

УДК 681.3

В.И. ХАХАНОВ<sup>1</sup>, В.В. ЕЛИСЕЕВ<sup>2</sup><sup>1</sup>Харьковский национальный университет радиоэлектроники, Украина<sup>2</sup>ЗАО «Северодонецкое НПО «Импульс», Украина

## ПРИМЕНЕНИЕ IEEE СТАНДАРТОВ ДЛЯ ТЕСТИРОВАНИЯ ПРОГРАММНО-ТЕХНИЧЕСКИХ КОМПЛЕКСОВ

Предлагается подход к использованию IEEE стандартов тестирования для диагностирования сложных иерархических программно-технических комплексов, относящихся к критическим технологиям. Используются различные ad-hoc решения тестирования систем на кристаллах. Предложен алгоритм тестирования программно-технических комплексов на различных уровнях иерархии.

**System on Chip (SoC), Network on Chip (NoC), IEEE стандарты, ad-hoc технологии, CTL**

### Актуальность исследования

Электронные технологии все более продвигают на мировой рынок системы и сети на кристаллах (SoC, NoC) обладающие, по сравнению с системами на платах, высокой частотой синхронизации и быстрым действием, низким энергопотреблением, малыми размерами (1 – 4 см<sup>2</sup>), широкой функциональностью (включающей и приемопередатчик – Bluetooth), высокой степенью интеграции и интеллектуальной собственностью (IP) [1 – 3]. Компоненты NoC [4], по сравнению с системами на кристаллах, имеют дополнительную возможность обмена информацией по протоколу TCP/IP на всех шести уровнях OSI-стандарта. Что касается развития и интеграции в цифровую систему Wireless технологий, то их уже нельзя не учитывать при создании новых компьютерных систем и сетей. «Наша цель – по известному высказыванию вице-президента Intel Патрика Гелсингера – предоставить каждому жителю планеты возможности для вычислений и коммуникаций. Цифровые wireless-технологии приводят к технической революции в радиотехнике на основе использования микропроцессоров для создания программно-управляемых радиоустройств и интеллектуальных антенн. Новое поколение компьютеров или

цифровых систем будет содержать встроенные средства радиообмена, если они обеспечивают новые полезные свойства для потребителя». Создается беспроводное общество в глобальном масштабе, когда пользователь остается на связи по Internet или другой сети с помощью любого устройства, в каждой географической точке планеты (дома, в дороге или в офисе). Wireless для цифровых систем – это: 1) SoC с приемопередатчиком Bluetooth; 2) новые беспроводные архитектуры компьютеров и периферийных устройств (Wi-Fi); 3) цифровой офис и цифровой дом (Wi-Max), определяемые взаимодействием различных систем (компьютеры, бытовая электроника, карманные устройства); 4) RFID – радиочастотная идентификация для распознавания объектов на расстоянии с помощью цифровых радиосигналов. Таким образом, цифровая- и радио-техника более не могут сосуществовать и развиваться раздельно. Они революционно объединяются в настоящее время в беспроводные компьютеры или в компьютерную радиотехнику, что, в конечном счете – одно и то же, объединенное понятием цифровой системы. Одним из главных требований, предъявляемых к информационным и управляющим системам (ИУС), важным для безопасности АЭС, является обеспечение высокой надежности ее функциони-

рования [5]. К таким ИУС, в частности относятся система внутрореакторного контроля, аппаратура нейтронного потока, блочная информационно-вычислительная система [6, 7]. Программно-технические комплексы (ПТК), реализующие функции ИУС, являются сложными системами, состоящими из большого количества взаимодействующих компонентов [8]. При этом возникает ряд проблем, существенных с точки зрения безопасности [5], связанных со структурной и функциональной сложностью, повышающей вероятность ошибки при разработке; высокой трудоемкостью создания диагностического обеспечения; разработкой полных тестов для обнаружения отказов; выполнением жестких требований к времени поиска дефекта и идентификации неисправного компонента. Важнейшим этапом разработки компонентов ПТК является процесс верификации, связанный с устранением всех ошибок проектирования как можно на более ранней стадии, что приводит к значительной экономии временных и материальных ресурсов (рис. 1).

