АВТОМАТИЗАЦИЯ ПРОЕКТИРОВАНИЯ
COMPUTER-AIDED DESIGN

УДК 519.873:519.718.7
https://doi.org/10.37661/1816-0301-2021-18-3-7-17

Оригинальная статья
Original Paper

Контроль надежности защиты интегральных схем от троянов: кодирование и декодирование комбинационных структур

Л. А. Золоторевич✉, В. А. Ильинков

Белорусский государственный университет информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь
✉*E-mail: zolotorevichla@bsuir.by*

Аннотация. Интегральные схемы и системы на кристалле являются ключевыми звеньями различных промышленных систем и систем обороноспособности государства. Появление контрафактных интегральных схем, проблемы пиратства, перепроизводства, несанкционированное вмешательство в проект микросхемы, аппаратные трояны требуют развития методов и средств их своевременного обнаружения. Трояны могут быть внесены в структуру интегральных схем при разработке и в процессе производства на этапах спецификации, проектирования, верификации и изготовления. Включение в структуру интегральных схем дополнительных элементов ставит под угрозу функциональную пригодность и надежность системы в целом. С целью аппаратной защиты проектов в настоящее время применяются методы аппаратного кодирования.

В работе рассматриваются особенности и надежность логического кодирования комбинационных схем. Предлагается алгоритм взлома кода комбинационных схем, основанный на описании закодированной структуры функцией разрешения и сведении задачи к КНФ-выполнимости. Исходными данными для декодирования структуры цифрового устройства являются структурная реализация закодированной схемы, полученная, например, методом обратного проектирования (проектирования по прототипу), а также активированный физический образец интегральной схемы, в защищенную от несанкционированного доступа память которой загружено правильное значение ключа. Этот образец может использоваться в виде модели черного ящика. Основная идея взлома ключа состоит в том, чтобы решить задачу, не прибегая к исследованиям на большом интервале значений входных и выходных переменных.

Ключевые слова: цифровое устройство, логическое кодирование, декодирование, функция разрешения, выполнимость КНФ-функции

Для цитирования. Золоторевич, Л. А. Контроль надежности защиты интегральных схем от троянов: кодирование и декодирование комбинационных структур / Л. А. Золоторевич, В. А. Ильинков // Информатика. – 2021. − Т. 18, № 3. – С. 7–17. https://doi.org/10.37661/1816-0301-2021-18-3-7-17

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Поступила в редакцию | Received 21.04.2021
Подписана в печать | Accepted 24.05.2021
Опубликована | Published 29.09.2021

Monitoring the reliability of integrated circuits protection against Trojans: encoding and decoding of combinational structures

Lyudmila A. Zolotorevich✉, Valery A. Ilyinkov

Belarusian State University of Informatics and Radioelectronics,
st. P. Brovki, 6, Minsk, 220013, Belarus
✉*E-mail: zolotorevichla.bsuir.by*

Abstract. Integrated circuits, systems on a chip are the key links in various industrial systems and state defense systems. The emergence of counterfeit integrated circuits, problems of piracy, overproduction, unauthorized interference in the design of microcircuit, hardware Trojans require the development of methods and means of their timely detection. Trojans can be introduced into the integrated circuits structure both on the development stage and during the production process, including the stages of specification, design, verification and manufacturing. The inclusion of additional elements in the integrated circuits structure jeopardizes the functional suitability and reliability of the system as a whole. For the purpose of hardware protection of projects, the methods of hardware coding are currently used.

The paper discusses the features and reliability of logical coding of combinational circuits. An algorithm for cracking the code of combinational circuits is proposed, based on the description of encoded structure by the resolution function and reducing the problem to SAT CNF. The initial data for decoding the structure of a digital device is the structural implementation of encoded circuit, obtained, for example, by reverse engineering (prototype design), as well as an activated physical sample of an integrated circuit, when into protected from unauthorized access memory the correct key value is loaded. This sample can be used as a black box model. The main idea of breaking a key is to solve a problem without research on a large interval of values of input and output variables.

Keywords: digital device, logical coding, decoding, resolution function, SAT CNF

For citation. Zolotorevich L. A., Ilyinkov V. A. Monitoring the reliability of integrated circuits protection against Trojans: encoding and decoding of combinational structures. *Informatics*, 2021, vol. 18, no. 3, pp. 7–17 (In Russ.). https://doi.org/10.37661/1816-0301-2021-18-3-7-17

Conflict of interest. The authors declare of no conflict of interest.

Введение. Серьезными проблемами для электронной и оборонной промышленности в последние годы стали пиратство, перепроизводство и контрафакция, что привело к необходимости защиты проектов СБИС и систем на кристалле (СнК) от несанкционированного вмешательства в цикл проектирования и (или) производства интегральных схем (ИС) [1]. По оценкам службы обработки информации IHS (Technology Information Handling Services), финансовый риск из-за контрафактных и несанкционированных микросхем оценивается более чем в 169 млрд долл. в год, что примерно в 10 раз превышает ущерб от пиратства в области программного обеспечения [2]. Для оборонной промышленности важной задачей является возможность использования контрафактных ИС с модифицированными функциями, что в определенное время может деструктивно повлиять на функционирование структуры, ухудшить ее эксплуатационные характеристики, привести к раскрытию конфиденциальной информации и т. д. Кроме больших финансовых потерь, существует реальная проблема обеспечения национальной безопасности, так как 15 % ИС в системах оборонной промышленности являются контрафактными. В связи с этим стала очевидной необходимость защиты проектов на основе создания таксономии нарушений и отклонений, общего подхода к контролю СБИС и СнК, с моделями которых приходится работать при проектировании и организации контроля на всех этапах жизненного цикла цифровой системы с учетом злонамеренных внедрений в цикл проектирования и производства ИС. Как развитие теории контролепригодного проектирования (Design-for-Testability, DfT) в работе [3] предлагается подход к проектированию Design for-Trust (DfTr), который дополнительно включает средства для контроля и предотвращения аппаратных атак при проектировании и изготовлении СБИС.

