использовать «жадные» к аппаратуре модели и методы для создания быстродействующих средств параллельного решения практических задач. Имея в виду дискретность и многозначность алфавитов описания информационных процессов, свойство параллелизма (одновременности процессов), заложенное в квантовых вычислениях, является особенно востребованным при создании эффективных и интеллектуальных «движков» для киберпространства, облачных структур и сервисов Интернета; средств синтеза отказоустойчивых цифровых

устройств; тестирования и моделирования цифровых систем на кристаллах; технологий защиты информации и компьютерных изделий. Здесь не рассматриваются физические основы квантовой механики, касающиеся недетерминированного взаимодействия атомных частиц [1], но при этом используется понятие информационного кванта как двоичного или многозначного вектора для совместного задания булеана состояний дискретной области киберпространства, определяющего высокий уровень параллелизма предлагаемых квантовых моделей и методов.

Квантовые вычислители достаточно эффективно применяются для отказоустойчивого проектирования и решения оптимизационных задач, связанных с полным перебором вариантов решений на основе использования теории множеств [8]. Особенность в том, что множество элементов в компьютере всегда является упорядоченным, поскольку каждый бит, байт или другой компонент имеет свой адрес. Поэтому все теоретико-множественные операции, так или иначе, сводятся к полному перебору адресов примитивных элементов. Адресный порядок структур данных хорош для задач, где компоненты моделей можно строго ранжировать, что дает возможность выполнять их анализ за один проход или одну итерацию. Там, где нет порядка в структуре, например, множество всех подмножеств, классическая модель памяти и вычислительных процессов наносит вред времени анализа ассоциации равных по рангу примитивов, или, в лучшем случае, обработка ассоциативных групп является неэффективной. Что можно предложить для неупорядоченных данных вместо строгого порядка? Процессор, где элементарной ячейкой служит образ или шаблон универсума из  $n$  примитивов, который генерирует  $Q = 2^n$  всех возможных состояний такой ячейки в виде булеана или множества всех подмножеств. Прямое решение, ориентированное на создание такой ячейки, использует унитарное позиционное кодирование состояний примитивов, которое с помощью суперпозиции последних образует универсум примитивов, формирующее в пределе булеан или множество всех подмножеств [8,9].

## **2. Квантовое диагностирование цифровых систем**

Предлагается матричный метод квантового диагностирования функциональных нарушений и константных неисправностей в программных или аппаратных блоках, которые используют кубитные структуры данных, составляющих диагностическую информацию. Это дает возможность существенно уменьшить вычислительную сложность процессов моделирования и диагностирования за счет введения параллельных логических операций над матричными данными. Представлен квантовый метод исправного моделирования цифровых устройств с восстановлением работоспособности компонентов цифровой системы в режиме online, который имеет существенно более высокое быстродействие благодаря адресной реализации процедуры обработки функциональных примитивов, заданных  $Q$ -покрытиями.

Модель объекта диагностирования представлена в форме графа цифровой системы, которая имеет функциональные элементы, соединенные линиями связей. Среди них имеются ассерционные точки наблюдения, необходимые для верификации, тестирования и диагностирования неисправностей [2]. Диагностическая информация представлена компонентами:

1) Тест проверки или диагностирования неисправностей заданного класса, в данном случае рассматриваются одиночные константные дефекты  $\{0, 1\}$  линий схемы.

2) Таблица неисправностей [6], строки которой задают векторы проверяемых на каждом тестовом наборе дефектов, привязанных к линиям схемы.

3) Матрица достижимостей, которая определяет достижимость каждой ассерционной точки со стороны множества предшествующих линий [8].

4) Матрица состояния ассерционного механизма или матрица экспериментальной проверки, которая определяет состояние каждой ассерции на тестовых наборах путем сравнения эталонной реакции в данной точке с реальным сигналом в процессе выполнения диагностического эксперимента [2,7].

Базовая модель диагностирования цифрового изделия, дискретного процесса или явления представлена компонентами, которые создают 4 измерения в пространстве признаков:

$$\begin{cases} D_b = \langle S, A, F, T \rangle \\ D = \{ \langle S, A \rangle, \langle F, T \rangle \}; \\ V_b = (|S| \times |A| \times |F| \times |T|); \\ V = (|S| \times |A|) + (|F| \times |T|); \\ V_b \gg V; \\ S^* = f(S, A, T); \\ A^* = g(T, A); \\ F^* = h(S, A, F, T); \end{cases}$$

При этом в модели объем диагностической информации  $V$  формируется декартовым произведением (мощностей) четырех компонентов в порядке следования, указанном выше: 1) структура объекта; 2) механизм ассерций или мониторинга; 3) совокупность неисправностей или модулей, подверженных функциональным нарушениям; 4) тестовые наборы или сегменты для диагностирования неисправностей или совокупности упомянутых модулей. Актуально и существенно уменьшить объем диагностической информации можно путем понижения размерности пространства признаков за счет разделения базовой модели на два непересекающихся подмножества  $\langle S, A \rangle, \langle F, T \rangle$ . В этом случае оценка объема диагностической информации становится не мультипликативной, а аддитивной по отношению к мощности полученных в результате разбиения подмножеств без какого-либо уменьшения глубины диагностирования. Здесь первый компонент модели диагностирования представлен матрицей достижимостей, которая позволяет минимизировать маску возможных дефектов на основе анализа структуры схемы путем сравнения истинных и реальных результатов моделирования выходных сигналов на каждом тестовом наборе или сегменте. Число строк такой матрицы равно количеству наблюдаемых выходов или ассерций.

В процессе выполнения метода диагностирования создается двоичная матрица структурной активизации неисправностей, которая служит маской для существенного уменьшения множества подозреваемых дефектов при совместном анализе таблицы неисправностей. При этом символы одиночных константных дефектов  $\{0, 1, X, \emptyset\}$ ,  $X = \{0, 1\}$  в ячейках таблицы неисправностей [6] кодируются соответствующими состояниями кубита  $(10, 01, 11, 00)$  многозначного алфавита Кантора  $A^k = \{0, 1, X, \emptyset\}$ , что дает возможность исключить из вычислительных процессов теоретико-множественные процедуры, заменив их на векторные логические операции.

