

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ УНИВЕРСИТЕТ
РАДИОЭЛЕКТРОНИКИ

ISSN 1563-0064

РАДИОЭЛЕКТРОНИКА И ИНФОРМАТИКА

Научно-технический журнал

Основан в 1997 г.

№ 2(61), апрель – июнь 2013

Выходит 4 раза в год

© *Харьковский национальный
университет радиоэлектроники, 2013*

Свидетельство о государственной регистрации КВ № 12097-968 ПР 14.12.2006

РИ, 2013, № 2

СОДЕРЖАНИЕ

РАДИОТЕХНИКА

МАРТЫНОВ С.А., ХАЖМУРАДОВ М.А. МАТЕМАТИЧЕСКИЕ МОДЕЛИ МЕХАНИЧЕСКИХ НАПРЯЖЕНИЙ И ДЕФОРМАЦИЙ В СИЛОВЫХ КАРКАСАХ ВИНТОВЫХ ОБМОТОК 3

СЛАБОСПИЦКИЙ Р.П., ХАЖМУРАДОВ М.А., ЛУКЬЯНОВА В.П. АНАЛИЗ ПЕРСПЕКТИВНЫХ СИСТЕМ ОХЛАЖДЕНИЯ АККУМУЛЯТОРНЫХ БАТАРЕЙ 8

ЭЛЕКТРОНИКА

ЧЕРНЫШОВ Н.Н. ИССЛЕДОВАНИЕ КРИСТАЛЛОВ БЕЗ ЦЕНТРА ИНВЕРСИИ НА ОСНОВАНИИ КВАНТОВОГО КИНЕТИЧЕСКОГО УРАВНЕНИЯ 13

ТЕЛЕКОММУНИКАЦИИ

СУНДУЧКОВ К.С., КОБЗАР Л.С. МЕТОДИ ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ НАВЧАННЯ НЕЙРОННОЇ МЕРЕЖІ ПРИ СТВОРЕННІ СИСТЕМИ ОЦІНКИ ЯКОСТІ СПРІЙНЯТТЯ ПОСЛУГИ IPTV 15

СИСТЕМЫ И ПРОЦЕССЫ УПРАВЛЕНИЯ

КАЧУР С.А. ДВИЖЕНИЕ КОСМИЧЕСКОГО АППАРАТА НА ОСНОВЕ МОДУЛЯЦИИ КРИВИЗНЫ ПРОСТРАНСТВА..... 22

КОРОБСКАЯ А. В. ОБ ОДНОМ КЛАССЕ НЕСТАЦИОНАРНЫХ СЛУЧАЙНЫХ ПРОЦЕССОВ 27

ГРИШКО С.В., ГУЦА О.М., СУХОМЛІНОВ А.І. МОДЕЛЮВАННЯ ВИРОБНИЧОЇ ЛОГІСТИКИ В УМОВАХ ПЕРЕБУДОВИ ПІДПРИЄМСТВА 31

КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ И ТЕХНИЧЕСКАЯ ДИАГНОСТИКА

БАГДАДИ АММАР АВНИ АББАС (BAHDADI AMMARAWNI ABBAS), ХАХАНОВ В.И., ЛИТВИНОВА Е.И., БУТЕНКО С.А., ЧУМАЧЕНКО С.В. КВАНТОВЫЕ МОДЕЛИ ДИАГНОСТИРОВАНИЯ ЦИФРОВЫХ СИСТЕМ 35

ВЛАСОВ А.В., ЛУКИН В.В., КОМОЛОВ Д.И. КОДИРОВАНИЕ ИНФОРМАЦИОННЫХ РЕСУРСОВ СИСТЕМ ВИДЕОКОНФЕРЕНЦСВЯЗИ ДЛЯ ПОВЫШЕНИЯ ИХ БЕЗОПАСНОСТИ..... 44

КОМПЬЮТЕРНЫЕ НАУКИ

ГУРЖИЙ П.Н., БОЙКО Ю.П., ТРЕТЬЯК В.Ф. АДАПТИВНОЕ ОДНООСНОВНОЕ ПОЗИЦИОННОЕ КОДИРОВАНИЕ МАССИВОВ ДЛИН СЕРИЙ ДВОИЧНЫХ ЭЛЕМЕНТОВ 49

ТУРЕНКО С.В. МОДЕЛЬ ОЦЕНКИ ИНФОРМАТИВНОСТИ НИЗКОЧАСТОТНОЙ СОСТАВЛЯЮЩЕЙ СПЕКТРАЛЬНОГО ПРЕДСТАВЛЕНИЯ СЕГМЕНТА ИЗОБРАЖЕНИЯ 53

