

МАТРИЧНЫЙ СПОСОБ АНАЛИЗА ТЕСТОВОЙ РЕАКЦИИ В ПРОГРАММИРУЕМЫХ ЛОГИЧЕСКИХ ИНТЕГРАЛЬНЫХ СХЕМАХ

Г.П. Аксёнова

Рассмотрены способы анализа тестовой реакции объекта диагностирования с помощью сигнатурного анализатора. Предложен новый способ анализа, обеспечивающий локализацию неисправности и уменьшение объема контролирующей аппаратуры.

Ключевые слова: дискретное устройство, ПЛИС, тестирование, анализ тестовой реакции, сигнатурный анализатор, локализация неисправности.

ВВЕДЕНИЕ

Один из основных методов повышения надежности заключается в создании систем, которые при отказах элементов сохраняют свою работоспособность или восстанавливают ее в течение заданного отрезка времени. Требование отказоустойчивости наиболее часто возникает при создании систем специального назначения (например, для военного и космического применения). Более того, в связи с воздействием радиации и других видов естественного или искусственного облучения такие системы требуют проведения их тестирования в полевых условиях (т. е. в условиях их применения по назначению), где отсутствует необходимое внешнее тестовое оборудование.

Построение отказоустойчивых систем подразумевает введение избыточных (резервных) элементов, которые способны заменить отказавшие элементы системы. Обычно применяется резервирование на уровне системы в целом (например, ее дублирование) или на уровне крупных элементов (например, введение резервных процессоров в многопроцессорную систему). Резервирование на уровне мелких элементов неэффективно по объему затрат. Это объясняется тем, что суммарная стоимость восстанавливающих блоков (кросс-коммутаторов, мажоритарных блоков) для мелких элементов намного превышает стоимость введения целой резервной системы и одного восстанавливающего блока.

Появление высокопроизводительных ПЛИС способно изменить ситуацию. В ПЛИС для резервирования на низком уровне не требуется введения дополнительных кросс-коммутаторов, поскольку они уже в них имеются. Вместо введения резервных элементов требуется лишь оставить часть элементов незадействованными, чтобы в случае отказа путем перепрограммирования их и соответствующих связей произвести замену.

Таким образом, ПЛИС представляет собой идеальную аппаратную базу для создания отказоустойчивых систем на кристалле [1, 2]. В последнее время опубликовано немало работ на эту тему (см., например, [3–6]). В работе [3] проблема восстановления делится на три задачи: 1) проверка исправности системы в целом; 2) установление источника неисправности, т. е. локализация отказавшей компоненты; 3) системная реконфигурация/восстановление. Первая задача (проверка исправности) решается путем тестирования системы в целом. Этим большим и сложным вопросом в течение многих лет занимается ряд разработчиков-диагностов (см., например, работы [7–14]). Сюда же относятся и работы автора [15–17], на которые далее в статье будем опираться. Цель данной работы — решение второй задачи, т. е. задачи локализации неисправного компонента в системе, реализованной в ПЛИС. Предлагается новый способ анализа тестовой реакции в ПЛИС, обеспечивающий локализацию неисправности и уменьшение объема контролирующей аппаратуры.

1. ПРЕДВАРИТЕЛЬНЫЕ СВЕДЕНИЯ

В современных ПЛИС высокой степени интеграции протестировать реализованный в ПЛИС объект путем подачи тестовых воздействий на входы ПЛИС и наблюдением реакции на ее выходах, обеспечив при этом стопроцентное покрытие неисправностей, практически невозможно. Появление систем на кристалле (*system-on-chip* — SoC) и систем, содержащих встроенные IP-ядра, еще больше усложнило проблему тестирования ПЛИС. Решить проблему можно путем «залезания» внутрь кристалла, чтобы подавать тестовые воздействия на внутренние узлы (которые еще нужно определить в целях минимизации) и снимать реакцию тоже с каких-то внутренних узлов (чтобы не дать ей исказиться на пути к внешним выходам ПЛИС). Одним из таких подходов является давно существующая и продолжающая развиваться общепринятая техника проверки BIST (*built-in-self-test*) [18–24], т. е. встроенное самотестирование, при котором тестовые наборы генерируются внутри кристалла, обычно с помощью генератора псевдослучайной последовательности (ГПСП), и подаются в цепи сканирования, которые доставляют их к тестируемым частям схемы. Затем эти цепи используются для захвата выходной реакции частей и передачи ее в сигнатурный анализатор (в сигнатурные анализаторы), расположенный тоже внутри кристалла. Результирующая сигнатура сравнивается с эталонной, и на выход кристалла выдается лишь заключение о техническом состоянии объекта: годен или не годен.

