

УДК 681.3

## АЛГОРИТМЫ ПАРАЛЛЕЛЬНОГО ЛОГИЧЕСКОГО МОДЕЛИРОВАНИЯ И ПСЕВДОСЛУЧАЙНОЙ ГЕНЕРАЦИИ ТЕСТОВ ДЛЯ МОП-СТРУКТУР

© 1995 г. А. И. Андрюхин

*Институт прикладной математики и механики Украинской АН*

Поступила в редакцию 20.05.94 г.

Рассматривается метод псевдослучайного синтеза проверяющих тестов МОП-структур, основанный на параллельном моделировании на переключательном уровне определяемых пользователем классов неисправностей. Практическая реализация его, основанная на модифицированном параллельном алгоритме Бриана, использует видеосистему инструментального компьютера IBM PC.

### 1. ВВЕДЕНИЕ

На первоначальном этапе построения тестов для дискретных устройств целесообразно использовать псевдослучайные методы построения тестов, которые достаточно просто реализуются программно и для некоторых классов устройств дают приемлемые показатели и по полноте теста для рассматриваемого множества неисправностей, и по временным затратам.

В статье рассматривается метод псевдослучайного синтеза проверяющих тестов, использующий вместо известных методов логического моделирования на вентиляльном уровне параллельное моделирование на переключательном уровне определяемых пользователем классов неисправностей. Новой является и программная реализация метода, основанная на видеосистеме компьютера IBM PC вместо традиционной параллельной обработки разрядов машинного слова логическими командами процессора. Для дальнейшего опишем лежащее в основе метода псевдослучайной генерации теста моделирование устройств.

Цифровые устройства, создаваемые по МОП- и КМОП-технологиям, имеют особенности своего функционирования, которые трудно моделировать при их представлении моделями вентиляльного уровня, и моделирование неисправностей ясно показывает недостаточность традиционного вентиляльного подхода [1 - 3]. В [3] определены шесть недостатков классической теории переключательных схем в применении ее к анализу современных СБИС на МОП-структурах. Учитывают специфику работы МОП-структур моделированием на переключательном уровне [1 - 4], предлагая различные модификации алгоритма Бриана [5].

Логическое моделирование, выполняемое на компьютерах "фон-неймановской" архитектуры, — принципиально медленный процесс, так как параллельные изменения в схеме имитиру-

ются последовательно. Поэтому проблема повышения быстродействия при увеличении размеров исследуемых устройств является насущной. В настоящее время существуют следующие направления решения этой проблемы:

а) построение специализированных машин логического моделирования и соединения программных и аппаратных средств [6];

б) повышение уровня моделирования путем использования многоуровневого языка описания цифровых устройств (язык VHDL [7] и др.);

в) использование аппаратных особенностей инструментальных компьютеров в моделирующих программах.

Хорошо известен способ ускорения логического моделирования — параллельное моделирование, основа которого — способность компьютеров одновременно и независимо выполнять логические операции над разрядами машинного слова — обычно 2.4 или в случае ЕС ЭВМ над полем в 256 байт. В [8] предложено использовать структурные особенности списка машинных команд серий IBM 370 и совместимых с ними для ускорения моделирования определенного типа логических схем и приведены оценки производительности описанного способа моделирования.

Аналогично способу [8] предлагается выполнять моделирование на персональных компьютерах IBM, используя его аппаратные особенности, а конкретно его возможности работы с видеопамью.

В основе используемых методов параллельного моделирования — наличие аппаратно реализованных логических битовых операций над содержимым поля оперативной памяти и буфера видеопамью, результат которых заносится в буфер [9, 10]. В число этих операций входят операции логического умножения, сложения, отрицания, "исключающего ИЛИ". Для использования их достаточно понимания работы функций `putimage` и

getimage при программировании на языке Си [10]. Параметрами этих функций являются координаты прямоугольной области на дисплее и тип логической операции. Таким образом, возможно выполнять одним действием вывода на экран логическую операцию над полями памяти компьютера длиной десятки килобайт.

