УДК 004.056.53

К. В. Защелкин, канд. техн. наук,
Е. Н. Иванова

МЕТОД СТЕГАНОГРАФИЧЕСКОГО СКРЫТИЯ ДАННЫХ В LUT–ОРИЕНТИРОВАННЫХ АППАРАТНЫХ КОНТЕЙНЕРАХ

Аннотация. *Рассмотрена задача стеганографического скрытия данных, отмечены типичные подходы к организации такого скрытия. Он основан на использовании аппаратных стего-контейнеров с LUT-ориентированной архитектурой. Описаны особенности и преимущества данных контейнеров в сравнении с традиционными стего-контейнерами. Показаны возможности использования предложенного метода как для решения основной задачи стеганографии – скрытой передачи данных, так и для организации цифровых водяных знаков в пространстве LUT-ориентированного стего-контейнера.*

Ключевые слова: *стеганография, цифровые водяные знаки, защита информации, внедрение данных, аппаратный стего-контейнер, LUT-ориентированная архитектура, FPGA, секретная передача данных, защита FPGA-проектов от несанкционированного использования*

K. V. Zashcholkin, PhD.,
E. N. Ivanova

METHOD OF STEGANOGRAPHICAL HIDING OF INFORMATION IN LUT–ORIENTED HARDWARE CONTAINERS

Abstract. *The problem of steganographical hiding of information was reviewed. Typical approaches to the organization of such hiding were marked. We propose a method of steganographical hiding of information based on the use of hardware stego-containers with LUT-oriented architecture. The features and advantages of these containers are described as compared with conventional stego-containers. Both the possibility of using the proposed method for solving the basic problem of steganography – hiding data and for the establishment of digital watermarks in the space of LUT-oriented stego-container.*

Keywords: *steganography, digital watermarks, data protection, embedding data, hardware stego-container, LUT-oriented architecture, FPGA, secret data transfer, protection of FPGA-projects from unauthorized use*

К. В. Защолкін, канд. техн. наук,
О. М. Іванова

МЕТОД СТЕГАНОГРАФІЧНОГО ПРИХОВУВАННЯ ДАНИХ В LUT–ОРІЄНТОВАНИХ АПАРАТНИХ КОНТЕЙНЕРАХ

Анотація. *Розглянуто задачу стеганографічного приховання даних, відзначено типові підходи до організації такого приховання. Він базований на використанні апаратних стего-контейнерів з LUT-орієнтованою архітектурою. Описано особливості та переваги даних контейнерів порівняно з традиційними стего-контейнерами. Показані можливості використання запропонованого методу як для вирішення основного завдання стеганографії - прихованої передачі даних, так і для організації цифрових водяних знаків у просторі LUT-орієнтованого стего-контейнера.*

Ключові слова: *стеганографія, цифрові водяні знаки, захист інформації, вбудовування даних, апаратний стего-контейнер, LUT-орієнтована архітектура, FPGA, секретна передача даних, захист FPGA-проектів від несанкціонованого використання*

Введение. Одним из актуальных направлений исследований в области защиты информации в компьютерных системах и сетях сейчас является цифровая стеганография. В отличие от подходов другого крупного направления теории защиты информации – криптографии, стеганографические подходы не закрывают данные от противной стороны, а скрывают от нее сам факт существования таких данных [1].

Цели методов цифровой стеганографии преимущественно такие:

1) организация скрытых каналов передачи данных внутри открытых каналов;

2) скрытое хранение данных на потенциально незащищенных от несанкционированного доступа носителях информации;

3) встраивание скрытых меток (*цифровых водяных знаков*) в различные информационные объекты с целью контроля их использования [2].

Современные стеганографические методы чаще всего основаны на встраивании секретной информации в мультимедийные контейнеры, информация в которых изначально имеет аналоговую природу: графические, звуковые, видео файлы [3]. Встраивание секретных данных в такие контейнеры основано на приближенном (не точном) представлении информации в пространственной или частотной областях контейнера.