Таким образом, для решения проблемы тестирования вся контролирующая аппаратура вносится внутрь кристалла. Эффективность встроенного самотестирования измеряется степенью покрытия неисправностей, аппаратурными затратами и временем выполнения теста. В BIST-технологиях применяют либо детерминированный, либо псевдослучайный тест для проверяемых частей. Детерминированный тест может обеспечить стопроцентное покрытие неисправностей, однако построение такого теста обходится слишком дорого и по времени, и по человеческим ресурсам. Поэтому разработчики перешли к псевдослучайному тестированию. Покрытие неисправностей у псевдослучайного теста приемлемой длины обычно ниже (до 80 %) [21]. Оставшиеся не обнаруженными неисправности требуют тестовых наборов, вероятность появления которых низка, и при генерации на ГПСП получается недопустимо большая длина теста, т. е. большие временные затраты на прогонку теста. Чтобы достичь стопроцентного покрытия одиночных константных неисправностей (наибо-

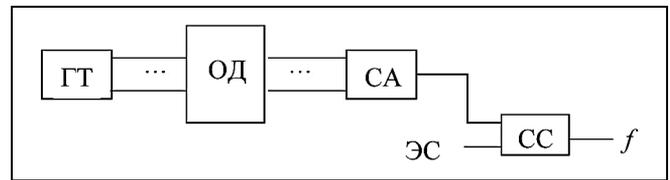


Рис. 1. Блок-схема тестирования объекта диагностирования: ГТ — генератор исчерпывающего теста; ОД — объект диагностирования; СА — сигнатурный анализатор; СС — схема сравнения; ЭС — эталонная сигнатура; f — сигнал ошибки

лее распространенный класс неисправностей) и сократить число требуемых для этого наборов ГПСП, в литературе предложен ряд методов [21–24], однако все они требуют дополнительных аппаратурных затрат.

В работах автора [15, 16] развит иной метод самотестирования ПЛИС, предназначенный для полевых условий и основанный на декомпозиции заключенной в кристалле схемы с использованием для полученных после декомпозиции частей схемы исчерпывающих тестов (тестов полного перебора), вырабатываемых внутри кристалла. Преимущества такого метода очевидны: не надо строить тест, так как на входах частей используется полный перебор; обнаруживаются не только одиночные, но и все кратные неисправности; малое время тестирования объекта благодаря параллельной проверке всех частей; отсутствие дополнительных задержек в цепях объекта при внесении в кристалл аппаратуры для самотестирования; независимость от сложности объекта.

В работе [16] в результате декомпозиции, проводимой путем частичного репрограммирования связей в ПЛИС, проверяемый объект, который будем называть далее объектом диагностирования (ОД), «рассыпается» на части, представляющие собой конфигурируемые логические блоки (КЛБ). На все КЛБ параллельно подается исчерпывающий тест, который в данном случае допустимо применять в связи с небольшим числом входов КЛБ (не превышающим десяти). В той же работе показано, как при этом проверяются связи между КЛБ. Выходы КЛБ соединены с сигнатурным анализатором (или с сигнатурными анализаторами, если это требуется для согласования числа КЛБ с числом входов анализатора). На рис. 1 изображена структурная схема, демонстрирующая используемый принцип тестирования. Сигнатурный анализатор (СА) представляет длинную выходную реакцию ОД в виде короткого ключевого слова — сигнатуры. Полученная сигнатура сравнивается с эталонной; при их отличии вырабатывается сигнал ошибки $f = 1$.

Несколько слов о СА [25].