ЛЕКАХ А.А. МЕТОД ОБРАБОТКИ ДИНАМИЧЕСКОЙ СОСТАВЛЯЮЩЕЙ НА ОСНОВЕ ПОЗИЦИОННОГО КОДИРОВАНИЯ С АДАПТИВНЫМ ВЫБОРОМ ОСНОВАНИЯ 57

ОЛЕЙНИК А.А., ЗАЙКО Т.А., СУББОТИН С.А. МЕТОД СИНТЕЗА БАЗ ЧИСЛЕННЫХ АССОЦИАТИВНЫХ ПРАВИЛ..... 61

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

ХАХАНОВ В.И., ЛИТВИНОВА Е.И., ЧУМАЧЕНКО С.В., ФИЛИППЕНКО О.И. ИНТЕЛЛЕКТУАЛЬНОЕ ОБЛАКО УПРАВЛЕНИЯ ДВИЖЕНИЕМ (SMART CLOUD TRAFFIC CONTROL) 67

РЕФЕРАТИ..... 77

КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ И ТЕХНИЧЕСКАЯ ДИАГНОСТИКА



УДК658.512.011:681.326:519.713

КВАНТОВЫЕ МОДЕЛИ ДИАГНОСТИРОВАНИЯ ЦИФРОВЫХ СИСТЕМ

БАГДАДИ АММАР АВНИ АББАС (Baghdadi Ammar Awni Abbas), ХАХАНОВ В.И.,
ЛИТВИНОВА Е.И., БУТЕНКО С.А.,
ЧУМАЧЕНКО С.В.

Предлагается новая модель представления данных и матричный метод квантового диагностирования функциональных нарушений в программных или аппаратных блоках, а также константных неисправностей, которые используют кубитные структуры данных квантовых вычислительных процессов, что дает возможность существенно уменьшить вычислительную сложность при установлении диагноза за счет введения параллельных логических операций над матричными данными, составляющими диагностическую информацию. Описываются кубитные структуры данных и квантовый метод исправного моделирования цифровых устройств, который имеет существенно более высокое быстродействие благодаря адресной реализации процедуры обработки функциональных примитивов, заданных Q-покрытиями.

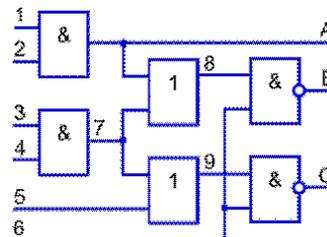
1. Введение

Модель объекта диагностирования представлена в форме цифровой системы, которая имеет функциональные элементы, соединенные линиями связей. Среди них имеются ассерционные точки наблюдения, необходимые для верификации, тестирования и диагностирования неисправностей [1]. Диагностическая информация представлена компонентами: 1) Тест проверки или диагностирования неисправностей заданного класса, в данном случае рассматриваются одиночные константные дефекты $\{ \equiv 0, \equiv 1 \}$ линий схемы. 2) Таблица неисправностей [2], 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 [3]. 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 [4].

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 [5] кодируются соответствующими кубитами $(10, 01, 11, 00)$ 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.

2. Квантовый метод диагностирования цифровых систем

Для рассмотрения сущности предлагаемого метода используется фрагмент цифровой схемы, представленной на рис. 1. Здесь имеются три ассерционные точки А, В, С для наблюдения за состоянием всех линий схемы в процессе тестирования (выполнения диагностического эксперимента) путем подачи пяти тестовых воздействий, заданных в таблице неисправностей $T \setminus F$. Координаты данной таблицы задают проверяемые на тест-векторах неисправности 0 и 1, а также имеются состояния координат: $\emptyset(\cdot)$ – 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. Значение 1 означает несравнение, 0 – совпадение упомянутых реакций.



T \ F	1	2	3	4	5	6	7	8	9	A	B	C	V _a	V _b	V _c
111101	0	0	0	0	.	0	0	0	0	1	1	1	1	0	0
010101	1	.	1	.	1	.	1	1	1	0	0	0	0	1	1
101001	.	1	.	1	1	.	1	1	1	0	0	0	0	0	0
000011	0	0	1	1	0	1	0	1	0	0	0
111110	0	0	.	.	1	.	.	.	0	0	0	0	1	1	1

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 _{ij}	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	.	.	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: $S^i(T_i) = (\bigvee_{A_j=1} S_j) \wedge (\overline{\bigvee_{A_j=0} S_j})$. Каждый

тест-вектор (-сегмент) активизирует собственную структуру, поэтому маска возможных дефектов функционально зависит от структуры, ассерций и тестовых наборов:

$S^i = f(S, A, T_i)$. Если предположить, что в матрице $S = S_{ij}$ состояния выходов равны (A, B, C) = (011), то маска возможных дефектов будет иметь следующий вид:

$S^i = (S_2 \vee S_3) \wedge (\overline{S_1}) = (111101110010 \vee 001111110100) \wedge (\overline{110000000100}) = (111111110101) \wedge (001111110101) = (001111110101)$.

