Министерство образования и науки РФ

ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ

Кафедра проектирования и производства электронно-вычислительных средств

УТВЕРЖДАЮ

Руководитель программы магистратуры

Автоматизация и системы управления

д-р техн. наук

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. В. Роженцов

« \_\_\_\_\_\_\_\_\_ » ноября 2015 г.

ОТЧЕТ

О НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

по теме:

Разработка алгоритмов внутрисистемного тестирования для встроенных систем диагностики

(промежуточный – за 3 триместр)

Научный руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В. В. Кошкин

подпись, дата

Магистрант группы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А. И. Тораев

подпись, дата

Йошкар-Ола 2016

1. ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Тема магистерской диссертации: разработка алгоритмов внутрисистемного тестирования для встроенных систем диагностики.

Аннотация. В рамках магистерской диссертации рассмотрены способы внутрисистемного тестирования для встроенных систем диагностики. Проведена работа по анализу и разработке одного из способов для внедрения в системы приема информации со спутников; системы наблюдения за воздушным пространством (гражданская авиация и др.); системы прямого управления (сопровождения) целей; системы автоматизированного управления быстрыми процессами (химия, ядерная физика). Цель работы алгоритма заключается в дополнительной проверке контрольных данных.

Объект исследования: цифровой автомат.

Предмет исследования: алгоритмы внутрисхемного тестирования для цифровых автоматов с повышенной надежностью.

Научная задача: разработка метода внутрисхемного тестирования для цифровых автоматов с повышенной надежностью.

1. Разработка математической модели предмета исследования

Разработка математической модели производится для модуля диагностирования.

Моделью называется представление объекта или системы в некоторой форме, отличной от формы их реального существования. К математическим моделям (ММ) относятся модели, в которых для представления процесса используются математические символы. ММ можно представить в виде множества величин, описывающих процесс функционирования реального объекта.[1]

Адекватное математическое моделирование технического состояния систем является важнейшим компонентом их диагностирования.

Модуль диагностирования (Мд) может быть представлен как динамическая система, состояние которой в момент времени t определяется значениями входных, внутренних и выходных координат (параметров). Частным является случай, когда состояние модуля не зависит от времени. [2] Другими словами, функционирование Мд подобно принципам функционирования, положенным в основу теории цифровых автоматов. Поэтому для написания математической модели используется язык описания цифровых автоматов.

Термин «автомат», как правило, используется в двух аспектах. С одной стороны, автомат – это устройство, выполняющее некоторые функции без непосредственного участия человека. В этом смысле мы говорим, что ЭВМ – автомат, так как после загрузки программы и исходных данных ЭВМ решает заданную задачу без участия человека. С другой стороны, термин «автомат» как математическое понятие обозначает математическую модель реальных технических автоматов. В этом смысле автомат представляется как «черный ящик», имеющий конечное число входов и выходов и некоторое множество внутренних состояний Q = {q (t), q (t), ..., q (t)}, в которые он под действием входных сигналов переходит скачкообразно, т. е. практически мгновенно, минуя промежуточное состояние. Конечно, это условие не выполняется в реальности, так как любой переходный процесс длится конечное время. [3]

Для построения математической модели Мд применим модель распознающего автомата с принципами модели микропрограммного автомата и абстрактный автомат модели дискретного преобразователя Глушкова В.М.

При рассмотрении моделей распознавания очень часто применяется понятие распознающей машины, под которой понимается некоторая система. На вход этой системы поступают объекты для распознавания, а на выходе появляются распознанные объекты, отнесенные к определенному классу. В большинстве задач распознавания имеется 2 этапа: обучение распознаванию на заданном количестве эталонных образов, принадлежность которых к определенному классу известна, и собственно распознавание, когда предъявляется реализация (объект) с неизвестной принадлежностью и требуется определить, к какому классу относится объект. Как правило, имеется набор классов или алфавит классов (образов):

A = {A1 , A2 , ..., Ai , ..., Am },

где Ai – отдельный i-класс; m – общее число классов.

Если m = 1, то никакого распознавания ненужно. Очень часто рассматривается задача отнесения объекта к одному из двух (m = 2) классов. Каждый класс в алфавите образов может быть представлен некоторым количество объектов или реализацией. Совокупность различных реализаций для всех классов образует множество возможных реализаций:

B = {b1 , b2 , ..., bj , ..., bT }.

В большинстве случаев Т – конечно и T >> m.

Совокупность признаков для данного алфавита А:

X = {x1 , x2 , ..., xk , ..., xN }.