В последние годы для защиты проектов ИС применяются методы и средства аппаратного кодирования комбинационных блоков. Для обеспечения надежности подобной защиты необходимы средства контроля эффективности применяемых методов кодирования и выявления внесенных троянов на основе создания общего подхода к контролю проектов на всех этапах проектирования и производства.

В настоящей статье рассматриваются некоторые особенности метода логического кодирования структурных схем цифровых устройств комбинационного типа. Предлагается способ взлома кода при наличии информации о структуре закодированного объекта и возможности доступа к физической модели. Задача решается на основе описания закодированной структуры в виде КНФ-функции разрешения, решения задачи выполнимости (SAT) и физического моделирования объекта.

Логическое кодирование ИС как метод аппаратной защиты. В работе [2] проанализированы различные модели процесса злонамеренного искажения проекта, описывающие условия, при которых подобное искажение может внедриться в цифровую систему. В числе возможных источников искажений рассматриваются поставщики базовых функциональных блоков интеллектуальной собственности (IP's), которые приобретаются разработчиками СнК, собственно разработчики СнК, а также кремниевые фабрики – изготовители СнК. Методы несанкционированного доступа в проект могут быть различными. Например, они могут основываться на применении специальных средств САПР, способных исказить проект на RTL-уровне. В современных условиях наиболее уязвимым может оказаться этап производства.

Одним из методов борьбы с вышеупомянутыми угрозами является логическое кодирование, которое обеспечивает доступ к объекту только авторизованным пользователям [4]. Метод предполагает сокрытие функциональности проекта и использование ключа, который выводит систему в область правильного функционирования.

Основная идея кодирования состоит в том, чтобы изменить конструкцию ИС, добавив в нее дополнительные логические элементы и новые входы, называемые ключевыми, т. е. применить метод обфускации структуры объекта. В такой постановке если злоумышленник не владеет ключом, то ему недоступна внутренняя реализация объекта. Задача структурной обфускации и логического кодирования заключается в том, чтобы затруднить или сделать невозможным получение правильного ключа. Ключевые входы подсоединяются к защищенной от несанкционированного доступа памяти, а закодированная схема будет работать правильно, только если поданы верные значения на ее ключевые входы. Значения ключевых входов передаются после изготовления микросхем конечным пользователям (рис. 1). Таким образом, логическое кодирование основывается на предположении, что производитель не знает и не может вычислить правильные значения ключевых входов. В противном случае поиск правильного ключа должен быть для злоумышленника затруднителен.

Существуют различные методы кодирования комбинационной логики, в которых в качестве ключевых вентилей используются элементы XOR / XNOR [1, 5–7], AND / OR [8], мультиплексоры [9] или комбинации этих вентилей [10]. Выбор линии для включения вентиля и типа применяемого вентиля существенно влияет на эффективность кодирования. Воздействие неправильного ключа можно сравнить с влиянием неисправности константного типа на данной линии (рис. 1). В отличие от вентилей OR, NOR, AND, NAND при выборе в качестве ключевых вентилей XOR или NXOR применение неправильного ключа приводит к появлению неисправности константного типа в любом случае и на любом входном воздействии, что влияет в целом на эффективность кодирования. Кроме типа применяемого вентиля есть еще два основных способа увеличить влияние кодовых вентилей на значения выходов схемы. Один из них заключается в выборе линий, сигналы в которых влияют на максимально возможное количество выходов схемы, другой – в повышении чувствительности схемы в ответ на применение неправильного ключа.

Выбор линии для включения вентиля в большой степени влияет на эффективность кодирования. Один из подходов основан на случайном выборе линии схемы [11]. В работе [1] показана недостаточная эффективность этого подхода. Во-первых, вставка ключевого вентиля в случайно выбранную линию схемы не может гарантировать необходимое расстояние Хэмминга

между истинным выходным вектором и полученным в случае применения неправильного ключа. Оптимальное расстояние Хэмминга 50 % максимизирует двусмысленность злоумышленника относительно реакции схемы в случае применения неправильного ключа.