- Он реализуется в виде регистра сдвига с обратными связями.
- Его достоверность оценивается как вероятность p необнаружения ошибок в анализируемой последовательности: $p = 1/2^l$, где l — число разрядов СА (или старшая степень l полинома, реализуемого анализатором). Один из способов повышения достоверности СА — это увеличение значения l . При $l = 16$ значение $p = 1/2^{16} = 0,0000152$ достаточно близко к нулю, и дальнейшее увеличение l не приводит к заметному повышению достоверности, а приводит к нежелательному повышению разрядности СА, т. е. к возрастанию аппаратных затрат. На практике целесообразно применять СА с числом разрядов, не превышающим 32. Наиболее часто принимаемое значение $l = 16$.
- Сигнатурные анализаторы бывают одноканальными (одноходовыми) и многоканальными (многоходовыми). Появление *многоканальных* СА было вызвано тем, что при тестировании *многовыходных* дискретных устройств требуется исследование достаточно большого числа выходных реакций, достигающего нескольких сотен. И если при анализе N -выходной схемы использовать одноканальный СА, то это приведет к увеличению в N раз времени, необходимого для последовательного анализа всех выходов, либо оборудования, требуемого для реализации N СА (т. е. по одному СА на каждом выходе).

2. СПОСОБЫ АНАЛИЗА ТЕСТОВОЙ РЕАКЦИИ МНОГОВЫХОДНОГО ОБЪЕКТА ДИАГНОСТИРОВАНИЯ С ПОМОЩЬЮ СИГНАТУРНОГО АНАЛИЗАТОРА

Как было отмечено, в режиме самотестирования ОД в ПЛИС представляет собой блочную систему, состоящую из N одновыходных блоков — КЛБ.

Анализ тестовой реакции такого многовыходного ОД с помощью СА обычно проводится двумя способами, которые назовем *раздельным* и *совместным*.

Раздельный способ анализа осуществляется путем подключения к выходу каждого блока «собственного» одноканального СА. Этот способ обеспечивает локализацию неисправного блока, но при больших аппаратных затратах (все l триггеров СА на один блок).

Совместный способ — применение многоканального СА, реакция со всех анализируемых выходов сворачивается в единую сигнатуру. Ясно, что это сопровождается потерей способности к ло-

кализации неисправного блока. Достоинство этого способа — малые аппаратные затраты (один триггер СА на один блок).

Предлагаемый в настоящей работе матричный способ анализа тестовой реакции многовыходного ОД занимает промежуточное положение по отношению к существующим способам, а именно:

- он обладает способностью к локализации (но при уменьшенном классе неисправностей);
- требует небольших затрат (два триггера СА на один блок).

3. МАТРИЧНЫЙ СПОСОБ АНАЛИЗА ТЕСТОВОЙ РЕАКЦИИ

Отвлечемся от ПЛИС и поясним матричный способ анализа тестовой реакции абстрактно:

— расположим блоки произвольно в виде $(m \times n)$ -матрицы, где m — число ее строк, n — число столбцов;

— присвоим каждому блоку номер (i, j) , где i — номер строки, j — номер столбца, в которых находится этот блок;

— блоки каждой строки подключим к своему многоканальному СА, а блоки каждого столбца — к своему СА' (рис. 2). Для этого потребуется m

