

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 174

**Харьков
2016**

СОДЕРЖАНИЕ

ЛЕВЫКИН В.М., ЮРЬЕВ И.А.

РАЗРАБОТКА МОДЕЛЕЙ ИНФРАСТРУКТУРЫ ПРЕДПРИЯТИЯ И
ИНФРАСТРУКТУРЫ ИНФОРМАЦИОННОЙ СИСТЕМЫ,
ИХ ОЦЕНКА И УСТАНОВЛЕНИЕ СООТВЕТСТВИЯ
МЕЖДУ ЭЛЕМЕНТАМИ МОДЕЛЕЙ.....4

ОКСАНИЧ И.Г., ПРИТЧИН О.С.

РЕАЛИЗАЦИЯ ИНФОРМАЦИОННОЙ ТЕХНОЛОГИИ МОНИТОРИНГА
РАСПРЕДЕЛЕНИЯ ПРИМЕСИ В СЛИТКАХ КРЕМНИЯ.....10

НАЙДА В.В., ШЕВЧЕНКО И.В., КОНОХ И.С.

ИНФОРМАЦИОННОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ОБРАБОТКИ
ВРЕМЕННЫХ РЯДОВ ТЕХНОЛОГИЧЕСКИХ СИГНАЛОВ
ДЛЯ ТЕСТИРОВАНИЯ УПРАВЛЯЮЩИХ АЛГОРИТМОВ
СИСТЕМ ПРОМЫШЛЕННОЙ АВТОМАТИЗАЦИИ.....15

TAMER BANI AMER, ХАХАНОВ И.В., ЛИТВИНОВА Е.И., ЕМЕЛЬЯНОВ И.В.

СИНТЕЗ И АНАЛИЗ КУБИТНЫХ МОДЕЛЕЙ ЦИФРОВЫХ СИСТЕМ.....24

ЮДИН А.К., БАРАННИК В.В., ФРОЛОВ О.В.

МЕТОД КОСВЕННОГО СТЕГАНОГРАФИЧЕСКОГО ВСТРАИВАНИЯ
НА ОСНОВЕ ФУНКЦИОНАЛЬНОГО ПРЕОБРАЗОВАНИЯ
ДЛЯ АДАПТИВНОГО ПОЗИЦИОННОГО ЧИСЛА.....42

БАРАННИК В.В., ХАХАНОВА А.В., СИДЧЕНКО С.А.

РАЗРАБОТКА ТЕОРЕТИЧЕСКИХ ОСНОВ СТРУКТУРНОГО
ПРЕДСТАВЛЕНИЯ ДАННЫХ В ДВОИЧНОМ ПОЛИАДИЧЕСКОМ
ПРОСТРАНСТВЕ.....50

БАРАННИК В.В., ПОДЛЕСНЫЙ С.А., ТУРЕНКО С.В.

ТЕХНОЛОГИЯ ЛОКАЛИЗАЦИИ ОШИБКИ ПУТЕМ
ПОЗИЦИОНИРОВАНИЯ КОДА ПЕРЕМЕННОЙ ДЛИНЫ.....54

РЫБИН В.В., КАКУРИН Н.Я.

ЭЛЕМЕНТАРНАЯ ТЕОРИЯ ТЯГОТЕНИЯ.....59

РЕФЕРАТЫ.....67

ПРАВИЛА ОФОРМЛЕНИЯ РУКОПИСИ.....70

РАЗРАБОТКА МОДЕЛЕЙ ИНФРАСТРУКТУРЫ ПРЕДПРИЯТИЯ И ИНФРАСТРУКТУРЫ ИНФОРМАЦИОННОЙ СИСТЕМЫ, ИХ ОЦЕНКА И УСТАНОВЛЕНИЕ СООТВЕТСТВИЯ МЕЖДУ ЭЛЕМЕНТАМИ МОДЕЛЕЙ

Рассматриваются вопросы оценки качества инфраструктуры информационной системы (ИИС) и инфраструктуры предприятия (ИП). Разрабатывается модель ИИС, основанная на применении процессного подхода. ИИС рассматривается как множество взаимосвязанных ИТ-процессов, которое можно декомпозировать на группы, что дает возможность возложить ответственность за функционирование конкретного ИТ-процесса на конкретных сотрудников. Также на основе ISO 9004:2000 разработана модель ИП, которая дополнена необходимыми компонентами. Устанавливается соответствие между моделями ИИС и ИП. Выделяются наиболее важные соответствия, наличие которых необходимо для правильного функционирования предприятия и его информационной системы.

Введение

Интенсивное внедрение информационных систем не всегда приводит к получению желаемых результатов в автоматизации бизнес-процессов. Это связано, в основном, с недостаточной подготовленностью инфраструктуры предприятия к работе в условиях постоянного развития информационных технологий. Определение и использование модели зрелости в оценке эффективности функционирования информационных систем позволяет руководству предприятий адекватно реализовывать современные методы управления, основанные на процессном подходе и информационных технологиях.

Прибыль и результативность деятельности предприятия непосредственно зависит от информационных систем и технологий, используемых на предприятии. От руководства требуется умение оценить пользу, которую они могут принести, знать, какие средства необходимо инвестировать в системы автоматизации обработки информации и как можно измерить полученные результаты.

1. Разработка модели инфраструктуры информационной системы.

Инфраструктура информационной системы (ИИС) представляет собой комплекс технических, программных, методических, организационных и других средств, обеспечивающих функционирование информационной системы и способствующих эффективной работе всех подразделений предприятия, повышению его конкурентоспособности.

Цели бизнеса и ИТ должны быть взаимосвязаны, что требует от руководства предприятия максимального внимания к организации и функционированию инфраструктуры. Четкое и сбалансированное взаимодействие бизнеса и ИТ реализуется методами эффективного управления, классическая модель которого представлена на рис. 1 [1].



Рис. 1. Классическая модель управления

На практике у многих предприятий такая модель управления не всегда приводит к желаемым результатам, что вынуждает их использовать более прогрессивные методы, а именно: управление бизнес-процессами или процессное управление. В основе процессного управления лежит понятие процесса – последовательность шагов, направленная на достижение определенной цели или результата [2], или последовательность исполнения работ (операций), направленных на создание результата, имеющего ценность для потребителя [1].

Инфраструктуру информационной системы (ИИС) можно рассматривать как множество ИТ-процессов. Их количество в ИИС определяется бизнес-целями предприятия и стратегиями их достижения (рис. 2).

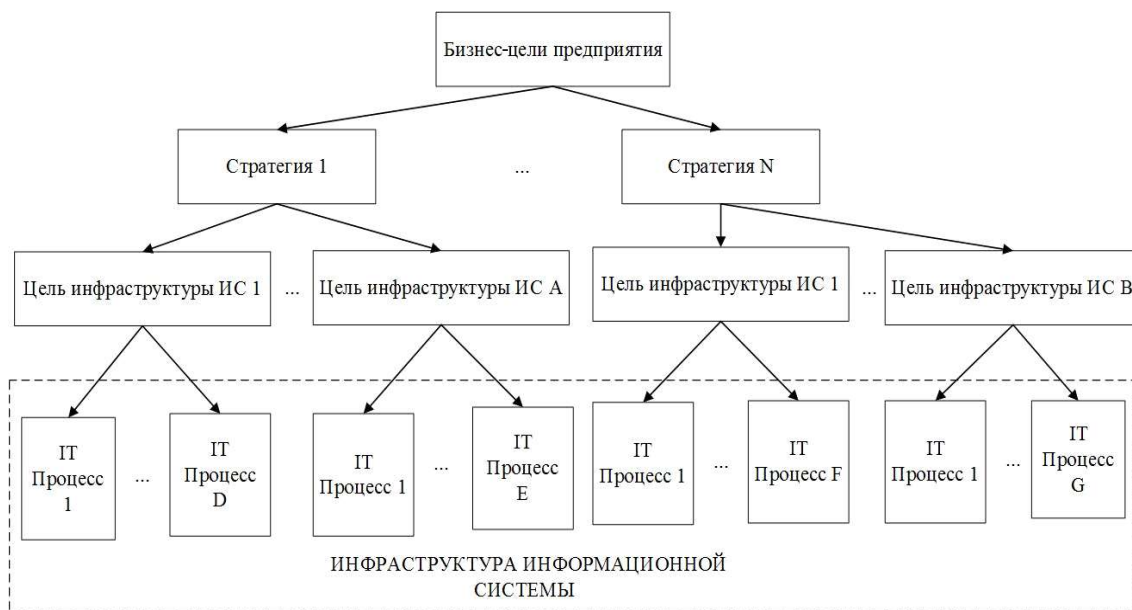


Рис. 2. Процессы в инфраструктуре информационной системы

Каждый ИТ-процесс появляется в связи с требованиями бизнеса, имеет свои входы и выходы, ответственных исполнителей (или владельцев) и средства измерения. ИТ-процессы характеризуются следующими свойствами [3, 4]:

- измеряемость, т.е. любой ИТ-процесс должен иметь количественную характеристику;
- наличие цели, т.е. любой ИТ-процесс должен служить для достижения конкретных результатов, связанных с бизнес-целью предприятия;
- наличие потребителей, т.е. любой ИТ-процесс должен предоставлять свои результаты конкретным потребителям или другим бизнес-процессам;
- наличие действий или видов деятельности, в результате выполнения которых ИТ-процесс может выдать результаты в соответствии с его целью, при этом критерием эффективности является достижение или недостижение результата.

Используя процессный подход в управлении, модель инфраструктуры информационной системы (M_{inf}) можно представить как множество ИТ-процессов P_i :

$$M_{inf} = \{P_i\}. \quad (1)$$

Каждый i -й ИТ-процесс можно представить как:

$$P_i = \{\{M\}, C_i, L_i, A_i, O_i\}, \quad (2)$$

где M – набор метрик; C_i – цель i -го ИТ-процесса, $i = 1, \dots, n$; L_i – потребитель или другой i -й ИТ-процесс, $i = 1, \dots, n$; A_i – действия или виды деятельности, производящиеся внутри i -го ИТ-процесса, $i = 1, \dots, n$; O_i – оценка i -го ИТ-процесса, $i = 1, \dots, n$.

Оценка ИТ-процессов является необходимой составляющей процессного управления. На основании оценки ИТ-процесса применяются управляющие воздействия в целях повышения качества процесса. Результат оценки всех ИТ-процессов, используемых в системе, может позволить оценить качество ИС предприятия.

Множество ИТ-процессов можно декомпозировать на группы, что дает возможность возложить ответственность за функционирование конкретного ИТ-процесса на конкретных сотрудников. В этом случае модель инфраструктуры ИС можно представить в виде:

$$M_{inf} = \{P(ts) \vee P(ns) \vee P(ds) \vee P(qs) \vee P(as) \vee P(ms) \vee P(mtds) \vee P(os) \vee P(js) \vee P(ss) \vee P(fs)\} , \quad (3)$$

где $P(ts)$ – процессы, поддерживающие техническое обеспечение ИС; $P(ns)$ – процессы, поддерживающие сетевое обеспечение ИС; $P(ds)$ – процессы, определяющие качество языков программирования и входящие в лингвистическое обеспечение ИС; $P(qs)$ – процессы, определяющие качество данных и входящие в информационное обеспечение ИС; $P(as)$ – процессы, определяющие качество программных продуктов и входящие в состав программного обеспечения ИС; $P(ms)$ – процессы, входящие в состав математического обеспечения ИС; $P(mtds)$ – процессы, определяющие качество сервисов и входящие в состав методического обеспечения ИС; $P(os)$ – процессы, входящие в состав организационного обеспечения ИС; $P(ss)$ – процессы, определяющие качество управления и входящие в состав кадрового обеспечения ИС; $P(js)$ – процессы, входящие в состав правового обеспечения ИС; $P(fs)$ – процессы, поддерживающие функциональность ИС.

2. Разработка модели оценки качества инфраструктуры информационной системы

На основании такого процессного представления о функционировании инфраструктуры ИС можно говорить о модели оценки ИИС (O_{inf}), которую можно представить в общем виде:

$$O_{inf} = \{O(P(ts)) \vee O(P(ns)) \vee O(P(ds)) \vee O(P(qs)) \vee O(P(as)) \vee O(P(ms)) \vee O(P(mtds)) \vee O(P(os)) \vee O(P(ss)) \vee O(P(js)) \vee O(P(fs))\} , \quad (4)$$

где $O(P(ts))$ – оценка качества процессов технического обеспечения ИС; $O(P(ns))$ – оценка качества процессов сетевого обеспечения ИС; $O(P(ds))$ – оценка качества процессов лингвистического обеспечения ИС; $O(P(qs))$ – оценка качества процессов информационного обеспечения ИС; $O(P(as))$ – оценка качества процессов программного обеспечения ИС; $O(P(ms))$ – оценка качества процессов математического обеспечения ИС; $O(P(mtds))$ – оценка качества процессов методического обеспечения ИС; $O(P(os))$ – оценка качества процессов организационного обеспечения ИС; $O(P(ss))$ – оценка качества процессов кадрового обеспечения ИС; $O(P(js))$ – оценка качества процессов правового обеспечения ИС; $O(P(fs))$ – оценка качества процессов, поддерживающих функциональность ИС.

Разработанная модель оценки ИИС, основанная на оценке компонентов инфраструктуры и процессном подходе, позволяет использовать полученные результаты при разработке и совершенствовании стратегий развития предприятия. Применение процессного подхода позволяет: определить проблемные процессы для каждого вида обеспечения ИС, определить ответственных за проблемный процесс и как следствие применить необходимое управляющее воздействие.

В отличие от классической модели управления, оценки, полученные с применением разработанной модели, дают возможность производить анализ ИТ-процессов и выработку управленческих решений с формированием необходимых управленческих воздействий. Но реализация этих управленческих решений и внедрение новых управленческих воздействий (усовершенствованных или обновленных) в целях повышения качества функционирования информационных систем требует определенных усилий от руководства, а также соответствующей подготовки персонала, потому что эффективная реализация новых методов управления на предприятии возможна при достижении им достаточного уровня зрелости.

Существующая практика эксплуатации информационных систем показывает, что для улучшения работы их инфраструктуры и повышения эффективности функционирующей информационной системы предприятие должно не только активно инвестировать в ИТ, но и

повышать уровень зрелости своей инфраструктуры так как качество IT-процессов напрямую зависит от уровня зрелости инфраструктуры предприятия.

3. Разработка модели оценки инфраструктуры предприятия

Основной задачей инфраструктуры предприятия является обеспечение доступности приложений для пользователей и поддержка роста бизнеса. В процессе своей деятельности компании решают различные задачи: выход на новые рынки, снижение себестоимости продукции, соблюдение государственных регламентных документов от ведения бухгалтерского учета до обработки персональных данных. Сотрудникам предприятия необходимо эффективно взаимодействовать между собой, с существующими и потенциальными заказчиками, своевременно обрабатывать множество данных. Для всего этого необходимо наличие соответствующей инфраструктуры предприятия (ИП).

Согласно ISO 9004:2000 [6] ИП включает ресурсы, такие как производственное помещение, рабочее пространство, средства труда и оборудование, вспомогательные службы, информационные и коммуникационные технологии, а также транспортные средства. Основываясь на таком определении, можно построить модель инфраструктуры предприятия:

$$M_{ip} = \{P, Ws, We, E, Ss, It, Ct, T\}, \quad (5)$$

где P – производственное помещение; Ws – рабочее пространство; We – средства труда; E – оборудование компьютерной сети; Ss – вспомогательные службы; It – информационные технологии; Ct – коммуникационные технологии; T – транспортные средства.

Однако данная модель не обладает достаточной полнотой, так как не учитывает важные компоненты инфраструктуры реального предприятия. Рассматривая объект управления необходимо говорить о взаимосвязи конечных пользователей (U), выполняющих основные бизнес-процессы (Bp) и соответствующей ИП. Поэтому бизнес-процессы выполняемые конечными пользователями обязательно должны входить в состав ИП. Важной частью инфраструктуры также являются финансы (F) и рабочая документация (D), они являются основой правовой и экономической деятельности предприятия. Также руководству предприятием необходимо иметь полную информацию о его функциональной структуре (FS). В данной модели информационные технологии (It) рассматриваются как компонент инфраструктуры предприятия, хотя более уместно рассматривать их в контексте компонента инфраструктуры информационной системы. Поэтому предлагается заменить It понятием IT-сервисы (ITs), что даст возможность более ёмко описать ИП. Дополнив модель ИП (5) вышеперечисленными компонентами, получим:

$$M_{ip} = \{P, Ws, We, E, Ss, ITs, Ct, T, Bp, Fi, FS, D\}. \quad (6)$$

На основании такого представления о функционировании инфраструктуры предприятия можно построить модель оценки качества ИП. Общая оценка строиться на основе всех компонентов инфраструктуры. В общем виде модель можно представить как:

$$M_{ip} = \{O(P), O(Ws), O(We), O(E), O(Ss), O(ITs), O(Ct), O(T), O(Bp), O(Fi), O(FS), O(D)\}, \quad (7)$$

где O(P) – оценка состояния производственных помещений, O(Ws) – оценка состояния рабочего пространства, O(We) – оценка качества средств труда, O(E) – оценка состояния оборудования, O(Ss) – оценка качества работы вспомогательных служб, O(Ct) – оценка качества коммуникационных технологий, O(T) – оценка состояния транспортных средств, O(ITs) – оценка качества существующих IT-сервисов, O(Bp) – оценка качества бизнес-процессов предприятия, O(Fi) – оценка финансового состояния предприятия, O(FS) – оценка функциональной структуры предприятия, O(D) – оценка качества рабочей документации.

Данная модель оценки качества ИП может быть использована для разработки стратегии развития ИП или ее совершенствования.

Инфраструктура предприятия и инфраструктура ИС тесно связаны. ИИС выдвигает соответствующие требования к ИП. Вследствие этого, если уровень ИП не соответствует требованиям ИИС, работа всего предприятия в целом может оказаться неэффективной. В связи с этим стоит задача установления соответствия между ИП и ИИС. Разработанные модели ИП и ИИС имеют схожую многокомпонентную структуру, поэтому задача установления соответствия между моделями является достаточно актуальной.

4. Определение соответствия между инфраструктурой информационной системы и инфраструктурой предприятия

Рассматривая предприятие в целом, необходимо установить соответствующие связи между его структурными элементами. Эти связи представлены на рис.3.

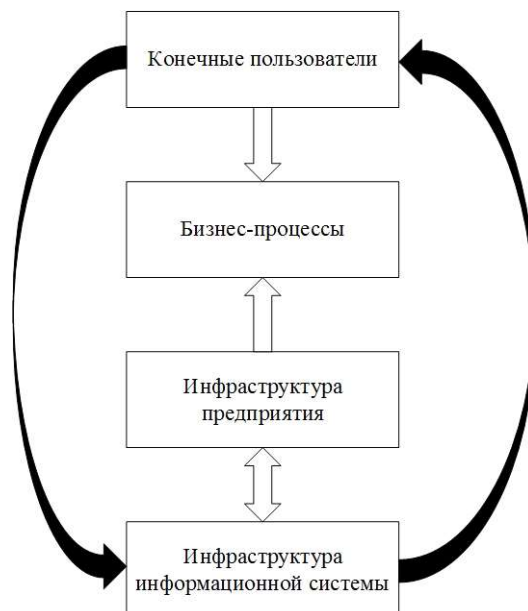


Рис. 3. Основные связи между структурными элементами предприятия

Сотрудники предприятия ежедневно пользуются сервисами, которые предоставляет ИС, поддерживая тем самым выполнение основных бизнес-процессов. Однако в определенный момент сотруднику может понадобиться наличие нового сервиса или изменение в старом. В связи с этим сотрудники могут периодически выдвигать новые требования к существующей ИС, что может повлечь за собой изменения в ее инфраструктуре. В зависимости от сложности и трудоемкости изменений, вносимых в ИС, могут потребоваться корректировки или расширения инфраструктуры предприятия. Поэтому крайне важно понимать какие компоненты ИИС соответствуют компонентам ИП. Наличие такого соответствия значительно упростит процесс внесения изменений в ИИС и разработку новых сервисов. В связи с этим актуальной задачей является установление соответствия между моделью ИП (6) и моделью ИИС (3).

Для начала необходимо выделить компоненты, которые присутствуют только в ИП и не связаны с информационной системой. Таковыми являются производственное помещение (P), рабочее пространство (Ws), средства труда (We), вспомогательные службы (Ss), транспортные (T) и финансовые (Fi) средства. Также необходимо учитывать, что некоторые компоненты ИИС есть локальными и не зависят от ИП, таковыми являются: лингвистическое обеспечение ИС(d), кадровое обеспечение ИС (ss), методическое обеспечение ИС (mtds) математическое обеспечение ИС(ms).

Исключив данные составляющие из инфраструктуры предприятия и инфраструктуры информационной системы, можно установить следующие соответствия:

1) $js \rightarrow D$, процессы, поддерживающие правовое обеспечение ИС, должны выполняться на основании рабочей документации предприятия и полностью ей соответствовать;

2) $fs \cup os \rightarrow FS$, процессы, поддерживающие функциональность ИС и организационное обеспечение ИС, тесно связаны, поэтому их совокупность должна обеспечить поддержку функциональной структуры предприятия на всех уровнях. Организационная структура управления на предприятии определяет состав и функции управления структурных подразделений;

3) $ns \in \rightarrow E \cup Ct$, процессы, поддерживающие сетевое обеспечение ИС, должны соответствовать уровню и качеству оборудования компьютерной сети и коммуникационных технологий, которые используются на предприятия;

4) $qi \rightarrow BP$, процессы, поддерживающие информационное обеспечение ИС, должны соответствовать требованиям поддержки основных бизнес-процессов для максимального сокращения сроков предоставления той или иной информации, а также повышения качества и доступности необходимой информации;

5) $as \cup ms \rightarrow ITs$, процессы, поддерживающие математическое и программное обеспечение ИС, формируют основу для разработки новых и совершенствования существующих IT-сервисов предприятия, поэтому необходимо, чтобы качество математического и программного обеспечения ИС соответствовало требованиям существующих IT-сервисов;

Исходя из установленных соответствий, можно сделать вывод о том, что наиболее трудоемким является:

1) установление соответствия между сетевым обеспечением ИС и оборудованием компьютерной сети предприятия. Необходимо, чтобы оборудование компьютерной сети соответствовало установленным требованиям, выдерживало общую нагрузку, формируемую информационной системой;

2) установление соответствия между процессами, поддерживающими информационное обеспечение ИС, и основными бизнес-процессами предприятия, что требует автоматизации производственных процессов предприятия;

3) установление соответствия между совокупностью процессов, поддерживающих программное и математическое обеспечение ИС, и совокупностью спецификаций рабочих станций предприятия;

4) установление соответствия между существующей функциональной структурой предприятия и процессами функционирования ИС и ее организационным обеспечением.

Выводы

В отличие от классического подхода к определению ИИС, разработанная модель ИИС основана на применении процессного подхода и позволяет описать инфраструктуру как структурированное множество процессов. Рассматривая ИИС как множество взаимосвязанных процессов, можно говорить об использовании сформированного множества процессов при формировании ИТ-стратегии предприятия и реализации его бизнес-целей. Разработанная модель ИП основана на ISO 9004:2000 и имеет многокомпонентную структуру. Как и в случае с моделью оценки ИИС, модель оценки ИП основана на оценке качества каждого элемента модели. Они должны соответствовать друг другу для обеспечения качественного функционирования предприятия.

Список литературы: 1. *Карабанов Б.М.* 5 шагов к процессному управлению. СПб., Инталев, 2002. 127с. 2. ITIL. The key to Managing IT services Office of Government Commerce. London: TSO.2005. 418с. 3. *Гузык С.* Стандарт CobiT. Управление и аудит информационных технологий. Особенности проведения внешнего аудита ИТ. Портал CIT Forum. [Электронный ресурс]. Режим доступа: http://citforum.ru/consulting/standart_cobit/article2003130.html 4. *Советов Б.Я., Водяхо А.И., Дубенецкий В.А., Цехановский В.В.,* Архитектура информационных систем. М: Издательский центр "АКАДЕМИЯ", 2012. 288 с. 5. *Аверченков В.И., Федоров В.П., Хейфец М.Л.* Основы математического моделирования технических систем: Учеб. пособие. Брянск: Изд-во БГТУ, 2004. 271 с. 6. ДСТУ ISO 9004:2000 Системи управління якістю. Рекомендації щодо поліпшення діяльності [Електронний ресурс]. К.: Держспоживстандарт України, 2008. Режим доступу : <http://www.document.org.ua>.

Поступила в редколлегию 18.03.2016

Левыкин Виктор Макарович, д-р техн. наук, профессор, зав. каф. ИУС ХНУРЭ. Научные интересы: разработка распределенных информационных систем Адрес: Украина, 61023, Харьков, пр. Науки, 14, тел. 702-13-26.

Юрьев Иван Алексеевич, аспирант кафедры ИУС ХНУРЭ. Научные интересы: методы, модели, информационные технологии управления информационными системами. Адрес: Украина, 61023, Харьков, пр. Науки, 14, тел. 702-13-26.

РЕАЛИЗАЦИЯ ИНФОРМАЦИОННОЙ ТЕХНОЛОГИИ МОНИТОРИНГА РАСПРЕДЕЛЕНИЯ ПРИМЕСИ В СЛИТКАХ КРЕМНИЯ

На основании информационной технологии мониторинга распределения примеси разрабатывается программный продукт, который реализует настройку математических моделей, входящих в состав системы виртуального мониторинга, непосредственно мониторинг условий в тепловом узле ростовой установки и позволяет выводить полученную информацию в виде таблиц и графиков.

1. Введение

Повышение качества монокристаллических слитков кремния требует более точного воспроизведения оптимальных условий выращивания. Для этого необходим непрерывный контроль распределения легировки в зоне кристаллизации и выращенной части слитка. Однако такой контроль возможен только косвенными методами, с привлечением специфических математических моделей, способных идентифицировать температурное поле расплава и слитка в реальном времени по результатам прямых измерений температуры в отдельных точках теплового узла, скорости вращения затравки и тигля и других параметров.

Целью данного исследования является разработка специализированного программного обеспечения для расчёта и мониторинга температурных полей в системе кристалл-расплав.

2. Постановка задачи

На основании информационной технологии мониторинга распределения примеси был создан программный продукт, который реализует настройку математических моделей, входящих в состав системы виртуального мониторинга, непосредственно мониторинг условий в тепловом узле ростовой установки и позволяет выводить полученную информацию в виде таблиц и графиков.

Программный комплекс предназначен для применения непосредственно в процессе выращивания монокристаллов, кроме того, он может быть использован при численных экспериментах по исследованию и оптимизации физических параметров теплового узла.

Отказ от использования готовых программных продуктов и создание собственного обусловлено рядом факторов – «тяжелые» программные пакеты, представленные на рынке, многофункциональны и ресурсоёмки, поскольку предназначены для решения широкого круга задач. Кроме того, в эти пакеты чрезвычайно трудно встраивать дополнительные процедуры и модули, необходимость в которых может возникнуть в процессе совершенствования системы мониторинга и системы управления процессом выращивания монокристаллов.