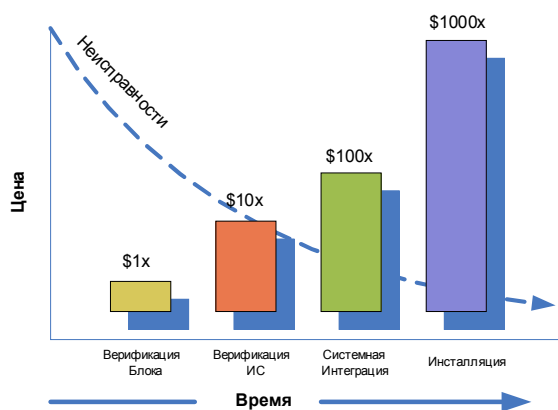


Рис. 1. Стоимость ошибок на стадиях проектирования сложных систем

Поэтому допустимое внесение тестопригодной избыточности в состав проекта на ранней (системной) стадии разработки является актуальной проблемой.

Другая проблема связана с использованием теста верификации, полученного на стадии системного проектирования, для проверки уже готового компо-

нента (изделия) с минимальными дополнительными аппаратными и программными затратами. Она может быть рассмотрена и решена в контексте использования нового стандарта IEEE 1500 SECT [9]. При этом механизм программно-аппаратной избыточности должен включать дополнительные точки контроля, которые необходимо ввести в проект с помощью Boundary Scan регистра или специальных (ad hoc) технологий. Таким образом, однажды созданная избыточность проекта может быть многократно использована для проверки компонентов ПТК на всех стадиях его жизненного цикла.

Сложные цифровые системы в настоящее время рассматриваются как объекты, имеющие несколько уровней иерархии. На самом низком уровне система представлена современными интегральными схемами (PLD, ASIC), в которые имплементированы SoC, NoC, память, процессоры. Второй уровень формируется цифровыми устройствами, представленными на платах (system on board), где в качестве примитивов выступают интегральные схемы первого уровня. Третий уровень представляет собой совокупность плат, которые объединяются в понятие крейта (system on crate). Типичным представителем такой системы является компьютер IBM PC. Четвертый уровень объединяет совокупность крейтов или шкафов в сложную систему управления технологическими процессами, производствами или критическими технологиями (авиация, космонавтика, атомная энергетика, метеорология, оборона, экология) [8]. Пятый уровень можно рассматривать как рассредоточенную в пространстве систему, примером которой может быть Internet. Далее рассматриваются иерархии систем от первого до четвертого уровней в целях создания моделей и методов их тестирования с заданной глубиной диагностирования.

Цель исследования – существенное уменьшение времени тестирования и диагностирования сложных цифровых систем на основе разработки общей модели организации и проведения диагностического

експеримента, включающего условные и безусловные алгоритмы поиска дефектов с использованием IEEE стандартов тестопригодного проектирования.

Для достижения поставленной цели рассматриваются следующие задачи:

1) обзор методов тестирования сложных цифровых систем всех, обозначенных выше уровней иерархии;

2) разработка иерархической модели организации и проведения диагностического эксперимента, включающего условные и безусловные алгоритмы поиска дефектов с использованием стандартов тестопригодного проектирования;

3) построение ad-hoc моделей тестирования, ориентированных на специфику представления цифровых устройств упомянутых уровней иерархии;

4) практическая реализация моделей тестирования сложных цифровых систем и экспериментальная оценка из состоятельности.

Объект исследования – цифровая система, представленная несколькими уровнями иерархии, разрабатываемая на основе IEEE стандартов тестопригодного проектирования. В частности, рассматривается программно-технический комплекс  $F$ , как объект тестирования, который может быть представлен в виде кортежа:  $F = \langle C, B, P, M \rangle$ , где  $C = \{c_1, c_2, \dots, c_n\}$  – конечное непустое множество крейтов в системе;  $B = \{b_1, b_2, \dots, b_m\}$  – множество плат в крейте;  $P = \{p_1, p_2, \dots, p_k\}$  – интегральных схем на печатной плате;  $M = \{m_1, m_2, \dots, m_l\}$  – множество IP-модулей в интегральной схеме.

