Вячеслав Н. Ярмолик[1], Александр А. Иванюк[2]
Белорусский государственный университет информатики и радиоэлектроники,
ул. П. Бровки, 6, Минск, 220013, Беларусь
[1]e-mail: yarmolik10ru@yahoo.com, https://orcid.org/0000-0003-3995-1463
[2]e-mail: ivaniuk@bsuir.by, https://orcid.org/0000-0002-6541-7742

СБАЛАНСИРОВАННЫЕ ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫЕ ФУНКЦИИ ТИПА АРБИТР

DOI: http://dx.doi.org/10.26583/bit.2023.1.07

Аннотация. Решается задача построения нового класса физически неклонируемых функций типа арбитр (АФНФ), основанного на применении сбалансированных пар путей, что позволило существенно повысить стабильность, уникальность и единообразие АФНФ. Актуальность предлагаемого исследования связана с активным развитием физической криптографии, применяемой для целей идентификации электронных изделий и формирования криптографических ключей. Показано, что в классических АФНФ используется стандартный базовый элемент, который выполняет три функции, а именно, функцию генерирования задержки сигнала *Generate*, функцию выбора пары путей *Select* и функцию переключения путей *Switch*. Выполнение базовым элементом всех функций одновременно приводит к асимметрии пар путей, приводящей к ухудшению характеристик АФНФ, и предполагает выполнение балансировки путей. Как альтернатива стандартному базовому элементу в статье предлагаются две его модификации, в которых функция *Generate* выполняется на дополнительных линиях задержки, а функция *Switch* на мультиплексорах. Применение линий задержки со значениями времен задержки сигнала значительно больше, чем на мультиплексорах позволяет строить сбалансированные АФНФ, характеризующиеся высокой степенью симметрии. Предложенный подход построения сбалансированных АФНФ, основанный на применении модифицированных базовых элементов, показал свою работоспособность и перспективность, в том числе, при реализации АФНФ на программируемых структурах. Практические исследования проводились путем сравнительного анализа классической и сбалансированных АФНФ, реализованных на современных FPGA. Экспериментально подтвержден эффект улучшения характеристик нового класса ФНФ, и в первую очередь заметное улучшение стабильности, уникальности и единообразия АФНФ.

Ключевые слова: физически неклонируемые функции, физически неклонируемые функции типа арбитр, идентификация электронных изделий, стабильность, уникальность, единообразие
Для цитирования: ЯРМОЛИК, Вячеслав Н.; ИВАНЮК, Александр А. СБАЛАНСИРОВАННЫЕ ФИЗИЧЕСКИ НЕКЛОНИРУЕМЫЕ ФУНКЦИИ ТИПА АРБИТР. Безопасность информационных технологий, [S.l.], т. 30, № 1, с. 92–107, 2023. ISSN 2074-7136. URL: https://bit.spels.ru/index.php/bit/article/view/1481. DOI: http://dx.doi.org/10.26583/bit.2023.1.07.

Vyacheslav N. Yarmolik[1], Alexander A. Ivaniuk[2]
Belarusian State University of Informatics and Radioelectronics,
str. P. Brovki, 6, 220013, Minsk, Belarus
[1]e-mail: yarmolik10ru@yahoo.com, https://orcid.org/0000-0003-3995-1463
[2]e-mail: ivaniuk@bsuir.by, https://orcid.org/0000-0002-6541-7742

Balanced arbiter physical uncloneable functions

DOI: http://dx.doi.org/10.26583/bit.2023.1.07

Abstract. The problem of constructing a new class of physically unclonable functions of the arbiter type (APUF) based on the construction of balanced pairs of paths is solved. This makes it possible to significantly increase the stability, uniqueness and uniformity of the APUF. The relevance of the proposed research is associated with the active development of physical cryptography used for the identification of electronic devices and for the generation of cryptographic keys. It is shown that the classical APUF uses a standard basic element that performs three functions, namely, the function of

generating a signal delay Generate, the function of choosing a pair of paths Select and the function of switching paths Switch. The execution of all functions by the basic element simultaneously leads to the asymmetry of pairs of paths, which causes a deterioration in the characteristics of the APUF and entails balancing of the paths. As an alternative to the standard basic element, two of its modifications are proposed, in which the Generate function is performed on additional delay lines, and the Switch function on the multiplexer. The use of delay lines with signal delay times much longer than on multiplexers allows constructing the balanced APUF. The proposed approach for building balanced APUF, based on the use of modified basic elements, has demonstrated its efficiency and prospects, including the case of implementing APUF on programmable structures. Practical studies were carried out by a comparative analysis of the classical APUF and balanced APUF implemented on modern FPGAs. The effect of improvement of the characteristics of similar PUFs has been experimentally confirmed, and, first of all, a noticeable improvement in the stability, uniqueness and uniformity of APUFs.
Keywords: physical unclonable functions, arbiter physical unclonable functions, identification of electronic devices, stability, uniqueness, uniformity
For citation: YARMOLIK, Vyacheslav N.; IVANIUK, Alexander A. Balanced arbiter physical uncloneable functions. IT Security (Russia), [S.l.], v. 30, no. 1, p. 92–107, 2023. ISSN 2074-7136. URL: https://bit.spels.ru/index.php/bit/article/view/1481. DOI: http://dx.doi.org/10.26583/bit.2023.1.07.

Введение

Одним из способов идентификации и аутентификации цифровых устройств являются физически неклонируемые функции (ФНФ) (Physical Unclonable Functions – *PUF*), которые весьма эффективны для защиты электронных изделий от нелегального копирования [1–4]. В настоящее время ФНФ активно применяются в криптографии для целей генерирования криптографических ключей, а также реализации различных криптографических приложений и протоколов [5–7].

Физически неклонируемыми функциями являются физические системы, определяющим свойством которых является неклонируемость (Unclonability), то есть невозможность воспроизведения двух ФНФ, поведение которых будет идентичным. Подобные системы имеют свойство неклонируемости, так как состоят из множества компонент, параметры которых, в процессе создания подобных физических систем, принимают случайные значения. ФНФ описываются входными и соответствующими им выходными параметрами сигналов. Пара, состоящая из входного параметра запроса (Challenge – *C*) и выходного параметра ответа (Response – *R*), называется парой запрос-ответ (Challenge-Response Pair – *CRP*). В простейшем случае, ФНФ можно рассматривать как функцию $R = F(C)$, которая преобразует запросы *C* в ответы *R* [1, 2, 8–10].

Анализ большого числа исследований в области ФНФ [5–10] показывает, что, в общем случае, из всех характеристик, описывающих поведение ФНФ, на первом месте стоит *стабильность* (level of robustness). Затем идет *уникальность* (no two PUFs are the same), далее *простота технической реализации* (to be feasibly implemented), *неклонируемость* (PUF cannot be copied) и, наконец, *непредсказуемость* (randomness). Наиболее полно всем приведенным характеристикам отвечают ФНФ основанные на задержках распространения (delay based) электрических сигналов [8–13].

В настоящее время существует множество разнообразных реализаций ФНФ на основе задержек распространения тестового сигнала, среди которых лидирующую позицию занимают, так называемые, ФНФ типа арбитр (АФНФ (*APUF*)) [8, 10–14]. В общем случае, в АФНФ с помощью значения запроса *C* задается конфигурация, как правило, двух функционально и топологически симметричных путей, по которым распространяются идентичные копии тестового сигнала. Ответом *R* АФНФ является результат сравнения временных задержек распространения сигнала по двум путям [8–10]. Симметричность путей обеспечивает близкие значения величин задержек

распространения по ним сигналов, которые в силу технологических вариаций, имеющих случайный характер, в процессе производства будут иметь незначительные отличия. Пары симметричных путей для задержки электрического сигнала изготавливаются, таким образом, чтобы подобных пар было огромное множество, из которого по конкретному запросу C выбирается одна из них. Процедура измерения времени распространения сигнала заключается в одновременной подаче на входы обоих путей сигнала, и определении арбитром, на выходе которого из них сигнал появится быстрее.

