http://dx.doi.org/10.35596/1729-7648-2025-23-5-75-82

УДК 004.312

УПРАВЛЕНИЕ МЕТАСТАБИЛЬНЫМ СОСТОЯНИЕМ ЭЛЕМЕНТА ПАМЯТИ С ЦЕЛЬЮ ГЕНЕРАЦИИ СЛУЧАЙНЫХ ДАННЫХ

М. Н. КАЙКЫ, А. А. ИВАНЮК

Белорусский государственный университет информатики и радиоэлектроники (Минск, Республика Беларусь)

Аннотация. Предлагается новая схема построения генераторов истинно случайных последовательностей с использованием блока управления источниками случайности. В качестве источника случайности рассматривается обобщенный управляемый бистабильный элемент, для которого предложена логическая модель на основе управляемых инверторов с обратной связью. Установлено, что переход бистабильного элемента в состояние метастабильности возможен независимо от его внутренней структуры. Метастабильное состояние выражается в осцилляции выходного сигнала с уникальной частотой, что в дальнейшем позволит генерировать непредсказуемые случайные последовательности. Созданная программная модель управляемого бистабильного элемента на языке SystemVerilog в процессе тестирования доказала свою состоятельность.

Ключевые слова: управляемый бистабильный элемент, генератор истинно случайного числа, физически неклонируемая функция, аналитическая модель.

Конфликт интересов. Авторы заявляют об отсутствии конфликта интересов.

Благодарность. Авторы выражают искреннюю благодарность резиденту ПВТ компании «Инженерный Центр Ядро», которая является одним из центров разработки YADRO, за предоставленное оборудование для проведения экспериментов в рамках работы совместной учебной лаборатории с Белорусским государственным университетом информатики и радиоэлектроники.

Для цитирования. Кайкы, М. Н. Управление метастабильным состоянием элемента памяти с целью генерации случайных данных / М. Н. Кайкы, А. А. Иванюк // Доклады БГУИР. 2025. Т. 23, № 5. С. 75–82. http://dx.doi.org/10.35596/1729-7648-2025-23-5-75-82.

CONTROL OF THE METASTABLE STATE OF A MEMORY ELEMENT FOR THE PURPOSE OF RANDOM DATA GENERATION

MIKHAIL N. KAIKY, ALEXANDER A. IVANIUK

Belarusian State University of Informatics and Radioelectronics (Minsk, Republic of Belarus)

Abstract. A new scheme for constructing truly random sequence generators using a randomness source control unit is proposed. A generalized controlled bistable element is considered as a source of randomness, for which a logical model based on controlled feedback inverters is proposed. It is established that the transition of a bistable element to a metastable state is possible regardless of its internal structure. The metastable state is expressed in oscillation of the output signal with a unique frequency, which will further allow generating unpredictable random sequences. The created software model of a controlled bistable element in the SystemVerilog language proved its viability during testing.

Keywords: controlled bistable element, true random number generator, physically unclonable function, analytical model.

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

Gratitude. The authors express their sincere gratitude to the HTP resident company “Engineering Center Yadro”, which is one of the YADRO development centers, for providing equipment for conducting experiments within the framework of the joint educational laboratory with the Belarusian State University of Informatics and Radioelectronics.

For citation. Kaiky M. N., Ivaniuk A. A. (2025) Control of the Metastable State of a Memory Element for the Purpose of Random Data Generation. *Doklady BGUIR*. 23 (5), 75–82. http://dx.doi.org/10.35596/1729-7648-2025-23-5-75-82 (in Russian).

Введение

С развитием технологий и увеличением потребностей в безопасной передаче и хранении информации актуальным является вопрос генерации случайных чисел. Генераторы случайных чисел лежат в основе множества криптографических систем, протоколов передачи данных, механизмов защиты от атак по побочным каналам, используемых в современных устройствах и приложениях. Так, в системах на кристалле применяются два типа генераторов: генераторы истинно случайных чисел (ГИСЧ) и генераторы псевдослучайных чисел, при этом именно ГИСЧ являются основой и неотъемлемой частью систем защиты информации и безопасности цифровых устройств благодаря возможности генерировать невоспроизводимые, непредсказуемые последовательности чисел, обладающие характеристиками, близкими к случайным.

