

УДК 519.713 : 681.326

М.А. КАМИНСКАЯ, С.А. ЗАЙЧЕНКО

Харьковский национальный университет радиоэлектроники, Украина

ПОВЫШЕНИЕ КАЧЕСТВА ТЕСТА НА ОСНОВЕ МЕТОДА АНАЛИЗА ТЕСТОПРИГОДНОСТИ УСТРОЙСТВА НА РАЗЛИЧНЫХ УРОВНЯХ ОПИСАНИЯ

Предлагается метод анализа тестопригодности цифрового устройства, представленного на системном уровне в виде кода VHDL и трансформированного в ориентированный граф для упрощения задач верификации, синтеза тестов и/или улучшения покрытия неисправностей для заданных входных наборов. Метод основан на топологическом анализе синтезированного устройства и последующей его модификации путем разделения режимов работы (тестовый и функциональный) в целях улучшения тестопригодности и упрощения решения задач тестирования.

верификация, тестопригодность, управляемость, наблюдаемость, путь сканирования, детерминированный тест, взвешенный тест

Введение

Тестопригодность – свойство изделия, направленное на соблюдение в установленных пределах материальных и временных затрат показателей, характеризующих приспособленность объекта к диагностированию, генерации тестов, моделированию дефектов за счет увеличения оборудования и дополнительных временных затрат на отдельных этапах проектирования. Методы проектирования тестопригодных схем: 1) анализ структурно-функционального исполнения объекта, численная оценка управляемости и наблюдаемости как меры тестопригодности схемы, которая может использоваться на этапе проектирования; 2) способы структурного проектирования тестопригодных схем, в пределе самотестируемых, основанных на использовании свойств сканируемого пути, обеспечивающего доступ к внутренним точкам схемы.

В настоящее время тестопригодность должна учитываться на самых ранних стадиях разработки при проектировании сложных цифровых систем наряду с энергопотреблением, надежностью, быстродействием, габаритами и стоимостью. Низкий уровень тестопригодности приводит к уменьшению процента выхода годных изделий на рынок, увели-

чению времени их верификации, тестирования и диагностирования на стадиях проектирования, производства и эксплуатации. Учитывая, что стоимость ошибки существенно возрастает по мере продвижения проекта к имплементации в кристалл ASIC, то необходимо иметь возможность ее диагностирования на ранних стадиях проектирования, в том числе и путем анализа тестопригодности в целях ее последующего учета для изменения структуры и введения аппаратурной избыточности.

Последний тезис послужил причиной разработки метода анализа тестопригодности на системном или алгоритмическом уровне (уровне регистровых передач), применимом также на вентиляльном уровне.

Цель работы: существенное уменьшение времени верификации, синтеза тестов и/или повышение степени покрытия неисправностей для заданных входных наборов путем модификации структуры цифрового автомата на основе анализа его тестопригодности.

Объект исследования – цифровая схема, представленная в виде соединенных между собой RTL элементов или логических вентилей на вентиляльном уровне. Для достижения поставленной цели необходимо решить следующие задачи: 1. Описание моде-

ли цифрового устройства на языке VHDL (или с помощью других языков описания аппаратуры); 2. Представление устройства в послесинтезном виде; 3. Оценка тестопригодности устройства, представленного в виде соединенных RTL элементов; 4. Оценка тестопригодности устройства на вентиляном уровне; 5. Разработка алгоритма модификации устройства для возможности разделения режимов тестирования и нормального функционирования; 6. Верификация и тестирование метода анализа тестопригодности на реальных цифровых схемах.

Развитие методов проектирования цифровых устройств

Современные технологии проектирования цифровых устройств позволяют размещать на одном кристалле до нескольких миллионов вентилях, при этом размер кристалла может составлять 1 – 2 квадратных сантиметра. Плотность интеграции элементов на кристалле увеличивается с каждым годом для элементов памяти в 1,5 раза и микропроцессоров в 1,35 раза. Среднее количество вентилях в устройстве в 1991 году составляло 25000 вентилях, при среднем времени проектирования 8,1 месяца; в 1993 среднее количество вентилях на кристалле составляло 50000 при среднем времени проектирования устройства 5,5 месяцев. Кристаллы ASIC, изготавливаемые фирмой Xilinx могут насчитывать порядка 10 миллионов логических вентилях. Такая плотность интеграции не позволяет адекватно проводить процедуру тестирования общепринятыми методами.