Основные особенности традиционных стего-контейнеров состоят в следующем [4]:

1) контейнеры являются *пассивными блоками данных*, так как выполняют только пассивную функцию хранения информации.

2) контейнеры состоят из не оказывающих явного воздействия друг на друга *автономных элементарных единиц* (например, пикселей для растровой графики и видео, семплов для оцифрованного звука);

3) информация, находящаяся в каждой из элементарных единиц контейнера, *является приближенной*, так как имеет аналоговую природу (пиксели и семплы получены в результате измерения непрерывных физических величин). Неточность этой информации используется для скрытия секретных данных. Однако, с другой стороны, неточность применяется и для *активных атак* на стего-систему [5, 6];

4) для значений элементарных единиц контейнера обычно наблюдается корреляция как на множестве отдельных разрядов, так и на множестве значений соседних элементарных единиц. Эта особенность лежит в основе многих методов *пассивных стего-атак* [5, 6].

В последнее время активизировались исследования в области использования нетрадиционных стего-контейнеров. В частности, появились работы, предлагающие использовать в качестве стего-контейнеров не пассивные информационные объекты, а активные объекты, выполняющие некоторую вычислительную или управляющую функцию. В рамках таких подходов, например, предлагается использовать в качестве стего-контейнеров для внедрения цифровых водяных знаков или для задач скрытого хранения и пересылки защищенной информации исполняемые файлы [7–9] или исходные коды программ [10, 11] для вычислительной машины.

Несмотря на большое количество исследований в области стеганографии, вопросы применения немультимедийных стего-контейнеров и разработки методов скрытия данных в таких контейнерах сейчас находится только на начальном этапе их разрешения. В данной работе предлагается метод внедрения данных в стего-контейнеры с LUT-ориентированной архитектурой, которые отличаются от традиционных контейнеров тем, что является *активными* информационными объектами, состоящими из *неавтономных* элементарных единиц, данные в которых представлены *точно*.

LUT-ориентированная архитектура

LUT (Look Up Table – таблица поиска, таблица просмотра) представляет собой структуру данных, используемую с целью заменить вычисления на операцию поиска заготовленных данных [12]. Подход, основанный на применении LUT, получил название «Вычисления с памятью» (Computing with Memory) [13]. Наибольшего своего развития этот подход достиг в структуре программируемых логических интегральных схем (ПЛИС), в частности в наиболее современной их разновидности FPGA [14]. Упрощенно архитектура FPGA представляет собой совокупность вычислительных модулей, упорядоченных в виде двухмерной матрицы. Основную вычислительную функцию этих модулей выполняют блоки LUT, имеющие обычно четыре (реже пять или шесть) входа и один или два выхода. Блоки LUT могут быть определенным образом соединены между собой и со специализированными модулями (памяти, аппаратного умножения) и выводами микросхемы. Определенная конфигурация соединения блоков LUT, а так же запись в них определенного содержимого приводит к организации вычислительной среды, требуемой для данной задачи.

Блоки LUT в FPGA обычно представляют собой одноразрядную оперативную память. Входы блока LUT при этом являются адресными входами такой памяти (рис. 1).

При наличии n входов блок LUT хранит в себе 2^n бит информации и способен вы-

полнить вычисление значения одной *n*-аргументной булевой функции.

