

ДЕДУКТИВНО-ПАРАЛЛЕЛЬНЫЙ МЕТОД МОДЕЛИРОВАНИЯ НЕИСПРАВНОСТЕЙ НА РЕКОНФИГУРИРУЕМЫХ МОДЕЛЯХ ЦИФРОВЫХ СИСТЕМ

ХАХАНОВ В.И., СЫСЕНКО И.Ю.,
КОЛЕСНИКОВ К.В.

Предлагается быстродействующий метод моделирования неисправностей, представляющий сочетание достоинств дедуктивного и параллельного алгоритмов, ориентированный на обработку цифровых устройств вентильного и регистрового уровней описания.

1. Введение

Достижения микроэлектроники позволяют создавать интегральные микросхемы, содержащие миллионы эквивалентных вентилях на кристалле. Это дает возможность проектировать сложные специализированные вычислительные устройства с помощью средств автоматизированного проектирования известных фирм (Aldec, Cadence, Altera, Xilinx, Synopsys). Однако автоматизация процесса верификации таких структурно- и функционально-сложных цифровых систем требует создания новых методов и средств синтеза тестов и моделирования неисправностей, способных за приемлемое время построить входные последовательности проверки дефектов необходимой полноты. Быстродействие реализации особенно псевдослучайных алгоритмов генерации тестов на 80-90% зависит от времени анализа неисправностей. Поэтому разработка нового быстродействующего метода моделирования одиночных константных дефектов для синтеза тестов верификации проектируемых цифровых систем на основе программируемой логики является актуальной проблемой.

Объект тестирования представлен в форме булевых уравнений, записанных на языке VHDL, реализующих сложную цифровую систему, имплементируемую в кристаллы программируемой логики.

Ниже рассмотрены вопросы, связанные с теоретическим обоснованием объединения методов кубического [1-4], дедуктивного и параллельного [5-7] моделирования в целях повышения быстродействия анализа дефектов и определения качества теста. Также рассмотрены особенности и достоинства алгоритмической и аппаратной реализации предлагаемого метода, его вычислительная сложность и примеры моделирования тестовых схем.

2. Математическая модель анализа неисправностей

Основное уравнение тестирования цифровой системы $F = (F_1, F_2, \dots, F_1, \dots, F_n)$, включающей n линий и/или функциональных элементов, представлено в виде [2-4]:

$$L \oplus F = T, \quad (1)$$

где тест $T = (T_1, T_2, \dots, T_t, \dots, T_k)$ для реконфигурируемых на нем моделей $L = (L_1, L_2, \dots, L_t, \dots, L_k)$ анализа неисправностей определяется линейным взаимодействием упомянутых компонентов по правилу (1). Данное уравнение может быть трансформировано к виду

$$L \oplus T = F. \quad (2)$$

В этом случае формализуется процесс идентификации функции при решении задач диагностирования.

Последний вариант линейной перестановки, задаваемый уравнением

$$T \oplus F = L, \quad (3)$$

определяет правила формирования множества дедуктивных функций параллельного моделирования неисправностей (ДФПМН или ДФ) на тесте T для модели исправного поведения F . Естественно, что каждой ДФПМН можно поставить в соответствие схему, тогда аббревиатура будет иметь вид ДСПМН или ДС.

Компонент функционального описания цифровой системы $F_i \in F$ представляет собой булеву функцию для вычисления состояния i -й линии:

$$F_i = f_i(X_{i1}, X_{i2}, \dots, X_{ij}, \dots, X_{in_i}). \quad (4)$$

В качестве результата исправного моделирования функции f_i выступает значение координаты $T_{ti} \in T_t$, причем $T_{ti} = F_i$ на тесте t . При этом имеется в виду, что тест есть матрица исправного поведения цифровой системы

$$T = [T_{ti}] = (T_{t1}, T_{t2}, \dots, T_{ti}, \dots, T_{tn}). \quad (5)$$

С учетом разбиения теста на составляющие векторы уравнение (3) получения ДФПМН для $T_t \in T$ принимает следующий вид:

$$L_t = T_t \oplus F.$$

При условии, что функциональное описание цифровой системы представлено компонентами, формирующими состояния всех линий схемы, в качестве формулы преобразования исправной модели в дедуктивную функцию выступает выражение

$$L_{ti} = f_i[(X_{i1} \oplus T_{t1}), (X_{i2} \oplus T_{t2}), \dots, (X_{ij} \oplus T_{tj}), \dots, (X_{in_i} \oplus T_{tn_i})] \oplus T_{ti}, \quad (6)$$

которое по существу аналогично формуле дедуктивного анализа цифровых схем, представленной в [6,8].

Пример 1. Для логического элемента 2И определить ДФПМН на тест-векторе $T = (111)$.

Решение. В соответствии с (6) выполняются эквивалентные преобразования с использованием тождеств алгебры логики (\neg – знак инверсии):

$$\begin{aligned} L(T=11, Y=1, X_1 \wedge X_2) &= [(X_1 \oplus T_1) \wedge (X_2 \oplus T_2)] \oplus T_3 = \\ &= [(X_1 \oplus 1) \wedge (X_2 \oplus 1)] \oplus 1 = (\bar{X}_1 \wedge \bar{X}_2) \oplus 1 = \\ &= -(\bar{X}_1 \wedge \bar{X}_2) = (X_1 \vee X_2). \end{aligned}$$

Аналогично получается ДФПМН для функции 2ИЛИ и тест-вектора $T = (111)$:

$$\begin{aligned} L(T=11, Y=1, X_1 \vee X_2) &= [(X_1 \oplus T_1) \vee (X_2 \oplus T_2)] \oplus T_3 = \\ &= [(X_1 \oplus 1) \vee (X_2 \oplus 1)] \oplus 1 = (\bar{X}_1 \vee \bar{X}_2) \oplus 1 = \\ &= -(\bar{X}_1 \vee \bar{X}_2) = (X_1 \wedge X_2). \end{aligned}$$