При реализации проекта разработчик сначала создает описание проекта на языках описания аппаратуры VHDL, Verilog, затем описание преобразуется в блочную структуру с помощью средств логического синтеза, например, Synplify, Synplify Pro. После этих двух этапов послесинтезная модель раскладывается до вентиляного уровня, затем до транзисторного, после чего происходит непосредственная имплементация в кристалл. Следует отметить,

что на каждом уровне представления цифрового устройства необходимо проводить верификацию и тестирование устройства для уменьшения вероятности появления неисправностей на этапе имплементации. Каждый иерархический уровень имеет свое название и свой базовый набор структурных примитивов.

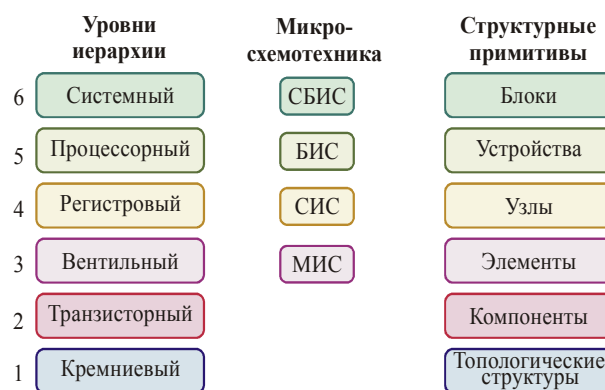


Рис. 1. Уровни иерархии

На более высоком уровне описания расширяются возможности проектировщика, увеличивается продуктивность процедуры тестирования, и улучшается процесс передачи информации между соединенными блоками устройства. Важным моментом здесь является развитие и разработка новых технологий на основе спецификаций, а затем и разработка новых стандартов, которые открывают новые, более быстрые пути поступления изделия на мировой рынок. Например, разработка языков регистровых передач повлекла за собой возникновение средств синтеза цифровых устройств. Обеспечение корректности функционирования устройства на уровне регистровых передач является одной из самых больших трудностей для изготовителей современных систем на кристалле и заказных интегральных схем (ASIC). Главной целью разработчиков является сокращение времени верификации устройства, что требует разработки новых технологий проектирования и верификации.

В настоящее время все более популярным становится альтернативный подход к тестированию (тестирование White-box) представляет собой добавле-

ние ассерций (или дополнительной сканирующей логики) в структуру кода (устройства), которые позволяют наблюдать и управлять «внутренностями» верифицируемого устройства (поведением внутренних блоков устройства и их межсоединений); как результат, повышается управляемость и наблюдаемость поведения верифицируемого устройства.

Кроме несинтезируемых ассерций, которые выбираются на основе спецификации, существуют также различные AD-HOC технологии, то есть, введение в устройство дополнительной синтезируемой логики, не влияющей на работу устройства в режиме нормального функционирования, но обеспечивающую наблюдаемость внутренних «проблемных» частей (блоков) устройства или межсоединений (узких мест) в режиме сканирования. Такой подход хотя и требует дополнительных аппаратурных затрат, но позволяет существенно повысить процент покрытия неисправностей, а значит и качество проектируемого устройства и ускорить его выход на рынок. Введение дополнительной сканирующей логики часто используется на вентиляльном уровне и на уровне регистровых передач, на основе применения методов анализа тестопригодности на вентиляльном уровне.

Чтобы понять необходимость и предназначение введения дополнительной сканирующей логики в устройство, необходимо ввести понятия наблюдаемости, управляемости и тестопригодности. *Наблюдаемость* – способность наблюдать снимаемые с выходов устройства реакции на входные воздействия (тесты), которые генерируются с помощью тестовой программы testbench, то есть возможность наблюдать результат воздействия на входные (и внутренние) линии кода или структуры. В этом случае testbench можно рассматривать как средство для “ограниченной” наблюдаемости, то есть, наблюдения только внешних портов устройства или модели. *Управляемость* – возможность стимулировать (активизировать) определенную линию кода или струк-

туры внутри проекта. В рамках управляемости можно рассматривать такие параметры как: покрытие линий кода, покрытие разветвлений кода, покрытие путей, покрытие выражений.