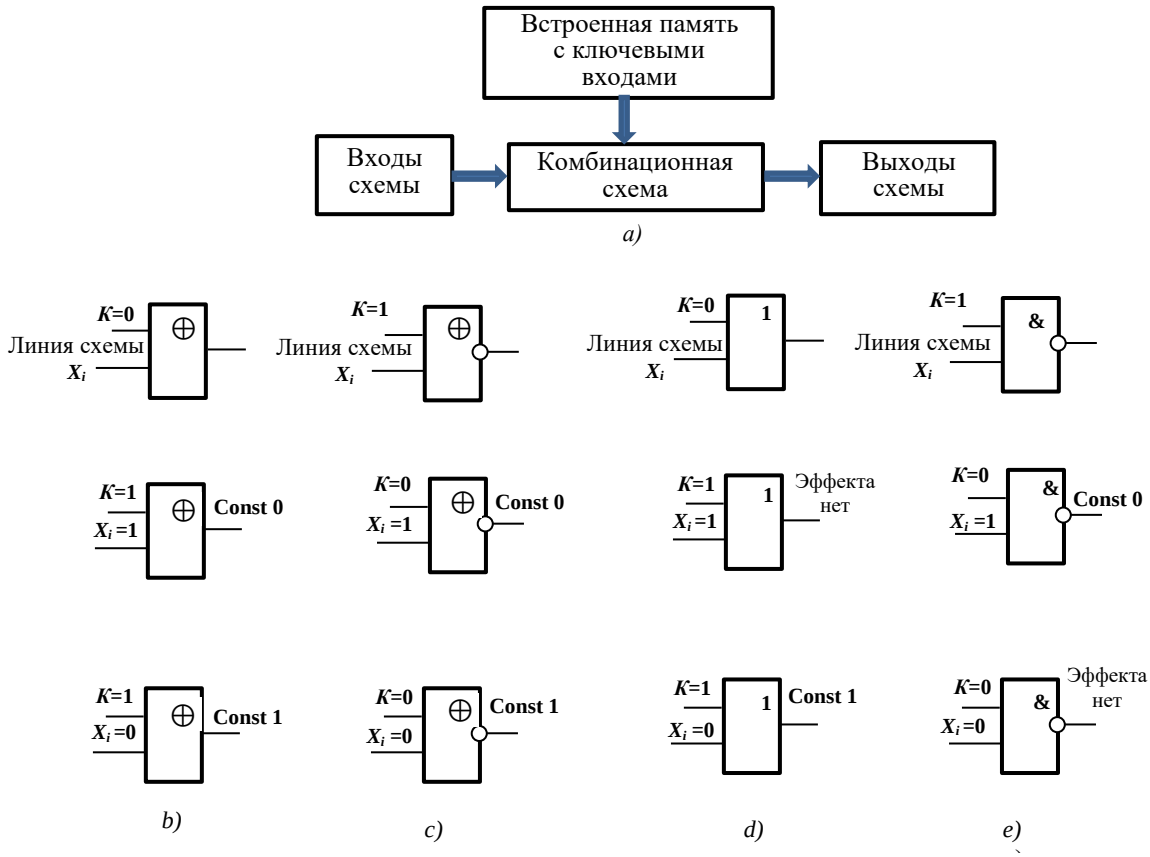


Рис. 1. Логическое кодирование цифровых устройств: *a)* общая идея кодирования; *b)* эффекты применения ключевого вентиля XOR-типа; *c)* NXOR; *d)* OR; *e)* NAND

Fig. 1. Logical coding of digital devices: a) the general idea of coding; b) the effects of using an XOR-type key gate; c) NXOR; d) OR; e) NAND

В работе [12] для характеристики эффективности выбора линии в схеме для введения ключевого вентиля предложено использовать метрику $\mathrm{M} = N_0P_0 \cdot N_0O_0 + N_0P_1 \cdot N_0O_1$, где N_0P_0 (N_0P_1) – количество входных наборов, которые обнаруживают неисправность типа const 0 (const 1), а N_0O_0 (N_0O_1) – количество ошибочных бит выходного вектора в результате появления неисправности const 0 (const 1). Данная метрика может быть усовершенствована для получения возможности отслеживать в динамике параметры N_0O_0 (N_0O_1) для анализа неактивированных выходов при кодировании. Использование метрики M при кодировании можно сформулировать как нахождение множества неисправностей кодируемой схемы, которые вместе будут влиять на 50 % выходных линий при их активизации. Кодирование на основе использования метрики M требует моделирования схемы $Q = 2s \cdot 2^n$ раз, где s – общее количество линий схемы (переменных полного состояния схемы), n – количество входных переменных схемы. Для схемы на рис. 2 $\mathrm{M} = 256 \cdot 34 = 8704$. Для реальных схем подобный подход практически неприемлем по причине высоких вычислительных затрат. В то же время с целью оптимизации вычислительных процедур предлагается эвристическое решение – сократить количество моделируемых входных наборов до 1000 [12].

Основная задача, которая должна быть решена при практической реализации данной общей идеи, заключается в том, чтобы определить оптимальное множество внутренних линий схемы

и количество ключевых элементов для создания максимальных трудностей для злоумышленника по поиску правильного ключа. При включении очередного вентиля в процессе кодирования логических устройств необходимо проводить анализ на появление эффекта маскирования неисправностей, который способен блокировать эффект кодирования [1, рис. 2]. При наличии избыточности отдельные линии схемы не могут быть активированы ни одним входным набором, поэтому вставка ключевого вентиля в данном случае может быть бесполезной [1, рис. 1].

В работе [1] на примере рассмотрена возможность повышения эффективности кодирования за счет включения в структуру схемы вентилей управления, которые позволяют активизировать влияние неправильного состояния каждого отдельного бита ключа на формирование выходного вектора закодированной схемы. Для того чтобы усилить влияние неправильного бита кодового слова на результат функционирования схемы, управляющие вентили объединяют биты кодового слова в группы, использовав при этом их выходы в качестве входов ключевых вентилей. В этом случае реализуется групповое воздействие нескольких битов кодового слова на активацию ключевого вентиля. Если хотя бы один из ключевых входов, включенных в группу, принимает неправильное значение, ключевой вентиль окажется активированным.