Полученная маска накладывается на первую строку таблицы неисправностей, что определяет множество подозреваемых неисправностей, формирующих ассерционную 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	.	1	.	1

В соответствии с предложенной процедурой получения маски одной строки выполняется построение матрицы структурной активизации неисправностей на основе использования матрицы экспериментальной проверки (состояния ассерционного механизма):

S = S _{ij}	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	.	.	1

 $\xrightarrow{S^i(T_i)}$

T \ A B C	T1	1	0	0
T2	0	1	1	
T3	0	0	0	
T4	0	0	0	
T5	1	1	1	

 $=$

S(T)	1	2	3	4	5	6	7	8	9	A	B	C
T1	1	1	0	0	0	0	0	0	1	0	0	0
T2	0	0	1	1	1	1	1	0	1	1	1	1
T3	0	0	0	0	0	0	0	0	0	0	0	0
T4	0	0	0	0	0	0	0	0	0	0	0	0
T5	1	1	1	1	1	1	1	1	1	1	1	1

Модель (базовая) диагностирования дискретного процесса или явления представлена компонентами, которые создают 4 измерения в пространстве признаков:

$$\begin{aligned} \#D_b &= \langle S, A, F, T \rangle \\ \#D &= \langle S, A \rangle, \langle F, T \rangle; \end{aligned}$$

$$\begin{aligned} \#P_b &= (\#S \times \#A \times \#F \times \#T); \\ \#P &= (\#S \times \#A) + (\#F \times \#T); \\ \#P_b &\gg P; \end{aligned}$$

$$\begin{aligned} \#S^i &= f(S, A, T); \\ \#V^t &= g(T, A); \end{aligned}$$

При этом объем диагностической информации P формируется произведением мощностей четырех компонентов: 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. 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 объемной сложности становится не мультипликативной, а аддитивной по отношению к мощности подмножеств без какого-либо уменьшения глубины диагностирования. 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:

F(T)	1	2	3	4	5	6	7	8	9	A	B	C
T1	1	1	0	0	0	0	0	1	0	0	0	0
T2	0	0	1	1	1	1	1	0	1	1	1	1
T3	0	0	0	0	0	0	0	0	0	0	0	0
T4	0	0	0	0	0	0	0	0	0	0	0	0
T5	1	1	1	1	1	1	1	1	1	1	1	1

 $\xrightarrow{S(T_i)}$

F(T)	1	2	3	4	5	6	7	8	9	A	B	C
T1	1	1	0	0	0	0	0	0	1	0	0	0
T2	0	0	1	1	1	1	1	0	1	1	1	1
T3	0	0	0	0	0	0	0	0	0	0	0	0
T4	0	0	0	0	0	0	0	0	0	0	0	0
T5	1	1	1	1	1	1	1	1	1	1	1	1

F	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
10	11	01	11	01	11	01	01	01	10	10	10	10
11	10	11	10	10	11	10	10	10	10	01	01	01
11	11	11	11	01	01	10	10	01	10	01	10	10
10	10	11	11	11	01	11	11	11	10	10	10	10
C =	10	10	01	00	00	01	00	00	00	10	00	00
D =	0	0	1	.	.	1	.	.	.	0	.	.

Введенная операция, близкая к координатной операции отрицания кубитов (двухразрядных векторов, кодирующих алфавит Кантора), позволяет на практике реализовать функциональность диагностирования только путем логического умножения векторов матрицы кодов неисправностей:

$$\begin{aligned} F(T_i) &= (\bigvee_{A_j=1} F_j) \wedge (\overline{\bigvee_{A_j=0} F_j}) = \\ &= (\bigwedge_{A_j=1} F_j) \wedge (\overline{\bigvee_{A_j=0} F_j}) = (\bigwedge_{A_j=1} F_j) \wedge (\bigwedge_{A_j=0} \overline{F_j}) = (\bigwedge_{j=1}^n F_j) \cdot \end{aligned}$$

Предварительно выполняется операция инверсии над ячейками матрицы кодов неисправностей, маскируемыми нулевыми сигналами матрицы активизации, а также над всеми нулевыми кодами таблицы неисправностей: $F_{ij} = F_j \leftarrow (F_j = 00) \vee (S_{ij} = 0)$:

Δ	00	01	10	11
0	11	10	01	00
1	11	01	10	11

Δ	\emptyset	1	0	X
0	X	0	1	\emptyset
1	X	1	0	X

Таблица истинности 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 и блокирует все вычисления конъюнкции по столбцу, превращая результат в 00. Инверсия кода дает возможность не маскировать при умножении действительно присутствующих дефектов, любых знаков. При этом предполагается, что невозможностей-вектором проверить на одной линии схемы дефекты разных знаков.

