На правах рукописи

НИКОЛАЕВА Екатерина Александровна

АЛГОРИТМЫ СИНТЕЗА ЛЕГКО ТЕСТИРУЕМЫХ КОМБИНАЦИОННЫХ СХЕМ И ТЕСТОВ ДЛЯ КРАТНЫХ КОНСТАНТНЫХ НЕИСПРАВНОСТЕЙ И НЕИСПРАВНОСТЕЙ ЗАДЕРЖЕК ПУТЕЙ

05.13.01 – Системный анализ, управление и обработка информации
(в отраслях информатики, вычислительной техники и автоматизации)

Автореферат диссертации
на соискание ученой степени кандидата технических наук

Томск – 2011

Работа выполнена в ГОУ ВПО «Томский государственный университет» на кафедре программирования

Научный руководитель: доктор технических наук, профессор
Матросова Анжела Юрьевна

Официальные оппоненты: доктор технических наук, профессор
Сперанский Дмитрий Васильевич

кандидат технических наук, доцент
Липский Валерий Борисович

Ведущая организация: ГОУ ВПО «Национальный исследовательский Томский политехнический университет»

Защита состоится: 12 мая 2011 г. в 10.30 час. на заседании диссертационного совета Д 12.267.12 при ГОУ ВПО «Томский государственный университет» по адресу: 634050, г. Томск, пр. Ленина, 36, 2 уч. корпус, ауд. 212б.

С диссертацией можно ознакомиться в Научной библиотеке ГОУ ВПО «Томский государственный университет»

Автореферат разослан 8 апреля 2011 г.

Ученый секретарь
диссертационного совета,
к.ф.-м.н., доцент

П.Ф. Тарасенко

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность проблемы. Область применения дискретных устройств управления постоянно расширяется, они становятся все более сложными. Увеличивающаяся сложность и значимость дискретных устройств и систем требуют высокой надежности, что приводит к большим затратам на разработку и реализацию методов их тестирования. Контролепригодное проектирование позволяет снизить эти затраты, так как ориентировано одновременно на обеспечение функционирования устройства и решение проблемы его тестирования. Контролепригодные свойства схем, то есть существование для них достаточно коротких тестов высокого качества, могут быть заложены на уровне описания поведения схем. При этом требуется, чтобы построенная схема сохраняла это описание. Речь идет о сохранении системы ДНФ, являющейся заданием на синтез. Проблема исследования контролепригодных свойств схем, сохраняющих системы ДНФ, является актуальной.

Переход к наноразмерам транзисторов, повышение скорости обработки информации, снижение напряжения питания приводят к необходимости тестирования задержек путей схемы. Задержка смены состояния каждого элемента, проходимого на некотором пути схемы, и линии связи может быть незначительной и потому не обнаружимой проверяющими тестами для одиночных и даже кратных константных неисправностей. Однако задержка пути в целом на высокой рабочей частоте функционирования схемы может приводить к искажению функционирования. Такие неисправности должны быть обнаружены. Это значит, что проверяющие тесты, ориентированные только на обнаружение одиночных и кратных константных неисправностей не могут считаться качественными для современных дискретных устройств управления, если они не проверяют еще и неисправности задержек путей. Исследование контролепригодных свойств схем, ориентированное на обнаружение неисправностей задержек путей, наряду с константными неисправностями, является актуальной проблемой обеспечения надежности функционирования современных дискретных устройств.

Цель работы. Целью диссертационной работы является исследование контролепригодных свойств схем, сохраняющих системы ДНФ. Для ее реализации были рассмотрены существующие методы синтеза схем, использующие в качестве задания на синтез либо системы ДНФ, либо системы *BDD* (*Binary Decision Diagram*)-графов. Для методов синтеза по системе ДНФ, обеспечивающих сохранение систем ДНФ, были предложены эвристики, ориентированные на сокращение аппаратурных затрат, а для методов синтеза по системе *BDD*-графов сформулированы требования к покрытию графов программируемыми логическими блоками (ПЛБ), обеспечивающие сохранение систем ОДНФ. Для схем, полученных покрытием систем *BDD*-графов программируемыми логическими блоками, разработаны алгоритмы построе-

ния проверяющих тестов для кратных константных неисправностей на полюсах логических элементов схем и неисправностей задержек путей этих схем.

Методы исследования. В работе используется аппарат дискретной математики, в частности, алгебра логики, теория автоматов и теория графов. Эффективность разработанных методов подтверждается компьютерными экспериментами.

Научная новизна.

– Предложены эвристики для сохраняющих систему ДНФ двухуровневого и многоуровневого факторизационных методов синтеза, ориентированные на сокращение аппаратурных затрат получаемых комбинационных схем.

– Для схем, полученных покрытием системы *BDD*-графов программируемыми логическими блоками (ПЛБ) и сохраняющих системы ОДНФ, предложен алгоритм построения проверяющего теста для кратных константных неисправностей на полюсах ПЛБ, не требующий перечисления всевозможных кратных неисправностей и получающийся расширением проверяющего теста для одиночных неисправностей.

– Для схем, полученных покрытием системы *BDD*-графов программируемыми логическими блоками (ПЛБ) и сохраняющих системы ОДНФ, предложен алгоритм построения проверяющего теста, обнаруживающего одиночные робастные неисправности задержек путей в схеме, не требующий введения дополнительных входов и ориентированный на сокращение длины теста. Установлено, что неисправность задержки каждого пути схемы проявляется как робастная и, следовательно, проверяющий тест, обнаруживающий одиночные робастные неисправности задержек путей, обнаруживает всевозможные кратные неисправности задержек путей.

Достоверность полученных результатов. Все научные положения и выводы, содержащиеся в диссертации, доказаны с использованием аппарата дискретной математики. Эффективность предложенных методов подтверждена компьютерными экспериментами.

Практическая значимость работы. Разработанные алгоритмы реализованы в виде программ, которые могут быть включены в существующие системы САПР с целью получения средствами САПР контролепригодных схем с одновременным построением для них проверяющих тестов высокого качества.

Реализация полученных результатов. Исследования, результаты которых изложены в диссертации, проводились в рамках следующих проектов.

– «Участник молодежного научно-инновационного конкурса 2008» («УМНИК») по направлению Информационные технологии.

– Государственный контракт на выполнение научно-исследовательских работ для государственных нужд № П1157.

Апробация работы и публикации. Научные результаты, составляющие основу данной работы, обсуждались на заседаниях объединенного семинара

кафедры программирования факультета прикладной математики ТГУ и кафедры информационных технологий в исследовании дискретных структур радиофизического факультета ТГУ.

Результаты работы представлялись на следующих всероссийских и международных научных конференциях:

1. The 22th IEEE international Symposium on Defect and Fault-Tolerance in VLSI System (Rome, Italy, 2007).

2. The 7th East-West Design & Test International Symposium (Lviv, Ukraine, 2008).

3. 7-ая Всероссийская конференция с международным участием «Новые информационные технологии в исследовании сложных структур» (Томск, Россия, 2008).

4. 8-ая Сибирская научная школа-семинар с международным участием «Компьютерная безопасность и криптография» (Омск, Россия, 2009).

5. The 8th East-West Design & Test International Symposium (Moscow, Russia, 2009).

6. The 9th East-West Design & Test International Symposium (St. Petersburg, Russia, 2010).

По результатам выполненных исследований опубликовано 9 печатных работ, в том числе одна из перечня ВАК.