Для рассмотрения сущности предлагаемого метода используется фрагмент цифровой схемы, представленной на рис. 1. Здесь имеются три ассерционных точки A, B, C для наблюдения за состоянием всех линий схемы в процессе тестирования (выполнения диагностического эксперимента) путем подачи пяти тестовых воздействий, заданных в таблице неисправностей  $F(T)$ . Координаты данной таблицы задают проверяемые на тест-векторах неисправности 0 и 1, а также имеются состояния координат:  $\emptyset$  (.) – отсутствие проверяемых дефектов и  $X$  – проверка на линии константы 0 и 1 одновременно. Правая часть таблицы есть матрица состояний ассерционного механизма в виде результатов сравнения эталонной и реальной реакций цифрового устройства на тестовые наборы. 1 означает несравнение, 0 – совпадение упомянутых реакций.

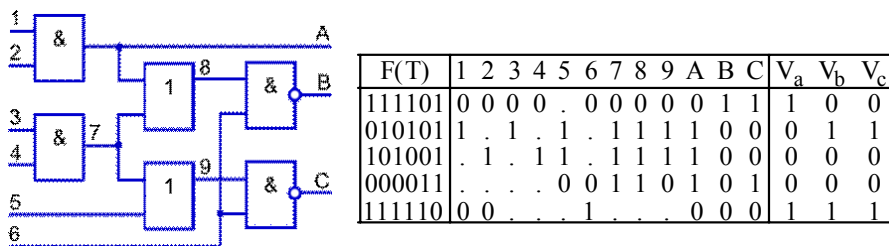


Рис. 1. Фрагмент цифровой схемы и таблица неисправностей

В таблице неисправностей не учитывается структура схемы для повышения глубины диагностирования на основе вычисления реальной матрицы состояний ассерционного механизма, которая совместно с матрицей достижимостей создает структурную маску, минимизирующую множество подозреваемых дефектов.

Для фрагмента цифровой схемы, представленной на рис. 1, матрица достижимостей имеет следующий вид:

S =   S <sub>ij</sub>	1	2	3	4	5	6	7	8	9	A	B	C
1	1	1	.	.	.	.	.	.	.	1	.	.
2	1	1	1	1	.	1	1	1	.	.	1	.
3	.	.	1	1	1	1	1	.	1	.	.	1

Здесь выходы-ассерции A,B,C являются мониторами технического состояния объекта диагностирования. Каждый из них может иметь два состояния:

$A_{ij} = \{0,1\}, A_{ij} = T_i \oplus U_j$ , которые формируют маску возможных дефектов с помощью следующего выражения:  $S(T_i) = \triangleleft = (\bigvee_{A_j=1} S_j) \wedge (\overline{\bigvee_{A_j=0} S_j})$ . Каждый тест-вектор (-сегмент)

активизирует собственную структуру, поэтому маска возможных дефектов функционально зависит от структуры, ассерций (состояния наблюдаемых выходов) и тестовых наборов:

$S = f(S, A, T_i)$ . Если предположить, что в матрице  $S = |S_{ij}|$  состояния выходов равны (A,B,C) = (011), где значение 1 идентифицирует проявление дефекта в устройстве, то маска возможных дефектов будет иметь следующий вид:

$$S = (S_2 \vee S_3) \wedge (\overline{S_1}) = (111101110010 \vee 0011111101001) \wedge (\overline{110000000100}) = \\ = (11111111011) \wedge (00111111011) = (00111111011).$$

Полученная маска накладывается на первую строку таблицы неисправностей, что определяет множество подозреваемых дефектов, формирующих ассерционную выходную реакцию (A,B,C) = (011) устройства на первый тест-вектор:

Test \ Faults	1	2	3	4	5	6	7	8	9	A	B	C
T1	0	0	0	0	.	0	0	0	0	0	1	1
S1	0	0	1	1	1	1	1	1	1	0	1	1
T1(S1)	.	.	0	0	.	.	0	0	0	.	1	1

В соответствии с предложенной процедурой получения маски одной строки выполняет построение матрицы структурной активизации неисправностей S(T) на основе использования таблицы экспериментальной проверки, задающей состояния ассерционного механизма в процессе выполнения тестирования:

S =   S <sub>ij</sub>	1	2	3	4	5	6	7	8	9	A	B	C	$\triangleleft$	T \ A	A	B	C	A(T)	=	S(T)	1	2	3	4	5	6	7	8	9	A	B	C
														T1	1	0	0	1		T1	1	1	0	0	0	0	0	0	0	0	0	1
1	1	1	.	.	.	.	.	.	.	1	.	.		T2	0	1	1	1	1	T2	0	0	1	1	1	1	1	1	0	1	1	
2	1	1	1	1	.	1	1	1	.	.	1	.		T3	0	0	0	0	0	T3	0	0	0	0	0	0	0	0	0	0	0	0
3	.	.	1	1	1	1	1	.	1	.	.	.	1	T4	0	0	0	0	0	T4	0	0	0	0	0	0	0	0	0	0	0	0
														T5	1	1	1	1	1	T5	1	1	1	1	1	1	1	1	1	1	1	1

Для формирования структур данных, удобных для компьютерной обработки, необходимо перевести символы таблицы неисправностей в двухразрядные коды в соответствии с правилами  $\triangleright$ -кодирования:  $\triangleright = \{0 = 10, 1 = 01, X = 11, \emptyset = 00\}$ , применение которых к таблице неисправностей F(T) дает следующий результат:

F(T)	1	2	3	4	5	6	7	8	9	A	B	C	$\triangleright$	F(T)	1	2	3	4	5	6	7	8	9	A	B	C
T1	0	0	0	0	.	0	0	0	0	0	1	1		T1	10	10	10	10	00	10	10	10	10	10	01	01
T2	1	.	1	.	1	.	1	1	1	1	0	0	T2	01	00	01	00	01	00	01	01	01	01	10	10	
T3	.	1	.	1	1	.	1	1	1	1	0	0	T3	00	01	00	01	01	00	01	01	01	01	10	10	
T4	.	.	.	.	0	0	1	1	0	1	0	1	T4	00	00	00	00	10	10	01	01	10	01	10	01	
T5	0	0	.	.	.	1	.	.	.	0	0	0	T5	10	10	00	00	00	01	00	00	00	10	10	10	