Теорема 1. *Одиночные константные дефекты цифровой схемы, заданные кубитами на тестовых наборах многозначной таблицы неисправностей, определяются с помощью векторной and-операции, маскируемой по строкам вектором экспериментальной проверки:*

$$S(T_i) = (\bigvee_{A_j=1} S_j) \wedge (\overline{\bigvee_{A_j=0} S_j}) = (\bigwedge_{A_j=1} S_j) \wedge (\bigwedge_{A_j=0} \bar{S}_j) = (\bigwedge_{j=1}^n S_j)$$

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. Действительно, на одной линии или переменной не могут присутствовать одновременно две противоположных по знаку проверяемых неисправности. Поэтому в базовой формуле дизъюнкция дефектов в большей степени ориентирована на поиск кратных неисправностей, ноне 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- and- операций, маскируемых по строкам вектором экспериментальной проверки:*

$$S(T_i) = (\bigvee_{A_j=1} S_j) \wedge (\overline{\bigvee_{A_j=0} S_j}) = (\bigvee_{A_j=1} S_j) \wedge (\bigwedge_{A_j=0} \bar{S}_j)$$

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 кратных дефектов в предположении, что на одной линии или переменной могут присутствовать одновременно две противоположных по знаку проверяемых неисправности. Данная формула в большей степени ориентирована на поиск кратных неисправностей в блоках цифровых систем, не связанных с одной линией. Кратность дефектов 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, не влияющие на формирование некорректных реакций по выходам.

Интерес представляет поиск кратных дефектов на основе мультипроцессора Хассе, который ориентирован на решение задачи покрытия путем полного перебора событий, обеспечивающих точное покрытие вектора экспериментальной проверки столбцами таблицы неисправностей: $S(T_i) = (\bigvee_j S_j) \oplus A = 0$.

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

F	1	2	3	4	5	6	7	8	9	A	B	C	V _a	V _b	V _c
111101	10	10	01	01	11	01	01	01	01	10	10	10	1	0	0
010101	10	11	01	11	01	11	01	01	01	10	10	10	0	1	1
101001	11	10	11	10	10	11	10	10	10	10	01	01	0	0	0
000011	11	11	11	11	01	01	10	10	01	10	01	10	0	0	0
111110	10	10	11	11	11	01	11	11	11	10	10	10	1	1	1
D	10	10	01	00	00	01	00	00	00	10	00	00			
D*	0	0	1	.	.	1	.	.	.	0	.	.			

Таким образом, модель диагностирования содержит следующие преобразователи:

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

Моделирование реального устройства на тестовых наборах и сравнение результатов с эталоном по наблюдаемым линиям-ассерциям, что дает возможность сформировать матрицу выходных реакций или экспериментальной проверки в двоичном алфавите.

Вычисление матрицы активности графовой структуры на каждом входном тестовом наборе, равной по размерности таблице неисправностей, спомощью матрицы экспериментальной проверки и матрицы достижимостей.

Модификация содержимого таблицы неисправностей путем применения матрицы активности графовой структуры, в целях индикации неисправностей, действительно проверяемых тестовыми наборами.

Выполнение процедуры векторного логического умножения над строками таблицы для получения вектора подозреваемых дефектов

3. Кубитные модели описания цифровых систем

Дефект определяется как каждое отдельное несоответствие требованиям спецификации, но при этом модель дефекта не должна выводить изделие за преде-

Таким образом, кубит, составленный из четырех битов, способен описать 16 состояний пространственно-временного фрейма функциональностей и при этом оперировать векторными логическими операциями для решения задач синтеза и анализа.

4. Квантовое моделирование цифровых систем

Например, для описания цифровой схемы, представленной на рис. 2, необходимо иметь структуру взаимосвязанных элементов и кубическое покрытие логического элемента И-НЕ.