Структура и объем диссертации. Диссертация состоит из введения, 4-х глав, заключения и списка используемой литературы, включающий 95 наименований. Общий объем диссертации составляет 134 страницы текста, включая 33 рисунка и 7 таблиц.

СОДЕРЖАНИЕ РАБОТЫ

Во **введении** обосновывается актуальность задачи, решению которой посвящена диссертация, а также излагаются научная новизна и практическая значимость результатов.

Первая глава диссертации содержит основные понятии из области тестирования и контролепригодного проектирования дискретных устройств. На основании приведенного обзора делается вывод о необходимости проведения исследований в области контролепригодного проектирования.

Во **второй главе** рассматриваются методы синтеза комбинационных схем, обеспечивающие сохранение исходного представления системы булевых функций в виде системы ДНФ или системы ОДНФ, являющегося заданием на синтез. Сохранение системы означает, что полученная по построенной схеме система ДНФ представляется тем же множеством элементарных конъюнкций для каждой функции, что и исходная система-задание на синтез. Речь идет о получении по построенной схеме системы ДНФ в результате подстановок вместо внутренних переменных схемы функций соответствующих элементов, зависящих от входных переменных элементов схемы, последующего раскрытия скобок и опускания инверсий с учетом запрета на склеи-

вание, поглощение конъюнкций, выбрасывание пустых конъюнкций и выбрасывание повторяющихся букв (литер) в конъюнкциях.

Применение к системам ДНФ методов синтеза, позволяющих сохранить эти системы, дает возможность использовать свойства системы ДНФ для построения по ней тестовых наборов для схемы, не обращаясь непосредственно к схеме.

К методам синтеза, сохраняющим систему ДНФ, относятся факторизационные методы синтеза, которые разделяются на двухуровневые и многоуровневые.

Рассмотрим метод многоуровневого факторизационного синтеза, представленный в книге *Murgai R., Brayton R., Sangiovanni-Vincentelli A. «Logic Synthesis for Field Programmable Gate» (Kluver Academic Publishers. 1995. P. 425)* и основанный на выделении ядер и соядер.

Определение 1. ДНФ, в которой выполнены поглощения конъюнкций, будем называть алгебраической ДНФ.

Определение 2. Произведение двух алгебраических ДНФ, D_1 и D_2, называется алгебраическим, если множества переменных этих ДНФ не пересекаются.

Определение 3. Представление ДНФ в виде $D = D_1 D_2 \vee D_3$, где D_1 – делитель, D_2 – частное, D_3 – остаток от деления, называется делением ДНФ D. Если $D_1 D_2$ – алгебраическое произведение, то деление ДНФ D называется алгебраическим.

Если ДНФ нельзя представить в виде: $D=kD_1$, где k – элементарная конъюнкция, то говорят, что D не разложима по конъюнкциям.

Главные делители ДНФ D есть ДНФ D_{11}, …, D_{1r}, такие, что $D = D_{11} K_{21} \vee D_{31}, \ldots D = D_{1r} K_{2r} \vee D_{3r}$, и D_{11}, …, D_{1r} не разложимы по конъюнкциям. Главные делители D_{11}, …, D_{1r} называются ядрами этой ДНФ, а соответствующие им частные (конъюнкции) – соядрами.

Опишем рассматриваемый в упомянутой выше книге метод минимизации системы ДНФ, основанный на выделении ядер и соядер. Пусть выходным полюсам синтезируемой комбинационной схемы сопоставлены ДНФ в виде алгебраических выражений. Извлечём ядра для каждого из алгебраических выражений. Некоторое извлечённое ядро (соядро) C порождает новую систему ДНФ после выполнения следующих шагов:

1. Сопоставление ядру (соядру) C новой переменной u_i.

2. Замена ядра (соядра) C в каждой ДНФ, его содержащей, на эту переменную.

3. Введение нового уравнения $u_i = D_i$ в систему, где D_i, представляющая ядро (или соядро) ДНФ, а u_i – внутренняя переменная.

Промежуточные переменные могут присваиваться также и соядрам. Если число букв в полученной системе меньше, чем в исходной, то исходную сис-

тему ДНФ имеет смысл заменить новой системой. Буквой будем называть переменную вместе с ее знаком инверсии.

Теперь в новой системе выделяем ядра (или соядра) и выбираем в определённом выше смысле лучшее из них; заменяем систему и так далее, пока переход к более простой системе оказывается возможным.

Полученная минимизированная система используется в качестве задания на синтез комбинационной схемы.

При минимизации системы ДНФ, основанной на выделении ядер и соядер, в диссертационной работе предлагается решать две задачи. Первая из них – оценка качества выделенного в очередной рассматриваемой системе соядра (конъюнкции). Под качеством соядра будем понимать число букв, на которое сократится система после его выделения и соответствующих замен вхождений ядер и соядер на переменные, а также введения в систему дополнительных функций. Вторая задача связана с сокращенным выбором соядер-претендентов с целью последующей оценки их качества.

Предлагается две оценки качества выделяемого соядра.

Пусть K – некоторое оцениваемое соядро, K_j – j-ая конъюнкция системы, r – ранг соядра K, r_j – ранг конъюнкции K_j, n – количество различных конъюнкций системы, $N = \{1, \ldots, n\}$ – множество номеров конъюнкций, J – множество номеров конъюнкций, разложимых по K, то есть поглощаемых K, m – количество функций в системе.

Число вхождений конъюнкции в систему будем называть частотой вхождения. Частоту вхождения j-ой конъюнкции в систему обозначим символом k_i. k – частота вхождения выделяемого соядра K. Обозначим через ε_1 число букв в исходной системе.

Преобразуем систему, заменив соядро K промежуточной переменной.

Пусть ε_2 – число букв в системе, в которой соядру K сопоставлена новая переменная u_j и произведена замена каждого вхождения соядра этой переменной. При замене ранги конъюнкций K_j системы, где $j \in J$, уменьшатся на r–1. В систему также будет добавлена новая конъюнкция ранга r. Тогда:

$$\varepsilon_1 = \sum_{j \in N \setminus J} k_j r_j + \sum_{j \in J} k_j r_j \quad \text{и} \quad \varepsilon_2 = \sum_{j \in N \setminus J} k_j r_j + \sum_{j \in J} k_j \left(r_j - r + 1\right) + r$$

Следовательно:

$$\Delta_1 = \varepsilon_1 - \varepsilon_2 = \sum_{j \in J} k_j r_j + \sum_{j \in N \setminus J} k_j r_j - \sum_{j \in N \setminus J} k_j r_j - \sum_{j \in J} k_j \left(r_j - r - k\right) - r =$$

$$= \sum_{j \in J} \left[k_j \left(r_j - r_j + r - 1\right)\right] - r = (r-1) \underbrace{\sum_{j \in J} k_j}_{k} - r = rk - k - r.$$

Итак, величина $\Delta_1 = kr - k - r$ может быть использована в качестве оценки качества соядра K снизу. Уточним оценку с учетом замены ядра. Будем считать, что замена соядра K промежуточной переменной выполнена. В сис-

тему добавлена функция вида $u = K$, где u промежуточная переменная. Значения рангов преобразованных конъюнкций, изменились: $r_j = r_j - r + 1$, где $j \in J$.

