

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ УКРАИНЫ

ХАРЬКОВСКИЙ НАЦИОНАЛЬНЫЙ
УНИВЕРСИТЕТ РАДИОЭЛЕКТРОНИКИ

ISSN 0135-1710

АВТОМАТИЗИРОВАННЫЕ СИСТЕМЫ УПРАВЛЕНИЯ И ПРИБОРЫ АВТОМАТИКИ

**Всеукраинский межведомственный
научно-технический сборник**

Основан в 1965 г.

Выпуск 168

Харьков
2014

В сборнике представлены результаты исследований, касающихся компьютерной инженерии, управления, технической диагностики, автоматизации проектирования, оптимизированного использования компьютерных сетей и создания интеллектуальных экспертных систем. Предложены новые подходы, алгоритмы и их программная реализация в области автоматического управления сложными системами, оригинальные информационные технологии в науке, образовании, медицине.

Для преподавателей университетов, научных работников, специалистов, аспирантов.

У збірнику наведено результати досліджень, що стосуються комп'ютерної інженерії, управління, технічної діагностики, автоматизації проектування, оптимізованого використання комп'ютерних мереж і створення інтелектуальних експертних систем. Запропоновано нові підходи, алгоритми та їх програмна реалізація в області автоматичного управління складними системами, оригінальні інформаційні технології в науці, освіті, медицині.

Для викладачів університетів, науковців, фахівців, аспірантів.

Редакционная коллегия:

В.В. Семенец, д-р техн. наук, проф. (гл. ред.); *М.Ф. Бондаренко*, д-р техн. наук, проф.; *И.Д. Горбенко*, д-р техн. наук, проф.; *Е.П. Пулятин*, д-р техн. наук, проф.; *В.П. Тарасенко*, д-р техн. наук, проф.; *Г.И. Загарий*, д-р техн. наук, проф.; *Г.Ф. Кривуля*, д-р техн. наук, проф.; *Чумаченко С.В.*, д-р техн. наук, проф.; *В.А. Филатов*, д-р техн. наук, проф.; *Е.В. Бодянский*, д-р техн. наук, проф.; *Э.Г. Петров*, д-р техн. наук, проф.; *В.Ф. Шостак*, д-р техн. наук, проф.; *В.М. Левыкин*, д-р техн. наук, проф.; *Е.И. Литвинова*, д-р техн. наук, проф.; *В.И. Хаханов*, д-р техн. наук, проф. (отв. ред.).

Свидетельство о государственной регистрации
печатного средства массовой информации

КВ № 12073-944ПР от 07.12.2006 г.

Адрес редакционной коллегии: Украина, 61166, Харьков, просп. Ленина, 14, Харьковский национальный университет радиоэлектроники, комн. 321, тел. 70-21-326

© Харківський національний університет
радіоелектроніки, 2014

СОДЕРЖАНИЕ

БАРАННИК Д.В., БЕКИРОВ А.Э. КОНЦЕПЦИЯ СТРУКТУРНОГО СТЕГАНОГРАФИЧЕСКОГО КОДИРОВАНИЯ С МАСКИРОВАНИЕМ.....	4
ЛЕВЫКИН В.М., ВОРОНИН А.А., ГАРЯЧЕВСКАЯ И.В. РАЗРАБОТКА МОДЕЛИ КОНСТРУКТОРА WEB ФОРМ “ALVOR FORM BUILDER” И ЕЁ РЕАЛИЗАЦИЯ.....	11
ГАЛУШКА И.Н., ЩЕРБАК С.С. ОЦЕНКА ЭФФЕКТИВНОСТИ ИНТЕГРАЦИОННЫХ РЕШЕНИЙ НА ОСНОВЕ ХРАНИЛИЩ ТРИПЛЕТОВ.....	18
ЛУГОВОЙ А.В., ПРИТЧИН А.С. УСОВЕРШЕНСТВОВАНИЕ МАТЕМАТИЧЕСКОЙ МОДЕЛИ РАСПРЕДЕЛЕНИЯ ЛЕГИРУЮЩЕЙ ПРИМЕСИ В ПРОЦЕССЕ ВЫРАЩИВАНИЯ СЛИТКОВ КРЕМНИЯ.....	24
ОКСАНИЧ А.П., КОГДАСЬ М.Г., АНДРОСЮК М.С. ИССЛЕДОВАНИЕ СТРУКТУРНЫХ И ОПТИЧЕСКИХ ХАРАКТЕРИСТИК СЛИТКОВ ПОЛУИЗОЛИРУЮЩЕГО GaAs БОЛЬШОГО ДИАМЕТРА.....	30
МОСКАЛЕНКО В.В., РЫЖОВА А.С. ИНТЕЛЛЕКТУАЛЬНАЯ СИСТЕМА ПОДДЕРЖКИ ПРИНЯТИЯ РЕШЕНИЙ С ОПТИМИЗАЦИЕЙ ПРОСТРАНСТВЕННО-ВРЕМЕННЫХ ПАРАМЕТРОВ ФУНКЦИОНИРОВАНИЯ.....	36
ШКИЛЬ А.С., ФАСТОВЕЦ Г.П., СЕРОКУРОВА А.С. АВТОМАТИЗАЦИЯ ПОИСКА ОШИБОК ПРОЕКТИРОВАНИЯ В HDL-МОДЕЛЯХ КОНЕЧНЫХ АВТОМАТОВ.....	43
БАБЕНКО В.Г., ЗАЖОМА В.М., НЕСТЕРЕНКО О.Б. МЕТОД ВБУДОВУВАННЯ СТЕГОПОВІДОМЛЕННЯ НА ОСНОВІ КЛЮЧОВОГО ЕЛЕМЕНТА.....	53
КОНАХОВИЧ Г.Ф. ОЦЕНКА ЭФФЕКТИВНОСТИ МЕТОДОВ СТЕГАНОГРАФИЧЕСКОГО ВСТРАИВАНИЯ ИНФОРМАЦИИ В СПЕКТРАЛЬНУЮ ОБЛАСТЬ ИЗОБРАЖЕНИЙ.....	59
ХАХАНОВ В.И., ОБРИЗАН В.И., ЗАЙЧЕНКО С.А., ХАХАНОВ И.В. MQT-АВТОМАТ ДЛЯ АНАЛИЗА БОЛЬШИХ ДАННЫХ.....	64
РЕФЕРАТИ	73
ПРАВИЛА ОФОРМЛЕНИЯ РУКОПИСЕЙ ДЛЯ АВТОРОВ НАУЧНО-ТЕХНИЧЕСКОГО СБОРНИКА.....	77