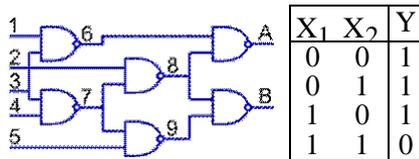


Рис. 2. Фрагмент цифровой схемы

Данное покрытие можно трансформировать с учетом унитарного кодирования входных векторов. Как следствие, для любого покрытия функционального одно-выходового примитива всегда имеется только 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:

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

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 одно-выходового 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, моде-

лирования 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):

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

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

$$F = \langle L, M, X, Q \rangle,$$

$$L = (L_1, L_2, \dots, L_j, \dots, L_n);$$

$$M = (M_1, M_2, \dots, M_j, \dots, M_n);$$

$$X = (X_{x+1}, X_{x+2}, \dots, X_{x+i}, \dots, X_{x+n});$$

$$Q = (Q_{x+1}, Q_{x+2}, \dots, Q_{x+i}, \dots, Q_{x+n}).$$

Здесь представлены: вектор-идентификаторы эквипотенциальных линий схемы цифровой системы, 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 входным переменным примитивов; n – число линий в схеме, x – количество входных переменных.

В качестве примера кубитного задания модели цифрового устройства $F = \langle L, M, X, Q \rangle$, представленно на рис. 2, 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

Процедура исправного моделирования сводится к определению адреса, формируемого двоичными состояниями входных переменных каждого примитива цифровой схемы. 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 пос-

ледовательной процедурой моделирования на основе простых итераций или итераций Зейделя. 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. 2, 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.

Структуры кубитных данных модифицируются в сторону дополнения строкой типов примитивов $F = \langle L, M, X, P, Q \rangle$, $P = (P_1, P_2, \dots, P_1, \dots, P_m)$, действующих при синтезе цифровой системы, если необходимо в процессе функционирования выполнять ремонт или восстановление работоспособности путем введения запасных примитивов, которые так же как и основные, реализуются на основе элементов памяти. В следующей таблице представлены строка типов логических элементов, а также адреса 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
Q	1	1	1	1	1	1	1	1	1	1	1
	1	1	1	1	1	1	1	1	1	1	1
	1	1	1	1	1	1	1	1	1	1	1
	0	0	0	0	0	0	0	0	0	0	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	1	1	1	1
	1	1	1	1	1	1	1	1	1	1	1
	0	0	0	0	0	0	0	0	0	0	0

Вторая таблица оперирует номерами 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 первой таблице, ориентированной на программную реализацию моделирования, ремонтные примитивы начинаются с номера 2. Если существует возможность перепрограммирования логики в элемент 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
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.

5. Модель комбинационной самовосстанавливаемой структуры

Немногочисленные работы, посвященные восстановлению работоспособности логических схем [9], описывают две идеи. Первая заключается в реконфигурации структуры логических элементов в режиме off-line, которая обеспечивает возможность замены каждого из неисправных примитивов. Вторая создает условия для замены неисправных элементов путем использования запасных логических компонентов и расширения мультиплексоров для переадресации отказавших примитивов. Далее предлагается в качестве примера для рассмотрения теории и практики адресации примитивов комбинационных схем в целях встроеного ремонта функциональных нарушений логических элементов использовать описание простейшей схемной структуры (см.рис. 2).

Она содержит шесть однотипных логических элементов, которые можно представить в адресном пространстве следующим списком (двумерным массивом):

$$S = \begin{matrix} N_0 = & \begin{matrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \end{matrix} \\ P = & \begin{matrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{matrix} \\ L_1 = & \begin{matrix} 1 & 3 & 2 & 7 & 6 & 8 & X & X & X \end{matrix} \\ L_2 = & \begin{matrix} 3 & 4 & 7 & 5 & 8 & 9 & X & X & X \end{matrix} \\ L_3 = & \begin{matrix} 6 & 7 & 8 & 9 & A & B & Y & Y & Y \end{matrix} \end{matrix} \quad F(1) = \begin{matrix} X_1 & X_2 & Y \\ 0 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{matrix}$$

Каждый столбец соответствует логическому элементу схемы, а примитивы с номерами 7, 8, 9 являются запасными, которые используются для замены любых

трех из шести элементов при диагностировании в последних каких-либо функциональных нарушений. В строке Р указаны типы примитивов, ниже – номера входных и выходных переменных, вектор моделирования М содержит результат анализа входного слова 11111 на схемной структуре, представленной рис. 3.