В работе рассматривается понятие управляемости и наблюдаемости внешних и внутренних узлов синтезированного программного кода и на основе полученных показателей управляемости и наблюдаемости предполагается введение в структуру схемы управляющей логики для труднодостижимых узлов. Выбранные контрольные точки с наихудшими показателями управляемости и/или наблюдаемости должны указывать на места в схеме, которые требуют дополнительной проверки. Такую процедуру может проводить любой человек, даже не участвующий изначально в проекте. В этом случае инженеру или разработчику необходимо только внести в разрабатываемый код дополнительную логику (в места, выбранные по предложенному методу). В случае же использования ассерций нет четкого алгоритма их использования, только разработчик спецификации и кода может определить, в какой именно части кода необходима ассерция. Новому программисту, который может сменить своего предшественника, придется заново разбираться в коде и самому «прочувствовать», где именно должна быть внедрена ассерция. Именно это и послужило поводом для разработки метода анализа тестопригодности и целенаправленного выбора узких мест в устройстве для введения контрольных точек.

Методы вычисления показателей тестопригодности

Методы анализа тестопригодности разрабатывались для различных уровней описания цифрового изделия, в зависимости от тенденций развития средств проектирования. Наиболее известные методы анализа тестопригодности на вентиляльном уровне – это CAMELOT, метод Питерсона, SCOAP, TADATPG [1,2]. Они ориентированы на структур-

ный анализ разрабатываемого устройства и использование детерминированного теста. Так, в методе Питерсона вычисление показателей тестопригодности производится на схеме, представленной в виде ориентированного графа. Позднее был разработан метод Паркера-МакКласки [3], и на его основе разрабатывались другие подходы к анализу тестопригодности на вентиляном уровне и уровне регистровых передач [4]. В настоящее время актуальным для рынка EDA является анализ на более высоких уровнях описания устройства – системном или регистровых передач. Так, в [5] была предложена технология повышения тестопригодности устройства, представленного на поведенческом уровне. Алгоритм анализа тестопригодности основан на модификации исходного программного кода, в который вносится избыточность в виде дополнительного условия, позволяющего сделать заданную строку кода наблюдаемой. Использование такого подхода позволяет повысить на 50% процентов покрытие неисправностей при несущественных аппаратурных издержках. В [6] представлена технология анализа тестопригодности на поведенческом уровне описания. Здесь учитывается область существования переменной, эксплуатационная тестопригодность, достижимость на графе взамен управляемости и наблюдаемости линий схемы на вентиляном уровне описания. Технология позволяет достаточно легко вычислять показатели тестопригодности при невысокой вычислительной сложности.

Экспериментальные результаты показали, что такой подход позволяет эффективно оценить трудно тестируемые части устройства и может быть использован при внедрении технологий сканирования. В [7] представлен метод анализа тестопригодности, который выполняется в процессе синтеза цифровой схемы на системном уровне. В методе вычисляется вероятность распространения тестовых данных через логический элемент по каждому пути в схеме. Полученные показатели тестопригодности коррели-

руют с показателями в методе SCOAP. Но поскольку в методе используются относительные значения управляемости и наблюдаемости, то оценка покрытия неисправностей может быть пессимистической. Позднее был представлен метод анализа тестопригодности на системном уровне, который основан на взаимодействии конструкций HDL языка и их имплементации в структуру схемы. Такой подход также может быть использован для оценки тестопригодности на схеме после выполнения процедуры синтеза. Результаты, полученные при досинтезном анализе, могут быть использованы для модификации высокоуровневого кода для избежания трудностей при тестировании изделия.

Ниже предлагается метод анализа тестопригодности на системном уровне (уровне регистровых передач). Метод позволяет проводить простой анализ структуры схемы и производить ее модификацию для повышения тестопригодности устройства. Метод применим как на уровне регистровых передач, так и на вентиляном уровне.