## 2. АЛГОРИТМЫ МОДЕЛИРОВАНИЯ ИСПРАВНЫХ МОП-СХЕМ

Отметим исходные данные алгоритмов. В первых, описание схемы на полевых транзисторах, различающихся большими и малыми сопротивлениями в узлах с большой и малой емкостью. Это описание для конкретности предполагается эквивалентным описанию на языке esim [4]. Во-вторых, задана последовательность входных сигналов. Результатом работы алгоритма являются значения сигналов на каждом узле устройства на данные входные воздействия.

Пусть  $\{Z, (C0, C1), CX, (SC0, SC1), SCX, (W0, W1), WX, (D0, D1), DX\}$  – простейший набор сигналов, необходимых для моделирования схем, выполненных по  $n$ -МОП-технологии и упорядоченных по отношению  $\leq$ . Мы применяли стандартные обозначения  $D, W, SC$  и  $C$  для описания сил (управляемых, слабых, сверхзарядных и зарядных) и обозначения  $0, 1, X$  для описания состояния узлов [4]. Таким образом, любой сигнал  $S = (H, G)$ , где  $H(G)$  – значение силы (состояния) соответственно. Особое место занимает сигнал  $Z = (Z_H, Z_G)$ , понимаемый как отключение узла.

Значение сигнала в узле  $v$  МОП-схемы можно определить как значение наиболее “сильного” сигнала из всех сигналов, пришедших в него по всем ациклическим путям, проходящим через открытые транзисторы из входных контактных узлов устройства и узлов, где изначально была емкость. Есть простой итерационный способ вычисления реакций узлов на входные воздействия этого достаточно сложного определения. Пусть значение сигнала на  $u$  ( $val(u)$ ) есть  $S_u = (H_u, G_u)$ . Для всех узлов  $v$  вычисляется новое значение сигнала на нем  $S_v^{(i)}$  согласно объединению предыдущего значения  $v$  и значений функций  $F_{uv}(val(u))$ , ассоциированных с транзистором между узлами  $u$  и  $v$  для всех соседей  $u$  узла  $v$  [4]. Вычисления продолжаются до тех пор, пока  $S_v^{(i+1)}$  не станет равным  $S_v^{(i)}$ . Обозначим  $F_{uv}(val(u))$  через  $f(T, R, H_u, G_u)$ , где  $u$  – узел, связанный непосредственно с  $v$  через транзистор типа  $T$ , который имеет значение состояния  $R$  на своем затворе. Функцию преобразования сигнала  $f$  при прохождении через транзистор рассмотрим ниже. Эти итерационные вычисления дают неадекватные результаты для некоторых сложных схем. Алгоритм Бриана модифицирует расчеты, основа которых есть эта вычислительная схема, служащая базой так называемого про-

стого алгоритма при использовании дистрибутивных транзисторных функций  $f$ . В целях уяснения предлагаемого алгоритма сначала рассмотрим простой алгоритм с дистрибутивными транзисторными функциями преобразования и далее параллельный алгоритм Бриана.

Напомним правила преобразования сигнала при прохождении через транзисторы [2, 4]. Будем рассматривать схемы, состоящие из  $n$ -МОП-,  $p$ -МОП- и нагрузочных транзисторов, соответствующие ключам с прямым и инверсным управлением и аттенюатору в терминологии СКА-теории [3]. Для кодирования типов этих элементов используем булев вектор  $T = (T_1, T_2)$ . Собственно  $n$ -МОП ( $p$ -МОП)-транзистор будем кодировать  $T_1 = 1, T_2 = 1$  ( $T_2 = 0$ ), а тип нагрузочного транзистора (заменяет резистор в схемах МОП-технологии) через  $T_1 = 0$ . Обозначим значение затвора на транзисторе через  $R$ . Первое правило указывает, что для закрытого транзистора его сток и исток не влияют друг на друга, т.е. отключены, и формально это правило можно записать как  $f(T, R, H, G) = Z$  при  $R = 0, T_1 = 1, T_2 = 1$  и  $f(T, R, H, G) = (H, G)$  при  $T_1 = 1, T_2 = 0$  и  $R = 0$ . Второе правило запишем как  $f(T, R, H, G) = (H, G)$  при  $T_1 = 1, T_2 = 1$  и  $R = 1$ , и  $f(T, R, H, G) = Z$  при  $R = 1, T_1 = 1, T_2 = 0$ . Третье правило определяет функционирование нагрузочного транзистора ( $T_1 = 0$ ). Запишем его при  $H = D$  в виде  $f(T, R, H, G) = (W, G)$  и  $f(T, G, H, G) = (H, G)$  в противном случае.