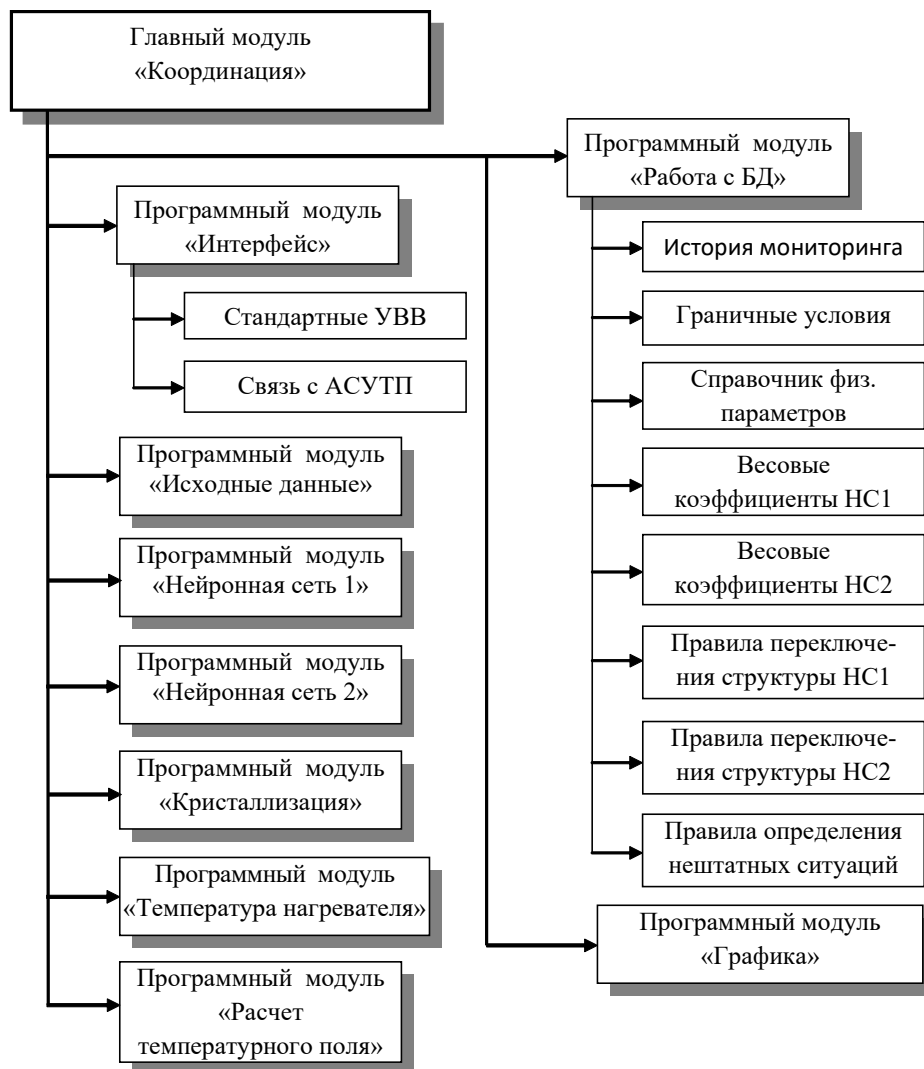
Программный комплекс позволяет задать:

- размеры и форму тигля и слитка;
- размеры, форму и положение экрана и нагревателя;
- температуропроводность кристалла и расплава и их начальные температуры;
- граничные условия первого, второго и третьего рода;
- ряд других параметров, таких как приведенные коэффициенты черноты слитка, экрана, тигля, коэффициент конвективного теплового потока во внешнюю среду и др.

Применяемая в расчете математическая модель является плоской и осесимметричной, т.е. рассматривает тепловой узел в радиальном разрезе, секущая плоскость которого проходит через ось симметрии узла. В модель включены: слиток; система тепловых экранов, включая нагреватель; расплав кремния и тигель. Тепловой экран считается активным и может составлять с нагревателем единое целое.

Важным элементом математической модели являются граничные условия, которые используются для сопоставления процесса моделирования определённым реальным условиям. Для повышения адекватности результатов расчетов в различных режимах и на различных стадиях выращивания применяются специальные нейронные сети.

Структура программного комплекса показана на рисунке.



Программный комплекс реализует следующие функции:

1. Инициализацию модели в соответствии с заданными параметрами.
2. Моделирование лучевого теплообмена.
3. Моделирование теплообмена за счёт температуропроводности и конвекции.
4. Определение предполагаемых формы и расположения фронта кристаллизации.
5. Предоставляет данные, полученные в результате моделирования в виде графиков и таблиц вертикального и горизонтального распределения температур, предполагаемых расположения и формы фронта кристаллизации.
6. Предоставляет возможность экспорта полученных данных в форматах, принятых в программах MS Excel и MathCAD.

Первая из задач решается методом конечных разностей (МКР), основанным на замене истинных значений производных, входящих в дифференциальное уравнение, приближёнными значениями в узлах сетки [1,2]. Расчетная схема построена с помощью метода переменных направлений [3]. Конечно-разностный подход является наиболее эффективным при решении квазистационарных задач. Простота записи конечно-разностных выражений и

отсутствие необходимости построения конечно-элементных сеток делают МКР весьма популярным среди многих пользователей. В данном случае применение МКР способствует ускорению расчетов и позволяет при небольших затратах вычислительных ресурсов сделать ряд «снимков» теплового поля в реальном времени технологического процесса.

Расчетная схема построена с помощью метода переменных направлений [4,5]. Он позволяет значительно сократить объем вычислений и получить вполне адекватные результаты при счете даже на сравнительно грубых сетках.

Процесс моделирования включает в себя итеративное исполнение всех этапов расчета. После каждой из таких итераций известны: текущее распределение лигатуры в теле монокристалла и предполагаемые размещение и форма фронта кристаллизации. Моделирование заканчивается после достижения требуемой точности, расчёт которой проводится анализом изменения распределения лигатуры в теле монокристалла на текущей итерации по сравнению с предыдущей, либо по достижении заданного числа итераций.

Для реализации пакета программ системы виртуального мониторинга была выбрана модульная структура. Она предусматривает наличие ряда элементов (модулей), которые составляют единый программный комплекс. Каждый из таких элементов решает определённую задачу или ряд родственных задач.

Основным преимуществом модульной структуры программного обеспечения по сравнению с монолитной структурой является возможность построения сложных систем за счёт распределения их на множество модулей, каждый из которых выполняет специфические функции. Кроме того, модульная архитектура упрощает командную разработку программных продуктов, поскольку предоставляет большее количество возможных вариантов распределения заданий между разработчиками. Однако наибольшим преимуществом модульной структуры является значительная расширяемость построенных на её основе программных продуктов, что становится возможным благодаря созданию новых модулей без необходимости доступа к программному коду уже существующих модулей.

Идеальным случаем использования модульной структуры программного обеспечения являются проекты, которые предусматривают некую начальную функциональность, однако в будущем будут расширяться за счёт решения новых задач и их интеграции с уже существующими. Данное преимущество модульной структуры особенно проявляется при доработке уже существующей системы другими разработчиками, которые могут не иметь полного представления о тонкостях внутреннего функционирования уже существующих модулей.

Таким образом, учитывая возможность дальнейшего расширения разрабатываемого программного обеспечения, модульная структура была выбрана как более перспективная. Особенно актуальным этот выбор становится с учётом того, что разрабатываемая система виртуального мониторинга распределения лигатуры является составной частью системы «Советчик мастера». Она разрабатывается другими разработчиками, которые, используя существующие модули, могут реализовать дополнительный функционал.

Разработанная подсистема «Виртуальный мониторинг» обеспечивает решение следующих основных задач:

1. Измерение температуры нагревателя с погрешностью $\pm 6^{\circ}\text{C}$.
2. Воссоздание температурного поля (абсолютных значений, $^{\circ}\text{C}$ и градиентов температуры по радиусу и по вертикальной оси, $^{\circ}\text{C}/\text{см}$) в подкристальной области и в слитке и вывод результатов измерений и вычислений на монитор оператора-технолога.
3. Вывод на монитор предполагаемой формы фронта кристаллизации в графическом виде.
4. Формирование предпосылок для прогнозирования вероятности превышения плотности дислокаций в выращиваемом слитке с учетом следующих факторов:
 - уровень радиальных и осевых градиентов в зоне фронта кристаллизации и в слитке;
 - соотношение диаметров слитка и тигля;
 - стадия процесса выращивания;
 - текущий уровень расплава в тигле;
 - скорость вытягивания, скорости вращения слитка и тигля.
5. Периодическое уточнение параметров моделей и алгоритмов прогнозирования.

6. Ведение базы данных подсистемы и ее синхронизация с централизованной базой данных.

Подсистема реализована на промышленном компьютере и интегрирована в автоматизированную систему управления ростовой установкой.

Внедрение подсистемы виртуального мониторинга температурных полей позволило более подробно исследовать влияние тепловых условий в зоне кристаллизации на уровень термоупругих напряжений в слитках GaAs.

Для минимизации уровня напряжений и соответственно плотности дислокаций идеальна ситуация, в которой существует постоянный осевой градиент температуры, тогда как радиальный градиент температуры близок к нулю. В этом случае термоупругие напряжения будут минимальными. Исследования показали, что осевой градиент температуры в слитке можно сделать практически постоянным путем выбора оптимального по размерам и положению теплового экрана. Радиальные градиенты температуры, однако, чрезвычайно зависимы от изменений режима оттока тепла от слитка в зоне фронта кристаллизации и герметизатора и являются причиной избыточных термоупругих напряжений.

Для подтверждения эффективности разработанных методов, моделей и информационных технологий мониторинга температурных полей в решении проблемы качества монокристаллов GaAs были проведены экспериментальные сравнительные исследования остаточных напряжений и плотности дислокаций в слитках кремния, выращиваемых на стандартной установке и модернизированной установке, в систему автоматизированного управления которой встроена подсистема виртуального мониторинга.

Остаточные напряжения исследовались по длине выращенных слитков использовалась автоматизированная установка измерения внутренних напряжений в пластинах GaAs «Полярон-2», реализующая метод SIRP и экспрессную методику определения структурного совершенства в пластинах GaAs.

Исследование остаточных напряжений проводилось на специально подготовленных шлифовано-травленных пластинах толщиной 450 ± 15 мкм, вырезанных из верхней, центральной и нижней части слитка.

При исследовании плотности дислокаций применялся автоматизированный телевизионный метод подсчета дислокаций. Суть этого метода состоит в подсчете числа ямок травления в поле зрения металлографического микроскопа, подключенного к телевизионной камере, изображение с которой вводится в компьютер и обрабатывается специальной программой.

Результаты измерений показали, что значения остаточных напряжений в слитках, выращенных на усовершенствованной ростовой установке «Арсенид-1М», снизились по сравнению со слитками, выращенными на установке «Арсенид-1», в верхней части слитка на 24 %, в средней части – на 32 % и в нижней – на 26 %. По мнению специалистов предприятия это стало возможным благодаря тому, что мастер-технолог использует результаты мониторинга для своевременного регулирования распределения лигатуры на разных стадиях выращивания.

3. Выводы

1. Из экспериментальных данных исследования плотности дислокаций по длине слитков, выращенных на ростовой установке «Арсенид 1М», можно сделать вывод о том, что применение системы виртуального мониторинга приводит к уменьшению остаточного напряжения и, как следствие, к уменьшению плотности дислокаций в слитке.

2. Разработано программное обеспечение для решения задач виртуального мониторинга распределения лигатуры и определения формы и расположения фронта кристаллизации.

3. Разработанный программный комплекс включает программные модули, реализующие задачи распределения лигатуры расплава, слитка и моделирование процесса кристаллизации, которые связаны между собой через главный модуль «Координация», обеспечивающий согласованную работу всех частей системы. Пользователь имеет возможность задавать геометрические и физические параметры элементов теплового узла ростовой установки, вводить фиксированные и начальные значения параметров режима выращивания, при необходимости проводить дообучение системы.

Список литературы: 1. *Каханер Д.* Численные методы и программное обеспечение : Уч. пособие / Д.Каханер, К. Моулера, С. Нэш. М.: Мир, 1998. 435 с. 2. *Джордж А.* Численное решение больших разреженных систем уравнений : Уч. пособие / А. Джордж, Дж. Лю. М. : Мир, 1984. 343 с. 3. *Тихонов А.Н.* Уравнения математической физики : Уч. пособие / А. Н. Тихонов, А. А. Самарский. М. : Наука, 1977. 735 с. 4. *Формалев В.Ф.* Метод переменных направлений с экстраполяцией по времени для параболических задач со смешанными производными / В. Ф. Формалев // Вычислительные технологии. 1996. Том 1. № 2. С. 99-103. 5. *Тихонов А. Н.* Уравнения математической физики : Уч. пособие / А. Н. Тихонов, А. А. Самарский. М.: Наука, 1977. 735 с.

Поступила в редколлегию 12.02.2016

Оксанич Ирина Григорьевна, канд.т. техн. наук, профессор кафедры информационно-управляющих систем КрНУ им. М. Остроградского. Научные интересы: информационные технологии Адрес: Украина, 39600, Кременчуг, ул. Першотравнева, 20, тел.: (05366) 30157. Email: kafius@kdu.edu.ua

Притчин Алексей Сергеевич, аспирант кафедры компьютерных и информационных систем КрНУ им. М. Остроградского. Научные интересы: информационные технологии Адрес: Украина, 39600, Кременчуг, ул. Першотравнева, тел.: (05366) 30157. Email: alpritchin@ukr.net

ИНФОРМАЦИОННОЕ И ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ОБРАБОТКИ ВРЕМЕННЫХ РЯДОВ ТЕХНОЛОГИЧЕСКИХ СИГНАЛОВ ДЛЯ ТЕСТИРОВАНИЯ УПРАВЛЯЮЩИХ АЛГОРИТМОВ СИСТЕМ ПРОМЫШЛЕННОЙ АВТОМАТИЗАЦИИ

Анализируются подходы к автоматизации тестирования управляющих программ систем промышленной автоматизации. Выделяются существующие недостатки и определены перспективные пути их устранения. Предлагается схема информационной технологии, обеспечивающей обработку, упорядоченное хранение и поиск временных рядов технологических сигналов для автоматизированного тестирования управляющих программ. Описывается метод классификации, распознавания и ускоренного поиска образов технологических сигналов. Разрабатывается алгоритм синтеза реакции динамического объекта управления на основании классифицированных образов.

Ключевые слова: пусконаладочные работы, АСУТП, идентификация, НПЛ.

1. Введение

В настоящее время в сфере промышленной автоматизации инструментальные программные и аппаратные средства достигли высокой степени совершенства. Это позволяет сократить временные и трудовые затраты при решении типовых задач автоматизации технологических процессов. К ним можно отнести:

- контроль параметров технологического объекта с заданной точностью;
- подключение датчиков и исполнительных устройств к управляющим устройствам;
- ввод-вывод дискретных и аналоговых сигналов, их обработка в реальном времени;
- проверка логических условий и блокировок, формирование дискретных управляющих сигналов;
- расчет аналоговых и дискретных управляющих воздействий по стандартным алгоритмам регулирования (П-, И-, ПИ-, ПИД-контроллеров);
- формирование питающего напряжения и управление скоростью и перемещением исполнительных приводов;
- интеграция средств автоматизации в сетевую структуру и обеспечение прохождения информационных потоков;
- решение интерфейсных задач (взаимодействие с пользователями);
- протоколирование сигналов и событий.

Большинство технологических установок, для которых проектируются системы автоматического управления, имеют свои особенности, которые необходимо учесть при составлении управляющих алгоритмов. На итоговое качество работы также влияет множество факторов, которые невозможно учесть заранее:

- нелинейность реакции объекта управления;
- нестабильность работы отдельных технических компонент;
- особенности настройки и работы программных компонент управляющих контроллеров;
- логические ошибки и непроработанность отдельных ситуаций в проектных алгоритмах, которые можно обнаружить только в процессе пуско-наладочных работ на объекте;
- требования заказчика на реализацию дополнительных функций, облегчающих решение технологических задач.

2. Анализ литературных источников и постановка задач исследования

В настоящее время можно выделить несколько подходов к тестированию и наладке промышленных систем управления, включающих в свой состав программируемые логические контроллеры (ПЛК).

Прежде всего – это компьютерное имитационное моделирование объекта управления и реализация функций расчета управляющих воздействий в специализированных программных пакетах. Подход отличается низкими издержками и универсальностью, однако требует соответствующей компетентности для синтеза моделей, отсутствием связи с программной и аппаратной частью реальной системы управления технологическим процессом (АСУТП). Также идеализация объекта управления и его сигналов не позволяет до проведения основных пуско-наладочных работ решить проблемы обработки сигналов системой управления и точно настроить управляющие алгоритмы.

Следующий подход можно рассматривать как один из этапов разработки программного обеспечения, заключающийся в эмуляции работы ПЛК на рабочей станции для исправления синтаксических ошибок и наиболее общей проверки работоспособности программы. Имеется возможность просматривать и модифицировать состояние программных переменных, что позволяет выполнить простые тесты, хотя и с большими затратами труда разработчиков.

Более совершенен подход с использованием специализированных средств эмуляции объектов управления FAT – Factory Acceptance Test simulator (симулятор для заводской приёмки), которые имеют различные свойства с точки зрения степени достоверности симуляции и хорошо интегрируются со средой разработки и эмуляции программ ПЛК. К ним можно отнести средства разработки – WinMOD, Mynah MiMiC, Siemens SIMIT, APROS, Process Simulator [1]. Преимуществом является более высокая достоверность тестирования сложных систем управления и приближенность тестируемого исходного кода к конечному релизу. Недостаток подхода – идеализация объекта управления и высокая трудоемкость разработки моделей технологического процесса.

Полунатурное моделирование представляет собой подход к исследованию устройства, реализованного в виде системы из натуральных элементов, сопрягаемых с остальной частью системы, реализованной в виде модели. Создание аппаратных блоков, максимально повторяющих элементы готового изделия и управляемых моделью с компьютера, позволяет исследовать и оценить воздействие факторов, моделировать которые невозможно или нецелесообразно. Программная модель при этом дает возможность исследовать множество различных алгоритмов работы и обработки информации. Часто различные алгоритмы обработки информации можно исследовать параллельно, что также сильно повышает эффективность работ [3, 4].

Преимущественно проектирование и разработка сложных систем управления осуществляется на основе двух принципов. При первом работоспособность системы исследуют и настраивают с помощью инструментов моделирования. Программное обеспечение для моделирования обеспечивает гибкость проверки системы при отработке различных сценариев. Второй принцип заключается в разработке проекта на конкретной платформе и оценке работоспособности оборудования системы в реальных условиях. При исследовании реализованных проектов нужно запускать как модель, так и само устройство, получить данные эксперимента и сравнить полученные результаты.

Преимущества этих подходов объединены в методе аппаратно-программного моделирования Hardware in the Loop (HIL). Методология HIL [2] обеспечивает промежуточный уровень контроля разрабатываемых систем, а именно между программным обеспечением для моделирования и тестированием оборудования проектируемой системы.

Для реализации полунатурного моделирования необходимо организовать обмен информацией между численной моделью, реализуемой на компьютере, и реальным объектом, работающим в реальном времени. Обмен информацией физически реализуется с помощью интерфейса и адаптера, а алгоритмически – с помощью программных средств.

Если для рассмотрения взять *двухуровневую систему*, то можно выделить ряд задач, которые нужно решить на этапе имитационного моделирования:

- программное или аппаратное моделирование объекта управления;
- моделирование управляющей логики (ПЛК);
- централизованное управления и визуализация (SCADA);
- реализация обмена данными (интерфейс, протокол, шлюз, etc.).

Производители тестовых программно-аппаратных комплексов решают технические задачи сопряжения модельного объекта (эмулируемого на рабочей станции) с натурной системой управления через средства ввода-вывода и синхронизации.

Сокращение издержек на внедрение новых систем промышленной автоматизации и повышение качества автоматического управления кроется в совершенствовании средств отладки управляющих программ для промышленных программируемых контроллеров. Оно может заключаться в автоматизированном построении программной модели технологического объекта, реализующей его динамические свойства и эмулирующей осведомительные сигналы с как можно более полным учетом их фактических характеристик.

Целью работы является повышение эффективности тестирования управляющих алгоритмов систем промышленной автоматизации путем использования метода автоматизированного построения модели объекта управления на основе обработки временных рядов технологических сигналов. Эта цель обуславливает следующие *задачи исследования*:

- анализ и упорядоченное хранение характерных образов входных и выходных сигналов в виде временных рядов;
- разработка метода быстрого поиска наиболее подходящих фрагментов временных рядов и синтеза реакции технологического объекта управления;
- разработка элементов программного обеспечения для обработки временных рядов технологических сигналов для тестирования управляющих алгоритмов систем промышленной автоматизации;
- обобщение предложенного подхода поддержки пусконаладочных работ и тестирования систем автоматического управления сложными технологическими процессами в виде схемы информационной технологии.

3. Материал и результаты исследований

Производители промышленных программируемых логических контроллеров и средств разработки управляющих программ стремятся усовершенствовать программные средства эмуляции и отладки, однако для реализации моделей технологического оборудования необходимо непосредственное участие разработчиков конкретной АСУТП. Это требует временных и денежных затрат, сравнимых с затратами на разработку управляющих программ [1].

Процесс тестирования, отладки и испытаний систем управления и программного обеспечения непосредственно на промышленном объекте приводит к неконтролируемому росту финансовых издержек и временных затрат предприятий-разработчиков. В итоге процесс пуско-наладки может затягиваться на несколько рабочих недель по сравнению с утвержденным планом работ. Имеющиеся программные средства отладки не обеспечивают автоматизированной идентификации объекта управления и автоматизированного проведения тестов с фиксацией логических ошибок.

При наличии полигонного ПЛК со всеми интерфейсными модулями отладка по принципу Hardware-in-the-loop будет заключаться в подключении входов и выходов через дополнительные модули ввода-вывода к рабочей станции, на которой выполняются задачи (рис. 1):

- формирование тестовых сценариев;
- вывод к исследуемой системе задающих сигналов и эмулированных сигналов обратной связи (осведомительных);
- считывание управляющих сигналов, формируемых ПЛК в соответствии со своей программой;
- моделирование объекта управления и расчет его мгновенных значений выходных сигналов;
- контроль корректности работы управляющих программ ПЛК и формирование сообщений оператору-тестировщику.

Стоит отметить, что при описанном подходе модельное время должно как можно точнее совпадать с реальным, иначе точность формирования осведомительных сигналов существенно падает, что может сказаться на качестве тестирования.

Подобные комплексы позволяют значительно снизить издержки по отладке систем управления, однако такая концепция бесполезна при отсутствии действенных средств автоматизации построения модели технологического объекта, максимально приближенной

по характеру реакции к реальному технологическому объекту. Крайне желательно, чтобы метрологические характеристики тестовых сигналов были близки к реальным.

Решить указанную проблему можно путем разработки метода формирования ситуационных динамических моделей технологических объектов и анализа ситуационного отклика объекта. Такой метод будет включать этапы классификации архивной информации, содержащей историю изменения всех сигналов технологического объекта, быстрого поиска подобных ситуаций по образам сигналов и расчета на их основе прогнозного значения выходного сигнала объекта. В итоге схема отладки и тестирования управляющих устройств и алгоритмов, изображенная на рис. 1, становится действенным и эффективным средством автоматизация тестирования, увеличения безопасности испытаний систем управления.

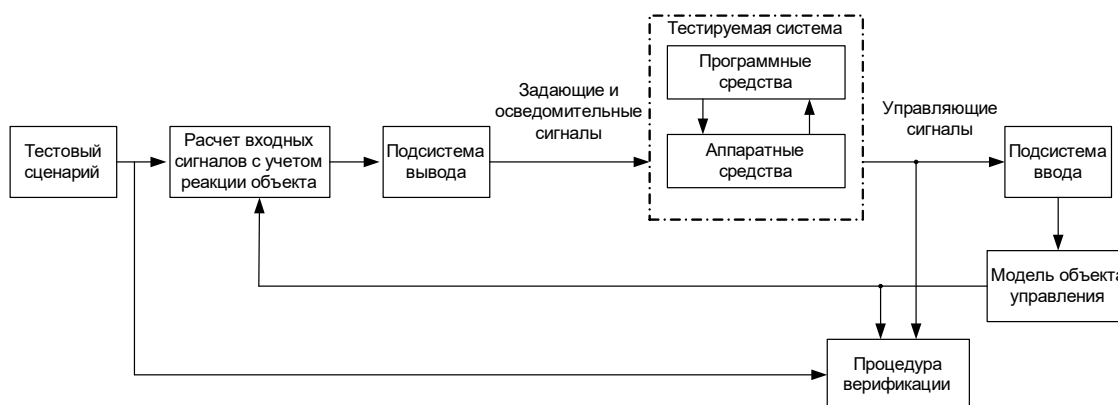


Рис. 1. Принцип Hardware-in-the-loop для отладки управляющих систем

Для сохранения форм и характеристик реальных сигналов задержка расчета прогнозного значения выходного сигнала моделью объекта управления не должна превышать интервал дискретизации тестируемой системы управления. Чаще всего для систем управления технологическими процессами горно-обогатительного производства такие интервалы лежат в пределах от 0,1 до 5 секунд.

Рис. 2 демонстрирует предложенный подход к разбиению временных отчетов технологических сигналов на характерные фрагменты (в рассмотренном примере имеется два канала входных воздействий и один канал обратной связи, по которому можно судить о реакции объекта). На основе фрагментов можно уверенно спрогнозировать будущую реакцию технологического объекта в тестовых целях, даже если точно такой же ситуации в архиве не содержится.

Детальнее процесс интерполяционных расчетов выходной реакции описан в [2]. Стоит отметить, что динамические свойства технологического оборудования вынуждают учитывать более протяженную ретроспективу для выходных сигналов.

Разбиение на фрагменты, классификацию и упорядоченное хранение исходных архивов дата-лоджинга целесообразно организовать следующим образом.

Последовательно из архива в массив оперативной памяти извлекаются значения временных отчетов. Если выбранный интервал дискретизации больше, чем в исходном архиве, итоговое значение можно получать как результат операции медианного сглаживания. В итоге количество точек отсчета каждого сигнала должно быть не менее десяти.

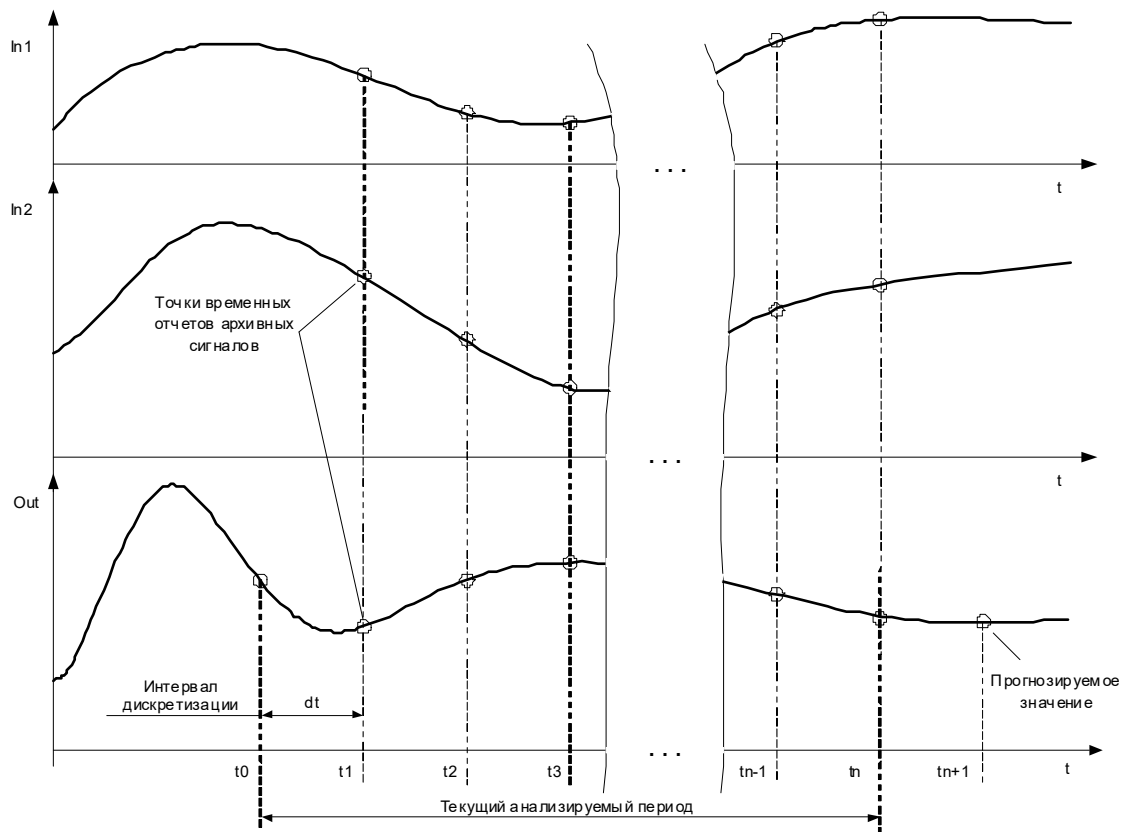


Рис. 2. Схема получения временных рядов технологических сигналов (In1, In2 – сигналы входных воздействий на объект; Out – сигнал реакции объекта)

Каждый фрагмент временных рядов сигналов хранится в виде структуры, подобной той, что приведена на рис. 3.