Контроль надежности кодирования комбинационных схем. В работе [2] предлагается подход SAT-атаки для определения кода аппаратной защиты комбинационных схем цифровых устройств на структурном уровне. Подход основан на сведении задачи к определению выполнимости булевой функции.

Исходными данными для декодирования структуры цифрового устройства являются структурная реализация закодированной схемы, полученной, например, методом обратного проектирования (проектирования по прототипу), а также активированный физический образец ИС, в защищенную от несанкционированного доступа память которой заказчик загрузил правильное значение ключа. Этот образец может использоваться в виде модели черного ящика $Y = eval\ (X)$. Основная идея SAT-атаки взлома ключа состоит в том, чтобы определить правильный ключ, не прибегая к исследованиям на большом интервале входных (выходных) переменных [2].

Обозначим $\vec{Y} = f(\vec{X})$ функцию, реализуемую комбинационной схемой с первичными входами $\vec{X}$ и выходами $\vec{Y}$, а $Cir_a(\vec{X},\vec{Y})$ – КНФ функции разрешения исходной схемы. Сведем задачу получения ключа к описанию закодированной схемы в виде КНФ-представления булевой функции разрешения $Cir_b(\vec{X},\vec{K},\vec{Y})$, где $\vec{X}$ – первичные входы схемы, $\vec{X} = (x_1, x_2, ..., x_n)$; $\vec{K}$ – ключевые входы схемы, $\vec{K} = (k_1, k_2, ..., k_r)$; $\vec{Y}_i$ – выходные линии схемы, $\vec{Y} = (y_1, y_2, ..., y_m)$.

Если $F = f(\vec{X},\vec{Y})$ – функция, реализуемая исходной схемой, то для любого $\vec{X}$ $F = Cir_b(\vec{X},\vec{K},\vec{Y})$, если применить к закодированной схеме правильное значение ключа. Цель злоумышленника состоит в том, чтобы найти такой ключ $\vec{K} = (k_1, k_2, ..., k_r)$, при котором $\forall \vec{X} \quad Cir_b(\vec{X},\vec{K},\vec{Y}) \wedge Cir_a(\vec{X},\vec{Y})$. Однако злоумышленник не может получить формулу $Cir_a(\vec{X},\vec{Y})$, так как ему недоступно структурное описание исходной схемы. Не получив доступ к структуре исходной схемы и не имея, таким образом, возможности построить отношение $Cir_a(\vec{X},\vec{Y})$, злоумышленник может наблюдать реакцию схемы на требуемое входное воздействие по активированной ИС, выполнив функцию черного ящика *eval*:

$$\vec{X}_i = (x_1, x_2, ..., x_n) \rightarrow \vec{Y}_i = (y_1, y_2, ..., y_m).$$

Для заданного набора входных векторов $\vec{X}_1, \vec{X}_2, ..., \vec{X}_p$ и соответствующих выходных наблюдений $\vec{Y}_i, \vec{Y}_2, ..., \vec{Y}_P$ определение ключевого значения, которое согласуется с p наблюде-

ниями, является достаточно простым, если свести задачу к решению выполнимости формулы $\wedge_{j=1}^{p} Cir_b(\vec{X}_j, \vec{K}, \vec{Y}_j)$. Однако если теперь выполнить новое наблюдение на физическом образце схемы $eval(\vec{X}_s) = \vec{Y}_s$, то нет гарантии, что удовлетворительное присваивание $\vec{K}$ для формулы $\wedge_{j=1}^{p} Cir_b(\vec{X}_j, \vec{K}, \vec{Y}_j)$ также будет удовлетворительным присваиванием $\vec{K}$ для формулы $\wedge_{j=p+1}^{2^n} Cir_b(\vec{X}_j, \vec{K}, \vec{Y}_j)$.

Для практической атаки при большом числе входных переменных функция *eval* может быть определена только на небольшом числе входных векторов $Cir_b(\vec{X}, \vec{K}, \vec{Y}) \Leftrightarrow eval(\vec{X}) = \vec{Y}$, в то время как $\exists \vec{K} : \forall \vec{X} \quad Cir_b(\vec{X}, \vec{K}, \vec{Y}) \wedge Cir_a(\vec{X}, \vec{Y})$.

Решение проблемы заключается в том, что вместо поиска правильного ключа выполняется определение ключа как члена класса эквивалентности ключей, который дает на выходах правильный результат для всех входных состояний.

Определение 1. Два ключа $\vec{K}_1$ и $\vec{K}_2$ являются эквивалентными ($\vec{K}_1 = \vec{K}_2$) тогда и только тогда, когда для входного значения $\vec{X}_i$ закодированная схема выдает одинаковое выходное значение $\vec{Y}_i$ для ключей $\vec{K}_1$ и $\vec{K}_2$.

Для определения правильного ключа итеративно исключаются ключи из класса эквивалентности, которые выдают неправильные значения выходов по крайней мере для одного входного шаблона. Класс эквивалентных ключей определяется на некотором входном (выходном) векторе путем решения выполнимости функции $Cir_b(\vec{X}_j, \vec{K}, \vec{Y}_j)$ полным методом.

Определение 2. Входной вектор $\vec{X}^d$ называется различающим, если реакция схемы при использовании ключа $\vec{K}_1$ равна $\vec{Y}_1^d$ и отличается от реакции $\vec{Y}_2^d$ при использовании ключа $\vec{K}_2$.