Пусть I_2 – множество номеров ДНФ, в которых можно выделить ядро, соответствующее соядру K, и $p = |I_2|$. Пусть n' – количество ДНФ, в которые входит хотя бы одна конъюнкция, содержащая промежуточную переменную, сопоставленную соядру K. Пусть $I_1 = \{1 \ldots m\} \setminus I_2$.

Обозначим символом J_i множество номеров конъюнкций, которые составляют i-ую ДНФ. J_i' – множество номеров конъюнкций, составляющих i-ую ДНФ и содержащих промежуточную переменную, сопоставленную K. J_i'' – множество номеров конъюнкций, составляющих i-ую ДНФ и не содержащих промежуточную переменную, сопоставленную K.

$$\varepsilon_2 = \sum_{i=1}^{m} \sum_{j \in J_i} r_j + r = \sum_{i \in I_1} \sum_{j \in J_i} r_j + \sum_{i \in I_2} \sum_{j \in J_i} r_j + r .$$

Сопоставим промежуточные переменные ядрам (делителям), порожденным рассматриваемым соядром K (частным). Изменения коснутся только функций, номера которых принадлежат I_2. Конъюнкции, соответствующие множеству J_i' ($i \in I_2$) в i-ой функции, будут заменены конъюнкцией вида $u_i u$, где u_i – переменная, сопоставленная очередному ядру. Таким образом, число букв в ДНФ, номер которой (i) содержится в I_2: $2 + \sum_{j \in J_i''} r_j$. В систему также будут добавлены ДНФ, соответствующие ядрам. Число букв в каждой такой ДНФ определяется выражением: $\sum_{j \in J_i'} (r_j - 1)$.

Обозначим символом ε_3 число букв в системе, полученной заменой выделенных ядер внутренними переменными и добавлением ДНФ, сопоставленных ядрам.

$$\varepsilon_3 = \sum_{i \in I_1} \sum_{j \in J_i} r_j + \sum_{i \in I_2} \left(2 + \sum_{j \in J_i''} r_j \right) + \sum_{i \in I_2} \sum_{j \in J_i'} \left(r_j - 1 \right) + r \quad \text{и}$$

$$\varepsilon 2 - \varepsilon 3 = \sum_{i \in I_2} \left(\sum_{j \in J_i'} r_j + \sum_{j \in J_i''} r_j - 2 - \sum_{j \in J_i''} r_j - \sum_{j \in J_i'} \left(r_j - 1 \right) \right) = -2p + \sum_{i \in I_2} |J_i'|$$

Значение параметра k можно выразить следующим образом: $k = \sum_{i=1}^{m} |J_i'| = \sum_{i \in I_1} |J_i'| + \sum_{i \in I_2} |J_i'| = n' - p + \sum_{i \in I_2} |J_i'|$, следовательно:

$$\Delta_2 = \varepsilon_1 - \varepsilon_3 = (\varepsilon_1 - \varepsilon_2) + (\varepsilon_2 - \varepsilon_3) = kr - r - n' - p .$$

Итак, имеем более точную, чем Δ_1, оценку качества соядра K, учитывающую замену соответствующих ему ядер. Построение оценки Δ_2 более

сложно, чем Δ_1, так как требует дополнительного определения значений n' и p.

Задача выделения соядра не может быть решена методом полного перебора для используемых на практике систем, поскольку число соядер, как правило, слишком велико. В работе предлагается алгоритм сокращенного перебора соядер, основанный на отсечении некоторых ветвей дерева по заранее определенному правилу. Опишем алгоритм сокращенного перебора.

Предварительно упорядочим буквы по частоте их встречаемости в системе. Исключим из полученного множества буквы, которые встречаются в системе один раз. Для конъюнкций, состоящих из одной буквы, оценку качества будем полагать равной нулю. На вход алгоритма подаётся упорядоченное множество букв системы.

Алгоритм выделения соядра.

1. В корень дерева поместим фиктивное значение – конъюнкцию, множество букв которой пусто.

2. Пусть очередной рассматриваемой вершине соответствует конъюнкция K. Конъюнкции K в дочерней вершине может быть приписана одна из букв, следующая по порядку за последней буквой конъюнкции K. Причем, число дочерних вершин равно числу оставшихся букв. Крайней левой вершине приписывается первая по порядку буква, крайней правой – последняя по порядку.

Добавление вершины, которой соответствует конъюнкция Ka, образованная из K (ранг Ka больше единицы) не происходит, если оценка качества конъюнкции, соответствующей добавляемой вершине, меньше, чем оценка качества соядра, расположенного на уровень выше, или уже имеется соядро того же ранга с лучшей оценкой.

3. Продолжаем построение дерева в соответствие с пунктом 2, пока это возможно.

Далее в качестве соядра выбирается конъюнкция с лучшей оценкой качества, представляемая одной из концевых вершин сокращенного дерева.

С целью сокращения объема вычислений, связанных с построением дерева, вводятся предварительные оценки качества конъюнкций, требующие для своего получения меньших вычислений, чем введенные ранее.

Структуру минимизированной системы ДНФ, полученной из исходной системы ДНФ, в упомянутой выше книге предлагается представлять комбинационной логической сетью. Эта сеть используется в дальнейшем для построения *s*-графа (*subject graph*).

Процедура построения *s*-графа заключается в реализации фиктивными базисными элементами каждой ДНФ, сопоставленной вершине комбинационной логической сети. В качестве фиктивных базисных элементов выбираются двухвходовой элемент НЕ-И (НЕ-ИЛИ) и элемент НЕ. Реализация со-

храняет ДНФ, сопоставляемую вершине сети. Затем происходит замещение вершин исходной сети соответствующими подграфами.

Утверждение 1. Схема, представляемая *s*-графом, сохраняет исходную систему ДНФ.

Во второй главе далее предлагается модификация двухуровневого факторизационного метода синтеза, позволяющая сохранять исходную систему ДНФ (традиционные методы двух уровневого синтеза приводят к усложнению исходных ДНФ, получаемых из схемы при упомянутых ранее подстановках за счет повторения конъюнкций и повторения букв в конъюнкциях)

Введем ряд определений для описания стандартных факторизационных двухуровневых методов синтеза.

Пусть $F = \{F_1, F_2, \cdots, F_m\}$ – система ДНФ, являющаяся заданием на синтез комбинационной схемы.

Определение 1. ДНФ D^* является дизъюнктивным фактором системы *F,* если D^* получается из $F_i (i \in \{1,...,m\})$ вычеркиванием некоторых конъюнкций, $D^* \subset F_i$.

Будем говорить, что D^* является частью F_i.

Определение 2. Порождающим множеством дизъюнктивного фактора D^* будем называть множество ДНФ системы, для которых D^* является частью. В порождающее множество могут быть включены не все такие ДНФ.

Пусть $K = \{K_1, K_2, \ldots, K_n\}$ – множество всех конъюнкций системы *F*.

Определение 3. Конъюнкция K^* является конъюнктивным фактором системы ДНФ, если K^* получена из конъюнкции K_i системы *F* вычеркиванием некоторых букв, $K^* \subset K_i$.

Будем говорить, что K^* является частью K_i.

Определение 4. Порождающим множеством конъюнктивного фактора K^* будем называть множество конъюнкций системы, для которых K^* является частью. В порождающее множество могут быть включены не все такие конъюнкции.

