

АНАЛИТИЧЕСКИЙ МЕТОД ПОИСКА ДЕФЕКТОВ В ЦИФРОВЫХ СИСТЕМАХ НА КРИСТАЛЛАХ

Хаханов В.И., Сушанов А.В., Хаханова А.В., Чумаченко С.В.
Харьковский национальный университет радиоэлектроники
Украина, 61166, Харьков, пр. Ленина 14
Тел., факс: (057) 702-13-26, E-mail: hahanov@kture.kharkov.ua

Предлагаются методы диагностирования цифровых систем на кристалле на основе использования дизъюнктивной нормальной формы, представляющей собой матрицу покрытия дефектов тестовыми последовательностями. Метод ориентирован на встроенное сервисное обслуживание функциональностей представленными IP модулями. Рассмотрена стратегия выбора контрольных точек, которая направлена на поиск кратного дефекта в функциональности системы на кристалле в реальном масштабе времени.

1. Суть метода половинного деления

При разработке цифровой системы на кристалле приходится решать задачи, которые обеспечили бы проверку наличия неисправностей в схеме. Существует достаточно много способов анализ дефектов, но сложность состоит в быстром поиске нужного термина среди возможного сочетания дефектов, представленного в виде дизъюнктивной нормальной формы (ДНФ). Эффективное решение этой проблемы обеспечит уменьшение временных и материальных затрат при проектировании SOC.

Метод половинного деления используется для существенного уменьшения области подозреваемых дефектов. Для уменьшения множества дефектов необходимо получить дополнительную информацию для ДНФ дефектов. Для этого используется процедура зондирования внутренних точек контроля, с помощью которой определяется состояние внутренней линии в целях исключения дефектов или их повреждение. Стратегия выбора контрольной точки ориентирована на примерно половинное деление подозреваемого множества, исключая на каждом шаге половину дефектов путём моделирования – минимизации исходной ДНФ.

Суть метода половинного деления на ДНФ, которая представляет собой все возможные сочетания дефектов в цифровой схеме, можно продемонстрировать на следующем примере: $F = F_1 F_2 F_3 \vee F_2 F_3 F_4 \vee F_1 F_3 F_5 F_6$. Выбор первой контрольной точки контроля, например, $F_5 = 0$ превращает функцию в уменьшенное выражение, в то время как $F_5 = 1$ не уменьшает размерность ДНФ.

$$F = \begin{cases} F_5 = 0 \Rightarrow (F = F_1 F_2 F_3 \vee F_2 F_3 F_4 \vee F_1 F_3 F_5 F_6) = F_1 F_2 F_3 \vee F_2 F_3 F_4 \\ F_5 = 1 \Rightarrow (F = F_1 F_2 F_3 \vee F_2 F_3 F_4 \vee F_1 F_3 F_5 F_6) = F = F_1 F_2 F_3 \vee \\ \vee F_2 F_3 F_4 \vee F_1 F_3 F_5 F_6 \end{cases}$$

Это подтверждает присутствие дефекта на линии. Алгоритм выбора точек контроля необходимо ориентировать на максимальное уменьшение исходной ДНФ. Критерием выбора точки контроля может служить взвешенность мощностей ДНФ, полученных после моделирования обоих состояний проверки.

Для выбора контрольной точки используются следующие утверждения:

Утверждение 1. Если F_i присутствует во всех терминах ДНФ, то данный дефект существует в схеме обязательно, а данную линию не следует тестировать. Иначе, если предположить, что она равна нулю, все термины обращаются в нуль, что противоречит условию существования ненулевых значений вектора экспериментальной проверки V .

Утверждение 2. В схеме присутствует только одно сочетание дефектов, которой определено одним термом ДНФ. Если найдено одно подтверждённое решение в виде терма ДНФ, то остальные термы следует исключить из рассмотрения путём их обращения в нуль.

Утверждение 3. Вероятность того, что рассматриваемый терм является дефектом, меньше, если в нём содержится элемент, который встречается реже в других термах.

Поэтому задача точек контроля для минимизации сводится к выполнению двух альтернативных стратегий: 1. Рассмотрение переменных в термах минимальной длины для подтверждения всех дефектов в терме путём их зондирования; 2. Проверка таких переменных, которые обращают в нуль не более половины термов ДНФ.