Современные ГИСЧ основаны на источниках энтропии типа физически неклонируемой функции (ФНФ) [1], а сами методы генерации относятся к определению физической криптографии [2]. ФНФ – это функции, свойства которых таковы, что становится невозможным создать точную копию (или клонирование) их поведения. ФНФ основываются на протекании неуправляемых физических процессов в интегральной микросхеме (ИС), например, на колебаниях квантовых частиц, шуме в электронных компонентах, на технологических вариациях при изготовлении ИС (разброс параметров транзисторов, таких как длина канала, пороговое напряжение, неоднородность металлизации и др.). ФНФ описываются значениями пар «запрос–ответ» (Challenge-Response Pair, CRP) и являются функциями преобразования запросов C_i в ответы R_i.

Обобщенная схема генератора истинно случайных чисел

Построение ГИСЧ в настоящее время является перспективным и значимым процессом при проектировании современных защищенных и доверенных устройств, систем на кристалле. При этом структура генератора описывается в ряде международных и государственных стандартов, таких как NIST SP 800–90 (A, B, C)[1] и ТС 26.4.001–2019 [3]. На рис. 1 предлагается усовершенствованная обобщенная схема для построения ГИСЧ на основе цифровых систем, элементной базы программируемых логических интегральных схем (ПЛИС) и заказных ИС.

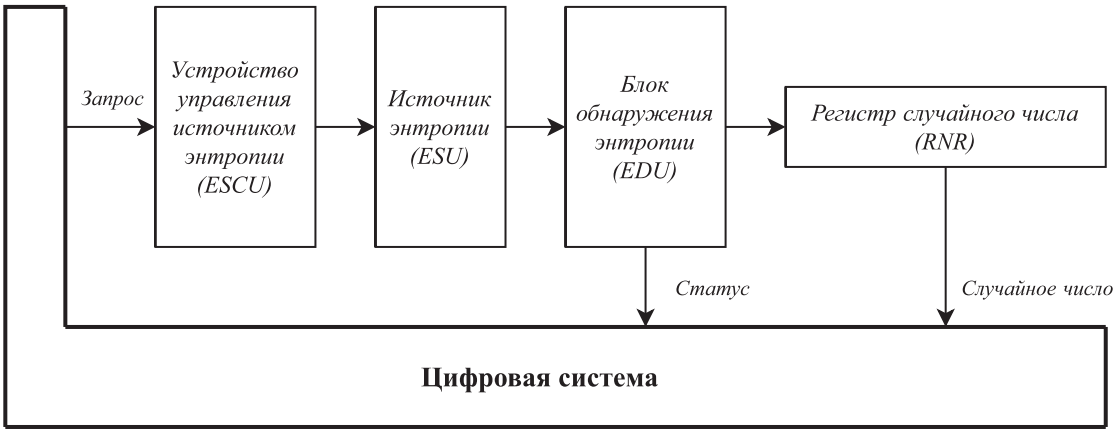

Рис. 1. Усовершенствованная схема генератора истинно случайных последовательностей
Fig. 1. Improving the true random sequence generator circuit

[1] Recommendation for Random Number Generation Using Deterministic Random Bit Generators (Revised) [Electronic Resource]. Mode of access: https://csrc.nist.gov/pubs/sp/800/90/r1/final.

В конечном устройстве, реализованном в виде цифровой системы, в некоторый момент времени формируется запрос на генерацию случайного числа. Данный запрос адресуется в устройство управления источником энтропии ESCU (Entropy Source Control Unit), который, в свою очередь, формирует серию из управляющих сигналов для источника энтропии ESU (Entropy Source Unit). В проводимом исследовании источник энтропии представлял собой цифровой управляемый бистабильный элемент, способный контролируемо переходить в состояние метастабильности и формировать на выходе источника случайные биты информации. Сгенерированные случайные биты поступают на вход блока обнаружения энтропии EDU (Entropy Detecting Unit), который, в свою очередь, проводит оценку качества выработанной случайной последовательности по некоторым статистическим критериям [3]. При их успешном прохождении случайная последовательность попадает в регистр случайного числа RNR (Random Number Register) для дальнейшего использования в цифровой системе.