После получения структурной матрицы, предназначенной маскировать реальные дефекты в таблице неисправностей и ее кодированной формы, необходимо выполнить #-суперпозицию двух матриц:  $F(T) = S(T) \# F(T)$ , которая сводится к #-операции над одноименными координатами  $F_{ij} = \bar{F}_j \leftarrow (F_j = 00) \vee (S_{ij} = 0)$ , что означает модификацию кодов координат таблицы  $F(T)$  при выполнении заданных условий. Иначе, данная операция сводится к инверсии ячеек матрицы кодов неисправностей, маскируемых нулевыми сигналами структурной матрицы активизации, а также всех нулевых кодов таблицы неисправностей. Таблица истинности данной #-операции в символьном и кодированном виде представлена ниже:

$\# = S_{ij} \setminus F_{ij}$	$\emptyset$ 1 0 X	$\# = S_{ij} \setminus F_{ij}$	00 01 10 11
0	X 0 1 $\emptyset$	0	11 10 01 00
1	X 1 0 X	1	11 01 10 11

Она скорректирована относительно инверсии состояния 00 в 11 при единичном значении сигнала активизации неисправности, потому что такой код (00) означает присутствие в схеме на линии пустого множества проверяемых дефектов, что невозможно. Но код 00 еще блокирует все вычисления конъюнкции по столбцу, превращая результат в 00. Инверсия кода дает возможность не маскировать при логическом умножении действительно присутствующих дефектов, любых знаков. При этом предполагается, что невозможно тест-вектором проверить на одной линии схемы дефекты разных знаков.

Выполнение процедуры суперпозиции структурной матрицы с кодированной таблицей неисправностей в соответствии с #-операцией дает следующий результат:

S(T)	1 2 3 4 5 6 7 8 9 A B C	#	F(T)	1 2 3 4 5 6 7 8 9 A B C	=
T1	1 1 0 0 0 0 0 0 0 1 0 0		T1	10 10 10 10 00 10 10 10 10 10 01 01	
T2	0 0 1 1 1 1 1 1 1 0 1 1		T2	01 00 01 00 01 00 01 01 01 01 01 10 10	
T3	0 0 0 0 0 0 0 0 0 0 0 0		T3	00 01 00 01 01 00 01 01 01 01 01 10 10	
T4	0 0 0 0 0 0 0 0 0 0 0 0		T4	00 00 00 00 10 10 01 01 10 01 10 01	
T5	1 1 1 1 1 1 1 1 1 1 1 1		T5	10 10 00 00 00 01 00 00 00 10 10 10	

=	^	F(T)	1 2 3 4 5 6 7 8 9 A B C
			10 10 01 01 11 01 01 01 01 10 10 10
			10 11 01 11 01 11 01 01 01 10 10 10
			11 10 11 10 10 11 10 10 10 10 01 01
			11 11 11 11 01 01 10 10 01 10 01 10
			10 10 11 11 11 01 11 11 11 10 10 10
	F(C) =	10 10 01 00 00 01 00 00 00 10 00 00	
	F =	0 0 1 . . 1 . . . 0 . .	

На заключительной стадии диагностирования выполняется единственная и векторная операция логического умножения всех строк кодированной модифицированной таблицы истинности  $F(T)$ :

$$F(T_i) = \left( \bigvee_{A_j=1} F_j \right) \wedge \left( \overline{\bigvee_{A_j=0} F_j} \right) = \left( \bigwedge_{A_j=1} F_j \right) \wedge \left( \overline{\bigvee_{A_j=0} F_j} \right) = \left( \bigwedge_{A_j=1} F_j \right) \wedge \left( \bigwedge_{A_j=0} \bar{F}_j \right) = \left( \bigwedge_{j=1}^n F_j \right).$$

Это дает возможность точно определить все дефекты, присутствующие в объекте диагностирования, которые представлены в двух нижних строках приведенной выше кодированной таблицы неисправностей  $F(T)$ :  $F = \{1^0, 2^0, 3^1, 6^1, A^0\}$ .

Теоретическое доказательство матричного диагностирования одиночных и кратных дефектов представлено ниже в виде двух теорем.

**Теорема 1.** *Одиночные константные дефекты цифровой схемы, заданные кубитами (двухразрядными векторами) на тестовых наборах многозначной таблицы неисправностей, определяются с помощью векторной and-операции, маскируемой по строкам вектором экспериментальной проверки  $A(T)$  всех ассерционных точек:*



$$F(T_i) = \left( \bigvee_{A_j=1} F_j \right) \wedge \left( \overline{\bigvee_{A_j=0} F_j} \right) = \left( \bigwedge_{A_j=1} F_j \right) \wedge \left( \bigwedge_{A_j=0} \bar{F}_j \right) = \left( \bigwedge_{j=1}^n F_j \right).$$

Выражение является верным, потому что: 1) второй сомножитель – чистая математика – отрицание дизъюнкции есть конъюнкция отрицаний, что означает умножение кодов таблицы с их предварительным отрицанием; 2) первый сомножитель ориентирован на поиск непротиворечивых дефектов, поэтому он заменяется на  $\bigwedge_{A_j=1} F_j$ . Действи-

тельно, на одной линии или переменной не могут присутствовать одновременно две противоположных по знаку проверяемых неисправности. Поэтому в базовой формуле дизъюнкция дефектов  $\bigvee_{A_j=1} F_j$  в большей степени ориентирована на поиск кратных

неисправностей, но не связанных с одной линией. Кратность противоречивых дефектов на одной линии, равно как и инверсия пустого множества неисправностей, теоретически создает условия беспрепятственного умножения других ячеек столбца в целях формирования на каждой линии результата в виде дефекта одного знака или пустого множества неисправностей.