Основное уравнение диагноза для рассматриваемого объекта исследования может быть представлено в следующем виде:

$$D = R \wedge L = (R_m \oplus R_r) \wedge L = [(T \oplus F) \oplus (T \oplus F_r)] \wedge L,$$

где параметры  $D, L, R, R_m, R_r, T, F, F_r$  – есть: множество обнаруженных в устройстве дефектов; покрытие неисправностей тестом; двоичный вектор экспериментальной проверки (ВЭП), равный числу

наблюдаемых выходов; эталонный вектор состояния выходов; вектор экспериментальной проверки состояний выходов; тест проверки неисправностей; эталонная модель устройства; реальное изделие.

Методы исследования – теория графов, дискретная математика, техническая диагностика, теория цифровых систем, средства цифрового моделирования.

### Методы тестирования с использованием IEEE стандартов

Три параметра: выход годных изделий (yield ratio) при производстве, time-to-market при проектировании и надежность при эксплуатации являются определяющими при оценке эффективности средств тестирования цифровой системы. Естественно, что революционные преобразования в нано- и субмикронных технологиях изготовления кристалла и его использования для создания SoC и NoC, создают новые проблемы тестирования и диагностирования сложных цифровых систем и сетей [10, 11]. Решение одной из них связано с уменьшением количества дефектных изделий DL в зависимости от качества теста T [12]:  $DL = 1 - Y^{(1-T)}$ . При тесте, покрывающем 100% всех неисправностей, выход годных изделий будет максимальным, что означает – на рынке не появятся дефектные изделия. Что касается time-to-market, то и здесь одним из существенных критериев является качество теста, влияющее на время верификации цифрового проекта или тестирования готового изделия. Также качество покрытия неисправностей тестом является определяющим для обеспечения надежности работающего изделия, поскольку несвоевременное диагностирование возникшего дефекта может привести к катастрофическим последствиям. Для решения упомянутых проблем ведущие фирмы используют мировой опыт тестопригодного проектирования, обобщенный в стандарты общества IEEE [13 – 15].

IEEE 1149.1-2001 Standard Test Access Port and

Boundary-Scan Architecture [13] определяет архитектуру и функции тестовой логики, которая встраивается в интегральную схему и предоставляет стандартные средства для: тестирования соединений на плате между интегральными схемами; проверки самой интегральной схемы; наблюдения и управления значениями входов и выходов логики в режиме нормальной работы. Тестовая логика содержит регистр граничного сканирования (BSR, Boundary Scan Register) и тестовый порт (TAP, Test Access Port). Схемотехнические решения, определенные в стандарте, позволяют загружать в микросхему инструкции и входные воздействия, а также наблюдать результаты выполненных тестовых операций. Вся информация (инструкции, тестовые воздействия и реакции) передается в режиме последовательной передачи данных. Проблема тестирования любого цифровой системы на плате, состоящей из нескольких взаимосвязанных микросхем, заключается в решении трёх задач: 1) проверка функциональности каждого компонента; все микросхемы имеют корректное соединение; вся система на плате выполняет требуемую функцию. Такой подход можно применить не только для плат, составленных из интегральных микросхем, но и к сложным системам на кристалле, состоящим из множества более простых функциональных модулей. IEEE 1149.4-1999 Standard for a Mixed-Signal Test Bus [14] определяет средства тестирования и протоколы проверки аналого-цифровых компонентов для решения следующих задач: тестирование межсоединений, обрывов, замыканий, временных параметров, цифровых и аналоговых характеристик элементов на печатной плате. IEEE 1149.6-2003 Standard for Boundary-Scan Testing of Advanced Digital Networks [15] расширяет стандарт IEEE Std 1149.1 в целях улучшения возможностей тестирования дифференциальных межсоединений и/или цепей питания переменного тока в интегральных схемах на печатных платах и определяет дополнительные элементы, позволяющие

проводить диагностирование с высокой разрешающей способностью.