Основные проблемы при создании АФНФ состоят в противоречивости требования высокой стабильности, которое характеризуется минимизацией метастабильных состояний, с непредсказуемостью, то есть случайностью таких функций [13]. В простейшем случае случайность оценивается метрикой единообразия (uniformity), которая определяет равновероятность появления ответов 0 и 1. Значение данной метрики меньше 1,0 свидетельствует о наличии асимметрии в генерируемых парах путей, что в особенности характерно для АФНФ, реализованных на программируемой логике (FPGA) [14, 15]. Как отмечается в ряде литературных источников, попытка увеличить стабильность ФНФ увеличивает их предсказуемость, и, соответственно, уязвимость для различного рода атак, в особенности, с применением современных достижений машинного обучения [17, 18]. Одним из наиболее эффективных методов увеличения стабильности АФНФ является их балансировка [15, 16]. Однако эта процедура, требующая дополнительных индивидуальных настроек АФНФ, технологически может быть сложной задачей, а в ряде случаев ASIC технологий, и невыполнимой. Более того, балансировка путей означает отход от основополагающей концепции ФНФ, заключающейся в использовании при изготовлении ФНФ их единого схемотехнического описания для получения отличающегося (неповторяемого) поведения ФНФ, описываемого уникальной функцией $R = F(C)$.

Таким образом, проблема построения эффективных АФНФ, как наиболее распространенной разновидности ФНФ, является практически открытой. В данной статье рассматривается задача построения сбалансированных АФНФ, которые характеризуются обеспечением высоких показателей их характеристик, таких как стабильность, уникальность и единообразие. Важным достоинством нового вида АФНФ является исключение процедуры балансировки путей из процесса изготовления АФНФ.

1. ФНФ типа арбитр

Классической схемой ФНФ типа арбитр является схема, приведенная на рис. 1 [2, 8–13]. Эта схема строится с использованием n последовательно подключенных пар двухвходовых мультиплексоров (MUX).

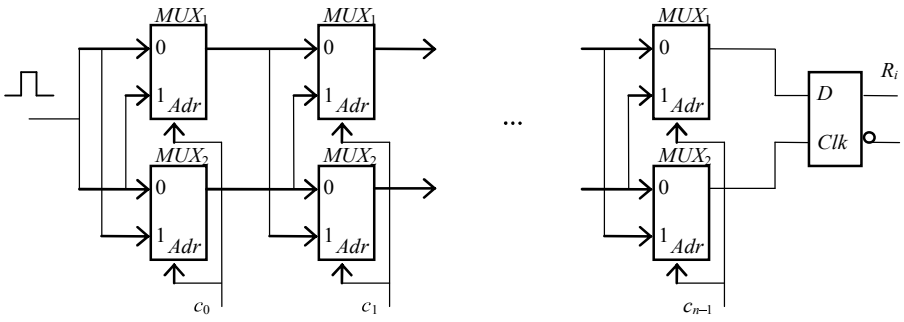


Рис. 1. ФНФ типа арбитр на базе двухвходовых мультиплексоров
Fig. 1. Arbiter-type PUF based on two-input multiplexers

Адресные входы (*Adr*) обоих мультиплексоров MUX_1 и MUX_2 каждой пары являются одним из входов для задания значения одного бита запроса c_j. Запрос в данном случае представляет собой n-разрядный двоичный вектор $C_i = c_0\, c_1\, c_2 \ldots\, c_{n-1}$, где $c_j \in \{0, 1\}$, $j \in \{0, 1, 2, \ldots, n-1\}$. Значение запроса C_i в схеме АФНФ (см. рис. 1) формирует два пути, таким образом, что если для j-ой ступени АФНФ $c_j = 0$, то для построения первого пути используется мультиплексор MUX_1, а для второго MUX_2, а при $c_j = 1$, наоборот. Каждая пара путей имеет общий вход, а выходы первого и второго пути подключены, соответственно, к D входу D-триггера и к его синхронизирующему входу *Clk*. Триггер типа D является арбитром, который перед проведением эксперимента устанавливается в исходное нулевое состояние. Конкретному запросу C_i соответствует ответ $R_i \in \{0, 1\}$, как результат эксперимента по определению по какому из путей выбранной запросом C_i пары путей – первому или второму, задержка входного импульсного сигнала меньше. Если по первому – то предположим, что $R_i = 1$, а если по второму – то $R_i = 0$. Количество пар путей с увеличением n растет экспоненциально, и равняется 2^n.

Все известные решения построения ФНФ, в том числе и АФНФ, основаны на том, что задержка по конкретному пути (элементу) имеет случайное значение, определяемое множеством факторов, влияющих на ее величину в процессе производства ФНФ. Однако у реальных ФНФ эти случайные задержки имеют неизменное и неуправляемое значение, исключая влияние внешних факторов (температуру, давление, электромагнитное излучение и др.), а также временную деградацию. Их неизменность, с одной стороны, обеспечивает стабильность функционирования ФНФ, а с другой стороны, открывает возможности для различного рода атак на ФНФ при их взломе [17].

Базируясь на детерминированном поведении АФНФ, чаще всего и строятся различные их математические модели. Наиболее распространенная модель основана на том, что каждая j-ая ступень АФНФ, состоящая из пары мультиплексоров MUX_1 и MUX_2, описывается двумя параметрами, а именно разностями задержек $\delta_{0,j}$ и $\delta_{1,j}$:

$$\delta_{0,j} = \Delta(0)_{1,j} - \Delta(0)_{2,j};\ \delta_{1,j} = \Delta(1)_{1,j} - \Delta(1)_{2,j}. \qquad (1)$$

Величина $\delta_{0,j}$ для j-ой ступени АФНФ определяется для $c_j = 0$ как добавленная разность задержек $\Delta(0)_{1,j}$ и $\Delta(0)_{2,j}$ прохождения сигнала по двум путям через MUX_1 и MUX_2, а $\delta_{1,j}$ – как разность $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ при $c_j = 1$. Численное значение $\Delta(0)_{1,j}$ определяет временную задержку прохождения сигнала с нулевого входа, обозначенного символом 0, для 1-го мультиплексора (MUX_1) j-ой ступени АФНФ на его выход, а $\Delta(0)_{2,j}$ – задержку на втором мультиплексоре (MUX_2). Величины $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ представляют собой задержки сигналов по единичным входам соответствующих мультиплексоров. Все четыре значения, а именно $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$, являются источниками непредсказуемости поведения АФНФ, представленной на рис. 1. Четыре величины задержек каждой ступени принимают случайные значения как результат влияния множества неуправляемых факторов при изготовлении АФНФ. В процессе функционирования АФНФ эти величины, в идеальном случае, имеют неизменные значения и участвуют в определении величин добавленной разности задержек $\delta_{0,j}$ и $\delta_{1,j}$ согласно (1). В этом случае базовый элемент выполняет функцию генерирования (*Generate*) добавленной разности задержек и эти задержки ($\delta_{0,j}$ и $\delta_{1,j}$) уникальны и непредсказуемы. Под неизменностью указанных величин принимаются такие изменения их значений, которые не нарушают повторяемость ответов для одного и того же запроса.

Если эти два параметра ($\delta_{0,j}$ и $\delta_{1,j}$) известны для каждой ступени АФНФ, то разница задержки для каждой пары путей может быть легко определена путем учета возможного эффекта переключения (*Switch*) на каждой ступени. Переключение одного пути

на j-ой ступени АФНФ с MUX_1 на MUX_2, а второго с MUX_2 на MUX_1 эквивалентно изменению знака разницы задержек сигналов пары путей на предыдущих ступенях АФНФ. Таким образом, разница задержки d_j после j-ступени может вычисляться рекурсивно в соответствии со следующим соотношением:

$$d_j = d_{j-1} \times (-1)^{c_j} + \delta_{c_j,j}. \quad (2)$$

Анализ приведенного выражения показывает, что ответ R_i на запрос C_i для АФНФ, приведенной на рис. 1, определяется знаком разницы d_{n-1} задержек импульсного тестового сигнала по выбранным путям в соответствии с запросом C_i. В рамках подобных моделей (2) описания АФНФ определяющим фактором являются две функции, а именно, функция *Select* и функция *Switch*, выполняемые базовым элементом. Функция *Select* определяет выбор одной из двух величин добавленной разности задержек $\delta_{0,j}$ или $\delta_{1,j}$, генерируемых базовым элементом (*Generate*), по значению бита запроса $c_j \in \{0, 1\}$. В идеальном случае подобная функция может обеспечить высокое качество АФНФ. Под идеальной ситуацией понимают обеспечение идентичности технологического процесса при изготовлении всех компонент АФНФ и симметричности, заключающейся в равенстве геометрических размеров и длин всех их межсоединений.

Имеющиеся технологические вариации, влияющие на свойства АФНФ, хорошо нивелируются функцией *Switch* базового элемента. В особенности это важно для реализации АФНФ на программируемой логике типа FPGA. В [14] показано, что наличие различного рода асимметричных аномалий особенно присуще реализациям АФНФ на FPGA. Показано, что в ряде случаев асимметрия задержки АФНФ, реализованной на FPGA, из-за асимметрии маршрутизации более чем в 10 раз выше, чем случайная ее вариация из-за особенностей производственного процесса [14].

