

5. ДИАГНОСТИКА ВЫЧИСЛИТЕЛЬНОЙ ТЕХНИКИ

УДК 681.325.36

АЛГОРИТМ ПОИСКА ЦЕЛЕВЫХ СИГНАЛОВ В ГЕНЕРАТОРЕ ТЕСТОВ С ИСПОЛЬЗОВАНИЕМ МЕТОДА ВРЕМЕННЫХ ДИАГРАММ

Калоша Е.П., Грудовик К.В.

*Белорусский государственный университет
информатики и радиоэлектроники*

Генерация тестов для последовательностных схем является сложной проблемой[1]. Об этом свидетельствует широкое распространение различных методов сканирования, сводящих задачу тестирования последовательностных схем к тестированию комбинационных.

Существует два основных подхода к построению тестов для цифровых схем: по таблицам состояний; посредством представления схемы в виде сети логических элементов. При использовании первого подхода моделируется граф состояний, нет привязки к конкретной реализации схемы. Ко второму относятся *d*-алгоритм[2], PODEM[3] и другие, разработанные для комбинационных схем и обладающие свойством переборности, т.е. при их помощи можно найти тест для неисправности, если он существует. Кроме того, имеется возможность привязки к конкретной реализации схемы (на уровне вентилях).

Первоначально тестирование последовательностных схем решалось адаптацией уже разработанных алгоритмов для комбинационных схем[1]. Полученные посредством адаптированных алгоритмов тесты могут рассматриваться только как кандидаты для исходной схемы и требуют верификации посредством моделирования.

Предложенный в [1] генератор тестов для асинхронных последовательностных схем основан на алгоритме PODEM с использованием временных диаграмм для представления состояний. Алгоритм обладает следующими достоинствами:

- схема сразу рассматривается как асинхронная последовательностная, учитываются временные задержки, а следовательно и побочные эффекты (гонки, генерации), поэтому не требуется верификация полученного теста;
- переборность и эффективность, достигаемая за счет направленного поиска PODEM-алгоритма.

В процессе работы алгоритма PODEM выбираются целевые сигналы в узлах схемы, которые необходимы для обнаружения неисправности и транспортировки сигнала различия между исправным и неисправным состоянием на выход схемы. Достижение целевого сигнала состоит из двух попарно повторяющихся шагов: выбор сигнала на входе схемы по определенным эвристикам и моделирование схемы для нового выбора.

В предлагаемом переборном алгоритме для асинхронных схем состояние схемы описывается временной диаграммой. Временная диаграмма представляет собой совокупность пар (*время, сигнал*), описывающих порядок изменения сигнала на выходе соответствующего узла схемы [1].

В классическом PODEM-алгоритме тестируемая схема представляется в виде сети комбинационных элементов типа И, ИЛИ, И-НЕ и др. В предлагаемом генераторе алгоритм выбора целевого сигнала дополнен правилами для новых элементов (триггеров, тристабильных элементов и др.) и обработки колец обратной связи [1]. В отличие от комбинационных, новые элементы имеют специализированные входы (например, в элементе памяти могут быть входы информационный, синхронизации, сброса). Кроме того, в процессе перемещения от целевого узла ко входам схемы следует двигаться не только в пространстве (по сети элементов), но и во времени (по тестовым векторам).

5. Диагностика вычислительной техники

Для достижения высокой эффективности работы алгоритма при разработке новых правил предлагается использовать следующие эвристики. Для сокращения длины теста производится попытка установить в первую очередь те входы, которые приводят к установке желаемого сигнала на текущем тестовом наборе. Например, в триггерах с установочными входами будет в первую очередь осуществлена попытка получить сигнал на них. Для сокращения затрат на моделирование производится попытка установить в первую очередь управляющие входы, например, входы синхронизации для триггера.