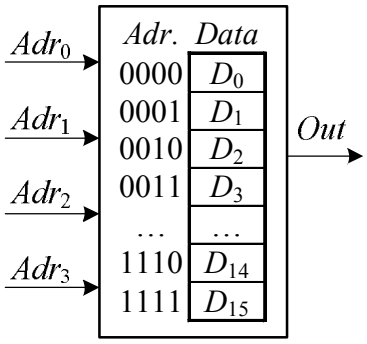

Рис. 1. Организация четырехвходового блока LUT микросхемы FPGA

Цель работы. Микросхемы FPGA на текущий момент являются весьма используемой элементной базой для построения компьютерных и управляющих систем. По многим параметрам FPGA конкурируют с микропроцессорами и микроконтроллерами. По параметрам производительности и возможности организации параллельных вычислений они превосходят микропроцессоры [14]. Микросхемы FPGA построены на основе описанной выше LUT-ориентированной архитектуры. Исходя из этого, можно констатировать перспективность исследования возможности внедрения информации в стего-контейнеры с LUT-ориентированной архитектурой с целью контроля использования FPGA-проектов, а так же для организации стего-защищенных систем передачи и хранения данных на их основе. Цель данной работы состоит в развитии методов цифровой стеганографии путем исследования стего-контейнеров с LUT-ориентированной архитектурой (далее LUT-контейнеров).

Особенности LUT-контейнеров

Предлагаем рассматривать LUT-контейнер как четверку вида

$$LC = (L, E, Interface, Extern), \quad (1)$$

где L – множество блоков LUT; E – множество связей между элементами множества L (отношение инцидентности на множестве L); *Interface* – множество входов и выходов контейнера; *Extern* – множество связей блоков LUT с входами и выходами контейнера.

Каждый компонент LUT_i множества L представляет собой следующую тройку:

$$LUT_i = (In_i, Out_i, Mem_i), \quad (2)$$

где In_i – множество входов блока LUT; Out_i – множество выходов блока LUT; Mem_i – содержимое памяти блока LUT (его внутреннее значение).

Таким образом, LUT-контейнер представляет собой логическую схему, элементами которой являются настраиваемые блоки LUT.

Следует отметить особенности LUT-контейнеров, отличающие их от традиционных мультимедийных контейнеров.

1) LUT-контейнер является *активным* информационным (аппаратным) объектом, выполняющим некоторую вычислительную или управляющую функцию.

2) Элементарные единицы контейнера (блоки LUT) в общем случае связанны друг с другом, так как могут вычислять часть общей булевой функции. Таким образом, элементарные единицы контейнера *не являются автономными*, а оказывают взаимное влияние на функционирование друг друга.

3) Информация, находящаяся в каждой из элементарных единиц контейнера, является *точной*. Произвольное изменение содержимого блока LUT приводит к разрушению контейнера, которое выражается в невозможности выполнения им целевой функции.

Далее предлагается метод внедрения информации в LUT-контейнер. Метод основан на указанных особенностях контейнера, позволяющих выполнить его эквивалентное преобразование, сопряженное с внедрением секретных данных.

Основные положения предлагаемого метода скрытия данных в LUT-контейнере

Основное содержание метода далее излагается в виде совокупности специальных положений, определяющих последовательность и условия реализации процесса скрытия данных в LUT-контейнере.

Первое положение метода – для встраивания одного разряда секретной последовательности используется один из разрядов блока LUT, задействованного в выполнении вычислений. Номер этого разряда (адрес) или правило его определения является элементом стего-ключа.

Второе положение метода – встраивание разряда секретной последовательности основано на следующем утверждении.

Пусть задана некоторая булева функция

$$y = f(y_1^{\lambda_1}, y_2^{\lambda_2} \dots, y_m^{\lambda_m}, a_1, a_2, \dots, a_p). \qquad (3)$$

При этом часть аргументов данной функции является результатом вычислений значений следующей совокупности булевых функций:

$$\begin{aligned} y_1^{\lambda_1} &= f_1^{\lambda_1}(x_1^1, x_2^1, \dots, x_{n_1}^1); \\ y_2^{\lambda_2} &= f_2^{\lambda_2}(x_1^2, x_2^2, \dots, x_{n_2}^2); \\ &\dots\dots\dots\dots\dots\dots\dots\dots\dots \\ y_m^{\lambda_m} &= f_m^{\lambda_m}(x_1^m, x_2^m, \dots, x_{n_m}^m); \end{aligned} \qquad (4)$$

и для функций (3) и (4) установлена такая система правил:

$$\begin{aligned} y_i^{\lambda_i} &= \begin{cases} y_i, \text{ при } \lambda_i = 1; \\ \overline{y_i}, \text{ при } \lambda_i = 0; \end{cases} \\ f_i^{\lambda_i}(x_1^i, x_2^i, \dots, x_{n_i}^i) &= \\ = \begin{cases} f_i^{\lambda_i}(x_1^i, x_2^i, \dots, x_{n_i}^i), \text{ при } \lambda_i = 1; \\ \overline{f_i^{\lambda_i}}(x_1^i, x_2^i, \dots, x_{n_i}^i), \text{ при } \lambda_i = 0. \end{cases} \end{aligned} \qquad (5)$$

где $\lambda_i \in \{0, 1\}, i = 1 \dots m$.