Практически числовые значения признаков изменяются в некоторых пределах. Каждый признак xk может принимать одно значение из совокупности:

Xk = { x11, x22, ..., xkp, ..., xkR}.

Каждая конкретная реализация bj задается совокупностью значений признаков:

b = { x1j1 , x 2j2 , ..., xkjk , ..., xNJN },

которая называется описанием реализаций[3].

В данном случае в модуль диагностирования для распознавания поступают ответные реакции ОД R = {r1.1(t3.1), r2.1(t3.1) … rj.1(ti.1)} на стимулирующее воздействиеS = { qT1 (t3), qT2 (t4),… qTj(ti) }, которые сравниваются с эталонными и относятся к одному из классов «совпадение» или «несовпадение», из которых формируется выходной файл Z2= { z12, z22,… zj2}, пересылаемый в БД.

Объектом изучения в абстрактной теории автоматов являются абстрактные автоматы вместе с реализуемыми ими отображениями и событиями. Дискретный преобразователь представляет собой абстрактный автомат А, функционирующий по соответствующим законам в дискретном времени.

Абстрактный автомат А модели дискретного преобразователя Глушкова В.М., задается совокупностью шести объектов:

А = (Х, Y, Q, q0, δ, λ),

Где Х – конечное множество входных сигналов, называемое входным алфавитом автомата;

Y – конечное множество выходных сигналов, называемое выходным алфавитом автомата;

Q – произвольное множество, называемое множеством состояний автомата;

q0 – элемент из множества Q, называемый начальным состоянием автомата;

δ(q, x) и λ(q, x) – две функции, задающие однозначные отображения множества пар (q, x), где q ∈Q и x ∈X, в множества Q и Х. Функция δ(q, x) называется функцией переходов автомата, а функция λ(q, x) – функцией выходов, либо сдвинутой функцией выходов.

Автомат, заданный функцией выходов, называется автоматом первого рода; автомат, заданный сдвинутой функцией выходов, – автоматом второго рода.

Абстрактный автомат функционирует в дискретном времени, принимающем

Целые неотрицательные значения t = 0, 1, 2, … В каждый момент t этого времени он находится в определенном состоянии q(t) из множества Q состояний автомата, причем в начальный моментов времени t = 0 автомат всегда находится в своем начальном состоянии q т. е. q(0) = q .

В момент времени t, отличный от начального, автомат способен воспринимать входной сигнал x(t) – произвольную букву входного алфавита Х и выдавать соответствующий выходной сигнал (t) – некоторую букву выходного алфавита Y.

Закон функционирования абстрактного автомата первого рода задается уравнением: q(t) = д(q(t−1),x(t)), y(t) = л(q(t−1),x(t)), где t 1, 2, ..., а абстрактного автомата второго рода – уравнением: q(t) = δ(q(t −1), x(t)), y (t)= λ(q(t), x(t)), где t 1, 2, ...

В абстрактной теории автоматов входные и выходные сигналы рассматриваются как буквы (символы) двух фиксированных для данного автомата алфавитов. Абстрактная теория изучает те переходы, которые претерпевает автомат под воздействием входных сигналов в дискретные моменты времени, и те выходные сигналы, которые он при этом выдает. [3]

На основе терминологии и способов описания цифровых автоматов перечисленных типов, создается математическая модель модуля диагностирования, или, другими словами математическая модель (ММ) процедуры диагностирования, реализуемой модулем диагностирования (Мд). Модуль диагностирования так же, как дискретный автомат описывается определенной совокупностью объектов: Мд = (...). Модуль реализует метод внутрисхемной эмуляции и работает по принципу микропрограммного управления, что и определяет правила проведения процедуры диагностирования.

В начальный момент времени t0 модуль диагностирования находится в состоянии ожидания управляющих воздействий от БД.

Множество состояний модуля диагностирования обозначим Q, элементами этого множества являются элементы q(t), которые являются состояниями модуля диагностирования, в которые он переходит в результате расшифровки определенных входных воздействий в определенные моменты времени t.

Т.е., в начальный времени t0 модуль находится в состоянии ожидания q0.