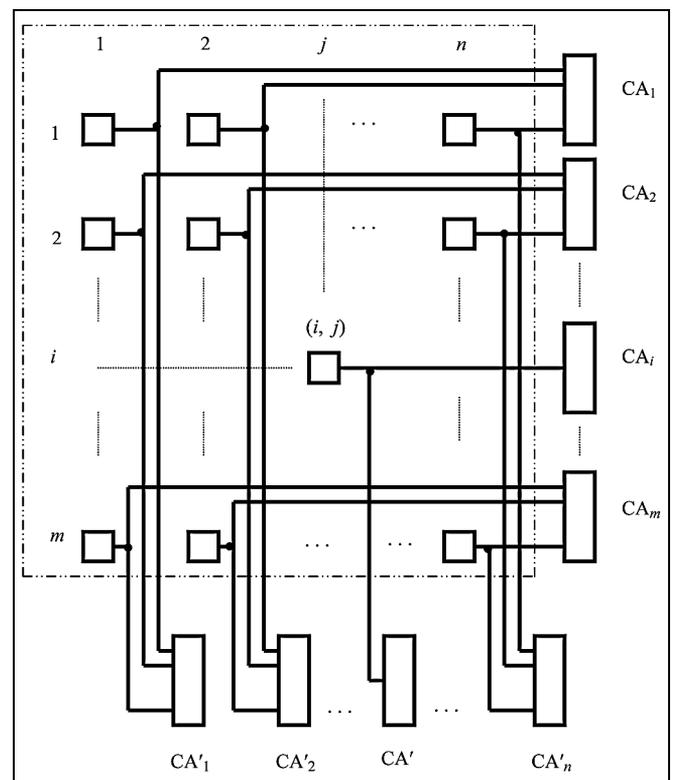


Рис. 2. К пояснению матричного способа анализа тестовой реакции

n -разрядных СА для строк и n m -разрядных СА' для столбцов.

Как видим, при матричном способе каждый блок контролируется дважды: в составе строки и в составе столбца (при одноразовой подаче исчерпывающего теста). Например, блок с номером 24 подключается к СА₂ и к СА'₄, причем безразлично к какому именно разряду. Следовательно, на один блок тратится два триггера контролирующей аппаратуры (по одному триггеру в двух СА). И аппаратные затраты на все СА составят $2N$ триггеров, где N — число блоков.

Посмотрим, какие неисправности обнаруживаются при таком способе анализа. Рассмотрим класс одиночных неисправностей матрицы, когда неисправным может быть только один блок (хотя в самом блоке допускается кратная неисправность).

Матричный способ контроля обеспечивает локализацию любого неисправного блока в классе одиночных неисправностей.

Доказательство этого положения очевидно. Если в матрице неисправен блок (i, j) (см. рис. 2), то при тестировании неисправность зафиксируют СА i -й строки и СА' j -го столбца. Отсюда можно сделать вывод, что неисправен блок, находящийся на пересечении этой строки и этого столбца, так как этот блок единственный, принадлежащий и строке i , и столбцу j . И этот элемент есть (i, j) .

Теперь рассмотрим кратные неисправности. Пусть, например, матрица содержит три неисправных блока с номерами 13, 22, 33 (они выделены жирным шрифтом):

11	12	13	14
21	22	23	24
31	32	33	34
41	42	43	44

При проведении тестирования неисправность зафиксируют СА строк 1, 2, 3 и СА столбцов 2, 3. Следовательно, под подозрение попадают блоки, стоящие на пересечении этих строк и столбцов, а именно: 12, 13, 22, 23, 32, 33. Неисправные блоки все попали в подозреваемое множество, однако туда попали и исправные блоки. Отсюда делаем следующий вывод.

Матричный способ контроля обеспечивает локализацию неисправных блоков при кратной неисправности следующего вида: неисправные блоки находятся все в одной строке или все в одном столбце.

Действительно, пусть в матрице в i -й строке находятся k неисправных блоков. В этом случае при контроле будут зафиксированы как неисправные i -я строка и k столбцов. Так как неисправная строка одна, то неисправные блоки однозначно будут находиться на пересечениях этой строки и

неисправных столбцов. Аналогичные рассуждения можно провести, если неисправные блоки находятся все в j -м столбце.

Как уже отмечалось, при матричном способе анализа тестовой реакции ОД общие аппаратные затраты на СА составляют $2N$ триггеров. Раздельный способ, также обеспечивающий локализацию неисправности, потребует lN триггеров. Следовательно, матричный способ дает выигрыш в аппаратуре в $Nl/2N = l/2$ раз, где l — число рядов СА, $l \geq 16$.

Изложенный матричный способ анализа тестовой реакции пригоден не только для ПЛИС, но и для любых дискретных блочных систем. В ПЛИС хорошо то, что матрица уже имеется, и матричный способ здесь лишь предписывает, с какими СА надо соединить выходы КЛБ.