В выражении (3) допускается отсутствие аргументов $a_1, a_2, \dots, a_p$.

Утверждение. Значения булевой функции (3) при аргументах (4) и правилах (5) на любых двоичных наборах не зависят от значений переменных $\lambda_1, \lambda_2, \dots, \lambda_m$.

Действительно, при инвертировании любой из функций $f_i^{\lambda_i}$ (4) и сопровождающейся повторным инвертированием (5) подстановке ее значения в функцию (3), в действительности в функцию (3) в соответствии с законом двойного отрицания будет подставлено прямое значение $f_i^{\lambda_i}$, которое имело место до инвертирования. Таким образом, инвертирование любых функций из (4) с учетом (5) не приводит к изменению значений функции (3).

Из данного утверждения следует, что любые комбинации значений переменных $\lambda_1, \lambda_2, \dots, \lambda_m$ в системе выражений (3) – (5) порождают равносильные булевы функции (3), дающие одинаковые значения на любых наборах.

Принцип встраивания разряда секретной последовательности связан с приведенным *утверждением* следующим образом. Пусть функцию (3) реализует некоторый блок *LUT'*, а функции (4) некоторый набор блоков LUT_1, LUT_2, … , LUT_m. Тогда выборочным инвертированием значений LUT_1, LUT_2, …, LUT_m можно добиться наличия в каждом из них таких значений определенных разрядов, которые соответствуют разрядам внедряемой в контейнер секретной последовательности. При этом значения, вычисляемые блоком *LUT'* с учетом выражений (5), не претерпят каких-либо изменений.

Третье положение метода определяет ограничения на структуру схемы LUT-контейнера, в который внедряется секретная информация. Предлагаемый метод может быть применен только к схеме, имеющей более одного уровня блоков LUT (ранг которой $r > 1$). Действительно, минимальный вариант схемы, совместно реализующей выражения (3) – (5), должен включать набор блоков LUT, расположенных на первом уровне и предназначенных для вычисления функций (4), а также один или несколько блоков LUT, расположенных на втором уровне и предназначенных для вычисления функции (3).

Четвертое положение метода определяет ограничения на использование блоков LUT, входящих в контейнер, для задачи встраивания секретной последовательности.

Первое ограничение такого рода состоит в том, что для встраивания нельзя использовать блоки LUT, непосредственно подключенные к выходам схемы. Это ограничение обусловлено тем, что встраивание секретной информации выполняется в блоки LUT, реализующие выражения (4), к выходам которых должны быть подключены блоки LUT следующего уровня, реализующие выражение (3). Если же блок LUT подключен к выходу схемы, то следующие за ним блоки, реализующие выражение (3), отсутствуют.

Второе ограничение состоит в том, что блок LUT, в который уже встроен разряд секретной последовательности, не может

быть подвергнут инвертированию при выполнении встраивания данных в другой блок LUT данного контейнера. Из этого ограничения следует, что количество блоков LUT, которые можно применить для внедрения разрядов секретной последовательности, зависит от порядка обхода контейнера.

Таким образом, первое указанное ограничение задает верхнюю оценку количества блоков LUT, которые можно использовать для внедрения секретной информации, а второе ограничение – нижнюю оценку. Указанные верхнее и нижнее значения зависят от общего количества блоков, конфигурации их соединения и порядка обхода контейнера в процессе встраивания в него секретной последовательности.

