ISSN 2658-5014 (Print) СИИТ ISSN 2686-7044 (Online)
http://siit.ugatu.su 2026. Т. 8, № 1(25)
СИСТЕМНАЯ ИНЖЕНЕРИЯ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ
Научная статья

УДК 004.056.55 DOI: 10.54708/SIIT-2026-no1-p32 EDN: RXOCGQ

Применение квантовых генераторов случайных чисел в системах защиты информации: анализ и прототипирование

Н. В. Кучкарова • А. Д. Ерошкин

Уфимский университет науки и технологий

В представленной работе проведен анализ существующих квантовых генераторов случайных чисел (КГСЧ) в контексте их использования в средствах защиты информации. Проведена классификация КГСЧ по трем основным критериям: природе квантового явления, уровню доверия и проверки, архитектуре и уровню обработки данных. Сделаны выводы о возможности применения определенных видов КГСЧ в задачах обеспечения информационной безопасности. В качестве примера реализации КГСЧ разработан прототип квантового генератора паролей, обеспечивающий криптографически стойкую случайность за счёт фундаментальной непредсказуемости квантовых измерений. Алгоритм генерации реализован на языке квантового программирования Q# и использует суперпозицию кубитов с последующим проективным измерением в вычислительном базисе

Генератор псевдослучайных чисел; квантовый компьютер; средства защиты информации; квантовый генератор случайных паролей.

Введение

Повсеместное внедрение цифровых технологий порождает новые угрозы в виде кибератак, направленных на государственные информационные системы, цифровую инфраструктуру промышленных предприятий, банковские системы и другие объекты с высоким социально-экономическим значением для страны. Целями киберпреступников чаще всего является получение несанкционированного доступа в ИС объектов с последующей кражей конфиденциальных данных, их модификацией и незаконным распространением, а также перехват данных при их передаче по каналам связи [Куч24, Хай25]. Классической стратегией обеспечения безопасности данных в таком случае является использование криптографических средств защиты[1] [Вил25].

Для решения криптографических задач с 50-х годов прошлого столетия используются генераторы псевдослучайных чисел (ГПСЧ). Но ГПСЧ обладают рядом недостатков[2] [Сле17]:

- предсказуемость генерируемых последовательностей. В результате использования детерминированных алгоритмов существует вероятность восстановления начального состояния, что позволяет восстановить всю последовательность генерируемых чисел;

[1] URL: https://www.securitylab.ru/news/308676.php. (дата обращения 10.10.2025).
[2] Случайные числа своими руками. № 1. URL: https://xakep.ru/2005/01/19/25259 (дата обращения 10.10.2025).

Кучкарова Н. В., Ерошкин А. Д. Применение квантовых генераторов случайных чисел в системах защиты информации: анализ и прототипирование // СИИТ. 2026. Т. 8, № 1(25). С. 32-46. DOI: 10.54708/SIIT-2026-no1-p32. EDN: RXOCGQ.

Kuchkarova N.V., Eroshkin A.D. “Application of Quantum Random Number Generators in Information Security Systems: Analysis and Prototyping I” // SIIT. 2026. Vol. 8, no. 1(25), pp. 32-46. DOI: 10.54708/SIIT-2026-no1-p32. EDN: RXOCGQ (In Russian).

- генерируемые последовательности носят цикличный характер ввиду ограниченности ресурсов большинства ГПСЧ;
- зависимость последующих чисел от предыдущих вследствие использования алгоритмических процессов при генерации чисел.

С появлением квантовых компьютеров использование традиционных ГПСЧ становится неэффективным, поскольку злоумышленники используют описанные выше недостатки в преступных целях [Son16]. В 2018 году Национальный институт стандартов и технологий США (NIST) разработал генератор случайных чисел, основанный на феномене квантовой запутанности. Считается [Aci16], что квантовые генераторы случайных чисел создают истинные непредсказуемые последовательности чисел, что лишает их недостатков, описанных выше, и соответственно позволяет обеспечить высокий уровень безопасности и случайности при решении криптографических задач, научных исследованиях, криптовалютных протоколах и иных сферах [Aci16]. В разделе 1 данной публикации будет проведен сравнительный анализ и представлена классификация КГСЧ в контексте применимости их в средствах защиты информации. Раздел 2 содержит прототип генератора случайных паролей, написанный на языке Q#.