Обобщая правила преобразования сигналов через транзисторы [4], можем определить функцию  $f$  преобразования сигнала  $S = (H, G)$  через транзистор типа  $T$  со значением сигнала состояния  $R$  на затворе:

$$f(T, R, H, G) = \begin{cases} Z \text{ при } T_1 = 1, T_2 = 1 \text{ и } R = 0 \\ (H, G) \text{ при } T_1 = 1, T_2 = 1 \\ \text{и } R = 1 \\ Z \text{ при } T_1 = 1, T_2 = 0 \text{ и } R = 1 \\ (H, G) \text{ при } T_1 = 1, T_2 = 0 \\ \text{и } R = 0 \\ (H, G) \text{ при } T_1 = 0 \text{ и } H < D \\ (W, G) \text{ при } T_1 = 0 \text{ и } H = D. \end{cases}$$

Представим  $f$  совокупностью булевых функций, которые можно вычислять параллельно для простого алгоритма и алгоритма Бриана. Вид этих функций зависит от кодировки значений сил и состояний.

Так как основной объем оперативной памяти занимает описание связей между узлами, то экономия памяти от вида кодировки сигналов силы и состояния узлов незначительна. Поэтому с целью

повышения быстродействия моделирования прием следующее кодирование сигналов:

$$Z = (Z_h, Z_g), \text{ где } Z_h = (0, 0, 0, 0) \text{ и } Z_g = (0, 0, 0),$$

$$D = (1, 0, 0, 0), W = (0, 1, 0, 0), SC = (0, 0, 1, 0),$$

$$C = (0, 0, 0, 1), X = (1, 0, 0), 1 = (0, 1, 0) \text{ и } 0 = (0, 0, 1).$$

Пусть  $F = f(T, R, H, G)$ , и тогда для битовых компонент  $F = (FH_1, FH_2, FH_3, FH_4, FG_1, FG_2, FG_3)$  мы получим для выбранной кодировки следующие выражения, считая  $R = (R_1, R_2, R_3)$ ,  $H = (H_1, H_2, H_3, H_4)$  и  $G = (G_1, G_2, G_3)$ :

$$FG_1 = G_1 T_1 (\overline{R_2 \oplus T_2}) \vee G_1 \overline{T_1},$$

$$FG_2 = G_2 T_1 (\overline{R_2 \oplus T_2}) \vee G_2 \overline{T_1}, \quad (1)$$

$$FG_3 = G_3 T_1 (\overline{R_2 \oplus T_2}) \vee G_3 \overline{T_1},$$

$$FH_1 = T_1 H_1 (\overline{R_2 \oplus T_2}),$$

$$FH_2 = T_1 H_2 (\overline{R_2 \oplus T_2}) \vee \overline{T_1} H_2 \overline{H_1} \vee \overline{T_1} H_1, \quad (2)$$

$$FH_3 = T_1 H_3 (\overline{R_2 \oplus T_2}) \vee \overline{T_1} H_3 \overline{H_1},$$

$$FH_4 = T_1 H_4 (\overline{R_2 \oplus T_2}) \vee \overline{T_1} H_4 \overline{H_1},$$

где  $\oplus$  – операция “исключающее ИЛИ”.