**Теорема 2.** *Кратные константные дефекты цифровой схемы, заданные кубитами на тестовых наборах многозначной таблицы неисправностей, определяются с помощью векторных *or*- и *and*- операций, маскируемых по строкам вектором экспериментальной проверки  $A(T)$  всех ассерционных точек:*

$$F(T_i) = \left( \bigvee_{A_j=1} F_j \right) \wedge \left( \overline{\bigvee_{A_j=0} F_j} \right) = \left( \bigvee_{A_j=1} F_j \right) \wedge \left( \bigwedge_{A_j=0} \bar{F}_j \right).$$

Выражение является верным, потому что: 1) второй сомножитель есть отрицание дизъюнкции или конъюнкция отрицаний, что означает умножение кодов таблицы с их предварительным отрицанием; 2) первый сомножитель ориентирован на поиск кратных дефектов в предположении, что на одной линии или переменной могут присутствовать одновременно две противоположные по знаку проверяемые неисправности. Данная формула в большей степени ориентирована на поиск кратных дефектов в блоках цифровых систем, не связанных с одной линией. Кратность неисправностей в цифровой системе теоретически создает условия для логического сложения других ячеек столбца в целях формирования результата в виде множества дефектов, формирующих заданный вектор экспериментальной проверки, из которых необходимо вычесть проверяемые на тесте неисправности, которые не влияют на формирование некорректных реакций по выходам.

Интерес представляет поиск кратных дефектов на основе мультипроцессора Хассе [4,5], который ориентирован на решение задачи покрытия путем полного перебора событий, обеспечивающих точное покрытие вектора экспериментальной проверки столбцами таблицы неисправностей:

$$F(T_i) = \left( \bigvee_j F_j \right) \oplus A = 0.$$

Здесь решением является такое сочетание столбцов, участвующих в векторной операции логического сложения, которое в совокупности дает результат, равный вектору экспериментальной проверки. Поскольку операция времязатратная, то для нее следует использовать мультипроцессор Хассе, ориентированный на взятие булеана в почти параллельном режиме.

Подводя итог, следует отметить, что модель процесса диагностирования цифровых устройств содержит функциональные преобразователи, связанные с выполнением следующих шагов (рис. 2):

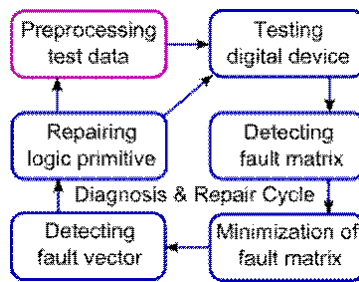


Рис. 2. Цикл диагностирования и ремонта логических блоков

1. Препроцессирование. Генерирование исходной диагностической информации в виде теста диагностирования, таблицы неисправностей и матрицы достижимостей цифровой системы.

2. Тестирование реального устройства на основе использования промышленного симулятора в целях сравнения фактических реакций устройства с эталонными значениями по наблюдаемым линиям-ассерциям, что дает возможность сформировать матрицу выходных реакций или вектор экспериментальной проверки в двоичном алфавите.

3. Вычисление матрицы активности графовой структуры на каждом входном тестовом наборе, равной по размерности таблице неисправностей, с помощью матрицы экспериментальной проверки и матрицы достижимостей, что дает возможность существенно сократить область подозреваемых дефектов.

4. Модификация содержимого таблицы неисправностей путем ее маскирования матрицей активности графовой структуры, в целях определения только тех неисправностей, которые действительно формируют матрицу экспериментальной проверки в процессе диагностирования.

5. Выполнение процедуры логического умножения над строками таблицы неисправностей для получения вектора подозреваемых дефектов.

6. Восстановление работоспособности цифрового устройства путем переадресации неисправных логических компонентов на их аналоги из ремонтного запаса и повторение процесса тестового диагностирования.

### 3. Квантовое моделирование цифровых систем

Рассматриваются структуры данных, эффективные с точки зрения программной или аппаратной реализации исправного моделирования дискретных систем, описанных в форме кубитных векторов. Для описания цифровой схемы, представленной на рис. 3, необходимо иметь структуру взаимосвязанных элементов и кубическое покрытие логического элемента И-НЕ.

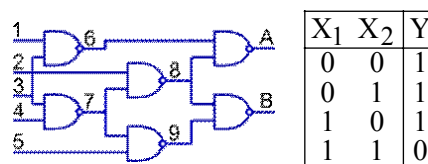


Рис. 3. Фрагмент цифровой схемы

Данное покрытие можно трансформировать с учетом унитарного кодирования входных векторов. Для этого необходимо использовать двухтактный алфавит [4,5,6,7] описания автоматных переменных, представляющий собой булеан на универсуме из четырех примитивов:

$V^*(Y) = \{Q=(1000), E=(0100), H=(0010), J=(0001), O=\{Q,H\}=(1010), I=\{E,J\}=(0101), A=\{Q,E\}=(1100), B=\{H,J\}=(0011), S=\{Q,J\}=(1001), P=\{E,H\}=(0110), C=\{E,H,J\}=(1110), F=\{Q,H,J\}=(1011), L=\{Q,E,J\}=(1101), V=\{Q,E,H\}=(1110), Y=\{Q,E,H,J\}=(1111), U=(0000)\}$ .

Любое покрытие функционального одновыходового примитива всегда имеет лишь два куба. Они показывают не только все решения, но и инверсию сигналов на выходе, что интересно с позиции активизации всех логических путей в схемной структуре. Например, в

следующем кубитном покрытии вентиля для изменения состояния выхода необходимо создать на входах пару следующих друг за другом условий, где в первом такте должны быть первые три вектора (адреса), а во втором – четвертый вектор, формируемый двумя входными переменными:

$$\begin{array}{|c|c|c|} \hline 00 & 1 & \\ \hline 01 & 1 & \\ \hline 10 & 1 & \\ \hline 11 & 0 & \\ \hline \end{array} \cdot \begin{array}{|c|c|} \hline Q & 1 \\ \hline E & 1 \\ \hline H & 1 \\ \hline J & 0 \\ \hline \end{array} \cdot \begin{array}{|c|} \hline V & 1 \\ \hline J & 0 \\ \hline \end{array} \cdot \begin{array}{|c|c|c|} \hline 1110 & 1 & \\ \hline 0001 & 0 & \\ \hline \end{array} \cdot \begin{array}{|c|c|c|} \hline 1 & 1 & 1 & 0 \\ \hline \end{array}$$

Для моделирования исправного поведения достаточно иметь только один куб (нулевой или единичный), поскольку второй всегда является дополнением к первому. Следовательно, ориентируясь, например, на единичный куб, формирующий на выходе 1, можно убрать бит состояния выхода примитива, что уменьшит размерность куба или модели примитива до количества адресов элемента, где бит с адресным номером  $i$  идентифицирует способность адреса, составленного из значений входных переменных, сформировать единичное состояние выхода примитива.