При наличии различающего набора можно проверить реакцию активированной схемы для входа $\vec{X}^d$ и использовать ее, чтобы исключить ключ $\vec{K}_1$ или $\vec{K}_2$ как не входящий в класс эквивалентности правильных ключей.

Алгоритм нахождения входного различающего набора:

1. i :=1.

2. $F_i = Cir_b(\vec{X}, \vec{K}_1, \vec{Y}_1) \wedge Cir_b(\vec{X}, \vec{K}_2, \vec{Y}_2)$.

3. Если условие $F_i \wedge \vec{Y}_1 \neq \vec{Y}_2$ не выполняется, переход к п. 8. Различающий набор не определен.

4. Решение $F_i = Cir_b(\vec{X}, \vec{K}_1, \vec{Y}_1) \wedge Cir_b(\vec{X}, \vec{K}_2, \vec{Y}_2) \wedge (\vec{Y}_1 \neq \vec{Y}_2)$, $\vec{X}_i{}^d := \vec{X}$. Входной набор $\vec{X}_i{}^d$ является различающим.

5. $\vec{Y}_i^d := eval(\vec{X}_i{}^d)$.

6. $i = i+1$.

7. $F_i = F_{i-1} \wedge Cir_b(\vec{X}_i^d, \vec{K}_1, \vec{Y}_i^d) \wedge Cir_b(\vec{X}_i^d, \vec{K}_2, \vec{Y}_i^d)$, переход к п. 3.

8. Выход.

Каждая итерация алгоритма исключает хотя бы один неверный член рассматриваемого класса эквивалентности ключей. Это связано с тем, что поиск различающего входного набора ведется с условием $\vec{Y}_1 \neq \vec{Y}_2$, т. е. при одинаковых входных данных выходные данные должны отличаться для разных ключей. Следовательно, хотя бы один ключ окажется неправильным. Алгоритм завершается, когда определен правильный ключ из класса эквивалентных ключей.

Покажем применение алгоритма на примере фрагмента схемы. Для полноты изложения рассмотрим получение функции разрешения F^f. Функция F^f, называемая функцией разрешения для логической функции f, зависит не только от аргументов функции f, но и от самой f и принимает значение логической единицы при всех допустимых состояниях входных и вы-

ходной переменных [14]. Функция F^f принимает значение 0 при всех недопустимых состояниях входных и выходной переменных. Представим функции разрешения F^f и запрета $\overline{F^f}$ в виде таблицы истинности для конъюнкции $f = a \cdot b$ (табл. 1).

Таблица 1. Функции разрешения и запрета для элемента AND

Table 1. Functions of permission and prohibition for AND element

a	***b***	***f***	F^f	$\overline{F^f}$
0	**0**	**0**	1	0
0	**1**	**0**	1	0
1	**0**	**0**	1	0
1	**1**	**1**	1	0
0	0	1	0	1
0	1	1	0	1
1	0	1	0	1
1	1	0	0	1

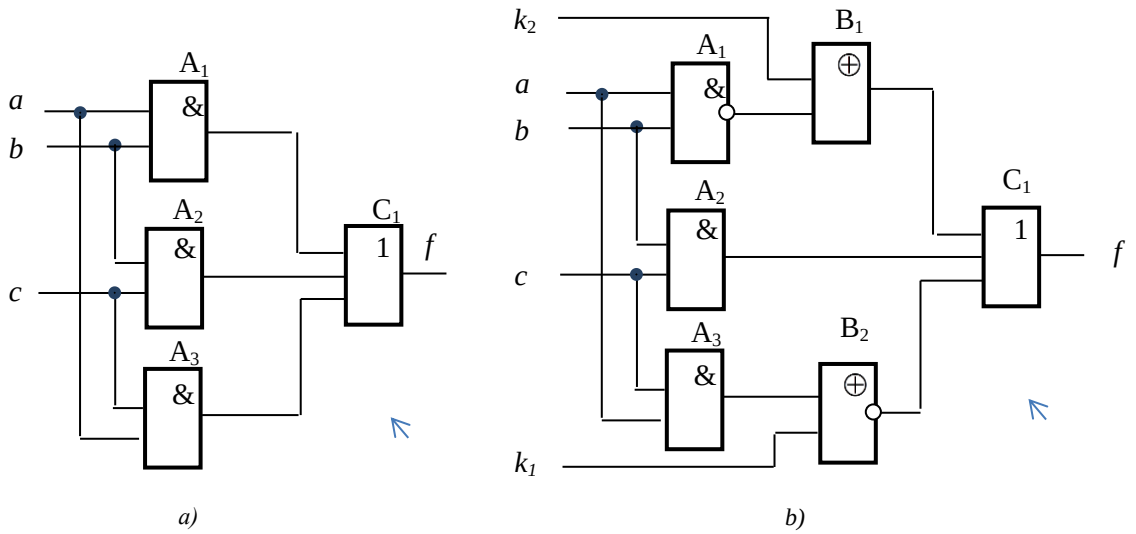
В табл. 2 приведены КНФ функций разрешения для некоторых типов вентильных элементов.