При  $T_2 = 1(0)$  эти уравнения упрощаются и соответствуют схемам, содержащим только  $n$ -МОП ( $p$ -МОП)-транзисторы соответственно.

Эти симметричные относительно стока и истока формулы используются для простого алгоритма моделирования при использовании дистрибутивных транзисторных функций (см. алгоритм 9.2 [4]). Для параллельного алгоритма определим структуры данных  $Q_1, Q_2, Q_3, T = (T_1, T_2)$ , представляющие собой массивы длиной  $L = \sum k_i, i = 1, N$ . Здесь  $k_i$  – число соседей  $i$ -го узла и  $N$  – число узлов схемы.

В программной реализации считается, что узлы с номерами 1(2) имеют логическое значение 0(1) соответственно. Элемент  $Q_2(k)$  представляет собой номер узла, который является затвором транзистора, соединяющего узлы с номерами  $Q_1(k)$  и  $Q_3(k)$ . Полное представление об этих данных можно получить, рассмотрев пример схемы на рис. 1. Соответствующие ей структуры  $Q_1, Q_2, Q_3, T$  будут выглядеть следующим образом:

$$Q_1 = (1, 2, 3, 4, 4, 5, 5), \quad Q_2 = (2, 2, 2, 3, 4, 4, 5),$$

$$Q_3 = (1, 2, 3, 1, 2, 1, 2) \text{ и } T_1 = (1, 1, 1, 1, 0, 1, 0),$$

$$T_2 = (1, 1, 1, 1, 0, 1, 0).$$

Здесь узлы 1 и 2 – соответственно земля и питание, а узел 3 – внешний вход.

Обозначим через  $X$  поле значений узлов схемы, длина компонента которого равна  $[N/8] + 1$  байт. После окончания работы алгоритма в нем находятся установившиеся значения сигналов в

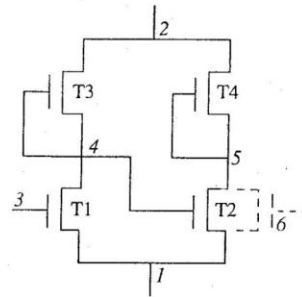


Рис. 1. Два МОП-инвертора.

узлах схемы. Предполагаются выделенными рабочие поля с компонентами длиной  $[L/8] + 1$  байт:

$S^0(S^1)$  – поля значений сигналов силы и состояния в узлах схемы, номера которых содержит массив  $Q_1$ ;

$R$  – поле значений сигналов состояния затворов транзисторов, номера узлов которых содержит массив  $Q_2$ ;

$(H, G)$  – поле значений сигналов сил (состояний) в узлах, соответствующих массиву связей  $Q_3$ ;

$F$  – рабочее поле значений преобразованных сигналов по функции  $f$ .

Напомним, что поля  $(X, S^0, S^1, F)$  состоят из 7 компонент.

При введенных обозначениях алгоритм запишем в следующем виде:

1. Если есть входные воздействия, занести их в  $X$ , иначе выход. (Обычно входные полюса устройства получают значения  $D0, D1$ , а значения внутренних узлов в начале моделирования –  $CX$ .)
  2. Занести значения из  $X$  в поле  $S^0$  согласно массиву  $Q_1$ .
  3. Занести значения из  $X$  в поля  $R, (H, G)$  согласно массивам  $Q_2$  и  $Q_3$  соответственно.
  4. Вычислить значения  $F$  согласно уравнениям (1) и (2).
  5. Определить в цикле для каждого узла  $v$  его новое значение путем выбора максимального из передаваемых значений множества соседей (вычисленных в поле  $F$ ) и своего предыдущего значения согласно решетке Хассе [3, 4] и занести это значение в  $S^1$  и  $X$ .
  6. Если  $S^0$  равно  $S^1$ , то записать решение  $X$  и перейти к п. 1.
  7. Поменять местами  $S^0$  и  $S^1$  (программно реализуется взаимной заменой соответствующих адресных указателей) и перейти к п. 3.
- Поясним, что в п. 4 происходит вычисление значений функций преобразования для узлов