Физически неклонируемые функции на базе элементов памяти

ФНФ на основе элементов памяти как источник случайности для генераторов истинно случайных чисел получили большое распространение в современных IoT-устройствах и TPM (Trusted Platform Module) ввиду низких аппаратных затрат на их реализацию и высокой энергоэффективности [4]. ФНФ основываются на неконтролируемых вариациях производственного процесса при изготовлении ИС и позволяют извлекать уникальные и непредсказуемые данные при инициализациях ИС или во время ее работы. Популярными примерами ФНФ на элементах памяти являются ФНФ типов SRAM [5], «Арбитр» [6], «Бабочка» [7]. Принцип работы ФНФ типа «Арбитр» основан на различиях в задержках распространения сигналов в идентичных логических цепях, а сам блок «Арбитра» при этом реализуется на D-триггерах или защелках, которые чувствительны к времени предустановки (tsetup) и времени удержания (thold) сигналов на его входе, нарушение которых приводит к попаданию «Арбитра» в состояние метастабильности и генерации случайного ответа *R* на его выходе. SRAM – это тип ФНФ, использующий хаотичное поведение ячеек памяти при включении питания для генерации уникальных и непредсказуемых значений. Основными причинами попадания ячеек памяти в различные состояния инициализации являются технологические вариации при производстве, приводящие к различной подвижности носителей заряда из-за неоднородности кремния, разным пороговым напряжениям, флуктуациям легирования, к неидентичной геометрии каналов и затворов на транзисторах. Вне зависимости от выбираемого типа ФНФ в их структуре всегда присутствует бистабильный элемент памяти, выполняющий роль источника энтропии для ФНФ типа SRAM или ФНФ на базе кольцевого осциллятора, или являющийся мажоритарным элементом для ФНФ типа «Арбитр». Повсеместная применимость бистабильных элементов памяти в структурах ФНФ для проектирования источников энтропии приводит к необходимости моделирования процессов, влияющих на их поведение.

Аналитическая модель бистабильного элемента памяти как источника энтропии

Рассмотрим функциональную модель бистабильного элемента памяти (рис. 2), главную роль в котором играет предлагаемая модель конфигурируемого переключательного элемента – управляемого инвертора. Предлагаемая аналитическая модель описывает поведение элемента памяти, состоящего из двух управляемых инверторов с обратной связью. Особенность подхода – выявление зависимости задержек в переключательных элементах от управляющих сигналов, что открывает возможности для оптимизации параметров ячейки памяти в процессе проектирования.

Полученные аналитические соотношения могут быть использованы как при ручном расчете временных параметров схемы, так и в алгоритмах автоматизированного проектирования цифровых устройств. Для построения аналитической модели идеального бистабильного элемента примем ряд обобщений:

– каждый бистабильный элемент состоит из идентичных, а, значит, и симметричных элементов;

– каждый из управляемых инверторов выполняет функцию инверсии по входу 1 двоичного значения;

– количество управляемых инверторов в петле обратной связи является четным, минимальное количество инверторов – два;