Таблица 2. Функции разрешения

Table 2. Functions of permission

Однобитовые арифметические и логические уравнения	КНФ функций разрешения
$f = b \vee c$	$(\bar{b} \vee f)(\bar{c} \vee f)(b \vee c \vee \overline{f})$
$f = b \cdot c$	$(b \vee \overline{f})(c \vee \overline{f})(\bar{b} \vee \bar{c} \vee f)$
$f = a \oplus b$	$(a \vee b \vee \overline{f})(a \vee \bar{b} \vee f)(\bar{a} \vee b \vee f)(\bar{a} \vee \bar{b} \vee \overline{f})$
$f = a \sim b$	$(a \vee b \vee f)(a \vee \bar{b} \vee \overline{f})(\bar{a} \vee b \vee \overline{f})(\bar{a} \vee \bar{b} \vee f)$

Пример. На рис. 2, *a* изображена схема и вариант ее кодирования, которое выполнено путем включения дополнительных вентилей B_1 XOR и B_2 NXOR (рис. 2, *b*).



a) *b)*

Рис. 2. Комбинационная схема для иллюстрации алгоритма взлома ключа:
a) исходная; *b)* закодированная

Fig. 2. Combinational scheme to illustrate the key cracking algorithm:
a) original; b) encoded

Приведем функцию разрешения закодированной схемы $Cir_b(\vec{X},\vec{K},\vec{Y})$. При формировании функции разрешения схемы *a*, *b*, *c* – входные переменные, а a_1, a_2, a_3, b_1, b_2, c_1 – выходы соответствующих элементов:

$$Cir_b = (a \vee b \vee a_1)(a \vee \bar{b} \vee a_1)(\bar{a} \vee b \vee a_1)(\bar{a} \vee \bar{b} \vee \bar{a_1}),$$

$$(b \vee c \vee \bar{a_2})(b \vee \bar{c} \vee \bar{a_2})(\bar{b} \vee c \vee \bar{a_2})(\bar{b} \vee \bar{c} \vee a_2),$$

$$(a \vee c \vee \bar{a_3})(a \vee \bar{c} \vee \bar{a_3})(\bar{a} \vee c \vee \bar{a_3})(\bar{a} \vee \bar{c} \vee a_3),$$

$$(a_3 \vee k_1 \vee b_2)(a_3 \vee \bar{k_1} \vee \bar{b_2})(\bar{a_3} \vee k_1 \vee \bar{b_2})(\bar{a_3} \vee \bar{k_1} \vee b_2),$$

$$(k_2 \vee a_1 \vee \bar{b_1})(k_2 \vee \bar{a_1} \vee b_1)(\bar{k_2} \vee a_1 \vee b_1)(\bar{k_2} \vee \bar{a_1} \vee \bar{b_1}),$$

$$(b_1 \vee a_2 \vee b_2 \vee \bar{c_1})(b_1 \vee a_2 \vee \bar{b_2} \vee c_1)(b_1 \vee \bar{a_2} \vee b_2 \vee c_1)(b_1 \vee \bar{a_2} \vee \bar{b_2} \vee c_1),$$

$$(\bar{b_1} \vee a_2 \vee b_2 \vee c_1)(\bar{b_1} \vee a_2 \vee \bar{b_2} \vee c_1)(\bar{b_1} \vee \bar{a_2} \vee b_2 \vee c_1)(\bar{b_1} \vee \bar{a_2} \vee \bar{b_2} \vee c_1).$$

Для декодирования выполним следующие действия:

1. В качестве входного вектора для поиска ключей используем случайный вектор $\vec{X} = 110$, для которого определим $\vec{Y}$ с помощью активированной схемы: $eval(\vec{X}) = 1$.

2. Найдем решение задачи SAT для функции $F = Cir_b ab\bar{c}\bar{c}_1$ на основе полного алгоритма решения выполнимости:

$$F = \bar{a}_1\bar{a}_2\bar{a}_3(k_1 \vee b_2)(\bar{k_1} \vee \bar{b_2})(k_2 \vee \bar{b_1})(\bar{k_2} \vee b_1)ab\bar{c}\bar{c}_1.$$

Функция выполнима при следующих условиях:

$$F = k_1k_2\bar{a}_1\bar{a}_2\bar{a}_3b_1\bar{b_2}c_1, \quad \vec{K}_1 = 11;$$

$$F = \bar{k_1}k_2\bar{a}_1\bar{a}_2\bar{a}_3b_1b_2c_1, \quad \vec{K}_2 = 01;$$

$$F = \bar{k_1}\bar{k_2}\bar{a}_1\bar{a}_2\bar{a}_3\bar{b_1}b_2c_1, \quad \vec{K}_3 = 00.$$

Таким образом, найдены три ключа: $\vec{K}_1 = 11$, $\vec{K}_2 = 01$, $\vec{K}_3 = 00$, которые составляют класс эквивалентных на данном этапе декодирования.

3. Найдем различающий входной набор для первых двух ключей $\vec{K}_1 = 11$ и $\vec{K}_2 = 01$ из найденного класса. Для этого необходимо вычислить булеву функцию

$$F_1 = Cir_b(\vec{X},\vec{K}_1,\vec{Y}_1) \wedge Cir_b(\vec{X},\vec{K}_2,\vec{Y}_2). \tag{1}$$

Для решения равенства (1) определим один из выполнимых входных (выходных) векторов для первого ключа $\vec{K}_1 = 11$. Задача решается на основе неполного алгоритма выполнимости функции

$$F = Cir_b k_1 k_2. \tag{2}$$

Получим $F = a\bar{b}ck_1k_2a_1\bar{a_2}a_3\bar{b_1}b_2c_1$. Таким образом определены новый входной $\vec{X} = 101$ и выходной $\vec{Y} = 1$ векторы. Проверим выполнимость функции $F = Cir_b a\bar{b}c\bar{k_1}k_2\bar{c_1}$ на входном

наборе $\vec{X} = 101$ при значении выходного вектора, отличного от полученного в равенстве (2) для второго ключа $\vec{K}_2 = 01$. В результате $F = a\bar{b}c\overline{k_1}k_2a_1\overline{a_2}a_3\overline{b_1}\overline{b_2}\overline{c_1}$.