Список литературы: 1. *Antsaklis P.J.* An Introduction to Intelligent and Autonomous Control / P.J. Antsaklis, K.M. Passino. Norwell, Massachusetts, USA: Kluwer Academic Publishers. 1992. 448 p. 2. *Bhagat P.* Pattern Recognition in Industry / P. Bhagat. Amsterdam, Netherlands : Elsevier Science. 2005. 200 p. 3. *Москаленко В.В.* Интеллектуальна автоматизована система керування з оптимізацією часових параметрів аналізу вхідних даних / В.В. Москаленко, А.С. Довбиш, А.С. Рижова // Вісник Сумського державного університету. Суми, Україна: СумДУ. 2013. №3. Р. 7-14. 4. *Lecoeuche S.* Modelling of non-stationary systems based on a dynamical decision space / S. Lecoeuche, G. Mercere, H. Amadou-Boubacar // 14th IFAC Symposium on System Identification. Newcastle, Australia : Elsevier Science. 2006. P.1222 – 1227. 5. *Довбиш А.С.* Основи проектування інтелектуальних систем : підручник / А.С. Довбиш. Суми, Україна : СумДУ. 2009. 171 p.

Поступила в редколлегию 26.09.2014

Москаленко Вячеслав Васильевич, канд. техн. наук, ассистент кафедры компьютерных наук Сумского государственного университета. Научные интересы: интеллектуальные системы управления слабоформализованными процессами. Адрес: Украина, 40035, Сумы, ул. Н.-Сыроватская, 66, кв. 84, м.т. +380664291318, e-mail: systemscoders@gmail.com.

Рижова Алёна Сергеевна, аспирантка Сумского государственного университета. Научные интересы: машинное обучение и распознавание образов. Адрес: Украина, 42303, Сумской район, с. Стецковка, ул. Школьная, 17, м.т. +38(095) 738-44-74, e-mail: alenarizhova@gmail.com.

УДК 681.518.5

А.С. ШКИЛЬ, Г.П. ФАСТОВЕЦ, А.С. СЕРОКУРОВА

АВТОМАТИЗАЦИЯ ПОИСКА ОШИБОК ПРОЕКТИРОВАНИЯ В HDL-МОДЕЛЯХ КОНЕЧНЫХ АВТОМАТОВ

Предлагается автоматизация диагностирования HDL-моделей конечных автоматов с использованием программы ASFTEST. Рассматривается вариант восстановления графа переходов по HDL-модели автомата в форме автоматного шаблона и анализ обхода всех дуг графа для поиска ошибок проектирования.

1. Введение

При проектировании операционных или управляющих устройств с использованием конечных автоматов алгоритм их функционирования, как правило, задается или в виде граф-схемы алгоритма (flow chart), или путем описания функции выходов в словесном или табличном виде. При автоматизированном проектировании подобных устройств их описание на языке аппаратуры (HDL) создается в форме автоматного шаблона, т.е. специальной структуры HDL-кода, которая строится на основе графа переходов автомата (state diagram). Переход от других способов описания закона функционирования конечного автомата к его графу переходов является искусством проектировщика и подробно описан в [1].

Одним из важных этапов автоматизированного проектирования цифровых устройств является верификация HDL-модели, т.е. определение соответствия полученного HDL-кода заданной спецификации. Один из традиционных подходов к верификации и диагностированию HDL-моделей состоит в следующем. С учетом выбранного стиля описания составляется список ошибок проектирования и для них строятся тесты. HDL-модель компилируется, и в ней устраняются синтаксические ошибки. Полученная HDL-модель моделируется в среде верификации на построенных тестах, и результат сравнивается с эталоном, который получен на основе спецификации. Если результат не совпал с эталоном, то выполняется диагностический эксперимент (ДЭ) по поиску места нахождения ошибки проектирования с последующим исправлением HDL-кода, и повторяется процедура верификации [2].

Возможные ошибки проектирования в HDL-моделях определяются стилем описания HDL-кода. Под ошибкой проектирования понимается ошибка в HDL-операторе, которая не относится к классу синтаксических и нарушает алгоритм функционирования модели устройства, заданный спецификацией.

Типы ошибок проектирования :

- «замена оператора» (логического или арифметического),
- «замена операнда» (в операторе назначения или условном).

Выделение фрагментов HDL-кода, описывающих поведение конечных автоматов стилем «автоматный шаблон», позволяет определить ошибки проектирования типа «неправильный переход в графе переходов автомата», что соответствует ошибке:

- в выборе текущего состояния в операторе when,
- в выборе следующего состояния в функции переходов (a_i вместо a_j),
- в операторе if() при анализе входного сигнала;
- в назначении выходного сигнала.

В [3] предложен «ручной» подход к поиску ошибок проектирования в автоматных HDL-моделях цифровых устройств. Но данный подход, во-первых, предусматривает наличие спецификации в виде графа переходов, а во-вторых, применим только для достаточно несложных HDL-моделей в форме автоматного шаблона. Таким образом, необходимо разработать автоматизированный метод поиска ошибок проектирования в HDL-моделях цифровых автоматов при задании спецификации в произвольном виде.

2. Подготовка к проведению диагностического эксперимента

Предлагается следующий подход к диагностированию HDL-моделей (локализации места возникновения ошибок проектирования в них). Имеется спецификация в словесной форме или в виде таблицы выходов. На ее основе построен HDL-код в форме автоматного шаблона. По HDL-коду с помощью специальных инструментальных средств (например, Code2Graphics в составе Active-HDL) восстанавливается граф переходов, соответствующий HDL-коду (с возможными ошибками). С помощью инструмента автоматического построения тестов (например, ASFTEST в составе Active-HDL) подготавливается диагностический эксперимент над HDL-моделью автомата (проверка достижимости вершин с использованием линий сброса, обход всех дуг, обход всех вершин) путем построения теста в форме TestBench и маршрутов обхода путей. ДЭ проводится с использованием инструментальных средств системы моделирования HDL-кода Active-HDL. По результатам прохождения теста и сравнения с эталоном спецификации (функцией выходов) строится вектор экспериментальной проверки (ВЭП), и путем анализа таблицы путей находится ошибочная дуга (дуги) в графе. При этом ошибка в HDL-коде, вероятнее всего, находится во фрагменте кода автоматного шаблона, связанного с вершиной (состоянием), откуда исходит ошибочная дуга.

В качестве элементарной проверки P_i при проведении ДЭ используется реализация определенного маршрута обхода графа, при этом номер маршрута (строка матрицы путей) соответствует номеру элементарной проверки. Результат элементарной проверки v_i считается отрицательным, если терминальная (конечная) вершина на этом маршруте достигнута, в противном случае результат элементарной проверки считается положительным:

$$v_i = \begin{cases} 0 \rightarrow \text{если на этом маршруте обхода выход совпадает с эталоном (тест прошел);} \\ 1 \rightarrow \text{если на этом маршруте обхода выход не совпадает с эталоном (тест не прошел).} \end{cases}$$

ВЭП для диагностического эксперимента определяется $V = (v_1, v_2, \dots, v_m)$, где m – число элементарных проверок P_i .