В момент времени t, отличный от начального, модуль диагностирования принимает по последовательной шине входную команду от блока диагностики (БД) на реализацию блокировки процессора ОД, (для возможности проведения дальнейшего тестирования методом внутрисхемной эмуляции), а так же тест – команды для осуществления диагностики функциональных элементов ОД (ОЗУ, ПЗУ, портов ввода/вывода) и команды, осуществляющие процедуру проверок. Первой обрабатывается команда реализации блокировки процессора ОД, которая является входным воздействием Vб., где индекс «б.» отражает назначение воздействия – для блокировки процессора ОД. Это входное воздействие представляет собой команду для реализации блокировки - Кб. Тогда Vб. = {Кб.}. Управляющее устройство модуля диагностирования расшифровывает эту команду, в результате чего модуль диагностирования переходит в новое состояние – состояние выполнения команды Кб. Переход в новое состояние выполнения команды Кб осуществляется в момент времени t1. В течение времени, входящего в промежуток t ϵ [t1; t2), модуль выполняет команду Кблок., а в момент времени t2 переходит в новое состояние для выполнения следующей команды. Состояние, в котором находится модуль при выполнении команды Кб обозначается как qб(t1). Можно так же сказать, что происходит преобразование команды Кб в состояние qб.(t1) модуля диагностирования и записать: Кб. → qб.(t1). Во время состояния qб.(t1) модуль выполняет команду Кб., т.е. формирует и отправляет управляющий сигнал блокировки на определенный вход процессора ОД. Т.е., состояние модуля диагностирования является функцией выполняемой им команды. Но для каждой команды (входного воздействия) функция будет своя, поэтому ѱ = {φ1, φ2… φn}, где n – общее число состояний модуля диагностирования и, соответственно, команд. Поэтому, qб. (t1) = φ1(Кб.). А т.к. функция φ1 преобразует входную команду в сигнал блокировки hб., записываем: qб. (t1) = φ1(Кб.) = hб.

Отправленный сигнал блокировки процессора ОД изменяет состояние ОД, в результате чего в модуль диагностирования из ОД приходит сигнал подтверждения блокировки и модуль может приступать к реализации следующей команды. Т.е., изменение состояния ОД будет служить подтверждением блокировки, или, изменение состояния Мд при выполнении команды Кб приведет к изменению состояния ОД, что эквивалентно приходу сигнала подтверждения блокировки. Поскольку ответная реакция ОД происходит в момент времени, входящий в интервал t ϵ [t1; t2), (из-за того, что состояние Мд при этом не меняется), но позже t1, обозначим данное состояние ОД qод.б (t1.1). Поэтому, можно записать: qблок. (t1) → qод.б (t1.1) или, φ1(Кб.) = qод.б. (t1.1). Состояние ОД qод.б (t1.1) характеризуется для модуля диагностирования приходом сигнала подтверждения блокировки от ОД.

1. Для участия в научной конференции был подготовлен следующий материал.

**Необходимость тестирования при производстве техники.**

При производстве цифровой техники прежде всего нужно четко определить задачи, которые должно выполнять тестирование на предприятии. Если взять функциональную проверку как основу производственного тестирования, это означает проверку работы изделия без учета времени проверки, регулировки, локализации дефектов и ремонта. Тем не менее, по мировым стандартам производственное тестирование должно обеспечивать выход на рынок годного продукта в кратчайшие сроки и с минимальными затратами. Кроме того, тестирование должно служить гарантом и индикатором качества произведенной продукции. В сущности, это входит в задачи производства в целом. Из этих соображений и необходимо подходить к политике тестирования на предприятии.

**Тенденции развития тестирования.**

Первая тенденция – это усложнение самого функционального тестирования, увеличение числа проверок, их трудоемкости и, как следствие, увеличение объема инструкций по проверке и регулировке. Сюда же нужно прибавить трудоемкость по созданию самих тестовых программ, оснастки и методик проверки. Современный модуль цифровой обработки сигналов может содержать несколько тысяч цепей и десятки тысяч паяных соединений. Один из методов производственного тестирования цифровой техники – периферийное сканирование JTAG – позволяет получать карту найденных дефектов за 1–2 минуты. Функциональная проверка всех узлов такого модуля при хорошей организации может длиться несколько дней. Однако это только проверка, которая не указывает, какие дефекты имеются на плате и сколько времени понадобится на их поиск (недели, месяцы).

Вторая тенденция – это использование на производстве методик проверки, которые применялись разработчиками для создания и отладки прототипов изделий. Такой подход обусловлен нехваткой времени на создание производственных методик и программ. Вследствие этого возникают такие казусы, как проверка на производстве абсолютно лишних параметров модуля, которые необходимо проверять только при квалификационных испытаниях. Очень часто можно встретить эмуляцию процессоров, применяемых для тестирования, тогда как такая эмуляция, в сущности, должна использоваться только для отладки опытных образцов. Отсутствуют унифицированные рабочие места, количество оснастки превышает все допустимые пределы. Чаще всего тестовые программы и методики в этом случае "обтачиваются" на уже выпускаемом серийном продукте с многократными изменениями КД.