В качестве примера рассмотрим реализацию $АФНФ_1$, состоящую из $n = 4$ ступеней. Для построения $АФНФ_1$ используется базовый элемент, описываемый соотношением (2), т.е. выполняющий три функции *Generate*, *Select* и *Switch.* На каждой j-ой, $j \in \{0, 1, 2, 3\}$, ступени $АФНФ_1$, в зависимости от значения c_j запроса C_i, формируются задержки распространения сигнала по выбранной паре путей. Соотношение этих задержек на каждой ступени определяется величиной их добавленной разности, а значение разности задержки сигнала по двум путям величиной d_j (2). Знак плюс либо минус значения d_3 разности задержек и определяет значение ответа $R_i \in \{0, 1\}$.

Функция $АФНФ_1$ приведена в качестве примера весьма неудачного аномального случая синтеза АФНФ, когда из-за вариаций производственного процесса задержки $\Delta(0)_{1,j}$ и $\Delta(1)_{1,j}$ мультиплексоров MUX_{1j} всех $n = 4$ ступеней $АФНФ_1$ оказались больше задержек $\Delta(0)_{2,j}$ и $\Delta(1)_{2,j}$ мультиплексоров MUX_{2j}. Соответственно все величины добавленной разности задержек $\delta_{0,j}$ и $\delta_{1,j}$, согласно (1), примут положительные значения. Более того, предположим, что на каждой ступени добавленные задержки одинаковы, т.е. $\delta_{0,0} = \delta_{0,1} = \delta_{0,2} = \delta_{0,3} = \delta_0$ и $\delta_{1,0} = \delta_{1,1} = \delta_{1,2} = \delta_{1,3} = \delta_1$. Отметим реальность таких аномальных ситуаций в технологических процессах изготовления подобных функций, в особенности, при реализации АФНФ на программируемых структурах [13–15].

Описание функционирования $АФНФ_1$ для $n = 4$ и определенных для нее величин добавленной разности задержек $\delta_{0,j} = \delta_0$ и $\delta_{1,j} = \delta_1$, $j \in \{0, 1, 2, 3\}$, приведены в табл. 1. Для каждого из 16 значений запроса C_i приводятся значения задержек на всех ступенях функции $АФНФ_1$. Знак задержки d_3 после четвертой ступени определяет значение ответа $R_i \in \{0, 1\}$, при этом учитывается ранее принятое допущение, что δ_0 и δ_1 принимают положительные значения задержек. Символ X в табл. 1 означает метастабильное

состояние АФНФ$_1$, которое относится к нежелательному ее поведению [13, 15]. Этот случай возникает для значений запросов C_i, ответом для которых является равенство нулю, либо близкое значение к нулю, задержки d_3.

Таблица 1. Описание функционирования АФНФ$_1$

i	C_i				АФНФ$_1$				
					$d_j = d_{j-1} \times (-1)^{c_j} + \delta_{c_j,j}$				
	c_0	c_1	c_2	c_3	d_0	d_1	d_2	d_3	R_i
0	0	0	0	0	δ_0	$2\delta_0$	$3\delta_0$	$4\delta_0$	1
1	0	0	0	1	δ_0	$2\delta_0$	$3\delta_0$	$-3\delta_0+\delta_1$	0
2	0	0	1	0	δ_0	$2\delta_0$	$-2\delta_0+\delta_1$	$-\delta_0+\delta_1$	X
3	0	0	1	1	δ_0	$2\delta_0$	$-2\delta_0+\delta_1$	$2\delta_0$	1
4	0	1	0	0	δ_0	$-\delta_0+\delta_1$	δ_1	$\delta_0+\delta_1$	1
5	0	1	0	1	δ_0	$-\delta_0+\delta_1$	δ_1	0	X
6	0	1	1	0	δ_0	$-\delta_0+\delta_1$	δ_0	$2\delta_0$	1
7	0	1	1	1	δ_0	$-\delta_0+\delta_1$	δ_0	$-\delta_0+\delta_1$	X
8	1	0	0	0	δ_1	$\delta_0+\delta_1$	$2\delta_0+\delta_1$	$3\delta_0+\delta_1$	1
9	1	0	0	1	δ_1	$\delta_0+\delta_1$	$2\delta_0+\delta_1$	$-2\delta_0$	0
10	1	0	1	0	δ_1	$\delta_0+\delta_1$	$-\delta_0$	0	X
11	1	0	1	1	δ_1	$\delta_0+\delta_1$	$-\delta_0$	$\delta_0+\delta_1$	1
12	1	1	0	0	δ_1	0	δ_0	$2\delta_0$	1
13	1	1	0	1	δ_1	0	δ_0	$-\delta_0+\delta_1$	X
14	1	1	1	0	δ_1	0	δ_1	$\delta_0+\delta_1$	1
15	1	1	1	1	δ_1	0	δ_1	0	X

Главный вывод, который можно сделать в результате анализа данных, приведенных в табл. 1, касается высокой эффективности классической АФНФ за счет того, что базовый элемент выполняет, в том числе, и функцию переключения (*Switch*) путей *j*-ой ступени АФНФ$_1$. Применение функции *Switch* позволяет нивелировать асимметрию задержек двух путей вызванную аномальными значениями временных характеристик элементов реализующих АФНФ, и в особенности асимметрию их межсоединений. Этот факт подтверждается примером АФНФ$_1$, которая несмотря на детерминированное отклонение задержек ее элементов, в принципе, может рассматриваться в качестве рабочей версии АФНФ.

В тоже время следует отметить, что поведение классических АФНФ также отличается от желаемого, и в особенности при их реализации на программируемой логике типа FPGA, что объясняется сложностью, а в большинстве случаев невозможностью обеспечения физической идентичности элементов и симметричности их межсоединений [13, 15]. Зачастую наблюдается абсолютная асимметрия, требующая дальнейшей балансировки путей, что относится к нежелательной, но вынужденной процедуре [15].

2. Сбалансированные АФНФ

Анализ функционирования классической реализации АФНФ показал, что каждый ее базовый элемент одновременно выполняет три функции, а именно, конкретный, *j*-ый базовый элемент реализует функции *Generate*, *Select* и *Switch*. Как показывалось ранее, весьма эффективной является функция *Switch*, которая необходима для нивелирования недостатков, присущих функции *Generate*. Так как только в

идеализированном случае функция *Generate* обеспечивает формирование случайных и независимых величин задержек, имеющих одинаковое вероятностное описание. Как показано в ряде работ [12–16], асимметричность двух путей АФНФ обусловлена, прежде всего, заведомой неуправляемостью автоматизированного построения межсоединений CLB и SLICE-блоков, а также уникальностью и неповторимостью значений задержек распространения сигналов через технологические компоненты FPGA на кристалле. Наличие асимметричности в АФНФ предполагает их балансировку, необходимую для обеспечения требуемого уровня основных их характеристик [15, 16].

Очевидным решением для построения сбалансированных АФНФ на основе FPGA, является использование базовых элементов, временные задержки на компонентах которых превышают задержки на межсоединениях АФНФ. Простейшим решением может быть включение по входам классического базового элемента АФНФ линий задержки, либо r последовательно подключенных логических элементов, с целью увеличения значений задержек по каждому из двух входов базового элемента. Исполнение базового элемента сбалансированной АФНФ приведено на рис. 2*a*. При этом по первому и второму входам базового элемента включены r последовательно соединенных повторителей $Delay_1$ и $Delay_2$, на которых реализуется задержка тестового сигнала на временные величины, определяемые типом и разновидностью повторителей, и технологическими особенностями их изготовления. Отметим, что обеспечение симметрии при изготовлении схем $Delay_1$ и $Delay_2$ в силу их регулярности и простоты является менее сложной задачей по сравнению с решением этой же задачи для мультиплексоров базового элемента.