Если предположить наличие в схеме одиночной ошибки проектирования, то по результатам ДЭ область подозреваемых дефектов формируется на основе пересечения строк таблицы путей, отмеченных единичными значениями в ВЭП, по формуле:

$$D = \bigcap_{v_j=1} M_j - \bigcup_{v_j=0} M_j, \quad (1)$$

где M_j – j -я строка таблицы путей.

Если результат вычислений по формуле (1) оказался пустой, то предполагается, что в схеме присутствует кратная ошибка проектирования, и область подозреваемых ошибок определяется объединением строк, соответствующих единичным значениям координат ВЭП по формуле [4]:

$$D = \bigcup_{v_j=1} M_j - \bigcup_{v_j=0} M_j. \quad (2)$$

Для проведения ДЭ по поиску ошибок проектирования используется VHDL-модель автомата Мили, который выполняет распознавание набора 1100 во входной битовой последовательности. На рис.1 представлен фрагмент архитектуры двухпроцессного автоматного шаблона HDL-модели без ошибок проектирования (хотя в реальности такой эталонной модели при проведении ДЭ не существует).

```

begin
p1: process (state, data)
begin
  case state is
    when a0 =>
      if data='0' then nextstate<= a0; y<='0';
        else nextstate<= a1; y<='0';
      end if;
    when a1 =>
      if data='1' then nextstate<= a2; y<='0';
        else nextstate<= a0; y<='0';
      end if;
    when a2 =>
      if data='0' then nextstate<= a3; y<='0';
        else nextstate<= a2; y<='0';
      end if;
    when a3 =>
      if data='0' then nextstate<= a0; y<='1';
        else nextstate<= a1; y<='0';
      end if;
  end case;
end process;
p2: process (clk,reset)
begin
if reset='1' then state <= a0;
elsif clk'event and clk = '1' then state <= nextstate;
end if;
end process;

```

Рис. 1. Фрагмент эталонной VHDL-модели автомата

Используя программный инструмент Code2Graphics в составе Active-HDL [5], восстанавливаем граф переходов данного автомата (рис.2).

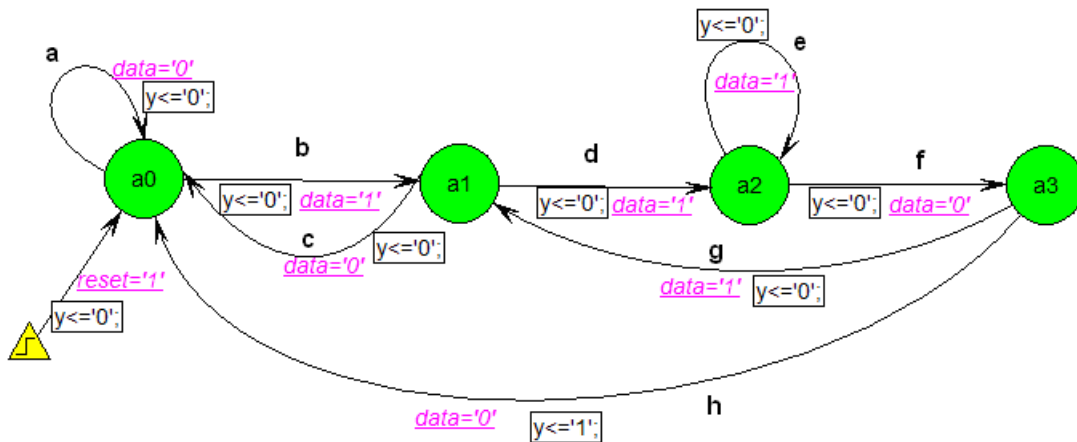


Рис. 2. Восстановленный граф переходов в диаграмме состояний Active-HDL

Для данного графа с помощью инструмента автоматического построения тестов ASFTEST в составе Active-HDL автоматически генерируются тестовые наборы в форме TestBench. Входной информацией является содержательный граф автомата, представленный в формате ASF. В результате работы ASFTEST формируются минимизированные тестовые последовательности в виде файла на языке VHDL (Verilog) для проверки правильности (в англоязычной литературе используется термин – validation) и верификации проекта с помощью программ моделирования на языках VHDL и Verilog. Возможна генерация теста по 3 различным стратегиям, в зависимости от целей моделирования и необходимой полноты теста. ASFTEST также выполняет анализ модели на корректность, генерирует описание на языке VHDL, генерирует макрокоманды для среды Active-HDL, фиксирует в файле отчета статистическую информацию [6].

При компиляции *.asf-файла генерируется модель автомата на языке VHDL, но при данном подходе к проведению ДЭ не используется.

Запуск программы автоматической генерации тестов осуществляется с помощью следующей командной строки:

```
asftest.exe -in [path]\fsm.asf – tb2 fsm_tb2.vhd,
где [path]\ - путь к файлу, fsm.asf – имя входного файла.
```

В результате работы программы ASFTEST генерируются следующие файлы:

fsm_asft.cov – файл отчета, содержит информацию о покрытии состояний и путей при каждой стратегии;

fsm_asft.err – файл отчета об ошибках. Ошибок не содержится. Файл пуст;

fsm_asft.rpt – файл статистики, содержит информацию об обрабатываемом графе;

fsm_gen.vhd – файл описания автомата на языке VHDL;

fsm_tb2.vhd – файл, содержащий сгенерированную по стратегии 2 (обход всех дуг графа переходов автомата) тестовую последовательность на языке VHDL (рис.3);

fsm_tb2.do – файл с макрокомандами для моделирования теста из файла fsm_tb2.vhd.

```
architecture fsm_arch_tb2 of fsm_ent_tb2 is
...begin
... begin
  clk <= '0'; cycle_num <= 0;
  wait for delay_wr_in;
  data <= '0'; reset <= '1';
  wait for delay_pos_edge;
  test_sreg <= a0; clk <= '1';
  wait for delay_wr_out;
  wait for delay_neg_edge; — a0
...end architecture fsm_arch_tb2;
...

