

следующий вид: $L_t = T_t \oplus F$. Если функциональное описание цифрового устройства представлено компонентами (примитивами), формирующими состояния всех линий схемы, то в качестве формулы преобразования исправной модели примитива F_i на тест-векторе T_t в дедуктивную функцию L_{ti} выступает выражение

$$L_{ti} = T_t \oplus F_i = f_{ti}[(X_{i1} \oplus T_{t1}), (X_{i2} \oplus T_{t2}), \dots, (X_{ij} \oplus T_{tj}), \dots, (X_{in_i} \oplus T_{tn_i})] \oplus T_{ti}, \quad (4)$$

которое является основой дедуктивного анализа цифровых проектов [3, 6].

В целях описания алгоритма анализа дефектов на основе выражения (4) вводятся следующие определения.

Определение 1. Вектор проверяемых на тест-векторе $T_t \in T$ нулевых (единичных) дефектов $S^0 = (S_1^0, \dots, S_i^0, \dots, S_n^0)$ ($S^1 = (S_1^1, \dots, S_i^1, \dots, S_n^1)$) есть упорядоченное в соответствии с нумерацией линий схемы множество одиночных константных неисправностей, где единичное значение координаты вектора $S_i^0 = 1$; ($S_i^1 = 1$) свидетельствует о проверке дефекта $\equiv 0 (\equiv 1)$ линии с номером i на текущем входном наборе $T_t \in T$ и непроверке этой неисправности в противном случае, когда $S_i^0 = 0$; ($S_i^1 = 0$).

Определение 2. Вектор проверенных на тесте T нулевых (единичных) дефектов

$$D^0 = (D_1^0, \dots, D_i^0, \dots, D_n^0) \quad (D^1 = (D_1^1, \dots, D_i^1, \dots, D_n^1))$$

есть упорядоченное в соответствии с нумерацией линий схемы множество одиночных константных неисправностей, где единичное значение координаты вектора $D_i^0 = 1$; ($D_i^1 = 1$) свидетельствует о проверке дефекта $\equiv 0 (\equiv 1)$ линии с номером i на хотя бы одном наборе $T_t \in T$ и непроверке этой неисправности в противном случае, когда $D_i^0 = 0$; ($D_i^1 = 0$).

Определение 3. Матрица проверяемых на тест-векторе $T_t \in T$ дефектов $M = [M_{ij}]$, размерностью $n \times n$, есть форма задания списка одиночных константных неисправностей, где ее координаты первоначально инициализируются в соответствии с выражением

$$[M_{ij}] \Big|_{(i,j=\overline{1,n})} = \begin{cases} 0 \leftarrow (i \neq j); \\ 1 \leftarrow (i = j). \end{cases} \quad (5)$$

В процессе моделирования тест-вектора нулевые координаты матрицы могут доопределяться единицами $M_{ij} = 1$, что соответствует проверке неисправностей, инверсных состояниям координат тест-вектора $T_{ti} \in T_t$.

Определение 4. Линия называется невходной \bar{X} , если она является выходной Y или внутренней Z , т. е. не относится к внешним входам X схемы. Невходная линия является выходной Y , если она соединена с выходным контактом схемы.

С учетом введенных определений практическая реализация выражения (4) оформляется в следующий алгоритм дедуктивно-параллельного анализа дефектов только линий сходящихся разветвлений.

1. Определение начального значения индекса обрабатываемого тест-вектора $t=0$.

Инициализация векторов проверенных дефектов:

$$\forall i (D_i^0 = 0; D_i^1 = 0).$$

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 всех примитивов $F_i (i = \overline{1,n})$ схемы на входном наборе $T_t^X \in T_t$ в целях доопределения невходных координат вектора $T_t^{\bar{X}} \in T_t$:

$$T_t^{\bar{X}} = f(T_t^X, F). \quad (6)$$

Идентичность вектора исправного моделирования линий в двух соседних итерациях $T_t^r = T_t^{r-1}$ является условием перехода к следующему пункту.