1. Классификация квантовых генераторов случайных чисел

Разработанная классификация представлена на рис. 1

Классификация группирует КГСЧ по следующим классификационным признакам:

- по природе квантового явления;
- по уровню доверия и проверки;
- по архитектуре реализации.

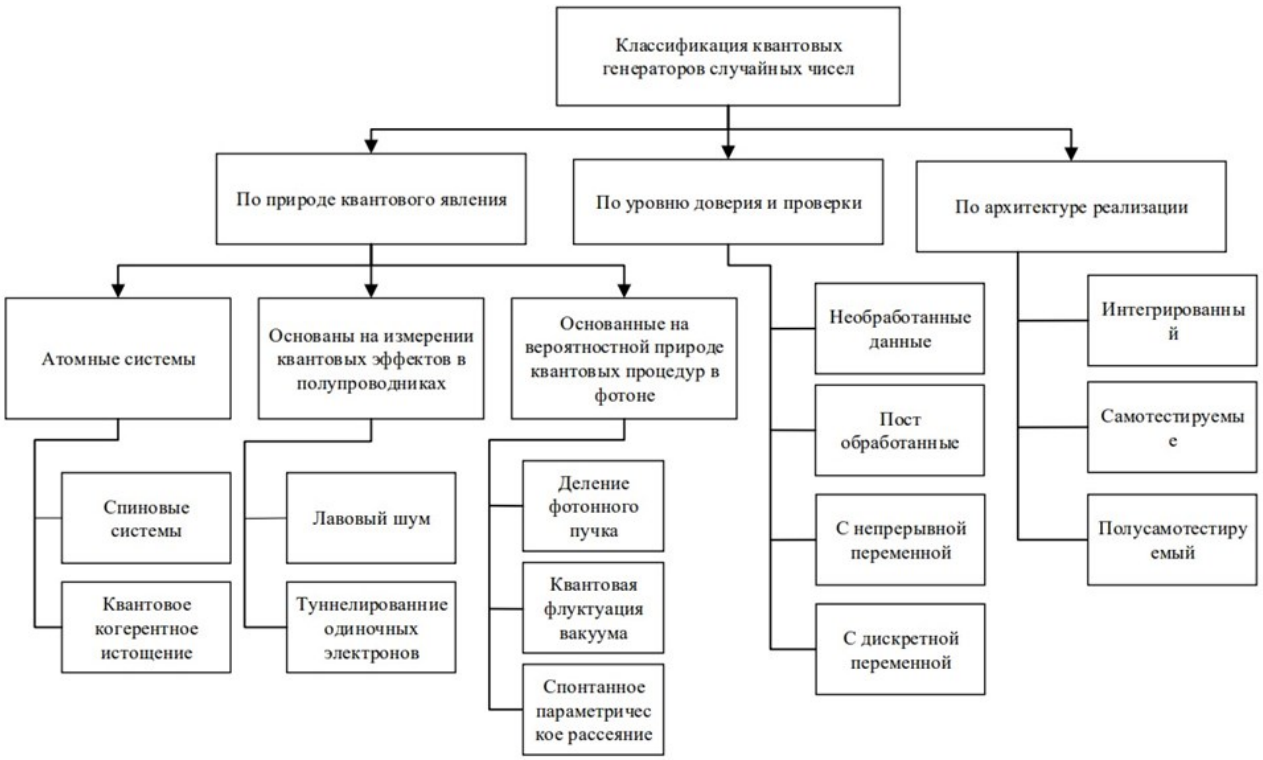

Рис. 1 Классификация квантовых генераторов случайных чисел

Далее каждая классификационная группа будет рассмотрена подробнее.