Пятое положение метода определяет порядок формирования стего-ключа для встраивания и извлечения секретной последовательности. Ключ определяется как двойка следующего вида:

$$key = (set, order), \qquad (6)$$

где *set* – номер (адрес) разряда LUT, в который выполняется внедрение бита секретной последовательности. Вместо фиксированного значения *set* этот компонент ключа может содержать некоторое правило, позволяющее получить номер разряда встраивания для каждого шага встраивания; *order* – порядок обхода блоков LUT в контейнере для выполнения встраивания или извлечения секретной последовательности. Вместо фиксированного порядка обхода блоков LUT этот компонент ключа может содержать правило, задающее порядок обхода на каждом шаге встраивания.

Пример реализации предлагаемого метода. Рассмотрим пример, иллюстрирующий основные положения предлагаемого метода. На рис. 2, а представлена схема, состоящая из пяти блоков LUT и реализующая две логические функции y' и y''.

Три из этих блоков расположены на первом уровне схемы и два на втором. Необходимо внедрить в данную схему секретную последовательность $M = (1, 0, 0)$, используя для этого значения, расположенные в блоках LUT по адресу 3, при этом порядок обхода блоков схемы определен их нумераций. Для блоков LUT первого уровня на рис. 2, а показаны хранимые в них значения по адресу 3.

Структура данной схемы соответствует выражениям (3) и (4), для которых может быть применена система правил (5) второго положения метода. В соответствии с третьим положением метода схема, показанная на рис. 2, а может быть использована в качестве контейнера для встраивания секретной информации, так как она имеет два уровня. В соответствии с четвертым положением метода для встраивания могут быть использованы блоки LUT не подключенные к выходам схемы, т.е. блоки LUT_1, LUT_2, LUT_3.

Для внедрения разрядов последовательности будем в порядке нумерации использовать блоки LUT первого уровня. В блоке LUT_1 по адресу 3 хранится значение «0». По этому адресу необходимо поместить значение «1». В соответствии со вторым положением метода выполним инвертирование всех значений, хранящихся в блоке LUT_1. В соответствии с правилами (5) будем инвертировать значения на входе блока LUT, принимающего данные от блока LUT_1, т.е. блока LUT_4. В результате этого функции, вычисляемые схемой, останутся неизменными, однако в блок LUT_1 по адресу 3 будет внедрен первый разряд секретной последовательности, равный значению «1» (рис. 2, б).

Аналогично, используя первое и второе положение предложенного метода, заменяем значение «1», хранящееся в блоке LUT_2 по адресу 3, на второй разряд секретной последовательности «0». Для этого инвертируем все значения, хранящиеся в блоке LUT_2, с одновременным инвертированием значений на входах блоков LUT, принимающих данные от блока LUT_2, т.е. блоков LUT_4 и LUT_5.

Для внедрения третьего разряда секретной последовательности «0» в блок LUT_3 нет необходимости выполнять какие-либо изменения значений блоков LUT данной схемы, так как в блоке LUT_3 по адресу 3 уже хранится значение «0».

Рассмотренный пример показывает возможность внедрения секретной последовательности в LUT-контейнер в соответствии с предложенным методом. В результате такого внедрения функционирование контейнера не изменяется, т.е. его целевая функция (вычислительная, управляющая и т.п.) остается неизменной. Однако контейнер становится носите-

лем информации, которую можно использовать в качестве элемента цифрового водяного знака или для организации стего-защищенных систем передачи и хранения данных.