Структурный анализ схемы

Анализ тестопригодности разрабатываемой модели системы необходимо проводить на всех стадиях проектирования. При этом самый адекватный анализ соответствует наиболее точной модели, которая определяется вентиляным уровнем описания, поскольку структура устройства здесь представлена максимально детализировано. Тем не менее, анализ тестопригодности на более высоких уровнях описания, где модель проекта отражает лишь структуру взаимосвязанных компонентов, имеет место быть, поскольку здесь трудоемкость процедуры анализа минимальна, но оценки тестопригодности и последующая модификация проекта на основе технологий граничного сканирования могут существенно повлиять на стоимость диагностического обеспечения и обслуживания (временные и материальные затраты на синтез тестов, моделирование неисправностей

и диагностирование дефектов для каждой стадии проектирования).

Предлагаемый метод заключается в вычислении значений управляемости и наблюдаемости узлов схемы, формирующих оценку тестопригодности. Метод может быть использован как на досинтезной структуре цифрового устройства (устройство, описанное на ЯОА), так и на более детализированных уровнях представления – вентильном уровне и уровне регистровых передач. Подход вычисления тестопригодности основан на вероятностном подходе вычисления показателей тестопригодности узлов устройства (узлы – эквивалентные линии схемы, представленной на вентильном уровне; межсоединения между блоками устройства на уровне регистровых передач, и сигнальные линии, соединяющие операторы кода VHDL).

Значения управляемости на первичных входах равны 0,5 (так как первичные входы устройства или входные порты устройства могут быть управляемы как нулем, так и единицей). Значения управляемости вычисляются от первичных входов схемы до первичных выходов и изменяются в диапазоне от 1 до 0. Ноль имеет линия, которую невозможно установить ни в одно из логических значений. «Переломным» значением является 0,5. Если к примеру управляемость нуля C^0 меньше 0,5, то такую линию трудно установить в логический ноль. Значения управляемости зависят от значений управляемости на линиях схемы, принадлежащих предыдущему рангу схемы ($r - 1$) и от коэффициента распространения логического значения ($K(0)$, $K(1)$) сквозь блок схемы от его входов к выходам. Коэффициент распространения вычисляется на основе кубического покрытия RTL блока или логического вентиля. Однако, когда устройство описывается с помощью кубического покрытия, возникает трудность в моделировании символа 'X'. Чтобы избежать эту проблему, предлагается использовать процедуру доопределения символа 'X'. Значения управляемости могут

быть вычислены по следующим формулам:

$$C^0(X_{i+1}) = K(0) \cdot \prod_{i=1}^m C^0(X_i);$$

$$C^1(X_{i+1}) = K(1) \cdot \prod_{i=1}^m C^1(X_i),$$

где m – количество входов в логическом блоке. В случае, когда в схеме присутствуют сходящиеся разветвления (рис. 2), формулы вычисления управляемости сводятся к следующему виду:

$$C^0(X_{i+1}) = K(0) \cdot \prod_{i=1}^m C^0(X_i) \cdot 2^{k+1};$$

$$C^1(X_{i+1}) = K(1) \cdot \prod_{i=1}^m C^1(X_i) \cdot 2^{k+1},$$

где k – количество сходящихся разветвлений

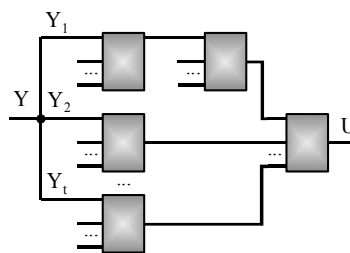


Рис. 2. Случай сходящихся ветвлений

Показатели наблюдаемости и тестопригодности предлагается вычислять, как показано в [8], а также использовать метод модификации, как представлено в [9].

Пример 1. Рассмотрим небольшую схему, представленную на вентильном уровне:

```
library IEEE;
use IEEE.STD_LOGIC_1164.all;
entity sh1 is
    port (
        w1,w2,w3,w4,w5,w6: in std_logic;
        wout: out std_logic);
end sh1;
architecture sh1 of sh1 is
    signal
        wk, wb, wc, we, wf, wa, wd: std_logic;
begin
    wa <= w1 and w2;
    wb <= w2 or w3;
    wc <= w3 or w4;
    wd <= not (w5 or w6);
    wk <= not (wc and wd);
    we <= not (wa or wb);
    wf <= wb or wk;
    wout <= not (we and wf);
end sh1;
```

Схема после синтеза без оптимизации представлена на рис. 3, а значения управляемости, наблюдаемости и тестопригодности приведены в табл. 1.