По результатам классификации была составлена таблица, в которой проведен сравнительный анализ КГСЧ по следующим критериям: тип, безопасность, скорость генерации бит, удобство генерации, стоимость, требования к среде.

Таблица

Сравнительный анализ КГСЧ

Критерий классификации	Подкатегория	Тип / реализация	Безопасность (криптографическая надёжность)	Скорость генерации (бит/с)	Удобство интеграции (в систему / API / SDK)	Стоимость (относительная)	Требования к среде / стабильности
По природе квантового явления	Атомные системы	Спиновые системы	Высокая (источник истинной случайности)	10^6–10^7	Низкое (требует лазерной оптики, магнитных полей)	Очень высокая	Чувствителен ко внешним полям, температуре
		Квантовое когерентное истощение	Высокая (на основе когерентных состояний)	10^7–10^8	Среднее (требует сложной оптики)	Высокая	Требует стабильного лазера и изоляции от шумов
	Основанные на измерении квантовых эффектов в полупроводниках	Лавовый шум	Высокая (квантовый шум в p-n-переходах)	10^7–10^{10}	Высокое (интегрируется в cmos-чипы)	Средняя	Устойчивко внешним воздействиям
		Туннелирование одиночных электронов	Высокая (квантово-механический процесс)	10^6–10^8	Среднее (требует низких температур или специальных чипов)	Высокая	Чувствителен к температуре и напряжению
	Основанные на вероятностной природе квантовых процедур в фотоне	Деление фотонного пучка	Высокая (истинная случайность из квантового разделения)	10^8–10^{10}	Среднее (требует оптики, сплиттеров)	Высокая	Чувствителен к вибрациям и дрейфу оптики
		Квантовая флуктуация вакуума	Высокая (вакуумные флуктуации – фундаментальный источник)	10^7–10^9	Низкое (сложная оптика, требует усиления сигнала)	Очень высокая	Чувствителен к температуре и вибрациям
		Спонтанное параметрическое рассеяние	Высокая (источник пар фотонов)	10^7–10^9	Среднее (требует нелинейных кристаллов)	Высокая	Чувствителен к температуре и выравниванию оптики
По уровню доверия и проверки	Необработанные данные	Сырые выходные данные	Низкая (без постобработки – может содержать корреляции)	10^8–10^{11}	Высокое (минимальная обработка)	Низкая	Чувствителен к шумам и дрейфу
	Постобработанные	Алгоритмическая очистка (напр., von Neumann, SHA-3)	Гарантированно высокая (при корректной реализации)	Ниже (задержка обработки)	Среднее (требуется по или аппаратный модуль)	Средняя	Устойчив, но зависит от алгоритма
	Непрерывные / С непрерывной переменной	Континуальное измерение (например, гомо-	Высокая (при правильной калибровке)	10^8–10^{10}	Среднее (требует аналоговой электроники)	Высокая	Чувствителен к калибровке и шумам

Критерий классификации	Подкатегория	Тип / реализация	Безопасность (криптографическая надёжность)	Скорость генерации (бит/с)	Удобство интеграции (в систему / API / SDK)	Стоимость (относительная)	Требования к среде / стабильности
		динное детектирование)					
	Дискретные переменные	Бинарное измерение (например, поляризация фотона)	Высокая (истинная случайность)	10^7–10^9	Среднее (требует детекторов и синхронизации)	Высокая	Чувствителен к эффективности детекторов
По архитектуре реализации	Интегрированный	На кремниевой платформе (Si photonics, CMOS)	Высокая (при корректном дизайне)	10^7–10^{10}	Высокое (usb, pcie, ethernet интерфейсы)	Средняя	Устойчив, совместим с промышленными стандартами
	Самотестируемые	С постоянной самодиагностикой (например, NIST SP 800-90B compliant)	Очень высокая (обнаружение сбоев в реальном времени)	Ниже (из-за проверок)	Низкое (сложная логика, дополнительные модули)	Высокая	Требует стабильного питания и контроля
	Полусамотестируемый	Частичная самопроверка (напр., периодические тесты)	Высокая (при регулярной проверке)	Средняя	Среднее (гибридная архитектура)	Средняя–высокая	Умеренно устойчив