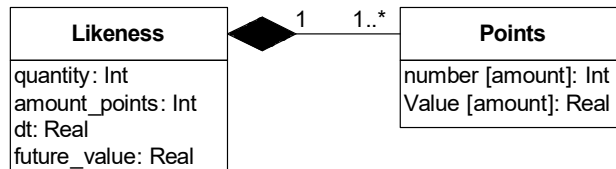


Рис. 3. Диаграмма структуры хранения временных рядов

Структура *Likeness* содержит информацию о количестве рядов, количестве точек в хранимом фрагменте, величине интервала дискретизации и прогнозом значения выходного сигнала. С этой структурой отношением композитной ассоциации связаны структуры *Points*, содержащие массивы номеров точек и абсолютных значений сигнала. Такого рода программные структуры при условии упорядоченного хранения являются основой для поведенческой эмуляции технологического объекта.

Для работы метода эмуляции следует рассчитать классификационные признаки для каждого сигнала в виде вещественных чисел. Такими признаками могут быть: среднее значение каждого сигнала на анализируемом периоде, производные сигнала в начале и конце периода, среднее значение производной, конечное значение сигнала.

Среднее значение:

$$y_{av} = \frac{\sum_{i=0..n} y_i}{n},$$

где y_{av} – среднее значение сигнала на периоде; y_i – текущее значение сигнала.

Производные вычисляются в конечных разностях:

$$dy_{av} = \frac{\sum_{i=0..n-1} \frac{y_{i+1} - y_i}{dt}}{n}, \quad dy_0 = \frac{y_{i+1} - y_i}{dt}, \quad dy_n = \frac{y_n - y_{n-1}}{dt},$$

где dy_{av} – среднее значение производной на периоде; dy_0 – значение производной на начале периода; dy_n – значение производной в конце периода.

Переход от классификационных признаков к адресам хранения структур данных выполняется посредством многомерного массива указателей, в котором каждый элемент содержит один указатель или цепочку указателей на структуры, близкие по квалификационным признакам. Мерность массива определена количеством классификационных признаков и количеством сигналов. Преобразование вещественного значения каждого признака в целочисленный индекс выполняется по формуле:

$$\text{index} = \text{round}\left(\frac{P_k - P_{\min}}{dP}\right), \quad (1)$$

где P_k – значение классификационного признака; $P_{\min}$ – минимально возможное значение классификационного признака; dP – цена дискреты классификационного признака.

Например, если реакция объекта управления – обжиговой печи – определяется температурой материала в диапазоне 650 – 1100 градусов Цельсия и количество дискретных значений выберем 20, то для среднего значения ряда 830°C по формуле (1) значение индекса будет равно 8.

В процессе отладки по принципу Hardware-in-the-loop временные ряды формируются на основе тестового сценария, который регламентирует задающие сигналы системы управления, прошлых и текущих значений сигналов управления, формируемых тестируемым ПЛК, прошлых и текущих значений осведомительных сигналов, формируемыми моделью объекта управления. Так как для модельных расчетов важен быстрый поиск фрагментов, наиболее близких к текущей ситуации, для анализируемого периода следует рассчитать признаки, от них перейти к индексам массива и извлечь указатели на структуры хранения архивных данных.

После вычисления указателя на структуры, хранящие временные ряды технологических сигналов, производится поиск наиболее подходящих путем простого перебора и расчета расстояния между рядами сигналов для каждой структуры:

$$S = \frac{1}{k} \sum_{j=1}^k \frac{\sum_{i=0}^n |y_i^* - y_i|}{y_{av}^*}, \quad (2)$$

где k – количество рядов (сигналов) в структуре; y_i^* – мгновенные значения одного сигнала текущего сценария; y_i – мгновенные значения проверяемого фрагмента; y_{av}^* – среднее значение сигнала на текущем периоде.

Если для текущей ситуации и определенных классификационных признаков не найдены близкие фрагменты, процедура верификации формирует сообщение пользователю. Для обеспечения непрерывности процесса тестирования и синтеза выходного сигнала объекта могут использоваться фрагменты из соседних ячеек массива.

На основе найденных фрагментов производится расчет выходных сигналов технологического объекта управления с помощью нечеткой интерполяции [5].

Функциональная модель проведения процессов тестирования систем управления сложными технологическими процессами описывается UML-диаграммой на рис. 4.

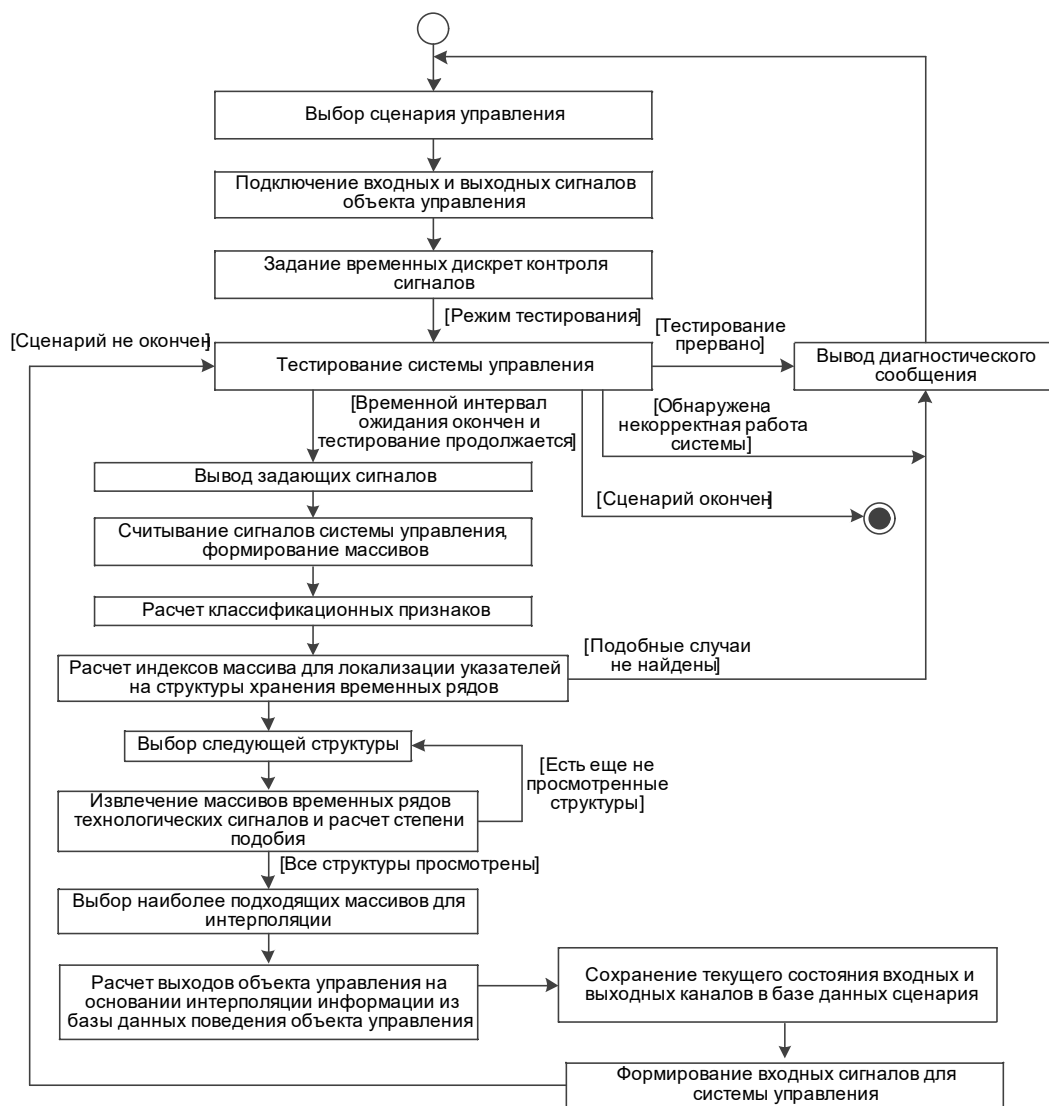


Рис. 4. UML-диаграмма состояний программного обеспечения автоматизированного тестирования систем управления

Сценарий тестирования содержит информацию об изменении во времени задающих сигналов, начальном состоянии технологического объекта и ожидаемых реакциях объекта с указанием соответствующего времени. Это позволяет запустить процесс автоматического управления модельным объектом и по ключевым моментам судить о корректности функционирования системы управления. Так как выходные сигналы моделируемого объекта формируются на основе сохраненных реальных технологических сигналов, то становится возможным обеспечить такие же метрологические характеристики и повысить достоверность тестирования [4]. Каждое последующее значение выходных сигналов рассчитывается на основе нескольких прогнозируемых значений, которые являются результатом циклической работы процедуры поиска наиболее соответствующих данной ситуации фрагментов.

Перейдем к описанию информационной технологии поддержки пусконаладочных работ и тестовых процедур систем автоматического управления сложными технологическими процессами.

Согласно [6], при разработке информационной технологии необходимо определить её статическую часть – структуру данных (DS) и комплекс математических методов и моделей (СММ), а также описать динамическую составляющую информационной техноло-

гии – информационные процессы (IP) и их взаимодействие между собой (IPI). Исходя из этого информационная технология должна представляться набором [6]:

$$IT = \langle DS, CMM, IP, IPI \rangle. \quad (3)$$

Каждый информационный процесс в свою очередь представляется блоком, имеющим информационные входы, управляющие входы, выходы и регламент выполнения, который определяется методами и моделями, привлекаемыми по мере необходимости для реализации процесса (рис. 5).

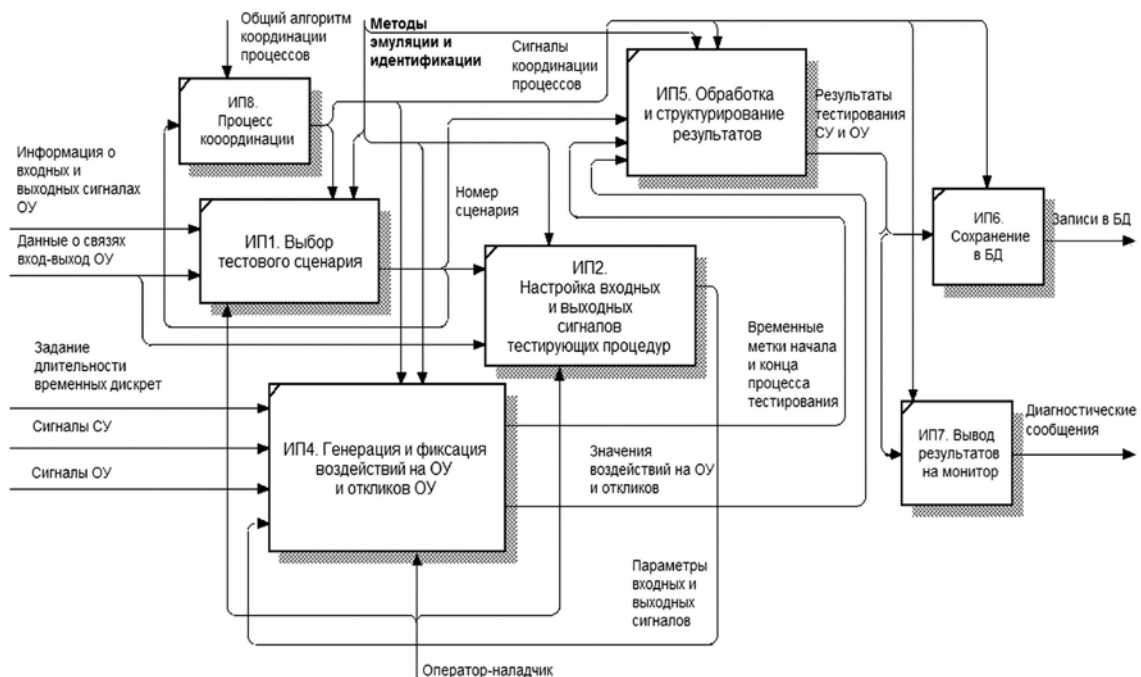


Рис. 5. Информационная технология поддержки пусконаладочных работ и тестовых процедур систем автоматического управления сложными технологическими процессами

Сущность показанных на схеме информационных процессов достаточно подробно описана выше. Процесс координации (ИП8) реализуется главным программным модулем, осуществляющим диспетчерские функции при отработке заданного тестового сценария.

4. Выводы

Развитие технологий тестирования управляющего программного обеспечения АСУТП и испытаний промышленных систем управления на базе ПЛК по-прежнему остается актуальной задачей. Современные средства отладки требуют большого объема труда квалифицированных инженеров для разработки моделей объектов управления и проведения тестов. Затраты на создание тестовых моделей «с нуля» имеют сравнимый объем с затратами на разработку управляющих программ – порядка 15–25 %. Автоматизация тестирования, увеличение эффективности и безопасности испытаний систем управления может быть достигнуто за счет внедрения сценариев управления моделью объекта с сохранением форм и характеристик реальных сигналов и автоматическим контролем корректности работы управляющих алгоритмов.

Получила дальнейшее развитие функциональная модель процесса проведения пусконаладочных работ и формирования процессов тестирования систем управления сложными технологическими процессами, отличающаяся тем, что включает формирование ситуационных динамических моделей технологических объектов, анализ ситуационного отклика объекта и позволяет снизить время и затраты на наладку систем управления, а также повысить безопасность процесса наладки сложных объектов.

Разработана информационная технология поддержки пусконаладочных работ и тестовых процедур систем автоматического управления сложными технологическими процессами в горно-обогатительном производстве.

Построение моделей поведения многоканального объекта управления на основе интерполяции временных отчетов выходных сигналов и идентификации ситуаций по массиву входных и выходных предыдущих сигналов позволяет уменьшить объем исследовательских работ и внедрить модель объекта в программные пакеты тестирования управляющих программ.

Список литературы: 1. Конох И. С. Использование информационных технологий для повышения эффективности пусконаладочных работ и испытаний систем автоматического управления / Найда В.В., Сухомлин Л.В. / Вісник Кременчуцького національного університету ім. М. Остроградського. Кременчук: КрНУ, 2015. Вип. 3/2015 (92), част. 1. С. 40–47. 2. Вычужанин В. Методика аппаратно-программного моделирования и тестирования проектируемых систем // Современная электроника. 2015. №2. С. 62–67. 3. William F. Waite, Stephen J. Swenson, Alexander C. Jolly, Seth Shepherd, Robert M. Gravitz. Validation of Hardware-In-The-Loop (HWIL) And Distributed Simulation Systems. Foundations For V&V In The 21st Century Workshop (Foundations '02), Kossiakoff Conference And Education Center, Johns Hopkins University /Advanced Physics Laboratory, Laurel, Maryland, USA, 2002. 4. Gomez M. Hardware-in-the-Loop Simulation /M. Gomez// Embedded. 2001. Режим доступа: <http://www.embedded.com/design/prototyping-and-development/4024865/Hardware-in-the-Loop-Simulation>. 5. Конох И. С. Эмуляция сигналов объектов управления в электромеханических системах с использованием нечетких логических контроллеров /И.С. Конох, В.В. Найда // Електромеханічні і енергозберігаючі системи: щоквартальний науково-виробничий журнал. Кременчук: КрНУ, 2013. Вип. 4/2013 (24). С. 70–83. 6. Левыкин В. М. Метод построения информационной технологии диагностики состояния сложного технологического процесса / В. М. Левыкин, И. В. Шевченко // Управляющие системы и машины. 2014. №3. С. 33–38.

Поступила в редколлегию 11.01.2016

Найда Виталий Владимирович, аспирант кафедры информационно-управляющих систем Кременчугского национального университета им. М. Остроградского. Научные интересы: интеллектуальные информационные технологии процессов тестирования и наладки сложных систем управления. Адрес: Украина, 39600, Кременчуг, ул. Первомайская, 20, E-mail: vitalik.najda@yandex.ru.

Шевченко Игорь Васильевич, д-р техн. наук, доцент, профессор кафедры информационно-управляющих систем Кременчугского национального университета им. М. Остроградского. Научные интересы: интеллектуальные информационные технологии контроля и управления в сложных технологических процессах. Адрес: Украина, 39600, Кременчуг, ул. Первомайская, 20, E-mail: ius.shevchenko@gmail.com.

Конох Игорь Сергеевич, канд. техн. наук, доцент, доцент кафедры информационно-управляющих систем Кременчугского национального университета им. Михаила Остроградского. Научные интересы: промышленная автоматизация, интеллектуальные системы управления технологическими процессами. Адрес: Украина, 39600, Кременчуг, ул. Первомайская, 20, E-mail: kis_saue@mail.ru.

КУБИТНЫЕ МОДЕЛИ ОПИСАНИЯ ЦИФРОВЫХ УСТРОЙСТВ

Предлагается облачный сервис QuaSim для моделирования и верификации цифровых систем, основанный на транзакциях между адресуемыми компонентами памяти для реализации любой функциональности. Описывается новый подход к синтезу и анализу цифровых систем, использующий векторную форму (квант) задания комбинационных и последовательностных структур для их имплементации в элементы памяти, что существенно отличается от общепринятой теории проектирования дискретных устройств на основе таблиц истинности компонентов. Используются квантовые или кубитные структуры данных [1-5] для реализации вычислительных процессов в целях повышения быстродействия анализа цифровых систем и уменьшения объемов памяти на основе унарного кодирования состояний входных, внутренних и выходных переменных и имплементации кубитных векторов в элементы памяти FPGA, реализующих комбинационные и последовательностные примитивы.

1. Общая характеристика исследования

Цель – повышение надежности вычислительных устройств за счет адресуемости схемных элементов, что позволяет выполнять online ремонт, а также повышает быстродействие методов моделирования, тестирования и верификации сложных цифровых систем, благодаря уменьшению размерности моделей функциональных примитивов и адресной реализации всех компонентов структур данных.

Задачи: 1) Создание автоматной модели кубитного процессора. 2) Синтез кубитных моделей цифровых примитивов. 3) Синтез и анализ кубитных моделей цифровых схем. 4) Анализ цифровых систем на основе использования квантовых векторов описания примитивов.

Актуальность исследования: 1) Современная система на кристалле содержит 94% памяти и лишь 6% логики, которая доставляет более 90% проблем, связанных с верификацией, тестированием, диагностированием и восстановлением работоспособности [6-7]. Конечно, быстродействие логических схем на порядок выше, чем у памяти, однако большая доля вычислительных процессов приходится на обмен информацией в структурах памяти. Поэтому преимущества комбинационной логики в реальных вычислительных системах обработки больших данных компенсируются большими временными затратами (порядка 90%), связанными с транзакциями в памяти. 2) Реализация процессора только на основе использования элементов памяти делает его однородным по структуре и типам функциональных примитивов, что доставляет очевидные технологические удобства процессам проектирования, производства и эксплуатации, включая верификацию, встроенные тестирование и диагностирование, а главное – ремонт в режиме online с помощью использования на кристалле универсальных адресуемых spare-компонентов памяти. 3) Моделирование в процессе верификации проектируемых вычислителей на основе адресных моделей компонентов делает данную процедуру технологически простой из-за регулярных структур данных и применения единственной операции транзакции на элементах памяти, а также более быстродействующей, благодаря параллельной квантоподобной обработке больших массивов однотипной памяти [3-5, 8, 11, 12]. 4) Энергопотребление при замене логики на элементы памяти возрастает на несколько процентов, что на самом деле будет платой за перечисленные выше существенные преимущества, связанные с увеличением выхода годной продукции, повышением надежности вычислительных изделий, снижением стоимости проектирования и изготовления, а также автономным восстановлением работоспособности в режиме remote & online без участия человека. Однако энергосберегающие решения по вычислительным процессам на памяти дают основания полагать, что такого проигрыша вообще не будет [9-10, 13-14].

2. Кубитные структуры данных

На мировом рынке электронных технологий наблюдается конкуренция между базами имплементации идеи [1-4,12]: 1) Гибкая (мягкая) реализация проекта связана с синтезом интерпретативной модели программной формы функциональности или в аппаратном исполнении программируемых логических устройств на основе FPGA, CPLD; преимущества – в технологичности модификации проекта, недостатки – в невысоком быстродействии функционирования цифровой системы. 2) Жесткая реализация ориентирована на использование компилятивных моделей при разработке программных приложений или на имплементацию проекта в кристаллы VLSI [6-7,13-14]. Преимущества и недостатки жесткой реализации инверсны по отношению к мягкому исполнению проектов: высокое быстродействие и невозможность модификации. С учетом изложенных базовых вариантов реализации идеи предлагаются квантовые структуры данных, ориентированные на повышение быстродействия гибких моделей программного или аппаратного исполнения проекта, а также на возможность online ремонта в процессе эксплуатации.

Кубитные структуры описания цифровых систем. Кубит (n -кубит) есть векторная форма унитарного (унарного) кодирования универсума из n примитивов для задания булеана состояний 2^{2^n} с помощью 2^n двоичных переменных.

Например, если $n=2$, то 2-кубит задает 16 состояний с помощью четырех переменных. Если $n=1$, то кубит задает четыре состояния на универсуме из двух примитивов (10) и (01) с помощью двух двоичных переменных (00,01,10,11) [3,12]. При этом допускается суперпозиция (одновременное существование) в векторе 2^n состояний, обозначенных примитивами. Кубит (n -кубит) дает возможность использовать параллельные логические операции вместо поэлементных теоретико-множественных для существенного ускорения процессов анализа дискретных систем.

Кубит отождествляется с n -кубитом или двоичным вектором, если это не мешает пониманию излагаемого материала. Поскольку квантовые вычисления связаны с анализом кубитных структур данных, то далее будем применять определение «квантовый» для идентификации технологий, использующих три свойства квантовой механики: параллелизм обработки (двоичных векторов), суперпозицию состояний и их перепутывание. Синонимами кубита при задании двоичного вектора описания логической функции являются: Q-покрытие, Q-вектор, квантовый вектор [3-4,12,15] как унифицированная векторная форма суперпозиционного задания выходных состояний, соответствующих адресным кодам входных переменных логического элемента.

Кубит в цифровой системе используется в качестве формы задания структурного примитива, инвариантной к технологиям реализации функциональности (hardware, software). Более того, синтез цифровых систем на основе кубитных структур не привязан жестко к теореме Поста, определяющей пять условий (классов) существования функционально полного базиса. На предлагаемом уровне абстракции n -кубит дает более широкие возможности для векторного задания любой n -входовой функции из булеана мощностью $|B(A)| = 2^{2^n}$, которое непременно содержит все функциональности, удовлетворяющие пяти классам теоремы Поста. Формат структурного кубитного компонента цифровой схемы $Q^* = (X, Q, Y)$ включает интерфейс (входные и выходную переменные), а также кубит-вектор Q , задающий функцию $Y = Q(X)$, размерность которого определяется степенной функцией от числа входных линий $k = 2^n$.

3. Синтез кубитного покрытия комбинационной схемы

Кубит комбинационной схемы представляет собой вектор состояний выхода на упорядоченном множестве всех входных слов, который отождествляется с адресами ячеек памяти вектора. Синтез Q-покрытия схемной структуры (без таблиц истинности логических элементов) на основе примитивов, заданных Q-векторами, сводится к получению обобщенного кубит-вектора путем выполнения логической операции над разрядами кубитных векторов с помощью декартовой процедуры – для двух 4-разрядных кубитов по суперпозиции логической операцией or (and, xor):

$\vee, \wedge, \oplus$	b(0)	b(1)	b(2)	b(3)
a(0)	$c(0) = a(0) \vee b(0)$	$c(1) = a(0) \vee b(1)$	$c(2) = a(0) \vee b(2)$	$c(3) = a(0) \vee b(3)$
a(1)	$c(4) = a(1) \vee b(0)$	$c(5) = a(1) \vee b(1)$	$c(6) = a(1) \vee b(2)$	$c(7) = a(1) \vee b(3)$
a(2)	$c(8) = a(2) \vee b(0)$	$c(9) = a(2) \vee b(1)$	$c(10) = a(2) \vee b(2)$	$c(11) = a(2) \vee b(3)$
a(3)	$c(12) = a(3) \vee b(0)$	$c(13) = a(3) \vee b(1)$	$c(14) = a(3) \vee b(2)$	$c(15) = a(3) \vee b(3)$

Например, для логических суперпозиций двух кубитов при получении Q-векторов схемных структур $c_1 = (a_1 \wedge a_2) \vee (b_1 \wedge b_2)$, $c_2 = (a_1 \wedge a_2) \wedge (b_1 \vee b_2)$, $c_3 = (a_1 \wedge a_2) \oplus (b_1 \vee b_2)$, имеет место таблица:

a(and) =	0001
b(or) =	0111
$c_1 = a(\text{and}) \vee b(\text{or})$	0111 0111 0111 1111
$c_2 = a(\text{and}) \wedge b(\text{or})$	0000 0000 0000 0111
$c_3 = a(\text{and}) \oplus b(\text{or})$	0111 0111 0111 1000

При построении Q-покрытия трех схем из трех элементов каждая используется суперпозиция двух логических примитивов с третьим элементом (or, and, xor), вследствие чего получаются три вектора размерности 16 бит каждый. Вычислительная сложность процедуры синтеза Q-покрытия комбинационной схемы определяется произведением длин Q-векторов p примитивов, входящих в нее: $\eta = \prod_{i=1}^p \text{card}(Q_i)$.

Проблема синтеза Q-покрытия схемы, входные линии/сходящиеся разветвления которой имеют гальванические/проводные соединения (здесь по переменной a_2): $c = (a_1 \wedge a_2) \vee (a_2 \vee a_3)$, является более сложной задачей. В данном случае после синтеза Q-покрытия схемы следует выполнить его верификацию относительно существования противоречивых адресов на переменных a_2 в целях минимизации Q-вектора путем последующего исключения упомянутых адресов из рассмотрения. При этом размерность Q-покрытия уменьшается до $\text{card}(Q) = 2^q$ координат, где q – общее число входных переменных схемы:

Q = 0111 0111 0111 1111	Q = 0111 0111 0111 1111	Q = 0111 0111	Q = 0111 0111
a ₁ = 0000 0000 1111 1111	a ₁ = 0000 0000 1111 1111	a ₁ = 0000 1111	a ₁ = 0000 1111
a ₂ = 0000 1111 0000 1111	a ₂ = 00xx xx11 00xx xx11	a ₂ = 0011 0011	a ₂ = 0011 0011
a ₂ = 0011 0011 0011 0011	a ₂ = 00xx xx11 00xx xx11	a ₂ = 0011 0011	a ₂ = 0011 0011
a ₃ = 0101 0101 0101 0101	a ₃ = 0101 0101 0101 0101	a ₃ = 0101 0101	a ₃ = 0101 0101

Синтез Q-покрытия включает: 1) построение таблицы соответствия адресов разрядам Q-вектора схемы, 2) отметку символами x противоречивых координат по двум строкам a_2 , 3) исключение из таблиц всех столбцов с данными символами, 4) объединение в одну получившихся двух идентичных строк a_2 , 5) результирующий Q-вектор комбинационной схемы имеет существенно меньшую размерность. Преимущества предложенного Q-метода синтеза вычислительных устройств заключаются в компактности их описания Q-векторами и в высоком быстродействии адресного моделирования логических элементов, создаются условия для рыночно привлекательной «квантовой» теории проектирования цифровых систем на кристаллах, использующей векторно-кубитную форму задания структурных компонентов.

4. Минимизация кубитного покрытия схемы

Синтез кубит-вектора схемы по Q-покрытиям компонентов связан с уменьшением размерности Q-вектора путем исключения несущественных переменных. Существенность зависит от гальванических соединений входных и внутренних линий цифрового устройства, которые накладывают ограничения, связанные с противоречивостью сигналов на линиях связи. Правило минимизации адресного пространства заключается в устранении адресных кодов, которые создают противоречия по соединенным переменным.

Пусть имеется Q-вектор схемы и его адресное пространство, где переменные b,c,d (a,b,c) соединены гальванически. Ниже приведены таблицы минимизации адресного пространства для получения уменьшенного Q-вектора:

Q =	0111 0111 0111	Q =	0xxx xxx1 0xxx xxx1	Q	0111 0111 0111 1111	Q	0111
a =	0000 0000 1111	a =	0000 0000 1111 1111	a =	00xx xxxx xxxx xx11	a =	0011
b =	0000 1111 0000	b =	0xxx xxx1 0xxx xxx1	b =	00xx xxxx xxxx xx11	b =	0101
c =	0011 0011 0011	c =	0xxx xxx1 0xxx xxx1	c =	00xx xxxx xxxx xx11	c =	0101
d =	0101 0101 0101	d =	0xxx xxx1 0xxx xxx1	d =	0101 0101 0101 0101	d =	0101

В таблицах наблюдается зеркальная осевая симметрия с инверсией сигналов на координатах адресного пространства, которая создает свойство, описываемое следующим выражением: $L \oplus R = 1 \rightarrow L_{ij} \oplus R_{ij} = 1$. Данное обстоятельство следует использовать для уменьшения размерности анализируемого пространства в два раза и соответствующего снижения вычислительной сложности задачи синтеза квантовой вектор-функциональности цифровой схемы.