Таблица 1. Исправная схема

Итерация	Имя узла	Значение			Узлов в полях			
1	$S^0$	D0	D1	D	CX	CX	CX	CX
	$R$	1	1	1	1	X	X	X
	$H, G$	D0	D1	D	D0	D1	D0	D1
2	$S^0$	D0	D1	D	D0	D0	W1	W1
	$R$	1	1	1	1	0	0	1
	$H, G$	D0	D1	D	D0	D1	D0	D0
	$F$	D0	D1	D	0	W1	Z	W1

всего устройства, и выполняется это с помощью функций работы с видеопамятью. В п. 5 происходит определение  $\text{lub}\{u_i\}$  для узла  $v$  с множеством соседей  $u_i$  [2, 4]. Понятно, что линии устройства с одинаковым номером  $v$  в упорядоченном массиве  $Q_1$  образуют множество соседей  $u_i$  для узла  $v$ . Определение наименьшей верхней границы по решетке Хассе ( $\text{lub}$ ) ввиду упорядоченности узлов в структуре  $S^0$  и соответствующим им сигналам в  $F$  представляет собой достаточно простую процедуру. Таким образом, определение значения в узле  $v$ , как наименьшей верхней границы ( $\text{lub}$ ) множества корневых неблокируемых путей, ведущих к вершине  $v$  в тричном переключательном графе [4, 5], сводится к итерационному процессу, имеющему простую программную реализацию.

Модифицированный алгоритм Бриана описан в следующем виде:

1. Установить для всех внутренних узлов значение силы  $C$  или  $SC$ .
2. Установить все сигналы состояния внутренних узлов в  $X$ .
3. Вычислить значение сил по (1), (2), считая, что транзистор с состоянием затвора  $X$  открыт (вместо  $R_2$  используем  $\bar{R}_3$ ).

Обозначим решение  $S^x = (G_1^x, G_2^x, G_3^x, H_1^x, H_2^x, H_3^x, H_4^x)$ .

4. Вычислить решение  $S^1 = (G_1^1, G_2^1, G_3^1, H_1^1, H_2^1, H_3^1, H_4^1)$  по (1), (2). Однако при определении  $\text{lub}$  учитываются только те  $f(T, R, H, G)$ , для которых переменная  $Y = Y(f)$  равна 1. Значение  $Y$  равно 0 в случаях:

- а) передачи состояния 1 через затвор с неопределенным значением  $X$ ;
- б) если сила сигнала  $S^1$  не равна силе  $S^x$ .

Условие а) записывается как  $\bar{R}_1 \vee \bar{G}_2$ . Условие б) можем записать как  $H_1^1 H_1^x \vee H_2^1 H_2^x \vee H_3^1 H_3^x \vee H_4^1 H_4^x$ , и поэтому  $Y$  равно произведению этих выра-

жений. Далее при определении  $\text{lub}$  используем  $uf(T, R, H, G)$ .

5. Этот шаг аналогичен 4, только теперь запрещен проход 0 через транзистор с затвором  $X$  и решение  $S^0 = (G_1^0, G_2^0, G_3^0, H_1^0, H_2^0, H_3^0, H_4^0)$ . Аналогичные п. 4 условия записываются соответственно как  $\bar{R}_1 \vee \bar{G}_3$  и как  $H_1^0 H_1^x \vee H_2^0 H_2^x \vee H_3^0 H_3^x \vee H_4^0 H_4^x$ .

6. Заменяем решения шага 3, меняя состояния по признаку изменения  $Y = G_1^0 G_2^1 \vee G_1^1 G_2^0 \vee G_1^0 G_2^1 \vee G_1^1 G_2^0$  и решение  $S = (G_1, G_2, G_3, H_1^x, H_2^x, H_3^x, H_4^x)$ , где  $G = YG^0 \vee \bar{Y}G_2 = YG^0, G_3 = YG^0$ .

Пример работы параллельного простого алгоритма представлен в табл. 1 для устройства на рис. 1.