В двухуровневых методах синтеза предполагается выполнение следующих шагов:

– выбор множества $D_1^*, D_2^*, ..., D_p^*$ дизъюнктивных факторов системы и построения для каждого фактора порождающего множества;

– выбор множества $K_1^*, K_2^*, ..., K_s^*$ конъюнктивных факторов системы и построения для каждого фактора порождающего множества;

– построение на основе выбранных факторов и порождающих множеств схемы *C*, реализующей систему$\{ F_1, F_2, ..., F_m \}$.

Выполним суперпозицию по полученной схеме с учетом запрета на поглощение и склеивание конъюнкций, а также запрета на удаление повторяю-

щихся букв в конъюнкциях и повторяющихся конъюнкций. Обозначим полученную систему ДНФ D'.

Определение 5. Будем говорить, что конъюнкции являются пересекающимися по буквам, если пересечение составляющих их букв не пусто.

Определение 6. Две ДНФ будем называть пересекающимися, если они содержат одинаковые конъюнкции.

Утверждение 2. Если конъюнкция системы F принадлежит порождающим множествам пресекающихся по буквам конъюнктивных факторов, то в системе D' присутствуют конъюнкции, содержащие повторяющиеся буквы.

Утверждение 3. Если ДНФ системы F принадлежит порождающим множествам пресекающихся дизъюнктивных факторов, то в системе D' присутствуют ДНФ, содержащие повторяющиеся конъюнкции.

Утверждение 4. Схема C^*, построенную при условии, что порождающие множества пересекающихся конъюнктивных (дизъюнктивных) факторов не содержат одинаковых конъюнкций (дизъюнкций), сохраняет систему ДНФ F.

Для факторизационного двухуровневого метода синтеза, сохраняющего исходную систему ДНФ, в работе предлагаются оценки и способы выбора факторов, аналогичные разработанным для многоуровневого факторизационого метода.

Были составлены программы, реализующие приведенные во второй главе методы синтеза комбинационных схем по системам ДНФ, сохраняющие эти системы. Использовалось *STG* (*State Transition Graph*)-описание поведения синхронного последовательностного устройства, в котором входные и выходные состояния реализуемого им автомата уже закодированы. Внутренние состояния автомата кодировались плотным кодом.

При переходе от системы частичных функций к системам полностью определенных функций строилась безызбыточная система ДНФ, представляющая комбинационную составляющую последовательностного устройства. Использовалась программа, разработанная А.В. Мельниковым, реализующая метод конкурирующих интервалов, предложенный А.Д. Закревским. Далее применялась описанная во второй главе процедура минимизации, основанная на выделение ядер и соядер. Для построения *s*-графа использовался фиктивный базис <НЕ, НЕ-И>.

Экспериментальные результаты показали, что применение предложенной процедуры позволяет в среднем уменьшить сложность схемы на 36,58%, тогда как использование случайного выбора – на 31,94%. Также экспериментальные результаты показали что применение модифицированного двухуровневого метода, по сравнению с многоуровневым методом, требует значительно больших временных и вычислительных затрат и не даёт существенного сокращения аппаратурных затрат.

В заключительной части второй главы рассмотрены методы синтеза комбинационных схем по системе *BDD*-графов.

BDD-графы являются компактным представлением систем ортогональных ДНФ (ОДНФ). В разделе 2.2 приводится описание известного подхода к построению комбинационных схем покрытием системы *BDD*-графов программируемыми логическими блоками (ПЛБ) и выделяются требования к покрытию, сохраняющему систему. Под покрытием системы *BDD*-графов ПЛБ здесь и далее понимается покрытие системы *BDD*-графов *BDD*-деревьями, реализующими функции не более чем от k переменных. Здесь k – число входов ПЛБ, причем ПЛБ реализует функцию, представленную соответствующим покрывающим деревом.

Пусть $f(x_1, x_2, \ldots, x_n)$ – булева функция от n переменных.

Определение 7. *BDD*-графом булевой функции f называется корневой ориентированный ациклический связный граф, обладающий следующими свойствами:

– вершины, не имеющие исходящих дуг, называемые терминальными, отмечаются символом 0 или 1;

– все нетерминальные вершины отмечаются символом переменной, или её индексом, и имеют две исходящие дуги, одна из которых отмечается нулём, другая – единицей.

Каждой нетерминальной вершине v ставится в соответствие функция f_v, корню – функция f. Построение *BDD*-графа основано на использовании разложения Шеннона в каждой нетерминальной вершине v.

Каждой нетерминальной вершине v *BDD*-графа соответствует функция f_v ($\overline{f_v}$), представленная в виде ОДНФ соответствующими простыми цепями, соединяющими v с 1(0)-концевой вершиной *BDD*-графа.

Определение 8. *BDD*-граф называется упорядоченным, если переменные перечисляются в одном и том же порядке во всех простых цепях, соединяющих корень *BDD*-графа с терминальными вершинами.

Определение 9. *BDD*-граф называется сокращенным, если он не содержит ни изоморфных подграфов, ни вершин v, для которых обе дуги, исходящие из вершины, заходят в изоморфные подграфы.

Определение 10. Упорядоченный сокращенный *BDD*-граф называется *ROBDD* (*Reduced Ordered Binary Decision Diagram*)-графом

При построении *Free BDD*-графа порядок переменных, по которым производится разложение, заранее не фиксирован. *Free BDD*-граф может быть сокращённым, т.е. может не содержать ни изоморфных подграфов, ни вершин v, дуги которых заходят в изоморфные подграфы. В каждой нетерминальной вершине переменная разложения выбирается по тому или иному критерию независимо от других нетерминальных вершин, что позволяет в некоторых случаях сократить число вершин *BDD*-графа.

Процедура построения *BDD*-графа системы функций или *SBDD*-графа (*Shared Binary Decision Diagram*) состоит в следующем. Первоначально каждая функция в системе представляется в виде *BDD*-графа одного и того же

типа, то есть либо в виде *ROBDD*-графа, либо *Free BDD*-графа. Далее, 1(0)-концевые вершины различных графов совмещаются в одну 1(0)-концевую вершину. Представление системы булевых функций в виде *SBDD*-графа, как правило, является более компактным по сравнению с использованием отдельных *BDD*-графов для каждой из функций системы.

Удалим из системы *SBDD*-графа все дуги, заходящие в 0-концевую вершину, и получим *SBDD*-граф, представляющий поведение комбинационной схемы. Назовем его *SBDD*-графом схемы. Покроем его программируемыми логическими блоками (ПЛБ), создаваемыми на базе *LUT* (*Look up Table*)-технологий. Рассматриваемым в работе условием построения покрытия является сохранение системы ОДНФ, порожденной *SBDD*-графом схемы. Это значит, что, выполнив по синтезированной схеме подстановки функций, реализуемых соответствующими ПЛБ с учетом запрета на упрощения формул при раскрытии скобок, получим систему ОДНФ, содержащую все конъюнкции, порожденные простыми цепями, соединяющими корни *SBDD*-графа схемы с его 1-концевой вершиной, и только их.