```

Рис. 3. Фрагмент TestBench на языке VHDL для теста, полученного по стратегии 2

ASFTEST генерирует файл fsm_asft.cov с описанием всех дуг графа переходов (рис.4), что в дальнейшем значительно упрощает создание таблицы путей.

```
28=(a0->a0)31=(a0->a1)33=(a1->a2)35=(a1->a0)
38=(a2->a3)40=(a2->a2)42=(a3->a0)45=(a3->a1)48=(@any_state->a0)
```

Рис. 4. Листинг файла fsm_asft.cov по стратегии 2

Каждая дуга в файле fsm_asft.cov имеет символическое (цифровое) имя, а присвоенные ниже для наглядности имена дуг графа эталонного кода показаны на рис.2:

```
a=(a0->a0), b=(a0->a1), c=(a1->a0), d=(a1->a2), e=(a2->a2),
f=(a2->a3), g=(a3->a1), h=(a3->a0).
```

Для упрощения изложения материала будем рассматривать автомат с тривиальной функцией выходов, т.е. выходной сигнал непосредственно связан с переходом (дугой). Это

упрощает наблюдаемость обхода графа автомата, но является частным случаем HDL-модели конечного автомата.

3. Проведение диагностических экспериментов

В качестве примеров применения изложенной стратегии проведем несколько ДЭ по поиску ошибок проектирования разных типов. Последовательность этапов проведения ДЭ следующая. По HDL-коду восстанавливается граф переходов, по графу строится тест, тест моделируется и по waveform выполняется сравнение результатов моделирования с эталоном (спецификацией). На основании результатов сравнения строится ВЭП и по нему по формулам (1) и (2) находятся ошибочные дуги в графе. После этого выполняется визуальная инспекция HDL-кода и находится ошибочный оператор.

Вначале рассмотрим вариант HDL-модели, где заведомо отсутствуют ошибки проектирования (см.рис.2). Такой вариант кода в дальнейшем будем называть исправным.

На рис.5 приведены результаты моделирования теста по стратегии 2 в Active-HDL для HDL-модели исправного автомата.

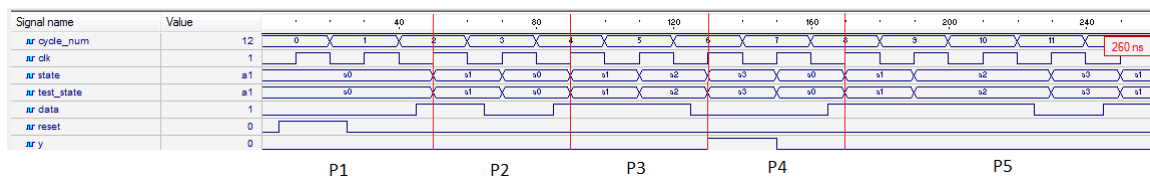


Рис. 5. Waveform с временными границами элементарных проверок P_i для исправной HDL-модели автомата

Результаты проведения ДЭ представим в виде таблицы путей (маршрутов) обхода графа, строками которой являются элементарные проверки, а столбцами – имена дуг графа переходов. Кроме того, в данной таблице представлены реальное и эталонное значение функции выходов y и ВЭП (рис.6). Отметим, что столбец $y_{эт}$ связан со спецификацией и используется для получения ВЭП.

	data	a	b	c	d	e	f	h	g	y	$y_{эт}$	ВЭП(V)
P_1	0	1								0	0	0
P_2	10		1	1						0	0	0
P_3	11		1		1					0	0	0
$P_3 + P_4$	1100		1		1		1	1		1	1	0
$P_3 + P_5$	11101		1		1	1	1		1	0	0	0

Рис. 6. Маршруты обхода исправного графа переходов и ВЭП

Анализируя таблицу путей на рис.6, можно сделать два вывода:

- 1) в тесте присутствует последовательность 1100 (элементарные проверки $P_3 + P_4$), дающая значение 1 на выходе, что соответствует спецификации;
- 2) для исправного кода ВЭП=0, что соответствует отсутствию ошибок в HDL-коде.

Рассмотрим вариант ошибочного HDL-кода с ошибкой в выборе текущего состояния (рис.7) (в операторе when вместо состояния a2 задано a4, что помечено серым цветом).

```
when a4 => (вместо when a2)
    if data='0' then      nextstate<= a3; y<='0';
    else nextstate<= a2; y<='0';
    end if;
```

Рис. 7. Фрагмент VHDL-модели автомата с ошибочным оператором when

Ошибка такого вида не пропускается компилятором, так как a4 – это необъявленное состояние. Поэтому, такая ошибка проектирования обнаруживается еще на этапе компиляции VHDL-модели.

Рассмотрим вариант ошибочного кода с ошибкой в выборе следующего состояния. На рис.8 приведен фрагмент VHDL-модели рассматриваемого автомата с ошибочным оператором назначения нового состояния a3 вместо a2 (помечено в коде серым цветом).

```

when a2 =>
  if data='0' then      nextstate<= a3; y<='0';
  else nextstate<= a3; y<='0'; (вместо nextstate <= a2)
  end if;
  
```

Рис. 8. Фрагмент VHDL-модели автомата с ошибочным оператором назначения

Моделирование кода с внесенной ошибкой встроенным инструментом Code2Graphics генерирует граф автомата (рис.9), новый asf-файл, из которого генерируются тесты. Необходимо отметить, что в Active-HDL граф-модель автомата является более сложной, чем модели Мили и Мура (на рис.9 выходной сигнал автомата Мили привязан к выходу из вершины графа переходов), но это не влияет на результаты проведения ДЭ.

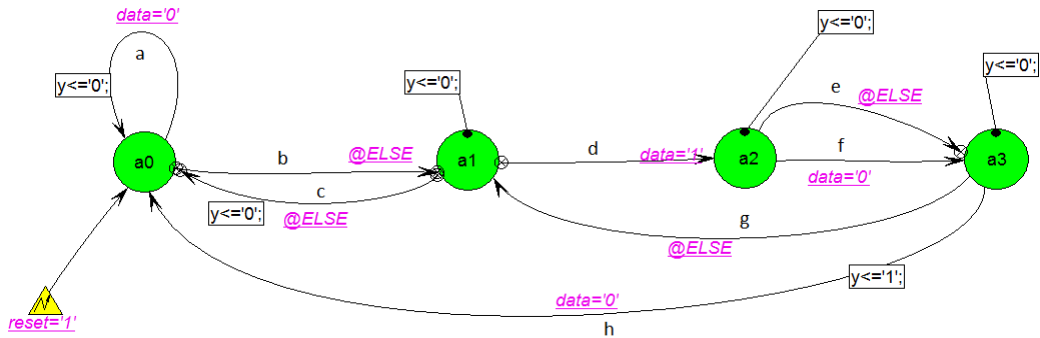


Рис. 9. Граф переходов с ошибочным оператором назначения нового состояния a3

Результаты моделирования тестов в среде Active-HDL приведены на рис. 10.