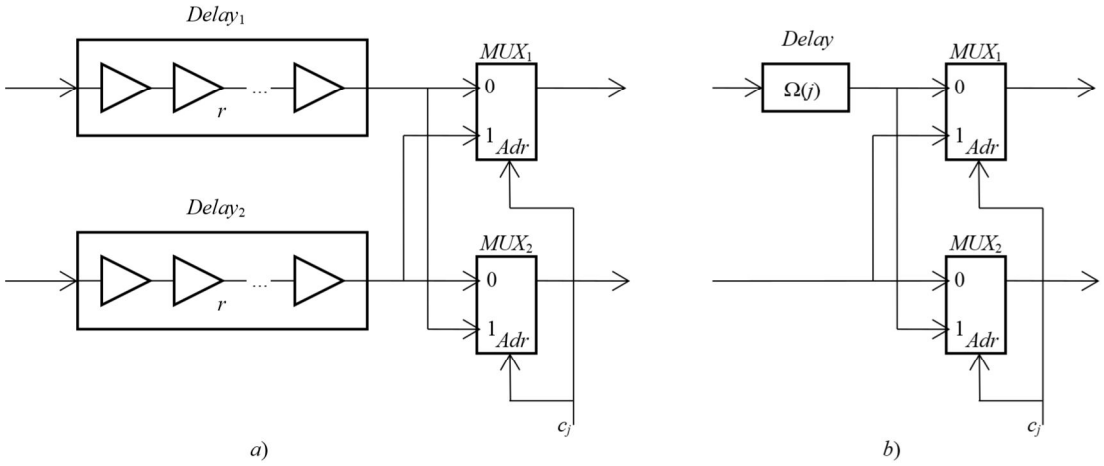


Рис. 2. Базовый элемент сбалансированной АФНФ
Fig. 2. The basic element of a balanced APUF

Задержка на каждом повторителе схем $Delay_1$ и $Delay_2$ представляет собой случайную величину Δ_l, где $l \in \{0, 1, \ldots, r-1\}$, которая описывается математическим ожиданием (средним значением) $\mu(\Delta_l)$ и дисперсией (отклонением значений) $Var(\Delta_l)$. Соответственно, величины задержек $\Omega_1(j)$ и $\Omega_2(j)$ последовательно соединенных повторителей, соответственно, схем $Delay_1$ и $Delay_2$, и их разброс по обоим входам базового элемента (рис. 2*a*) описываются выражениями:

$$\mu(\Omega_1(j)) = \mu(\Omega_2(j)) = \mu(\Delta_0 + \Delta_1 + \ldots + \Delta_{r-1}) = \mu(\Delta_0) + \mu(\Delta_1) + \ldots + \mu(\Delta_{r-1}) = r \times \mu(\Delta_l);$$
$$Var(\Omega_1(j)) = Var(\Omega_2(j)) = Var(\Delta_0 + \Delta_1 + \ldots + \Delta_{r-1}) = Var(\Delta_0) + Var(\Delta_1) + \ldots + Var(\Delta_{r-1}) = r \times Var(\Delta_l).$$

Приведенные соотношения справедливы для случайных независимых величин Δ_l, формируемых технологическим процессом изготовления повторителей, и они могут быть использованы для оценки задержек схем $Delay_1$ и $Delay_2$. Среднеквадратичное (стандартное) отклонение $\sigma = (r \times Var(\Delta_l))^{1/2}$ растет с ростом величины r, которое определяет диапазон 3σ изменения задержек $\Omega_1(j)$ и $\Omega_2(j)$ тестового сигнала. Таким образом, использование схем $Delay_1$ и $Delay_2$ в классическом базовом элементе позволяет нивелировать влияние задержек на межсоединениях и мультиплексорах. Это достигается тем, что величины задержки сигнала $\Omega_1(j)$ и $\Omega_2(j)$ на схемах $Delay_1$ и $Delay_2$ задаются существенно большими по сравнению с задержками на мультиплексорах $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$, и межсоединениях между ними. Более того, количество r элементов в линиях задержки $Delay_1$ и $Delay_2$ может быть произвольным и по необходимости увеличиваться для обеспечения требуемых соотношений задержек и отклонений тестовых сигналов.

В предложенной структуре базового элемента (рис. 2*а*) мультиплексоры выполняют только одну функцию, а именно *Switch*, так как величины задержек $\Omega_1(j)$ и $\Omega_2(j)$ на схемах $Delay_1$ и $Delay_2$ являются доминирующими. Соответственно, функция *Generate* выполняется на схемах $Delay_1$ и $Delay_2$, результатом которой является одна случайная величина $\delta_{0,j}$, определяемая разностью $\Omega_1(j) - \Omega_2(j)$, и имеющая знак + при $c_j = 0$. При $c_j = 1$ разность задержек схем $Delay_1$ и $Delay_2$ будет такой же, но изменит знак + на –, т.е. $\delta_{1,j} = -\delta_{0,j} = \Omega_2(j) - \Omega_1(j)$. С учетом ранее принятого ограничения, что разность задержки $\delta_{0,j} = \Omega_1(j) - \Omega_2(j)$ существенно больше задержек на мультиплексорах и их межсоединениях, соответственно, их величинами можно пренебречь. Тогда описание (2) функционирования j-го базового элемента АФНФ примет вид:

$$d_j = (d_{j-1} + \delta_{0,j}) \times (-1)^{c_j}. \tag{3}$$

Суммарное значение разности задержек d_{n-1}, по выбранной запросом C_i паре путей, а именно, его знак плюс, либо минус и определяет ответ R_i на запрос C_i. Выражение для вычисления d_{n-1} имеет вид:

$$d_{n-1} = \sum_{j=0}^{n-1} (\delta_{0,j} \times \prod_{k=j}^{n-1} (-1)^{c_k}) = \sum_{j=0}^{n-1} (\delta_{0,j} \times \prod_{k=j}^{n-1} (1-2\times c_k)) = \sum_{j=0}^{n-1} (\delta_{0,j} \times (1-2\times \bigoplus_{k=j}^{n-1} c_k)). \tag{4}$$

Соотношение (4) представляет собой три различных формулы для вычисления величины d_{n-1}, в которых используются арифметические операции +, – и ×, а также логическая операция сложения по модулю два $\oplus$. Значение разности задержки $\delta_{0,j}$ каждой ступени сбалансированной АФНФ входит со знаком плюс либо минус в выражение (4) в зависимости от запроса C_i. Приведенные три выражения (4) отличаются друг от друга формулами вычисления знака величины $\delta_{0,j}$. Значение d_{n-1}, например, для $n = 4$ и $C_i = c_0\, c_1\, c_2\, c_3 = 1001$ вычисляется с применением первой формулы (4) следующим образом:

$$\begin{aligned} d_3 &= \delta_{0,0} \times (-1)^{c_0} \times (-1)^{c_1} \times (-1)^{c_2} \times (-1)^{c_3} + \delta_{0,1} \times (-1)^{c_1} \times (-1)^{c_2} \times (-1)^{c_3} + \delta_{0,2} \times (-1)^{c_2} \times (-1)^{c_3} + \\ &+ \delta_{0,3} \times (-1)^{c_3} = \delta_{0,0} \times (-1)^1 \times (-1)^0 \times (-1)^0 \times (-1)^1 + \delta_{0,1} \times (-1)^0 \times (-1)^0 \times (-1)^1 + \delta_{0,2} \times (-1)^0 \times (-1)^1 + \\ &+ \delta_{0,3} \times (-1)^1 = \delta_{0,0} - \delta_{0,1} - \delta_{0,2} - \delta_{0,3}. \end{aligned} \tag{5}$$

В зависимости от абсолютных значений и знаков случайных величин $\delta_{0,0}$, $\delta_{0,1}$, $\delta_{0,2}$ и $\delta_{0,3}$, ответом R_i на запрос $C_i = 1001$ будет значения знака d_3 (5). Применив иное значение запроса, например, $C_i = c_0\, c_1\, c_2\, c_3 = 0101$, получим другие значения знаков (+, –) перед разностями задержки $\delta_{0,j}$ (5), а именно, $d_3 = \delta_{0,0} + \delta_{0,1} - \delta_{0,2} - \delta_{0,3}$.

Подробное описание поведения АФНФ$_2$ состоявшего из $n = 4$ базовых элементов, приведенных на рис. 2*a*, для случая равенства всех задержек $\delta_{0,0} = \delta_{0,1} = \delta_{0,2} = \delta_{0,3} = \delta_0$ и, соответственно, $\delta_{1,0} = \delta_{1,1} = \delta_{1,2} = \delta_{1,3} = -\delta_0$, приведено в табл. 2. Отметим, что данный аномальный случай рассматривался и для классической АФНФ$_1$ (см. табл. 1).