Один из возможных способов покрытия, удовлетворяющих рассматриваемому условию, заключается в следующем:

– выходу ПЛБ ставится в соответствие либо корневая вершина, либо нетерминальная вершина *SBDD*-графа схемы;

– булева функция, реализуемая ПЛБ, представляется соответствующим подграфом *SBDD*-графа схемы, покрытым этим ПЛБ;

– если несколько дуг в *SBDD*-графе схемы заходят в одну и ту же нетерминальную вершину, то она может быть покрыта различными ПЛБ.

Будем иметь в виду, что контролепригодные свойства схем, сохраняющих системы безызбыточных ДНФ и безызбыточные системы ДНФ достаточно хорошо исследованы, в данной работе исследуются контролепригодные свойства схем, сохраняющих системы ОДНФ, представляемые *SBDD*-графами.

В **третьей главе** исследуются контролепригодные свойства схемы, построенной путём покрытия *SBDD*-графа программируемыми логическими блоками. Используются введенные в работе *А. Ю. Матросовой, Е. С. Луковниковой «Построение проверяющих тестов для одиночных и кратных неисправностей на полюсах элементов схем, синтезированных на базе ПЛИС (FPGA)-технологий» Приложение № 23. Доклады Сибирской научной школы-семинара с международным участием «Компьютерная безопасность и криптография». – 2007. – С. 229–241.* базовые неисправности *SBDD*-графа, представляющие одиночные константные неисправности на полюсах ПЛБ комбинационной схемы, полученной покрытием графа с сохранением соответствующей системы ОДНФ. Предлагается алгоритм построения тестового набора для произвольной кратной константной неисправности на полюсах

ПЛБ рассматриваемой комбинационной схемы и алгоритм построения проверяющего теста для всех кратных константных неисправностей схемы.

Воспользуемся определениями базовых неисправностей.

Определение 11. Неисправность нетерминальной вершины v *SBDD*-графа, в присутствии которой функция f_v обращается в константу 0(1) и $\overline{f_v}$ обращается в константу 1(0), будем называть неисправностью константа 0(1) вершины v.

Определение 12. 01(10)-неисправностью нетерминальной вершины *SBDD*-графа будем называть неисправность дуг вершины, в присутствии которой дуга, отмеченная символом 0, заменяется константой 0(1), а дуга, отмеченная символом 1, заменяется константой 1(0). Дугу, заменённую в S*BDD*-графе константой 0(1), будем называть 0(1)-дугой.

Так как порядок разложения по переменным при построении *Free BDD*-графа не фиксирован, в *SBDD*-графе возможно существование вершины v, отмеченной индексом i, для которой $f_v^{x_i=0} = f_v^{x_i=1}$, то есть существование неизоморфных подграфов, реализующих одну и ту же функцию. Неисправность 10 (01) в вершине такого типа не существенна, так как при замене соответствующими константами переменных $\overline{x}_i$ и x_i функции f_v и $\overline{f_v}$ не изменяются.

Пусть вершина v соединяется с j-м корнем *SBDD*-графа.

Будем говорить, что набор α значений входных переменных $x_1, \ldots, x_n$ представляется некоторой простой цепью *SBDD*-графа, если он обращает в единицу конъюнкцию, сопоставляемую этой простой цепи.

В упомянутой в третьей главе работе для схем, полученных покрытием системы *ROBDD*-графов программируемыми логическими блоками, и константной неисправности на входном полюсе некоторого ПЛБ, совпадающем с входным полюсом x_i схемы, установлен следующий факт. Для нахождения тестового набора, обнаруживающего 10(01)-неисправность, достаточно ограничиться рассмотрением 10(01)-неисправности только одной из вершин v *SBDD*-графа, покрытой рассматриваемым ПЛБ и помеченной символом i, и одной простой цепью, заходящей в эту вершину из j-го корня *SBDD*-графа.

В диссертационной работе установлено, что для схем, полученных покрытием системы *Free BDD*-графов программируемыми логическими блоками, рассмотрение одной вершины указанного типа может оказаться недостаточным. Если построение теста для очередной рассматриваемой вершины невозможно, то необходимо перейти к следующей вершине этого типа, если таких вершин больше нет, то данная неисправность не существенна.

Рассмотрим кратную неисправность $V_{кр}$, составленную из произвольного множества $v_1, \ldots, v_s$ базовых неисправностей *SBDD*-графа. Составляющие неисправности множества представляются 0(1)- и 01(10)-неисправностями *SBDD*-графа.

Некоторые из базовых 01(10)-неисправностей могут быть несущественными. Неисправности и представляющие их вершины в дальнейшем будем отождествлять, используя термин «неисправная вершина», если это удобно.

Определение 13. Пара базовых неисправностей связана отношением подчинения, если представляющие эти неисправности вершины лежат на одной и той же простой цепи, соединяющей некоторый корень *SBDD*-графа с его 1(0)-концевой вершиной.

Определение 14. Базовая неисправность, представленная ближайшей к корню вершиной, называется подчиняющей; неисправность, представленная более удаленной вершиной, – подчиненной.

Определение 15. Базовую подчиняющую неисправность v_i будем называть доминирующей по отношению к подчиненной базовой неисправности v_j, если в присутствии неисправности v_i *SBDD*-граф представляет ту же систему булевых функций, что и в присутствии кратной неисправности (v_i, v_j).

Вместо подчиненной базовой неисправности v_j можно рассматривать подмножество подчиненных базовых неисправностей V_j, введя аналогичное отношение доминирования между v_i и V_j.

Рассмотрим базовую 0(1)-неисправность v_i из $V_{кр}$. Ее проявление в *SBDD*-графе приводит к совмещению вершины v_i с 0(1) -концевой вершиной графа.

Пусть V_i множество подчиненных v_i одиночных неисправностей из $V_{кр}$, не содержащее v_i. Вершина v_i является корневой для подграфа, в пределах которого находятся неисправности из V_i.

Теорема 1. Базовая 0(1)-неисправность v_i является доминирующей по отношению к любой базовой неисправности из V_i.

Следствие. Базовая 0(1)-неисправность v_i является доминирующей по отношению к кратной неисправности V_i.

Рассмотрим базовую 01(10)-неисправность v_i. Её присутствие в *SBDD*-графе приводит к отсоединению от v_i (путем обрыва 0-дуги) подграфа, корнем которого является вершина, в которую эта дуга заходит, и совмещению вершины v_i с вершиной, в которую заходит 1-дуга вершины v_i.

Обозначим через V_i^0 множество неисправностей из V_i, лежащих в отсоединяемом подграфе. Будем говорить, что подграф подчинен 0-дуге.

Обозначим через V_i^1 множество неисправностей из V_i, подчиненных 1-дуге вершины v_i. Неисправности множества V_i^1 лежат в подграфе, корнем которого является вершина, в которую заходит 1-дуга. Множества V_i^1, V_i^0 могут пересекаться при наличии общих вершин в соответствующих им подграфах. Заметим, что если V_i^1 пусто, то базовые неисправности из V_i^0 не подчинены 1-дуге.

Теорема 2. Базовая 01(10)-неисправность v_i является доминирующей по отношению к множеству неисправностей V_i^0, если V_i^1 пусто.

Теорема 3. Кратная неисправность $\left(v_{i,}\ V_i^1\right)$ является доминирующей по отношению к кратной неисправности $\left(v_{i,}\ V_i^1\ ,V_i^0\right)$.

Из теорем 2, 3 следует, что неисправности множества V_i^0 можно исключить из рассмотрения при анализе влияния подчиненных неисправностей на 01(10)-неисправность v_i. Отметим, что это справедливо и в случае если 01(10)-неисправность вершины v_i не существенна.