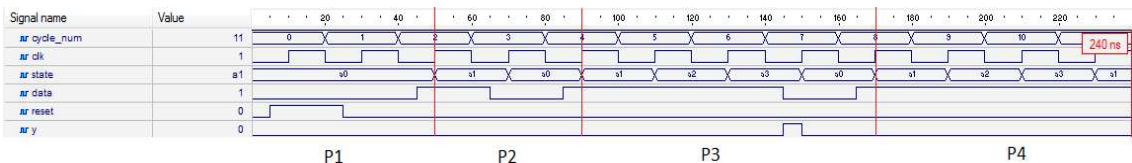


Рис. 10. Результаты моделирования автомата с ошибочным оператором назначения

Результатом проведения ДЭ по обходу графа является вектор экспериментальных проверок, в котором на P3 произошло несовпадение с эталоном. Исходя из спецификации, у может быть равен 1 только при подаче на вход последовательности 1100, соответственно, а возникновение 1 на выходе при подаче 1110 неверно, что свидетельствует об ошибке.

Таблица путей для данного ДЭ (маршруты обхода графа) и ВЭП представлены на рис.11. Отметим, что тестовые входные последовательности представлены в столбце data.

	data	a	b	c	d	e	f	h	g	y	y _{эт.}	ВЭП(V)
P1	0	1								0	0	0
P2	10		1	1						0	0	0
P3	1110		1		1	1		1		1	0	1
P4	1111		1		1	1			1	0	0	0

Рис. 11. Таблица путей для ошибочного графа переходов и ВЭП

Место возникновения ошибки в маршруте обхода графа переходов находится по формуле (1):

$$D = \{b, d, e, h\} - \{a\} \cup \{b, c\} \cup \{b, d, e, g\} = \{b, d, e, h\} - \{a, b, c, d, e, g\} = \{h\},$$

т.е. в графе переходов подозревается ошибочная дуга h .

Для нахождения места возникновения ошибки проектирования необходимо возвратиться к HDL-коду автоматного шаблона и выполнить визуальное инспектирование участка кода для вычисления ошибочной дуги. Анализ фрагмента кода, связанного с состоянием $a3$, показывает отсутствие ошибок. Попадание же в список подозреваемых ошибочных переходов дуги h связано с тем, что ASFTEST генерирует однократное покрытие обхода дуг графа и маршрута $b - d - f - h$ в списке обхода нет.

В этом случае надо возвратиться к предыдущему состоянию $a2$. Анализ фрагмента кода, связанного с состоянием $a2$, показывает наличие ошибочного оператора (`else nextstate <= a3`), данный оператор корректируется (`else nextstate <= a2`), и ДЭ по проверке корректности VHDL-кода повторяется.

Следующий вариант ошибочного кода – ошибка в операторе `if`, что выделено на рис. 12 жирным текстом.

```
when a3 =>
  if data='1' then (вместо data='0')
    nextstate<= a0; y<='1';
  else nextstate<= a1; y<='0';
  end if;
```

Рис. 12. Фрагмент VHDL-модели автомата с ошибочным оператором `if`

По результатам моделирования тестов в среде Active-HDL строим таблицу путей и ВЭП (рис. 13). Здесь и далее ошибочный граф переходов и сравнение графических результатов моделирования по waveform не приводятся ввиду громоздкости рисунков.

	data	a	b	c	d	e	f	h	g	y	y _{эт.}	ВЭП(V)
P ₁	0	1								0	0	0
P ₂	10		1	1						0	0	0
P ₃	11		1		1					0	0	0
P ₃ + P ₄	1101		1		1		1	1		1	0	1
P ₃ + P ₅	11100		1		1	1	1		1	0	1	1

Рис. 13. Таблица путей для ошибочного графа переходов и ВЭП

Место возникновения ошибки в маршруте обхода графа переходов будет:

$$D = \{b, d, f, h\} \cap \{b, d, e, f, g\} - \{a\} \cup \{b, c\} \cup \{b, d\} = \{b, d, f\} - \{a, b, c, d\} = \{f\},$$

т.е. подозревается ошибочная дуга f в графе переходов.

Инспекция фрагмента кода, связанного с исходящей вершиной $a2$, не находит ошибки проектирования. Данная ситуация состоит в следующем. Фрагмент ошибочного кода `if data='1'` порождает фактически две ошибочные дуги: g и h , которые, маскируя друг друга, приводят к ошибке в общей для них входящей дуге f . Это связано с особенностью построения маршрутов обхода графа в ASFTEST (переход $48=(@any_state->a0)$ на рис. 4 фактически является безусловным). В этом случае ошибку нужно искать в дугах, исходящих из $a3$. А там находится ошибочный фрагмент кода: `if data='1'`.

Следующий пример ошибочного кода связан с наличием ошибки в назначении выходного сигнала в состоянии $a3$, что показано на рис. 14 жирным текстом.

```
when a3 =>
  if data='0' then nextstate<= a0; y<='1';
  else nextstate<= a1; y<='1'; (вместо y<='0')
  end if;
```

Рис. 14. Фрагмент VHDL-модели автомата с ошибкой в назначении выходного сигнала

По результатам моделирования тестов в среде Active-HDL строим таблицу путей и ВЭП (рис.15).

	data	a	b	c	d	e	f	h	g	y	y _{эт.}	ВЭП(V)
P ₁	0	1								0	0	0
P ₂	10		1	1						0	0	0
P ₃	11		1		1					0	0	0
P ₃ +P ₄	1100		1		1		1	1		1	1	0
P ₃ +P ₅	11101		1		1	1		1		1	0	1

Рис. 15. Таблица путей для ошибочного графа переходов и ВЭП

Место возникновения ошибки в маршруте обхода графа переходов будет:

$$D = \{b, d, e, f, g\} - \{a\} \cup \{b, c\} \cup \{b, d\} \cup \{b, d, f, h\} = \{b, d, e, f, g\} - \{a, b, c, d, f, h\} = \{e, g\},$$

т.е. подозреваются ошибочными дуги e и g в графе переходов.

Анализ фрагмента кода, связанного с состоянием a₂ (дуга e), показывает отсутствие ошибок, а анализ фрагмента кода, связанного с состоянием a₃ (дуга g), показывает ошибку в назначении выходного сигнала y (else nextstate<= a1; y<='1').