Количество различных вариантов взаимодействий на q входных переменных, связанных с гальваническим соединением сочетаний входных линий, определяется функциональной зависимостью, граничные значения которой находятся в интервале: $\text{card}(Q) = [2^q - 3^q]$.

Существует эффективная процедура для минимизации размерности Q-вектора путем выявления противоречий в кодах-столбцах, на координатах (A_{ij}), соответствующих гальванически связанным w-переменным по j-параметру. Очевидно такую процедуру достаточно выполнить на половине адресного пространства $\text{card}(Q) = 2^q / 2$, тогда оставшая часть противоречивых столбцов удаляется в соответствии с зеркальным отображением номеров тех столбцов, которые были удалены из первой половины таблицы кодов адресов:

$$\{Q_i, Q_{2^q-i}\} = \emptyset \leftrightarrow \left(\bigwedge_{j=1}^w A_{ij} \right) \oplus \left(\bigvee_{j=1}^w A_{ij} \right) = 1, i \leq 2^q / 2.$$

Если в столбце A_i на группе из w связанных переменных зафиксировано, что конъюнкция их состояний равна нулю, а дизъюнкция имеет значение единицы, то i-столбец и его зеркальное отображение $2^q - i$ удаляются из адресного пространства A. Это автоматически приводит к исключению из Q-вектора двух полученных $\emptyset$ -координат (в таблицах обозначены символами x), соответствующих данным столбцам.

Наблюдается также симметрия пространства векторов-расстояний по Хэммингу, полученных путем хог-взаимодействия между соседними строками таблицы адресного пространства, для которых суперпозиция левой и правой частей дает результат

$$L \oplus R = 0 \rightarrow L_{ij} \oplus R_{ij} = 0:$$

Q =	0111 0111 0111 1111	Q =	0111 0111 0111 1111	
a ⊕ b	0000 1111 1111 0000	a ⊕ b	0000 0000	= (L, R); (L ⊕ R) = ↔ (L = $\bar{R}$)
b ⊕ c	0011 1100 0011 1100	b ⊕ c	0000 0000	
c ⊕ d	0110 0110 0110 0110	c ⊕ d	0000 0000	
d ⊕ a	0101 0101 1010 1010	d ⊕ a	0000 0000	

Целесообразность минимизации логической функции, описанную квант-вектором: минимизация Q-векторов для получения нормальных или скобочных форм не имеет практического значения, существенно только уменьшение размерности вектора функционального описания, что может быть лишь следствием определения несущественности некоторых

входных (адресных) переменных. Тем не менее, существует проблема разбиения квант-вектора на составляющие части меньшей размерности, что связано с имплементацией функциональности в конструктивные компоненты LUT FPGA. В этом случае выполняется разбиение Q-вектора на два равных подвектора $Q=(L,R)$, которые соединяются в структурно-адресную организацию функциональности с помощью мультиплексора $Q = (\bar{a} \wedge L) \vee (a \wedge R)$. Если переменная мультиплексирования $a=0$, то функциональность Q формируется с помощью ячеек левого L-вектора, в противном случае, когда $a=1$, значение функции Q формируется битами правого R-вектора. Алгоритмы разбиения и имплементации сложных логических функций имеются в каждой промышленной системе синтеза, моделирования и верификации компонентов SoC.

5. Модель кубитного процессора

Квантовый процессор может быть любой конечной размерности: вектор, матрица, куб. Для структуры, содержащей два измерения, он представлен матрицей столбцов или Q-векторов, которые формируют соответствующие им ячейки M-вектора моделирования (рис. 1, а). Вектор M совместно с X-вектором кортежей входных переменных примитивов создает структуру взаимных связей между столбцами-элементами. Адрес ячейки Q-покрытия, формирующей состояние невыходного i-разряда M-вектора, определяется содержанием ячеек M-вектора, найденным по адресам, заданным i-кортежем вектора входных переменных. Каждый вектор Q_i , равно как и кортеж X_i вектора номеров входных линий, имеет адресную связь с M_i -ячейкой вектора моделирования. Квантовый процессор может входить компонентом в состав более сложной системы. Квантовая модель процессора имеет следующую структуру:

$$\begin{aligned} W &= \langle Q, M, X \rangle, \\ Q &= (Q_1, Q_2, \dots, Q_i, \dots, Q_n), \\ Q_i &= (Q_{i1}, Q_{i2}, \dots, Q_{ij}, \dots, Q_{ik_i}); \\ M &= (M_1, M_2, \dots, M_i, \dots, M_n); \\ X &= (X_1, X_2, \dots, X_i, \dots, X_n), \\ X_i &= (X_{i1}, X_{i2}, \dots, X_{ij}, \dots, X_{im_i}); \\ M_i &= Q_i[M(X_i)]; k_i = 2^{m_i}. \end{aligned}$$

В аналитической модели W представлены: 1) Упорядоченная адресно-доступная Q-совокупность квантовых примитивов, формирующих функциональность системы. 2) Вектор моделирования M, связывающий все примитивы в единую систему на основе идентификации эквипотенциальных линий, создающих формат из существенных переменных: входных, внутренних и выходных. 3) Вектор X кортежей упорядоченных номеров входных переменных для каждого квантового примитива, которые формируют адреса доступа к ячейкам Q-векторов примитивов (рис. 1, а). Вектор количества входных переменных примитива $|X|$ формирует адресное пространство или длину каждого Q-покрытия. Его можно представить в виде таблицы кортежей входных переменных, которые формируют номера линий вектора моделирования для вычисления адресов доступа к квантовым покрытиям (рис. 1, б). Таблицу кортежей можно также представить в виде матрицы масок входов, определенных в формате вектора моделирования, для параллельного формирования адресов и одновременного считывания выходных состояний примитивов из матрицы Q-покрытий (рис. 1, в). Из структуры X-матрицы входных линий видно, что кванты, формирующие выходы: (8, 9, 10), (11, 12) и (13, 14), можно обрабатывать параллельно. 4) Характеристическое уравнение, задающее алгоритм функционирования квантового процессора на основе использования только операций транзакции (считывание-запись) между Q-векторами примитивов и вектором моделирования.

Схема цифрового устройства, соответствующая приведенному выше описанию структур данных: M-вектор моделирования, X-матрица входов и Q-матрица покрытий, представлена на рис. 2. Она содержит 9 примитивов, каждый из которых имеет Q-покрытие в форме квант-вектора, реализующего некоторую функциональность. Особенность квантовых структур данных, представляющих модель цифровой схемы, заключается в полной адресуемости всех компонентов устройства без гальванических проводных соединений.

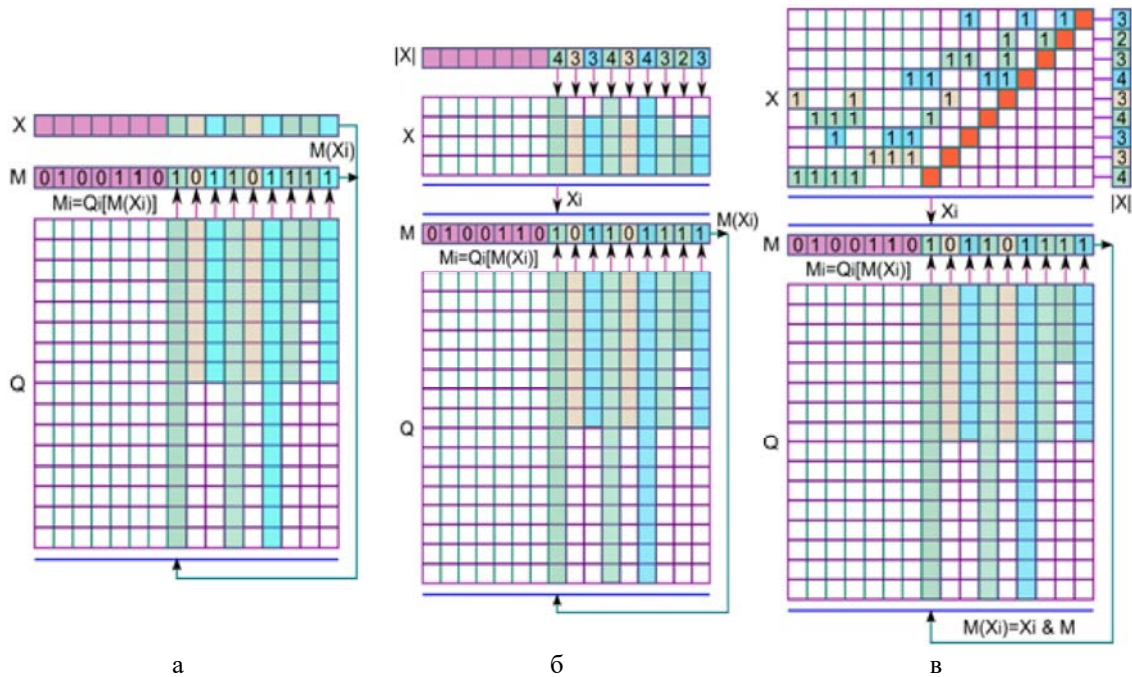


Рис. 1. Кубитные структуры данных кубитного процессора

Для кубитного(only memory-based) процессора имеют место следующие аксиомы: 1) В квантовом процессоре нет ничего, кроме адресуемой памяти. 2) Вычислительный процесс представлен единственной универсальной транзакцией между адресуемыми компонентами памяти $M_i = Q_i[M(X_i)]$. 3) Транзакция есть универсальная процедура считывания-записи данных на непустом множестве адресуемых элементов памяти. 4) Все компоненты памяти являются online-repaired, благодаря их псевдогальванической адресной (address-connected) связности. 5) Комбинационные логические элементы (reusable logic), равно как и последовательностные (sequential components), исполняются на элементах памяти. 6) Связывание всех компонентов в вычислительную систему осуществляется посредством (цифровой) идентификации псевдо-гальванических соединений вход-выходных переменных компонентов схемы, формирующих вектор моделирования, который хранит состояния всех существенных линий цифровой системы. 7) Все компоненты кубитной модели цифровой системы: $W = \langle Q, M, X \rangle$, включая функциональные модули, вектор моделирования, вектор адресов входных переменных, являются online перепрограммируемыми, а значит – online ремонтпригодными. 8) Прimitив цифровой системы имеет формат $W = \langle Q, Y, X \rangle$, поскольку отдельный элемент не имеет связей и вектора М, создающих из отдельных компонентов систему.

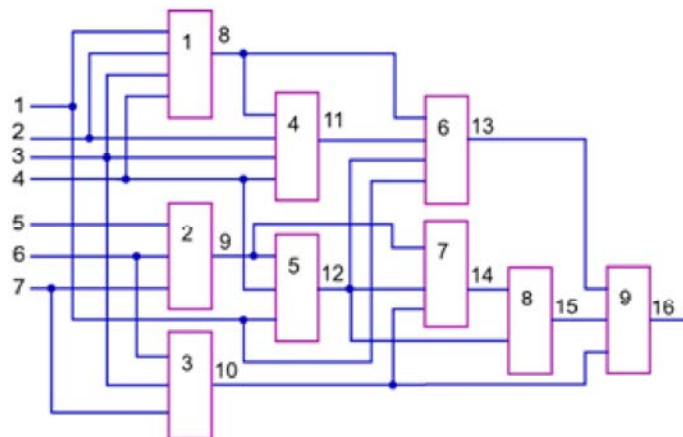


Рис. 2. Схема цифрового устройства

Согласно введенной квантовой модели, описания последовательностных примитивов (триггеры, регистры, счетчики) можно представлять Q-покрытиями или кубитными векторами, которые имеют псевдопеременные для задания внутреннего состояния. Например, функциональное описание SR-триггера трансформируется в квантовый примитив, заданный Q-покрытием, а затем реализуется на адресуемом элементе памяти FPGA с диаграммами проверки, что представлено на рис. 3.

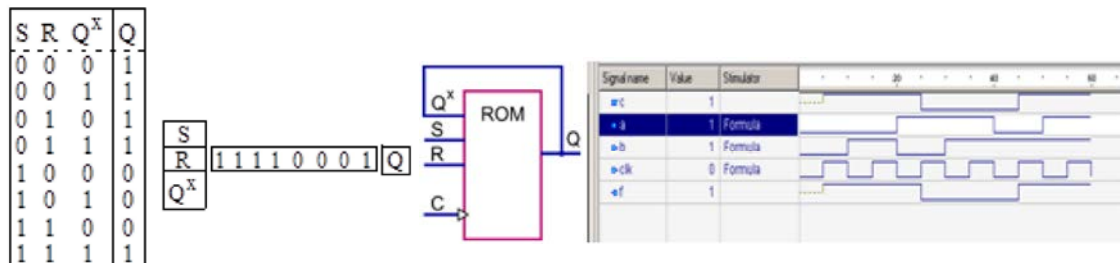


Рис. 3. SR-триггер на элементе памяти

Таблица истинности триггера представлена в форме вектора выходных состояний $Q(S, R, Q^x) = (11110001)$, который записывается в элемент постоянной памяти, имеющий три адресных входа, сигнал синхронизации, а также обратную связь, которая соединяет выход элемента памяти с одним адресным входом. HDL-реализация в системе проектирования Active HDL 9.1 (Aldec Inc.), а также результаты верификации синтезированного SR-триггера подтверждают корректность схемотехнического решения.

Другой пример связан с синтезом на элементе постоянной памяти синхронного DV-триггера. Таблица истинности триггера трансформирована в вектор выходных состояний $Q(D, V, Q^x) = (01000111)$, который записывается в элемент памяти, имеющий три адресных входа, сигнал синхронизации, а также обратную связь, которая соединяет выход примитива памяти с одним адресным входом. Все упомянутые компоненты, включая временные диаграммы верификации HDL-кода модели DV-триггера, представлены на рис. 4.

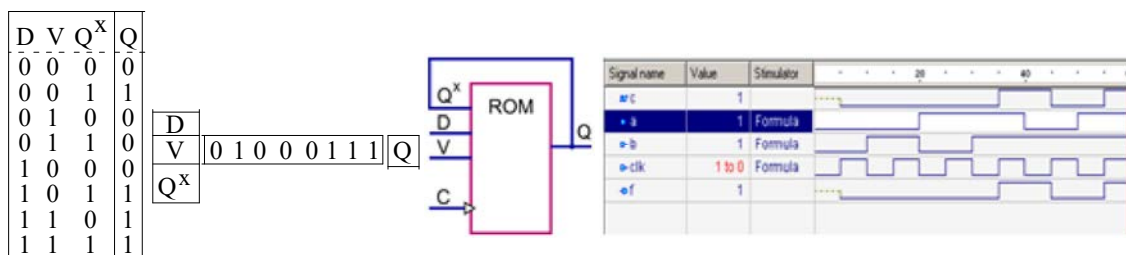


Рис. 4. DV-триггер на элементе памяти

На рис. 5 представлены модели двух последовательностных примитивов: двухразрядных регистра и счетчика. Их отличие заключается в задании двух выходов, состояния которых формируются одним и тем же множеством входных переменных.

Регистр на переменных (C,D,Q1,Q2,Y1,Y2) выполняет функцию сдвига вправо информации от входа D по разрядам: (D-Y1-Y2), при R=1, и сохранение данных при C=0. Счетчик, определенный на переменных (R,C,Q1,Q2,Y1,Y2), реализует функцию инкремента по разрядам (Y1,Y2), при RC = (11), а также режим хранения информации, при (R or C = 0). Таким образом, для реализации двухразрядного регистра или счетчика необходимо два 16-битовых элемента памяти, работающих синхронно от одних и тех же входов:

C		R	
D	0 0 1 1 0 0 1 1 0 0 0 0 1 1 1 1	C	0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0
Q1	0 1 0 1 0 1 0 1 0 0 1 1 0 0 1 1	Q1	0 0 0 0 0 0 0 0 0 0 1 0 1 1 0 1 0
Q2		Q2	

Здесь каждая квантовая модель представлена двумя векторами, где каждый из них формирует функцию разряда регистра или счетчика, как состояние ячейки вектора, получаемое при формировании адреса A входными переменными: $\{Y1, Y2\} = A(C, D, Q1, Q2)$, $\{Y1, Y2\} = A(R, C, Q1, Q2)$ соответственно. Моделирование примитива сводится к тривиальной процедуре формирования адреса, по которому находится состояние выхода примитива, как содержимое ячейки квантового вектора.

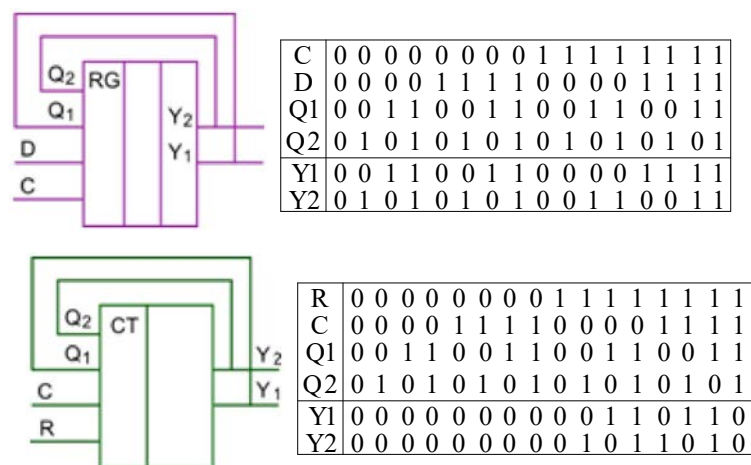


Рис. 5. Memory-based модели регистра и счетчика

6. Алгоритм моделирования кубитных покрытий цифровых компонентов

Использует memory-based only модели для адресного анализа цифровых систем в целях их верификации. Реализация таких структур связана с ячейками памяти (LUT (Look Up Table) FPGA), которые способны хранить информацию в виде Q-вектора, где каждый бит или разряд имеет свой адрес, отождествляемый с входным словом. Программная реализация алгоритма моделирования таких структур становится конкурентоспособной по быстродействию на рынке проектирования цифровых систем на кристаллах за счет адресации функциональных примитивов.

Одномерный Q-вектор описания функциональности можно привязать к выходной (внутренней) линии устройства, состоянием которой формируется в процессе моделирования рассматриваемого Q-покрытия. Тогда регистровая реализация комбинационного устройства может быть представлена вектором моделирования M , невходные линии которого непосредственно связаны с выходами функциональных элементов. Упорядоченные значения входных переменных задают адрес бита Q-вектора, формирующего состояние рассматриваемой невходной линии. Если функциональности описываются одновыходовыми примитивами, то каждый из них можно отождествить с номером или координатой невходной линии, на которую нагружен данный элемент. Если функциональность многovýchодовая, то Q-покрытие представляется матрицей с числом строк, равным числу выходов. Эффект от такого примитива заключается в параллелизме одновременного вычисления состояний нескольких выходов за одно обращение к матрице по текущему адресу. Данное обстоятельство является существенным аргументом в пользу синтеза обобщенных кубитов для фрагментов цифрового устройства или всей схемы в целях их параллельной обработки на одном временном такте. Модель функционирования цифровой структуры упрощается до вычисления двух адресов при формировании вектора моделирования $M_i = Q_i[M(X_i)]$ путем исключением сложного адреса выхода примитива в процессе записи состояний выходов в координаты M-вектора.

Алгоритм моделирования квантовых примитивов цифровой системы использует аналитическую структуру (k – число входных переменных i -примитива, $*$ – операция конкатенации битов, A – адрес бита Q -вектора):

$$M_i = Q_i(A) \leftarrow A = \bigstar_{j=1}^k M(X_{ij}) \leftarrow \begin{matrix} M \\ X_i \\ Q_i \end{matrix}$$

Данному аналитическому выражению можно поставить в соответствие следующие пункты алгоритма формирования двоичных состояний M -вектора моделирования цифровой схемы, изображенные на рис. 6:

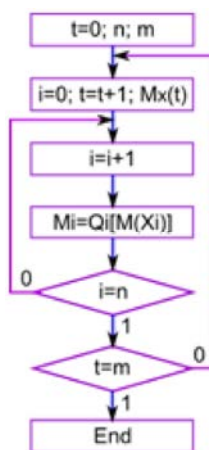


Рис. 6. Алгоритм моделирования квантовых покрытий цифровой системы

0) Инициирование начальных условий и параметров. 1) Задание очередного набора двоичных состояний на входных координатах вектора моделирования. 2) Определение i -номера очередного обрабатываемого примитива путем выполнения операции инкрементирования. 3) Выполнение процедуры конкатенации состояний битов M -вектора, соответствующих номерам вектора входных переменных X_i . Считывание соответствующего бита из функционального кубит-покрытия Q_i по двоичному вектор-адресу сконкатенированных битов M -вектора. Занесение считанного из кубита бита в вектор моделирования M по адресу i . (M -вектор может иметь координаты с символами X , что дает возможность выполнять тройное моделирование цифровых устройств для решения задач тестирования и верификации.) 4) Если не все примитивы обработаны $i < n$, выполняется переход к пункту 2 алгоритма. 5) Если не все входные наборы обработаны $t < m$, выполняется переход к пункту 1. 6) Конец моделирования.

Исходя из характеристического уравнения квантовой модели цифровой системы можно сделать вывод, что современный <MQT> (Memory-Quant-Transaction) процессор следует представлять как адресную организацию структуры функциональных примитивов памяти без гальванических или проводных связей, на которых определены адресные транзакции данных во времени и пространстве для достижения поставленной цели.

На рис. 7 представлена схема с триггерами и комбинационной логикой, которая также описана в виде элементов памяти, куда занесены выходные состояния таблицы истинности каждого логического элемента. Структуры данных, необходимые для моделирования цифрового устройства, сведены в таблицу, где основными компонентами являются: M – вектор моделирования или состояния занумерованных линий, который в данном случае имеет 5 входных, 6 внутренних и выходных линий, состояния которых подлежат определению; X – вектор кортежей номеров входных линий примитивов, которые необходимы для формирования адреса в целях извлечения по нему состояния выхода элемента Q_i , функциональность которого задается Q -вектором.

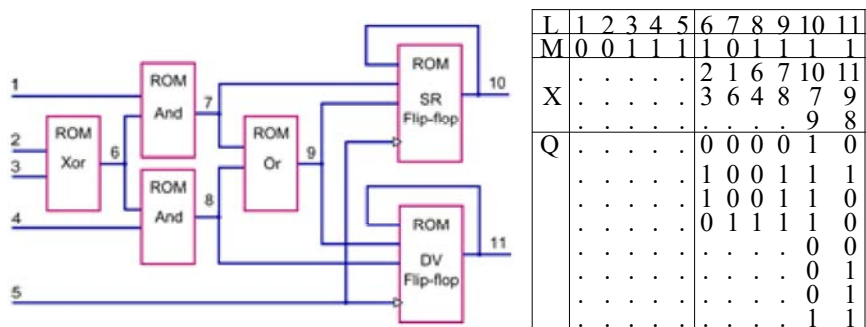


Рис. 7. Memory-based комбинационная схема с триггерами

Пример выполнения алгоритма моделирования схемной квантовой структуры. Все примитивы должны быть упорядочены по принципу: очередной элемент анализируется, если все предшественники для него были обработаны. В процессе моделирования адресно извлеченное состояние ячейки текущего Q-покрытия заносится в разряд M_i вектора моделирования. Результаты последовательной обработки всех Q-векторов схемной структуры формируют состояния линий M-вектора для приведенного выше примера ячейки (6 – 11). Первоначальные состояния неопределенностей на псевдовходах функциональных примитивов доопределяются сигналами нуля или единицы в зависимости от внутренней технологической культуры компании, предоставляющей промышленные средства моделирования и верификации. Количество входных переменных примитива q связано с длиной Q-вектора соотношением: $\text{card}(Q) = 2^q$. Правильность работы алгоритма моделирования была верифицирована на тестовых и реальных схемах с привлечением средств Active HDL 9.1 (Aldec Inc.). Особенность структурно-функционального задания цифровой системы заключается в представлении всех примитивов элементами памяти, куда записываются Q-векторы выходных состояний.

Таким образом, можно сделать следующие выводы: 1) Любые структурные компоненты вычислительных устройств, комбинационные и/или последовательностные, а также системы в целом можно описывать кубитными Q-векторами и реализовывать в элементах памяти FPGA, CPLD или VLSI. Это предоставляет рынку электронных технологий возможность не использовать комбинационную reusable логику при синтезе вычислительных устройств, которая доставляет разработчикам серьезные проблемы, связанные с тестированием, верификацией и ремонтом жесткой проводной реализации цифровых изделий. 2) Memory-based интерпретативное адресно-ориентированное моделирование комбинационных и последовательностных примитивов цифровых устройств становится соизмеримым по быстродействию с компилятивным анализом дискретных объектов. Кроме того, становится возможным реализовывать на программируемых логических устройствах аппаратное моделирование цифровых систем, где комбинационные и последовательностные функциональные примитивы будут представлены стандартными элементами памяти, в которые зашиваются Q-векторы.

7. Анализ вычислительных структур

Сигналы синхронизации доставляют определенные неудобства для описания моделей последовательностных компонентов (триггеры, регистры, счетчики) и реализации алгоритмов анализа. Это связано со схемотехническим исполнением управления по переднему или заднему фронту, которые разрешают выполнение транзакций между master-slave компонентами. Другими словами, синхронные примитивы имеют два последовательно соединенных элемента, ориентированные на низкий и высокий уровни сигналов записи данных в первую и вторую ступени соответственно. Однако для логического моделирования учет подробностей, связанных со схемотехническими решениями, может существенно замедлить время анализа цифровых схем. Поэтому здесь необходимы логически адекватные модели реальных процессов, приводящие к повышению быстродействия алгоритмов обработки компонентов. При этом накладывается ограничение, связанное только с адресным

характером анализа всех компонентов схемы. Для обеспечения возможности рассмотрения синхровхода, как логической переменной, формирующей адрес квантового вектора, предлагается модель разбиения последовательностного примитива на два элемента: 1) Логический квант выдачи разрешающего сигнала при формировании переднего (заднего) фронта в двух временных модельных тактах. 2) Квант реализации штатной функциональности (триггера регистра, счетчика) последовательностного компонента. Таким образом, модель синхронного D-триггера может быть описана в форме двух Q-покрытий, адресно вычисляющих состояния выходов:

CLK(t-1)	0	1	0	0	C(t)
CLK(t)					

Q(t-1)	0	0	0	1	0	0	1	1	Q(t)
D(t)									
C(t)									

С учетом изложенного выше модель цифровой схемы с двумя сигналами синхронизации, представленная на рис. 8, будет иметь структуры данных, состоящие из совокупности Q-покрытий, которые формируют текущий вектор моделирования M, но с учетом значений координат данного вектора в предыдущий момент времени M(t-1). Увеличение числа переменных за счет введения двух элементов синхронизации уменьшает совокупную размерность таблицы квантовых векторов, которая при 7 переменных будет иметь 56 координат.

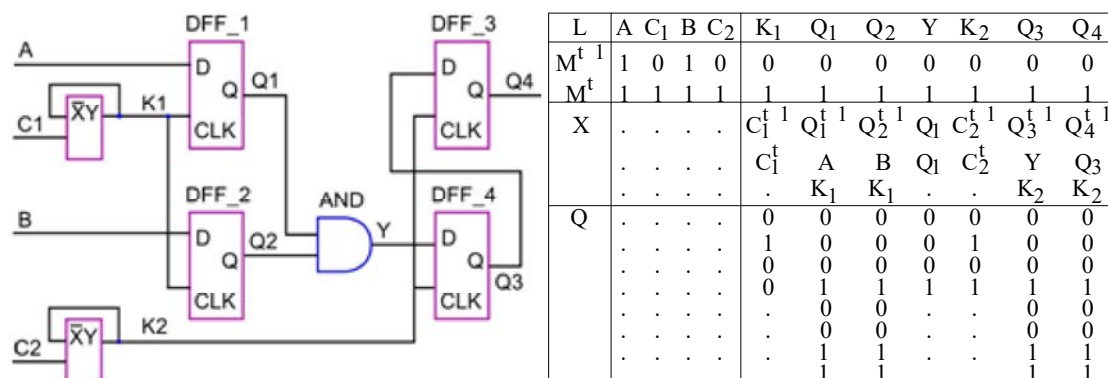
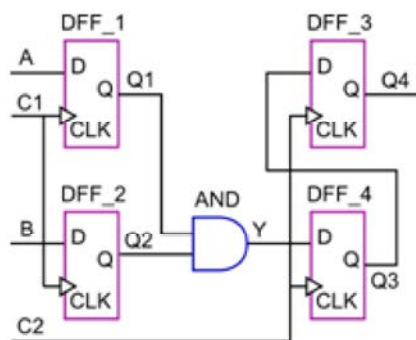


Рис. 8. Синхронизированная структура триггера

Если не вводить две дополнительные переменные (элементы синхронизации), то объем памяти для Q-покрытий будет равен 80 ячейкам (рис. 9). Данная схемная реализация максимально ориентирована на структуры данных промышленных средств моделирования и верификации. Однако квантовые векторы для задания функциональностей триггеров создают необходимые условия для повышения быстродействия интерпретативного анализа, тестирования и диагностирования схемных компонентов.