Кубитное или квантовое покрытие одновыходового примитива всегда представлено двумя кубами, размерность которых равна степени двойки от числа входных переменных, где единичное значение координаты определяет участие адреса рассматриваемого бита в формировании соответствующего (0,1) состояния выхода примитива. Квантовые модели примитивов требуют создания новой теории моделирования, прямой и обратной импликации, синтеза тестов, моделирования неисправностей, поиска дефектов. Здесь основные процедуры связаны с манипулированием адресами, неявно представленными в координатах двух кубов квантового (кубитного) покрытия, далее именуемого Q-покрытием (Q-coverage):

$$\begin{array}{|c|c|} \hline 00 & 0 \\ \hline 01 & 0 \\ \hline 10 & 1 \\ \hline 11 & 0 \\ \hline \end{array} = \begin{array}{|c|c|} \hline Q & 0 \\ \hline E & 0 \\ \hline H & 1 \\ \hline J & 0 \\ \hline \end{array} = \begin{array}{|c|c|} \hline F & 0 \\ \hline H & 1 \\ \hline \end{array} = \begin{array}{|c|c|c|} \hline 1101 & 0 \\ \hline 0010 & 1 \\ \hline \end{array}$$

Модель для анализа цифровой системы на основе использования кубитных структур данных и информационных кубов может быть описана четырьмя компонентами:

$$\begin{aligned} &F \ 00 \ L, M, X, Q \ 0, \\ &L \ 0 \ (L_1, L_2, \dots, L_j, \dots, L_n); \\ &M \ 0 \ (M_1, M_2, \dots, M_j, \dots, M_n); \\ &X \ 0 \ (X_{n_x 01}, X_{n_x 02}, \dots, X_{n_x 0i}, \dots, X_n); \\ &Q \ 0 \ (Q_{n_x 01}, Q_{n_x 02}, \dots, Q_{n_x 0i}, \dots, Q_n). \end{aligned}$$

Здесь представлены: вектор идентификаторов эквипотенциальных линий схемы цифровой системы, который ввиду своей тривиальности может быть исключен из модели, но при этом необходимо иметь число входных переменных устройства и общее количество линий:  $n_x, n$ ; вектор моделирования состояний всех линий схемы, упорядоченная совокупность векторов входных переменных каждого примитива схемы, привязанных к номерам выходов, совокупность векторов – квантовых покрытий примитивов, строго привязанных к номерам выходов и входным переменным примитивов;  $n$  – число линий в схеме,  $n_x$  – количество входных переменных.

В качестве примера кубитного задания модели цифрового устройства  $F = \langle L, M, X, Q \rangle$ , представленного на рис. 6, ниже приведены два варианта таблиц описания схемы для анализа исправного поведения (fault free simulation):

L	1	2	3	4	5	6	7	8	9	A	B
Q	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	0	0	0	0	0	0
X	13 34 27 75 68 89										
M	1	1	1	1	1	0	0	1	1	1	0

L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
Q	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	1	1	1	1	1	1
	.	.	.	.	.	0	0	0	0	0	0

Метод квантового моделирования исправного поведения сводится к определению значения выхода элемента по адресу, формируемому двоичными состояниями входных переменных каждого примитива цифровой системы  $M(Y_i) \rightarrow Q_i[M(X_{i1} * X_{i2} \dots * X_{ij} \dots * X_{ik_i})]$ . Здесь  $k_i$  – число входных линий в квантовом примитиве с номером  $i$ . Поскольку номера невходных линий вектора  $L$  однозначно идентифицируют по выходам обрабатываемые примитивы, то формула моделирования может быть приведена к циклу определения состояний всех невходных переменных:

$$M_i \cdot Q_i[M(X_{i1} * X_{i2} \dots * X_{ij} \dots * X_{ik_i})] \cdot Q_i[M(A_i)], \quad i = \overline{n_x + 1, n}.$$

Здесь процесс моделирования связан с конкатенированным формированием адреса бита в кванте функциональности для вычисления по нему состояния невходной линии цифровой структуры, начиная с номера  $i \in \overline{n_x + 1, n}$ . Если переменные создают не двоичный адрес, то в данном случае существует возможность формирования не двоичного состояния выхода примитива, которое определено в троичном алфавите символом  $X$ . Состояния выходов формируются последовательной процедурой моделирования на основе простых итераций или итераций Зейделя [6,8]. Во втором случае необходима препроцессорная процедура ранжирования линий и примитивов схемы, которая позволяет существенно уменьшить количество проходов по примитивам схемы для достижения сходимости, когда фиксируется равенство состояний всех линий схемы в двух соседних итерациях. Кроме того, ранжирование примитивов по уровням формирования выходов дает возможность существенно повысить быстродействие моделирования за счет параллельной обработки функциональных элементов одного уровня. Например, для схемы, представленной на рис. 6, одновременно можно обрабатывать элементы с номерами 6,7, затем – 8,9 и далее – А,В. В первом случае, когда используются простые итерации, ранжирования не требуется, но платой за простоту алгоритма моделирования является существенно большее число итеративных проходов по примитивам схемы для достижения упомянутого критерия сходимости. Вычислительная сложность предложенного метода моделирования на основе квантовых функциональностей определяется процедурами формирования адреса – входного вектора, содержащего  $k_i$  переменных, для каждого  $i$ -го примитива  $[(r + w) \times k_i]$ , считыванием бита из квант-вектора по конкатенированному адресу и записью  $(r \rightarrow w)$  данного бита в вектор моделирования:

$$\eta = \sum_{i=n_x+1}^n \{[(r + w) \times k_i] + (r + w)\} = \sum_{i=n_x+1}^n [(r + w) \times (k_i + 1)] = (r + w) \times \sum_{i=n_x+1}^n (k_i + 1).$$