Таблица 2. Описание функционирования АФНФ$_2$

i	C_i				АФНФ$_2$ $d_j = (d_{j-1} + \delta_{0,j}) \times (-1)^{c_j}$				R_i АФНФ$_2$	R_i АФНФ$_1$
	c_0	c_1	c_2	c_3	d_0	d_1	d_2	d_3		
0	0	0	0	0	δ_0	$2\delta_0$	$3\delta_0$	$4\delta_0$	1	1
1	0	0	0	1	δ_0	$2\delta_0$	$3\delta_0$	$-4\delta_0$	0	0
2	0	0	1	0	δ_0	$2\delta_0$	$-3\delta_0$	$-2\delta_0$	0	X
3	0	0	1	1	δ_0	$2\delta_0$	$-3\delta_0$	$2\delta_0$	1	1
4	0	1	0	0	δ_0	$-2\delta_0$	$-\delta_0$	0	X	1
5	0	1	0	1	δ_0	$-2\delta_0$	$-\delta_0$	0	X	X
6	0	1	1	0	δ_0	$-2\delta_0$	δ_0	$2\delta_0$	1	1
7	0	1	1	1	δ_0	$-2\delta_0$	δ_0	$-2\delta_0$	0	X
8	1	0	0	0	$-\delta_0$	0	δ_0	$2\delta_0$	1	1
9	1	0	0	1	$-\delta_0$	0	δ_0	$-2\delta_0$	0	0
10	1	0	1	0	$-\delta_0$	0	$-\delta_0$	0	X	X
11	1	0	1	1	$-\delta_0$	0	$-\delta_0$	0	X	1
12	1	1	0	0	$-\delta_0$	0	δ_0	$2\delta_0$	1	1
13	1	1	0	1	$-\delta_0$	0	δ_0	$-2\delta_0$	0	X
14	1	1	1	0	$-\delta_0$	0	$-\delta_1$	0	X	1
15	1	1	1	1	$-\delta_0$	0	$-\delta_1$	0	X	X

Последний столбец табл. 2 повторяет данные об ответах R_i АФНФ$_1$, рассмотренной в табл. 1 и показывает, что в случае АФНФ$_2$ достигается абсолютная равновероятность (uniformity) значений 0 и 1 при том же количестве метастабильных состояний.

Вторая модификация базового элемента (см. рис. 2*b*) включает одну линию задержки *Delay*, описываемую величиной задержки $\Omega(j) = \Omega + \delta(j)$, где Ω представляет собой среднее значение этой величины, а $\delta(j)$ отклонение от среднего значения на j-м базовом элементе. Величина $\delta(j) << \Omega$ и ее знак являются источником случайности j-го базового элемента. В рассматриваемом варианте базового элемента разность задержки на нем будет определяться только задержкой $\Omega(j)$ линии задержки *Delay*. Здесь предполагается, что, как и в предыдущем случае (см. рис. 2*a*), задержки сигнала $\Delta(0)_{1,j}$, $\Delta(0)_{2,j}$, $\Delta(1)_{1,j}$ и $\Delta(1)_{2,j}$ на мультиплексорах существенно меньше величины $\Omega(j)$. Тогда, в зависимости от значения бита запроса c_j, разность задержки будет определяться как $\delta_{0,j} = \Omega(j)$ или $\delta_{1,j} = -\Omega(j)$.

Последовательное соединение базовых элементов, приведенных на рис. 2*b*, представляет собой потенциальную структуру АФНФ, в которой функция *Generate* выполняется схемами *Delay*, а функция *Switch* на мультиплексорах базовых элементов. Подобная схема АФНФ также описывается соотношением (4) в котором вместо значений $\delta_{0,j}$ используются $\Omega(j)$. Если предположить, что для всех ступеней подобной АФНФ $\delta(j) = 0$, то в зависимости от запроса выходная разность задержки, формирующая ответ R_i, будет определяться величиной $w \times \Omega$, где $w \in \{0, 1, 2, \ldots, n\}$, имеющей знак плюс или минус. В этом случае поведение подобной АФНФ описывается известной классической *Урновой*

схемой, выполняющей выбор Ω или –Ω с возвращением и без учета порядка, т.е. номера ступени *j*. Очевидно, что для всех значений $w \neq 0$ подобная схема не может быть использована в качестве АФНФ в силу однозначной предсказуемости ответа R_i по ненулевой величине $w \times \Omega$ и ее знаку, что также справедливо и для случая когда $\delta(j) \neq 0$, но $\delta(j) << \Omega$.

При выполнении равенства $w = 0$, АФНФ, построенная на базовых элементах, представленных на рис. 2*b*, может рассматриваться как работоспособный вариант, так как сумма детерминированных составляющих Ω и –Ω равняется нулю, а ответ *R* будет определяться величинами $\delta(j)$ каждой ступени. Значения $\delta(j)$ имеют случайную природу и принимают случайные величины $\delta(j) << \Omega$ с произвольным знаком. Необходимо отметить, что в этом случае в выражении (4) будут использоваться величины $\delta(j)$, ровно половина из которых будет иметь знак плюс, а вторая половина знак минус, что следует из равенства $w = 0$ и определяется запросом C_i. Если представить знаки величин $\delta(j)$ в указанном выражении (4) в виде вектора $B_i = b_0\, b_1\, b_2 \ldots b_{n-1}$, где $b_j \in \{1, -1\}$, $j \in \{0, 1, 2, \ldots, n-1\}$, то соотношение определяющее зависимость B_i от C_i и его последовательные преобразования имеют следующий вид:

$$b_j = (1 - 2 \times \bigoplus_{k=j}^{n-1} c_k); \quad 1 - b_j = 2 \times \bigoplus_{k=j}^{n-1} c_k. \tag{6}$$

Принимая во внимание, что $b_j \in \{1, -1\}$, можно заключить, что $(1 - b_j)/2$ равняется 0 для $b_j = 1$ и 1 для $b_j = -1$. Отсюда следует, что для обеспечения знака +, т.е. значения $b_j = 1$, необходимо, чтобы для значений элементов запроса C_i выполнялось условие $\bigoplus_{k=j}^{n-1} c_k = 0$, а для обеспечения $b_j = -1$, это условие представляется как $\bigoplus_{k=j}^{n-1} c_k = 1$.

Таким образом, задачу вычисления сбалансированного запроса C_i можно свести к задаче решения системы из *n* логических уравнений (7). Под сбалансированным запросом будем понимать входной запрос C_i, который обеспечивает равное количество символов 1 и –1 в векторе $B_i = b_0\, b_1\, b_2 \ldots b_{n-1}$:

$$\begin{aligned} & b_0 = c_0 \oplus c_1 \oplus c_2 \oplus \ldots \oplus c_{n-2} \oplus c_{n-1}; \\ & b_1 = c_1 \oplus c_2 \oplus c_3 \oplus \ldots \oplus c_{n-2} \oplus c_{n-1}; \\ & \ldots \\ & b_{n-2} = c_{n-2} \oplus c_{n-1}; \\ & b_{n-1} = c_{n-1}. \end{aligned} \tag{7}$$

Исходными данными системы (7) является вектор $B_i = b_0\, b_1\, b_2 \ldots b_{n-1}$ с указанными ранее ограничениями на соотношение значений 1 и –1 его компонент. В приведенной системе логических уравнений символы 1 и –1 знаков b_j заменяются логическими значениями 0 и 1, соответственно, $b_j = 1$ заменяется на логический ноль, а $b_j = -1$ на логическую единицу. Приведенная система из *n* уравнений имеет *n* неизвестных, которыми являются биты запроса C_i и для нее существует единственное решение.

В качестве примера рассмотрим вектор, состоящий из четырех знаков $B_i = b_0\, b_1\, b_2\, b_3 = 1\ 1\ -1\ -1$, который преобразуется в двоичный вектор $B_i^* = b_0\, b_1\, b_2\, b_3 = 0\ 0\ 1\ 1$. Подставив значения вектора B_i^* в систему уравнений (7) получим:

$$0 = c_0 \oplus c_1 \oplus c_2 \oplus c_3;$$
$$0 = c_1 \oplus c_2 \oplus c_3;$$
$$1 = c_2 \oplus c_3;$$
$$1 = c_3.$$

В результате, значение сбалансированного запроса $C_i = c_0\, c_1\, c_2\, c_3 = 0\ 1\ 0\ 1$. Важно отметить, что для любого вектора B_i^* существует запрос C_i, и, наоборот. Все множество сбалансированных запросов C_i, полученных на основании B_i^*, для $n = 4$ показаны в табл. 3.