Основной функцией IEEE 1500 Standard for Embedded Core Test (SECT) [9] является стандартизация интерфейса доступа для тестирования внутреннего модуля, входящего в состав цифровой системы на кристалле, и изоляции его от окружающей среды. Он позволяет также тестировать среду, внешнюю по отношению к модулю, путем изоляции последнего и обеспечения его прозрачности. Стандарт предоставляет разработчику аппаратные средства обеспечения доступа к внутренним модулям для управления входными переменными и наблюдения выходных линий. При этом каждый модуль имеет стандартизированный интерфейс для тестирования, независимо от производителя и выполняемых функций. При создании тестового обеспечения системы в целом разработчик в полной мере может использовать тесты компонентов (IP-cores), которые поставляются производителями совместно с IP-модулями. Следует подчеркнуть, что стандарт не регламентирует внутренних требований и методов тестирования модулей, принципов построения и методологии разработки системы. Первое полностью является прерогативой разработчиков модулей, а второе – разработчика системы на кристалле. Для проверки работоспособности системы на кристалле и входящих в нее модулей применяются два подхода: тестирование с помощью внешнего автоматического тестирующего оборудования (ATE), либо с помощью интегрированного в систему оборудования BIST. В обоих случаях необходимо обеспечить возможность загрузки тестовых наборов на входы соответствующего модуля и вывода результатов тестирования на выходы системы для дальнейшего их анализа, что возлагается на механизмы доступа к встроенным в систему модулям. Стандарт IEEE 1500 SECT включает два компонента: техническое обеспечение и язык тестирования модулей, расположенных на кристалле (Core Test Language).

Он обеспечивает понимание проекта разработчиками и повторное использование тестового обеспечения составных компонентов, что дает возможность уменьшить время выхода нового изделия на рынок и повышение процента годных изделий (yield ratio). Основное достоинство тестопригодных стандартов заключается в обеспечении сканирования внутренних переменных цифровой системы на плате и в кристалле, включая переменные функциональных блоков, размещенных на кристалле. На рис. 2 представлен интерфейс стандарта IEEE 1500 SECT.

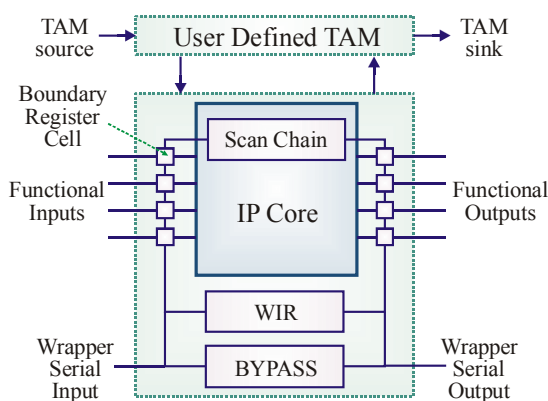


Рис. 2. Standard IEEE P1500 Wrapper Components

Каждый из компонентов системы подключается к TAP-контроллеру и тестируется как независимая схема, которая должна иметь отдельный тест, TAP-контроллер, регистр команд, boundary scan регистр и дополнительную логику для декодирования команд. Использование стандарта позволяет проверять функциональность модуля, ошибки межсоединений, определять место, причину и вид дефекта. Основным компонентом Boundary Scan регистра является ячейка сканирования, представленная на рис. 3. Ее следует рассматривать как аппаратную избыточность для каждой линии, которая подлежит наблюдению и управлению. Поэтому число таких линий в проекте должно быть ограниченным.

Кроме того, следует учитывать временной фактор тестирования, связанный с использованием Boundary Scan регистра. Чем больше разрядов  $L(R)$  он имеет, тем существеннее время, затрачиваемое на

тестирование, зависящее от числа входов модуля  $N(X)$  и длины теста  $L(T)$ , определяемой следующей формулой:  $Q = N(X) \times L(T) + L(R) \times L(T)$ .