1.1. По природе квантового явления

КГСЧ, основанные на вероятностной природе квантовых процедур в фотоне. Квантовая оптика является наиболее популярной и коммерчески доступной группой КГСЧ. Работа генераторов основана на вероятностной природе квантовых процедур в фотоне [Kai20]. Данную группу можно условно разделить на три подгруппы, описанные ниже.

- Деление фотонного пучка (Beam Splitters). Работа генератора основана на принципе суперпозиции: одиночный фотон попадает на полупрозрачное зеркало (сплиттер). С вероятностью 50% он может либо отразиться состояние$|0\rangle$, либо пройти $|1\rangle$, результат непредсказуем. Преимуществами данного вида генераторов являются высокая криптографическая надежность и высокая скорость генерации чисел. Недостаток: чувствительность к температуре и вибрациям [Kai20, Aci16, Dyn08].
- Квантовая флуктуация вакуума (Vacuum Fluctuations). Даже при отсутствии фотонов (в вакуумном состоянии) электромагнитное поле подвержено квантовым флуктуациям. Эти вакуумные флуктуации регистрируются методом гомодинного детектирования и проявляются как непредсказуемый квантовый шум, который может использоваться в качестве источника истинной случайности. Преимущества такого подхода: высокая криптографическая стойкость и высокая скорость генерации чисел. Недостаток: чувствительность к изменениям температуры и механическим вибрациям [Wan22].
- Спонтанное параметрическое рассеяние (Spontaneous Parametric Down-Conversion, SPDC). Фотон лазера накачки проходит через нелинейный кристалл и с малой вероятностью спонтанно распадается на пару коррелированных фотонов – сигнальный и холостой. Момент и направление распада случайны. Детектирование одного из фотонов (например, холостого)

маркирует случайное событие, связанное с другим. Преимущество: позволяет генерировать случайные биты на основе различных фотонных параметров – времени прихода, поляризации. Недостатки: требует сложной оптической настройки и чувствителен к стабильности накачки и температуре кристалла[3] [Wan22, Her17].

КГСЧ, основанные на измерении квантовых эффектов в полупроводниках. Генераторы, основанные на измерении квантовых эффектов в полупроводниках, представляют собой важную альтернативу оптическим КГСЧ. Их работа основана на фундаментальной случайности квантовых процессов, происходящих в твердотельных структурах – таких как туннелирование электронов или инициация лавинного пробоя. Данную группу можно условно разделить на две подгруппы: туннелирование одиночных электронов и лавинный (лавовый) шум.

- Лавовый шум в полупроводниках. При обратном смещении p-n-перехода в режиме лавинного пробоя носители заряда ускоряются и вызывают цепную реакцию ионизации. Начало каждой лавины инициируется квантовым туннелированием или тепловым возбуждением и носит случайный характер. Преимущества: простота реализации на стандартной полупроводниковой базе, высокая скорость генерации. Недостатки: присутствие классического (теплового) шума, требует тщательной фильтрации и постобработки для выделения квантовой составляющей[4,5].
- Туннелирование одиночных электронов. Электрон проходит через тонкий изолирующий барьер (туннельный переход) в наноструктуре с вероятностью, определяемой законами квантовой механики. Момент туннелирования непредсказуем и используется как источник случайности. Преимущества: высокая стабильность, компактность и совместимость с интегральной электроникой. Недостаток: требует криогенных температур или сложных наноструктур для подавления классического шума.