ЗАКЛЮЧЕНИЕ

В статье рассмотрены вопросы анализа тестовой реакции многовыходного объекта диагностирования с помощью сигнатурного анализатора. Приведены существующие способы анализа и дано их сравнение по объему аппаратных затрат на сигнатурный анализатор и способности к локализации неисправного блока.

Показано, что каждый из способов хорош только по одному параметру: либо минимальны аппаратные затраты при отсутствии способности к локализации, либо аппаратные затраты максимальны, но зато имеется способность к локализации.

Предложен новый способ анализа, приемлемо сочетающий в себе достоинства обоих предшествующих способов: он обладает способностью к локализации, но при несколько уменьшенном классе неисправностей, и затраты в нем невелики — два триггера из двух сигнатурных анализаторов на один блок объекта диагностирования. Способ назван матричным.

Способ может найти применение при прототипировании отказоустойчивых заказных или полузаказных схемных решений на ПЛИС с последующей их реализацией в заказных СБИС.

ЛИТЕРАТУРА

1. Lach J., Mangione-Smith W.H., Potkonjak M. Low Overhead Fault-Tolerant FPGA Systems // IEEE Trans. Very Large-Scale Integrat. (VLSI) Syst. — 1998. — Vol. 6, N 2. — P. 212—221.
2. Уваров С.С. Проектирование реконфигурируемых отказоустойчивых систем на ПЛИС с резервированием на уровне ячеек // Автоматика и телемеханика. — 2007. — № 9. — С. 176—89.



3. Gupta S., Feng S., Ansari A., Mahlke S. StageNet: A Reconfigurable Fabric for Constructing Dependable CMPs // IEEE Trans. Computers. — 2011. — Vol. 60, N 1. — P. 5—19.
4. Banerjee P., Sangtani M., Sur-Kolay S. Floorplanning for partially reconfigurable FPGAs // IEEE Trans. Comput. Aided Design of Integr. Circuits Syst. — 2011. — Vol. 30, N 3. — P. 8—17.
5. Tang X., Wang S. A low hardware overhead self-diagnosis technique using Reed-Solomon codes for self-repairing chips // IEEE Trans. Computers. — 2010. — Vol. 59, N 10. — P. 1309—1319.
6. Huang Y.-J., Li J.-F. Built-in self-repair scheme for the TSVs in 3-D Ics // IEEE Trans. Comput. Aided Design of Integr. Circuits Syst. — 2012. — Vol. 31, N 10. — P. 1600—1612.
7. Quasem M., Jiang Z., Gupta S. Benefits of a SoC-specific test methodology // IEEE Design & Test Comput. — 2003. — May—June. — P. 68—77.
8. Menon P., Xu W., Tessier R. Design-specific path delay testing in lookup table-based FPGAs // IEEE Trans. Comput. Aided Design of Integrated Circuits and Systems. — 2007. — Vol. 26, N 12. — P. 2201—2214.
9. Cota E., Kastensmidt F., Cassel M., et al. A high-fault-coverage approach for the test of data, control, and handshake // IEEE Trans. Comput. Aided Design of Integrated Circuits and Systems. — 2006. — Vol. 25, N 5. — P. 867—877.
10. Tahoori M., Mitra S. Application-independent testing of FPGA interconnects // IEEE Trans. Comput. Aided Design of Integrated Circuits and Systems. — 2005. — Vol. 24, N 11. — P. 1774—1783.
11. Chakrabarty K. Low-cost modular testing and test resource partitioning for SoCs // IEE Proc. Comput. Digit. Techn. — 2005. — Vol. 152, N 3. — P. 427—441.
12. Lin Y.-C., Lu F., Cheng K.-T. Pseudofunctional testing // IEEE Trans. Comput. Aided Design of Integrated Circuits and Systems. — 2006. — Vol. 25, N 8. — P. 1535—1546.
13. Grecu C., Ivanov A., Saleh R., Pande P. Testing network-on-chip communication interconnects in mesh network-on-chip // IEEE Trans. Comput. — 2008. — Vol. 57, N 9. — P. 1202—1214.
14. Li L., Chakrabarty K., Kajihara S., Swaminathan S. Three-stage compression approach to reduce test data volume and testing time for IP cores in SoCs // IEE Proc. Comput. Digit. Techn. — 2005. — Vol. 152, N 6. — P. 704—712.
15. Аксенова Г.П., Халчев В.Ф. Метод параллельно-последовательного самотестирования в интегральных схемах типа FPGA // Автоматика и телемеханика. — 2007. — № 1. — С. 163—174.
16. Аксенова Г.П. Контролепригодная архитектура для самотестирования в программируемых логических матричных структурах // Автоматика и телемеханика. — 2010. — № 12. — С. 154—165.
17. Аксенова Г.П. Сжатие тестовой реакции при самотестировании в программируемых логических матрицах // Автоматика и телемеханика. — 2013. — № 2. — С. 124—138.
18. Das S.R. Self-testing of cores-based embedded systems with built-in hardware // IEE Proc. — Circuits Devices Syst. — 2005. — Vol. 152. — N 5. — P. 539—546.
19. Tehranipoor M., Rad R. Built-in-self-test and recovery procedures for molecular electronics-based nanofabrics // IEEE Trans. Comput. Aided Design of Integrated Circuits and Systems. — 2007. — Vol. 26, N 5. — P. 943—957.
20. Dutt S., Verma V., Suthar V. Built-in-self-test of FPGAs with provable diagnosabilities and high diagnostic coverage with application to online testing // IEEE Trans. Comput. Aided Design of Integrated Circuits and Systems. — 2008. — Vol. 27, N 2. — P. 309—326.
21. Kay D., Mourad S. Interactive built-in-self-test compression for testing a system-on-a-chip // IEE Proc. Comput. Digit. Techn. — 2003. — Vol. 150, N 4.
22. Wang S. Low hardware overhead scan based 3-weight weighted random BIST // Proc. Int. Test Conf. 2001. — P. 868—877.
23. Li L., Chakrabarty K. Test set embedding for deterministic BIST using a reconfigurable interconnection network // IEEE Trans. Comput. Aided Design of Integrated Circuits and Systems. — 2004. — Vol. 23, N 9. — P. 1289—1305.
24. Chatterjee M., Pradhan D.K. A BIST pattern generator design for near-perfect fault coverage // IEEE Trans. Comput. — 2003. — Vol. 52, N 12. — P. 1543—1557.
25. Ярмолик В.Н. Контроль и диагностика цифровых узлов ЭВМ. — Минск: Наука и техника, 1988. — С. 137—235.