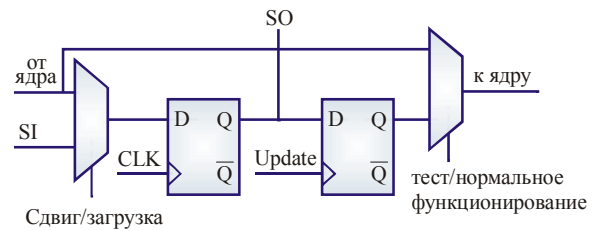


Рис. 3. Ячейка регистра граничного сканирования

### Ad-hoc технологии тестирования цифровых систем

В целях минимизации времени тестирования модулей (IP-cores) в [16] предлагается существенное дополнение к IEEE Boundary Scan стандарту. Его суть заключается в следующем. Каждый IP-core имеет собственный BIST, который генерирует псевдослучайные входные последовательности. Это дает возможность тестировать IP-cores параллельно. При использовании детерминированного теста, хранящегося в памяти, вводятся периоды простоя в целях избежания конфликтных ситуаций. Для подачи тестов к модулям используется шина AMBA (Advanced Microcontroller Bus Architecture). Авторами предложена методология оптимальной комбинации псевдослучайного и детерминированного тестов для всех IP-cores системы. Стандарты следует рассматривать как базовые средства, которые могут быть успешно дополнены ad-hoc технологиями для решения сложных задач синтеза тестов, моделирования неисправностей, верификации и диагностирования некорректностей и дефектов. Далее предлагаются специализированные технологии тестирования сложных цифровых систем, развивающие гибридную методологию [17], сочетающую IEEE стандарты и эвристики. Система тестирования, представленная на рис. 4, ориентирована на использование шины в качестве средства доставки проверяющих последовательностей от test- процессора и реакции

проверяемого компонента обратно. При такой архитектуре, используя Decoder, можно выполнять тестирование N IP-cores или интегральных схем, а также самого ARM-процессора. Подготовка к диагностическому эксперименту заключается в предварительном создании тестов и получении эталонных реакций для всех модулей цифровой системы. Диагностическое обеспечение (тест, реакции и алгоритмы поиска дефектов) сохраняется в одноименном процессоре. При выполнении самой процедуры тестирования управление передается от ARM-процессора к test-процессору с прерыванием выполнения основных функций цифровой системы. Преимущества такой системы заключаются в простоте конфигурации, возможности диагностирования с точностью до микросхемы (IP-core) и высоком быстродействии, определяемом тактовой частотой и разрядностью шины:

$$\varphi = \frac{k_t \times n_t}{f \times r}, \quad (1)$$

где  $k_t, n_t, f, r$  – число строк и столбцов (размерность) теста, тактовая частота и разрядность шины передачи данных соответственно. Недостаток данной архитектуры заключается в невозможности определить место и вид дефекта внутри микросхемы.

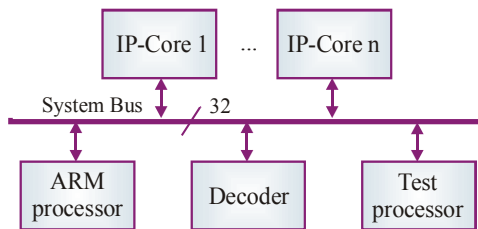


Рис. 4. Шинная архитектура тестирования цифровой системы

В целях устранения данного недостатка на рис. 5 предложена структура, сочетающая преимущества шинной организации обмена данными между IP-cores с высокой глубиной диагностирования, имеющей место в стандарте IEEE 11.49. Платой за качество диагноза является существенное замедление тестового эксперимента, длительность которого определяется следующей формулой:

$$\varphi = \frac{k_t \times n_t}{f \times r \times L_{bsr}}, \quad (2)$$

где  $L_{bsr}$  – длина boundary scan регистра тестируемого IP-core. Следует заметить, что такой подход ориентирован на проверку функциональностей компонентов системы и не рассматривает вопросы тестирования соединений между IP-cores, которые здесь отсутствуют.

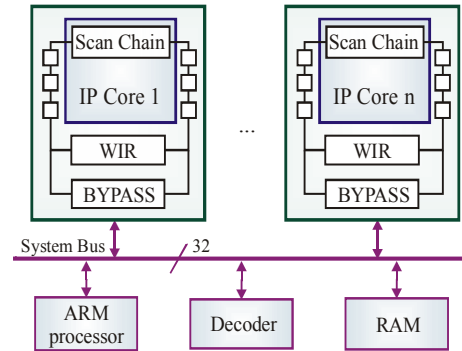


Рис. 5. Шинная архитектура и Boundary Scan стандарт

Следующая архитектура, представленная на рис. 6, объединяет преимущества шинной организации, Boundary Scan стандарта и встроенных средств тестирования BIST. Это дает возможность значительно уменьшить время выполнения тестового эксперимента при высокой глубине диагностирования дефекта в пределах одного компонента системы:

$$\varphi = \frac{\eta(k_t \times n_t)}{f \times r \times L_{bsr}}, \quad (3)$$

где  $\eta$  – коэффициент, учитывающий часть детерминированного теста в общей длине входных наборов (плюс псевдослучайные векторы) для проверки рассматриваемого IP-core.