Определение 16. Будем говорить, что неисправность v_j разрушает тестовый набор α для существенной неисправности v_i, если тестовый набор для v_i перестает быть таковым, когда к v_i добавляется неисправность v_j.

Теорема 4. Подчиненная существенная 0(1)-неисправность или 01(10)-неисправность нетерминальной вершины v_j из V_i^1 может быть обнаружима в присутствии несущественной подчиняющей 01(10)-неисправности v_i.

Теорема 5. Подчиненная 0(1)-неисправность нетерминальной вершины v_l из V_i^1 может разрушить тестовый набор α для существенной подчиняющей 01(10)-неисправности v_i.

Теорема 6. Существенная, подчиненная 01(10)-неисправность v_j из V_i^1 может разрушить тестовый набор для существенной подчиняющей 01(10)-неисправности v_i.

Теорема 7. Тестовый набор α, построенный для подчиненной 0(1)- или существенной 01(10)-неисправности v_i, может быть разрушен другой подчиняющей неисправностью v_l из $V_{кр}$, если представляющая последнюю неисправность вершина принадлежит простой цепи ε, соединяющей v_i с j-м корнем *SBDD*-графа, и эта простая цепь используется для получения тестового набора α.

В диссертационной работе предлагается отношения между одиночными неисправностями, составляющими кратную неисправность $V_{кр}$, представлять специальным графом. Алгоритм построения тестового набора для произвольной кратной константной неисправности на полюсах логических элементов комбинационной схемы рассматриваемого вида строится с использованием этого графа и теорем 2–7.

Из алгоритма построения тестового набора для произвольной кратной неисправности следует, что 1) для $V_{кр}$ может не существовать тестового набора среди построенных независимо друг от друга тестовых наборов для составляющих ее одиночных базовых, существенных неисправностей; 2) произвольный проверяющий тест для одиночных константных неисправностей можно расширить до проверяющего теста для кратных константных неисправностей схемы.

Для построения проверяющего теста для кратных константных неисправностей нужно рассмотреть все сопоставляемые одиночным константным неисправностям на полюсах ПЛБ базовые неисправности *SBDD*-графа и

представляющие их вершины. Для каждой из вершин этого множества необходимо выполнить следующие процедуры.

1. Найти все простые цепи, соединяющие выбранную неисправную вершину v_i с j - м корнем *SBDD*-графа, причем, в случае, если рассматривается 0(1)-неисправность, сопоставляется входному полюсу ПЛБ, то эти простые цепи одновременно проходят через полюс *SBDD*-графа, соответствующий выходу неисправного ПЛБ.

2. Найти множество конъюнкций K_i, представляемое этими цепями (из построения *SBDD*-графа следует, что конъюнкции множества попарно ортогональны).

3. Выполнить шаги 1, 2 для всех корней *SBDD*-графа. Найти минимальное по мощности множество K_i. Полученное множество обозначаем K^*. Пусть q – его мощность.

4. Заменить имеющийся тестовый набор для одиночной константной неисправности на q тестовых наборов, которые будут отличаться друг от друга только по переменным множества K^*.

Выполнив шаги 1–4 для всех порожденных комбинационной схемой и ее программируемыми логическими блоками базовых неисправностей *SBDD*-графа, получим проверяющий тест для кратных неисправностей комбинационной схемы.

Таблица 1

пример	i	o	l	v	число входов ПЛБ: 4			число входов ПЛБ: 8		
					n	T_s	T_m	n	T_s	T_m
BBARA	4	2	4	74	40	57	70	6	31	54
CSE	7	7	4	235	140	194	243	23	81	233
DK14	3	5	3	76	44	47	58	8	28	43
DK15	3	5	2	48	19	21	23	7	18	23
DK27	1	2	3	21	5	11	15	5	11	15
EX2	2	2	5	119	59	87	107	7	30	66
KIRKMAN	12	6	4	416	253	732	4122	124	974	1960
PMA	8	8	5	340	254	407	668	33	123	491
S1A	8	6	5	1012	867	1546	3432	104	486	2391
S386	7	7	4	252	150	208	348	25	97	331
SAND	11	9	5	2648	2434	3389	16877	365	1479	13428
SSE	7	7	4	189	124	184	271	15	52	222
STYR	9	10	5	689	517	638	1138	75	255	987

Для контрольных примеров (*benchmarks*), задающих поведение комбинационной составляющей последовательностного синхронного устройства, строились безызбыточные системы ДНФ, по ним находились *SBDD*-графы. Для кодирования внутренних состояний использовался плотный код. Далее строились соответствующие графы схемы, которые покрывались ПЛБ. В результате были получены комбинационные схемы из ПЛБ, реализующие исходное поведение синхронного устройства. Для каждой схемы строился проверяющий тест, обнаруживающий все одиночные константные неисправности и проверяющий тест для кратных константных неисправностей полюсов ПЛБ. Результаты экспериментов приведены в таблице 1.

В таблице 1 использовались следующие обозначения: *i, o, l, v* – числа входов, выходов, линий обратной связи и вершин *SBDD*-графа; *n* – число ПЛБ; T_s, T_m – число тестовых наборов в проверяющем тесте, обнаруживающим одиночные и кратные константные неисправности на полюсах ПЛБ схемы соответственно. Анализ экспериментальных результатов показал, что число тестовых наборов в проверяющем тесте, обнаруживающем кратные константные неисправности на полюсах ПЛБ схемы, превышает число тестовых наборов в проверяющем тесте, обнаруживающем одиночные константные неисправности, в среднем в 2.5 раза, и не более чем в 9 раз.

Четвертая глава посвящена тестированию неисправностей задержек путей в схемах, построенных путём покрытия *SBDD*-графа программируемыми логическими блоками, сохраняющими систему ОДНФ, представляемую *SBDD*-графом. Предлагается алгоритм построения пар тестовых наборов для обнаружения неисправностей задержек путей схемы. Устанавливается, что для каждого пути в схеме существует пара тестовых наборов, на которой неисправность задержки пути проявляется как робастная. В отличие от результатов, полученных в работе: *R. Drechsler, J. Shi, G. Fey "Synthesis of Fully Testable Circuits From BDDs", IEEE Trans. on CAD, vol.* 23*, No.* 3*, pp.* 440–443 в комбинационную схему не требуется вводить дополнительный вход для обеспечения аналогичных свойств пар тестовых наборов неисправностей задержек путей. Введение дополнительных входов, как правило, не приемлемо на практике. Исследуются свойства построенного теста. Показывается, что тест, обнаруживающий все одиночные робастные неисправности задержек путей в схеме, является так же проверяющим тестом для одиночных константных неисправностей на полюсах логических элементов схемы.

Также показывается, что тест, обнаруживающий все одиночные робастные неисправности задержек путей, обнаруживает все кратные неисправности задержек путей.