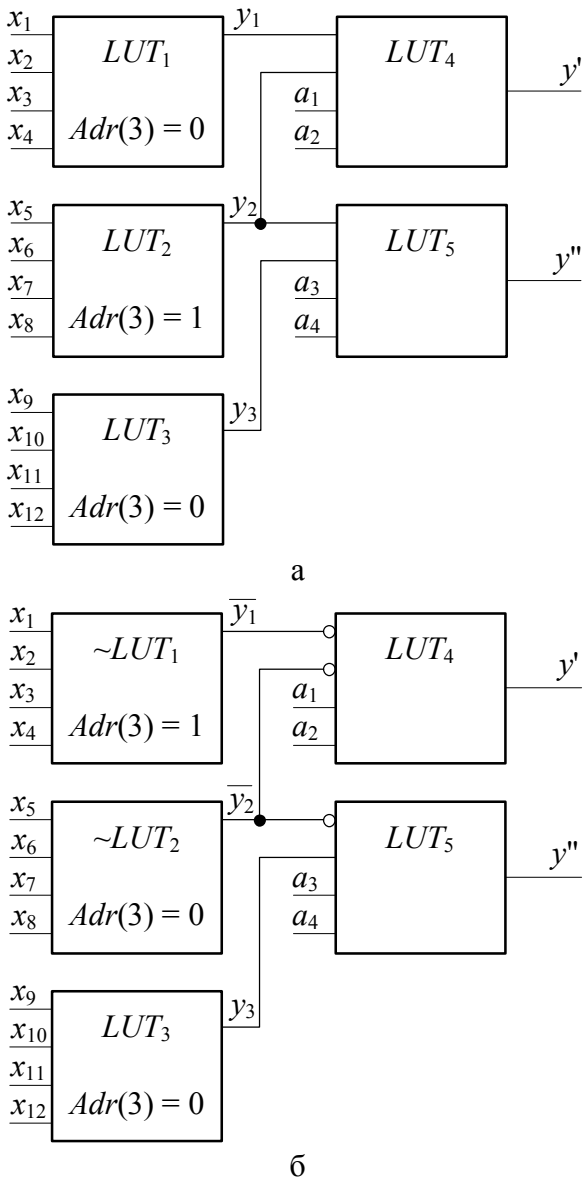

Рис. 2. Пример встраивания секретной последовательности $M = (1, 0, 0)$ в LUT-ориентированный контейнер

Экспериментальное исследование предлагаемого метода. Для экспериментального исследования предлагаемого метода были разработаны аппаратно-программные средства, основанные на использовании микросхем FPGA Altera Cyclone II и САПР Altera Quartus II. На языке TCL была организована группа скриптов, выполняющих взаимодействие с САПР Altera Quartus II для считывания и записи содержимого блоков LUT. Непосредственно подсистема обработки считанных данных, в соответствии с предложенным методом, реализована на языке C# в рамках платформы .Net.

Материалом для экспериментов выступили 25 FPGA-проектов разного объема и назначения. Эксперименты состояли во внедрении случайных секретных последовательностей в LUT-контейнеры FPGA-проектов; исследовании влияния такого внедрения на скоростные характеристики проекта, показатели энергопотребления и тепловыделения; извлечении секретных последовательностей из заполненных контейнеров. Характеристики проекта измерялись средствами САПР Altera Quartus II Timing Analyzer и Power Play.

Экспериментальное исследование показало незначительное влияние применения метода на скоростные характеристики проекта (изменение в среднем на 0,1%), характеристики энергопотребления и тепловыделения (изменение в среднем на 0,25%), величина которого находится на уровне погрешности средств измерения.

Преимущества предлагаемого метода

В отличие от традиционных мультимедийных стего-конвейеров, контейнеры, используемые в рамках предлагаемого метода, являются активными, имеют влияющие друг на друга элементарные единицы, которые хранят в себе представленные точно данные.

Взаимное влияние элементарных единиц контейнера друг на друга дает возможность производить локальные изменения их содержимого, не меняя при этом глобальной функциональности контейнера.

Точное представление данных порождает подходы к противодействию активным стего-атакам типа «стирание секретной информации», недоступные для традиционных мультимедийных контейнеров.

Природа данных, находящихся в элементарных единицах LUT-контейнеров, не дает существенной статистической связи как между разрядами отдельных единиц, так и между разрядами соседних единиц контейнера. Это не позволяет произвести пассивную статистическую стего-атаку на такой контейнер методами стего-анализа, применяемыми для традиционных контейнеров.