В качестве примера применения этих эвристик на рис. 1 приведена программа выбора целевого сигнала для D-триггера, синхронизируемого фронтом, с асинхронным входом сброса.

```
void УстановитьТриггер(узел n, сигнал s, время t){
    if(R[t]=='X'){
        if(s=='H') УстановитьВход(R,'L',t);
        else{
            УстановитьВход(R,'H',t);
            if(ЭлементУстановлен(n,t) return;
            УстановитьВход(R,'L',t);
        }
    }
    if(УстановленФронт(n,t){
        if(D[t-1]=='X') УстановитьВход(D,s,t-1);
        if(D[t]=='X') УстановитьВход(D,s,t);
    }else if(C[t-1]=='H' || C[t]=='L') УстановитьТриггер(n,s,t-1);
    else УстановитьФронт(n,t);
}
```

Рис. 1. Алгоритм выбора целевого сигнала для D-триггера. Обозначения: R[t] - значение входного сигнала в момент t; C, R, D - соответственно входы синхронизации, сброса, информационный; 'H', 'L', 'X' - соответственно сигналы высокого, низкого уровня и неустановленный.

В связи с тем, что предложенный алгоритм обрабатывает не только комбинационные элементы, обладающие симметричными входами, но и триггеры, используется две различные процедуры установки целевого сигнала. Вызов первой происходит при установке сигнала логического 0 или 1 на элементе, на входах которого неисправность не проявляется. Другая используется при установке выхода элемента, находящегося на d -границе, когда на одном из его входов различаются временные диаграммы “исправной” и “неисправной” схемы. На рис. 2 приведен пример такой процедуры для D-триггера.

```
void УстановитьТриггерНаDГранице(узел n, время t){
    if(R[t]=='X') УставитьВход(R,'L',t);
    else if(C[t-1]=='H' || C[t]=='L') УстановитьТриггерНаDГранице(n,t+1);
    else if(УстановленФронт(n,t){
        if(СущественноеРазличие(R)) УстановитьВход(D,'H',t);
        else if(СущественноеРазличие(C)){
            if(D[t]=='X'){
                УстановитьВход(D,'H',t);
                if(ЭлементУстановлен(n,t)) return;
                УстановитьВход(D,'L',t);
            }else if(D[t-1]=='X') УстановитьВход(D,D[t],t);
            else УставитьТриггер(n,Inv(D[t]),t-1);
        }
    }else УстановитьФронт(n,t);
}
```

Рис. 2. Алгоритм установки D-триггера на d -границе.

В статье предложен генератор тестов для асинхронных последовательностных схем. Описан алгоритм выбора входных сигналов схемы. Ряд предложенных эвристик обеспечивает его высокую эффективность.

Литература

1. Е.П. Калоша, К.В. Грудовик. Генерация тестов для асинхронных последовательностных схем с использованием временных диаграмм //Автоматизация проектирования дискретных систем. Материалы II междунар. конф.- Мн.: Институт технической кибернетики НАН Беларуси, 1997, с. 88-95.
2. J.P. Roth. Diagnosis of Automata Failures: A Calculus and a Method //IEEE Trans. Comput., v C-15, № 7, July 1966, pp.278-291.
3. P. Goel. An Implicit Enumeration to Generate Tests for Combinational Logic Circuits //IEEE Trans. Comput., v C-30, № 3, Mar. 1981, pp. 215-222.

СРАВНИТЕЛЬНЫЙ АНАЛИЗ ДОСТОВЕРНОСТИ НЕКОТОРЫХ МЕТОДОВ КОМПАКТНОГО ТЕСТИРОВАНИЯ

Махнист Л. П.

Брестский политехнический институт

Под достоверностью методов компактного тестирования понимается эффективность обнаружения ошибочных двоичных последовательностей в потоке сжимаемых данных. Для оценки этой характеристики методов сжатия могут использоваться различные подходы и методы [1]. В данной работе предлагается сравнительный анализ достоверности некоторых методов компактного тестирования, имеющих общую специфику построения.

Рассмотрим два метода компактного тестирования, один из которых построен на основе композиции метода счета единиц сжатия двоичной по-