Атомные системы. Атомные системы представляют собой перспективный класс квантовых генераторов случайных чисел, основанный на непредсказуемости квантовых переходов в атомах или ионах. Работа таких генераторов опирается на фундаментальную случайность моментов спонтанного излучения, квантовых скачков между энергетическими уровнями или измерений состояний отдельных атомов. Данную группу можно условно разделить на две подгруппы, описанные ниже.

- Спиновые системы. Электрон или ядро с неопределённой ориентацией спина помещается в магнитное поле и измеряется в заданном базисе. С вероятностью, определяемой начальной суперпозицией, результат измерения даёт «вверх» $|0\rangle$ или «вниз» $|1\rangle$ – исход непредсказуем. Преимущества: высокая криптографическая надёжность, совместимость с твердотельными квантовыми устройствами. Недостатки: чувствительность к магнитным шумам и декогеренции, требует сложных методов считывания [Жиз23, Фай16].
- Квантовое когерентное истощение. В когерентной квантовой системе (например, ансамбле кубитов) слабое возбуждение с небольшой вероятностью вызывает случайный переход, «истощающий» когерентное состояние. Момент и канал такого события непредсказуемы и используются как источник случайности. Преимущества: высокая скорость генерации, потенциал для масштабируемой интеграции. Недостатки: требует поддержания когерентности, чувствителен ко внешним возмущениям и шуму [Her17, Wan22].

---

[3] Неслучайная случайность, или Атака на ГПСЧ в .NET 1. URL: https://habr.com/ru/companies/skbkontur/articles/347758 (дата обращения 10.10.2025).

[4] Физики создали самый надежный генератор случайных чисел. URL: https://hi-tech.mail.ru/news/129031-kvantovye-fiziki-predstavili-samyj-nadezhnyj-generator-sluchajnyh-chisel (дата обращения 10.10.2025).

[5] Методы экстракции квантовой случайности. URL: http://qutes.org/wp-content/uploads/2023/03/QTS_2023.pdf (дата обращения 10.10.2025).

1.2. По уровню доверия и проверки

Эта классификационная группа критически важна с точки зрения качества и скорости генерации конечного случайного числа.

Необработанные данные. Сигнал, полученный непосредственно с детектора квантового генератора, отражает физический отклик на случайное квантовое событие – будь то флуктуация поля, прохождение фотона или туннелирование электрона. Этот сигнал содержит истинную квантовую случайность, но также может включать технические и классические шумы.

Преимущества: максимальная близость к физическому источнику энтропии, минимальная задержка. Недостатки: неидеальная статистика, возможны смещённость и корреляции, требует последующей обработки [Jac21, Man22, Aci16].

Пост-обработанные данные представляют собой результат обработки «сырого» сигнала КГСЧ с помощью алгоритмов выравнивания, сжатия или экстракции случайности. Цель постобработки – удалить остаточные корреляции, снизить влияние технических шумов и гарантировать статистическую близость выходной последовательности к идеальному равномерному распределению. К данной группе относится «алгоритмическая очистка»: сырые данные подвергаются детерминированной обработке с помощью криптографических примитивов (например, хеширования или экстракторов энтропии), чтобы устранить смещение, корреляции и влияние классического шума. Выходная последовательность становится статистически близкой к равномерному распределению. Преимущества: гарантирует высокое качество случайности, совместима с криптографическими стандартами**.** Недостатки: снижает итоговую скорость генерации, безопасность зависит от корректности реализации алгоритма[6] [Dyn08].

Генераторы с непрерывной переменной основаны на измерении квадратур электромагнитного поля или других аналоговых квантовых наблюдаемых, принимающих значения в непрерывном диапазоне. Источником случайности служат квантовые флуктуации вакуума или шум сжатого света, регистрируемые, например, методом гомодинного детектирования. К данной группе относится «континуальное измерение». Электромагнитное поле (часто вакуумное или сжатое) интерферирует с опорным лазерным пучком на светоделителе, и разность фототоков на выходе даёт непрерывный аналоговый сигнал, отражающий квантовые флуктуации квадратур поля. Преимущества: высокая скорость генерации, прямой доступ к квантовому шуму вакуума. Недостатки: требует высокоточной оптической стабилизации, чувствителен к электронному и оптическому шуму [Man22, Aci16].