Примечание. Для моделирования последовательных схем и организации событийности используется анализ пары соседних векторов (T_{t-1}, T_t) . Примитив $F_i (i = \overline{1,n})$ моделируется, если на линиях выполняется условие $[T_{t-1}^X(F_i) \neq T_t^X(F_i)]$ – наличие изменений на входах рассматриваемого элемента.

4. Инициализация матрицы проверяемых на тест-векторе дефектов $M = [M_{ij}]$ в соответствии с выражением (5). Инициализация векторов проверяемых дефектов $\forall i (S_i^0 = 0; S_i^1 = 0)$. Реконфигурирование всех примитивов $F_i (i = \overline{1,n})$ на основе применения формулы (4) для текущего вектора исправного состояния $T_t \in T$ в целях получения дедуктивной схемы $L_t \leftarrow \forall i (L_{ti} = T_t \oplus F_i)$.

5. Параллельное моделирование неисправностей с помощью полученных дедуктивных функций $L_{ti} \in L_t$ путем выполнения регистровых операций над строками матрицы проверяемых дефектов M в целях доопределения координат, соответствующих невходным линиям схемы.

6. Формирование векторов проверяемых дефектов путем применения формул:

$$S^0 = (\bigvee_{\forall i \in Y} M_i) \wedge T_t; \quad S^1 = (\bigvee_{\forall i \in Y} M_i) \wedge \bar{T}_t \quad (7)$$

ко всем строкам матрицы, соответствующим выходным наблюдаемым линиям схемы.

7. При идентичности списков неисправностей в двух соседних итерациях $(S^0 \vee S^1)^{r-1} = (S^0 \vee S^1)^r$ (r – индекс итерации) определяется качество тест-вектора $T_t \in T$ по формуле

$$Q(T_t) = \frac{1}{2^n} \left[\sum_{i=1}^n (S_i^0 + S_i^1) \right] \quad (8)$$

и осуществляется переход к следующему пункту, иначе, если наблюдается исчезновение проверяемых неисправностей в итерации Γ по сравнению с $\Gamma-1$: $\exists i[(S_i^0 \vee S_i^1 = 1)^{\Gamma-1} \& (S_i^0 \vee S_i^1 = 0)^\Gamma]$, выполняется исключение таких дефектов из процесса моделирования по правилу

$$(S_i^0 = S_i^1 = 0) \leftarrow \forall i[(S_i^0 \vee S_i^1 = 1)^{\Gamma-1} \& (S_i^0 \vee S_i^1 = 0)^\Gamma]. \quad (9)$$

Переход к п. 5.

8. Формирование векторов проверенных дефектов в соответствии с выражением

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

и вычисление качества теста по формуле

$$Q(T) = \frac{1}{2^n} \left[\sum_{i=1}^n (D_i^0 + D_i^1) \right]. \quad (11)$$

Переход к п. 2.

Предложенная алгоритмическая реализация ориентирована как на табличное описание примитивов произвольной сложности RTL-уровня, так и на вентиляльное представление цифровых систем. Быстродействие алгоритма практически инвариантно компилятивным и интерпретативным моделям цифровых устройств, однако чисто интерпретативная реализация является более технологичной с позиции программирования.

3. Топологический ОДП-метод моделирования

Проведем модификацию процедуры суперпозиции в сторону обратного прослеживания дефектов на топологии схемы. Использованию процедуры обратной суперпозиции или обратного прослеживания в общем случае препятствует невозможность выполнения одномерной активизации, которая может быть проиллюстрирована двумя вариантами некорректности, представленными следующим примером.

Пример 1. Выполним анализ схем (рис. 1) в целях определения множества проверяемых дефектов на заданных тест-векторах с помощью процедуры обратной суперпозиции.

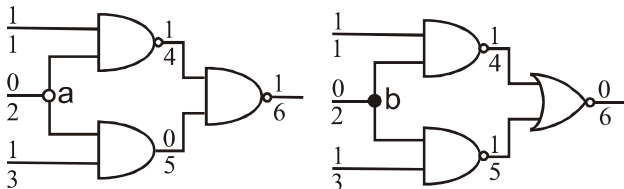


Рис. 1. Ложная проверка и не проверка дефектов

Для обеих схем активизация неисправностей на линиях a и b дает некорректные результаты их проверки.