Путем построения ДФПМН на всех возможных входных наборах для двухвходового элемента И можно получить универсальную дедуктивную функцию, инвариантную инверсии:

$$\begin{aligned} L\{T=(00,01,10,11), (X_1 \wedge X_2)\} &= \\ &= L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2)[(X_1 \oplus T_1 \wedge X_2 \oplus T_2) \oplus T_3]\} = \\ &= (\bar{x}_1\bar{x}_2)[(X_1 \oplus 0) \wedge (X_2 \oplus 0)] \oplus 0 \vee \\ &\vee (\bar{x}_1x_2)[(X_1 \oplus 0) \wedge (X_2 \oplus 1)] \oplus 0 \vee \\ &\vee (x_1\bar{x}_2)[(X_1 \oplus 1) \wedge (X_2 \oplus 0)] \oplus 0 \vee \\ &\vee (x_1x_2)[(X_1 \oplus 1) \wedge (X_2 \oplus 1)] \oplus 1 = \\ &= (\bar{x}_1\bar{x}_2)(X_1 \wedge X_2) \vee (\bar{x}_1x_2)(X_1 \wedge \bar{X}_2) \vee \\ &\vee (x_1\bar{x}_2)(\bar{X}_1 \wedge X_2) \vee (x_1x_2)(X_1 \vee X_2). \end{aligned}$$

Здесь строчными буквами представлены термы, обозначающие входные тест-наборы, прописными – регистровые переменные, кодирующие векторы проверяемых неисправностей каждой существенной переменной или линии цифрового устройства.

Аналогичное преобразование для функции ИЛИ дает следующую ДФПМН:

$$\begin{aligned} L\{T=(00,01,10,11), (X_1 \vee X_2)\} &= \\ &= L\{(\bar{x}_1\bar{x}_2 \vee \bar{x}_1x_2 \vee x_1\bar{x}_2 \vee x_1x_2)[(X_1 \oplus T_1 \vee X_2 \oplus T_2) \oplus T_3]\} = \\ &= (\bar{x}_1\bar{x}_2)[(X_1 \oplus 0) \vee (X_2 \oplus 0)] \oplus 0 \vee \\ &\vee (\bar{x}_1x_2)[(X_1 \oplus 0) \vee (X_2 \oplus 1)] \oplus 1 \vee \\ &\vee (x_1\bar{x}_2)[(X_1 \oplus 1) \vee (X_2 \oplus 0)] \oplus 1 \vee \\ &\vee (x_1x_2)[(X_1 \oplus 1) \vee (X_2 \oplus 1)] \oplus 1 = \\ &= (\bar{x}_1\bar{x}_2)(X_1 \vee X_2) \vee (\bar{x}_1x_2)(\bar{X}_1 \wedge X_2) \vee \\ &\vee (x_1\bar{x}_2)(X_1 \wedge \bar{X}_2) \vee (x_1x_2)(X_1 \wedge X_2). \end{aligned}$$

Как следствие объединения полученных ДФПМН логических элементов И, ИЛИ дедуктивная схема их параллельного моделирования представлена на рис.1.

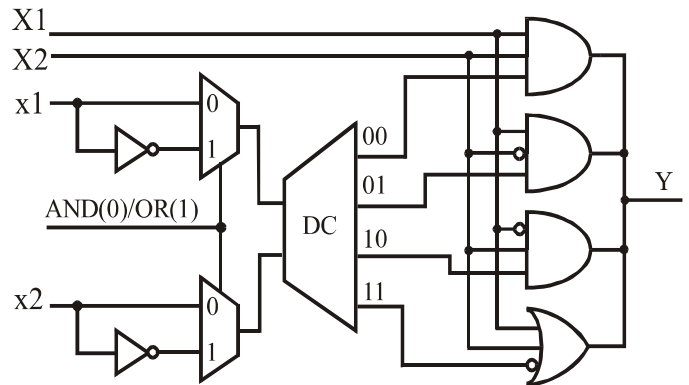


Рис. 1. Схема параллельного моделирования неисправностей для логических элементов AND, OR

В схеме имеют место регистровые переменные $X1$ и $X2$, которые представляют собой векторы проверяемых дефектов для каждой линии цифрового устройства, объединенные в матрицу $M = [M_{ij}]$ размерностью n^2 . Предварительно данная матрица инициализируется нулями с единичными диагональными элементами, иначе является единичной матрицей:

$$[M_{ij}]_{(i,j=\overline{1,n})} = \begin{cases} 0 & \leftarrow (i \neq j); \\ 1 & \leftarrow (i = j). \end{cases} \quad (7)$$

Затем ее строки обрабатываются с помощью векторных операций AND, OR, NOT, в соответствии с моделью анализа неисправностей, представленной в виде ДСПМН.

Пример 2. Дана вентильная структура цифрового устройства, изображенная на рис.2.

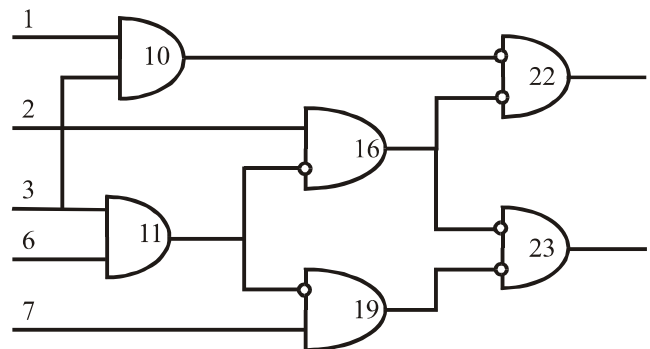


Рис.2. Структура цифрового устройства

Матрица $M = [M_{ij}]$, соответствующая приведенной выше схеме, имеет вид

M^0	1	2	3	6	7	10	11	16	19	22	23
1	1	0	0	0	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0	0	0
6	0	0	0	1	0	0	0	0	0	0	0
7	0	0	0	0	1	0	0	0	0	0	0
10	0	0	0	0	0	1	0	0	0	0	0
11	0	0	0	0	0	0	1	0	0	0	0
16	0	0	0	0	0	0	0	1	0	0	0
19	0	0	0	0	0	0	0	0	1	0	0
22	0	0	0	0	0	0	0	0	0	1	0
23	0	0	0	0	0	0	0	0	0	0	1

На входном наборе $T=1111110001$ схема исправного поведения трансформируется в ДФПМН, изображенную на рис.3.