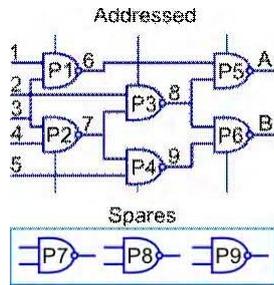


Рис. 3. Пример схемной структуры из адресуемых и запасных элементов

Процесс-модель формирования выходных значений схемы в зависимости от конкатенированных состояний входов, формирующих адрес ячейки состояния выхода, можно представить в абстрактном или аппаратно ориентированном, на использование вектора состояния М, виде:

$$\begin{aligned} \{Y_6 = P_1(X_1 * X_3); \} & \quad \{M_6 = P_1(M_1 * M_3); \} \\ \{Y_7 = P_2(X_3 * X_4); \} & \quad \{M_7 = P_2(M_3 * M_4); \} \\ \{Y_8 = P_3(X_2 * X_7); \} & \quad \{M_8 = P_3(M_2 * M_7); \} \\ \{Y_9 = P_4(X_7 * X_5); \} & \quad \{M_9 = P_4(M_7 * M_5); \} \\ \{Y_A = P_5(X_6 * X_8); \} & \quad \{M_A = P_5(M_6 * M_8); \} \\ \{Y_B = P_6(X_8 * X_9); \} & \quad \{M_B = P_6(M_8 * M_9); \} \end{aligned}$$

Здесь М – вектор состояния линий схемы, P_i – логическая функция И-НЕ, имеющая два входа, реализованная в виде элемента памяти LUT. Поскольку все шесть примитивных элементов реализуют одну логическую функцию И-НЕ, то предыдущее выражение можно упростить путем замены всех структурных элементов одним функциональным примитивом F с последующим использованием для операции конкатенации двумерного массива линий связи (L) между входами и выходами логических элементов:

$$\begin{aligned} \{M_6 = F(M_1 * M_3); \} & \quad \{M_6 = F[M(L_{11}) * M(L_{12})]; \} \\ \{M_7 = F(M_3 * M_4); \} & \quad \{M_7 = F[M(L_{21}) * M(L_{22})]; \} \\ \{M_8 = F(M_2 * M_7); \} & \quad \{M_8 = F[M(L_{31}) * M(L_{32})]; \} \\ \{M_9 = F(M_7 * M_5); \} & \quad \{M_9 = F[M(L_{41}) * M(L_{42})]; \} \\ \{M_A = F(M_6 * M_8); \} & \quad \{M_A = F[M(L_{51}) * M(L_{52})]; \} \\ \{M_B = F(M_8 * M_9); \} & \quad \{M_B = F[M(L_{61}) * M(L_{62})]; \} \end{aligned}$$

Таким образом, можно синтезировать структуру для реализации процесс-модели схемы, имеющей двух-входовые функциональные примитивы, в следующем виде:

$$M(L_{isp}) = F[M(L_{ij}) * M(L_{ir})] = F[M(L)].$$

Учитывая факт, что все вычисления в схеме привязаны структурным элементам, которые имеют идентификатор логической операции, предыдущую формулу можно трансформировать к виду:

$$M(L_{isp}) = P_i[M(L_{ij}) * M(L_{ir})] = P[M(L)].$$

В общем случае структура модели функциональности, ориентированной на реализацию в кристалле PLD, содержит пять компонентов:

$$S = \langle P, F, M, L, T \rangle,$$

$$P = (P_1, P_2, \dots, P_i, \dots, P_n);$$

$$F = (F_1, F_2, \dots, F_j, \dots, F_m);$$

$$M = (M_1, M_2, \dots, M_r, \dots, M_k);$$

$$L = [L_{pq}]; p = 1, n; q = 1, s_p;$$

$$T = [T_{te}]; t = 1, \eta; e = 1, \mu;$$

$$M(L) = P[M(L)].$$

Здесь представлены: 1) примитивы схемной структуры Р, определенные идентификаторами типа функциональности (номер или код команды); 2) типы функциональных элементов F – набор элементов памяти LUT, из которых реализуются примитивы, а также избыточные элементы для ремонта функциональности; 3) вектор моделирования М (двоичный), определяющий состояния всех линий (входные, внутренние, выходные); 4) матрица эквипотенциальных линий связи L для объединения логических элементов в структуру;

5) матрица входных тестовых (рабочих) наборов Т. Обработка схемы (processing) 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 L для формирования адреса состояния входной переменной вектора моделирования: $A = L_{ij}, i = 1, n; j = 1, s_p - 1$.

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

3) Запись результата выполнения логической функции как состояния выхода в соответствующий разряд вектора моделирования

$$M(L_{isp}) = F[M(L_{ij}) * M(L_{ir})].$$

6. Операционное устройство для моделирования комбинационной структуры

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 восста-

новления работоспособности цифровой структуры, который является одним из этапов функционирования инфраструктуры обслуживания SoC, представленной на рис. 4.