Внесем в код сразу две ошибки: ошибку назначения состояния и ошибку в условии перехода. Ошибочный код приведен на рис.16, а результаты ДЭ приведены на рис.17.

```

when a2 =>
  if data='0' then
    nextstate<= a3; y<='0';
  else nextstate<= a3; y<='0'; (вместо nextstate <= a2)
  end if;
...
when a3 =>
  if data='1' then (вместо data='0')
    nextstate<= a0; y<='1';
  else nextstate<= a1; y<='0';
  end if;

```

Рис. 16. Фрагменты VHDL-модели автомата с ошибочным оператором назначения и ошибкой в условии перехода

	data	a	b	c	d	e	f	h	g	y	y _{эт.}	ВЭП(V)
P ₁	0	1								0	0	0
P ₂	10		1	1						0	0	0
P ₃	11		1		1					0	0	0
P ₃ +P ₄	1101		1		1	1		1		1	0	1
P ₃ +P ₅	1110		1		1	1		1		1	0	1

Рис. 17. Таблица путей для ошибочного графа переходов и ВЭП

Место возникновения ошибки в маршруте обхода графа переходов находится по формуле (1):

$$D = \{b, d, e, h\} \cap \{b, d, e, g\} - \{a\} \cup \{b, c\} \cup \{b, d\} = \{b, d, e\} - \{a, b, c, d\} = \{e\},$$

т.е. подозревается ошибочная дуга e. Анализируя фрагмент кода, связанный с a₂, находим ошибочный фрагмент кода else nextstate<= a3; y<='0'и исправляем его. После этого ДЭ

повторяется, и по аналогичному принципу находим вторую ошибку проектирования (результаты на рис.12 и 13).

Рассмотрим еще один вариант ошибочного кода с двумя ошибками проектирования: ошибка назначения выходного сигнала и ошибка в условии перехода. Ошибочный код приведен на рис.18, а результаты ДЭ приведены на рис.19.

```

when a0 =>
  if data='0' then
    nextstate<= a0; y<='1'; (вместо y<='0')
    else nextstate<= a1;y<='0';
  end if;
...
when a3 =>
  if data='1' then (вместо data='0')
    nextstate<= a0; y<='1';
  else nextstate<= a1; y<='0';
  end if;

```

Рис. 18. Фрагменты VHDL-модели автомата с ошибкой назначения выходного сигнала и ошибкой условия перехода

	data	a	b	c	d	e	f	h	g	y	y _{эт.}	ВЭП(V)
P ₁	0	1								1	0	1
P ₂	10		1	1						0	0	0
P ₃	11		1		1					0	0	0
P ₃ + P ₄	1101		1		1		1	1		1	0	1
P ₃ +P ₅	11100		1		1	1	1		1	0	1	1

Рис. 19. Таблица путей для ошибочного графа переходов и ВЭП

Место возникновения ошибки в маршруте обхода графа переходов пытаемся найти по формуле (1):

$$D = \{a\} \cap \{b, d, f, h\} \cap \{b, d, e, f, g\} - \{b, c\} \cup \{b, d\} = \emptyset - \{b, c, d\} = \emptyset.$$

Значит, требуется использовать формулу нахождения кратных ошибок (2):

$$D = \{a\} \cup \{b, d, f, h\} \cup \{b, d, e, f, g\} - \{b, c\} \cup \{b, d\} = \{a, b, d, e, f, g, h\} - \{b, c, d\} = \{a, e, f, g, h\}.$$

Подозреваются ошибочные дуги a, e, f, g и h в графе переходов.

Инспекция кода показывает следующее. Дуга a связана с фрагментом ошибочного кода в состоянии a0 (nextstate<= a0; y<='1'). А дуги {e, f, g, h} связаны с маскируемой ошибкой проектирования (if data='1'), нахождение которой подробно рассматривалось в комментариях к рис.13.

4. Выводы

Показана принципиальная возможность автоматизации поиска ошибок проектирования в HDL-моделях конечных автоматов в форме автоматного шаблона с использованием программного продукта ASFTEST.EXE в составе среды проектирования Active-HDL. В качестве примера использована модель конечного автомата Мили на языке VHDL и рассмотрены примеры поиска различных ошибок проектирования. Используемая стратегия позволяет находить область местонахождения ошибки проектирования в фрагментах HDL-кода, т.е. значительно сузить область визуального инспектирования кода.

Вопросы эквивалентности исправного и неисправных автоматов, выбора стратегий обхода графа, длины диагностического эксперимента, существования диагностических тестов и глубины диагностирования не рассматривались в данной работе и являются предметом дальнейших исследований.

Список литературы: 1. Баранов С.И. Синтез микропрограммных автоматов (граф-схемы и автоматы). 2-е изд., перераб. и доп. / С.И. Баранов Л.: Энергия, 1979. 232 с. 2. Шкиль А.С. Структурное и функциональное диагностирование HDL-моделей цифровых устройств в САПР РЭА / А.С. Шкиль, Е.Е. Сыревич, С. Альмадхоун, Г.П. Фастовец // Інформаційно-керуючі системи на залізничному транспорті. 2013. № 2. С. 75-82. 3. Альмадхоун С. Поиск ошибок проектирования в HDL-моделях цифровых автоматов / С. Альмадхоун, Е.Е. Сыревич, А.С. Шкиль // Вестник Херсонского государственного технического университета. 2013. №2 (46). С. 377-383. 4. Шкиль А.С. Методы поиска ошибок проектирования в HDL-коде / А.С. Шкиль, Е.Е. Сыревич, Д.Е. Кучеренко, Г.П. Фастовец // Радиоэлектроника и информатика. 2008. №. 3. С. 47-53. 5. Code2Graphics™ Converter [Электронный ресурс] / ALDEC. The design verification company. Режим доступа : www\ URL : https://www.aldec.com/en/products/fpga_simulation/active-hdl/feature/29 – 10.05.14. Загл. с экрана. 6. Хаханов В.И. Система генерации тестов для проектирования цифровых автоматов в среде ACTIVE-HDL / В.И. Хаханов, Е.В. Ковалев, В.В. Ханько, Масуд М.Д. Мехеди // АСУ и приборы автоматики. Харьков. 2000. Вып. 111. С. 15-22.

Поступила в редколлегию 20.09.2014

Шкиль Александр Сергеевич, канд. техн. наук, доцент кафедры АПВТ ХНУРЭ. Научные интересы: диагностика цифровых систем, дистанционное образование. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-26.

Фастовец Геннадий Павлович, канд. техн. наук, доцент кафедры АПВТ ХНУРЭ. Научные интересы: диагностика цифровых систем. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-26.

Серокурова Анна Сергеевна, аспирантка кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых автоматов. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. 702-13-26.

РЕФЕРАТИ

УДК 629.391

Концепція структурного стеганографічного кодування з маскуванням / Д.В. Бараннік, А.Е. Бекіров // АСУ та прилади автоматики. 2014. Вип.168. С. 4–11.