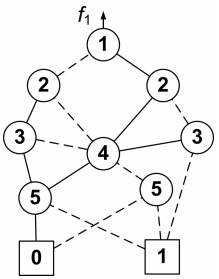

Рисунок 1 – ROBDD-граф схемы С

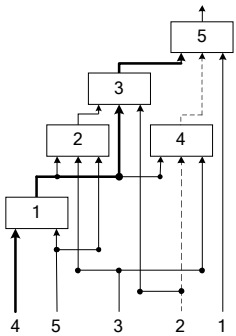

Рисунок 2 – схема С

Пусть комбинационная схема получена покрытием *SBDD*-графа схемы программируемыми логическими блоками, причем *SBDD*-граф построен объединением *ROBDD*-графов. На рис. 1 изображен *ROBDD*-граф, а на рис. 2 соответствующая ему схема. Припишем программируемым логическим блокам комбинационной схемы номера в порядке возрастания от входов к выходам схемы.

Путь α представляется последовательностью программируемых логических блоков (ПЛБ), проходимых на этом пути: $p_{i1}, \ldots, p_{ir}$. Пусть x_i – переменная, сопоставляемая входу схемы на рассматриваемом пути, выход элемента p_{ir} является выходом t схемы, сопоставляемым этому пути.

Замечание 1. Поскольку рассматриваются ПЛБ, создаваемые на базе *LUT*-технологий, то естественно предполагать, что время прохождения сигнала по любому из путей графа, покрытого одним и тем же ПЛБ, одинаково и определяется параметрами конкретного ПЛБ. Это значит, что при рассмотрении некоторого пути схемы не имеет значения, какой путь выбран в подграфе конкретного ПЛБ, проходимого на этом пути.

Рассмотрим программируемые логические блоки на пути α. Построим усеченную ОДНФ для каждого ПЛБ этого пути, составленную из конъюнкций, в которых либо присутствует переменная, соответствующая выходу ПЛБ, непосредственно предшествующему рассматриваемому блоку, либо содержится переменная x_i, сопоставляемая входу сети и являющаяся началом пути α.

Двигаемся по пути α от выхода t схемы к входу x_i, то есть от элемента p_{ir} к p_{i1}. Вместо внутренних переменных подставляем усеченные ОДНФ, им соответствующие. Если в полученном после прохождения всего пути множестве конъюнкций окажутся конъюнкции, в которых содержатся внутренние переменные, продолжаем подстановки, заменяя теперь каждую внутреннюю переменную соответствующей полной ОДНФ, до тех пор, пока множество внутренних переменных не окажется пустым. Полученное множество конъюнкций обозначаем символом K_α.

Теорема 8. Множество K_α содержит все конъюнкции, сопоставляемые пути α, такие, что в каждой конъюнкции присутствует либо переменная x_i, либо ее инверсия. Конъюнкции множества попарно ортогональны.

Пусть путь α начинается с входной переменной x_4 и проходит через $ПЛБ_1$, $ПЛБ_3$, $ПЛБ_5$, выход $ПЛБ_5$ является выходом подсхемы. Этот путь на рис. 2 выделен жирной линией.

В примере для пути α получаем из усеченных ОДНФ после первой подстановки выражение: $\bar{x}_1\bar{x}_2w_1$. Заменив w_1 усеченной ОДНФ, далее имеем выражение: $\bar{x}_1\bar{x}_2x_4\bar{x}_5 \vee \bar{x}_1\bar{x}_2\bar{x}_4x_5$. Оно представляет множество K_α.

Теорема 9. Каждой конъюнкции из K_α соответствует путь в *ROBDD*-графе схемы от корня, сопоставляемого выходу t схемы, до 1-концевой вершины графа.

Выберем для построения одного из наборов v_1 тестовой пары конъюнкцию K, которая может содержать либо x_i, либо $\bar{x}_i$. Тестовый набор обращает эту конъюнкцию в единицу. Конъюнкция $\underline{K}$ получается из конъюнкции K заменой переменной $x_i(\bar{x}_i)$ на инверсную переменную.

Другой набор v_2 тестовой пары должен обратить в единицу $\underline{K}$ и в ноль все конъюнкции из F_t.

Построим минимально покрывающий интервал для наборов v_1, v_2, обозначим его символом u. Пусть k_u – конъюнкция, его представляющая.

В диссертационной работе показывается, что в K_α найдется конъюнкция K, такая, что для конъюнкций K, $\underline{K}$ существует пара наборов v_1, v_2, причем k_u ортогональна всем конъюнкциям ОДНФ F_t, кроме K. Алгоритм построения одного из тестовых наборов пары совпадает с алгоритмом построения тестового набора для 01(10)-неисправности вершины v. Второй набор тестовой пары находится из полученного набора заменой i-ой компоненты инверсным значением. Здесь i – индекс переменной x_i, сопоставляемой вершине v.

Показывается, что пара тестовых наборов обнаруживает инверсные перепады значений сигналов пути α при изменении порядка их поступления на тестируемую схему. Это дает возможность для одного пути использовать вместо 4-х наборов, как это обычно делается, три набора v_1, v_2, v_1 или v_2, v_1, v_2 и тем самым на ¼ часть сократить длину проверяющего теста для неисправностей задержек путей комбинационной схемы.

Выявлены свойства проверяющего теста для неисправностей задержек путей, сформулированные в виде следующих теорем.

Теорема 10. Проверяющий тест T, обнаруживающий все одиночные робастные неисправности задержек путей, содержит проверяющий тест для всех одиночных константных неисправностей на полюсах ПЛБ схемы.

Теорема 11. Проверяющий тест T, обнаруживающий все одиночные робастные неисправности задержек путей, обнаруживает все кратные неисправности задержек путей.

Из последней теоремы следует, что проверяющий тест T позволяет гарантированно обнаруживать всевозможные неисправности задержек путей рассматриваемых комбинационных схем. Полученные результаты распространены на *SBDD*-графы, построенные объединением *Free BDD*-графов.

Для контрольных примеров (*bench-marks*) из набора LGSynth'91, строились *SBDD*-графы путем объединения *ROBDD*-графов, далее строились соответствующие графы схемы, которые покрывались ПЛБ с числом входов равным 2. В построенной схеме подсчитывались числа путей и числа элементов схемы.

Столбцы *MUX* и *LUT* таблицы содержат информацию о схемах, полученных покрытием соответствующих *SBDD*-графов мультиплексорами, и о схемах, построенных покрытием соответствующих *SBDD*-графов ПЛБ с тремя входами. *NoN* – число мультиплексоров, *Cld* – число ПЛБ, *NoP* – число путей в схеме, которые должны быть протестированы, *PDFC* – процент путей, для которых возможно построить тестовую пару.

Таблица 2

пример	*i*	*o*	*MUX*			*LUT*		
			NoN	*NoP*	*PDFC*	*clb*	*NoP*	*PDFC*
5xpl	7	10	90	273	89.0	69	175	100
C17	5	2	12	22	68.1	6	12	100
alu2	10	6	259	873	86.9	246	929	100
b9	41	21	237	1773	64.6	141	380	100
clip	9	5	256	954	79.4	235	597	100
conl	7	2	20	47	74.4	9	18	100
count	35	16	251	2248	66.1	199	642	100
il	25	13	60	137	74.4	41	85	100
i5	133	66	313	44198	61.3	169	941	100
t481	16	1	34	4518	86.1	26	1226	100
tcon	17	16	34	40	100.0	16	32	100
9sym	9	1	35	328	72.5	27	195	100
f51m	8	8	72	326	99.3	58	139	100
z4ml	7	4	66	175	77.1	50	118	100
x2	10	7	75	188	72.3	61	251	100

Анализ экспериментальных результатов показывает, что числа ПЛБ и числа путей в схемах, построенных покрытием *SBDD*-графов ПЛБ, как правило, меньше чисел мультиплексоров и чисел путей в схемах, построенных покрытием *SBDD*-графов мультиплексорами.