Проблема уменьшения совокупного объема Q-покрытий схемы связана с количеством переменных, формирующих адреса координат Q-вектора. Естественно, что любое разбиение числа переменных на два равных упорядоченных подмножества дает возможность существенно уменьшить размерность памяти для записи уже двух Q-векторов. В общем случае функциональная зависимость уменьшения размерности исходного Q-вектора, определенного на n-переменных, при делении на 2 подсхемы с равным числом результирующих переменных (n/2), имеет следующий вид:

$$Q = \frac{2^n}{2^{n/2} + 2^{n/2}} = \frac{2^n}{2 \times 2^{n/2}} = \frac{2^n}{2^{n/2+1}} = 2^{n-(n/2+1)} = 2^{n/2-1}.$$



L	A	C ₁	B	C ₂	Q ₁	Q ₂	Y	Q ₃	Q ₄
M ^{t-1}	1	0	1	0	0	0	0	0	0
M ^t	1	1	1	1	1	1	1	1	1
X	.	.	.	.	Q ₁ ^{t-1}	Q ₂ ^{t-1}	Q ₁	Q ₃ ^{t-1}	Q ₄ ^{t-1}
	.	.	.	.	A	B	Q ₂	Y	Q ₃
	.	.	.	.	C ₁ ^{t-1}	C ₁ ^{t-1}	.	C ₂ ^{t-1}	C ₂ ^{t-1}
	.	.	.	.	C ₁ ^t	C ₁ ^t	.	C ₂ ^t	C ₂ ^t
Q	.	.	.	.	0	0	0	0	0
	.	.	.	.	0	0	0	0	0
	.	.	.	.	0	0	0	0	0
	.	.	.	.	0	0	1	0	0
	.	.	.	.	0	0	.	0	0
	.	.	.	.	1	1	.	1	1
	.	.	.	.	0	0	.	0	0
	.	.	.	.	0	0	.	0	0
	.	.	.	.	1	1	.	1	1
	.	.	.	.	0	0	.	0	0
	.	.	.	.	1	1	.	1	1
	.	.	.	.	1	1	.	1	1
	.	.	.	.	1	1	.	1	1
	.	.	.	.	1	1	.	1	1
	.	.	.	.	1	1	.	1	1
	.	.	.	.	1	1	.	1	1

Рис. 9. Схема с D-триггерами на основе внутренней синхронизации

Например, разбиение вектора из восьми переменных на два Q-покрытия из 4 переменных уменьшает объем памяти в 8 раз. Однако следует иметь в виду, что каждое разбиение функционального модуля на k подсхем относительно внешних входов потребует k дополнительных d-циклов для вычисления состояния выхода всей схемы $T = d(k + 1)$. Снижение быстродействия разбитой функциональности является платой за существенное уменьшение объема памяти для хранения структур данных цифровой системы. В общем случае разбиение функциональности на k одинаковых частей приводит к получению следующей зависимости выигрыша объема памяти от числа разбиений на подмножества вектора

$$\text{входных переменных: } Q = \frac{2^n}{k \times 2^{n/k}} = \frac{1}{k} \times 2^{n-n/k}.$$

Здесь параметр разбиения k принимает значения, кратные степени двойки: 2, 4, 8, 16. Однако значение k не должно быть более, чем $n/2$. Следует заметить, что необязательно количество разбиений не обязательно должно принимать значения, кратные степени двойки. В общем случае, на векторе входных переменных может существовать m разбиений, каждое из которых имеет более одной переменной. При этом выполняется условие разбиения, что сумма всех переменных, участвующих в разбиениях, не может быть больше n :

$$Q = \frac{2^n}{2^{k_1} + 2^{k_2} + \dots + 2^{k_i} + \dots + 2^{k_m}}, \quad k_1 + k_2 + \dots + k_i + \dots + k_m = n.$$

Формула показывает выигрыш от разбиения функциональности в виде отношения размерности исходного квант-вектора к совокупному объему Q-векторов, полученных после разбиения. Чтобы оценить эффективность разбиения функциональности на схемные фрагменты, необходимо учитывать не только уменьшение объема памяти для хранения структур данных, но и негативные последствия, связанные со стоимостью анализа увеличенного

количества схемных компонентов, которое регламентируется в каждом конкретном случае коэффициентом d :

$$Q = \frac{2^n}{d \times m \times (2^{k_1} + 2^{k_2} + \dots + 2^{k_i} + \dots + 2^{k_m})}.$$

Подводя итог в части модификации теории исправного моделирования (fault-free) цифровых систем, можно отметить следующие факты. Автомат моделирования синхронных цифровых устройств, как правило, представлен моделью Мура:

$$S(t) = f[X(t), S(t-1)];$$

$$Y(t) = f[X(t), S(t)].$$

Здесь фигурируют входные (X) и внутренние состояния автомата в двух соседних временных фреймах $S(t)$, $S(t-1)$, а также правила определения выходных значений $Y(t)$ для инициирования вычислительных процедур. Предлагается модификация упомянутой модели автомата Мура для анализа цифровых систем, суть которой заключается в замене функциональных отношений адресными (A) операциями:

$$S(t) = A[X(t), S(t-1)];$$

$$Y(t) = A[X(t), S(t)].$$

Определенный выше адресный или квантовый автомат позволяет: 1) Избежать жестких гальванических межсоединений между элементами комбинационных и последовательностных схем при их аппаратной имплементации только в элементы памяти. 2) Получить свойство гибкой заменяемости компонентов цифровой системы в режиме online, благодаря их адресуемости. 3) Существенно упростить все процессы моделирования, верификации и тестирования путем использования только процедур вычисления адреса компонента схемы или ячейки его памяти. 4) Унифицировать процессы проектирования цифровых изделий путем их сведения к формированию функциональностей на основе вычисления адресов или к транзакциям на элементах памяти. 5) Повысить эффективность процедур моделирования цифровых схем за счет уменьшения объема интерпретативных моделей и упрощения способа их обработки, когда вместо исчерпывающего анализа таблиц предлагается вычисление адреса ячейки квантового вектора. 6) Выполнять все вычислительные процедуры на основе использования квантовых покрытий и вектора моделирования, заданного в двух соседних автоматных тактах, согласно определению квантового автомата. 7) Технологически проще становится использовать инфраструктуру [6, 7] стандартов тестопригодного проектирования (IEEE 1500, 1149) для покомпонентного тестирования, диагностирования и восстановления работоспособности адресно доступных функциональных блоков в режиме online.

8. Структура облачного сервиса QuaSim для моделирования цифровых устройств

QuaSim представляет собой средство для анализа, тестирования и верификации цифровых проектов небольшой размерности и предназначено для использования в учебном процессе в качестве облачного сервиса, доступного для студентов с любого мобильного устройства или компьютера.

Цель – существенное повышение качества учебного процесса путем предоставления технологичных микросервисов анализа цифровых устройств с одновременной визуализацией схем, тестов, результатов моделирования и таблиц истинности функциональных элементов.

Задачи: 1) Создание структуры облачного моделирования цифровых устройств на платформе компьютерных сервисов Google. 2) Разработка модуля (микросервиса) Q-element, реализующего создание модели и визуализацию логического или функционального примитива схемы. 3) Проектирование модуля генерирования кубитных моделей примитивов и более сложных цифровых устройств, а также средств их оперативной визуализации. 4) Разработка модуля или панели управления, интегрирующего симулятор, генератор логических элементов, схемных конструкций и вход-выходных портов. 5) Разработка собственно модуля для анализа цифровых схем на основе рекурсивной обработки логических

элементов. 6) Создание служебных библиотек для хранения: готовых функциональных элементов, сложных цифровых схем, тестов и результатов их анализа, а также служебной информации. 7) Тестирование и верификация облачного сервиса Q-simulator, предназначенного для интерпретативного моделирования цифровых устройств.

Сущность квантового метода анализа заключается в адресной реализации всех функциональных компонентов цифровых систем и структур данных, что дает возможность существенно повысить быстродействие интерпретативного моделирования и качество обслуживания проектов за счет быстрой замены некондиционных логических элементов путем их переадресации.

Структура облачного сервиса включает следующие основные микросервисы: 1) Q-element генерирует квантовые описания логических элементов в структуре цифровой функциональности. 2) Модуль View выполняет визуализацию схемных элементов, портов входов и выходов на экране монитора. 3) Модуль Collapse управляет окнами монитора и их размерами при помощи соответствующих иконок. 4) Контроллер Split визуализирует работу всех контроллеров при сборке схемы на экране, а также осуществляет масштабирование деталей проекта. 5) Функция Evaluate формирует состояние выхода текущего элемента путем выбора содержимого из ячейки кубита по ее адресу. 6) Модуль Q-sim реализует собственно алгоритм моделирования всех линий схемы путем построения на первом шаге рекурсивной модели для последовательно-параллельной обработки элементов. На втором шаге вычисляются состояния всех выходов логических элементов. Моделирование заканчивается после обработки всех тестовых входных последовательностей. Если схема имеет глобальные или локальные обратные связи, то моделирование осуществляется до фиксации одинаковых значений сигналов на всех линиях схемы. Если схема не устанавливается в устойчивое состояние на входных наборах, то фиксируется генераторный режим после выполнения $n (=20)$ итераций. В этом случае всем изменяющимся линиям присваивается значение двоичной неопределенности $X=\{0,1\}$. 7) Модуль управления библиотекой элементов, схем и проектов осуществляет считывание, запись и подключение фрагментов. 8) Модуль временных диаграмм осуществляет визуализацию теста с выходными сигналами на мониторе в форме непрерывных сигналов, разделенных на такты в абсолютном или модельном времени. 9) Все модули облачного сервиса запрограммированы на языке Swift, операционная система OSX 10.9, компилятор XCode 7. Количество исходных файлов 36, общее число строк кода – 1450.

На рис. 10 представлена визуализация результатов графического проектирования схемы с триггерами на мониторе компьютера. Данная схема полностью соответствует функциональности, представленной на рис. 7. Она содержит четыре входных порта для подачи рабочих или тестовых воздействий, а также два выходных порта. Структура содержит четыре логических элемента и два триггера. Мнемоническое описание компонентов схемы приведено к универсальной форме прямоугольника и различается только номером примитива в составе устройства, а также типом функциональности, которая задается кубит-вектором, представленным десятичным числом.

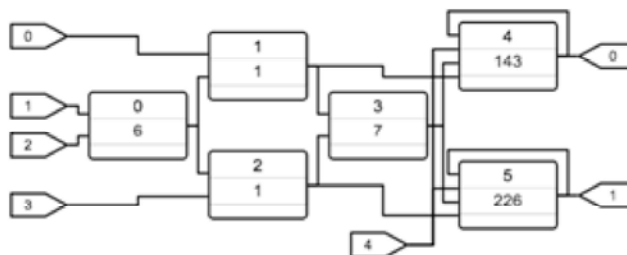


Рис. 10. Скриншот структуры с триггерами

В схеме, представленной на рис. 10, элементы имеют порядковые номера (в верхней части) и целые числа для идентификации функциональностей: 0/6 – 0110, 1/1 – 0001, 2/1 – 0001, 3/7 – 0111, 4/143 – 11110001, 5/226 – 01000111. Здесь двоичный вектор соответствует десятичному эквиваленту числа для задания функциональности. Поскольку кубит-вектор не имеет в явном виде задания входных наборов, то его можно рассматривать как неявную или компактную форму теоретико-множественной по сути таблицы истинности. Зачем явно указывать входные значения, если они составляют строго последовательную адресацию выходных значений? Таким образом, таблица истинности, как совокупность входных сигналов и соответствующих им выходных значений всегда проигрывает перед кубит-векторной формой представления функциональностей в плане объема и быстродействия анализа данных. Не существует принципиальных различий между описаниями комбинационного элемента, схемы или последовательностного примитива, поскольку все они формально представлены кубит-векторами, которые помещаются в адресуемую память. Более того, все примитивы схемы также являются адресуемыми, а структура схемы может быть описана в виде кубит-вектора. Таким образом, можно прийти к такой реализации вычислительного устройства, где нет ничего, кроме адресной памяти или кубитных векторов различной длины, в которых функциональности определяются упорядоченными наборами нулевых и единичных сигналов. Преимущества данного сервиса QuaSim кубитного описания и моделирования цифровых устройств заключаются в следующем: 1) Все функциональные элементы и схемы задаются Q-векторами, что унифицирует процедуры синтеза и анализа цифровых устройств. 2) Технологически просто менять или корректировать функциональность схемы или любого примитива путем замены отдельных битов Q-вектора. 3) Унификация кубитной формы описания примитивов схемы дает возможность применить к ним единственную процедуру анализа функциональностей, которая сводится к вычислению адреса $M_i = Q_i[M(X_i)]$, что делает процесс программирования облачного сервиса QuaSim технологически простым в исполнении и не зависящим от функциональной и структурной сложности цифровых структур. 4) Простой и понятный начинающему пользователю графический интерфейс делает облачный сервис конкурентоспособным на рынке образовательных услуг, где сложные и тяжеловесные средства моделирования от ведущих компаний планеты являются недоступными для университетов из-за их высокой стоимости, а для студентов – времязатратными по сложности подготовки HDL-спецификаций при рассмотрении небольших учебных проектов. 5) Унификация формы описания примитивов создает условия для технологичного решения задач синтеза, моделирования неисправностей, тестирования, верификации и диагностирования, основанные на операциях с кубит-векторами. 6) Недостатком кубитной или квантовой технологии описания и анализа цифровых структур можно считать некоторое уменьшение быстродействия моделирования по сравнению с существующими промышленными компиляторами, для ASIC и VLSI проектов, где объем reusable logic является доминирующим для достижения высокого быстродействия.

Структура взаимодействующих компонентов облачного сервиса QuaSim представлена на рис. 11. Квантовое или кубитное представление модели цифрового устройства вместе с интерпретативным симулятором составляют ядро системы, интегрированной в большие данные киберпространства или Интернета. Это дает возможность использовать в качестве исходных данных открытые спецификации и тестбенчи, описанные на языках VHDL, Verilog. Такие данные и/или тестовые примеры имеются практически во всех ведущих компаниях, университетах и тематических конференциях IEEE, TTTC, ISCAS. Кроме того, погружение Q-sim сервиса в интернет-пространство предполагает также выгрузку результатов его работы, связанную с анализом и синтезом учебных или рыночно ориентированных проектов в сервисы хранения данных на платформах Google, Amazon, Microsoft, IBM, Facebook. Естественно, что интеграция облачного сервиса с киберпространством предполагает наличие парсер-микросервисов для преобразования спецификаций из языков описания аппаратуры во внутренний язык QuaSim. Должно также существовать и обратное преобразование данных из кубитного представления в стандарты HDL-языков. Парсеризация обеспечивает возможность использования открытых в интернете проектов для их изучения и сравнения в системе моделирования Q-sim, а также делает доступными внутренние проектные решения QuaSim для всех желающих на рынке образовательных сервисов.

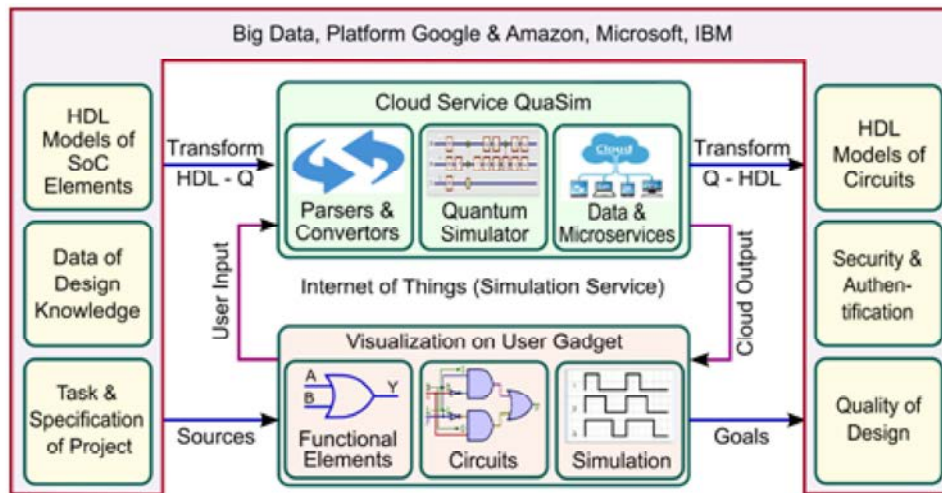


Рис. 11. Облачный сервис моделирования цифровых устройств

Блок Security контролирует доступ пользователей в целях их статистического учета и предполагает аутентификацию каждого на основе пароля, фамилии, имени, дополненной любым валидным (корпоративным) атрибутом из списка: {электронная цифровая подпись, e-mail, цифровой ключ, номер телефона}.

Тестирование и верификация облачного сервиса моделирования цифровых систем осуществлялись отдельно для каждого микросервиса, а затем во взаимодействии всех модулей.

1) Проверка правильности генерирования логических и более сложных функциональных элементов.

2) Проверка структурного синтеза цифровой схемы и средств визуализации.

3) Верификация и тестирование алгоритмов двоичного и троичного синхронного интерпретативного исправного моделирования входных воздействий на 40 схемах, комбинационного и последовательностного типов. При этом использовались тестовые наборы, алгоритмически генерируемые и составленные пользователем.

4) Проверка сервисных модулей, обеспечивающих работоспособность основных микросервисов: библиотеки элементов и схем, аутентификация пользователя, модуль формирования статистических данных по проектам и пользователям.

5) Верификация интерфейсных микросервисов, обеспечивающих интеракции между облачными back-end и пользовательскими front-end модулями.

9. Заключение

Сущность предлагаемого научно-технологического исследования заключается в создании векторных структур данных и кубитных методов синтеза, тестирования и моделирования, интегрированных в облачную инфраструктуру сервисного обслуживания компонентов цифровых систем на кристаллах в целях повышения качества изделий и выхода годной продукции за счет адресуемости всех вычислительных процессов и явлений. Основная инновационная идея Memory-Address-Transaction модели вычислений заключается в синтезе и анализе векторных цифровых структур на основе адресуемых элементов памяти, исключающих использование reusable or new logic. Трудно создать двумерный регистр, соответствующий матрице смежностей или таблице истинности, поэтому привести описание функции и структуры к единому одномерному формату, означает – технологично решать все задачи синтеза и анализа для функциональностей и графов в кубитно-векторной метрике, создающей memory-driven computing на основе выполнения параллельных логических операций.

Предлагается методология проектирования цифровых схем, на основе элементов памяти для синтеза компонентов операционного и управляющего автоматов, составляющих операционное устройство.

Показаны основы кубитно-векторного синтеза путем суперпозиции кубитных покрытий black box функциональностей, имплементируемых в элементы памяти, что дает возможность существенно повысить быстродействие средств моделирования, тестирования и верификации, а также упростить процессы создания реальных и виртуальных компьютерных систем.

Предложены кубитные структуры данных для моделирования и тестирования цифровых систем, которые дают возможность существенно упростить реализацию алгоритмов и повысить их быстродействие за счет адресуемости функциональных квантов и параллельности обработки примитивов.

Показана реализация вычислительных структур и процессов на основе использования адресного автомата, который дает возможность привлекать инфраструктуру стандартов тестопригодного проектирования для повышения выхода годной продукции, за счет online восстановления работоспособности функциональных примитивов.

Практическая значимость кубитной методологии синтеза и анализа цифровых систем заключается в имплементации процессора на основе элементов памяти, что делает его однородным по структуре функциональных примитивов и доставляет технологические удобства при реализации процессов проектирования, производства и эксплуатации, включая верификацию, встроенные тестирование, диагностирование и ремонт в режиме online за счет использования универсальных адресуемых spare-компонентов памяти. Кроме того, моделирование проектируемых вычислителей на основе адресуемых моделей элементов делает данную процедуру простой за счет регулярных структур данных и использования операции транзакции на элементах памяти, а также быстродействующей, благодаря параллельной обработке массивов однотипной памяти. Имплементация кубитных моделей описания цифровых компонентов и систем работает на увеличение выхода годной продукции, повышение надежности вычислительных изделий, снижение стоимости проектирования и изготовления, а также автономное восстановление работоспособности в режиме online без участия человека.

Предложен облачный сервис QuaSim для моделирования и верификации цифровых систем на основе транзакций между адресуемыми компонентами памяти для реализации любой функциональности. Описан новый подход к синтезу и анализу цифровых систем, использующий векторную форму (квант) задания комбинационных и последовательностных структур для их имплементации в элементы памяти, что существенно отличается от общепринятой теории проектирования дискретных устройств на основе таблиц истинности компонентов. Используются квантовые или кубитные структуры данных [1-5] для реализации вычислительных процессов в целях повышения быстродействия анализа цифровых систем и уменьшения объемов памяти на основе унарного кодирования состояний входных, внутренних и выходных переменных и имплементации кубитных векторов в элементы памяти FPGA, реализующих комбинационные и последовательностные примитивы.

Список литературы: 1. *Metodi T., Chong F.* Quantum Computing for Computer Architects. Synthesis Lectures on Computer Architecture. Morgan & Claypool. 2006. 154 p. 2. *Stenholm Stig, Kalle-Antti Suominen.* Quantum approach to informatics. John Wiley & Sons, Inc. 2005. 249p. 3. *Hahanov V.I., Wajeb Gharibi, Litvinova E.I., Shkil A.S.* Qubit data structure of computing devices // Electronic modeling. 2015. № 1. P.76-99. 4. *Vladimir Hahanov, Tamer Bani Amer, Ivan Hahanov.* MQT-model for Virtual Computer Design // Proc. of Microtechnology and Thermal Problems in Electronics (Microtherm). 23-25 June 2015. P. 182-185. 5. *Hahanov V.I., Litvinova E.I., Chumachenko S.V.* et al. Qubit Model for solving the coverage problem // Proc. of IEEE East-West Design and Test Symposium. Kharkov. 14-17 September, 2012. P.142 – 144. 6. *Zorian Y. Shoukourian S.* Test solutions for nanoscale Systems-on-Chip: Algorithms, methods and test infrastructure. Computer Science and Information Technologies (CSIT), 2013. P. 1 – 3. 7. *Zorian Y., Shoukourian S.* Embedded-memory test and repair: infrastructure IP for SoC yield. Design & Test of Computers, IEEE (Volume: 20, Issue: 3). P. 58 – 66. 8. *Dugganapally I.P., Watkins S.E., Cooper B.* Multi-level, Memory-Based Logic Using CMOS Technology. VLSI (ISVLSI), 2014 IEEE Computer Society Annual Symposium on. Tampa, FL. P. 583-588. 9. *Yueh W., Chatterjee S., Zia M., Bhunia S., Mukhopadhyay S.* A Memory-Based Logic Block With Optimized-for-Read SRAM for Energy-Efficient Reconfigurable Computing Fabric. Circuits and Systems II: Express Briefs, IEEE Transactions on. Vol. 62. Issue: 6. P. 593-597. 10. *Matsunaga S., Hayakawa J., Ikeda*

S., Miura K., Endoh T., Ohno H., Hanyu T. MTJ-based nonvolatile logic-in-memory circuit, future prospects and issues. Design, Automation & Test in Europe Conference & Exhibition, 2009. DATE '09. P. 433 – 435. **11.** *Harada S., Xu Bai, Kameyama M., Fujioka Y.* Design of a Logic-in-Memory Multiple-Valued Reconfigurable VLSI Based on a Bit-Serial Packet Data Transfer Scheme. Multiple-Valued Logic (ISMVL), 2014 IEEE 44th International Symposium on. P. 214 – 219. **12.** *Hahanov V.I., Tamer Bani Amer, Chumachenko S.V., Litvinova E.I.* Qubit technology analysis and diagnosis of digital devices // Electronic modeling. 2015. Vol. 37, № 3. P. 17-40. **13.** *Melikyan V.Sh.* A method of eliminating false paths during statistical static analysis of timing delays of digital circuits // Elektronika i svyaz. 2009. Vol. 2-3, No. 1. P. 93-96. **14.** *Melikyan V.Sh., Vatyana A.O.* Interconnections model delays for the logic analysis of ECL circuits // S UAB, Vol. 2, Computer Engineering, Moscow, 1997. P. 187-194. **15.** *Хаханов И.В., Литвинова Е.И.* Синтез та аналіз «квантових» моделей цифрових систем // АСУ та прилади інформатики. 2015. Вип. 172. С. 56–70.

Поступила в редколлегию 14.01.2016

Tamer Bani Amer, аспирант ХНУРЭ. Научные интересы: квантовые вычисления, тестирование и диагностика цифровых систем. Адрес: Украина, 61166, Харьков, пр. Науки, 14, тел. +3805770-21-326.

Хаханов Иван Владимирович, студент факультета компьютерной инженерии и управления ХНУРЭ. Научные интересы: техническая диагностика цифровых систем, программирование. Увлечения: горные лыжи, английский язык. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +380 57 70-21-326.

Литвинова Евгения Ивановна, д-р техн. наук, профессор кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем, сетей и программных продуктов. Адрес: Украина, 61166, Харьков, пр. Ленина, 14, тел. +380 57 70-21-326. E-mail: kiu@kture.kharkov.ua.

Емельянов Игорь Валерьевич, н.с. кафедры АПВТ ХНУРЭ. Научные интересы: техническая диагностика цифровых систем, сетей и программных продуктов. Адрес: Украина, 61166, Харьков, пр. Науки, 14, тел. +3805770-21-326.

МЕТОД КОСВЕННОГО СТЕГАНОГРАФИЧЕСКОГО ВСТРАИВАНИЯ НА ОСНОВЕ ФУНКЦИОНАЛЬНОГО ПРЕОБРАЗОВАНИЯ ДЛЯ АДАПТИВНОГО ПОЗИЦИОННОГО ЧИСЛА

Предлагается подход для проектирования стеганографической системы косвенного встраивания. Формулируются требования для функционального преобразования элементов изображения при стеганографическом встраивании. Приводятся свойства функционального преобразования для адаптивного позиционного числа. Разрабатывается метод косвенного стеганографического встраивания на основе использования функционального преобразования для адаптивного представления элементов изображения-контейнера. На основе метода проектируется стеганографическая система косвенного встраивания. На примере показывается процесс встраивания и извлечения встроенной информации и реконструкцией элементов изображения-контейнера.

1. Введение

Для повышения безопасности информационных ресурсов при передаче в инфокоммуникационных системах используются методы стеганографического встраивания информации в изображение.

В отличие от методов криптографической защиты стеганографические алгоритмы позволяют избежать прямых атак на скрываемое сообщение. Современные методы компьютерной стеганографии на базе изображения-контейнера условно разделены на алгоритмы:

- непосредственного стеганографического встраивания;
- косвенного стеганографического встраивания.

В сравнении с методами непосредственного встраивания методы косвенного встраивания обладают повышенной стойкостью встроенных данных в условиях применения злоумышленником активных воздействий (атак). Однако при повышенных вычислительных возможностях и разработке новых подходов для осуществления атак существующие методы косвенного встраивания не обеспечивают в полной мере системных требований к скрытию информации.

К общим недостаткам существующих методов относится использование для стеганографического встраивания психовизуальной избыточности цифровых изображений. Как правило, атаки, направленные на разрушение встроенной информации, представляют собой методы, позволяющие устранять психовизуальную избыточность. Поэтому для повышения стойкости встроенных данных в условиях изменения злоумышленником активных атак предлагается подход, основанный на использовании функционального преобразования элементов изображения-контейнера. При этом встраивание предлагается осуществлять за счет применения структурной избыточности контейнера.