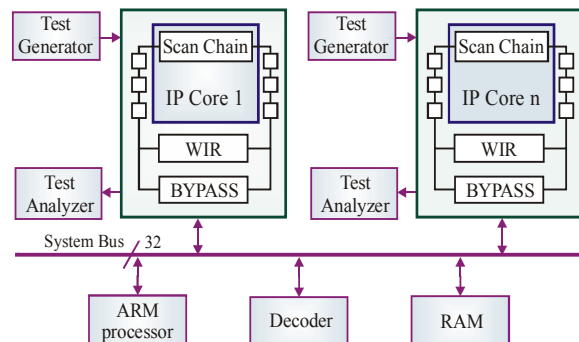


Рис. 6. Шинная архитектура, BIST и Boundary Scan стандарт

Естественно, что для верификации всех компонентов цифровой системы в целях диагностирования необходимо выполнить  $N$  таких экспериментов.

Общая структура процесса тестирования сложной иерархической цифровой системы в целях проверки работоспособности и установления места и вида дефекта имеет вид, который представлен на рис. 7.

Принципы организации системы тестирования:

1. Использование наиболее подходящих IEEE стандартов [8 – 11] для верификации компонентов рассматриваемого уровня.

2. Установление диагноза заданной глубины в автоматическом режиме, без использования условных алгоритмов поиска дефектов.

3. Диагноз выполняется при условии, что только один компонент может быть неисправным.

4. После восстановления работоспособности процедура тестирования начинается всегда с верхнего уровня иерархии к нижнему – нисходящее диагностирование.

5. Процедура тестирования может начинаться с любого уровня иерархии и заканчиваться на уровне, желаемом разработчиком.

Тест-процессор регламентирует подачу входных последовательностей необходимого формата с заданными свойствами для конкретных компонентов рассматриваемого уровня иерархии  $\{C, B, P, M\} \in F$ . Каждый компонент имеет собственный стандарт представления теста для проверки и диагностирования дефектов заданного класса. Тестирование поддерживает автоматический режим поиска дефекта с высокой глубиной диагностирования, до уровня группы вентилях или фрагмента программного кода. По желанию пользователя, процесс может быть остановлен при достижении заданной глубины диагностирования. После восстановления работоспособности процедура диагностирования выполняется с верхнего уровня организации диагностического эксперимента.

Входные данные для установления диагноза – состояния разрядов boundary scan регистра, оформленные в таблицу реакций  $T = [T_{tr}]$ ,  $t = \overline{1}, p; r = \overline{1}, q + n$  размерностью  $p \times n$ ,  $p$  – число тест-векторов,  $n$  – количество разрядов разрядов boundary scan регистра.