Для более быстрого поиска кратного дефекта можно параллельно зондировать в нескольких контрольных точках, но при этом каждая из них должна отвечать принятым утверждениям. В результате можно сформулировать четвёртое утверждение, которое способствует поиску дефектного терма.

Утверждение 4. Одновременно можно зондировать несколько термов в различных контрольных точках.

Критерием окончания процедуры диагностирования является получение одного терма ДНФ, которое идентифицирует дефекты в функциональности цифровой системы на кристалле.

2. Описание к программе «Defect Analyzer»

Программа «Defect Analyzer» является приложением под OS Windows, написана на языке C++ в среде программирования Borland C++ Builder 2006, содержит 659 операторов, 604 строк. «Defect Analyzer» предназначен для проведения алгебрологического метода диагностирования схемной структуры, позволяет проиллюстрировать последовательный и параллельный методы выбора контрольных точек, в которых проводится интерактивная процедура зондирования. В качестве входных данных в программе является таблица покрытия, которая построена при тестировании схемы на наличие дефектов. Таблица отображает уровни сигналов, которые подаются на вход, а также наличие неисправности в данном тесте, на что указывает значение результирующего вектора V (рис. 1).

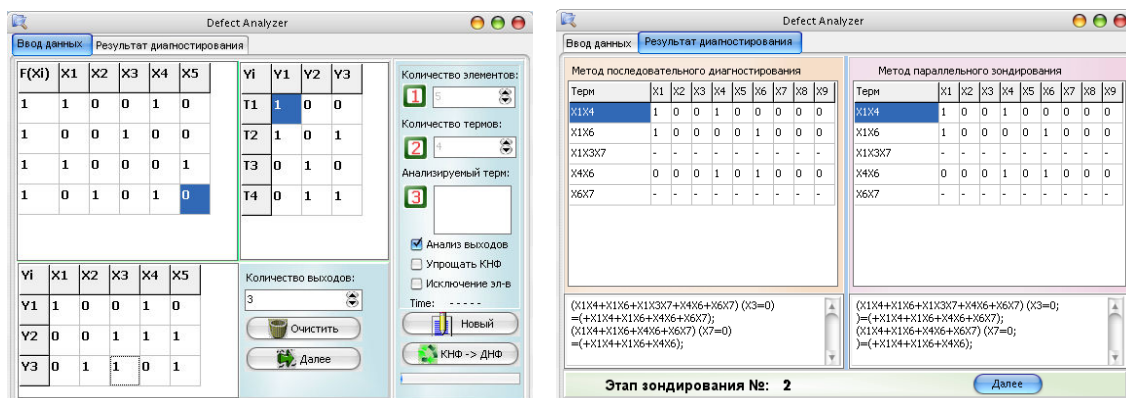


Рис. 1. Общий вид программного средства и пошаговое проведение процедуры зондирования.

Для упрощения исходной таблицы покрытия можно использовать информацию о нулевых элементах тестовых наборов, значениях сигналов на выходах схемной структуры для каждого тестового набора, а также благодаря предварительному упрощению КНФ. Затем пошагово проводится процедура зондирования в определённой

контрольной точке, что отображается для каждого из методов в виде изменяющейся таблицы покрытия, а также упрощения исходной ДНФ.

Объём данных, который позволяет обрабатывать программа ограничен количеством термов неупрощённой ДНФ, которое отображает таблица. Количество строк таблицы (термов) не должно превышать 32766. Численность термов ДНФ зависит от исходных данных и вычисляется по формуле:

$$N = \prod_{i=1}^n (N_i),$$

где N_i – количество единичных элементов в тестовом наборе.

Рассматриваются входные последовательности, где значение вектора экспериментальной проверки равно единице. Основные временные затраты программы приходятся на преобразование КНФ в ДНФ и её упрощение. Время необходимое для данных преобразований зависит от данных эксперимента и пропорционально некоторому коэффициенту k , вычисляемому по следующей формуле:

$$k = k_1 \prod_{i=1}^n ((N_i!)) + k_2 \frac{n!}{2(n-2)!} nm,$$

где n – количество тестовых наборов; m – количество элементов, участвующих в эксперименте; N_i – количество единичных элементов в i -м входном наборе; k_1 и k_2 – некоторые коэффициенты, зависящие от конкретных данных.

При использовании анализа нулевых элементов, упрощении таблицы покрытия неисправностей путём анализа выходов схемной структуры, упрощении исходной КНФ до преобразования в ДНФ значительно уменьшают время преобразования и упрощения ДНФ-дефектов.