2. Концепция функционального преобразования для косвенного стеганографического встраивания

Для реализации функционального преобразования элементов предлагается синтезировать функционал, который должен обеспечить следующие требования:

1. Функциональное преобразование должно обеспечивать взаимоднозначное кодирование $f(\bullet)$ и декодирование $f^{-1}(\bullet)$ массива A изображения-контейнера при наличии служебной информации Ψ , т.е. $S = f(C, \Psi)$, $C' = f^{-1}(S, \Psi)$ и $C' = C$.

Здесь C' – массив, восстановленный в результате обратного функционального преобразования $f^{-1}(S)$; Ψ – служебная информация; S – массив, полученный при выполнении прямого функционального преобразования.

2. В результате функционального преобразования массива C должна формироваться кодограмма S , которая состоит из двух частей:

- служебной, содержащей служебные данные Ψ ;
- информационной , содержащей кодовое представление массива A .

3. Значения реконструированных массивов C' и C'' не должны меняться в случае формирования кода при различных значениях служебной информации (Ψ и Ψ'), т.е.

$$C' = f^{-1}(S, \Psi) = f^{-1}(S', \Psi') = C'',$$

где C' – массив, реконструированный на основе кода, сформированного с учетом служебных данных Ψ ; C'' – массив, реконструированный на основе кода, сформированного с учетом модифицированных служебных данных Ψ' ; S – кодограмма, полученная с учетом служебных данных Ψ ; S' – кодограмма, полученная с учетом служебных данных Ψ' .

Предлагается использовать это свойство для косвенного стеганографического встраивания. Тогда процесс встраивания будет включать намеренное изменение служебной информации Ψ на основе ключевого условия. Сформированная кодограмма S' , содержащая модифицированные служебные данные Ψ' , передается по каналу. При этом на приемной стороне авторизованному пользователю известно условие косвенного встраивания, т.е. механизма модификации исходной служебной информации Ψ . В этом случае процесс стеганографического изъятия будет осуществляться путем анализа значений исходной Ψ и измененной Ψ' служебной информации.

Тогда прямое косвенное стеганографическое преобразование будет включать следующие этапы:

1. Формирование вектора служебных данных Ψ для массива C'' (2) изображения контейнера.

2. Модификацию вектора служебных данных Ψ с учетом встраиваемого элемента b_ξ на основе ключевого условия: $\Psi' = \Psi + b_\xi$. Здесь b_ξ – элемент скрываемого сообщения $B = \{b_1; \dots; b_\xi; \dots; b_v\}$, $\xi = \overline{1, v}$.

3. Функциональное преобразование массива C с учетом модифицированного вектора служебных данных Ψ' по правилу $f(C)$, т.е. $S = f(C, \Psi')$, где S – сформированное значение кодограммы.

Полученная кодограмма, содержащая в себе информационную составляющую S и служебную составляющую Ψ' , передается в канал передачи данных, где может подвергаться атакующим воздействиям.

Обратное косвенное стеганографическое преобразование осуществляется по биполярному принципу для авторизованного и неавторизованного пользователя.

При неавторизованном доступе, по правилу $f^{(-1)}(\bullet)$ осуществляется реконструкция исходного массива изображения-контейнера: $C'' = f^{(-1)}(S; \Psi')$. Здесь C'' – массив исходного изображения, полученный в результате неавторизованного доступа.

Наоборот, обратное косвенное стеганографическое преобразование для авторизованного пользователя осуществляется с учетом ключевого условия изъятия и содержит следующие этапы:

1. На первом этапе по правилу $f^{(-1)}(\bullet)$ реконструируется массив C' исходного изображения контейнера: $C' = f^{(-1)}(S; \Psi')$. Здесь S – принятая кодограмма, сформированная на передающей стороне с учетом модифицированных служебных данных Ψ' .

2. На втором этапе для реконструированного массива C' по ключевому правилу формируется исходный вектор служебных данных Ψ .

3. Третий этап включает косвенное изъятие встроенного элемента b'_ξ скрываемого сообщения $B' = \{b'_1; \dots; b'_\xi; \dots; b'_v\}$ на основе ключевого условия изъятия при анализе восстановленного Ψ и полученного Ψ' векторов служебных данных: $b'_\xi = \Psi' - \Psi$.

3. Метод косвенного стеганографического встраивания на основе использования функционального преобразования для адаптивного позиционного числа

В качестве такого функционального преобразования предлагается использовать кодообразующую функцию для адаптивного позиционного числа, а в качестве элемента изображения-контейнера – фрагмент изображения F , размерность m строк и n столбцов.

Данное функциональное преобразование позволяет выявить структурные закономерности в изображении. Такие закономерности обусловлены ограничением на динамический диапазон. Величина Ψ динамического диапазона представления фрагмента F изображения-контейнера определяется на основе следующего выражения:

$$\Psi = \max_{1 \leq i \leq m} \{c_{i,j}\} + 1, \quad j = \overline{1, n}.$$

Здесь $c_{i,j}$ – j -й элемент в i -й строке массива F .

В процессе реализации функционального преобразования на основе адаптивного позиционного кодирования фрагмент F исходного изображения рассматривается как множество адаптивных позиционных чисел $\{C(j)\}$: $C(j) = \{c_{1,j}; \dots; c_{i,j}; \dots; c_{m,j}\}$.

Значения кода $K(j)$ будет определяться как сумма произведений элементов позиционного числа $C(j)$ на их весовые коэффициенты $V_{i,j}$ по формуле: $K(j) = \sum_{i=1}^m c_{i,j} V_i$. Здесь $c_{i,j}$ – $(i; j)$ -й элемент адаптивного позиционного числа $C(j)$; V_i – весовой коэффициент элемента $c_{i,j}$ адаптивного позиционного числа $C(j)$ фрагмента F .

Весовой коэффициент V_i элемента $c_{i,j}$ зависит от его позиции в числе $C(j)$ и вычисляется как произведение оснований всех младших элементов. Весовой коэффициент V_i вычисляется по следующей формуле: $V_i = \prod_{\xi=1+1}^m \psi_{\xi}$.

Второй этап предусматривает формирование кодограммы $S(F)$, которая включает служебную составляющую $S(\Psi)$ и информационную составляющую $S(j)$. Данный этап реализуется при помощи оператора выделения разрядов $\varphi_c(\bullet)$ по формуле:

$$S(F) = \varphi_c(S(j), \Psi),$$

где Ψ – базис, содержащий информацию об основаниях для адаптивных позиционных чисел фрагмента F ; $S(j)$ – кодограмма кодового представления адаптивного позиционного числа $C(j)$.

Кодограмма $S(j)$ имеет следующий вид:

$$S(j) = \{s_1, \dots, s_{\xi}, \dots, s_{q(S(j))}\},$$

здесь $q(S(j))$ – длина двоичной кодограммы $S(j)$; s_{ξ} – ξ -й двоичный разряд кодограммы $S(j)$.

Процесс реконструкции элемента $c_{i,j}$ для адаптивного позиционного числа $C(j)$ на основе кода $K(j)$ выполняется по формуле

$$c'_{i,j} = [K(j)/V_i] - [K(j)/(\psi_i V_i)] \psi_i$$

или

$$c'_{i,j} = \left[\sum_{i=1}^m c_{i,j} V_i / V_i \right] - \left[\sum_{i=1}^m c_{i,j} V_i / (\psi_i V_i) \right] \psi_i.$$

Такое преобразование осуществляется без внесения искажений.

В случае адаптивного представления значение реконструированного элемента $c_{i,j}$ числа $C(j)$ фрагмента F не меняется при кодировании и декодировании с различными основаниями ψ_i и ψ'_i , т.е.

$$c'_{i,j} = c_{i,j} = [K(j)/V_i] - [(K(j)/(\psi_i V_i))] \psi_i = [K'(j)/V'_i] - [(K'(j)/(\psi'_i V'_i))] \psi'_i = c''_{i,j},$$

где $c'_{i,j}$ – элемент числа $C(j)$, реконструированный на основе системы оснований Ψ ; $c''_{i,j}$ – элемент числа $C(j)$, реконструированный на основе системы оснований Ψ' ; $K(j)$ – кодовое представление числа $C(j)$, сформированное в базисе оснований Ψ ; $K'(j)$ – кодовое представление числа $C(j)$, сформированное в базисе оснований Ψ' ; ψ'_i – значение модифицированного основания элемента $c'_{i,j}$.

Графически это показано на рис. 1. Предлагается использовать данное свойство для проектирования системы косвенного стеганографического встраивания. Косвенное встраивание элемента b_ξ скрываемого сообщения $B = \{b_1; \dots; b_\xi; \dots; b_v\}$ предлагается проводить в блок изображения-контейнера путем модификации основания ψ_i базиса Ψ на основе следующего правила: $\psi'_i = \psi_i + k$, где $k = b_\xi$. Здесь ψ' – основание, модифицированное в результате косвенного стеганографического встраивания; k – коэффициент модификации.

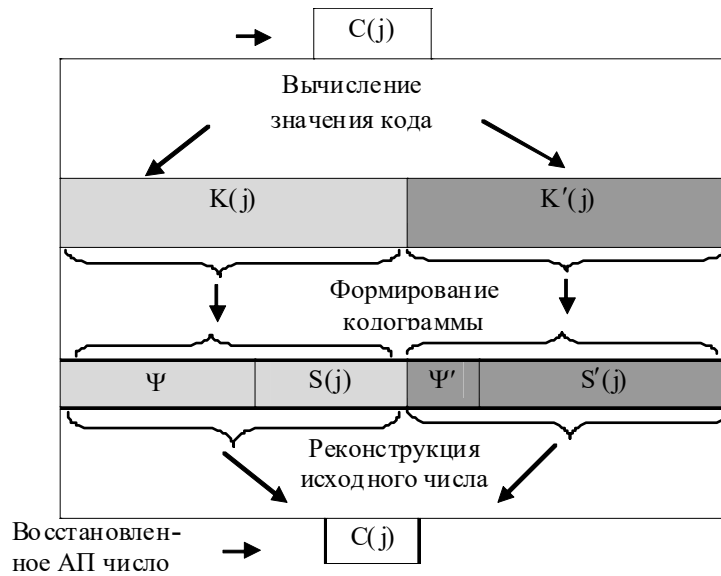


Рис. 1. Графическая интерпретация свойства адаптивного позиционного кодирования.

В процессе косвенного стеганографического встраивания необходимо учитывать потенциальную возможность увеличения длины $q(S'(j))$ кодограммы $S'(j)$ относительно длины $q(S(j))$ исходной кодограммы. Поэтому для уменьшения уровня вносимой избыточности R встраивать элементы предлагается в двоичном представлении $b_\xi \in [0; 1]$, а коэффициент k модификации выбирать на основе следующего правила:

$$k = \begin{cases} 0, & b_\xi \rightarrow 0; \\ 1, & b_\xi \rightarrow 1. \end{cases}$$

В этом случае косвенное встраивание бита $b_\xi \in [0; 1]$ будет выполняться по формуле:

$$\psi'_i = \begin{cases} \psi_i, & b_\xi \rightarrow 0; \\ \psi_i + 1, & b_\xi \rightarrow 1. \end{cases}$$

Таким образом, предложенный подход позволяет осуществить косвенное стеганографическое встраивание сообщения $B = \{b_1; \dots; b_\xi; \dots; b_v\}$, $b_\xi \in [0; 1]$, $\xi = \overline{1, v}$ в блоки исходного изображения-контейнера.

Теперь рассмотрим этапы функционирования разработанной стеганографической системы косвенного встраивания (рис. 2).

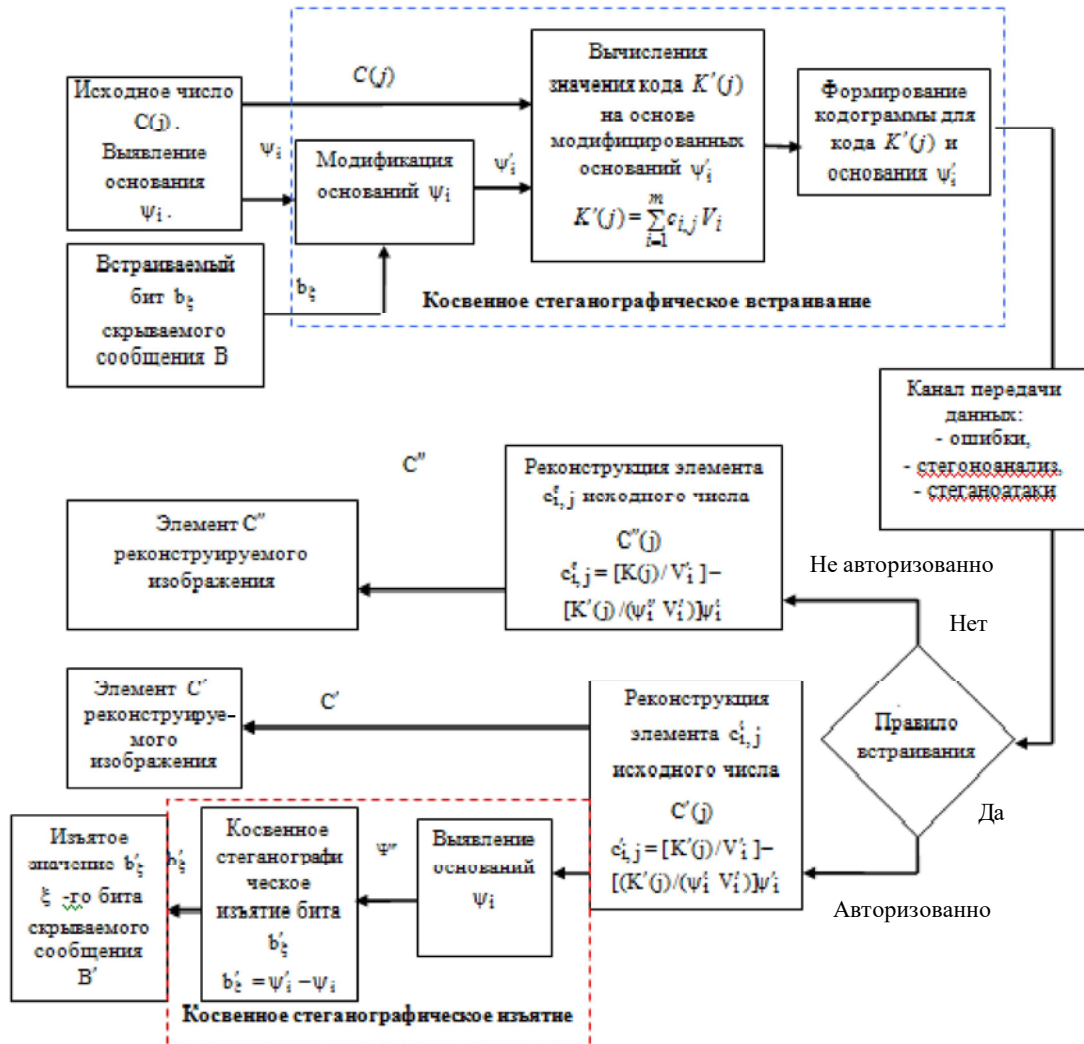


Рис. 2. Структурная схема стеганографической системы косвенного встраивания

Данная система позволяет встроить бит скрываемого сообщения путем модификации оснований адаптивного позиционного числа. Косвенное стеганографическое встраивание включает в себя следующие действия:

1. Выявление оснований в выбранном блоке $F_{\tau, \gamma}$.

Данный этап реализуется на основе следующего выражения: $\psi_i = \max_{1 \leq i \leq m} \{c_{i,j}\} + 1$.

2. Косвенное встраивание бита b_ξ скрываемого сообщения $B = \{b_1; \dots; b_\xi; \dots; b_v\}$, $\xi = \overline{1, v}$ путем модификации основания ψ_i по следующему правилу:

$$\psi'_i = \begin{cases} \psi_i, & b_\xi \rightarrow 0; \\ \psi_i + 1, & b_\xi \rightarrow 1. \end{cases}$$

3. Формирование кода $K'(j)$ для адаптивного позиционного числа $C(j)$ блока $F_{\tau,\gamma}$. Вычисление кода $K'(j)$ выполняется с учетом модифицированных оснований $\{\psi'_i\}$ по

формуле:
$$K'(j) = \sum_{i=1}^m c_{i,j} V_i.$$

4. Формирование кодограммы, которая содержит служебную $S(\Psi')$ (основания $\{\psi'_i\}$) и информационную $S(j)$ составляющую.

Схема косвенного стеганографического изъятия включает следующие этапы:

1. Извлечение из кодограммы кода $K'(j)$ при помощи оснований $\{\psi'_i\}$.
2. Восстановление элементов исходного числа: $c'_{i,j} = [K'(j)/V'_i] - [K'(j)/(\psi'_i V'_i)] \psi'_i$.
3. Выявление исходного основания $\{\psi_i\}$ по формуле:

$$\psi_i = \max_{1 \leq i \leq m} \{c_{i,j}\} + 1,$$

где ψ'_i – i-е основание восстановленного базиса Ψ'' .

4. Косвенное изъятие встроенного бита b'_ξ . Данный этап реализуется на основе сравнения модифицированного ψ'_i и восстановленного ψ''_i основания на основе следующего выражения:

$$b'_\xi = \begin{cases} 0, & \rightarrow \psi'_i - \psi''_i = 0; \\ 1, & \rightarrow \psi'_i - \psi''_i = 1, \end{cases}$$

или $b'_\xi = \psi'_i - \psi''_i$.

Теперь рассмотрим обратное стеганографическое преобразование при неавторизованном доступе. В этом случае у злоумышленника отсутствует ключевое правило встраивания, а декодирование будет содержать следующие действия:

1. Извлечение из кодограммы кода $K'(j)$ при помощи оснований $\{\psi'_i\}$.
2. Восстановление элементов исходного числа: $c''_{i,j} = [K'(j)/V'_i] - [K'(j)/(\psi'_i V'_i)] \psi'_i$,

где $c''_{i,j}$ – i-й элемент реконструируемого числа $C''(j)$ как составляющей реконструируемого фрагмента F при неавторизованном доступе.

Рассмотрим пример косвенного стеганографического встраивания бита $b_\xi = 1$ в фрагмент изображения F :

$$F = \begin{vmatrix} 2 & 2 & 3 & 1 \\ 1 & 4 & 3 & 5 \\ 4 & 1 & 4 & 2 \\ 1 & 4 & 1 & 4 \end{vmatrix}.$$

Определим основания для элементов фрагмента F : $\psi_i = \max_{1 \leq i \leq m} \{c_{i,j}\} + 1$.

На следующем этапе проведем косвенное встраивание бита $b_\xi = 1$ в полученный базис оснований $\psi = 5$. Для этого на основе ключевого правила вычислим модифицированное основание ψ' : $\psi' = \psi + 1 = 5 + 1 = 6$.

Теперь представим фрагмент изображения F в кодовом виде с учетом модифицированного основания ψ' . В этом случае для каждого адаптивного позиционного числа $C(j)$ фрагмента F сформируем значение кода $K(j)$:

$$K(1) = \sum_{i=1}^4 c_{i,1} \cdot V'_i = 2 \cdot 216 + 1 \cdot 36 + 4 \cdot 6 + 1 \cdot 1 = 493,$$

$$K(2) = \sum_{i=1}^4 c_{i,2} \cdot V'_i = 2 \cdot 216 + 4 \cdot 36 + 1 \cdot 6 + 4 \cdot 1 = 448,$$

$$K(3) = \sum_{i=1}^4 c_{i,3} \cdot V'_i = 3 \cdot 216 + 3 \cdot 36 + 4 \cdot 6 + 1 \cdot 1 = 761,$$

$$K(4) = \sum_{i=1}^4 a_{i,4} \cdot V'_i = 1 \cdot 216 + 5 \cdot 36 + 2 \cdot 6 + 4 \cdot 1 = 412.$$

В результате косвенного стеганографического встраивания бита $b_{\xi} = 1$ в фрагмент изображения-контейнера была получена кодовая последовательность, представленная на рис. 3.



Рис. 3. Кодовое представление фрагмента F, полученное в результате косвенного стеганографического встраивания

Теперь рассмотрим пример косвенного стеганографического изъятия при авторизованном доступе. В этом случае пользователю известна ключевая информация, которая представляет собой правило встраивания.

На первом этапе восстановим значения элементов $\{c'_{i,j}\}$ чисел $\{C'(j)\}$ исходного фрагмента F' . Для этого используем следующую формулу:

$$c'_{i,j} = [K'(j) / V'_i] - [K'(j) / (\Psi'_i V'_i)] \Psi'_i,$$

где $K'(j)$ – значение кода, считанное авторизованным пользователем.

В качестве примера осуществим восстановление числа $C'(2)$, которое соответствует второму столбцу фрагмента F' изображения-контейнера:

$$\begin{aligned} c'_{1,2} &= [K'(2) / V'_1] - [K'(2) / (\Psi'_1 V'_1)] \Psi'_1 = \\ &= [448 / 216] - [448 / 216 \cdot 6] \cdot 6 = 2; \end{aligned}$$

$$\begin{aligned} c'_{2,2} &= [K'(2) / V'_2] - [K'(2) / (\Psi'_2 V'_2)] \Psi'_2 = \\ &= [448 / 36] - [448 / 36 \cdot 6] \cdot 6 = 4; \end{aligned}$$

$$\begin{aligned} c'_{3,2} &= [K'(2) / V'_3] - [K'(2) / (\Psi'_3 V'_3)] \Psi'_3 = \\ &= [448 / 6] - [448 / 6 \cdot 6] \cdot 6 = 1; \end{aligned}$$

$$\begin{aligned} c'_{4,2} &= [K'(2) / V'_4] - [K'(2) / (\Psi'_4 V'_4)] \Psi'_4 = \\ &= [448 / 1] - [448 / 1 \cdot 6] \cdot 6 = 5. \end{aligned}$$

После декодирования восстановленный фрагмент F' изображения контейнера примет следующий вид:

$$F' = \begin{vmatrix} 2 & 2 & 3 & 1 \\ 1 & 4 & 3 & 5 \\ 4 & 1 & 4 & 2 \\ 1 & 5 & 1 & 4 \end{vmatrix}.$$

Для изъятия встроенного сообщения бита $b_{\xi} = 1$ авторизированному пользователю необходимо определить основание ψ'' для фрагмента F и сравнить его с основанием ψ' , изъятим из служебной составляющей кодограммы. Основания ψ'' определяются по формуле $\psi'_i = \max_{1 \leq j \leq m} \{c_{i,j}''\} + 1$. Тогда получим следующее значение исходного основания:

$\psi = 5$. Теперь на основе правила $b_{\xi} = \psi'_i - \psi''_i$ осуществим косвенное изъятие встроенного бита: $b_{\xi} = \psi' - \psi'' = 6 - 5 = 1$,

В результате выполнения обратного стеганографического преобразования был безошибочно восстановлен встроенный бит $b_{\xi} = 1$.

4. Выводы

Для устранения недостатков существующих стеганографических методов косвенного встраивания предложен подход в виде применения функционального преобразования элементов изображения-контейнера. Сформулированы требования для функционального преобразования.

Разработан метод косвенного стеганографического встраивания на основе использования функционального преобразования для адаптивного позиционного числа, который базируется на следующих этапах:

- формирование основания для фрагмента изображения;
- косвенное стеганографическое встраивание путем модификации оснований с учетом встраиваемой информации;
- формирование кодового представления элементов изображения-контейнера.

На основе сформулированного метода разработана система косвенного стеганографического встраивания скрываемой информации, которая включает следующие этапы:

1. Этап косвенного стеганографического встраивания.
2. Этап косвенного стеганографического изъятия встроенной информации.

Список литературы: 1. Грибунин В.Г., Оков И.Н., Туринцев И.В., Цифровая стеганография. М.: Солон-Пресс, 2002. 272 с. 2. Конахович Г.Ф., Пузыренко А.Ю., Компьютерная стеганография. Теория и практика. К.: МК-Пресс, 2006. 288 с. 3. Тарасов Д.О., Мельник А.С., Голобородько М.М. Класифікація та аналіз безкоштовних програмних засобів стеганографії // Інформаційні системи та мережі. Вісник НУ "Львівська політехніка" № 673. Львів. 2010. С. 365-374.

Поступила в редколлегию 23.12.2015

Юдин Александр Константинович, д-р техн. наук, профессор, директор института компьютерных информационных технологий Национального авиационного университета. Адрес: Украина, 01000, Киев, пр.Космонавта Комарова, 1.

Баранник Владимир Викторович, д-р техн. наук, профессор, начальник кафедры боевого применения и эксплуатации АСУ Харьковского университета Воздушных Сил. Научные интересы: обработка и передача информации. Адрес: Украина, 61023, Харьков, ул. Сумская, 77/79.

Фролов Олег Владимирович, соискатель Национального авиационного университета. Адрес: Украина, 01000, Киев, пр.Космонавта Комарова, 1.

РАЗРАБОТКА ТЕОРЕТИЧЕСКИХ ОСНОВ СТРУКТУРНОГО ПРЕДСТАВЛЕНИЯ ДАННЫХ В ДВОИЧНОМ ПОЛИАДИЧЕСКОМ ПРОСТРАНСТВЕ

Создается двумерное однопризнаковое структурное кодирование двоичных данных по количеству серий единиц в полиадическом пространстве. Обосновывается, что дополнительный учет ограничений на количество серий единичных элементов в двоичных полиадических числах обеспечивает увеличение степени сжатия сообщений произвольного источника информации.

1. Введение

Известные технологии сжатия статических и динамических изображений обеспечивают наибольшие степени сжатия за счет сокращения психовизуальной избыточности и последующего статистического кодирования компонент трансформант ортогональных преобразований. Психовизуальная избыточность сокращается в результате обнуления высокочастотных составляющих компонент трансформант.

Основными недостатками данных технологий являются:

- возможные потери информации, которые возникают на этапе самого преобразования и на этапе квантизации их компонент;
- зависимость эффективности сжатия от характеристик источника информации.

По этим причинам методы указанных технологий нельзя использовать для сжатия данных, полученных от различных источников информации и требующих различной степени достоверности (для некоторых приложений необходимо проводить обработку без внесения погрешностей).

Поэтому требуется разработать кодирование двоичных данных, на которые одновременно наложены ограничения на количество единичных серий и на позиции с запретом появления единичных элементов.

Однако теоретические основы и методы сжатия на основе структурного кодирования в двоичном полиадическом пространстве отсутствуют. Следовательно, целью данного исследования является разработка теоретических основ и методов сжатия данных, полученных от различных источников информации, на основе двухпризнакового представления в двоичном полиадическом пространстве с заданной степенью достоверности.

2. Обоснование возможности дополнительного сокращения структурной избыточности на основе кодирования по количеству единиц в двоичном полиадическом пространстве

Для обоснования того, что за счет выявления закономерностей по количеству серий единиц в полиадическом пространстве осуществляется дополнительное сокращение избыточности, необходимо доказать неравенства:

$$V(m, \Lambda, \vartheta) \leq V(m, \Lambda); \quad (1)$$

$$V(m, \Lambda, \vartheta) \leq V(m, \vartheta), \quad (2)$$

где $V(m, \Lambda)$, $V(m, \vartheta)$, $V(m, \Lambda, \vartheta)$ – множества двоичных последовательностей, удовлетворяющих соответственно ограничениям на позиции единиц, на количество серий единиц и на количество серий единиц в полиадическом пространстве.

Для доказательства неравенства (1) необходимо показать, что выполняется соотношение:

$$\Psi(m, \Lambda, \vartheta_k) \cap \Psi(m, \Lambda, \vartheta_u) = 0, \text{ где } k, u = \overline{0, \vartheta_{\max}}; k \neq u, \quad (3)$$

т.е. обосновать взаимонезависимость множеств двоичных последовательностей в полиадическом пространстве, полученных для разных значений признаков ϑ_k и ϑ_u (где ϑ_k, ϑ_u

– количество серий единиц соответственно для k -й и u -й двоичных последовательностей;
 $\vartheta_{\max}$ – максимальное количество серий единиц, которое может содержаться в двоичной последовательности длиной m элементов, $\vartheta_{\max} = \lfloor \frac{m+1}{4} \rfloor$).