Следовательно, $\vec{X} = 101$ является различающим входным набором, так как разным ключам соответствуют разные выходы.

4. Определим вектор $\vec{Y} \Rightarrow \overrightarrow{X} = 101$, $\vec{Y} = eval(\vec{X}) = 1$ с помощью активированной схемы.

5. Вычислим функцию (1): для $\vec{K}_1 = 11$ $F_{K_1} = Cir_b a\bar{b}ck_1k_2c_1$, для $\vec{K}_2 = 01$ $F_{K_2} = Cir_b a\bar{b}c\overline{k_1}k_2c_1$. Функция $F_{K_2} = Cir_b a\bar{b}c\overline{k_1}k_2c_1$ не выполняется. Следовательно, ключ $\vec{K}_2 = 01$ исключается из класса эквивалентности.

6. Вычислим функцию (1) для $\vec{K}_3 = 00$: $F_{K_3} = Cir_b a\bar{b}c\overline{k_1}\overline{k_2}c_1$. Функция F_{K_3} не выполняется, так как ключ $\vec{K}_3 = 00$ неправильный.

В связи с тем что ключи $\vec{K}_2 = 01$ и $\vec{K}_3 = 00$ оказались неверными, правильным является единственный ключ, оставшийся в классе эквивалентности правильных ключей, $\vec{K} = 11$.

Заключение. В работе рассмотрены некоторые особенности кодирования структурной реализации ИС на основе использования средств тестового диагностирования.

Для оценки надежности кодирования предлагается алгоритм декодирования, который проиллюстрирован на примере. Анализ надежности кодирования основан на решении SAT КНФ-функции разрешения, описывающей закодированную структуру.

Метод нахождения правильного ключа из класса эквивалентности предназначен для решения проблемы декодирования схем практических размеров без необходимости исследовать всю область возможных решений.

Вклад авторов. Л. А. Золоторевич – постановка задачи, анализ методов аппаратной защиты, разработка алгоритма декодирования; В. А. Ильинков – обоснование выбора вентильного элемента для кодирования, анализ результатов на основе проведения компьютерного эксперимента, оформление статьи.

Список использованных источников

1. Золоторевич, Л. А. Аппаратная защита цифровых устройств / Л. А. Золоторевич // Вестник Томского гос. ун-та. Управление, вычислительная техника, информатика. – 2020. – № 50. – С. 69–78. https://doi.org/10.17223/19988605/50/9

2. Subramanyan, P. Evaluating the security of logic encryption algorithms / P. Subramanyan, S. Ray, S. Malik // 2015 IEEE Intern. Symp. on Hardware Oriented Security and Trust (HOST), Washington, DC, USA, 5–7 May 2015. – Washington, 2015. – P. 137–143.

3. Security analysis of integrated circuit camouflaging / J. Rajendran [et al.] // ACM SIGSAC Conf. on Computer & Communications Security, Berlin, Germany, 4–8 Nov. 2013. – Berlin, 2013. – P. 709–720.

4. Roy, J. A. EPIC: Ending piracy of integrated circuits / J. A. Roy, F. Koushanfar, I. L. Markov // IEEE Computer. – 2010. – Vol. 43, no. 10. – P. 30–38.

5. On improving the security of logic locking / M. Yasin [et al.] // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. – 2016. – Vol. 35, no. 9. – P. 1411–1424.

6. Logic encryption: a fault analysis perspective / J. Rajendran [et al.] // DATE '12 : Proc. of the Conf. on Design, Automation and Test in Europe, Dresden, Germany, March, 2012. – Dresden, 2012. – P. 953–958.

7. Fault analysis-based logic encryption / J. Rajendran [et al.] // IEEE Transactions on Computers. – 2015. – Vol. 64, no. 2. – P. 410–424.

8. A novel hardware logic encryption technique for thwarting illegal overproduction and hardware trojans / S. Dupuis [et al.] // 20th IEEE Intern. On-Line Testing Symp., Platja d'Aro, Catalunya, Spain, 7–9, July 2014. – Platja d'Aro, 2014. – P. 49–54.

9. Plaza, S. M. Solving the third-shift problem in IC piracy with test-aware logic locking / S. M. Plaza, I. L. Markov // IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems. – 2015. – Vol. 34, no. 6. – P. 961–971.

10. Lee, Y.-W. Improving logic obfuscation via logic cone analysis / Y.-W. Lee, N. Touba // Proc. Latin-American Test Symp., Puerto Vallarta, Mexico, 25–27 March 2015. – Puerto Vallarta, 2015. – P. 1–6.

11. Roy, J. A. Ending piracy of integrated circuits / J. A. Roy, F. Koushanfar, I. L. Markov // IEEE Computer. – 2010. – Vol. 43, no. 10. – P. 30–38.