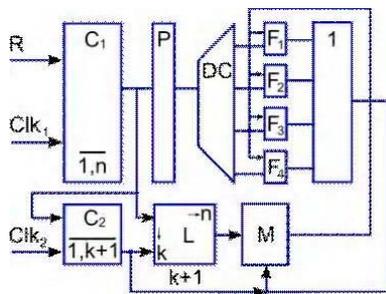


Рис. 4. Операционная структура комбинационной схемы

Комбинационная схема становится операционным устройством, где присутствуют операционный и управляющий автоматы. Заменяемыми компонентами в операционном автомате являются типы примитивов – функциональные элементы (рис. 5).

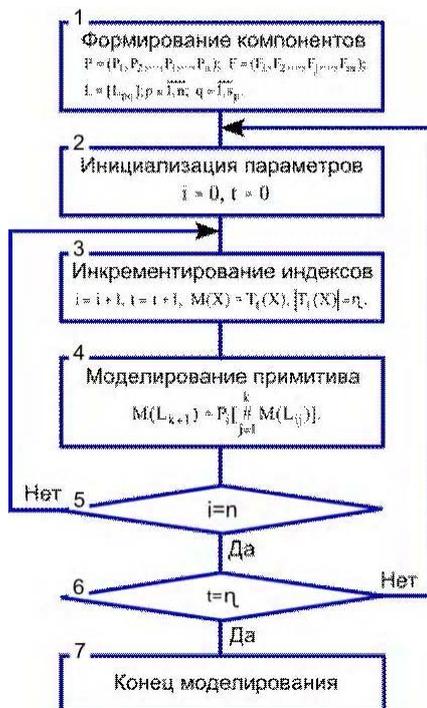


Рис. 5. Граф-схема алгоритма управления процессом моделирования

Операционное устройство реализации элементо-адресуемых комбинационных схем содержит: счетчик обработки текущего примитива C_1 ; память для хранения типов примитивов, соответствующих структурным элементам P ; счетчик считывания номеров входных и выходной переменных текущего примитива C_2 ; дешифратор типов примитивов DC ; память для хранения вектора моделирования M ; матричную память для хранения номеров входов-выходов структурных примитивов L ; линейку памяти, реализующих функциональные примитивы F ; регистр форми-

рования входного адресного слова для обрабатываемого примитива RG ; логический элемент Or для коммутации результатов обработки функциональных примитивов.

Граф-схема алгоритма управления процессом моделирования структуры комбинационной схемы представлена на рис. 5.

1. Инициализация (формирование) всех компонентов (номера и типы элементов, линии связей для входов и выходов логических элементов) схемной структуры:

$$P = (P_1, P_2, \dots, P_i, \dots, P_n); F = (F_1, F_2, \dots, F_j, \dots, F_m);$$

$$L = [L_{pq}]; p = \overline{1, n}; q = \overline{1, s_p}.$$

2. Инициализация параметра обрабатываемого примитива и номера входного набора $i = 0, t = 0$ для его моделирования в двоичном алфавите $M_T = \{0, 1\}$.

3. Инкрементирование индекса примитива, номера теста и инициализация входного тестового (рабочего) набора: $i = i + 1, t = t + 1, M(X) = T_t(X), |T_t(X)| = \eta$.

4. Конкатенация (#) разрядов слова для формирования входного воздействия $\#_{j=1}^k M(L_{ij})$ логического

элемента P_i и выполнение процедуры определения состояния его выхода с последующей записью в соответствующую координату вектора моделирова-

ния: $M(L_{k+1}) : M(L_{k+1}) = P_i [\#_{j=1}^k M(L_{ij})]$.

5. Повторение пунктов 3 и 4 в целях получения состояний выходов всех логических элементов до выполнения условия: $i = n$.

6. Повторение пунктов 2–4 в целях моделирования всех входных тестовых (рабочих) наборов до выполнения равенства: $t = \eta$, где η – длина теста.

7. Окончание процесса моделирования цифрового устройства.

7. Заключение

Представлен аппарат квантового описания цифровых компонентов, который дает возможность существенно повысить быстродействие процедур синтеза и анализа за счет увеличения размерности векторов для унитарного кодирования символов теоретико-множественного алфавита.

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

Направления будущих исследований:

1. Использование og-операции или для анализа таблицы неисправностей вцеляхпоиска кратных дефектов.
2. Исследование и применение Хассе, процессора для поиска одиночных и/или кратных дефектов по таблице неисправностей путем приведенияпроцедуры к задаче поиска оптимального покрытия.
3. Моделирование исправного поведения последовательностных схем на основе кубитных структур данных цифровых устройств.
4. Синтезкубитных моделейфункциональныхпримитивов.
5. Моделирование неисправностей переходов автоматных переменных путем модификации дедуктивного алгоритма.