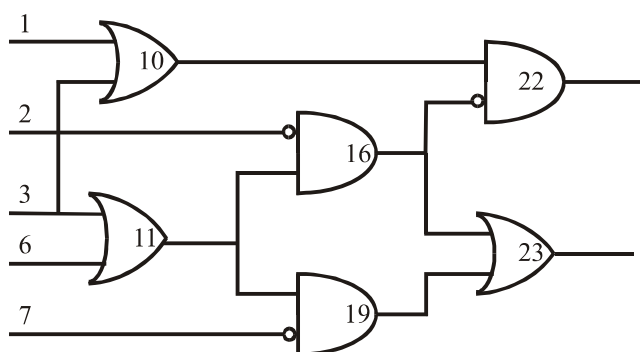


Рис.3. Дедуктивная модель к примеру 2

Результаты последовательной обработки логических элементов (см. рис.3) модифицируют исходную матрицу M^0 с помощью формулы

$$M_i = M_i \vee L_{ti},$$

где L_{ti} – дедуктивный элемент, формирующий состояние строки M_i на тесте T_t , приводя ее к следующему виду:

M^1	1	2	3	6	7	10	11	16	19	22	23
1	1	0	0	0	0	0	0	0	0	0	0
2	0	1	0	0	0	0	0	0	0	0	0
3	0	0	1	0	0	0	0	0	0	0	0
6	0	0	0	1	0	0	0	0	0	0	0
7	0	0	0	0	1	0	0	0	0	0	0
10	1	0	1	0	0	1	0	0	0	0	0
11	0	0	1	1	0	0	1	0	0	0	0
16	0	0	1	1	0	0	1	1	0	0	0
19	0	0	1	1	0	0	1	0	1	0	0
22	1	0	0	0	0	1	0	0	0	1	0
23	0	0	1	1	0	0	1	1	1	0	1

Упорядоченное множество строк матрицы делится на три подмножества (X, Z, Y) – входных, внутренних и выходных линий цифровой системы.

Строки, относящиеся к наблюдаемым выходам, формируют совместно с вектором исправного поведения $T=1111110001$ два вектора проверяемых дефектов (S^0, S^1), первый из которых идентифицирует единицами проверяемые константы нуля, второй – единицы:

N	1	2	3	6	7	10	11	16	19	22	23
22	1	0	0	0	0	1	0	0	0	1	0
23	0	0	1	1	0	0	1	1	1	0	1
S	1	0	1	1	0	1	1	1	1	1	1
T	1	1	1	1	1	0	0	1	1	1	0
S^0	1	0	1	1	0	0	0	1	1	1	0
S^1	0	0	0	0	0	1	1	0	0	0	1
D^0	1	0	1	1	0	0	0	1	1	1	0
D^1	0	0	0	0	0	1	1	0	0	0	1

Строка S определяется дизъюнкцией всех $M_i^1 (i \in Y)$, относящихся к наблюдаемым выходным линиям схемы:

$$S = \bigvee_{\forall i \in Y} M_i^1. \quad (8)$$

В данном случае $S = M_{23}^1 \vee M_{24}^1$. Функции для определения строк S^0, S^1 имеют следующий вид:

$$S^0 = S \wedge T, (S_j^0 = S_j \wedge T_j);$$

$$S^1 = S \wedge \bar{T}, (S_j^1 = S_j \wedge \bar{T}_j). \quad (9)$$

Вектор-строки D^0 и D^1 представляют проверенные на всех тестовых наборах неисправности, которые заполняются по мере обработки входных последовательностей, используя выражения

$$D^0 = D^0 \vee S^0, \quad D^1 = D^1 \vee S^1. \quad (10)$$

Естественно, что на первом тест-векторе наблюдается эквивалентность $D^0 = S^0, D^1 = S^1$. Тест T проверяет 100% всех одиночных константных неисправностей, если все координаты векторов D^0 и D^1 равны 1. В общем случае качество теста определяется выражением

$$Q(T) = \frac{1}{2^n} [\sum_{j=1}^n (D_j^0 + D_j^1)]. \quad (11)$$

Аналогично вычисляется процент покрытия неисправностей тест-вектором

$$Q(T_t) = \frac{1}{2^n} [\sum_{j=1}^n (S_j^0 + S_j^1)]. \quad (12)$$

Для повышения быстродействия метода моделирования цифровая система представляется двумя моделями $W = \{F, L_0\}$ [9]. Первая является компилятивной, включающей реализацию булевых уравнений в коде языка C++ и предназначена для исправного моделирования. Вторая – модифицированная интерпретация первой в табличном исполнении, что необходимо для быстрой модификации $L_0 \rightarrow L_1 \rightarrow \dots \rightarrow L_t \rightarrow \dots \rightarrow L_k$ в целях выполнения дедуктивно-параллельного алгоритма моделирования неисправностей цифровой системы.