### 3. МОДЕЛИРОВАНИЕ НЕИСПРАВНОСТЕЙ МОП-СХЕМ

Описанные параллельные алгоритмы позволяют естественным образом моделировать неисправности МОП-структур, которые при их представлении на вентиляльном уровне доставляют столько хлопот [1, 3]. Анализ причин дефектов схем, выполненных по МОП-технологии, показывает, что большая их часть может моделироваться неисправностями типа "устойчивый обрыв транзистора" (SOP) и "устойчивое замыкание транзистора" (SON) [11]. Первые переводят схему из класса комбинационных в класс последовательных, вторые обуславливают выходные неустойчивые сигналы.

Параллельное моделирование на переключательном уровне позволяет успешно использовать методологию [12, 13]. На рис. 2 представлены транзисторы, добавляемые в исправное устройство для моделирования различного типа неисправностей [12, 13]. Их затворы управляются дополнительными входами схемы, определяющими наличие неисправности. Массивы  $Q_1, Q_2, Q_3, T$  легко модифицируются для моделирования этих неисправностей. Так, короткое замыкание истоков на рис. 2г требует, во-первых, добавления информации о входе, управляющем наличием неисправности. Для этого в концы массивов  $Q_1, Q_2, Q_3, T_1, T_2$  заносятся соответственно числа  $N+1, 2, N+1, 1, 1$  (вход получает номер узла  $N+1$ ). Во-вторых, необходимо учитывать появление разветвлений для рассматриваемых стока и истока на добавочный транзистор неисправности. Это требует вставки в массивы  $Q_1, Q_2, Q_3, T$  справа или слева для любого элемента с номером  $k$ , того, что  $Q_1[k] = n(m)$ , следующих данных: для  $Q_1$  вставляется  $n(m)$ , для  $Q_2 - N+1 (N+1)$ , для  $Q_3 - m(n)$ , для  $T_1$  и  $T_2 - 1(1)$  соответственно. Число элементов в этих массивах увеличивается на 3 для одной неисправности этого типа. Аналогичные пре-

образования выполняются и для других неисправностей. Важно отметить, что учет логической силы для некоторых неисправностей при выборе максимального значения узла согласно решетке Хассе требует учитывать приоритет значения разветвления узла, управляемого дополнительным входом неисправности.

Этот метод реализован в двух вариантах:

1) моделируется только одна неисправность для больших устройств и компонента сигнала на узле занимает 1 бит;

2) одновременно моделируется  $M$  неисправностей, для чего необходимо выделять  $M$  бит для компонент сигналов. Понятно, что дополнительный вход, управляющий  $j$ -неисправностью ( $j = 1, M$ ), должен быть активным (установлен в 1) только в  $j$  бит.

Пример работы моделирования с неисправностью замыкания сток-исток, показанной для устройства на рис. 1 штриховой линией, представлен в табл. 2. Изменение структуры данных для схемы с этой неисправностью выразится в дополнении к концу массивов  $Q_1, Q_2, Q_3, T_1, T_2$  для исправного устройства двоек чисел (5, 6), (6, 2), (1, 6), (1, 1), (1, 1) соответственно. Здесь 6 – внешний вход активизации неисправности.

#### 4. МЕТОД ПСЕВДОСЛУЧАЙНОГО СИНТЕЗА ТЕСТОВ

Реализованный метод псевдослучайного синтеза тестов основан на параллельном моделировании неисправностей, которые по мере их обнаружения заменяются в группе моделируемых неисправностей на еще не обработавшиеся. Параметрами метода являются  $N_{max}$  ( $N_{min}$ ) – число наборов, на которых допускается (не допускается) модификация группы неисправностей при ее моделировании с целью обнаружения неисправностей. В сущности этот метод есть определение полноты случайного теста, выполняемого параллельным моделированием неисправностей, каждая из которых (если вошла в число обнаруживаемых) моделировалась на не более чем  $N_{min} + N_{max}$  наборах.