**Необходимость автоматизированного тестирования.**

Важным аргументом использования автоматических средств тестирования является так называемое "правило десяти", согласно которому стоимость нахождения дефекта на каждой последующей стадии тестирования (структурный тест, функциональная проверка, тест блока и т.д.) умножается на 10. Это правило впервые было сформулировано в 1982 году Бренданом Дэвисом в книге "Экономика автоматического тестирования".

**Преимущества JTAG-тестирования.**

1. В основу стандарта положена идея внедрение в компоненты цифрового устройства средств, обеспечивающих унифицированный подход к решению следующих задач:

* Тестирование связей между интегральными схемами, после того, как они были смонтированы на печатной плате или другой основе;
* Наблюдение за работой компонент без вмешательства в их нормальную работу, или непосредственное управление одним или более компонентом;
* Обеспечение стандартизованного доступа к произвольным средствам самотестирования, встраиваемым в БИС;

При этом в общем виде сам процесс тестирования выглядит следующим образом:

Тестируемая плата с расположенными на ней БИС подключается, через последовательный канал передачи данных (JTAG интерфейс), к некоторому ведущему устройству. Ведущее устройство, используя возможности предоставляемые JTAG, решает задачи связанные с диагностикой тестируемого устройства, локализации неисправностей, загрузкой конфигураций PLD и т.п.

Таким образом, стандарт JTAG определяет:

* интерфейс, через который осуществляется обмен тестовыми инструкциями и данными между ведущим устройством и встроенными средствами тестирования (TAP — Test Access Port) ;
* минимальный набор средств тестирования, встраиваемых в БИС (средства поддержки метода Граничного Сканирования); [4]

2. Любая схемная не-JTAG структура, состоящая из одного или более элементов (необязательно ИС), цифровых, аналоговых или пассивных, может рассматриваться как кластер, если обеспечены два непременных схемных условия: JTAG-управляемость входов этой структуры и JTAG-наблюдаемость ее выходов. Часто применяемые схемные не-JTAG элементы, пригодные для кластерного тестирования, обычно описываются моделями в той или иной форме в библиотеках кластеров и вызываются в программу построения теста кластера по имени или псевдониму. [5]

3. Как правило, ведущим устройством является персональный компьютер, оснащенный соответствующим программным обеспечением. Подключение к ведомому устройству осуществляется через параллельный или последовательный порт, или через плату расширения. Впрочем, если задачи решаемые через JTAG достаточно просты, как загрузка начальной конфигурации в PLD, или тестирование общей работоспособности устройства после включения питания, то роль ведущего устройства может играть простейший микропрограммный автомат построенный на базе ПЗУ и счетчика.[4]

3.Подготовка статьи для публикации в научном журнале.

Информация, изложенная выше, так же планируется для публикации после некоторой корректировки.

Список использованной литературы:

1. Стешина Л.А. Идентификация и диагностика систем. Конспект лекций. Учебное пособие. Йошкар –Ола: МарГТУ, 2004-78с ил.
2. МАТЕМАТИЧЕСКАЯ МОДЕЛЬ ТЕХНИЧЕСКОЙ ДИАГНОСТИКИ ГТД [Электронный ресурс]. URL: <http://econf.rae.ru/pdf/2009/08/Al-Hatim.pdf> (дата обращения 28.04.2016)
3. Н. Г. Захаров, В. Н. Рогов. Синтез цифровых автоматов. Учебное пособие. Ульяновск: УлГТУ, 2003.
4. JTAG – основы и немного теории [Электронный ресурс]. URL: <http://portal-ed.ru/index.php/testirovanie-i-kontrol/92-jtag-osnovy-i-nemnogo-teorii> (дата обращения 28.04.2016)
5. Городецкий Ами «JTAG-тестирование кластеров» Компоненты и технологии №1’2010 стр.38-39.

Оценка промежуточной аттестации за 3-й триместр:

- теоретическая часть (от 30 до 40 баллов) - \_\_\_\_\_\_\_\_ баллов;

- практическая часть (от 10 до 20 баллов) - \_\_\_\_\_\_\_\_ баллов;

- дополнительные баллы (от 0 до 5) - \_\_\_\_\_\_\_\_ баллов

за \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Научный руководитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.В.Кошкин

подпись, дата