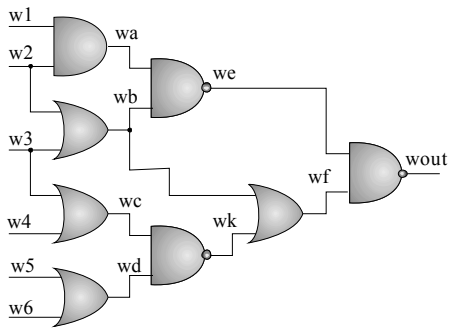


Рис. 3. Пример схемы для анализа

Таблица 1

Значения управляемости, наблюдаемости и тестопригодности схемы на рис. 3

Линия	$C^0(x)$	$C^1(x)$	$O(x)$	$T^0(x)$	$T^1(x)$
w1	0,500	0,500	0,695	0,347	0,347
w2	0,500	0,500	0,689	0,344	0,344
w3	0,500	0,500	0,695	0,347	0,347
w4	0,500	0,500	0,798	0,399	0,399
w5	0,500	0,500	0,664	0,332	0,332
w6	0,500	0,500	0,798	0,399	0,399
wa	0,937	0,812	0,620	0,581	0,504
wb	0,937	0,812	0,402	0,377	0,327
wc	0,812	0,937	0,671	0,545	0,629
wd	0,593	0,531	0,609	0,361	0,323
we	0,785	0,595	0,404	0,318	0,240
wk	0,553	0,785	0,734	0,407	0,577
wf	0,870	0,521	0,478	0,416	0,249
wout	0,538	0,675	1,000	0,538	0,675

Пример 2. Метод быть использован на модели устройства после синтеза. Рассмотрим небольшую схему, представленную на языке описания аппаратуры VHDL:

```

library IEEE;
use IEEE.STD_LOGIC_1164.all;
use IEEE.STD_LOGIC_ARITH.all;
entity karpati is
    port(
        a,b,c,d,e,f: in signed (1 downto 1);
        x,y: out signed(2 downto 0));
end karpati;
architecture karpati of karpati is
begin
    x <= (b*d)+(a*e);
    y <= (a*f)+(c*d)/(a*f)+(b*d);
end;
    
```

Ориентированный граф программы представлен на рис. 4.

Значения управляемости, наблюдаемости и тестопригодности приведены в табл. 2.

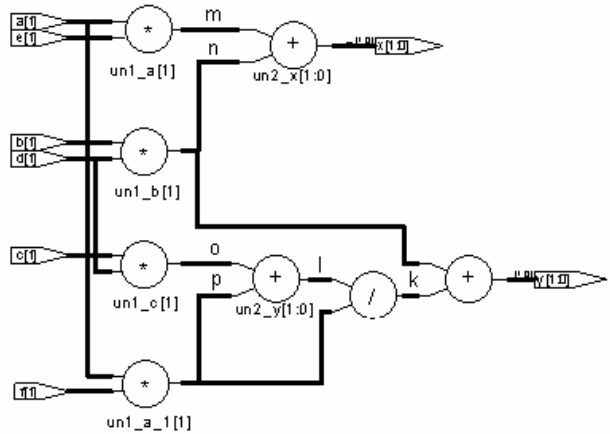


Рис. 4. Синтезированная схема для анализа

Таблица 2

Значения управляемости, наблюдаемости и тестопригодности схемы на рис. 4

Линия	$C^0(x)$	$C^1(x)$	$O(x)$	$T^0(x)$	$T^1(x)$
a	0,50	0,50	0,465	0,465	0,465
b	0,50	0,50	0,45	0,45	0,45
c	0,50	0,50	0,86	0,86	0,86
d	0,50	0,50	0,66	0,66	0,66
e	0,50	0,50	0,25	0,25	0,25
f	0,50	0,50	0,68	0,68	0,68
m	0,25	0,75	0,25	0,187	0,563
n	0,25	0,75	0,46	0,135	0,405
o	0,25	0,75	0,86	0,035	0,105
p	0,25	0,75	0,68	0,08	0,24
l	0,67	0,333	0,813	0,939	0,875
k	0,835	0,945	0,25	0,876	0,959
x	0,67	0,333	1	0,67	0,333
y	0,953	0,911	1	0,959	0,991