При этом интерпретативная модель получается из компилятивной путем устранения всех инверсий из термов функционального описания F: $L_0 = \{F \setminus \neg\}$. Такая модификация осуществляется на основании следующего правила: все знаки инверсии над термами должны быть удалены. Обоснование данной модификации представлено в следующем утверждении.

Утверждение 1. Если цифровая система представлена в базисе элементарных функций И, ИЛИ, НЕ, то для любого двоичного тест-вектора исходная ДФПМН $L_0 \in W$ не имеет инверсий на выходах логических элементов.

Доказательство. Описание цифровой системы, включающее операции инверсии, может быть упрощено на основании следующих тождеств:

$$\begin{aligned} \overline{(a \oplus 0)} \oplus 1 &= a; \\ (a \oplus 1) \oplus 0 &= a. \end{aligned}$$

С другой стороны, всякая инверсия на выходе функционального элемента может рассматриваться как отдельный инвертор. При этом применение выражения (6) к данной функции от одной переменной доказывает несущественность операции инверсии при построении модифицированной интерпретативной модели цифровой системы, в которой исключены все инверсии из термов.

Утверждение 2. Тестовая модификация интерпретативной модели цифровой системы может иметь инверсии только на входных переменных элементарных функций.

Доказательство. Основано на рассмотрении всех возможных вариантов двоичных условий на входах логических элементов И, ИЛИ:

$$\begin{aligned} a \wedge b &= [(a \oplus 0) \wedge (b \oplus 0)] \oplus 0 = a \wedge b; \\ a \wedge b &= [(a \oplus 0) \wedge (b \oplus 1)] \oplus 0 = a \wedge \bar{b}; \\ a \wedge b &= [(a \oplus 1) \wedge (b \oplus 0)] \oplus 0 = \bar{a} \wedge b; \\ a \wedge b &= [(a \oplus 1) \wedge (b \oplus 1)] \oplus 1 = \bar{a} \wedge \bar{b}; \end{aligned}$$

$$\begin{aligned} a \vee b &= [(a \oplus 0) \vee (b \oplus 0)] \oplus 0 = a \vee b; \\ a \vee b &= [(a \oplus 0) \vee (b \oplus 1)] \oplus 1 = \bar{a} \vee b; \\ a \vee b &= [(a \oplus 1) \vee (b \oplus 0)] \oplus 1 = a \vee \bar{b}; \\ a \vee b &= [(a \oplus 1) \vee (b \oplus 1)] \oplus 1 = \bar{a} \vee \bar{b}. \end{aligned}$$

Другие элементы, имеющие инверсии, приводятся к упомянутым выше в соответствии с утверждением 1.

Следствия: 1. Инвертор в исправной схеме не влияет на транспортирование дефектов. 2. Дедуктивный терм не может быть составлен только из инверсных переменных. 3. Дизъюнктивный дедуктивный терм не имеет инверсных переменных.

3. Дедуктивно-параллельный алгоритм моделирования неисправностей

1. Формирование компилятивной и исходной интерпретативной моделей цифровой системы $W = \{F, L_0\}$. Определение начального значения тест-вектора $t=0$.

Инициализация векторов проверенных на тесте дефектов

$$\forall_{j=1}^n (D_j^0 = 0; D_j^1 = 0).$$

2. Определение номера очередного входного набора $t=t+1$ для $T_t \in T$. Если входных наборов нет ($t > k$) – конец моделирования.

3. Исправное моделирование [9, 10] всех примитивов (невходных линий) $F_i (i = \overline{1, n})$ цифровой схемы на входном наборе $T_t^X \in T_t$ с использованием компиля-

тивной модели $F \in W$ в целях доопределения невходных координат вектора $T_t^{\bar{X}} \in T_t$:

$$T_t^{\bar{X}} = f(T_t^X, F). \quad (13)$$

Идентичность векторов исправных состояний линий в двух соседних итерациях $T_t^r = T_t^{r-1}$ является условием перехода к следующему пункту.

Для моделирования последовательных схем и организации событийности [8] используется анализ пары соседних векторов (T_{t-1}, T_t) .

4. Инициализация матрицы проверяемых на тест-векторе дефектов $M = [M_{ij}]$ в соответствии с выражением (7).

Инициализация векторов проверяемых на тест-векторе дефектов $\forall_{j=1}^n (S_j^0 = 0; S_j^1 = 0)$.

Реконфигурация примитивов $L_i (i = \overline{1, n})$ интерпретативной модели $L_i \in W$ на основе применения формулы (6) для текущего вектора исправного состояния

$$T_t = (T_{t1}, T_{t2}, \dots, T_{tj}, \dots, T_{tn})$$

в целях получения модификации $L_{ti} = T_t \oplus F_i$.

5. Формирование невходных строк матрицы проверяемых неисправностей путем их параллельного моделирования [7] с помощью примитивов $L_{ti} \in L_t$.

6. Формирование объединенного вектора проверяемых неисправностей S путем применения форму-

лы (8). При выполнении условия $\forall_{j=1}^n (S_j = S_j^0 \vee S_j^1)$

осуществляется вычисление качества тестового набора по (12) и переход к следующему пункту, иначе – формирование пары $\{S^0, S^1\}$ по (9) или, если наблюдается исчезновение проверяемых неисправностей в векторе S по отношению к паре $\{S^0, S^1\}$:

$\exists_{j=1}^n [(S_j = 0) \& (S_j^0 \vee S_j^1 = 1)]$, выполняется исключе-

ние таких дефектов из процесса моделирования по правилу

$$(S_j^0 = S_j^1 = 0) \leftarrow \forall_{j=1}^n [(S_j = 0) \& (S_j^0 \vee S_j^1 = 1)].$$

Переход к п. 5.

7. Формирование векторов проверенных неисправностей в соответствии с выражением (10) и вычисление качества теста по формуле (11). Переход к п. 2.