Введение дополнительного входа с целью обеспечения контролепригодных свойств схемы из мультиплексоров приводит к увеличению числа путей и числа мультиплексоров в этой схеме.

Пара тестовых наборов v_1, v_2, обнаруживающая робастную неисправность задержки пути в схемах, построенных покрытием *SBDD*-графов ПЛБ, может быть использована для тестирования обоих перепадов значений сигналов пути. Это значит, что для тестирования неисправностей задержек обоих перепадов значений сигналов пути достаточно тройки v_1, v_2, v_1, тестовых наборов вместо обычно используемых четырех наборов. Это позволяет сократить длину проверяющего теста схемы.

Заключение

В диссертационной работе рассмотрены методы синтеза схем, позволяющие сохранять исходное представление систем булевых функций в виде системы ДНФ. В таких схемах тестовые наборы для неисправностей схем могут быть построены непосредственно по исходным системам ДНФ с использованием свойств ДНФ.

Предложена модификация двухуровневого факторизационного метода синтеза, позволяющая сохранять исходную систему ДНФ.

Разработаны эвристики для двухуровневого и многоуровневого факторизационных методов синтеза, ориентированные на сокращение аппаратурных затрат при синтезе рассматриваемых схем.

Исследованы контролепригодные свойства схем, построенных покрытием *SBDD*-графов, полученных объединением *ROBDD* или *Free BDD*-графов программируемыми логическими блоками (ПЛБ).

Разработан алгоритм построения тестового набора для кратной константной неисправности на полюсах ПЛБ комбинационной схемы.

Построение проверяющего теста для кратных константных неисправностей на полюсах ПЛБ комбинационной схемы сведено к расширению прове-

ряющего теста для одиночных неисправностей на полюсах ПЛБ этой схемы, и предложена процедура расширения.

Установлено, что в комбинационных схемах, построенных покрытием систем из *Free BDD*-графов, не все одиночные константные неисправности являются существенными.

Для схем, построенных покрытием *SBDD*-графов, полученных объединением *ROBDD* или *Free BDD*-графов, программируемыми логическими блоками, предложен алгоритм построения проверяющего теста, обнаруживающего неисправности задержек путей.

Установлено, что неисправность задержки каждого пути схемы проявляется как робастная.

Основные положения, выдвигаемые на защиту:

– Эвристики для многоуровневого и двухуровневого факторизационных методов синтеза, ориентированные на сокращение аппаратурных затрат получаемых комбинационных схем. Модифицированный двухуровневый метод синтеза, позволяющий сохранять систему ДНФ. Если в качестве систем ДНФ используются системы безызбыточных ДНФ, то синтезируемые схемы допускают существование для них проверяющего теста для одиночных константных неисправностей, обнаруживающего все кратные константные неисправности на полюсах логических элементов. Неисправности задержек путей в таких схемах более чем в 95% случаев проявляются как робастные.

– Алгоритм построения тестового набора, обнаруживающего кратную константную неисправность на полюсах ПЛБ схемы, построенной покрытием *SBDD*-графа программируемыми логическими блоками; алгоритм построения проверяющего теста для всех кратных константных неисправностей на полюсах ПЛБ схемы, не требующий явного перечисления всевозможных кратных неисправностей и получаемый путем расширения проверяющего теста для одиночных константных неисправностей.

– Алгоритм построения проверяющего теста, гарантированно обнаруживающий всевозможные неисправности задержек путей в схеме, построенной покрытием *SBDD*-графа программируемыми логическими блоками.

СПИСОК ПУБЛИКАЦИЙ ПО ТЕМЕ ДИССЕРТАЦИИ

1. Николаева Е. А. Синтез проверяющих тестов для неисправностей задержек путей схем, построенных по системе *ROBDD*-графов / Е. А. Николаева, А. Ю. Матросова // Известия Томского политехнического университета. –2009. – Т. 315, № 5. – С. 153 – 159.

2. Николаева Е. А. Некоторые оценки метода минимизации систем ДНФ путем выделения ядер и соядер / Е. А. Николаева, Ю. В. Седов // Доклады VI Сибирской научной школы-семинара с международным участием «Компьютерная безопасность и криптография» – SIBECRYPT’07 (Республика Алтай, Горно-Алтайск, ГАГУ, 4–7 сентября 2007 г.). Томск, 2007. С. 256–264. (Вестник Томского государственного университета. Приложение; № 23, Август

2007) (Серия «Математика. Кибернетика. Информатика»). – 2007. – С. 256–264.

3. Matrosova A. Test Generation for Single and Multiple Stuck-at Faults of a Combinational Circuit Designed by Covering Shared ROBDD with CLBs / A. Matrosova, E. Lukovnikova, S. Ostanin, A. Zinchyk, E. Nikolaeva // Proc. The 22nd IEEE International Symposium on Defect and Fault-Tolerance in VLSI Systems (DTF 2007). Rome, Italy, 26–28 September, 2007. – [S. l.], 2007. – P. 356–364.

4. Матросова А. Ю. Синтез синхронных последовательностных устройств, устойчивых к кратковременным и перемежающимся неисправностям / А. Ю. Матросова, В. В. Андреева, Е. А. Николаева // Вестник ТГУ. Управление, вычислительная техника и информатика. – 2008. – №3 (4). – С. 99–109.

5. Matrosova A. Multiple Stuck-at Fault and Path Delay Fault Testable Design of Combinational Circuits / A. Matrosova, A. Melnikov, E. Nikolaeva // Proc. IEEE East-West Design & Test Symposium (EWDTS'08). Lviv, Ukraine, 9-12 October, 2008. – [S. l.], 2008. – P. 350–355.

6. Matrosova A. Multiple stuck-at fault and path delay fault testable circuit / A. Matrosova, V. Andreeva, A. Melnikov, E. Nikolaeva // Proc. IEEE East-West Design & Test Symposium (EWDTS'08). Lviv, Ukraine, 9-12 October, 2008. – [S. l.], 2008. – P. 356–364.

7. Николаева Е. А. Построение проверяющих тестов для одиночных и кратных константных неисправностей на полюсах элементов схем, синтезированных на базе ПЛИС(*FPGA*)-технологий по системе *Free BDD*-графов/ Е. А. Николаева // Вестник ТГУ. Управление, вычислительная техника и информатика. – 2009. – № 1 (6) . – С. 81–98.

8. Matrosova A. Path Delay Fault Classification Based on ENF Analysis / A. Matrosova, E. Nikolaeva // Proc. IEEE East-West Design & Test Symposium (EWDTS'09). Moscow, Russia, 18–21 Septembe, 2009. – [S. l.], 2009 – P. 526–531.

9. Nikolaeva E. PDFs Testing of Combinational Circuits Based on Covering ROBDDs / A. Matrosova, E. Nikolaeva // Proc. IEEE East-West Design & Test Symposium (EWDTS'10). St. Petersburg, Russia, 17–29 Septembe, 2010. – [S. l.], 2010 – P. 160–163.

Тираж 100 экз.
Отпечатано в ООО «Позитив-НБ»
634050 г. Томск, пр. Ленина 34а