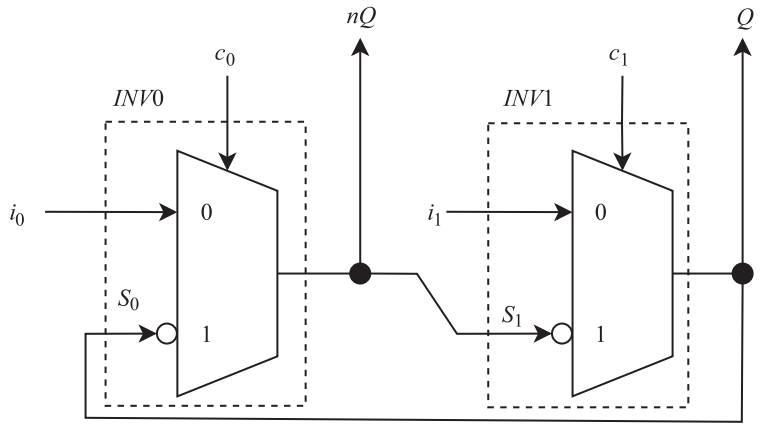

Рис. 2. Функциональная модель бистабильного элемента памяти на управляемых инверторах
Fig. 2. Functional model of a bistable memory element on controlled inverters

– бистабильный элемент способен работать в открытом режиме – режиме предустановки для передачи входного произвольного значения {0,1} на свой выход;

– бистабильный элемент способен работать в закрытом режиме – режиме сохранения.

Рассмотрим обобщенную модель управляемого инвертора на примере *INV*0:

$$nQ = \overline{S_0}c_0 + i_0\overline{c_0}, \tag{1}$$

где S_0 – входной информационный символ; c_0 – управляющий вход; i_0 – вход конфигурации; nQ – символ на выходе элемента.

Описываемый управляемый инвертор является универсальным переключательным компонентом, на базе которого можно описать любой другой управляемый инвертор на уровне логических вентилей, например, операции НЕ-И, НЕ-ИЛИ, исключающее ИЛИ. Для примера опишем элемент НЕ-И при помощи введенного ранее управляемого инвертора. Примем $i_0 = 1$, тогда из уравнения (1) следует (с применением правила поглощения и закона Де Моргана): $nQ = \overline{S_0}c_0 + i_0\overline{c_0} = \overline{S_0}c_0 + c_0 = S_0 + c_0 = \overline{S_0 c_0}$.

Введем понятие задержки переключения δ управляемого инвертора (где $\delta \in N$) и дискретного времени t ($t \in N$). При этом $\delta << t$. Уравнение для управляемого инвертора *INV*0 примет вид: $nQ(t+\delta) = \overline{S_0(t)}c_0 + i_0 c_0$. Примем дополнительное ограничение: входы S_0, c_0, i_0 не подвержены задержкам и изменяют свое состояние мгновенно. С учетом введенных ограничений и полученного выражения для построения аналитической модели запишем систему уравнений для управляемого бистабильного элемента:

$$\begin{cases} nQ(t) = \overline{Q(t-\delta)}c_0 + i\overline{c_0}; \\ Q(t) = \overline{nQ(t-\delta)}c_1 + i\overline{c_1}. \end{cases} \tag{2}$$

Для (2) рассмотрим режимы работы ячейки бистабильного элемента путем перебора входных состояний на входах c_0, c_1. Система уравнений (2) имеет всего четыре состояния, для каждого из которых запишем состояние системы в момент времени t:

– состояние 1 элемента памяти ($c_1 = 0$, $c_0 = 1$) – Set/Reset:

$$nQ(t) = \overline{Q(t-\delta)}c_0 + i\overline{c_0} = \overline{Q(t-\delta)};$$
$$Q(t) = \overline{nQ(t-\delta)}c_1 + i\overline{c_1} = i;$$
$$nQ(t+\delta) = \overline{Q(t)} = \bar{i};$$

– состояние 2 элемента памяти ($c_1 = 1$, $c_0 = 0$) – Reset/Set (следует отметить, что в случае состояний Set/Reset их кодирование зависит от входа конфигурации i, например, при $i = 0$ состояние элемента 1 будет соответствовать состоянию установки, а при $i = 1$ – состоянию сброса):

$$Q(t) = \overline{nQ(t-\delta)}c_1 + i\overline{c_1} = \overline{nQ(t-\delta)};$$

$$nQ(t) = \overline{Q(t-\delta)}c_0 + i\overline{c_0} = i;$$
$$Q(t+\delta) = i;$$