Предложенный алгоритм ориентирован как на табличное описание примитивов произвольной сложности RTL уровня, так и на вентиляльное представление цифровых систем. Быстродействие алгоритма зависит от представления моделей цифровых устройств, которые могут быть реализованы в компилятивном и интерпретативном исполнении [8].

Преимущества дедуктивно-параллельного метода моделирования неисправностей:

1. Исключение операций над списками неисправностей, характерных для дедуктивного метода, имеющих вычислительную сложность n^2 . Переход к регистровым параллельным логическим операциям (по 64 или 128 битов) над строками матрицы (таблицы) проверяемых дефектов.
2. Уменьшение времени обработки упомянутой выше таблицы путем использования матрицы достижимостей, позволяющей не обрабатывать координаты, функционально не связанные с моделируемой линией.
3. Возможность реализации параллельных операций на процессоре PRUS, акселераторе HEFS, использование компилятора Active-HDL [11] (рис. 4) для получения тест-векторов исправного поведения, доопределенных по невходным координатам.

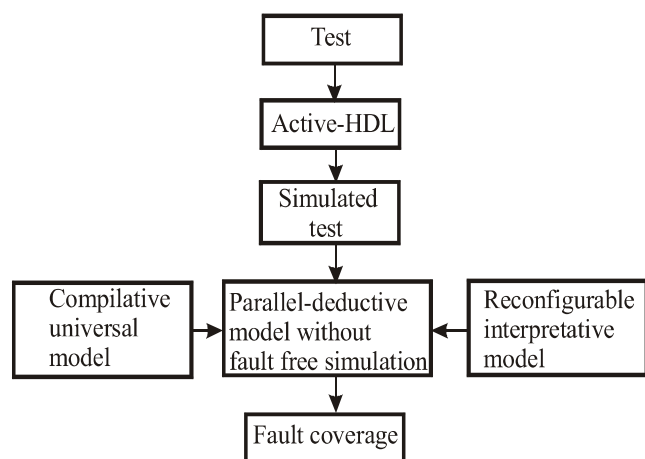


Рис. 4. Использование Active-HDL для моделирования тест-векторов

4. Событийность обработки матрицы проверяемых дефектов по принципу изменения хотя бы в одном разряде модельного регистра.
5. Реализация событийного моделирования исправного поведения на входном наборе при использовании компилятивной модели цифровой системы.
6. Интерактивная событийная модификация дедуктивной интерпретативной модели параллельного моделирования неисправностей на каждом входном наборе без инверторов на выходах логических элементов.
7. Возможность использования универсального элемента моделирования неисправностей для син-

теза схемы анализа дефектов цифровой системы или для построения компилятивной модели параллельно-дедуктивного моделирования неисправностей.

8. Возможность уменьшения размерности матрицы проверяемых дефектов путем сокращения множества неисправностей на основе их эквивалентирования [8] – неразличимости относительно наблюдаемых выходов.

4. Вычислительная сложность и сравнительный анализ с дедуктивным и параллельным методами

Учитывая, что разработанный метод моделирования неисправностей основывается на использовании преимуществ дедуктивного и параллельного алгоритмов [5-8], то естественным представляется выполнить сравнительный анализ всех трех реализаций.

Параллельный алгоритм имеет вычислительную сложность C_p , определяемую функциональной зависимостью от числа неэквивалентных неисправностей (b), длины компьютерного слова (W), количества эквивалентных вентилях (G):

$$C_p = (b^2 / W) \times G^3.$$

Дедуктивный алгоритм имеет некоторые отличия в формуле оценки быстродействия (система CHIEFS):

$$C_d = b^2 \times Q \times G^2 \Big|_{Q=G} = b^2 G^3,$$

где Q – среднее число активизированных неисправностями вентилях.

Предлагаемый дедуктивно-параллельный метод имеет быстродействие, определяемое выражением

$$C_{dp} = G^2 + (b^2 / W) \times G^2.$$

Первое слагаемое задает время исправного моделирования, второе – анализа неисправностей цифрового устройства, линии которого не ранжированы. Для комбинационной схемы с ранжированными линиями и элементами быстродействие метода будет иметь оценку

$$C_{dp}^r = G + (b^2 / W) \times G.$$

Быстродействие параллельно-дедуктивного метода выше параллельного и дедуктивного ($C_{dp}^r \ll \{C_p, C_d\}$), благодаря разделению фаз исправного и неисправного моделирования.

5. Особенности моделирования неисправностей в последовательностных схемах

Особенности анализа последовательностных схем связаны с возникновением генераторного режима – отсутствие условий проверки отдельных неисправностей на функциональных входных наборах.

Пример 3. Выполнить моделирование дефектов для триггерной схемы (рис. 5) на входном наборе 01.

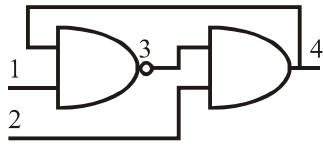


Рис. 5. Схема с обратными связями

Результаты исправного моделирования схемы определяются вектором 0111. Дедуктивно-параллельная схема для данного набора имеет вид, представленный на рис. 6.

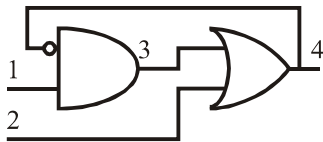


Рис. 6. Дедуктивная схема моделирования

Матрицы проверяемых неисправностей на трех итерациях иллюстрируют генераторный режим – появление и исчезновение дефекта 1¹ среди множества проверяемых:

M ¹	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	1	0	1	0
4	1	1	1	1

M ²	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	1	1	1

M ³	1	2	3	4
1	1	0	0	0
2	0	1	0	0
3	1	0	1	0
4	1	1	1	1

В этом случае необходимо корректировать списки проверяемых дефектов. В данном примере уже на второй итерации (матрица M²) следует исключить из рассмотрения неисправность 1¹. Тогда выполнение итерации I₃, представленное матрицей

M ³	1	2	3	4
1	0	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	1	1	1

дает окончательный список проверяемых неисправностей на тест-векторе 0111, равный L₄ = {2⁰, 3⁰, 4⁰}.