Области применения предлагаемого метода. Метод может быть использован для внедрения цифровых водяных знаков в компьютерные и управляющие устройства, построенные на основе элементной базы FPGA и ПЛИС со схожими архитектурами. Такое внедрение дает возможность контролировать правомерность использования проектной информации и самих устройств на различных этапах технологии проектирования и жизненного цикла (синтезированный FPGA проект, конфигурационный файл FPGA**,** действующее устройство).

Предложенный подход может найти применение при организации стего-защищенных систем передачи и хранения данных на основе LUT-контейнеров. Физически в качестве таких контейнеров могу выступать: 1) файлы проектов в САПР FPGA устройств; 2) конфигурационные файлы FPGA; 3) действующие микросхемы FPGA в составе функционирующих устройств.

Кроме того, предложения данной работы могут быть использованы при организации стего-систем на основе LUT-контейнеров иной природы (не связанных с архитектурой микросхем FPGA). Однако выявление и исследование таких контейнеров требует дополнительных исследований.

Выводы. Предложен метод стеганографического скрытия данных, основанный на использовании аппаратных стего-контейнеров с LUT-ориентированной архитектурой. Метод позволяет внедрять двоичную информацию в LUT-контейнер, подвергая элементарные единицы контейнера локальным изменениям, не меняя при этом глобальную функциональность контейнера. Метод предлагается использовать для внедрения секретных данных в LUT-контейнеры (на примере микросхем FPGA) с целью организации в их пространстве цифровых водяных знаков или систем стеганографической защиты информации на их основе.

Список использованной литературы

1. Конахович Г. Ф. Компьютерная стеганография [Текст] / Г. Ф. Конахович, А. Ю. Пузыренко. – К. : МК-Пресс, 2006. – 288 с.

2. Грибунин В. Г. Цифровая стеганография [Текст] / В. Г. Грибунин. – М. : Салон-пресс, 2002. – 344 с.

3. Аграновский А. В. Стеганография, цифровые водяные знаки и стегоанализ [Текст] / А. В. Аграновский, А. В. Балакин, В. Г. Грибунин. – М. : Вузовская книга, 2009. – 220 с.

4. Cox I., Miller M., Bloom J., and Fridrich J. Digital Watermarking and Steganography, (2008), Burlington : Morgan Kaufmann Publishers, 592 p.

5. Fridrich J. Steganography in Digital Media, (2010), New York : Cambridge University Press, 448 p.

6. Shih F. Watermarking, Steganography, and Forensics, (2012), New York : CRC Press, 424 p.

7. Skoudis E., and Zeltser L. Malware: Fighting Malicious Code, (2004), New Jersey : Prentice Hall, 672 p.

8. El-Khalil R., and Keromytis A. Hydan: Hiding Information in Program Binaries, (2004), Proceedings of International Conference on Informaton and Communications Security (ICICS), Malaga, Spain, pp. 187 – 199.

9. Hamilton A., and Danicic S. Survey of Static Software Watermarking, (2011), Proceedings of Internet.Security World Congress (WorldCIS-2011), London, pp. 100 – 107.

10. Hakun L., and Keiichi K. New approaches for software watermarking by register allocation, (2008), Proceedings of the ACIS International Conference on Software Engineering, Artificial Intelligence, and Networking, Parallel Distributed Computing, pp. 63 – 68.

11. Xiao Cheng L., and Zhiming C. Software Watermarking Algorithm Based on Register Allocation, (2010), Proceedings of International Symposium Distributed Computing and Applications to Business Engineering and Science (DCABES), Hong Kong, pp. 539 – 543.

12. Максфилд К. Проектирование на ПЛИС: архитектура, средства, методы [Текст] / К. Максфилд. – М. : Додека-XXI, 2007. – 408 с.

13. Paul S., and Bhunia S. Reconfigurable Computing Using Content Addressable Memory for Improved Performance and Resource Usage, (2008), Proceedings of Design Automation Conference, ACM/IEEE (DAC-2008), Anaheim, pp. 786 – 791.