12. Weighted logic locking: a new approach for IC piracy protection / N. Karousos [et al.] // IEEE 23rd Intern. Symp. on On-Line Testing and Robust System Design (IOLTS), Thessaloniki, Greece, 3–5 July 2017. – Thessaloniki, 2017. – P. 221–226.

13. Золоторевич, Л. А. Исследование методов и средств верификации проектов и генерации тестов МЭС / Л. А. Золоторевич // Сб. науч. тр. Всерос. науч.-техн. конф. «Проблемы разработки перспективных микроэлектронных систем» (МЭС–2006) / под общ. ред. А. Л. Стемпковского. – М. : ИППМ РАН, 2006. – С. 163–168.

14. Zolotorevich, L. A. Project verification and construction of superchip tests at the RTL level / L. A. Zolotorevich // Automation and Remote Control. – N. Y. : Plenum Press, 2013. – Vol. 74, iss. 1. – P. 113–122.

References

1. Zolotorevich L. A. *Hardware protection of digital devices*. Vestnik Tomskogo gosudarstvennogo universiteta. Upravlenie, vychislitel'naja tehnika, informatika [*Bulletin of Tomsk State University. Management, Computer Technology, Informatics*], 2020, no. 50, pp. 69–78. https://doi.org/10.17223/19988605/50/9 (In Russ.).

2. Subramanyan P., Ray S., Malik S. Evaluating the security of logic encryption algorithms. *2015 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), Washington, DC, USA, 5–7 May 2015*. Washington, 2015, pp. 137–143.

3. Rajendran J., Sam M., Sinanoglu O., Karri R. Security analysis of integrated circuit camouflaging. *ACM SIGSAC Conference on Computer & Communications Security, Berlin, Germany, 4–8 November 2013*. Berlin, 2013, pp. 709–720.

4. Roy J. A., Koushanfar F., Markov I. L. EPIC: Ending piracy of integrated circuits. *IEEE Computer*, 2010, vol. 43. no. 10, pp. 30–38.

5. Yasin M., Rajendran J., Sinanoglu O., Karri R. On improving the security of logic locking. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2016, vol. 35, no. 9, pp. 1411–1424.

6. Rajendran J., Pino Y., Sinanoglu O., Karri R. Logic encryption: a fault analysis perspective. *DATE '12: Proceedings of the Conference on Design, Automation and Test in Europe, Dresden, Germany, March, 2012*. Dresden, 2012, pp. 953–958.

7. Rajendran J., Zhang H., Zhang C., Rose G. S., Pino Y., …, Karri R. Fault analysis-based logic encryption. *IEEE Transactions on Computers*, 2015, vol. 64, no. 2, pp. 410–424.

8. Dupuis S., Ba P., Natale G. D., Flottes M., Rouzeyre B. A novel hardware logic encryption technique for thwarting illegal overproduction and hardware trojans. *20th IEEE International On-Line Testing Symposium, Platja d'Aro, Catalunya, Spain, 7–9 July 2014*. Platja d'Aro, 2014, pp. 49–54.

9. Plaza S. M., Markov I. L. Solving the third-shift problem in IC piracy with test-aware logic locking. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2015, vol. 34. no. 6, pp. 961–971.

10. Lee Y.-W., Touba N. Improving logic obfuscation via logic cone analysis. *Proceedings Latin-American Test Symposium, Puerto Vallarta, Mexico, 25–27 March 2015*. Puerto Vallarta, 2015, pp. 1–6.

11. Roy J. A., Koushanfar F., Markov I. L. Ending piracy of integrated circuits. *IEEE Computer*, 2010, vol. 43, no. 10, pp. 30–38.

12. Karousos N., Pexaras K., Karybali I. G., Kalligeros E. Weighted logic locking: a new approach for IC piracy protection. *IEEE 23rd International Symposium on On-Line Testing and Robust System Design (IOLTS), Thessaloniki, Greece, 3–5 July 2017*. Thessaloniki, 2017, pp. 221–226.

13. Zolotorevich L. A. Research of methods and means of project verification and test generation of MES. Sbornik nauchnyh trudov Vserossijskoj nauchno-tehnicheskoj konferencii "Problemy razrabotki perspektivnyh mikrojelektronnyh sistem" (MJeS–2006) [*Collection of Scientific Papers of the All-Russian Scientific and Technical Conference "Problems of Development of Promising Microelectronic Systems" (MES–2006)*]. In Stempkovskij A. L. (ed.). Moscow, Institut problem proektirovanija v mikrojelektronike Rossijskoj akademii nauk, 2006, pp. 163–168 (In Russ.).

14. Zolotorevich L. A. Project verification and construction of superchip tests at the RTL level. *Automation and Remote Control*. New York, Plenum Press, 2013, vol. 74, iss. 1, pp. 113–122.

Информация об авторах

Золоторевич Людмила Андреевна, кандидат технических наук, доцент, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: zolotorevichla@bsuir.by

Ильинков Валерий Андреевич, кандидат технических наук, доцент, Белорусский государственный университет информатики и радиоэлектроники.
E-mail: v.ilyinkov@gmail.com

Information about the authors

Lyudmila A. Zolotorevich, Cand. Sci. (Eng.), Assoc. Prof., Belarusian State University of Informatics and Radioelectronics.
E-mail: zolotorevichla@bsuir.by

Valery A. Ilyinkov, Cand. Sci. (Eng.), Assoc. Prof., Belarusian State University of Informatics and Radioelectronics.
E-mail: v.ilyinkov@gmail.com