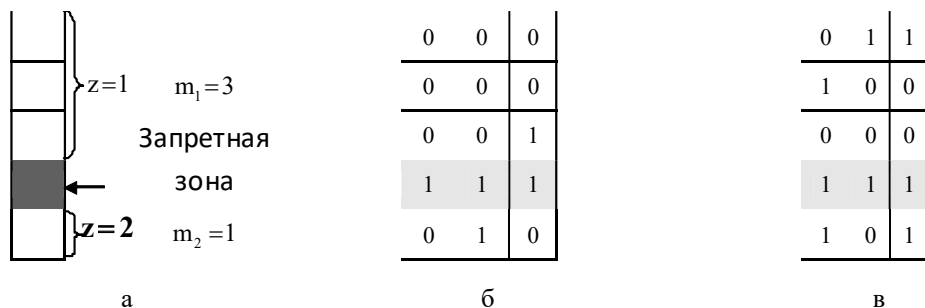
Действительно, соотношение (3) выполняется, поскольку выполняется условие взаимонезависимости для различных однопризнаковых множеств без ограничений на возможные позиции единиц:

$$\Psi(m, \vartheta_k) \cap \Psi(m, \vartheta_u) = \emptyset, \text{ где } k, u = \overline{0, \vartheta_{\max}}; k \neq u. \quad (4)$$

Поскольку условие (4) выполняется для произвольных ограничений на позиции единиц, то оно также будет выполняться в условиях наложения конкретных ограничений (соотношение (3)).

Из взаимонезависимости множеств $\Psi(m, \Lambda, \vartheta_k)$ следует, что они являются слагаемыми множества полиадических чисел $\Psi(m, \Lambda)$, причем знак равенства в выражении (1) будет стоять тогда, когда наложены запреты на появления единиц для всех позиций. Неравенство (1) доказано.

Рассмотрим доказательство неравенства (2). Двоичная последовательность будет принадлежать множеству $\Psi(m, \Lambda, \vartheta)$ тогда, когда через заданную позицию с $\lambda_i = 1$ ($s_i = 0$) не будет проходить серия единиц, т.е. полиадические ограничения трактуются как запрет появления единиц на определенной позиции. Значит, на расположение серий единичных элементов накладываются дополнительные запреты, задаваемые полиадическими ограничениями $0 \leq a_i \leq \lambda_i - 1$. Отсюда следует выполнение неравенства (2). Знак равенства в соотношении (2) будет тогда, когда для всех позиций разрешено появление единичных элементов. Примеры запрещенных двоичных последовательностей показаны на рисунке.



Пример запрещенных комбинаций для $m = 5$, $s_4 = 0$: а – общая схема выделения двух рабочих зон, состоящих соответственно из трех и одного двоичного элемента $m_1 = 3$ и $m_2 = 1$;
б – примеры запрещенных двоичных последовательностей с количеством серий, равным $\vartheta = 1$;
в – примеры запрещенных двоичных последовательностей с количеством серий, равным $\vartheta = 2$

3. Разработка кодирования двоичных полиадических чисел по ограниченному количеству серий единиц

Для определения объема множества $V(m, \Lambda, \vartheta)$ докажем следующую теорему.

Теорема об объеме множества двоичных полиадических чисел по количеству серий единиц. Количество двоичных последовательностей равно

$$V(m, \Lambda, \vartheta) = \sum_{k=1}^K V(\Theta^{(k)}) = \sum_{k=1}^K \prod_{z=1}^Z V(\vartheta_z^{(k)}, \Theta^{(k)}), \quad (5)$$

$$V(\vartheta_z^{(k)}, \Theta^{(k)}) = \binom{m_z + 1}{2\vartheta_z^{(k)}} = \frac{(m_z + 1)!}{(2\vartheta_z^{(k)})! (m_z + 1 - 2\vartheta_z^{(k)})!}, \quad (6)$$

где $\vartheta_z^{(k)}$ – значение числа серий для z -й допустимой зоны двоичной последовательности A (рисунок, поз. а); $\Theta^{(k)}$ – вектор, элементами которого является k -я комбинация количеств серий единиц $\vartheta_z^{(k)}$ в допустимых зонах $\Theta^{(k)} = \{\vartheta_1^{(k)}, \dots, \vartheta_z^{(k)}, \dots, \vartheta_Z^{(k)}\}$, $k = \overline{1, K}$; Z – количество допустимых зон в двоичной последовательности; K – количество векторов $\Theta^{(k)}$ (количество комбинаций длиной Z , составленных из элементов $\vartheta_z^{(k)}$); m_z – количество двоичных элементов в z -й допустимой зоне; $V(\vartheta_z^{(k)}, \Theta^{(k)})$ – количество допустимых двоичных последовательностей, полученных для z -й допустимой зоны по количеству серий единиц, равному $\vartheta_z^{(k)}$ для вектора $\Theta^{(k)}$; $V(\Theta^{(k)})$ – количество допустимых двоичных последовательностей, полученное с учетом обработки всех Z допустимых зон для k -го вектора значений величин $\vartheta_z^{(k)}$.

Доказательство. Система полиадических ограничений делит исходную двоичную последовательность на допустимые и запрещенные зоны (рисунок, поз. а). Запрещенные зоны состоят из элементов, на позициях которых запрещено появление единицы, т.е. $\lambda_i = 1$. Допустимые зоны располагаются между запрещенными зонами и на их позициях допустимо появление единиц. Обозначим число допустимых зон через Z , $0 \leq Z \leq \lfloor \frac{m+1}{2} \rfloor$, причем

$Z=0$, когда все элементы двоичной последовательности равны 0. Пример множества $V(m, \Lambda, \vartheta)$ двоичных полиадических чисел по количеству серий единиц для $m=5$, $\Lambda = \{2; 2; 2; 1; 2\}$ и $\vartheta = 1$ приведен в таблице. Для данного примера количество допустимых зон равно $Z=2$, а запрещенная зона состоит из одного элемента $a_4 = 0$.

За счет деструктуризации исходной последовательности на запрещенные и допустимые зоны исходное количество серий ϑ будет равняться сумме количеств серий единиц $\vartheta_z^{(k)}$ каждой допустимой зоны z :

$$\vartheta = \sum_{z=1}^Z \vartheta_z^{(k)}. \quad (7)$$

Множество $V(m, \Lambda, \vartheta)$ двоичных полиадических чисел с $m=5$, $\Lambda = \{2; 2; 2; 1; 2\}$ по числу серий $\vartheta = 1$

a_1	0	0	0	0	1	1	1
a_2	0	0	1	1	0	1	1
a_3	0	1	0	1	0	0	1
a_4	0	0	0	0	0	0	0
a_5	1	0	0	0	0	0	0
$N(m, \Lambda, \vartheta)$	0	1	2	3	4	5	6

Рассмотрим пример формирования кода-номера для двоичной последовательности $A^{(j)} = (1; 0; 0; 0; 0; 0; 0; 0; 1; 0; 1; 0)$, у которой пятый, шестой и седьмой элементы являются запрещенными, т.е. $\lambda_5 = 1$, $\lambda_6 = 1$ и $\lambda_7 = 1$. В этом случае исходная последовательность разбивается на две подпоследовательности, включающие в себя две допустимые зоны $Z=2$, $m_1=4$ и $m_2=5$: $A_1^{(j)} = (1; 0; 0; 0)$ и $A_2^{(j)} = (0; 1; 0; 1; 0)$. Количество серий единиц в последовательности $A^{(j)}$ равно $\vartheta=3$. Возможны четыре комбинации векторов $\Theta^{(\xi)}$, $K=4$: для $\xi=1$ $\vartheta_1^{(1)}=0$, $\vartheta_2^{(1)}=3$; для $\xi=2$ $\vartheta_1^{(2)}=1$, $\vartheta_2^{(2)}=2$; для $\xi=3$ $\vartheta_1^{(3)}=2$, $\vartheta_2^{(3)}=1$; для $\xi=4$ $\vartheta_1^{(4)}=3$, $\vartheta_2^{(4)}=0$. При этом поскольку длина первой зоны равна 4, то максимальное количество серий в первой зоне не должно превышать $\vartheta_{\max,1} = \left\lfloor \frac{m_1+1}{2} \right\rfloor = 2$. Значит, комбинация $\xi=4$ является запрещенной.

4. Выводы

Разработаны теоретические основы компактного представления двоичных данных на основе структурного кодирования по числу серий единиц в двоичном полиадическом пространстве, включающие в себя:

- формулировку основных понятий представления двоичных данных с ограниченным количеством серий единиц в двоичном полиадическом пространстве;
- доказательство теоремы о количестве допустимых двоичных полиадических чисел с ограниченным числом серий единиц, удовлетворяющих одновременно ограничениям на число серий единиц и на позиции с допустимым появлением единичных элементов. Это позволяет получить значение объема допустимого множества для заданных значений количества серий единиц и ограничений на позиции с возможным появлением единиц;
- систему правил, позволяющую сформировать код-номер для двоичного полиадического числа по заданному значению числа серий единиц и по заданным ограничениям на позиции с возможным появлением единиц (количеству допустимых зон и их длине).

Обосновано, что для повышения степени компактного представления двоичных данных с заданной степенью достоверности необходимо решить научную проблему, которая состоит в разработке теоретических основ и методов сжатия данных, полученных от различных источников информации, на основе двухпризнакового представления в двоичном полиадическом пространстве с заданной степенью достоверности.

Список литературы: 1. Комарова Л.О. Методи управління інформаційно-комунікаційними кластерами в кризових ситуаціях. Монографія / Л.О.Комарова // К.:ДУТ. 2014. 395 с. 2. Автоматизированная система коммерческого осмотра поездов и вагонов / Под ред. В.Н. Солошенко. М.: ГОУ „Учебно-методический центр по образованию на железнодорожном транспорте”, 2008. 3. Ватолин Д., Ратушняк А., Смирнов М., Юкин В. Методы сжатия данных. Устройство архиваторов, сжатие изображений и видео М.: Диалог-Мифи, 2003. 381с. 4. Баранник В.В. Кодирование трансформированных изображений в инфокоммуникационных системах / В.В. Баранник, В.П. Поляков. Х.: ХУПС, 2010. 212 с. 5. Баранник В.В. Модель оценки информативности слота Р-кадров на основе выявления структурно-градиентных межтрансформантных ограничений / В.В. Баранник, С.С. Шульгин // АСУ и приборы автоматики. 2015. №172. С. 76-81.

Поступила в редколлегию 25.12.2015

Баранник Владимир Викторович, д-р техн. наук, профессор, начальник кафедры Харьковского университета Воздушных Сил имени Ивана Кожедуба. Научные интересы: системы, технологии преобразования, кодирования, защиты и передачи информации, семантической обработки изображений. Адрес: Украина, 61023, Харьков, ул. Сумская, 77/79, тел. 8 050-3038971.

Хаханова Анна Владимировна, канд. техн. наук, доцент, докторант кафедры АПВТ ХНУРЭ. Адрес: Украина, 61166, Харьков, пр. Науки, 14.

Сидченко Сергей Александрович, канд. техн. наук, старший научный сотрудник научного центра Харьковского университета Воздушных Сил. Адрес: Украина, 61000, Харьков, ул. Сумская, 77/79, тел. 066-299-82-73.

**ТЕХНОЛОГИЯ ЛОКАЛИЗАЦИИ ОШИБКИ ПУТЕМ
ПОЗИЦИОНИРОВАНИЯ КОДА ПЕРЕМЕННОЙ ДЛИНЫ**

Обосновывается проблематичность обеспечения безопасности видеoinформационного ресурса в системах аэромониторинга и видеоконференцсвязи. Формируются основные недостатки применения статистического кодирования при обработке видео. Обосновывается необходимость позиционирования кодов переменной длины в видеопотоке. Разрабатывается технология распределения кодов в существующем методе. В результате этого формируется механизм локализации действия битовой ошибки на видеопоток. Рассматривается особенность влияния ошибки на компоненты трансформанты, которые относятся к разным частотным областям.

1. Введение

Стремительное развитие информационных технологий создало возможность использовать видеoinформационный ресурс при однонаправленной передаче в системах видеонаблюдения и двухсторонний обмен в системах видеосвязи. Эти системы широко распространены в государственных ведомственных учреждениях, особенно в Министерстве обороны Украины. Этому способствовало улучшение качества системы управления при использовании видеoinформационного ресурса [1], что в свою очередь вызвало необходимость защиты видеoinформационного ресурса, которая обусловлена важностью его содержания. Данная необходимость подтверждается ростом кибернетических угроз государственным учреждениям, возникающих в результате кибератак [2]. Чаще всего применяются атаки типа DDoS-атака [3], что обусловлено простотой осуществления данного вида атак и эффективностью использования. Внешнее проявление применения кибератак выражается в искажении, пропадании и полной недоступности видеoinформационного ресурса. При этом современные методы предотвращения киберугроз [4] реагируют при соответствии параметров сетевой активности. Результатом этого является недостаточная оперативность существующих методов защиты, что приводит к потере данных. Это вызывает необходимость улучшения информационной безопасности видеoinформационного ресурса путем разработки технологии кодирования с учетом воздействия кибератак на ресурс. Актуальность работы связана с важностью видеoinформационного ресурса и необходимостью увеличения его устойчивости к ошибкам. Для обоснования варианта решения поставленной задачи необходимо рассмотреть существующие технологии обработки видеoinформационного ресурса. Данные технологии (JPEG, MPEG-4, H.264) используют сжатие на базе статистического кодирования (кодов переменной длины) [5]. Особенностью данной технологии является использование корреляции между кадрами в группе кадров. Данная группа кадров состоит из опорного кадра и последовательности разностных и обратных кадров. При этом длительность опорного кадра видео значительно меньше длительности всей группы кадров. Применение опорных кадров в процессе обработки обуславливает высокую степень сжатия потока и в то же время создает уязвимость целостности потока к искажениям [6]. Результатом искажения небольшого участка опорного кадра при применении указанных кибератак является потеря содержимого всей группы кадров для потока видеoinформационного ресурса. Это приводит к нарушению таких характеристик информационной безопасности ресурса, как целостность и доступность. Так как для существующих систем киберзащиты характерно некоторое запаздывание реакции, то при наличии краткосрочных атак (медленная DDoS-атака) возможно отсутствие идентификации угрозы при фактической недоступности видеoinформационного ресурса. Для динамических условий использования видеoinформационного ресурса это считается недопустимым, так как приводит к проблеме недостаточной оперативности противодействия киберугрозам для существующих методов защиты. Для локализации действия кибератак предлагается разработать соответствующий метод обработки видеoinформационного ресурса. Поэтому возник-

кает необходимость рассмотреть недостатки современных технологий кодирования видеоданных. В существующей схеме обработки видеoinформационного ресурса применяется статистическое кодирование компонент линейаризованной трансформанты, которая образовалась после дискретно-косинусного преобразования сегментов исходного изображения. В силу префиксности статистического кода переменной длины требование к использованию символов-разделителей в процессе декодирования компонент трансформанты изображения будет отсутствовать. Из этого вытекает условие последовательности осуществления декодирования каждого кода. Последовательность осуществления декодирования обуславливает присутствие уязвимости к наличию ошибки в разряде кода. Результатом битовой ошибки в коде является неверная идентификация кода и его длины, что приводит к дальнейшей неверной идентификации всех последующих VLC кодов, которая выражается в разрушении всего изображения.

Цель работы – создать технологию локализации ошибки при позиционировании статистических кодов в потоке и рассмотреть изменение устойчивости видеoinформационного ресурса к ошибкам для предложенной технологии.

2. Основной материал

Для устранения уязвимости, связанной с неверным позиционированием кода переменной длины, предлагается применять перераспределение данных кодов в кодовые слова равномерной длины. Это достигается с помощью технологии упругого к ошибкам энтропийного кодирования (Error Resilient Entropy Code – EREC). Данная технология задается функцией преобразования f_{errec} согласно соотношению:

$$L(\theta) \xrightarrow{f_{\text{errec}}} S(\Lambda). \quad (1)$$

Здесь f_{errec} – функция распределения VLC кодов ℓ_ξ по слотам s_λ ; Λ – количество слотов, в которые распределены VLC коды.

Технология EREC характеризуется следующими этапами: размещение содержимого кодовых конструкций ℓ_ξ ; перераспределение содержимого кодовых конструкций ℓ_ξ . Данная технология размещает VLC коды ℓ_ξ в слоты пакета EREC $S(\Lambda)$ на основе перестройки битовой структуры. Здесь учитывается, что слотами являются кодовые слова равномерной длины, т.е.

$$|s_i|_2 = |s_j|_2 = \upsilon, \text{ при } i \neq j. \quad (2)$$

В результате преобразования последовательности $L(\theta)$ VLC кодов образуется пакет $S(\Lambda)$ слотов. Вектор слотов (пакет) записывается следующим образом:

$$S(\Lambda) = \{s_1; \dots; s_\lambda; \dots; s_\Lambda\}.$$

Здесь s_1 – начальный слот в пакете; s_Λ – последний слот в пакете. Схематическое распределение VLC кодов ℓ_ξ по слотам s_ξ , т.е: $\ell_\xi \rightarrow s_\xi^{(1)}$, в соответствии с технологией EREC, показано на рис. 1. В данной формуле $s_\xi^{(1)}$ – слот, сформированный при размещении кода ℓ_ξ ; (1) – индекс начального этапа, соответствующий первичному размещению VLC кодов по слотам. На данном рисунке представлено распределение кодового потока $L(\theta)$, состоящего из $\Lambda = 19$ VLC кодов ℓ_ξ по 19 слотам. Различные VLC коды ℓ_ξ представлены схематично в виде разных прямоугольников.

На первичном этапе рассматривается размещение VLC кодов ℓ_ξ по слотам s_ξ без учета их выравнивания по длине υ . Порядок заполнения слотов s_ξ определяется следующими правилами:

1) установление порядка расстановки VLC кодов по слотам пакета. Согласно этому правилу 1-я кодовая последовательность ℓ_1 размещается в слоте s_1 (как показано на рис. 1);

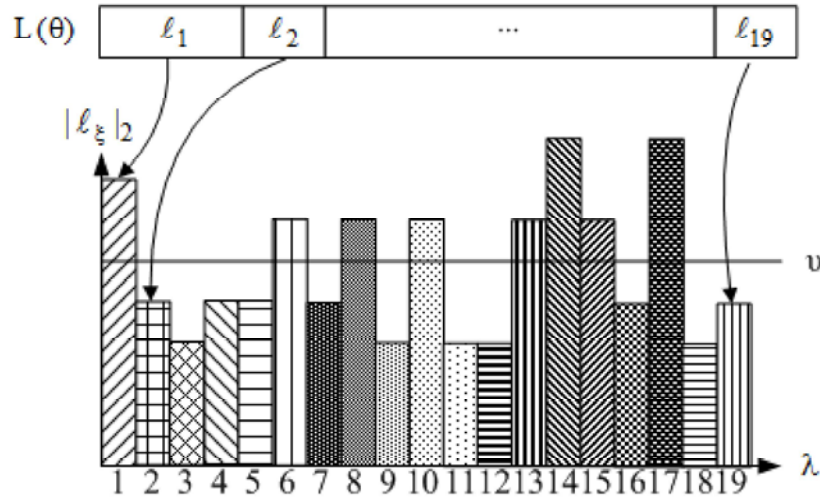


Рис. 1. Первичный этап размещения битовых составляющих кодовых конструкций при формировании пакета слотов

2) установка порядка заполнения слотов разрядами VLC кодов. Согласно этому правилу старшие разряды $q_{\xi,\gamma}; \gamma \rightarrow 1$ кодовой конструкции ℓ_ξ размещаются в нижних битовых позициях слота s_ξ . Наоборот, младшие разряды $q_{\xi,\gamma}; \gamma \rightarrow |\ell_\xi|_2$ кодовой конструкции ℓ_ξ размещаются в верхних битовых позициях слота s_ξ .

Следующим этапом формирования пакета слота является выравнивание длин кодовых слов s_ξ для выполнения условия (2) в случае, когда известно количество Λ слотов в пакете. Для этого сначала требуется определить длину v слота. Она определяется как отношение суммарной длины $|L(\theta)|_2$ последовательности $L(\theta)$ кодов к количеству Λ слотов. Формула расчета длины указана в следующем выражении:

$$v = \frac{|L(\theta)|_2}{\Lambda} = \left\lceil \frac{1}{\Lambda} \sum_{i=1}^{\theta} |\ell_i|_2 \right\rceil. \quad (3)$$

Здесь $\lceil x \rceil$ – оператор округления значения x до большего натурального числа.

Для повышения эффективности заполнения слотов в соответствии с технологией EREC происходит перераспределение избыточных составляющих $\Delta\ell_\xi$ VLC кодов ℓ_ξ по избыточным составляющим $\Delta s_i, i \neq \xi$ слотов $s_i, i \neq \xi$, что задается формулой:

$$\Delta L(\theta) \xrightarrow{f_p} \Delta S(\Lambda), \quad (4)$$

где $\Delta L(\theta) = \{\Delta\ell_1; \dots; \Delta\ell_\xi; \dots; \Delta\ell_\theta\}$ – совокупность избыточных составляющих $\Delta\ell_\xi$ VLC кодов ℓ_ξ ; $\Delta S(\Lambda) = \{\Delta s_1; \dots; \Delta s_\xi; \dots; \Delta s_\Lambda\}$ – совокупность избыточных составляющих Δs_ξ слота s_ξ .

Это позволит: 1) сохранить информацию VLC кодов ℓ_ξ на основе заполнения пустот слотов s_ξ ; 2) сократить избыточность, вызванную недогрузкой слота. В общем случае избыточная составляющая $\Delta\ell_\xi$ VLC кода ℓ_ξ будет размещена по нескольким слотам s_i (рис. 2).

В процессе декодирования при поразрядной сборке VLC кодов ℓ_ξ происходит обращение к содержимому избыточных составляющих Δs_i других слотов $s_i, i \neq \xi$. Это обращение производится только после идентификации других VLC кодов ℓ_i .

При этом внесения двоичных разрядов $h_{\xi+d_g,\mu}$ избыточных составляющих $\Delta s_{\xi+d_g}^{(\gamma)}$ слота $s_{\xi+d_g}$ в содержимое части кода $\partial\ell_\xi$ продолжают до идентификации окончания ξ -й кодовой комбинации ℓ_ξ . Следует заметить, что каждый этап сборки VLC кодов ℓ_ξ должен происходить после завершения предыдущего на позициях всех слотов s_i . Результат формирования кода ℓ_ξ при обращении к битовым составляющим нескольких слотов представлен на рис. 3.

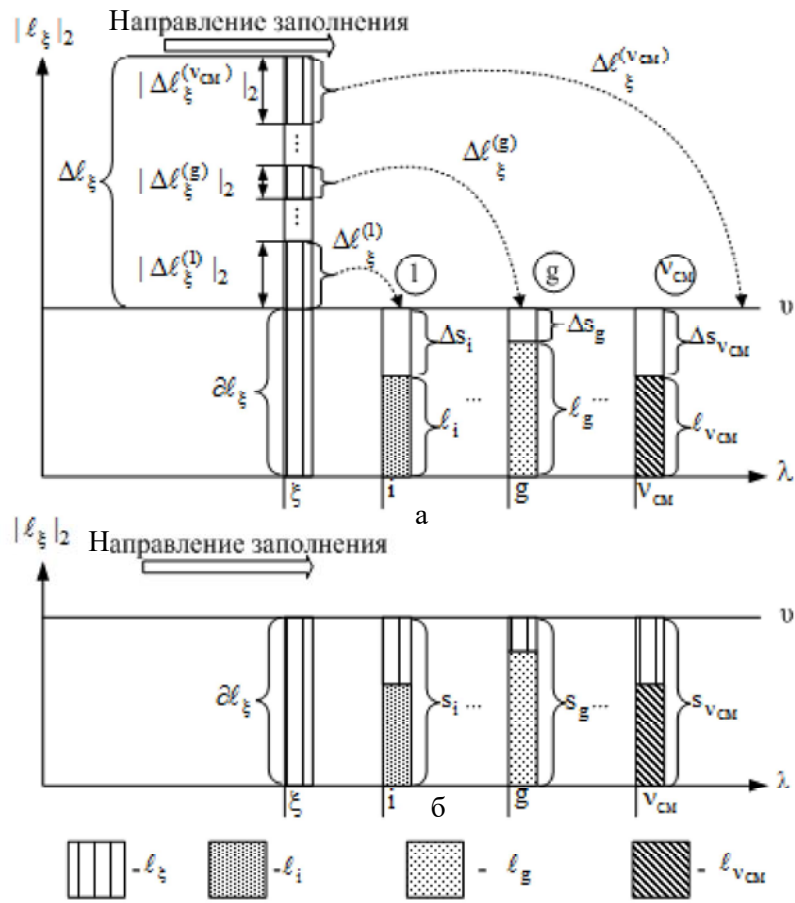


Рис. 2. Структурная схема многослотового распределения VLC: а – до распределения; б – после распределения

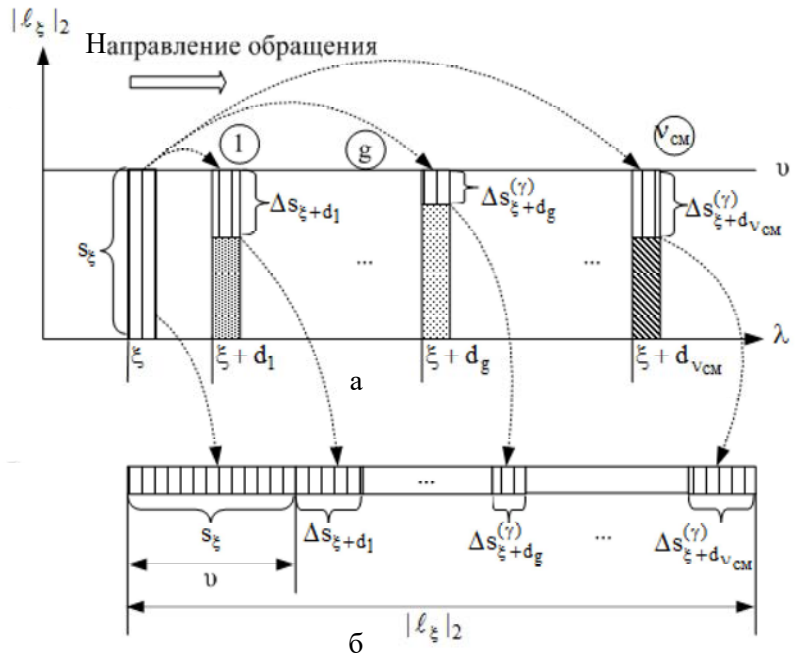


Рис. 3. Сборка VLC кода ℓ_ξ по подпоследовательности $\Delta S(v_{cm})$ до идентификации окончания кодовой комбинации ℓ_ξ : а – изъятие битовых составляющих других слотов $S_{\xi+d_g}$; б – результат формирования кода ℓ_ξ

Очевидно, что в силу последовательности этапов обращения к содержимому слотов, при идентификации λ -й кодовой комбинации ℓ_λ ошибка затрагивает избыточную составляющую $\Delta s_\lambda^{(\gamma)}$ слота s_λ , к которой идет обращение при сборке на g -м этапе ξ -го кода ℓ_ξ , $\xi = \lambda - d_g$. При этом также возможна неверная идентификация окончания ξ -й кодовой комбинации ℓ'_ξ . Идентификация кода ℓ_λ и ℓ_i кодов ℓ_i , избыточные составляющие которых входят в избыточные составляющие $\Delta s_\lambda^{(j)}$, $j < \gamma$ слота s_λ , будет произведена верно. В связи с особенностью построения статистического кода, которая заключается в обратной пропорциональности длины кода частоте его появления, применение технологии перераспределения защищает от битовой ошибки ВЧ компоненты трансформанты [7]. В то же время сохраняется лавинный эффект ошибки для НЧ компонент изображения. Для устранения лавинного эффекта предлагается использовать изменяющуюся избыточность распределения с механизмом обратной связи между кодером и декодером.

Выводы

Разработана технология локализации ошибки путем позиционирования кода переменной длины при незначительном увеличении длины пакета, который необходим для кодирования компонент. Создана технология, которая учитывает применение кибератак на видеoinформационный ресурс. Для локализации ошибки декодирования для НЧ компонент впервые предложено использовать динамическую избыточность технологии распределения с механизмом обратной связи между кодером и декодером.