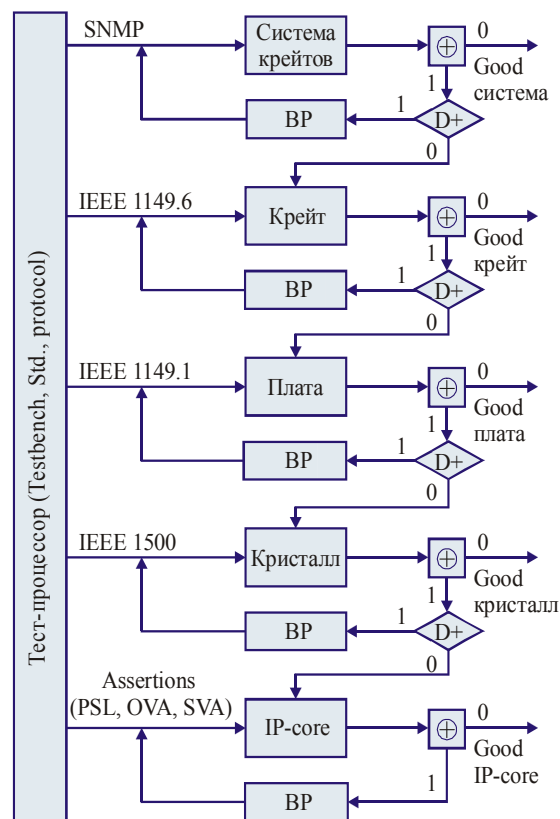


Рис. 7. Структура процесса тестирования сложной цифровой системы (BP – восстановление работоспособности; D+ – глубина диагноза удовлетворяет пользователя)

Результатом анализа каждого тест-вектора из таблицы реакций является вектор экспериментальной проверки  $Y = (Y_1, Y_2, \dots, Y_i, \dots, Y_n)$ , позволяющий идентифицировать место и тип дефекта путем применения следующих уравнений:

$$D^s = \left( \bigcap_{\forall i(Y_i=1)} D_i \right) \setminus \left( \bigcup_{\forall i(Y_i=0)} D_i \right); \tag{4}$$

$$D^m = \left( \bigcup_{\forall i(Y_i=1)} D_i \right) \setminus \left( \bigcup_{\forall i(Y_i=0)} D_i \right),$$

где  $(D_i, D^s; D^m)$  – соответственно: ошибки или дефекты, поведение которых описано предикатом

$Y_i = L_i(X_i)$ ; совокупность одиночных дефектов, обнаруживаемых тест-вектором; множество кратных дефектов, для формирования которого используется процедура, определяемая вторым уравнением в (4). Уравнения (4) позволяют существенно повысить глубину диагностирования, если тестом покрыты все неисправности и возможные некорректности. Например, имеется тест, формирующий следующую таблицу покрытия дефектов  $D_i$ :

$$\begin{array}{cccc|cc}
 D_1 & D_2 & D_3 & D_4 & Y^1 & Y^2 \\
 \hline
 1 & . & 1 & 1 & 1 & 1 \\
 . & 1 & 1 & . & 1 & 1 \\
 1 & 1 & . & . & 0 & 1 \\
 . & 1 & . & 1 & 0 & 0
 \end{array} \quad (5)$$

Применение первого уравнения из (4) для вектора  $Y^1$  дает результат:

$$D^s = [(D_1 \cup D_3 \cup D_4) \cap (D_2 \cup D_3)] \setminus [(D_1 \cup D_2) \cup (D_2 \cup D_4)] = D_3,$$

который позволяет исключить из списка подозреваемых дефекты  $D_1, D_2, D_4$ . Если рассматривается предположение наличия в проекте кратных дефектов, то, например для  $Y^2$  (5) необходимо использовать второе уравнение из (4), поскольку первое не определяет дефектов вообще:

$$D^m = [(D_1 \cup D_3 \cup D_4) \cup (D_2 \cup D_3) \cap (D_1 \cup D_2)] \setminus [(D_2 \cup D_4)] = (D_1 \cup D_3).$$

Предложенный подход дает возможность всегда обнаруживать одиночный дефект и, по меньшей мере, один из кратных, присутствующий в списке  $D^m$ .

### Заключение

В ходе исследований рассмотрены международные стандарты общества IEEE, пригодные для тестирования и диагностирования программно-технических комплексов и электронных систем широкого профиля: начиная от компонентов внутри интегральной схемы, заканчивая сложными систе-

мами, состоящими из печатных плат, крейтов. Рассмотренные стандарты IEEE Std 1149.1, 1149.4, 1149.6 и 1500 предоставляют широкие возможности для решения задач: тестирование функциональности компонента; тестирование соединений на печатной плате; тестирование функционирования системы на печатной плате; тестирование системы, состоящей из нескольких печатных плат.

Объектами тестирования являются:

- аналоговые компоненты;
- цифровые компоненты;
- компоненты смешанного типа;
- дискретные элементы;
- печатные платы, состоящие из вышперечисленных компонентов.

Рассмотрен блочно-иерархический метод тестирования и диагностирования программно-технических комплексов, который заключается в представлении сложной системы в виде иерархии более простых блоков.