В левой схеме константная неисправность 2^1 ложно проверяется на одномерном пути 2-5-6 с помощью процедуры обратного прослеживания. Однако при

этом не учитывается тот факт, что данный дефект изменяет состояние линии 4 с 1 на 0, что создает условия запрета транспортирования неисправности на выход схемы. Поэтому неисправность на линии $a - 2^1$ является ложнопроверяемой на наборе (101) при рассмотрении ее транспортирования по одномерному пути активизации.

Правая схема является примером альтернативной ситуации – к выходу 6 нет одномерного пути транспортирования дефектов с внешних входов и, в частности, от линии b . Тем не менее, неисправность на упомянутой линии – 2^1 проверяется, поскольку ее присутствие создает кратный дефект на линиях 4 и 5, который изменяет состояние выходной линии схемы. В обоих случаях имеется существенная некорректность, которая не позволяет использовать только идею одномерности при отслеживании пути транспортирования неисправностей от выходов ко входам схемы. Однако учитывая, что некорректность связана исключительно с одномерной активизацией неисправностей сходящихся разветвлений, необходимо сначала выполнить только их обработку, а затем исключить из рассмотрения, сделав структуру схемы древовидной и пригодной для одномерного обратного прослеживания. Естественно, для этого на стадии предварительного анализа (блок Р, рис. 2) следует выполнить дополнительную процедуру определения всех сходящихся разветвлений.

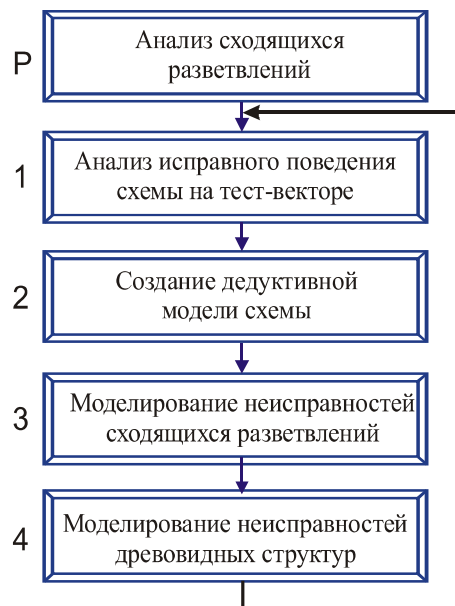


Рис. 2. Алгоритм топологического моделирования

Поскольку далее будет рассматриваться анализ неисправностей в привязке к топологии схемы цифрового устройства, то естественно представляется назвать алгоритм анализа топологическим. Таким образом, топологический с обратным прослеживанием дедуктивно-параллельный алгоритм (ТОДП) моделирования цифровых систем (см. рис. 2) можно представить шагами:

1. Анализ исправного поведения цифрового устройства на заданном входном наборе.

2. Преобразование схемы на текущем тест-векторе в дедуктивную модель.
3. Моделирование неисправностей СР сходящихся разветвлений по дедуктивной модели схемы.
4. Обратное прослеживание проверяемых неисправностей примитивов по ДС дедуктивной модели.

П. 3 ориентирован на обработку только сходящихся разветвлений, количество которых значительно меньше остальных линий; п. 4 — на моделирование неисправностей линий, относящихся к древовидным подграфам. Теоретическим обоснованием применения последнего пункта является доказательство следующих теорем и формулировка следствий из них.

Лемма. Сходящиеся разветвления $R_1^1 \in R^1$ в комбинационной схеме являются причиной появления кратных дефектов на входах примитивов.