Основу параллельного моделирования неисправностей составляют описанные методы моделирования исправных схем. Определим класс неисправностей, для которых мы будем генерировать тест псевдослучайным образом. Число элементов рассматриваемого класса неисправностей исследуемого устройства обозначим  $E$ . Подчеркнем, что моделирование на переключательном уровне содержит информацию, которая позволяет уменьшить количество рассматриваемых замыканий [3].

Определение номера неисправности для каждого разветвления узла и типа неисправности однозначно определяется по  $Q_1[k], Q_2[k], Q_3[k]$ . Главное

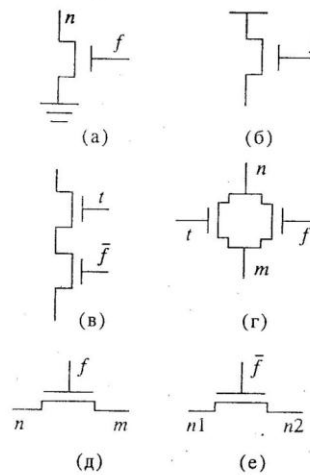


Рис. 2. Добавочные транзисторы для моделирования: 0-неисправностей (а); 1-неисправностей (б); обрыва транзистора (в); замыкания сток-исток (г); замыкания линий (д); разрыва линии (е).

отличие моделирования неисправностей от параллельного моделирования исправного устройства на входных наборах заключается в том, что значения входных полюсов всегда будут одинаковы для всей группы из  $M$  неисправностей, т.е. все  $M$  бит поля для каждого входа равны 1 или 0. Пусть  $N$  – номер набора случайного теста (при  $N=0$  инициализируем генератор псевдослучайных чисел).

Определим процедуру установки неисправности выбранного класса неисправностей как дополнение структуры данных, описывающих рассмотренное нами исправное устройство. Под процедурой удаления неисправности будем понимать обратное преобразование, результат которого – описание исправной схемы. Процедура замены

Таблица 2. Схема с неисправностью

Итерация	Имя поля	Значения узлов в полях
1	$S^0$	D0 D1 D1 CX CX CX CX CX D1
	R	1 1 1 1 X X X 1 1
	H, G	D0 D1 D1 D0 D1 D0 D1 D0 D1
	F	D0 D1 D1 D0 W1 Z W1 D0 D1
2	$S^0$	D0 D1 D1 D0 D0 D0 D0 D0 D1
	R	1 1 1 1 0 0 0 1 1
	H, G	D0 D1 D1 D0 D1 D0 D1 D0 D1
	F	D0 D1 D1 D0 W1 Z W1 D0 D1

неисправности состоит из удаления замещаемой неисправности и установки новой неисправности, которая может моделироваться на предмет ее обнаружения. Устанавливаемая неисправность получается путем замены неисправности  $l$  в пакете одновременно моделируемых неисправностей. При использовании этих процедур значение  $l$ -бит поля из  $M$  бит, отведенный под компонент многозначного сигнала, устанавливается в состояние  $X$ . Поэтому генерация тестов для замененной неисправности начинается с неопределенного состояния схемы.

Учитывая введенные обозначения, алгоритм генерации тестов опишем следующим образом:

1. Установить номер неисправности  $l = 1$ .
2. Если  $l$  равно  $E$ , то выход, иначе в цикле по  $l = 1, M$ , если  $l \leq E$ , установить  $l$ -ю неисправность и  $t = t + 1$ . Вычислить  $N = 0$ .
3. Если  $N > N_{\max} + N_{\min}$ , то перейти к 2, иначе генерировать новые значения внешних входов и установить  $F = 0$ .
4. Выполнить моделирование схемы с неисправностями на данном наборе.
5. В цикле по неисправностям  $l = 1, M$  выполнить п. 6.
6. В цикле по внешним выходам  $Y_s$  ( $s = 1, S$ ) установить  $F_l = 1$  и выйти из цикла, если для  $l$ -й неисправности значение состояния выходного полюса  $Y_s$  противоположно значению  $Y_s$  исправного устройства.
7. Если  $N - N_{\max} < 0$ , то в цикле по неисправностям ( $l = 1, M$ ), если  $F_l = 1$  и  $t < E$ , заменить обнаруженную неисправность следующей по списку и  $t = t + 1$ .
8. Вычислить  $N = N + 1$  и перейти к п. 3.