Пример.

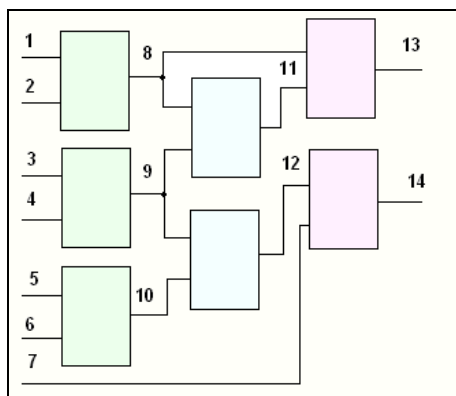


Рисунок 2. Пример схемной структуры

После проведения преобразований получаем наличие в схеме кратного дефекта $F_8 F_9$.

Для доказательства используется параллельное и последовательное зондирование. При последовательном зондировании в качестве первой контрольной точки следует взять F_1 . При параллельном зондировании в качестве контрольной точки может быть выбрана точка F_{11} . После проведенных преобразований остается единственный терм $F_8 F_9$, чем подтверждается, что в схеме существует кратный дефект, которым является единственный терм упрощённой ДНФ.

Для сравнения эффективности использования различных методов поиска дефективного термина ниже приведён график (рис.3) зависимости количества необходимых зондирований от значения оценки по Квайну Q (по результатам рассмотренных примеров).

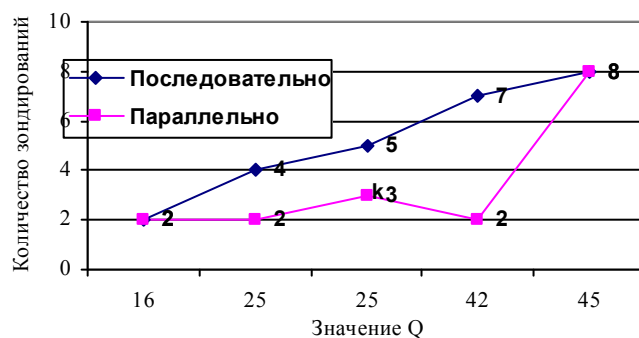


Рис.3. Зависимость количества зондирований от Q

3.Выводы

Метод половинного деления основан на использовании интерактивной процедуры зондирования внутренних точек контроля, благодаря которым определяются состояния внутренних линий в целях исключения дефектов. Метод параллельного зондирования позволяет выбирать несколько контрольных точек, что обеспечивает более быстрый поиск дефектного термина. Однако при некоторых исходных ДНФ параллельное зондирование не имеет преимуществ над последовательным. Это обусловлено отсутствием на любом из шагов проверки (исключением может быть последний этап зондирования) различных элементов с одинаковыми весовыми коэффициентами.

Научная новизна и практическая значимость. Предложен новый матричный метод диагностирования дефектов функциональностей в SoC, который характеризуется использованием дизъюнктивной нормальной формы и таблицы покрытия неисправностей, что дает возможность получать полные и минимальные сочетания кратных неисправностей на основе использования процедур мультizonдирования. Программная реализация последовательного и параллельного методов моделирования проведения процедуры тестирования схем на кристалле на наличие неисправностей. Данные методы ориентированы на встроенное сервисное обслуживание функциональностей представленными IP модулями.

Список литературы:

1. Zorian Yervant, Marinissen Erik Jan, and Dey Sujit Testing Embedded Core-Based System Chips // Proceedings IEEE International Test Conference (ITC).– Washington.– October 1998.– P. 130-143.
2. Zorian Yervant What is Infrastructure IP? // IEEE Design & Test of Computers.– May-June 2002.– P. 5-7.
3. Zorian Yervant Advances in Infrastructure IP // IEEE Design & Test of Computers. – May-June 2003.– 49 p.
4. 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 DSD 2001. Warsaw, Poland. September, 4-6, 2001. P. 228-235.
5. Abramovisi, M., Breyer M.A. and Friedman A.D. Digital systems testing and testable design. Computer Science Press 1998.652p.
6. 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.
7. Levendel Y.H., Menon P.R. Comparison of fault simulation methods – Treatment of unknown signal values // Journal of digital systems. Vol. 4. 1980. P. 443-459.