Время моделирования одного тест-вектора квантовым методом при условии, что цифровая схема, составленная из 900 четырехвходовых примитивов, имеет параметры:  $r = w = 5ns$ ,  $k_i = 4$ ,  $n_x = 100$ ,  $n = 1000$ , равно 45 микросекунд:

$$\eta = (r + w) \times \sum_{i=n_x+1}^n (k_i + 1) = (5 + 5) \times 900 \times (4 + 1) = 10 \times 900 \times 5 = 45000ns = 45\mu s.$$

Это означает, что быстродействие метода моделирования дает возможность для данной схемы обработать за одну секунду 22 222 входных наборов. При этом цифровое устройство имеет существенное преимущество – сервисную функцию online восстановления работоспособности в случае отказа примитива путем его переадресации на запасной элемент.

Для синтеза квазиоптимальных структур данных комбинационного устройства необходимо использовать следующие правила:

1) Ранжированная схема цифрового устройства по структурной глубине элемента для моделирования по способу Зейделя должна иметь по возможности однотипные примитивы в каждом уровне (слое) срабатывания.

2) В каждом уровне желательнее иметь одинаковое число примитивов. Поэтому синтез цифрового устройства следует ориентировать на создание прямоугольной (матричной) структуры однотипных логических элементов.

3) Реализация комбинационных примитивов предполагает применение адресуемых элементов памяти, имеющих место в программируемых логических устройствах (FPGA, CPLD), широко используемых для создания прототипов.

4) Формирование для каждого уровня комбинационного устройства ремонтных примитивов для восстановления работоспособности в режиме online из расчета – один запасной элемент на каждый тип компонента, используемый в уровне.

5) Стоимость аппаратных затрат для реализации комбинационного устройства, ориентированного на высокое быстродействие, должна определяться суммой всех примитивов, привязанных к уровням комбинационного устройства, дополненной линейкой запасных элементов по одному для каждого слоя (при условии существования в каждом слое одинаковых примитивов):

$$Q = \sum_{i=1, n}^{\overline{j=1, m}} P_{ij} + n.$$

6) Реализация комбинационного устройства, ориентированного на минимизацию аппаратных затрат, определяется суммой всех типов примитивов, инвариантных к уровням комбинационного устройства, дополненной линейкой запасных элементов по одному для каждого типа:

$$Q = \sum_{i=1}^m P_i + m.$$

7) Обработка матрицы комбинационных элементов с помощью процессорной линейки примитивов, число которых равно мощности максимального уровня или слоя в прямоугольной структуре, что обеспечивает условия для параллельной обработки всех примитивов в каждом уровне элементов в целях повышения быстродействия комбинационного прототипа, реализуемого в PLD.

#### 4. Восстановление работоспособности комбинационных устройств

Немногочисленные работы, посвященные восстановлению работоспособности логических схем [9-11], описывают две идеи. Первая заключается в реконфигурации структуры логических элементов в режиме offline, которая обеспечивает возможность замены каждого из неисправных примитивов. Вторая создает условия для замены неисправных элементов путем использования запасных логических компонентов и расширения мультиплексов для переадресации отказавших примитивов.

Структуры кубитных данных модифицируются в сторону дополнения строкой типов примитивов  $F = \langle L, M, X, P, Q \rangle$ ,  $P = (P_1, P_2, \dots, P_1, \dots, P_m)$ , задействованных при синтезе цифровой системы, если необходимо в процессе функционирования выполнять ремонт или восстановление работоспособности путем введения запасных примитивов, которые, также как и основные, реализуются на основе элементов памяти. На рис. 4 показан пример схемной структуры из адресуемых и трех запасных элементов. Структуры данных, соответствующие этой схеме с тремя дополнительными элементами, представлены здесь же.

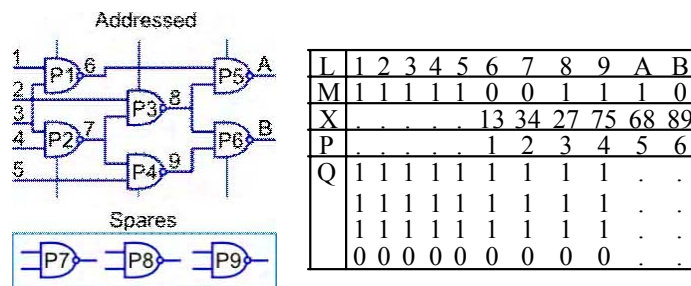


Рис. 4. Пример схемной структуры из адресуемых и запасных элементов

Таблица оперирует номерами структурных примитивов, что дает возможность заменить любой отказавший элемент исправным из ремонтного запаса путем изменения адресного номера в строке примитивов P. Ремонтные элементы в данной таблице начинаются с номера 7.

В следующей таблице представлены строка типов логических элементов, а также адреса типов этих примитивов, отмеченные номерами:

L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
P	.	.	.	.	1	1	1	1	1	1	1
Q	1	1	1	1	.	.	.	.	.	.	.
	1	1	1	1	.	.	.	.	.	.	.
	1	1	1	1	.	.	.	.	.	.	.
	0	0	0	0	.	.	.	.	.	.	.

Эта структура данных ориентирована на программную реализацию моделирования, а ремонтные примитивы начинаются с номера 2. Если существует возможность перепрограммирования логики в элементе памяти с одинаковым числом входных переменных, то данную процедуру следует выполнять после фиксации неисправного элемента, когда становится известно – какой элемент в структуре и какой тип примитива отказал. Такая процедура восстановления работоспособности ориентирована на PLD-реализацию цифровых систем. Если кубитные модели схем не имеют запасных примитивов, то соответствующие им таблицы будут иметь следующий вид:

L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
P	.	.	.	.	1	1	1	1	1	1	1
Q	1	.	.	.	.	.	.	.	.	.	.
	1	.	.	.	.	.	.	.	.	.	.
	1	.	.	.	.	.	.	.	.	.	.
	0	.	.	.	.	.	.	.	.	.	.