Доказательство. Пусть комбинационная схема не имеет сходящихся разветвлений. В этом случае она представлена древовидной структурой. Тогда пересечение подграфов-предшественников (экстраобразов) для любых входов примитивного элемента равно пустому множеству

$$f^*(X_i) \cap f^*(X_j) = \emptyset. \quad (12)$$

Это справедливо для примитивов первого уровня, где входы элементов являются входами схемы, занумерованными различными идентификаторами линий. Если примитив находится внутри схемы, то в силу древовидности (12) любые два входа не будут иметь хотя бы одного общего предшественника. Иначе нарушится условие древовидности графовой структуры. Таким образом, два любых входа примитива древовидной схемы, не имея общих предшественников, не будут иметь и общих неисправностей на конкретном двоичном входном наборе, которые могут быть протранспортированы через рассматриваемый элемент.

Теорема 1. Для древовидной структуры дедуктивной схемы инверсный вход примитива И: $\bar{X}_i \in L$ запрещает транспортирование всех неисправностей, принадлежащих линиям-предшественникам.

Доказательство. Во-первых, инверсия на входе может иметь место на примитиве, имеющем две и более входные линии, во-вторых, она означает вычитание списка неисправностей, принадлежащего данному входу, учитывая изоморфизм $\bar{X}_i X_j \approx X_j \setminus X_i$. Но согласно лемме, пересечение двух списков линий-предшественников, относящихся ко входам одного примитива, равно пустому множеству (12). Следовательно, вычитание списка линий-предшественников никогда не уменьшит мощность любого другого списка, относящегося к неинверсному входу рассматриваемого элемента. Таким образом, для древовидных структур комбинационных схем всегда будет выполняться условие:

$$f^*(X_i) \setminus f^*(X_j) = f^*(X_i) \leftarrow f^*(X_i) \cap f^*(X_j) = \emptyset.$$

Теорема 2. Если в древовидной структуре дедуктивной схемы L существует примитив И, имеющий более одного неинверсного входа $X_i X_j$, то такой элемент запрещает транспортирование всех неисправностей, принадлежащих линиям-предшественникам всех его входов.

Доказательство. Наличие двух и более неинверсных, прямых входов в дедуктивном элементе означает пересечение списков линий, являющихся предшественниками для рассматриваемых входов. Но поскольку, согласно лемме, входы одного примитива ДС не имеют общих предшественников, то отсюда следует, что пересечение упомянутых списков будет всегда равно пустому множеству:

$$f^*(X_i) \cap f^*(X_j) = \emptyset \leftarrow X_i X_j.$$

Если же дедуктивный элемент имеет и входы с инверсией, то согласно теореме 1 вычитание из полученного пустого множества предшественников любого непустого, принадлежащего входу с инверсией, дает также пустой результат.

Следовательно, дедуктивный элемент И, имеющий более одного неинверсного входа, не пропустит через себя неисправности линий предшественников.

Следствия: 1) Инверсный вход дедуктивного элемента И запрещает активизацию всех неисправностей линий-предшественников, относящихся к данному входу. 2) Если дедуктивный примитив имеет более одного неинверсного входа, то все его входы следует определить инверсными. 3) Дедуктивный элемент И пропустит неисправности линий-предшественников только по неинверсному входу, который должен быть единственным. 4) Дедуктивный элемент ИЛИ не может иметь инверсных входов. 5) Сходящееся разветвление отмечается инверсией (кружком на линии ветвления), если его неисправность не проверяется на тест-векторе. 6) Инверсия на линии, входной или выходной, является условием разрыва активизации неисправностей и прекращения обратного прослеживания дефектов по рассматриваемой ветви древовидной структуры. 7) Проверяемую на тест-векторе линию сходящегося разветвления (на топологическом рисунке схемы далее отмечается жирным кружочком) при выполнении процедуры обратного прослеживания следует рассматривать как наблюдаемый выход схемы. 8) Интерпретация результата топологического моделирования: неисправности линий, не отмеченные знаками инверсий на дедуктивной модели схемы, проверяются.

В качестве иллюстрации основных шагов топологического моделирования ниже предлагаются следующие два примера.

Пример 2. Пусть дано цифровое устройство (рис. 3, первая схема, при рассмотрении сверху вниз), содержащее 3 сходящихся разветвления.