Литература: 1. *Проектирование и тестирование цифровых систем на кристаллах* / В.И. Хаханов, Е.И. Литвинова, О.А. Гузь. Харьков: ХНУРЭ. 2009. 484 с. 2. *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. 3. *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. 4. *Бондаренко М.Ф., Хаханов В.И., Литвинова Е.И.* Структура логического ассоциативного мультипроцессора // Автоматика и телемеханика. 2012. № 10. С. 71-92. 5. *Хаханов В.И.* Техническая диагностика цифровых и микропроцессорных структур. К.: ИСИО. 1995. 242 с. 6. *Хаханов В.И., Баркалов О.О., Мариан Адамски.* Infrastructure intellectual property for SoC simulation and diagnosis service. Springer, 2011. P. 289-330. 7. *Чжен Г., Мэннинг Е., Метц Г.* Диагностика отказов цифровых вычислительных систем. М.: Мир, 1972. 230 с. 8. *Горбатов В.А.* Основы дискретной математики. М. : Высшая 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.

Поступила в редколлегию 11.06.2013

Рецензент: д-р техн. наук, проф. Баркалов А.А.

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.

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. +38 057 70-21-326, E-mail: hahanov@kture.kharkov.ua.

Литвинова Евгения Ивановна, д-р техн. наук, профессор кафедры АПВТ ХНУРЭ. Научные интересы: автоматизация диагностирования и встроенный ремонт компонентов цифровых систем в пакете кристаллов. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-421. E-mail: kiu@kture.kharkov.ua.

Бутенко Сергей Александрович, студент 4-го курса факультета КИУ ХНУРЭ. Научные интересы: автоматизация диагностирования и встроенный ремонт компонентов цифровых систем в пакете кристаллов. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-421.

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.

АННОТАЦИИ

УДК 519.216

УДК658.512.011:681.326:519.713

КВАНТОВЫЕ МОДЕЛИ ДИАГНОСТИРОВАНИЯ ЦИФРОВЫХ СИСТЕМ

БАГДАДИ АММАР АВНИ АББАС (Baghdadi Ammar Awni Abbas), ХАХАНОВ В.И., ЛИТВИНОВА Е.И., БУТЕНКО С.А., ЧУМАЧЕНКО С.В.

Предлагается новая модель представления данных и матричный метод квантового диагностирования функциональных нарушений в программных или аппаратных блоках, а также константных неисправностей, которые используют кубитные структуры данных квантовых вычислительных процессов, что дает возможность существенно уменьшить вычислительную сложность при установлении диагноза за счет введения параллельных логических операций над матричными данными, составляющими диагностическую информацию. Описываются кубитные структуры данных и квантовый метод исправного моделирования цифровых устройств, который имеет существенно более высокое быстродействие благодаря адресной реализации процедуры обработки функциональных примитивов, заданных Q-покрытиями.

ABSTRACTS

UDC 658.512.011:681.326:519.713

Quantum models of diagnosis of digital systems / Baghdad Ammar Abbas Avni, V.I. Hahanov, E.I. Litvinova, S.A. Butenko, S.V. Chumatchenko // Radioelektronika i informatika. 2013. N 2. P. 35-43.

A new model for data representation and quantum matrix method of diagnosing functional disorders in the software or hardware blocks, as well as constant faults, data structures used qubit quantum computational processes, which allows to significantly reduce the computational complexity at diagnosis due to the introduction of parallel logic operations on matrix information constituting diagnostic information. Qubit described data structures and quantum method serviceable modeling of digital devices, which has significantly higher performance thanks to the implementation of procedures for processing address functional primitives defined Q-covers.

Fig. 5 . Ref.: 9 items.

РЕФЕРАТИ

УДК 658.512.011:681.326:519.713

Квантові моделі діагностування цифрових систем / Баг дад Аммар Авні АББАС (Багдаді Аммар Авні Аббас), В.І. Хаханов, Є.І. Литвинова, С.А. Бутенко, С.В. Чумаченко // *Радіоелектроніка та інформатика*. 2013. № 2. С. 35-43.

Запропонована нова модель представлення даних і матричний метод квантового діагностування функціональних порушень в програмних або апаратних блоках, а також константні несправності, які використовують кубітні структури даних квантових обчислювальних процесів, що дає можливість істотно зменшити обчислювальну складність при встановленні діагнозу за рахунок введення паралельних логічних операцій над матричними даними, складовими діагностичної інформації. Описані кубітні структури даних і квантовий метод справного моделювання цифрових пристроїв, який має істотно більш високу швидкість завдяки адресній реалізації процедури обробки функціональних примітивів, заданих Q- покриттями.

Іл. 5. Бібліогр.: 9 назв.