L	1	2	3	4	5	6	7	8	9	A	B
M	1	1	1	1	1	0	0	1	1	1	0
X	.	.	.	.	.	13	34	27	75	68	89
P	.	.	.	.	1	2	3	4	5	6	.
Q	1	1	1	1	1	1	.	.	.	.	.
	1	1	1	1	1	1	.	.	.	.	.
	1	1	1	1	1	1	.	.	.	.	.
	0	0	0	0	0	0	.	.	.	.	.

Таким образом, квантовые структуры данных ориентированы на компактность описания функционалов цифрового изделия кубитными векторами, повышение быстродействия процедур моделирования за счет адресации состояний выходов примитивов, а также на восстановление работоспособности отдельных логических элементов, благодаря их реализации в элементах памяти PLD.

Обработка схемы (processing) схемы в кристалле сводится к определению адреса, составленного двоичными битами вектора моделирования, по которому находится логическая функция. Каждый примитив имеет цикл обработки, содержащий три процедуры:

1) Адресное считывание номеров входных переменных из соответствующего столбца матрицы X для формирования адреса состояния входной переменной вектора моделирования:  $A = 2 X_{ij}, i \in \overline{1, n}; j \in \overline{1, s_p} \cup \overline{21}$ .

2) Формирование адреса (двоичного кода) для вычисления логической функции путем конкатенации соответствующих состояний входных переменных в векторе моделирования  $A = 4 M(X_{ij}) * M(X_{ir})$ .

3) Запись результата выполнения логической функции как состояния выхода в соответствующий разряд вектора моделирования  $M(X_{is_p}) = 4 P[M(X_{ij}) * M(X_{ir})]$ .

Процесс обработки всех примитивов схемы в данном случае является строго последовательным, что представляет собой существенное замедление процедуры формирования состояний выходных переменных. Однако уменьшение быстродействия можно считать платой за сервис встроенного и автономного восстановления работоспособности цифровой структуры, который является одним из этапов функционирования инфраструктуры обслуживания SoC, представленной на рис. 5.



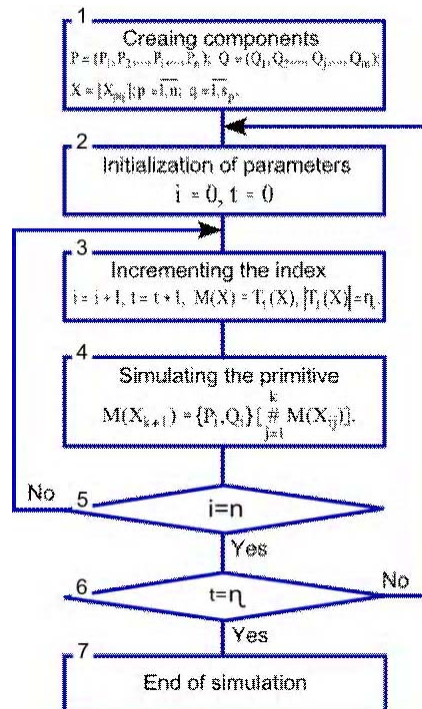


Рис. 6. Граф-схема алгоритма управления процессом моделирования

Здесь представлены операционный и управляющий автоматы, ориентированные на последовательную обработку комбинационных примитивов, что и позволяет осуществлять процедуру переадресации примитивов в случае отказа одного из них. Нетрудно создать аналогичные автоматы для параллельной обработки слоев из примитивов ранжированной схемы, что максимально приблизит быстродействие устройства к его реализации в кристаллах PLD.

## 5. Заключение

1. Предложена теория и реализация квантовых моделей, методов и алгоритмов для повышения быстродействия существующих программных и аппаратных средств анализа и синтеза цифровых вычислительных устройств путем увеличения размерности структур данных и памяти.

2. Описан матричный метод квантового диагностирования функциональных нарушений и константных неисправностей в программных или аппаратных блоках, которые используют кубитные структуры диагностической информации, что дает возможность существенно уменьшить вычислительную сложность процессов моделирования и диагностирования за счет введения параллельных логических операций над матричными данными.

3. Представлен квантовый метод исправного моделирования цифровых устройств с восстановлением работоспособности компонентов цифровой системы в режиме online, который имеет существенно более высокое быстродействие благодаря адресной реализации процедуры обработки функциональных примитивов, заданных кубитными покрытиями.

4. Показаны примеры использования кубитных структур данных и квантовых преобразований для моделирования цифровых схем и решения задач диагностирования с помощью векторных параллельных логических операций и ремонта неисправных модулей на основе адресуемых логических примитивов.

5. Основная инновационная идея квантовых вычислений заключается в переходе от вычислительных процедур над байт-операндом, определяющим в дискретном пространстве одно решение (точку), к квантовым параллельным процессам над кубит-операндом, одновременно формирующим булеан решений. В этом тезисе сформулировано будущее всех высокопроизводительных компьютеров для параллельного нецифрового анализа и синтеза структур и сервисов дискретного по своей сути киберпространства.



*Перспективы исследования:*

1. Использование *or*-операции или для анализа таблицы неисправностей в целях поиска кратных дефектов.
2. Исследование и применение Хассе-процессора для поиска одиночных и/или кратных дефектов по таблице неисправностей, путем приведения процедуры к задаче поиска оптимального покрытия.
3. Моделирование исправного поведения последовательностных схем на основе кубитных структур данных цифровых устройств.
4. Синтез кубитных моделей функциональных примитивов и цифровых систем на основе использования квантовых примитивов, а также разложение (анализ) квантов функциональностей на кубитные модели примитивов.
5. Моделирование неисправностей переходов автоматных переменных путем модификации дедуктивного алгоритма.
6. Формирование кодеров и декодеров для DSP-процессоров с помощью кубитных структур данных, позволяющих повысить быстродействие средств кодирования и компактность представления данных.