Здесь физическая причина исключения дефекта 1¹ заключается в иницировании им режима генерации изменяющихся сигналов при моделировании линий 3 и 4: 11-00-11-...-00. В этом случае состояния невходных линий принудительно определяются символами неопределенности X, что является условием отсутствия проверки дефекта 1¹.

6. Аппаратурная реализация дедуктивно-параллельного метода моделирования неисправностей

Учитывая, что параллельные процессы являются доминирующими и наиболее времяемкими в предлагаемом методе моделирования, представляется целесообразной его аппаратная реализация. Основная идея при этом

заключается в разделении устройства моделирования (рис. 7) на следующие компоненты:

1. Блок памяти для хранения матрицы $M = [M_{ij}]$ проверяемых дефектов, размерностью n^2 .
2. Регистр $S[1...n]$ для хранения признаков проверки (1), непроверки (0) неисправностей линий цифрового устройства, инверсных по отношению к состояниям линий T_{ti} тест-вектора T_t , формируемый по правилу, определенному в (8).
3. Регистровая пара ($S^0[1...n]$, $S^1[1...n]$) для хранения признаков проверки (1), непроверки (0) одиночных константных неисправностей ($\equiv 0, \equiv 1$) соответственно на тест-векторе T_t .
4. Регистровая пара ($D^0[1...n]$, $D^1[1...n]$) для хранения признаков проверки (1), непроверки (0) одиночных константных неисправностей ($\equiv 0, \equiv 1$) соответственно на тесте T .
5. Регистровая пара ($T_{t-1}[1...n]$, $T_t[1...n]$) для хранения двоичных состояний линий цифрового устройства при подаче на него двух соседних тест-векторов ($T_{t-1}T_t$).
6. Блок памяти для хранения схемного описания (BNF description), включающего: номер и тип элемента, число входов, номера (идентификаторы) входных переменных и выхода.

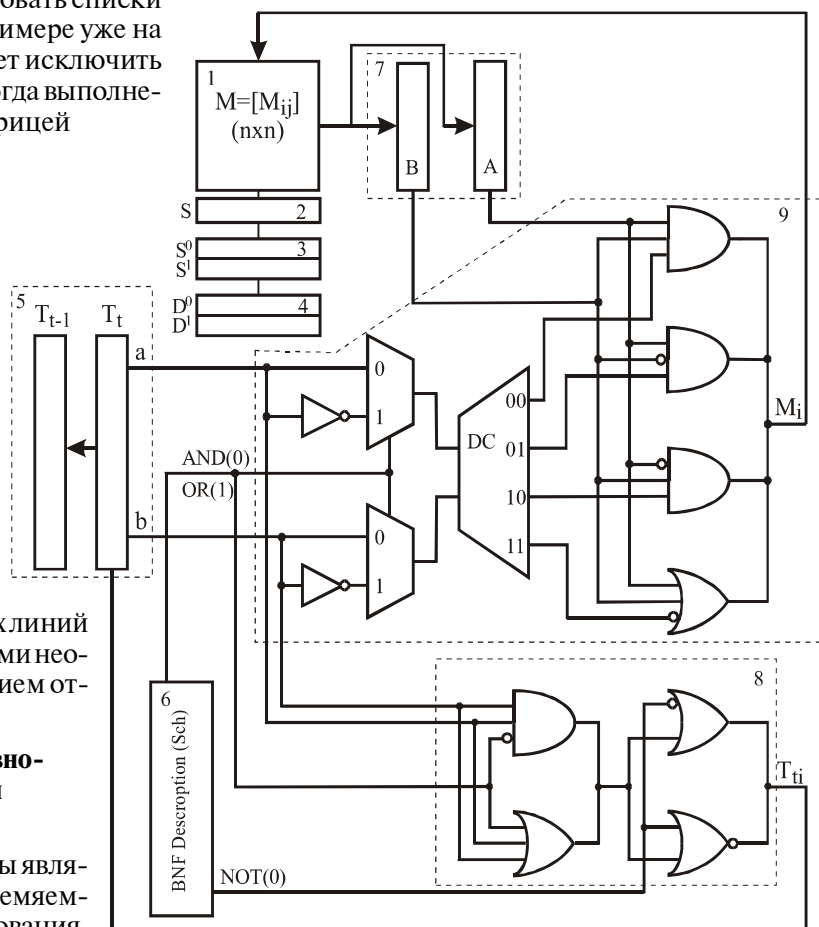


Рис. 7. Схема дедуктивно-параллельного моделирования неисправностей

7. Буферные регистры (А, В) для хранения операндов перед выполнением регистровых операций AND, OR, NOT над строками матрицы проверяемых дефектов $M = [M_{ij}]$.

8. Модуль вычисления исправных состояний не входных переменных $T_{ii} (i = \overline{1, n})$ цифрового устройства на тест-векторе T_t по значениям булевых переменных (а, b) бинарных логических операций.

9. Модуль векторных логических операций над регистровыми переменными (А, В), формирующий матрицу $M = [M_{ij}]$ на тест-векторе T_t .