Устройство имеет 16 линий, в том числе 7 входов, на них подается тест-вектор 1011111, для которого

следует определить проверяемые дефекты константного типа. Состояния линий после исправного моделирования представлены в скобках. Результат процесса преобразования исправной модели устройства в дедуктивную и моделирование неисправностей линий СР зафиксирован на второй схеме. Здесь определено, что все неисправности разветвлений (2, 10, 13), инверсные по отношению к исправному состоянию этих линий, являются проверяемыми. Факт проверки отмечен на схеме дополнительно черными кружочками.

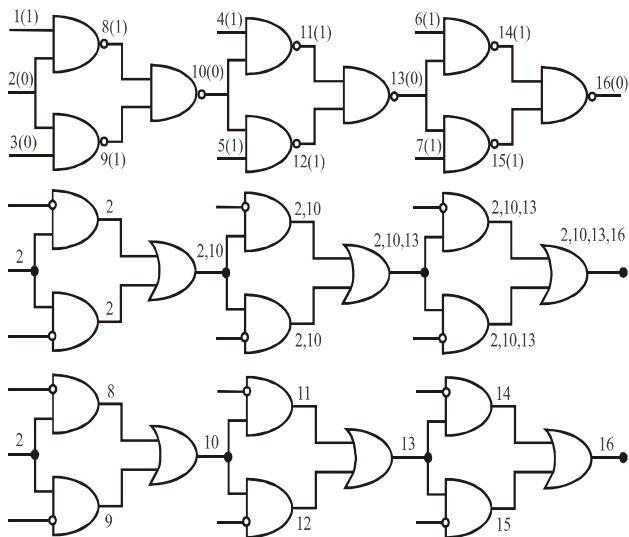


Рис. 3. Моделирование схемы с разветвлениями

Процедура обратного прослеживания неисправностей в целях определения списка проверяемых заключается в построении максимальных древовидных подграфов, ограниченных на топологии прозрачными кружочками. Черные кружочки есть проверяемые линии разветвлений, которые следует рассматривать на третьей схеме (см. рис. 3) как наблюдаемые выходы. Здесь же обозначены все неисправности, проверяемые на тест-векторе: {2, 8, 9, 10, 11, 12, 13, 14, 15, 16}. Знак дефекта определяется инверсией по отношению к состоянию исправного поведения линии. Для моделирования данного цифрового устройства истинным является утверждение: неисправности линий, не отмеченные прозрачными кружочками (знаками инверсии), проверяются.

Пример 3. Определить список проверяемых на тест-векторе 10111 неисправностей для цифрового устройства, представленного на рис. 4, содержащего 2 сходящихся разветвления.

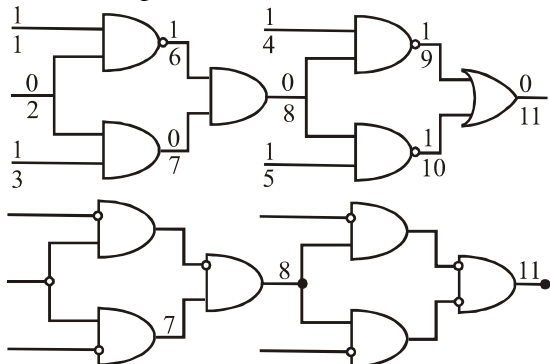


Рис. 4. Схема с двумя разветвлениями

Здесь интерес представляет непроверка неисправностей линий 9 и 10 наряду с фактом проверяемости линий 7 и 8, благодаря наличию сходящегося разветвления 8, моделирование неисправности которого показало, что она транспортируется на внешний выход 11.

Пример 4. Определить список проверяемых на тест-векторе 10111 неисправностей для цифрового устройства, представленного на рис. 5.