Таблица 3. Значения сбалансированных запросов C_i для n = 4

C_i	0 1 0 1	1 1 1 1	1 0 1 0	1 0 1 1	1 1 1 0	0 1 0 0
B_i^*	0 0 1 1	0 1 0 1	0 1 1 0	1 0 0 1	1 0 1 0	1 1 0 0

Необходимость выполнения условия $w = 0$, для которого соблюдается равенство количества знаков, накладывает ограничение на значение n, которое должно быть четным. Тогда множество сбалансированных запросов C_i генерирующих половину знаков плюс, а вторую минус, в выражении (4), и описываемых вектором B_i^*, в процентном отношении можно оценить выражением, приведенным в первом столбце табл. 4, и его численными значениями для различных величин n, в последующих столбцах.

Таблица 4. Оценка и численные значения количества сбалансированных запросов C_i

n	8	16	24	32	40	48	56	64
$\binom{n}{n/2} \times \frac{1}{2^n} \times 100\%$	27,3%	19,6%	16,1%	13,9%	12,5%	11,5%	10,6	9,9%

Как видно из приведенной табл. 4, количество сбалансированных запросов C_i для их реальных размерностей n, достаточно велико. Это позволяет сделать вывод о возможности построения сбалансированных АФНФ на базовом элементе, приведенном на рис. 2*b*, с применением для них сбалансированных запросов.

3. Экспериментальные исследования сбалансированных АФНФ

Для подтверждения эффективности предложенных в статье новых решений по построению АФНФ был проведен ряд экспериментов на программируемых логических интегральных схемах FPGA Xilinx Zynq7, входящих в состав плат быстрого прототипирования цифровых устройств Digilent Zybo Z7-10. Реализовывались четыре идентичных экземпляра классической схемы АФНФ и четыре экземпляра сбалансированной схемы АФНФ с применением базовых элементов, представленных на рис. 2 для $n = 32$. На рис. 3 приведен пример реализации базового элемента (рис. 2*a*) для $АФНФ_2$ в терминах технологических примитивов LUT-блоков FPGA.

Исследовались временные параметры множества пар путей, а именно, значения d_{31} для различных запросов, выраженных как $\Delta_{C_i}^f$, где $f \in [0,3]$ есть индекс экземпляра одной из четырех исследуемых схем АФНФ, реализованных на одном кристалле FPGA, отсортированные результаты которых приведены на рис. 4.

Значения математического ожидания $\mu(\Delta_{C_i}^f)$ (см. табл. 5) для 10^4 различных запросов, полученных с помощью 32-разрядого генератора М-последовательностей,

показывают асимметрию во всех четырех реализациях классической АФНФ, в то время как для реализаций сбалансированной АФНФ$_2$, свидетельствуют об их большей симметричности.

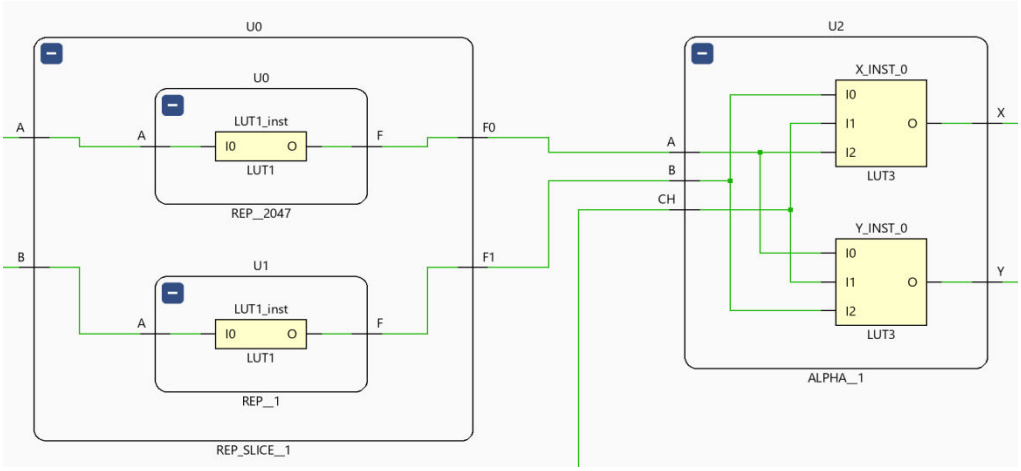


Рис. 3. Базовый элемент (Рис. 2, а) для схемы АФНФ$_2$
Fig. 3. Basic element (Fig. 2, a) for APUF$_2$

Таблица 5. Математическое ожидание $\mu(\Delta_{C_i}^f)$

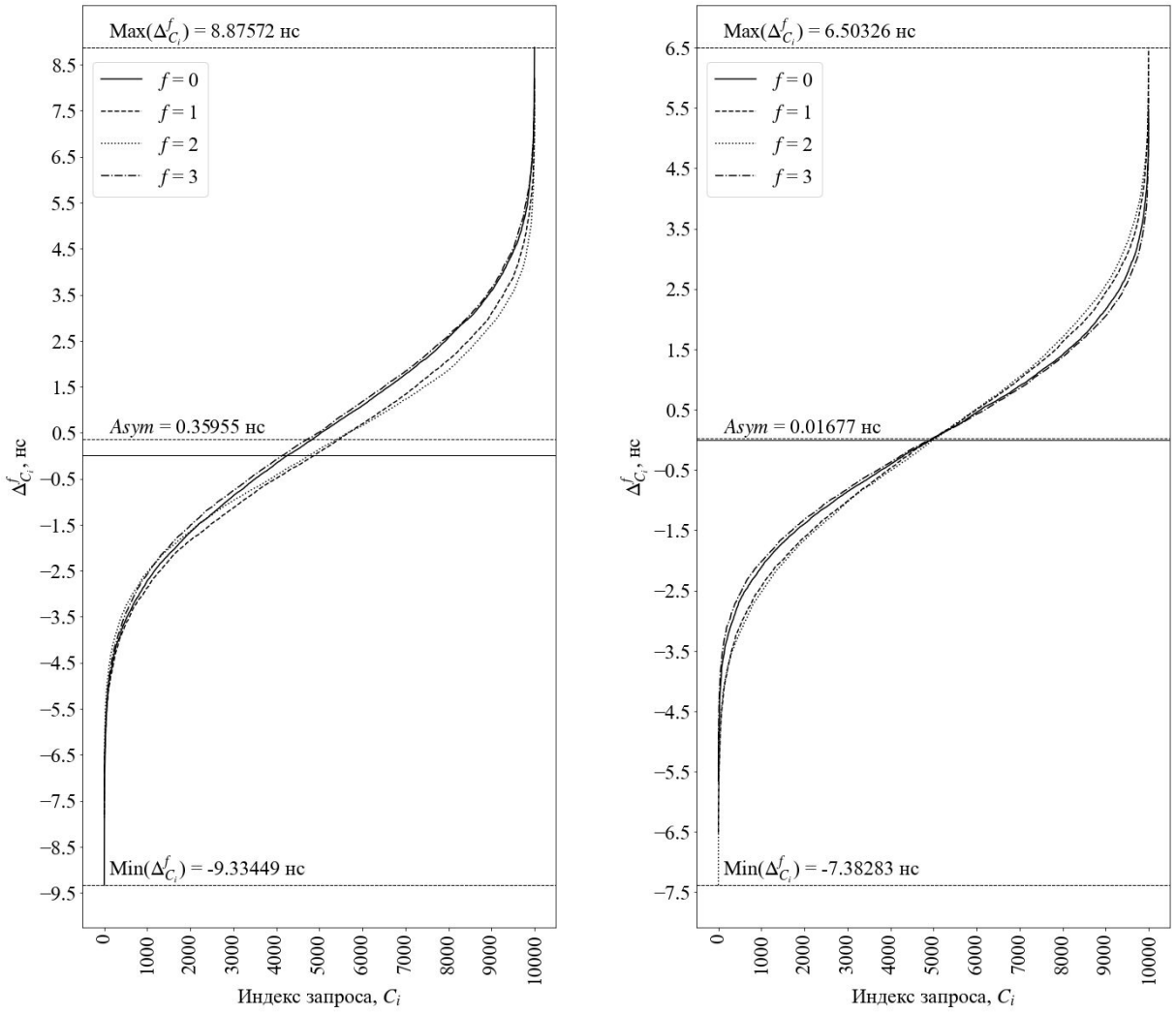
f	Классическая АФНФ, нс (*ns*)	Сбалансированная АФНФ$_2$, нс (*ns*)
0	0,4434	0,0188
1	0,1082	-0,0003
2	0,1289	0,0134
3	0,5406	0,0242

Для более детального сравнения реализованных схем АФНФ были определены их основные характеристики, а именно, стабильность (*St*), единообразие (*Un*) и внутрикристальная уникальность (U_{intra}) [13, 15]. Значения единообразия и внутрикристальной уникальность для сбалансированных АФНФ$_2$ заметно превышают аналогичные значения для классической АФНФ, а показатели стабильности являются сравнимыми (см. табл. 6).