Алгоритм работы устройства моделирования состоит из реализации процедур исправного моделирования и анализа дефектов [12]. Первоначально заносится информация – схемное описание в блок 6, формируется единичная матрица в блоке 1, обнуляются регистры блоков 2-5. Информация с блока 6 – тип элемента (И, ИЛИ) поступает на блок 9 в целях выбора операций для формирования векторов проверяемых дефектов совместно с тестовыми сигналами, поступающими на входы а, b блоков 8, 9 с блока 5, которые формируют сигналы на выходе блока 8, поступающие далее в блок 5, доопределяя не входные координаты. После выбранного дешифратором одного из четырех элементов на его входы подается содержимое регистровых переменных А и В, поступающее за два временных такта в блок 7. Результат векторной логической операции с выхода M_i в третьем такте поступает в блок памяти 1, где формируются векторы проверяемых дефектов для всех линий схемы. После обработки всех элементов, находящихся в блоке 6, выполняется формирование ячеек блока 2 и повторение процедуры в целях установления факта сходимости моделирования исправного поведения и неисправностей. После этого формируется содержимое ячеек блока 3 и 4. По окончании моделирования всех наборов теста в блоке 4 формируется вектор проверенных дефектов, на основании анализа которого определяется качество теста в процентах.

7. Заключение

Предложен быстродействующий метод дедуктивно-параллельного моделирования неисправностей на реконфигурируемых моделях цифровых устройств, представляющий сочетание достоинств дедуктивного и параллельного алгоритмов. Метод ориентирован на обработку цифровых устройств вентиляного и регистрового уровней описания. Рассмотрены вопросы, связанные с теоретическим обоснованием объединения методов дедуктивного и параллельного моделирования в целях повышения быстродействия анализа дефектов и определения качества тестов. Также рассмотрены особенности и достоинства алгоритмической и аппаратной, компилятивной и интерпретативной реализации предлагаемого метода, его вычислительная сложность и примеры моделирования тестовых схем. Технические характеристики программы мо-

делирования неисправностей, реализованной на языке Visual C++ для IBM PC (Pentium II, 500МГц): среднее быстродействие – 1000 векторов в секунду (в/с) для схем, содержащих 1 000 линий (2 000 вентиляей); 100 в/с при 3 000 линий (6 000 вентиляей); 30 в/с при 5 000 линий (10 000 вентиляей). Статистика обработки тест-примеров цифровых схем дедуктивным и дедуктивно-параллельным методами при моделировании 1000 входных последовательностей представлена в таблице. Графики сравнительного анализа реализации двух методов моделирования неисправностей цифровых устройств изображены на рис. 8, 9, 10.

Схема	n	Время, с		dt	Nt
		ДМ	ДПМ		
c17	17	1,372	0,010	1,362	137,200
c432	432	14,030	0,160	13,870	87,688
c499	499	18,206	0,240	17,966	75,858
c880	880	32,056	0,450	31,606	71,236
c1355	1355	46,037	0,751	45,286	61,301
c1908	1908	82,139	1,842	80,297	44,592
c2670	2670	134,566	6,409	128,157	20,996
c3540	3540	176,367	8,592	167,775	20,527
c5315	5315	285,356	18,787	266,569	15,189
c6288	6288	319,456	17,185	302,271	18,589
c7552	7552	502,082	44,004	458,078	11,410

Обозначения (таблица): n – количество вентиляей; ДМ – дедуктивный метод; ДПМ – дедуктивно-параллельный метод; $dt = t_{DM} - t_{DPM}$ – разность времен обработки двумя методами; $Nt = t_{DM} / t_{DPM}$ – отношение времен обработки двумя методами

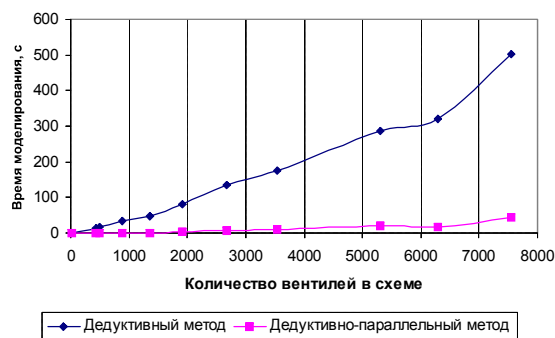


Рис. 8. Сравнительный анализ времени моделирования двух методов

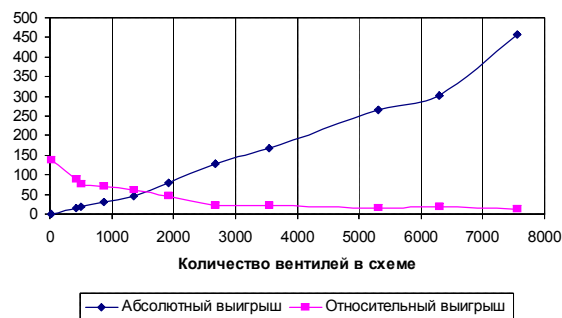


Рис. 9. Абсолютный и относительный $Nt = t_{DM} / t_{DPM}$ временной выигрыш дедуктивно-параллельного метода

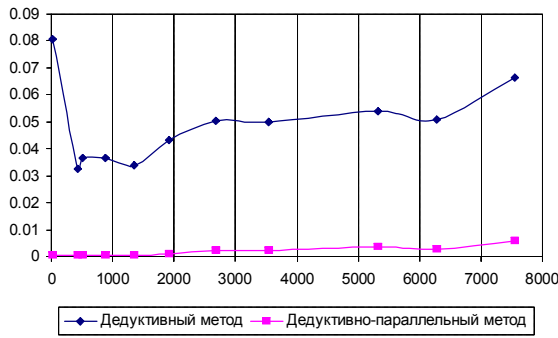


Рис. 10. Время обработки одного вентиля
($t_{\text{дм}}/n$), ($t_{\text{дпм}}/n$)