– состояние 3 элемента памяти ($c_1 = 1$, $c_0 = 1$) – Store:

$$Q(t) = \overline{nQ(t-\delta)}c_1 + i\overline{c_1} = \overline{nQ(t-\delta)};$$
$$nQ(t) = \overline{Q(t-\delta)}c_0 + i\overline{c_0} = \overline{Q(t-\delta)};$$

– состояние 4 элемента памяти ($c_1 = c_0 = 0$) – запрещенное состояние:

$$Q(t) = \overline{nQ(t-\delta)}c_1 + i\overline{c_1} = i;$$
$$nQ(t) = \overline{Q(t-\delta)}c_0 + i\overline{c_0} = i.$$

Комбинации символов c_0, c_1 в состоянии 4 противоречат нормальному режиму работы ячейки памяти, так как выход Q приравнивается к выходу nQ.

Рассмотрим работу ячейки во времени. Для этого введем понятие транзакции – любое изменение состояния входных символов c_0, c_1. Поскольку описанная модель оперирует двухзначным алфавитом состояний для c_0, c_1 соответственно, можно описать 12 типов транзакций в виде графа (рис. 3). Для примера проанализируем транзакцию перехода из состояния Set в состояние Store, соответствующую ребру 6 на рис. 3: для $c_1 = 0$, $c_0 = 1$ – $Q(t) = i$; $nQ(t) = \overline{i}$, тогда для $c_1 = 1$, $c_0 = 1$ в момент времени $t + \delta$ будут верны выражения:

$$\begin{cases} Q(t+\delta) = \overline{nQ(t)}c_1 + i\overline{c_1} = \overline{nQ(t)} + 0 = \overline{nQ(t)} = i; \\ nQ(t+\delta) = \overline{Q(t)}c_0 + i\overline{c_0} = \overline{Q(t)} + 0 = \overline{Q(t)} = \overline{i}. \end{cases} \quad (3)$$

Как видно из (3), ячейка, находившаяся в состоянии Set, способна перейти в состояние Store. Рассмотрим транзакцию перехода из запрещенного состояния в состояние Store, соответствующую ребру 10 на рис. 3: для $c_1 = c_0 = 1$ – $Q(t) = nQ(t) = i$, тогда для $c_1 = 1$, $c_0 = 1$ в момент времени $t + \delta$ будут верны выражения:

$$\begin{cases} Q(t+\delta) = \overline{nQ(t)}c_1 + i\overline{c_1} = \overline{nQ(t)} + 0 = \overline{nQ(t)} = \overline{i}; \\ nQ(t+\delta) = \overline{Q(t)}c_0 + i\overline{c_0} = \overline{Q(t)} + 0 = \overline{Q(t)} = \overline{i}. \end{cases} \quad (4)$$

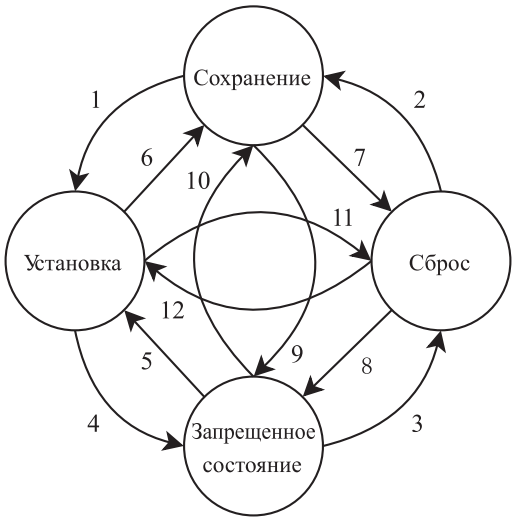

Рис. 3. Граф переходов между состояниями элемента памяти
Fig. 3. State transition graph of a memory element

Согласно (4), ячейка, находившаяся в запрещенном состоянии и имеющая значения выходов, равные i, переходит в состояние выходов $\overline{i}$, что соответствует операции инверсии выходных состояний ячейки, и переходит в режим автоколебаний. Ввиду того, что аналитическая модель