#### ЗАКЛЮЧЕНИЕ

Для схем, специально построенных для проверки возможностей предложенных алгоритмов, основанных на использовании вышеуказанных особенностей ПК (вентили заменялись на соответствующие транзисторные структуры), состоящих из 1179, 2979, 7107, 12771, 20938 транзисторов, время моделирования составляло для 10 входных наборов 10, 25, 60, 110, 170 с соответственно. Среднее число итераций равно 15. Моделирование производилось на IBM PC AT/386 с тактовой частотой 40 МГц. Полнота тестов, получаемых этим методом для вышеуказанных схем, была не ниже 90% для неисправностей SON и SOP. Достоинством описанного метода псевдослучайной генерации помимо программной простоты является возможность не решать сложную

задачу активизации путей для обнаруживаемых им неисправностей [14]. Добавим, что объем программной реализации описанных алгоритмов на языке Си оценивается в 900 операторов по сравнению с 5000 операторами системы ITEST [11]. Метод булевых представлений компонент многозначных сигналов расширяем для оценки динамических параметров исправных и неисправных МОП-структур посредством включения логической модели конденсатора и различных типов усилительных ключей.

#### СПИСОК ЛИТЕРАТУРЫ

1. Киносита К., Асада К., Карацу К. Логическое проектирование СБИС / Пер. с япон. М.: Мир, 1988. 309 с.
2. Randel E. Bryant. A Switch Level Model and Simulator for MOS Digital Systems // IEEE Trans. on Comp. 1984. V. 33. № 2. P. 160 - 177.
3. Хейес Дж.П. Обобщенная теория переключательных схем и ее применение для проектирования СБИС // Тр. Ин-та инж. по электротехнике и радиоэлектронике. 1982. Т. 70. № 10. С. 5 - 19.
4. Ульман Дж. Вычислительные аспекты СБИС / Пер. с англ. Под ред. Пархоменко П.П. М.: Радио и связь, 1990. 480 с.
5. Лобунов В.С. Логическое моделирование МОП-структур на переключательном уровне модифицированным методом Брайента // Электронное моделирование. 1991. № 2. С. 62 - 66.
6. Шмид А.В., Святский А.В. Автоматизация логического проектирования ЭВМ // Итоги науки и техники. Сер. Техническая кибернетика. М.: ВИНТИ, 1986. Т. 20. С. 136 - 204.
7. Армстронг Дж.Р. Моделирование цифровых систем на языке VHDL / Пер. с англ. М.: Мир, 1992. 175 с.
8. Андрюхин А.И. Об одном способе параллельного моделирования логических схем // Автоматика и вычислительная техника. 1990. № 6. С. 87 - 88.
9. Григорьев В.Л. Видеосистемы ПК фирмы IBM. М.: Радио и связь, 1993. 192 с.
10. Прокофьев Б.П., Сухарев Н.Н., Храмов Ю.Е. Графические средства Turbo C и Turbo C++ / Под ред. Генса Г.В., Храмова Ю.Е. М.: Финансы и статистика, 1992. 160 с.
11. Вейцман И.Н., Кондратьева О.М. Тестирование КМОП-схем // Автоматика и телемеханика. 1991. № 2. С. 3 - 34.
12. VLSI Testing / Ed. by Williams T.W. Elsevier Science Publishers B.V. 1986. 275 p.
13. Hayes J.P. Pseudo-Boolean Logic Circuits / IEEE Trans. 1986. V.C - 35. № 7. P. 602 - 612.
14. Люлькин А.Е. Построение тестовых наборов для МОП-схем // Микроэлектроника. 1993. № 5. С. 20 - 25.