Таблица 6. Усредненные значения St, Un, и U_{intra}

Тип базового элемента	*St*	*Un*	U_{intra}
Классическая схема базового элемента АФНФ	0,9920	0,8991	0,7675
Базовый элемент АФНФ$_2$	0,9945	0,9897	0,8982

Кроме того, на рис. 4 приведены значения метрики *Asym*, которая представляет собой среднеквадратичное значение $\sqrt{\sum_{f=0}^{3}\mu^2(\Delta_{C_i}^f)}$, определяющая степень асимметрии множеств нулевых и единичных ответов всех четырех экземпляров схем АФНФ. Так, полученные значения *Asym* для схем АФНФ$_2$ (0,0168 нс) существенно меньше аналогичного значения (0,3596 нс) для классической АФНФ, что подтверждается вычисленными характеристиками единообразия *Un* (см. табл. 6).



а) *б*)

Рис. 4. Значения величин задержек $\Delta^f_{C_i}$ для классической схемы АФНФ (а) и для АФНФ₂ (б)

Fig. 4. Delay values $\Delta^f_{C_i}$ for classical APUF (a) and for APUF₂ (b) schemes

Из данных, приведенных на рис. 4, также можно наблюдать различие в диапазонах изменения значений $\Delta^f_{C_i}$ для двух типов схем АФНФ. Так, для классической схемы АФНФ диапазон равен [-9,3345; 8,8757] нс, а для схемы АФНФ$_2$ – [-7,3828; 6,5033] нс. Отметим, что границы диапазонов выбирались как минимальное и максимальное значение соответственно для всех четырех экземпляров исследуемых схем. Уменьшение диапазона значений $\Delta^f_{C_i}$ для АФНФ$_2$ обусловлено компактным расположением LUT-блоков каждого базового элемента схем в пределах одного SLICE-блока кристалла FPGA (см. рис. 3), что в свою очередь уменьшило протяженность наиболее длинных межсоединений в сравнении с классической схемой АФНФ, а это повлияло на уменьшение результирующей разницы задержек распространения сигналов.

Асимметрия множеств значений $\Delta^f_{C_i}$ наблюдается и на результатах графического теста «Распределение на плоскости», результаты которого приведены на рис. 5 и коррелируются со значениями из табл. 5.

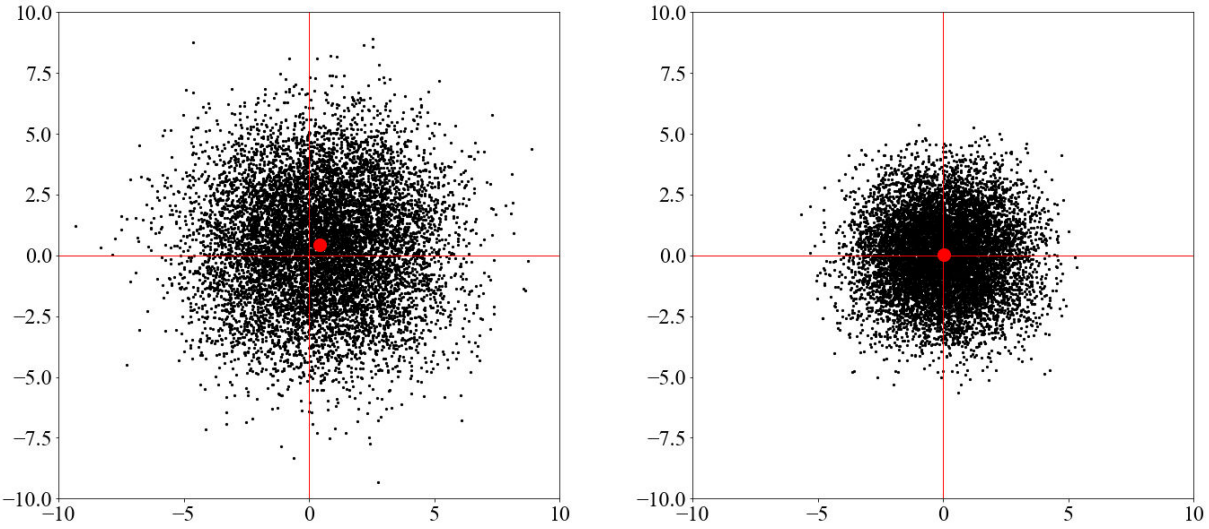


Рис. 5. Графический тест для классической схемы АФНФ (а) и для АФНФ₂ (б) (f=0)
Fig. 5. Graphical test for classical APUF (a) and for APUF₂ (b) schemes (f=0)

Заключение

В работе представлен новый класс физически неклонируемых функций типа арбитр АФНФ, основанных на применении сбалансированных пар путей. Приведено обоснование, как необходимости балансировки пар путей, так и сама их процедура с использованием для этих целей линий задержки. Предлагаются математические модели описания функционирования подобных АФНФ, вводится понятие сбалансированных запросов и алгоритм их определения. Экспериментально подтвержден эффект улучшения характеристик нового класса ФНФ, по сравнению с классической АФНФ, которые реализовались на программируемых логических интегральных схемах FPGA Xilinx Zynq7. Отмечено заметное улучшение стабильности, уникальности и единообразия сбалансированных АФНФ. Интересным представляется дальнейшее исследование сбалансированных АФНФ реализованных на других типах FPGA и как ASIC с различными технологическими нормами и особенностями.

СПИСОК ЛИТЕРАТУРЫ:

1. Suh G.E., Devadas S. 2007. Physical unclonable functions for device authentication and secret key generation. In Proceedings of the 44th annual Design Automation Conference (DAC '07). Association for Computing Machinery, New York, NY, USA, 9–14. DOI: https://doi.org/10.1145/1278480.1278484.
2. Rührmair U., Busch H., Katzenbeisser S. Strong PUFs: models, constructions, and security proofs. Towards Hardware-Intrinsic Security. Editors: A.-R. Sadeghi, D. Naccache. Berlin, Heidelberg: Springer Berlin Heidelberg. 2010, p. 79–96. DOI: https://doi.org/10.1007/978-3-642-14452-3.
3. Суханов С.В. Анализ физически неклонируемых функций на основе элементов памяти. Безопасность информационных технологий, [S.l.], т. 22, № 1, 2015. ISSN 2074-7136. URL: https://bit.mephi.ru/index.php/bit/article/view/130 (дата обращения: 02.02.2023).
4. Дураковский Анатолий П.; Кессаринский Леонид Н.; Ширин Алексей О. Маркировка и проверка подлинности изделий микроэлектроники на основе неклонируемости радиационного поведения. Безопасность информационных технологий, [S.l.], т. 27, № 3, с. 18–25. 2020. DOI: http://dx.doi.org/10.26583/bit.2020.3.02. – EDN: OJDTLM.
5. Škorić B., Tuyls P., Ophey W. (2005). Robust Key Extraction from Physical Uncloneable Functions. In: Ioannidis, J., Keromytis, A., Yung, M. (eds) Applied Cryptography and Network Security. ACNS 2005. Lecture Notes in Computer Science, vol 3531. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/11496137_28.