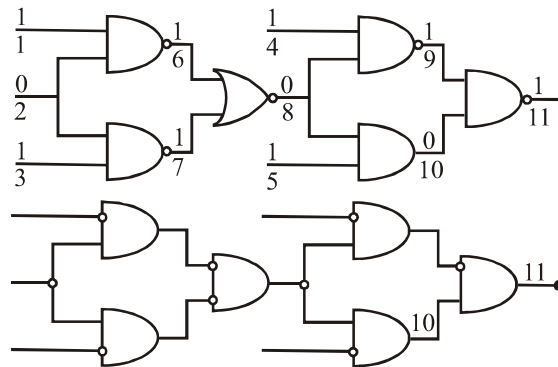


Рис. 5. Схема с непроверяемым разветвлением

Результат топологического моделирования показал, что непроверка сходящегося разветвления 8 запрещает транспортирование всех дефектов, которые относятся к линиям-предшественникам упомянутого разветвления. Поэтому проверяемыми зафиксированы неисправности только для линий 10 и 11. Иначе проверяются дефекты тех линий, которые не отмечены знаками инверсии.

На рис. 6 представлена структура эволюционного развития методов моделирования неисправностей. В основу положены дедуктивный (Д) и параллельный (П) алгоритмы. Далее был разработан универсальный, но относительно медленный, дедуктивно-параллельный метод (ДП). Затем был реализован ОДП-метод, ориентированный на быструю обработку моделей цифровых систем. Последний был модифицирован к ТОДП-методу, который ориентирован на вентильный уровень представления цифровых систем. При этом в описании устройства предварительно осуществляется поиск множества сходящихся разветвлений и выделение древовидных структур как дополнение к СР. Для моделирования сходящихся разветвлений используется универсальный дедуктивно-параллельный алгоритм, для анализа древовидных структур — ОДП- и ТОДП-методы. Такое разделение функций обработки цифровых схем большой размерности позволяет как минимум на порядок повысить быстродействие моделирования неисправностей по сравнению с базовыми методами (дедуктивный и параллельный).

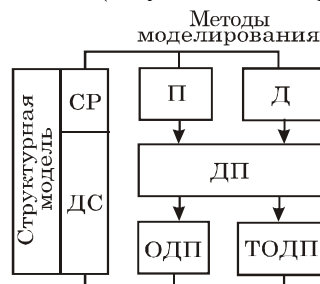


Рис. 6. Эволюция методов моделирования

4. Заключение

Предложенный метод моделирования неисправностей ориентирован на обработку цифровых устройств на основе ПЛИС, содержащих миллионы вентилей. Тестовые эксперименты программной реализации метода на сотнях цифровых комбинационных и последовательностных схем дали хорошие результаты по быстродействию по сравнению с традиционными алгоритмами параллельного и дедуктивного моделирования. Отдельные примеры анализа быстродействия разработанного метода (обработка тест-примеров на 1000 входных последовательностей, IBM PC 500 МГц, 256 Мбайт) и существующих базовых показаны на рис. 7. Ускорение моделирования составляет не менее десяти раз. На рис. 8 представлены результаты анализа быстродействия трех реализованных методов моделирования цифровых схем на одном и том же компьютере при обработке 1000 векторов. Показано преимущество ТОДП-метода перед (обратным) дедуктивно-параллельным. Выигрыш в быстродействии более существенен для схем большой размерности. Число сходящихся разветвлений в тест-схемах в среднем составляет 20% от общего количества линий.