Литература: 1. *Hahanov V.I., Babich A.V., Huduke S.M.* Test Generation and Fault Simulation Methods on the Basis of Cubic Algebra for Digital Devices. Proceedings of the Euromicro Symposium on Digital Systems Design DSD2001. Warsaw, Poland. September, 4-6, 2001. P. 228-235. 2. *Хаханов В.И., Хак Х.М. Джахирул, Масуд М.Д. Мехеди.* Модели анализа неисправностей цифровых систем на основе FPGA, CPLD // *Технология и конструирование в электронной аппаратуре.* 2001. № 2. С. 3-11. 3. *Хаханов В.И.* Кубическое моделирование неисправностей и генерация тестов для цифровых систем. В кн.: Ежегодный отчет ХТУРЭ. 1999-2000. С. 139-146. 4. *Хаханов В.И., Сысенко И.Ю., Хак Х.М. Джахирул, Масуд М.Д. Мехеди.* Кубическое моделирование неисправностей цифровых проектов на основе FPGA, CPLD // *Радиоэлектроника, информатика, управление.* 2001. № 1. С. 123-129. 5. *Menon P.R., Chappel S.G.* Deductive fault simulation with functional blocks // *IEEE Trans. on Computers.* Vol. C.27. No 8. 1978. P. 689-695. 6. *Levendel Y.H., Menon P.R.* Comparison of fault simulation methods – Treatment of unknown signal values // *Journal of digital*

systems. Vol. 4. 1980. P. 443-459. 7. *Chang H.Y., Chappel S.G., Elmendorf C.H., Smidt L.D.* Comparison of parallel and deductive fault simulation Methods // *IEEE Trans. on Computers.* Vol. C23, No 11. 1974. P. 1132-1138. 8. *Abramovici M., Breuer M.A. and Friedman A.D.,* Digital System Testing and Testable Design, Computer Science Press, 1998. 652 p. 9. *Хаханов В.И.* Техническая диагностика элементов и узлов персональных компьютеров. К.: ИЗМН. 1997. 308с. 10. *Бондаренко М.Ф., Кривуля Г.Ф., Рябцев В.Г., Фрадков С.А., Хаханов В.И.* Проектирование и диагностика компьютерных систем и сетей. К.: НМЦ ВО. 2000. 306 с. 11. *Active-VHDL Series.* Book #1 - #4. Reference Guide. ALDEC Inc. 1998. 206 p. 12. *Кизуб В.А., Кривуля Г.Ф., Хаханов В.И., Тыдыков В.П.* Устройство для моделирования конечных автоматов/ А.с.№1520534 от 07.11.89. Бюллетень № 41. 14с.

Поступила в редколлегию 02.02.2002

Рецензент: д-р техн. наук, проф. Кривуля Г.Ф.

Хаханов Владимир Иванович, д-р техн. наук, профессор кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика вычислительных устройств, систем, сетей и программных продуктов. Увлечения: баскетбол, футбол, горные лыжи. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 40-93-26.

E-mail: hahanov@kture.kharkov.ua

Сысенко Ирина Юрьевна, аспирантка кафедры АПВТ ХНУРЭ. Научные интересы: моделирование цифровых систем. Увлечения: аэробика, музыка, иностранные языки. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 40-93-26.

Колесников Константин Васильевич, преподаватель кафедры КС Черкасского государственного технологического университета. Научные интересы: тестирование цифровых систем и сетей. Увлечения: живопись, поэзия. Адрес: Украина, 18006, Черкассы, бул. Шевченко, 460, тел. 43-74-28, e-mail: kvvk601@mail.ru

УДК 519.711

МАТЕМАТИЧЕСКОЕ ОПИСАНИЕ СЛОЖНЫХ СЕТЕВЫХ СТРУКТУР

ШАПОВАЛ И.Н., ХАЖМУРАДОВ М.А.

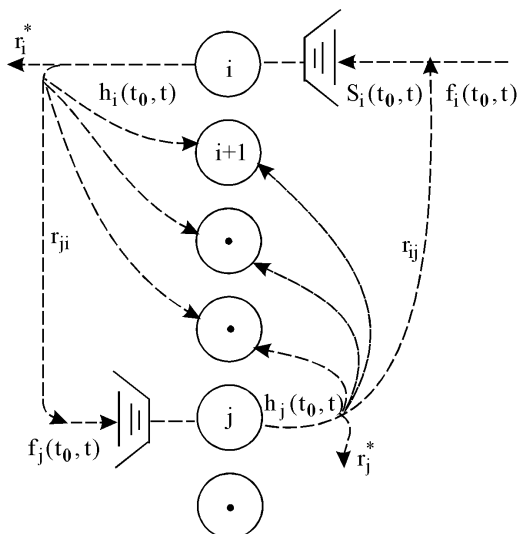
Приводится общее математическое описание сложных сетевых структур, не обладающих свойством декомпозиции к набору одноузловых систем очередей. Описание фактически реализует включение рассматриваемых сетей в класс нелинейных объектов с широко развиваемым в настоящее время инструментарием анализа и богатым множеством полученных нетривиальных свойств.

Аналитическое исследование модели сетевых структур (СС) обслуживания сопряжено со значительными трудностями [1]. Основной причиной является наличие более одного обслуживающего узла ($q > 1$) общего вида. Известный тезис Клейнрока [1] о независимости работы отдельных узлов, составляющих большую систему, позволяет редуцировать систему со сложной многоузловой структурой обслуживания к ряду независимых систем с одним узлом обслуживания. Однако этот тезис не является оправданным, когда требуется постоянство общего числа запросов, маршруты которых включают более одного обслуживающего узла.

Рассмотрим иной подход, приводящий к системе нелинейных интегральных уравнений. На основе РИ, 2002, № 1

этого результата многоузловая открытая система массового обслуживания (СМО) приводится к эффективно-разрешимой аналитической модели. Получаемые таким путем характеристики отличаются от результатов, базированных на тезисе Клейнрока, учетом интегральных свойств сети узлов.

Рассмотрим модель СС, включающую всю структуру обслуживания, как открытую СМО, но придадим ей более общий вид (рисунок).



Многоузловая открытая СМО