6. Lee J.W., Lim D., Gassend B., Suh T.G., Dijk M.V., Devadas S. A technique to build a secret key in integrated circuits for identification and authentication applications. 2004 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No.04CH37525), Honolulu, HI, USA. 2004, p. 176–179. DOI: https://doi.org/10.1109/VLSIC.2004.1346548.
7. Lim D., Lee J.W., Gassend B., Suh T.G., Dijk M.V., Devadas S. Extracting secret keys from integrated circuits. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. Oct. 2005, vol. 13, no. 10, p. 1200–1205. DOI: https://doi.org/10.1109/TVLSI.2005.859470.
8. Ярмолик В.Н., Вашинко Ю.Г. Физически неклонируемые функции. Информатика. 2011. Т. 30, № 2. С. 92–103. URL: https://inf.grid.by/jour/article/view/370 (дата обращения: 02.02.2023).
9. Rührmair U., Sölter J., Sehnke F. On the Foundations of Physical Unclonable Functions. IACR Cryptology. ePrint Archive. Paper 2009/277. 2009. – 20 p. URL: https://eprint.iacr.org/2009/277 (дата обращения: 02.02.2023).
10. Иванюк А.А., Заливако С.С. Физическая криптография и защита цифровых устройств. Доклады БГУИР. 2019, т. 120, № 2, с. 50–58. URL: https://libeldoc.bsuir.by/handle/123456789/34706 (дата обращения: 02.02.2023).
11. Xu X., Burleson W., Holcomb D. E. Using Statistical Models to Improve the Reliability of Delay-Based PUFs. 2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Pittsburgh, PA, USA. 2016, p. 547–552, DOI: https://doi.org/10.1109/ISVLSI.2016.125.
12. Ярмолик В.Н., Иванюк А.А., Шинкевич Н.Н. Физически неклонируемые функции с управляемой задержкой распространения сигналов. Информатика. 2022, т. 19, № 1, с. 32–49. DOI: https://doi.org/10.37661/1816-0301-2021-19-1-32-49.
13. Клыбик В.П., Заливако С.С., Иванюк А.А. Метод увеличения стабильности физически неклонируемой функции типа «Арбитр». Информатика. 2017, т. 53, № 1, с. 31–43. DOI: https://doi.org/10.37661/1816-0301-2021-19-1-32-49. – EDN: YLJOUJ.
14. Morozov S., Maiti A., Schaumont P. An Analysis of Delay Based PUF Implementations on FPGA. In: Sirisuk, P., Morgan, F., El-Ghazawi, T., Amano, H. (eds) Reconfigurable Computing: Architectures, Tools and Applications. ARC 2010. Lecture Notes in Computer Science, vol 5992. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/978-3-642-12133-3_37.
15. Шамына А.Ю., Иванюк А.А. Построение и балансировка путей физически неклонируемой функции типа арбитр на FPGA. Информатика. 2022, т. 19, № 4, с. 27–41. DOI: https://doi.org/10.37661/1816-0301-2022-19-4-27-41.
16. Ярмолик В.Н., Иванюк А. Физически неклонируемые функции типа арбитр с заведомо асимметричными парами путей. Доклады БГУИР. 2022, т. 20, № 4, с. 71–79. DOI: https://doi.org/10.35596/1729-7648-2022-20-4-71-79.
17. Delvaux J., Verbauwhede I. Side channel modeling attacks on 65nm arbiter PUFs exploiting CMOS device noise. IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), Austin, TX, USA. 2013, p. 137–142. DOI: https://doi.org/10.1109/HST.2013.6581579.
18. Rührmair U. et al. PUF Modeling Attacks on Simulated and Silicon Data. IEEE Transactions on Information Forensics and Security. Nov. 2013, vol. 8, no. 11, p. 1876–1891. DOI: https://doi.org/10.1109/TIFS.2013.2279798.

REFERENCES:

[1] Suh G.E., Devadas S. 2007. Physical unclonable functions for device authentication and secret key generation. In Proceedings of the 44th annual Design Automation Conference (DAC '07). Association for Computing Machinery, New York, NY, USA, 9–14. DOI: https://doi.org/10.1145/1278480.1278484.
[2] Rührmair U., Busch H., Katzenbeisser S. Strong PUFs: models, constructions, and security proofs. Towards Hardware-Intrinsic Security. Editors: A.-R. Sadeghi, D. Naccache. Berlin, Heidelberg: Springer Berlin Heidelberg. 2010, p. 79–96. DOI: https://doi.org/10.1007/978-3-642-14452-3.
[3] Sukhanov S.V. Analysis of Physical Unclonable Functions Based on Memory. IT Security (Russia), [S.l.], v. 22, no. 1, 2015. ISSN 2074-7136. URL: https://bit.mephi.ru/index.php/bit/article/view/130 (accessed: 02.02.2023) (in Russian).
[4] Durakovskiy Anatoly P.; Kessarinskiy Leonid N.; Shirin Alexey O. The use of microelectronics radiation behavior as physical uncloned function to find counterfeit. IT Security (Russia), [S.l.], v. 27, no. 3, p. 18–25. DOI: http://dx.doi.org/10.26583/bit.2020.3.02 (in Russian). – EDN: OJDTLM.
[5] Skoric B Škorić B., Tuyls P., Ophey W. (2005). Robust Key Extraction from Physical Uncloneable Functions. In: Ioannidis, J., Keromytis, A., Yung, M. (eds) Applied Cryptography and Network Security. ACNS 2005. Lecture Notes in Computer Science, vol 3531. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/11496137_28.

[6] Lee J.W., Lim D., Gassend B., Suh T.G., Dijk M.V., Devadas S. A technique to build a secret key in integrated circuits for identification and authentication applications. 2004 Symposium on VLSI Circuits. Digest of Technical Papers (IEEE Cat. No.04CH37525), Honolulu, HI, USA. 2004, p. 176–179. DOI: https://doi.org/10.1109/VLSIC.2004.1346548.

[7] Lim D., Lee J.W., Gassend B., Suh T.G., Dijk M.V., Devadas S. Extracting secret keys from integrated circuits. IEEE Transactions on Very Large Scale Integration (VLSI) Systems. Oct. 2005, vol. 13, no. 10, p. 1200–1205. DOI: https://doi.org/10.1109/TVLSI.2005.859470.

[8] Yarmolik V.N., Vashinko Y.G. Physical unclonable functions. Informatics. 2011, v. 30, no. 2, p. 92–103. URL: https://inf.grid.by/jour/article/view/370 (accessed: 02.02.2023) (in Russian).

[9] Rührmair U., Sölter J., Sehnke F. On the Foundations of Physical Unclonable Functions. IACR Cryptology. ePrint Archive. Paper 2009/277. 2009. – 20 p. URL: https://eprint.iacr.org/2009/277 (accessed: 02.02.2023).

[10] Ivaniuk A.A., Zalivaka S.S. Physical cryptography and security of digital devices. Doklady BGUIR. 2019, v. 120, no. 2, p. 50–58. URL: https://libeldoc.bsuir.by/handle/123456789/34706 (accessed: 02.02.2023) (in Russian).

[11] Xu X., Burleson W., Holcomb D. E. Using Statistical Models to Improve the Reliability of Delay-Based PUFs. 2016 IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Pittsburgh, PA, USA. 2016, p. 547–552, DOI: https://doi.org/10.1109/ISVLSI.2016.125.

[12] Yarmolik V.N., Ivaniuk A.A., Shynkevich N.N. Physically unclonable functions with controlled propagation delay. Informatics. 2022, v. 19, no. 1, p. 32–49. DOI: https://doi.org/10.37661/1816-0301-2021-19-1-32-49 (in Russian).

[13] Klybik V.P., Zalivaka S.S., Ivaniuk A.A. Reliability enhancement method for «Arbiter» physically unclonable function. Informatics. 2017, v. 53, n. 1, p. 31–43. URL: https://inf.grid.by/jour/article/view/199 (accessed: 02.02.2023) (in Russian). – EDN: YLJOUJ.

[14] Morozov S., Maiti A., Schaumont P. An Analysis of Delay Based PUF Implementations on FPGA. In: Sirisuk, P., Morgan, F., El-Ghazawi, T., Amano, H. (eds) Reconfigurable Computing: Architectures, Tools and Applications. ARC 2010. Lecture Notes in Computer Science, vol 5992. Springer, Berlin, Heidelberg. DOI: https://doi.org/10.1007/978-3-642-12133-3_37.

[15] Shamyna A.Yu., Ivaniuk A.A. Creating and balancing the paths of arbiter-based physically unclonable functions on FPGA. Informatics. 2022, v. 19, no. 4, p. 27–41. DOI: https://doi.org/10.37661/1816-0301-2022-19-4-27-41 (in Russian).

[16] Yarmolik V.N., Ivaniuk A.A. Arbiter Physical Unclonable Functions with Asymmetric Pairs of Paths. Doklady BGUIR. 2022, v. 20, no. 4, p. 71–79. DOI: https://doi.org/10.35596/1729-7648-2022-20-4-71-79 (in Russian).

[17] Delvaux J., Verbauwhede I. Side channel modeling attacks on 65nm arbiter PUFs exploiting CMOS device noise. IEEE International Symposium on Hardware-Oriented Security and Trust (HOST), Austin, TX, USA. 2013, p. 137–142. DOI: https://doi.org/10.1109/HST.2013.6581579.

[18] Rührmair U. et al. PUF Modeling Attacks on Simulated and Silicon Data. IEEE Transactions on Information Forensics and Security. Nov. 2013, vol. 8, no. 11, p. 1876–1891. DOI: https://doi.org/10.1109/TIFS.2013.2279798.

Поступила в редакцию – 02 февраля 2023 г. Окончательный вариант – 17 февраля 2023 г.
Received – February 02, 2023. The final version – February 17, 2023.