Розглянуто використання нерівновагового позиційного кодування як функціонального перетворення для числа з вбудованою інформацією. Обґрунтовано появу структурної надлишковості в процесі нерівновагового позиційного кодування. Запропоновано використовувати наявність потенційної надлишковості для стеганографічного вбудовування інформації. Розроблено стеганографічний метод на основі прямого та зворотного функціонального перетворення для нерівновагового позиційного числа з імплантованим елементом, який забезпечується вбудовування та вилученням. Створено правило вбудовування інформації для структурного стеганографічного кодування.

Л. 2. Бібліогр.: 4 назви.

УДК 629.391

The concept of structural steganographic encoding with masking / D. Barannik, A. Bekirov // Management Information System and Devices. 2014. N168. P.4-11.

In this article the using of nonequilibrium positional coding as the functional conversion for number with the embedded information is considered. Appearance of structural redundancy in the course of nonequilibrium positional coding is justified. It is offered to use existence of potential redundancy for steganographic embedding of information. The steganographic method on the basis of direct and inverse functional transformation for nonequilibrium positional number with the implanted element providing embedding and exception of the hidden information is developed. The rule of embedding of information for structural steganographic coding is created.

Fig. 2. Ref.: 4 items.

УДК 681.518

Розробка моделі конструктора WEB форм «Alvor form builder» та її реалізація / В.М. Левикін, А.А. Воронін, І.В. Горячевская // АСУ та прилади автоматики. 2014. Вип.168. С. 11–17.

Описано розроблений сервіс - конструктор веб-форм. Даний сервіс дозволяє створювати, зберігати і редагувати розроблені користувачами веб-форми, а також завантажувати файли розмітки і обробників.

Л. 9. Бібліогр.: 3 назви.

UDC 681.518

Model development Designer WEB forms «Alvor form builder» and its implementation / V. M. Levykin, A. A. Voronin, I. V. Karachevsky // Management Information System and Devices. 2014. N 168. P.11-17.

The developed tools – designer, web-forms and handlers to it. This service allows you to create, store and edit user developed web forms, and download files markup and handlers.

Fig. 9. Ref.: 3 items.

УДК 004.03

Оцінка ефективності інтеграційні рішення на основі сховищ триплетів / І.М. Галушка, С.С. Щербак // АСУ та прилади автоматики. 2014. Вип.168. С. 18–23.

Запропонована архітектура інформаційного простору територіально-розподіленого підприємства з вбудованою системою електронного документообігу на основі сховищ триплетів. Формалізовано процес і запропонована методика оцінки ефективності інтеграційних рішень на основі критеріїв тимчасових витрат для забезпечення можливості відстеження зміни продуктивності цих рішень залежно від використання різних типів транзакцій по обробці пов'язаних даних джерел. Запропоновано критерій оцінки ефективності інтеграційних рішень на основі пов'язаних даних, який базується на використанні показників тимчасових витрат, для забезпечення об'єктивної оцінки продуктивності цих рішень.

Л. 3. Бібліогр.: 13 назв.

UDC 004.03

Evaluation of efficiency of integration solutions based on triplets storages / I. Galushka, S. Shcherebak // Management Information System and Devices. 2014. N168. P.18-23.

The paper presents the architecture of information space of geographically distributed enterprise with built-in electronic document flow system based on triplets storages. The process is formalized and the technique for evaluating the effectiveness of integration solutions based on time-consuming criteria to enable tracking of change performance of these solutions based on the use of different types of transaction processing related data sources is proposed. The criterion for evaluating the effectiveness of integration solutions based on linked data based on time spent using indicators to ensure objective evaluation of the performance of these solutions.

Fig. 3. Ref.: 13 items.

УДК 621.315.592

Удосконалення математичної моделі розподілу легуючої домішки в процесі вирощування злитків кремнію. / А.В. Луговой, О.С. Притчин // АСУ та прилади автоматики. 2014. Вип.168. С. 24 – 29.

Показано, що легування є основним технологічним процесом, що забезпечує одержання напівпровідників заданого типу провідності, заданого питомого опору і заданої концентрації носіїв заряду. Виконано аналіз впровадження легуючої домішки в злиток при мінливій з часом швидкості росту злитка. Удосконалена модель легування злитка кремнію. Отримано уточнений аналітичний вираз для розрахунку легуючого профілю, який може використовуватися по діапазону типових параметрів росту злитка діаметром 150-300 мм.

Лл. 2. Бібліогр.: 5 назв.

UDC 621.315.592

Improvement of the impurities distribution model in the process of silicon growth / A.V. Lugovoy, O.S. Prytchyn // Management Information System and Devices. 2014. N168. P.24-29.

Doping is the main technological process for obtaining semiconductors of given type conductivity, resistivity and given concentration of charge carriers. As a result of the influence of a variety of complex identifiable factors on the process of doping of the melt distribution of ligature along the length of the ingot has a certain unevenness. In the paper an analysis of the introduction of the dopant in the ingot, with a time varying rate of growth of the ingot conducted. The model of silicon ingot doping improved. Accuracy of the analytical expression for the calculation of the doping profile, which can be used over a range of typical parameters of growth ingot with diameter of 150 - 300 mm increased.

Fig. 2. Ref.: 5 items.

УДК 621.315.59+546,681

Дослідження структурних і оптичних характеристик злитків напівізолюючих GaAs великого діаметра / А.П. Оксаніч, М.Г. Когдась, М.С. Андросюк // АСУ та прилади автоматики. 2014. Вип.168. С. 30 – 35.

Розглянуто питання удосконалення методу, методики та апаратури дослідження структурних і оптичних характеристик злитків GaAs. Визначено розподіл поглинання ІЧ-випромінювання по пластині GaAs діаметром 100мм і показується, що в напрямку <001> коефіцієнт поглинання відсутня, а у напрямку <011> зростає, що обумовлено формуванням аномальних оптичних островців по даному напрямку.

Лл. 2. Бібліогр.: 5 назв.

UDC 621.315.59+546.681

The study of structural and optical characteristics of ingots seminsulating GaAs large diameter / A.P. Oksanych, M.G. Cogdas, M.S. Androsiuk // Management Information System and Devices. 2014. N 168. P.30-35.

They discussed the issues of improvement of the method and apparatus of the study of structural and optical characteristics of GaAs ingots. Determined by the distribution of absorption of IR radiation by GaAs wafer with a diameter of 100mm and it is shown that in the direction <001>, the coefficient of absorption is absent, and in the direction of <011> increases, due to the formation of anomalous optical islets in this area.

Fig. 2. Ref.: 5 items.

УДК 681.518:004.93.1'

Інтелектуальна система підтримки прийняття рішень з оптимізацією просторово-часових параметрів функціонування / В.В. Москаленко, А.С. Рижова // АСУ та прилади автоматики. 2014. Вип. 168. С. 36-43.