не предусматривает девиации задержек и разброс технологических параметров транзисторов, колебания будут являться незатухающими. Доказать это можно при помощи уравнений:

$$\begin{cases} Q(t+k\delta) = \overline{nQ(t+(k\delta-\delta))}c_1 + ic_1; \\ nQ(t+k\delta) = \overline{Q(t+(k\delta-\delta))}c_0 + ic_0. \end{cases} \quad (5)$$

Подставим в выражение (5) различные значения k ($k \in N$), доказав периодичность изменения знака на выходе модели элемента, и вычислим период его колебаний (период колебаний $P_{osc} = 2\delta$):

$$\begin{cases} Q(t+\delta) = \bar{i} \\ nQ(t+\delta) = \bar{i} \end{cases}; \quad \begin{cases} Q(t+2\delta) = i \\ nQ(t+2\delta) = i \end{cases}; \quad \begin{cases} Q(t+3\delta) = \bar{i} \\ nQ(t+3\delta) = \bar{i} \end{cases}; \quad \begin{cases} Q(t+4\delta) = i \\ nQ(t+4\delta) = i \end{cases}. \quad (6)$$

В отличие от аналитической модели, в реальном устройстве вследствие протекающих переходных процессов между управляемыми инверторами в схеме и их асимметрии в физической реализации ($\delta_{INV0} \neq \delta_{INV1}$) подобные автоколебания являются затухающими [8]. Докажем, что для построенной аналитической модели, несмотря на бесконечные колебания на выходах ячейки, по-прежнему возможен переход из состояния колебаний в детерминированное состояние при помощи операций Set или Reset. Для этого пусть элемент памяти находится в состоянии колебаний, тогда его состояния соответствуют выражениям $Q(t+\delta) = nQ(t+\delta) = \bar{i};\ Q(t+2\delta) = nQ(t+2\delta) = i.$ В момент времени $t + 3\delta$ на вход элемента памяти поступают значения $c_1 = 0$, $c_0 = 1$, соответствующие транзакции установки. Тогда будут выполняться равенства:

$$\begin{cases} Q(t+3\delta) = \overline{nQ(t+2\delta)}c_0 + i\overline{c_0} = \overline{nQ(t+2\delta)} = i; \\ nQ(t+3\delta) = \overline{Q(t+2\delta)}c_1 + i\overline{c_1} = i; \end{cases} \quad (7)$$

$$\begin{cases} Q(t+4\delta) = \overline{nQ(t+3\delta)} = \bar{i}; \\ nQ(t+4\delta) = \overline{Q(t+3\delta)}c_1 + i\overline{c_1} = i. \end{cases} \quad (8)$$

Как видно из (7), (8), ячейка перешла в состояние $Q(t+4\delta) = i;\ nQ(t+4\delta) = \bar{i},$ что соответствует ее нормальному режиму работы и состоянию.

SystemVerilog-модель на базе аналитической модели бистабильного элемента

На высокоуровневом языке описания цифровых схем SystemVerilog с использованием подмножества несинтезируемых конструкций была разработана функциональная модель бистабильного элемента памяти согласно выражению (2) и в соответствии с рис. 2. Модель размещалась в тестовом окружении SystemVerilog для последующего моделирования, которое проводилось в среде Vivado XSIM, при этом $\delta = 10ns$.

На рис. 4 приведена временная диаграмма работы SystemVerilog модели упомянутого ранее бистабильного элемента памяти. Как видно из рисунка, модель элемента корректно реагирует на транзакции чтения, сброса, установки, а также способна переходить в состояние незатухающих автоколебаний на своих выходах. Регистрацию автоколебаний на выходе защелки предлагается выполнять в блоке EDU, например, с использованием блока семплирования данных при помощи системной частоты или выделенной частоты семплирования.