14. Грушвицкий Р. И. Проектирование систем на микросхемах с программируемой структурой [Текст] / Р. И. Грушвицкий, А.Х. Мурсаев, Е.П. Угрюмов. – СПб. : БХВ, 2010. – 650 с.

Получено 22.10.2013

References

1. Konakhovich G.F., and Puzyrenko A.U. Komp'yuternaya steganografiya [Computer Steganography], (2006), Kiev, Ukraine, *MK-Press Publ.*, 288 p. (In Russian).

2. Gribunin V.G. Tsifrovaya steganografiya [Digital Steganography], (2002), Moscow, Russian Federation, *Salon-Press Publ.*, 344 p. (In Russian).

3. Agranovsky A.V., Balkin A.V., and Gribunin V.G. Steganografiya, tsifrovye vodyanye znaki i stegoanaliz [Steganography Digital Watermarks and Stegoanalysis], (2009), Moscow, Russian Federation, *University Book Publ.,* 220 p. (In Russian).

4. Cox I., Miller M., Bloom J., and Fridrich J. Digital. Watermarking and Steganography, (2008), Burlington, *Morgan Kaufmann,* 592 p. (In English).

5. Fridrich J. Steganography in Digital Media, (2010), New York, *Cambridge University Press Publ.*, 448 p. (In English).

6. Shih F. Watermarking, Steganography, and Forensics, (2012), New York, *CRC Press Publ.*, 424 p. (In English).

7. Skoudis E., and Zeltser L. Malware: Fighting Malicious Code, (2004), New Jersey, *Prentice Hall Publ.*, 672 p. (In English).

8. El-Khalil R. and Keromytis A. Hydan: Hiding Information in Program Binaries, (2004), *Proceedings of International Conference on Informaton and Communications Security (ICICS–2004),* Malaga, Spain, pp. 187 – 199, (In English), doi: 10.1007/978-3-540-30191-2_15.

9. Hamilton A., and Danicic S. A Survey of Static Software Watermarking, (2011), *Proceedings of Internet Security World Congress (WorldCIS–2011),* London, pp. 100 – 107 (In English).

10. Hakun L., and Keiichi K. New Approaches for Software Watermarking by Register Allocation, (2008), *Proceedings of the ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel Distributed Computing*, pp. 63 – 68, (In English), doi: 10.1109/SNPD.2008.137.

11. Xiao Cheng L., and Zhiming C. Software Watermarking Algorithm Based on Register Allocation, (2010), *Proceedings of International Symposium Distributed Computing and Applications to Business Engineering and Science (DCABES),* Hong Kong, pp. 539 – 543, (In English), doi: 10.1109/DCABES.2010.114.

12. Maxfield C. Proektirovanie na PLIS: arkhitektura, sredstva, metody [Design on FPGA: Architecture, Tools, Methods], (2007), Moscow, Russian Federation, *Dodeka–XXI Publ.*, 408 p. (In Russian).

13. Paul S., and Bhunia S. Reconfigurable Computing Using Content Addressable Memory for Improved Performance and Resource Usage, (2008), *Proceedings of Design Automation Conference ACM/IEEE (DAC–2008),* Anaheim, pp. 786 – 791, (In English), doi: 10.1145/1391469.1391670.

14. Grushvitsky R.I., Mursaev A.H., and Ugryumov E.P. Proektirovanie sistem na mikroskhemakh s programmiruemoi strukturoi [Design of Systems on a Chip with Programmable Structure], (2010), St. Petersburg, Russian Federation, *BHV Publ.*, 650 p. (In Russian).

Защелкин Константин Вячеславович,
канд. техн. наук, доц. каф. компьютерных интеллектуальных систем и сетей Одесского нац. политехн. ун-та,
тел.: (048) 734-83-22,
e-mail: const-z@te.net.ua

Иванова Елена Николаевна,
старший преподаватель каф. компьютерных систем Одесского нац. политехн. ун-та,
тел.: (048) 734-83-91,
e-mail: enivanova@ukr.net