Розглянуто алгоритм оптимізації просторово-часових параметрів функціонування інформаційно-екстремальної системи підтримки прийняття рішень для керування нестационарним технологічним процесом. Запропоновано визначати з малими обчислювальними затратами межі квазістационарних часових інтервалів спостереження, використовуючи нормовані статистики числа потраплянь ознак розпізнавання у свої поля контрольних допусків.

Лл. 5. Бібліогр.: 5 назв.

UDC 681.518:004.93.1'

Intelligent decision support system with optimization of time-spatial parameters of its functioning / V.V. Moskalenko, A.S. Righova // Management Information System and Devices. 2014. N 168. P. 36-43.

In this article the algorithm of the time-spatial parameters optimization of functioning information-extreme Decision Support System for control of non-stationary technological process is considered. Determination of boundaries of quasi-stationary time-intervals of observation with small computational expenses using normalized statistics of number of occurrences features into its receptive fields is proposed.

Fig. 5. Ref.: 5 items.

УДК 681.518.5

Автоматизація пошуку помилок проектування в HDL-моделях кінцевих автоматів/ О.С. Шкіль, Г.П. Фастовець, А.С. Сірокурова // АСУ та прилади автоматики. 2014. Вип. 168. С.43-52.

Запропонована автоматизація діагностування HDL-моделей кінцевих автоматів з використанням програми ASFTEST. Розглянуто варіант відновлення графа переходів по HDL-моделі у формі автоматного шаблону та аналіз обходу усіх дуг графа для пошуку помилок проектування.

Л. 19. Бібліогр.: 6 назв.

UDC 681.518.5

Search automation of design errors in the HDL-models of finite machines \ A.S. Shkil, Г.П. Фастовець, А.С. Sirokurova // Management Information System and Devices. 2014. N 168. P.43-52.

Proposed automation of diagnosing HDL-models of finite state machines using the program ASFTEST. Consider the option restoring the transition graph of HDL-model of machine in the form of an automaton pattern and analysis of bypass all the arcs to find design errors.

Fig. 19. Ref.: 6 items.

УДК 004.056 (043.2)

Метод вбудовування стегосообщенія на основі ключового елемента / В.Г. Бабенко, В.М. Зажом, О.Б. Нестеренко // АСУ та прилади автоматики. 2014. Вип. 168. С. 53-58.

Розроблений стеганографічний метод вбудовування інформації, здійснюваний на основі використання випадково певного ключового елемента порожнього контейнера, значення якого забезпечує вибір способу вбудовування повідомлення в контейнер. Наведено опис етапів виконання стеганографічного перетворення для розробленого методу вбудовування інформації. Наведено формальну модель стегосистеми, що базується на використанні даного методу вбудовування інформації на основі ключового елемента. Визначено ряд переваг і недоліків розробленого методу вбудовування повідомлення в стегоконтейнер. Основною перевагою даного методу є відсутність необхідності передачі контейнера-оригіналу для відтворення прихованого повідомлення з стегоконтейнера.

Бібліогр.: 7 назв.

УДК 004.056 (043.2)

The method of embedding message based on the key element / V.G. Babenko, V.M. Zazhoma, O.B. Nesterenko // Management Information System and Devices. 2014. N 168. P.53-58.

This paper is designed steganographic method of embedding information, carried out through the use of randomly identify the key elements of the empty container, the value of which provides a variety of ways to embed the message in the container. The description of the stages of steganography conversion of developed method of embedding information. Shown stegosystem formal model based on the use of this method of embedding information based on the key element. Identified a number of advantages and disadvantages of this method of embedding the message in stegocontainer. The main advantage of this method is no need to transfer the original container, to reconstruct a hidden message from stegocontainer.

Ref.: 7 items.

УДК 629.391

Оцінка ефективності методів стеганографічного вбудовування інформації в спектральну область зображень / Г.Ф. Коначович // АСУ та прилади автоматики. 2014. Вип. 168. С. 59–63.

Розглянуті методи стеганографічного вбудовування інформації в спектральну область зображення контейнера. Проведено аналіз існуючих стеганографічних методів. Розглянуто показники ефективності функціонування стеганографічних методів для скритного вбудовування інформації. Проведена оцінка ефективності найбільш розповсюджених стеганографічних методів вбудовування в спектральну область.

Табл. 3. Бібліогр.: 7 назв.

UDC 629.391

Performance evaluation steganographic hiding information methods in the spectral region of the images / G.F. Konahovich // Management Information System and Devices. 2014. N 168. P. 59-63.

This article describes methods of steganographic embedding of hidden information in the spectral region of the image container. Analyzes of existing steganographic methods. The indicators of the of the steganographic methods of embed hidden information efficiency are considered. Assess the effectiveness of the most common methods of steganographic embedded in the spectral region.

Tab. 3. Ref.: 7 items.

УДК 004:519.713

MQT-автомат для аналізу великих даних / В.І. Хаханов, В.І. Обрізан, С.О. Зайченко, І.В. Хаханов // АСУ та прилади автоматички. 2014. Вип. 168. С. 64-72.

Запропоновано новий підхід векторно-логічної обробки великих даних з повним виключенням арифметичних операцій, що впливають на швидкодію і апаратну складність. Він може бути ефективно реалізований як на основі використання сучасних мультипроцесорних цифрових систем на кристалах, так і за допомогою віртуальних паралельних процесорів, що функціонують під егідою кіберфізичних систем або хмарних сервісів-фільтрів. Запропоновано модель обчислювального дискретного автомата, яка характеризується транзакційною взаємодією компонентів пам'яті, що виконують роль комбінаційних і послідовних елементів, реалізованих у формі кубітних або «квантових» примітивів, необхідних для створення паралельних віртуальних комп'ютерів і хмарно-орієнтованих процесорів.

Рис. 4. Бібліогр.: 9 назв.

UDC 004:519.713

MQT-automaton for big data analysis / V.I. Hahanov, V.I. Obrizan, S.A. Zaychenko, I.V. Hahanov // Management Information System and Devices. 2014. N 168. P. 64-72.

A new approach for vector-logical processing Big Data based on complete exception of arithmetic operations, which influent on the performance and hardware complexity, is proposed. It can be effectively implemented through the use of both modern multiprocessor digital systems on chips and virtual parallel processors of cyberphysical systems or cloud service-filters. A model of computing discrete automatonis offered. It is characterized by the transactional interaction of the memory components, which are combinational and sequential elements, implemented in the form of qubit or "quantum" primitives needed to create a parallel virtual computers and cloud-focused processors.

Fig. 4. Refs.: 9 items.