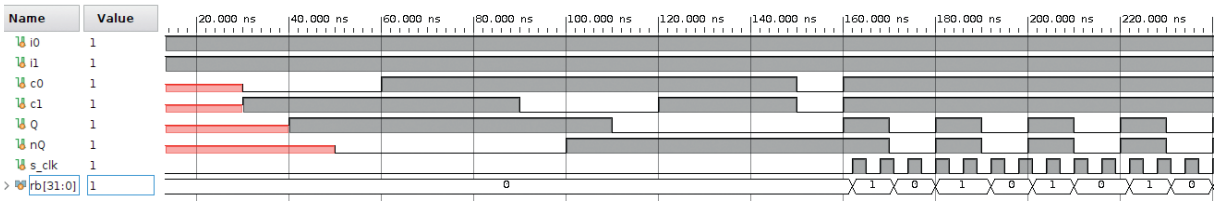

Рис. 4. Временная диаграмма работы бистабильного элемента с семплированием случайных данных
Fig. 4. Timing diagram of a bistable element with random data sampling

На рис. 4, согласно выражению (7) и моделированию, период осцилляции элемента памяти принят $P_{osc} = 2\delta = 20ns$, а период семплирования сигналом $s\_clk - P_{sample} = 3ns$. При этом на выходе устройства семплирования можно наблюдать двоичную последовательность символов «11011100100110», которая в дальнейшем может быть использована как случайные данные для размещения в RNR.

Заключение

1. Предложена модель управляемого бистабильного элемента памяти с целью генерации случайных чисел. В отличие от традиционных подходов, представленная архитектура генератора истинно случайных чисел включает устройство управления источником энтропии (ESCU), что позволяет не только детектировать, но и контролировать недетерминированное поведение элементов памяти.

2. Разработана аналитическая модель бистабильного элемента памяти на управляемых инверторах, описывающая его поведение в различных режимах работы, включая переход в метастабильное состояние для генерации случайных данных. Показано, что при нарушении условий устойчивости элемент памяти способен переходить в режим автоколебаний, что может быть использовано для извлечения энтропии.

3. Проведено моделирование предложенной модели на языке SystemVerilog, подтвердившее возможность управления метастабильностью и регистрации случайных последовательностей. Полученные результаты демонстрируют перспективность использования элементов памяти в качестве источника энтропии для генераторов истинно случайных чисел, особенно в условиях ограниченных аппаратных ресурсов, таких как IoT-устройства.

4. Дальнейшие исследования направлены на оптимизацию параметров управления метастабильным состоянием, а также на экспериментальную верификацию предложенного подхода на реальных интегральных схемах с учетом технологических вариаций.

Список литературы

1. Silicon Physical Random Functions / B. Gassend [et al.] // Proc. of the 9th ACM Conference on Computer and Communications Security (CCS ’02). 2002. P. 148–160.
2. Иванюк, А. А. Физическая криптография и защита цифровых устройств / А. А. Иванюк, С. С. Заливако // Доклады БГУИР. 2019. № 2. С. 50–58.
3. Физические генераторы случайных чисел для применения в СКЗИ, не содержащих сведения, составляющие государственную тайну: ТС 26.4.001–2019.
4. Refillable PUF Authentication Protocol for Constrained Devices / A. Desuert [et al.] // Journal of Ambient Intelligence and Smart Environments. 2022. Vol. 14. P. 1–18.
5. Кайкы, М. Н. Исследование стабильности промышленной SRAM памяти, используемой для неклонируемой идентификации / М. Н. Кайкы, А. А. Иванюк // Информационные технологии и системы 2022 (ИТС 2022): матер. Междунар. науч. конф., Минск, 23 нояб. 2022 г. Минск: Белор. гос. ун-т информ. и радиоэлек., 2022. С. 79–80.
6. Иванюк, А. А. Физически неклонируемая функция типа АРБИТР с нелинейными парами путей / А. А. Иванюк, А. Ю. Шамына // Системный анализ и прикладная информатика. 2023. № 1. С. 54–62.
7. Kumar, S. S. The Butterfly PUF: Protecting IP on Every FPGA / S. S. Kumar // Proc. IEEE International Workshop on Hardware-Oriented Security and Trust (HOST’2008). USA, CA: Anaheim, 2008. P. 67–70.
8. Kacprzak, T. Analysis of Oscillatory Metastable Operation of an RS Flip-Flop / T. Kacprzak // IEEE Journal of Solid-State Circuits. 1988. Vol. 23, No 2. P. 260–266.