Предложен метод диагностирования дефектов в сложных цифровых системах и ПТК на основе использования таблиц неисправностей и процедуры ее теоретико-множественного анализа. Таким образом, технологии граничного сканирования позволяют разработчикам быть на уровне современных требований и тенденций при проектировании, верификации, отладке, изготовлении, тестировании, внутрисхемном программировании и эксплуатации различных систем.

Соответствие конечного программного или аппаратного продукта международным стандартам IEEE – залог успешного продвижения на мировом рынке.

Научная новизна полученных результатов заключается в использовании ad hoc технологий, объединяющих существующие стандарты тестирования со специальными схмотехническими решениями, позволяющими в несколько раз уменьшить время



тестирования и диагностирования сложных цифровых изделий.

Практическая значимость заключается в возможности использования предложенной модели процесса диагностирования к проектированию сложных иерархических цифровых систем, управляющих критическими технологиями.

### Литература

1. Grant Martin. The Reuse of Complex Architectures // IEEE Design and Test of Computers. – November-December 2002. – P. 4-6.

2. Rajesh K. Gupta, Yervant Zorian. Introducing Core-Based System Design // IEEE Design & Test of Computers. – November-December 1997. – P. 15-25.

3. Yervant Zorian. What is Infrastructure IP? // IEEE Design & Test of Computers. – May-June 2002. – P. 5-7.

4. Benini L. and Micheli G.D. Networks on chips: A new soc paradigm // IEEE Computer. – 2002. – Vol. 35, No. 1. – P. 70-78.

5. Безопасность атомных станций. Информационные и управляющие системы / М.А. Ястребенецкий, В.М. Васильченко и др.; Под ред. М.А. Ястребенецкого. – К.: Техніка, 2004. – 472 с.

6. Горелик А.Х., Елисеев В.В., Орловский В.А. Опыт разработки новых и поэтапной реконструкции действующих информационно-вычислительной системы энергоблоков с реактором ВВЭР-1000 // Ядерная и радиационная безопасность. – 2005. – № 1. – С. 91-96.

7. Система контроля нейтронного потока для реактора ВВЭР-1000: обеспечение и оценка безопасности / В.В. Елисеев, Г.Ю. Пивоваров, А.С. Набатов и др. // Ядерная и радиационная безопасность. – 2005. – № 1. – С. 51-65.

8. Елисеев В.В., Ларгин В.А., Пивоваров Г.Ю. Программно-технические комплексы АСУ ТП:

Учебное пособие. – К.: Издательско-полиграфический центр «Киевский университет», 2003. – 429 с.

9. IEEE P1500/D11. January 2005. Draft Standard Testability Method for Embedded Core-based Integrated Circuits. New York. 2005. – 138 p.

10. Yervant Zorian. Advances in Infrastructure IP // IEEE Design & Test of Computers. – May-June 2003. – P. 49-56.

11. Yervant Zorian. Test Requirements for Embedded Core-Based Systems and IEEE P1500 // In Proceedings IEEE International Test Conference (ITC). Washington, DC. – November 1997. – IEEE Computer Society Press. – P. 191-199.

12. Abramovici M., Breuer M.A., Friedman A.D. Digital systems testing and testable design. – Computer Science Press. 1998. – 652 p.

13. IEEE Std 1149.1-2001. Standard Test Access Port and Boundary-Scan Architecture. – New York. 2001. – 208 p.

14. IEEE Std 1149.4-1999. IEEE Standard for a Mixed-Signal Test Bus. – New York., 2000. – 84 p.

15. IEEE Std 1149.6-2003. Standard for Boundary-Scan Testing of Advanced Digital Networks. – New York. 2003. – 139 p.

16. Gert Jervan, Petru Eles, Zebo Peng, Raimund Ubar, Maksim Jenihhin. Test Time Minimization for Hybrid BIST of Core-Based Systems // Proceedings of the 12th Asian Test Symposium (ATS'03). – P. 318-323.

17. Papachristou C. A., Martin F. and Nourani M. Microprocessor based testing for core-based system on chip // In Proceedings of the 36th ACM/IEEE conference on Design automation conference. – ACM Press, 1999. – P. 586-591.

*Поступила в редакцию 24.01.2006*

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