Список литературы: 1. *Анализ действия кибератак на видеoinформационный ресурс в информационно-телекоммуникационных сетях* / В.В. Баранник, С.А. Подлесный // АСУ и приборы автоматики. 2014. Вып. 164. С. 16-22. 2. *Захист інформаційних мереж є питанням державної безпеки - голова Держспецзв'язку Геннадій Резніков* [Електронний ресурс] http://www.dstszi.gov.ua/dstszi/control/uk/publish/article?art_id=104662&cat_id=38712, 2012. 3. *Звіт CERT-UA за 2014 рік* [Електронний ресурс]. <http://cert.gov.ua/?p=2019>, 2015. 4. *Мартынчук И.* Материалы технического тренинга «Построение безопасных сетей на оборудовании D-Link», [Електронний ресурс], <http://service.d-link.ua/sites/default/files/files/Security.zip>, Киев, 2012. 190 с. 5. *Richardson E.* "H.264 and MPEG-4 video compression". Chichester, UK: Wiley and Sons, 2003. 306 p. 6. *Wang Y. and Zhu Q.F.* Error control and concealment for video communication: A review. Proceedings of the IEEE, 1998. Vol. 86, no. 5. P. 974-997. 7. *Обґрунтування підходу щодо створення технології кіберзахисту відеоінформаційного ресурсу в інфокомунікаційному просторі дії кібератаки на відеоінформаційний ресурс в інформаційно-телекомунікаційних мережах* / В.В. Баранник, С.А. Підлісний // Радіoeлектроніка та інформатика. 2015. №3. С.62-66.

Поступила в редколлегию 12.01.2016

Баранник Владимир Викторович, д-р техн. наук, профессор, начальник кафедры Харьковского университета Воздушных Сил им. И. Кожедуба. Научные интересы: кодирование и защита информации для передачи в телекоммуникационных системах. Адрес: Украина, 61023, Харьков, ул. Сумская, 77/79. E-mail: barannik_v_v@mail.ru.

Подлесный Сергей Анатольевич, начальник отделения Харьковского университета Воздушных Сил им. И. Кожедуба. Научные интересы: кодирование и защита информации для передачи в телекоммуникационных системах. Адрес: Украина, 61023, Харьков, ул. Сумская 77/79. E-mail: serg380638472732@gmail.com.

Туренко Сергей Викторович, преподаватель ХНУРЭ. Научные интересы: кодирование и защита информации для передачи в телекоммуникационных системах. Адрес: Украина, 61000, Харьков, ул. Сумская 77/79. E-mail: barannik_v_v@mail.ru.

ЭЛЕМЕНТАРНАЯ ТЕОРИЯ ТЯГОТЕНИЯ

Рассматривается механизм тяготения. Отмечается целесообразность проведения экспериментальной проверки предложенной гипотезы о порождении тяготения “тормозным” излучением.

1. Введение

Человек живет в мире, где всё является проявлением электромагнитной материи (ЭММ). Созданные им устройства, естественно, реагируют на проявления определённых свойств ЭММ. Реакция устройств на другие виды материи не известна. Можно предположить, что тяготение представляет собой одно из проявлений ЭММ. Дальнейшее изложение элементарной теории тяготения основывается на понятиях теории электричества.

Поскольку атомы “образованы” из элементарных частиц, то логично рассматривать тяготение в системе, состоящей из двух электронов. Такое уплотнение ЭММ как электрон в свободном состоянии (или в составе атома) может существовать долго, что позволяет рассматривать его в виде авторегулирующейся системы с астigmatизмом первого порядка по количеству ЭММ (массе). На основе этого электрон можно представить в виде пульсирующего сгустка ЭММ. Частота пульсаций, сопровождающая процесс авторегулирования (из условий устойчивости), должна быть существенно меньше резонансной частоты электрона. Другими словами, “размер” электрона существенно меньше длины волны пульсаций. Пульсации массы электрона сопровождаются излучением ЭММ в течение части периода пульсаций и поглощением ЭММ из окружающей среды в течение другой части периода пульсаций, что компенсирует потери на излучение.

В 1900 г. П.Н. Лебедев измерил давление “света”, что следовало бы воспринимать как невозможность осуществления тяготения при распространении квантов гравитационного поля со скоростью “света”. При скорости гравитационного воздействия $v_{гп} \gg c$ кванты гравитационного поля “слабо” взаимодействуют с облучаемым объектом, скорость протекания электромагнитных процессов в котором не превышает скорости “света”. Всяма незначительная часть воздействующего излучения, которое имеет высокую проникающую способность, участвует во взаимодействии. Эта часть излучения обладает массой, движущейся внутри объекта со скоростью “света”, и, вследствие торможения, излучает кванты ЭММ по направлению воздействующего излучения, сообщая при этом силовое воздействие объекту по направлению к источнику излучения. Таким образом, тяготение порождается “тормозным” излучением.

В основе излагаемого материала лежит закон Ньютона, определяющий силу гравитационного взаимодействия двух тел с массами m_1 и m_2 [1], который запишем таким образом:

$$F_H = k\gamma \frac{m_1 \cdot m_2}{R^2},$$

где $[k] = \text{см}^3 \text{г}^{-1} \text{с}^{-2} = 10^{-3} \text{м}^3 \text{кг}^{-1} \text{с}^{-2}$, коэффициент размерности; R – расстояние между тяготеющими массами; $\gamma = 6.67259 \cdot 10^{-8}$ – безразмерный коэффициент, характеризующий интенсивность взаимодействия тяготеющих масс.

2. Электромагнитная природа гравитационного взаимодействия

Электрон, порожденный турбулентностью движения непрерывной электромагнитной материи, в первом приближении представляется объемным вихрем с однонаправленным

вращением ЭММ внутри ограниченного объема. Масса и размер такого вихревого образования стабилизируется за счет внутренней энергии электрона, которая пополняется посредством обмена энергией с ЭММ окружающей среды. Процесс стабилизации сопровождается пульсацией массы (объема) и излучением электромагнитной энергии в радиальных направлениях. Далее неподвижный электрон рассматривается как элементарный изотропный сферический излучатель. Энергия излучения каждого из статически взаимодействующих электронов за секунду определяется в соответствии с гипотезами Планка и Луи де Бройля выражением:

$$E_{\text{изл.}} = k_1 \cdot h \cdot f_n = m_e c^2, \quad (1)$$

где $h = 6.6260755 \cdot 10^{-27}$ – постоянная Планка, $[k_1] = \text{см}^2 \cdot \text{г} \cdot \text{с}^{-1} = 10^{-7} \cdot \text{м}^2 \cdot \text{кг} \cdot \text{с}^{-1}$; f_n – частота пульсаций массы m_e электрона, с^{-1} ; m_e – масса электрона, c – скорость распространения непрерывной части ЭММ и движения ЭММ внутри электрона.

Из (1) следует, что частота пульсаций (и излучений) f_n и соответствующая ей длина волны пульсаций массы электрона λ_n определяются следующим образом:

$$f_n = \frac{m_e c^2}{k_1 h} \cong 1.23559 \cdot 10^{20} \text{сек}^{-1} \left(\lambda_n = \frac{k_1 h}{m_e c} = \frac{c}{f_n} \cong 2.42631 \cdot 10^{-10} \text{см} \right). \quad (2)$$

Плотность потока энергии воздействия, излучаемой в область локализации другого электрона ЭММ, равна

$$E_{\text{вз.1}} = \frac{m_e c^2}{4\pi \cdot R^2}, \text{ при } R = 1 \text{ см } E_{\text{вз.1}} \cong \frac{m_e c^2}{4\pi}. \quad (3)$$

Представим электрон в виде системы автоматического регулирования (стабилизации) его параметров – массы, объема и, в соответствии с выражением (1), его частоты пульсаций. В высокоточной системе авторегулирования запаздывание стабилизирующего воздействия $t_{\text{зап.}}$ как отклика на возникающее отклонение стабилизируемого параметра должно быть меньше характерного времени нестационарности – в рассматриваемом случае периода пульсаций с указанной частотой f_n . Таким образом, необходимо, чтобы выполнялось условие: $t_{\text{зап.}} \cdot f_n \ll 1$.

Учитывая, что запаздывание определяется размером электрона (обозначим некий эффективный диаметр электрона $d_{\text{эф.е}}$) и длина волны пульсаций связана с периодом пульсаций соотношением $T_n = \lambda_n / c$, получаем $t_{\text{зап.}} / T_n = d_{\text{эф.е}} / \lambda_n \ll 1$, где $t_{\text{зап.}} = d_{\text{эф.е}} / c$.

По аналогии с выражением для полной мощности, излучаемой диполем Герца, примем, что для величины массы излучаемой ЭММ допустимо выражение:

$$M_{\text{изл.}} = m_e \frac{\pi}{3} \cdot \left(\frac{d_{\text{эф.е}}}{\lambda_n} \right)^2.$$

В системе, состоящей из двух электронов, расположенных на расстоянии $R = 1 \text{ см}$ друг от друга, плотность потока массы излучения одного из электронов в области локализации другого равна :

$$M'_{\text{изл.}} = \frac{M_{\text{изл.}}}{4\pi R^2}, \text{ при } R = 1 \text{ см } M'_{\text{изл.}} = \frac{M_{\text{изл.}}}{4\pi} = \frac{1}{12} \left(\frac{d_{\text{эф.е}}}{\lambda_n} \right)^2 \cdot m_e.$$

Плотность потока массы ЭММ, излучаемой двумя электронами и обеспечивающей взаимодействие, равна:

$$2M'_{\text{изл.}} = \left(d_{\text{эф.е}} / \lambda_n \right)^2 \cdot m_e / 6.$$

Поскольку коэффициенты $(d_{\text{эф.е}}/\lambda_n)^2/6$ и γ в законе всемирного тяготения характеризуют интенсивность взаимодействия, то можно предположить следующее равенство:

$$\gamma = \frac{1}{6} \left(\frac{d_{\text{эф.е}}}{\lambda_n} \right)^2. \quad (4)$$

Отсюда $d_{\text{эф.е}} = 1,5352 \cdot 10^{-13} \text{ см}$. Тогда $M'_{\text{изл.}} = 0.5 \cdot \gamma m_e$, что соответствует закону всемирного тяготения. Запишем следующим образом (положив $R = 1 \text{ см}$):

$$F_H = k' \cdot \frac{\gamma}{2} \cdot m_{e1} \cdot m_{e2} + k' \frac{\gamma}{2} \cdot m_{e2} \cdot m_{e1}, \quad (5)$$

где $m_{e1} = m_{e2} = m_e$ – массы “гравитирующих” электронов, $k' = k \cdot \text{см}^{-2}$.

Из приведенной записи (5) закона всемирного тяготения можно предположить, что гравитационное воздействие каждого из электронов друг на друга происходит за счет “излучения” массы, равной $0.5\gamma \cdot m_e$, испытывающей тяготение со стороны противоположного электрона. Суммарная же масса, расходуемая на взаимодействие $m_{\text{сум.}} = \gamma \cdot m_e$, характеризуется энергией в соответствии с (3), равной:

$$E_{\text{вз.}} = m_e c^2 / 2\pi.$$

Исходя из изложенного выше, можно записать:

$$\gamma \cdot m_e \cdot v_{\text{гр.}}^2 = \frac{1}{2\pi} m_e c^2.$$

Величина скорости распространения квантов гравитационного поля определяется выражением:

$$v_{\text{гр.}} = c / \sqrt{2\pi\gamma} \cong 4.63 \cdot 10^{11} \text{ м/с}. \quad (6)$$

При достижении максимального объёма масса электрона прекращает движение в радиальных направлениях и только незначительная её часть (масса кванта), освободившись от воздействия остальной массы, получает ускорение, в результате чего квант движется со скоростью $v_{\text{гр.}}$. Это результат “тормозного” излучения. Далее, ЭММ, излучаемая первым электроном с эквивалентной массой $0.5 \cdot \gamma m_e$, воздействует на второй электрон. Используя концепцию диполя Герца, можно записать для воспринимаемой вторым электроном ЭММ с массой M' , движущейся со скоростью “света” внутри второго электрона [4]:

$$M' = (0.5 \cdot \gamma m_e) \cdot \frac{1}{2} \frac{1}{6} \left(\frac{d_{\text{эф.е}}}{\lambda_{\text{гр.}}} \right)^2, \quad (7)$$

где $\lambda_{\text{гр.}} = v_{\text{гр.}} / f_n = \lambda_n (v_{\text{гр.}} / c)$ – длина волны гравитационного излучения.

Выражение (7) запишем в следующем образом :

$$M' = (0.5 \cdot \gamma m_e) \cdot \frac{\gamma}{2} \frac{c^2}{v_{\text{гр.}}^2} = \frac{\pi \cdot \gamma^3}{2} \cdot m_e, \quad (7')$$

Численно длина волны гравитационного излучения равна $\lambda_{\text{гр.}} = 3,7472 \cdot 10^{-7} \text{ см}$.

Таким образом, сквозь противоположный электрон без взаимодействия проходит излучение с массой

$$M'' = \frac{1}{2} \gamma m_e - \frac{\pi \cdot \gamma^3}{2} \cdot m_e = \frac{1}{2} \gamma m_e (1 - \pi \gamma^2).$$

В течение одной секунды электрон излучает кванты с массой, равной $4\pi \cdot (\gamma/2) \cdot m_e = 2\pi\gamma m_e$. По аналогии, воспринимаемая масса, равная $0.5 \cdot \pi \cdot \gamma^3 m_e$ (дополнительно к массе квантов, излучаемых вторым электроном), “излучает” (по направлению воздействующего излучения) ЭММ с массой, равной

$$M = 2\pi\gamma \cdot \frac{\pi \cdot \gamma^3}{2} \cdot m_e = \pi^2 \gamma^4 m_e,$$

что объясняется возникновением “тормозного” излучения, вызванного потерей скорости (движения) массой M' . Полная энергия массы M , распространяющейся со скоростью гравитационного взаимодействия, равна $E = M \cdot v_{\text{гр.}}^2$ или, расписывая, получим:

$$E = 2\pi\gamma \cdot \frac{\pi \cdot \gamma^3}{2} \cdot m_e \cdot \frac{c^2}{2\pi\gamma} = \frac{\pi \cdot \gamma^3}{2} m_e c^2.$$

Под действием этой энергии возникает силовое воздействие на второй электрон по направлению к первому: “притяжение”. Количество движения, сообщаемое второму электрону при излучении им массы M , равно: $K = M \cdot v_{\text{ад.}}$

Силовое воздействие, сообщаемое второму электрону первым, направленное в сторону первого электрона, равно:

$$F = \frac{dK}{dt} = v_{\text{гр.}} \cdot \frac{dM}{dt} = k' \cdot \frac{\gamma}{2} \cdot m_e^2,$$

что следует из закона тяготения при $R=1$ см. Производная dM/dt “используется” как в уравнении Мещерского [2].

В течение части периода пульсаций происходит увеличение объема электрона с излучением кванта ЭММ. В течение второй части периода происходит уменьшение объема до минимального значения с поглощением ЭММ из окружающей среды, т.е. имеет место компенсация потерь массы. За период пульсаций излучается “квант” с массой $M \cdot |f_n|^{-1}$, которая пропорциональна производной dM/dt . Представляется допустимым выражение (при $R=1$ см) вида:

$$F = v_{\text{гр.}} \cdot \frac{dM}{dt} = \frac{n}{c} v_{\text{гр.}} \cdot \frac{M}{|f_n|} = k' \cdot \frac{\gamma}{2} \cdot m_e^2,$$

где n – безразмерный коэффициент; c – секунда.

Коэффициент “ n ” характеризует часть периода пульсаций массы электрона, при которой происходит “тормозное” излучение. Из приведенного выражения следует, что $n \cong 0.41454$. Таким образом, длительность излучения электроном кванта ЭММ $\tau_{\text{кв.}} = 0.41454 \cdot T_{\text{п}}$ или $\tau_{\text{кв.}} = 0.41454 \cdot f_{\text{п}}^{-1}$, $n = \tau_{\text{кв.}} / T_{\text{п}}$.

Далее, электрон за одну секунду излучает кванты ЭММ с массой $2\pi\gamma \cdot m_e$ и эквивалентной энергией, равной $m_e c^2$. Поскольку масса электрона больше массы излучаемых им квантов в $(2\pi\gamma)^{-1}$ раз, то его внутренняя энергия, очевидно, во столько же раз больше величины $m_e c^2$.

Внутреннюю энергию электрона можно оценить из выражения:

$$E_{\text{вн.}} = \frac{m_e c^2}{2\pi\gamma} = m_e v_{\text{гр.}}^2 \approx 2.3852 \cdot 10^6 \cdot m_e c^2.$$

Оценим величину энергии электростатического взаимодействия электронов, используя известное соотношение силового взаимодействия электронов электростатического (кулоновского) и гравитационного (ньютоновского) воздействия:

$$\frac{F_K}{F_H} \cong 4.16688 \cdot 10^{42},$$

откуда при $R = 1$ см можно записать $F_K = k\gamma \left(m_e \sqrt{\frac{F_K}{F_H}} \right) \cdot \left(m_e \sqrt{\frac{F_K}{F_H}} \right)$.

Примем энергию электростатического воздействия одного электрона на другой, равной

$$E = k_2 \cdot m_e c^2,$$

где k_2 – безразмерный коэффициент.

Поскольку электроны для гравитационного воздействия “используют” энергию величиной $M'c^2$, то можно записать:

$$\frac{k_2 \cdot m_e c^2}{M'c^2} = \frac{2k_2}{\pi\gamma^3} = \sqrt{\frac{F_K}{F_H}}. \quad (8)$$

Из (8) следует: $k_2 = 0.952596$, $E' \cong 0.952596 \cdot m_e c^2$.

Взаимная потенциальная энергия системы, состоящей из двух электронов при расстоянии между их центрами, равном $R = d_{\text{эф.е}}$, определяется выражением [3]:

$$E_{\text{п}} = F_K d_{\text{эф.е}} = \frac{e^2}{4\pi\epsilon_0 d_{\text{эф.е}}}.$$

С другой стороны, $E_{\text{п}} = 2 \cdot E'$, тогда имеем

$$\frac{e^2}{4\pi\epsilon_0 d_{\text{эф.е}}} \approx 2 \cdot 0.952596 \cdot m_e c^2.$$

Отсюда получаем $d_{\text{эф.е}} = 1.4795 \cdot 10^{-13}$ см, что близко к полученному ранее.

На основании изложенного выше можно предположить, что энергетические потери электрона компенсируются ЭММ окружающей среды, испытывающей воздействие электрона с внутренней энергией:

$$E_{\text{вн.}} = \frac{m_e c^2}{2\pi\gamma} = m_e \cdot v^2 \approx 2.3852 \cdot 10^6 \cdot m_e c^2.$$

3. Образование уплотнений ЭММ [4]

По аналогии с выражением для внутренней энергии электрона можно предположить, что возможно существование устойчивого полярного уплотнения с минимальной массой, для которого связь с объёмной плотностью непрерывной ЭММ (размерность величин г/см^3) определяется выражением:

$$\rho_{v,\text{min}} = \rho_{v,\text{непр.}} / 2\pi\gamma,$$

где $\rho_{v,\text{непр.}}$ – объёмная плотность массы непрерывной ЭММ; $\rho_{v,\text{min}}$ – объёмная плотность массы уплотнения ЭММ с минимальной массой.

Из приведенного выражения следует:

$$\rho_{v,\text{непр.}} = 2\pi\gamma \cdot \rho_{v,\text{min}}.$$

Частицы и фотоны с массами, превышающими минимальную, сформированные определенным образом из разнополярных уплотнений с минимальной массой, могут иметь объём-

ные плотности масс, выраженные через $\rho_{v, \text{непр.}}$. Для электрона подобное выражение имеет вид:

$$\rho_{v, \text{непр.}} = (2\pi\gamma)^n \rho_{v, e},$$

где n – целое число, $\rho_{v, e} = 4,808 \cdot 10^{11} \text{ \AA} / \text{нм}^3$ – объёмная плотность электрона.

Для уплотнения с минимальной массой справедливо выражение:

$$\rho_{v, \text{min}} = (2\pi\gamma)^{n-1} \rho_{v, e}.$$

Для электрона и уплотнений ЭММ с массой, меньшей массы электрона, запишем следующую систему уравнений:

$$k_1 \cdot h \cdot f_n = m_e c^2,$$

$$k_1 \cdot h \cdot f_{n,i} = m_i c^2.$$

По аналогии с электроном, как с устойчивым уплотнением ЭММ, можно записать и для устойчивых уплотнений ЭММ с меньшими массами:

$$\frac{d_{\text{эф.и}}}{\lambda_{n,i}} = \frac{d_{\text{эф.е}}}{\lambda_n} = 0,63273 \cdot 10^{-3}.$$

Из приведенных выражений следует:

$$\frac{f_{n,i}}{f_n} = \frac{m_i}{m_e} = \frac{\lambda_n}{\lambda_{n,i}} = \frac{d_{\text{эф.е}}}{d_{\text{эф.и}}}. \quad (9)$$

Используя (9), можно определить параметры уплотнения ЭММ.

Следует заметить, что скорость распространения устойчивых уплотнений ЭММ не может превышать скорости “света”. При $v > c$ нарушаются условия компенсации потерь энергии и уплотнение теряет свою массу и переходит в итоге в непрерывную ЭММ, из которой оно и было создано. При распространении кванты гравитационного поля уменьшаются по плотности, уменьшение скорости распространения приводит к снижению тяготения.

4. Формирование рентгеновских излучений

Наиболее интенсивным является “тормозное” излучение при проникновении ускоренных электронов внутрь металлических “анодов”. Такое излучение возникает при ускоряющих напряжениях в несколько сотен киловольт. В этом случае электроны тормозятся (вплоть до остановки) при столкновениях с ядрами атомов анода, происходит и упругая деформация электронов со скоростью движения последних. Когда электроны поглощаются ядрами атомов, то последние интенсивно излучают ЭММ. Последующее восстановление формы электрона происходит со скоростью “света”. Похожие явления возникают и при столкновении ускоренных электронов с электронами атомов (“поляризация”), но интенсивность процессов существенно меньше.

Предположим периодичность процессов деформации и восстановления формы электронов. В таком случае можно записать систему уравнений:

$$k_1 \cdot h \cdot f_n = m_e \cdot c^2,$$

$$k_1 \cdot h \cdot f_j = m_e \cdot v_j^2,$$

где “ c ” – скорость движения ЭММ при восстановлении формы электрона, скорость “света”, при “ c ” > 0 возникает “тормозное” излучение; v_j – скорость движения (и деформации) электрона, уменьшение скорости (и деформации) вызывает “тормозное излучение”.

Из системы уравнений следует:

$$\frac{f_j}{f_n} = \frac{v_j^2}{c^2}.$$

Поскольку $\lambda_n = c/f_n$, $\lambda_j = v_j / f_j$, то

$$\frac{\lambda_n}{\lambda_j} = \frac{c}{v_j} \cdot \frac{f_j}{f_n} = \frac{c}{v_j} \cdot \frac{v_j^2}{c^2} \text{ и } \lambda_j = \lambda_n \cdot \frac{c}{v_j}.$$

По аналогии с выражением

$$\gamma = \frac{1}{6} \left(\frac{d_{\text{эф.е}}}{\lambda_n} \right)^2$$

запишем:

$$\gamma' = \frac{1}{6} \left(\frac{d_{\text{эф.е}}}{\lambda_j} \right)^2 = \gamma \frac{v_j^2}{c^2} = \gamma \frac{f_j}{f_n}.$$

Поскольку $v_{\text{гр.}}^2 = c^2 (2\pi\gamma)^{-1}$, то по аналогии можно записать:

$$v_{\text{изл.}}^2 = v_j^2 (2\pi\gamma')^{-1} = c^2 (2\pi\gamma)^{-1} = v_{\text{гр.}}^2.$$

Таким образом, скорость распространения квантов “тормозного” излучения равна скорости гравитационного взаимодействия. Масса квантов излучений тоже одинакова:

$$m_{\text{кв.рент.}} = 2\pi\gamma' \cdot m_e / |f_j| = 2\pi\gamma \cdot m_e / |f_n|, f_j = f_n \cdot v_j^2 / c^2.$$

Таким образом, “тормозное” и гравитационное излучение имеют одну и ту же электромагнитную природу. Собственно, “тормозное” излучение порождает тяготение. С помощью “тормозного” излучения возможна имитация больших масс. Следует отметить, что при напряжениях на рентгеновской трубке до (25..35) кВ максимум “тормозного” излучения находится в плоскости, перпендикулярной к направлению движения электронов. При напряжении в несколько сотен кВ почти все излучение направленно по направлению потока электронов [5]: максимум тяготения.

5. Заключение

В связи с важностью рассматриваемой научной и мировоззренческой проблемы целесообразно провести экспериментальную проверку положений, изложенных в статье.

Эксперимент может содержать такие пункты:

- 1) имитация массы с помощью “тормозного” излучения в диапазоне частот ($10^{16}..3 \cdot 10^{20}$) Гц. При этом, должны быть отфильтрованы (с помощью металлического экрана) излучения, распространяющиеся со скоростью “света”;
- 2) измерение скорости распространения излучений, указанных в п. 1;
- 3) измерение взаимодействия тел, находящихся в состоянии сверхпроводимости или поляризации.

Выводы:

1. Наличие тяготения при облучении объектов “тормозным” излучением раскрывает электромагнитную природу гравитации, что подтверждается и близостью вычисленных значений эффективного диаметра электрона, полученных на основании дипольной концепции и электростатического взаимодействия.

2. Высокая проникающая способность квантов гравитационного поля и “тормозного” излучения объясняется высокой скоростью распространения, намного превышающей скорость распространения света. Процессы в облучаемых объектах происходят со скоростью, не превышающей скорости света.

3. Пульсации массы электрона сопровождаются излучением квантов ЭММ. Длительность излучения квантов $\tau_{\text{кв.}} = 0.41454 \cdot T_n$. Уменьшение объема электрона сопровождается поглощением ЭММ из окружающей среды. Стабилизация параметров электрона “обеспечивается” внутренней энергией: $E_{\text{вн.}} = m_e c^2 / (2\pi\gamma)$.

4. "Тормозное" излучение присутствует при всех ограничениях движения ЭММ. Доля проникающего излучения зависит от скорости ограничения (торможения) движения ЭММ.

5. Поскольку "тормозное" излучение формируется со скоростью меньшей скорости света, а гравитационное со скоростью, намного превышающей скорость света, то можно предположить дискретность излучаемого электроном кванта в виде неустойчивых фотонов. Масса такого фотона определяется из выражения:

$$m_{\phi} = \pi^2 \gamma^4 m_e / |f_n| \cong 1,442 \cdot 10^{-75} \text{ г}.$$

В системе, состоящей из двух электронов, излучаемые в радиальных направлениях фотоны под воздействием тяготения со стороны противоположного электрона движутся по криволинейным траекториям (к противоположному электрону).

6. При переводе всей массы электрона в ЭММ, находящуюся в непрерывном состоянии, за время $\tau_{\text{кв.}}$, что исключает возможность пополнения внутренней энергии электрона из окружающей среды, необходимая "энергия разрушения" равна:

$$m_{\phi} = \pi^2 \gamma^4 m_e / |f_n| \cong 1,442 \cdot 10^{-75} \text{ г},$$

откуда скорость распространения ЭММ $v_{\text{эвб}} \cong 2,666 \cdot 10^{13} \cdot c$, где c – скорость света.

Возможно, такая ситуация может возникнуть при объединении огромных масс.

По аналогии с (7) можно утверждать, что воздействие ЭММ, движущейся со скоростью $v_{\text{эвб.}}$, будет ослаблено (по сравнению с "обычным гравитационным" с той же энергией) примерно в 10^{20} раз.

Список литературы: 1. *Фейнман Р.* Характер физических законов. М.: Наука, 1987. 168 с. 2. *Біленко, І.І.* Фізичний словник. Київ: Вища школа, 1993 р. 3. *Савельев И.В.* Курс общей физики. Уч. пос. Т.2. М.: Наука, 1968. 455 с. 4. *Рыбин В.В.* Взаимодействие уплотнений электромагнитной материи / АСУ и приборы автоматики. 2015. Вып. 171. С. 44-49. 5. *Бекман И.Н.* Курс лекций: Ядерная медицина. Лекция 2: Компьютерная томография. М., 2010. 511 с.

Поступила в редколлегию 04.02.2016

Рыбин Виктор Вячеславович, инженер. Научные интересы: теоретическая и практическая радиолокация, теоретическая физика. Адрес: Украина, 61204, Харьков, пр. Победы, 72, кв. 299, тел. (057)-336-21-72.

Какурин Николай Яковлевич, канд. техн. наук, проф. Адрес: Украина, 61166, пр. Ленина, 14.