Поступила 20.06.2025 Принята в печать 15.07.2025

References

1. Gassend B., Clarke D., Van Dijk M., Devadas S. (2002) Silicon Physical Random Functions. *Proc. of the 9th ACM Conference on Computer and Communications Security (CCS ’02)*. 148–160.
2. Ivaniuk A. A., Zalivaka S. S. (2019) Physical Cryptography and Security of Digital Devices. *Doklady BGUIR*. (2), 50–58 (in Russian).
3. Technical Specifications TS 26.4.001–2019. *Physical Random Number Generators for Use in Cryptographic Information Protection Tools That Do Not Contain Information Constituting a State Secret* (in Russian).

4. Desuert A., Chollet S., Pion L., Hély D. (2022) Refillable PUF Authentication Protocol for Constrained Devices. *Journal of Ambient Intelligence and Smart Environm*ents. 14, 1–18.
5. Kaiky M. N., Ivaniuk A. A. (2022) Research of the Stability of Industrial SRAM Memory Used for Unclonable Identification. *Information Technologies and Systems 2022 (ITS 2022), Proceedings of the International Scientific Conference, Minsk, Nov. 23*. Minsk, Belarusian State University of Informatics and Radioelectronics. 79–80 (in Russian).
6. Ivaniuk A. A., Shamina A. Y. (2023) Physically Non-Cloneable Arbiter-Type Function with Non-Linear Path Pairs. *Systems Analysis and Applied Informatics*. (1), 54–62 (in Russian)
7. Kumar S. S. (2008) The Butterfly PUF: Protecting IP on Every FPGA. *Proc. IEEE International Workshop on Hardware-Oriented Security and Trust (HOST'2008)*. USA, CA, Anaheim. 67–70.
8. Kacprzak T. (1988) Analysis of Oscillatory Metastable Operation of an RS Flip-Flop. *IEEE Journal of Solid-State Circuits*. 23 (2), 260–266.

Received: 20 June 2025 Accepted: 15 July 2025

Вклад авторов

Кайкы М. Н. построил аналитическую модель управляемого бистабильного элемента с целью генерации случайных чисел, доказал пригодность модели при помощи программной модели на SystemVerilog и симуляции, проанализировал и обобщил полученные результаты.

Иванюк А. А осуществил постановку задачи для проведения исследования, принял участие в обобщении результатов.

Authors' contribution

Kaiky M. N. constructed an analytical model of a controlled bistable element for the purpose of generating random numbers, proved the suitability of the model using a software model on SystemVerilog and simulation, analyzed and generalized the obtained results.

Ivaniuk A. A. carried out the formulation of the problem for the study, took part in the generalization of the results.

Сведения об авторах

Кайкы М. Н., магистр техн. наук, Белорусский государственный университет информатики и радиоэлектроники

Иванюк А. А., д-р техн. наук, проф., проф. каф. информатики, Белорусский государственный университет информатики и радиоэлектроники

Адрес для корреспонденции

220013, Республика Беларусь,
Минск, ул. П. Бровки, 6
Белорусский государственный университет
информатики и радиоэлектроники
Тел.: +375 29 386-91-82
E-mail: kaikymykhailo@gmail.com
Кайкы Михаил Николаевич

Information about the authors

Kaiky M. N., M. Sci. (Tech.), Belarusian State University of Informatics and Radioelectronics

Ivaniuk A. A., Dr. Sci. (Tech.), Professor, Professor at the Department of Computer Science, Belarusian State University of Informatics and Radioelectronics

Address for correspondence

220013, Republic of Belarus,
Minsk, P. Brovki St., 6
Belarusian State University
of Informatics and Radioelectronics
Tel.: +375 29 386-91-82
E-mail: kaikymykhailo@gmail.com
Kaiky Mikhail Nikolaevich