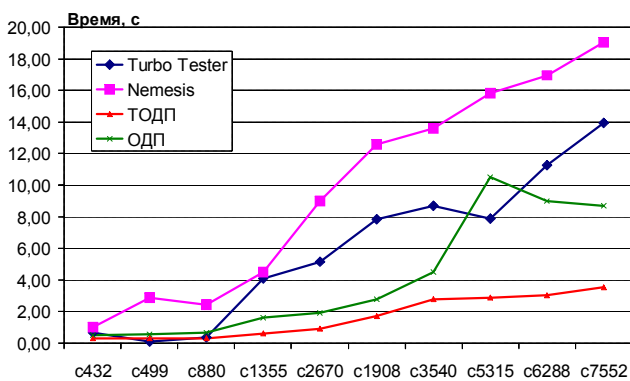


Рис. 7. Анализ быстродействия систем моделирования

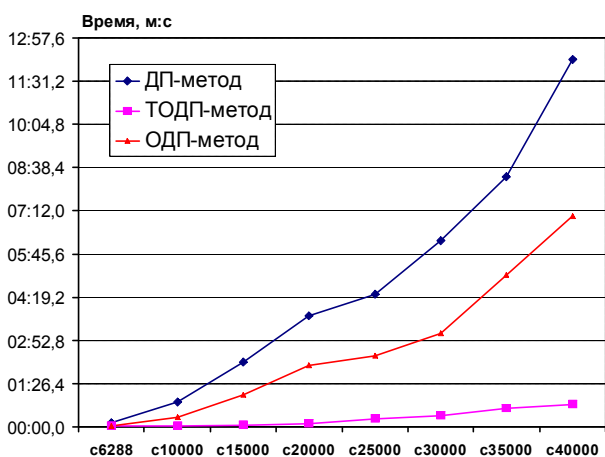


Рис. 8. Анализ быстродействия методов моделирования

Анализ дефектов топологическим методом требует линейных затрат памяти и времени в функции от числа линий и квадратичных затрат для обработки сходящихся разветвлений:

$Q = (r^2 / W) + 2n + n(1 - r/n)$, где (r^2 / W) — время моделирования неисправностей r сходящихся раз-

ветвлений; $2n = n_r + n_p$, $n_r = n$ — время реконфигурирования примитивов схемы на входном наборе; $n_p = n$ — время поиска подграфов линий, соответствующих непроверяемым сходящимся разветвлениям; $n(1 - 3r)$ — время выполнения процедуры суперпозиции на множестве линий схемы без сходящихся разветвлений и предшественников для непроверяемых СР.

Выигрыш в быстродействии предложенного метода тем больше, чем меньше сходящихся разветвлений в схеме цифрового устройства.

Таким образом, основным результатом данной работы является усовершенствование дедуктивно-параллельного метода моделирования неисправностей цифровых систем, заключающееся в:

- 1) создании обобщенной модели процесса дедуктивно-параллельного анализа цифровой схемы, ориентированного на обработку сходящихся разветвлений;
- 2) разработке топологического алгоритма моделирования неисправностей по древовидной структуре цифровой системы, имеющего линейную вычислительную сложность в зависимости от числа линий схемы.

Литература: 1. Wang X., Hill F.G., Mi Zh. A sequential circuit fault simulation by surrogate fault propagation // Proc. 1989 IEEE International test conference, IEEE Computer society, 1989. P. 9-18. 2. Nishida T., Miyamoto S., Kozawa T., Satoh K. RFSIM: Reduced fault simulator // IEEE Transactions on computer-aided design. 1987. Vol. CAD-6, No 3. P. 392-402. 3. Hahanov V.I., Babich A.V., Hyduke 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. 4. 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. 5. Levendel Y.H., Menon P.R. Comparison of fault simulation methods — Treatment of unknown signal values // Journal of digital systems. 1980. Vol. 4. P. 443-459. 6. Abramovici M., Breuer M.A. and Friedman A.D. Digital systems testing and testable design. Computer Science Press. 1998. 652 p. 7. Убар Р.П. Анализ диагностических тестов для комбинационных цифровых схем методом обратного прослеживания неисправностей // Автоматика и телемеханика. 1977. №8. С.168-176. 8. Автоматизированное проектирование цифровых устройств / С.С.Бадулин, Ю.М.Барнаулов и др. / Под ред. С.С. Бадулина. М.: Радио и связь. 1981. 240 с.

Поступила в редколлегию 11.12.2003

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

Хаханова Ирина Витальевна, канд. техн. наук, доцент кафедры АПВТ ХНУРЭ. Научные интересы: проектирование и диагностика цифровых устройств. Увлечения: английский язык, гитара. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: hahanov@kture.kharkov.ua

Чугуров Игорь Николаевич, соискатель кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых устройств. Увлечения: баскетбол, футбол, автопутешествия. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: hahanov@kture.kharkov.ua

Парфентий Александр Николаевич, аспирант кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых устройств. Увлечения: путешествия. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 70-21-326. E-mail: hahanov@kture.kharkov.ua