**Сисок литературы:** 1. *Michael A. Nielsen & Isaac L. Chuang. Quantum Computation and Quantum Information. Cambridge University Press. 2010. 676p.* 2. *Проектирование и тестирование цифровых систем на кристаллах / В.И. Хаханов, Е.И. Литвинова, О.А. Гузь. Харьков: ХНУРЭ. 2009. 484 с.* 3. *Hahanov V., Wajeb Gharibi, Litvinova E., Chumachenko S. Information analysis infrastructure for diagnosis // Information an int. interdisciplinary journal. 2011. Japan. Vol.14, № 7. P. 2419-2433.* 4. *Хаханов В.И., Мурад Али А., Литвинова Е.И., Гузь О.А., Хаханова И.В. Квантовые модели вычислительных процессов // Радиоэлектроника и информатика. 2011. № 3. С.35-40.* 5. *Бондаренко М.Ф., Хаханов В.И., Литвинова Е.И. Структура логического ассоциативного мультипроцессора // Автоматика и телемеханика. 2012. № 10. С. 71-92.* 6. *Хаханов В.И. Техническая диагностика цифровых и микропроцессорных структур. К.: ИСИО. 1995. 242 с.* 7. *Vladimir Hahanov, Alexander Barkalov and Marian Adamsky. Infrastructure intellectual property for SoC simulation and diagnosis service. Springer, 2011. P. 289-330.* 8. *Г о р б а т о в В.А. Основы дискретной математики. М.: Высш. школа. 1986. 311 с.* 9. *Хаханов В.И., Литвинова Е.И., Хаханова И.В., Murad Ali Abbas. Инфраструктура встроенного восстановления логических PLD-схем / Радиоэлектроника и информатика. 2012. №2. С. 54-57.* 10. *Hahanov V., Litvinova E., Gharibi W., Murad Ali Abbas. Qubit models for SoC Synthesis Parallel and cloud computing. USA. 2012. Vol.1. Iss 1. P. 16-20.* 11. *Hahanov V.I., Litvinova E.I., Chumachenko S.V., Baghdadi Ammar Awni Abbas, Eshetie Abebech, Mandefro. Qubit Model for solving the coverage problem // Proc. of IEEE East-West Design and Test Symposium. IEEE. USA. Kharkov. 14-17 September 2012. P.142 - 144.* 12. *Чжен Г., Мэннинг Е., Метц Г. Диагностика отказов цифровых вычислительных систем. М.: Мир. 1972, 230 с.* 13. *Koal T., Scheit D., Vierhaus H.T. A comprehensive scheme for logic self repair // Conf. Proc. on Signal Processing Algorithms, Architectures, Arrangements, and Applications. 2009. P. 13 – 18.*

*Поступила в редколлегию 11.05.2013*

**Хаханов Владимир Иванович**, декан факультета КИУ ХНУРЭ, д-р техн. наук, профессор кафедры АПВТ ХНУРЭ. IEEE Senior Member. IEEE Computer Society Golden Core Member. Научные интересы: проектирование и тестирование вычислительных систем, сетей и программных продуктов. Увлечения: баскетбол, футбол, теннис, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +380 57 70-21-326.

E-mail: hahanov@kture.kharkov.ua.

**Baghdadi Ammar Awni Abbas**, аспирант кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование вычислительных систем. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +380 57 70-21-326.

E-mail: hahanov@kture.kharkov.ua.

**Чумаченко Светлана Викторовна**, д-р техн. наук, профессор, зав. кафедрой АПВТ ХНУРЭ. Научные интересы: дискретная математика, моделирование вычислительных систем. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +380 57 70-21-326.

E-mail: ri@kture.kharkov.ua.

**Шкиль Александр Сергеевич**, канд. техн. наук, доцент кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и тестирование вычислительных систем. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +380 57 70-21-326.

E-mail: shkil@kture.kharkov.ua.

## АННОТАЦИИ

УДК 681.326:519.713

*В.И. ХАХАНОВ, BAGHDADI AMMAR AWNI ABBAS, С.В. ЧУМАЧЕНКО, А.С. ШКИЛЬ*

### **КВАНТОВЫЕ СТРУКТУРЫ ДЛЯ ТЕСТИРОВАНИЯ ЦИФРОВЫХ УСТРОЙСТВ**

Предлагаются теория и примеры реализации квантовых моделей, методов и алгоритмов для повышения быстродействия существующих программных и аппаратных средств анализа цифровых вычислительных устройств за счет увеличения размерности структур данных и памяти. Вводятся понятия, термины и определения, необходимые для понимания теории и практики квантовых вычислений. Описываются результаты исследований, касающихся моделей и методов квантового диагностирования цифровых систем, кубитного моделирования исправного поведения, восстановления работоспособности отказавших примитивов.

## ABSTRACTS

UDC 681.326:519.713

**Quantum structures for testing digital devices** / V. I. Hahanov, Baghdadi Ammar Awni Abbas, S.V. Chumachenko, A.S. Shkil // Management Information System and Devices. 2013. N 163. P.4-17 .

A theory and examples of quantum models, methods and algorithms to improve the performance of existing software and hardware for analyzing and synthesizing digital computing devices by increasing the dimension of the data structures and memory are proposed. The investigation results of design theory for computer products, as well as models and methods of quantum diagnosis of digital systems, qubit modeling fault-free behavior, analysis of digital devices with repair of failed entities are presented.

Fig. 6. Ref.: 13 items.

## РЕФЕРАТИ

УДК 681.326:519.713

**Квантові структури для тестування цифрових пристроїв** / В.І. Хаханов, Baghdadadi Ammar Awni Abbas, С.В. Чумаченко, А.С. Шкіль // АСУ та прилади автоматики. 2013. Вип. 163. С.4-17.

Запропоновані теорія і приклади реалізації квантових моделей, методів і алгоритмів для підвищення швидкодії існуючих програмних і апаратних засобів аналізу і синтезу цифрових обчислювальних пристроїв за рахунок збільшення розмірності структур даних і пам'яті. Представлені результати досліджень, що стосуються теорії проектування комп'ютерних виробів, а також моделей і методів квантового діагностування цифрових систем, кубітного моделювання справного поведінки, аналізу дискретних пристроїв з відновленням працездатності відмовили примітивів.

Іл. 6. Библиогр.: 13 назв.