Анализ можно проводить как на досинтезированной структуре, так и на устройстве, обработанном средствами синтеза (устройство может быть разложено до вентиляльного уровня, как видно из рисунка ниже). На послесинтезированной структуре анализ усложняется за счет появления дополнительных сигнальных линий (за счет увеличения разрядности операндов).

Выводы

Предложен метод анализа тестопригодности, применимый для анализа цифровых систем представленных на различных уровнях описания – системном, уровне регистровых передач (синтезиро-

ванный код на языке описания аппаратуры), вентиляционном уровне.

Практическая значимость:

1). Выбор 2 – 5% дополнительных по наихудшим показателям наблюдаемости позволяет повысить качество заданного теста на 20 – 30%.

2). Возможность проведения анализа на самых ранних стадиях разработки, что позволяет существенно снизить количество дефектов в устройстве на этапе имплементации в кристалл.

3). Возможность разделения режимов тестирования и нормального функционирования устройства.

Практическая значимость. Метод реализован в рамках системы SIGETEST и может быть использован при построении детерминированных и взвешенных тестов для комбинационных и последовательностных схем, а также для нужд SoC.

Преимущества:

1) простота метода, при анализе последовательностных схем (не нужно решать систему линейных уравнений);

2) простота в выборе точек для модификации схемы;

3) простота и регулярность способа модификации схемы, что упрощает автоматизацию процесса анализа и синтеза (модификации) схемы;

4) возможность обеспечить высокий уровень тестируемости схемы (минимум или ноль непроверяемых линий) до генерации теста.

Недостаток метода: снижение быстродействия схемы за счет введения дополнительных аппаратных средств.

2. Kaminska M.A., Hahanov V.I., Wade Ghribi, Kamenuka E.A. Quantitative Estimation Of The Testability For Complexity Digital Devices Before Test Generation // Proceedings of the MIXDES'05 Conference. – Poland. – 2005. – P. 451-456.

3. Parker K.P. and McCluskey E.J. Probabilistic Treatment of General Combinational Networks // IEEE Trans. on Computers. – Vol. C-24. – No. 6. – 1975. – P. 668-670.

4. Larsson E., Peng Z. A Behavioral-Level Testability Enhancement Technique // IEEE European Test Workshop. – Constance. – Germany. – 1999.

5. Flottes M.L., Pires R., Rouzeyre B. Analyzing Testability from Behavioral to RT Level // Proc. European Design&Test Conf. – 1997. – P.158-165.

6. Zdenek Kotasek, Richard Ruzicka, Josef Strnadel, Jan Hlavicka Interactive Tool for Behavioral Level Testability Analysis // Proceedings of the IEEE ETW2001. – Stockholm. – SE. – 2001. – P. 117-119

7. Chen T-H. and Breuer M.A. Automatic Design for Testability Via Testability Measures // IEEE Trans. on Computer-Aided Design. – Vol. CAD-4. – 1985. – P. 3-11.

8. Кулак Э.Н., Каминская М.А. Модификация цифровых схем с использованием метода анализа тестопригодности TADATPG // Радиоэлектроника и информатика. – 2005. – № 3. – P. 113-119.

9. Кулак Э.Н., Каминская М.А. Модификация цифровых схем с использованием метода анализа тестопригодности TADATPG (часть 2) // Радиоэлектроника и информатика. – 2005. – № 4. – P. 60-68.

Литература

1. Goldstein L.M. and Thigen E.L. SCOAP: Sandia Controllability/Observability Analysis Program // Proc. 17th Design Automation Conf. – 1980. – P. 190-196.

Поступила в редакцию 5.02.2006

Рецензент: д-р техн. наук, проф. В.И. Хаханов, Харьковский национальный университет радиоэлектроники, Харьков.