Статья представлена к публикации членом редколлегии чл.-корр. РАН П.П. Пархоменко.

Аксенова Галина Петровна — канд. техн. наук, ст. науч. сотрудник, ☎ (495) 334-77-39, ✉ aksenova@ipu.ru, Институт проблем управления им. В.А. Трапезникова РАН, г. Москва.

Содержание сборника «Управление большими системами», 2014, вып. 51

- ✓ Корноушенко Е.К. Целенаправленное управление состоянием когнитивной линейной модели с ограниченным множеством состояний. — С. 6—25.
- ✓ Сидельников Ю.В., Шалышкин М.И., Шевыренков М.Ю. Обзор зарубежных сценарных прогнозов и форсайтов: инструменты информационного управления. — С. 26—59.
- ✓ Подлазов В.С. Повышение характеристик многомерных торов. — С. 60—81.
- ✓ Губанов Д.А., Микулич Л.И., Наумкина Т.С. Использование языковых игр для исследования социальных сетей на примере поиска сообществ и влиятельных агентов. — С. 82—106.
- ✓ Акинфиев В.К. Выбор инвестиционных стратегий компаний в условиях нестабильности рынков. — С. 107—129.
- ✓ Искаков М.Б., Искаков А.Б. Равновесие, сдерживаемое контругрозами, и сложное равновесие в безопасных стратегиях. — С. 130—157.
 - ✓ Усков А.А., Киселев И.А. Оценка показателей эффективности инвестиционных проектов при использовании заемных средств в нечетких условиях. — С. 158—173.
 - ✓ Кулинич А.А. Модель командного поведения агентов (роботов): когнитивный подход. — С. 174—196.

Тексты статей доступны на сайте <http://ubs.mtas.ru/>