Дискретные генераторы используют квантовые процессы с конечным числом исходов – например, прохождение или отражение одиночного фотона, спиновое состояние электрона или факт детектирования пары фотонов при спонтанном параметрическом рассеянии. Результат каждого измерения дискретен (обычно бинарен: $|0\rangle$ или $|1\rangle$) и непосредственно преобразуется в случайный бит. К данной группе относится «бинарное измерение». Одиночный фотон в суперпозиции поляризационных состояний (например, $|H\rangle$ и $|V\rangle$) проходит через поляризационный светоделитель. С вероятностью 50% он детектируется в одном из двух выходов, соответствующих $|0\rangle$ или $|1\rangle$ – результат непредсказуем.

Преимущества: простая интерпретация результата как случайного бита, высокая криптографическая надёжность. Недостатки: ограничена скоростью одиночных детекторов, чувствительна к не идеальностям оптики и потере фотонов [Aci16, Dyn08].

1.3. По архитектуре реализации

Интегрированные генераторы по архитектуре и способу проверки реализуются в виде компактных микросхем – фотонных или полупроводниковых, – где источник квантовой случайности, оптические/электронные элементы и детекторы размещены на одном кристалле.

---

[6] NIST Special Publication 800-90B. Recommendation for the Entropy Sources Used for Random Bit Generation. U.S. Department of Commerce, 2018. URL: https://csrc.nist.gov/publications/detail/sp/800-90b/final (дата обращения 14.10.2025)

Проверка случайности осуществляется через мониторинг стабильности выходного сигнала, калибровку на этапе производства и анализ статистических свойств «сырых» данных в реальном времени. Данная группа состоит из одной подгруппы генераторов, реализуемых на кремниевой платформе (Si photonics, CMOS). Генератор реализован с использованием стандартных технологий кремниевой микроэлектроники или интегральной фотоники, где квантовый источник (например, лавинный диод или спонтанное рассеяние в волноводе) и считывающая схема интегрированы на одном чипе. Преимущества: совместимость с массовым производством, низкая стоимость, компактность и энергоэффективность. Недостатки: ограниченная чистота квантового сигнала из-за технологических шумов, сложность верификации квантовой природы случайности «на чипе» [Kel98Kel98Kel98Kel98, Max16, Jof11].

Самотестируемые генераторы строятся на основе пространственно разделённых квантовых систем (например, запутанных фотонов или ионов), измеряемых независимыми устройствами, без предположений об их внутреннем устройстве. Проверка случайности основана на наблюдении нарушения неравенств Белла или других device-independent корреляций, что гарантирует присутствие квантовой энтропии без необходимости доверять аппаратному обеспечению. Данная группа включает в себя генераторы с постоянной самодиагностикой. Генераторы данного типа включают в себя встроенную систему непрерывного мониторинга статистических свойств выходного потока (энтропия, автокорреляция, равномерность) в соответствии со стандартами, такими как NIST SP 800-90B, и автоматически приостанавливает выдачу данных при отклонении от нормы. Преимущества: соответствие криптографическим стандартам, надёжное обнаружение деградации или атак, пригодность для сертифицированных применений. Недостатки: не гарантирует квантовое происхождение случайности – только статистическое качество; уязвим к скрытым смещениям, не проявляющимся в тестах [Сле17, Son16].

Полусамотестируемые генераторы предполагают частичное знание о внутренней структуре устройства – например, о размерности квантового состояния или о базисе измерения, но не требуют полной калибровки всех компонентов. Проверка случайности выполняется с использованием semi-device-independent протоколов, таких как ограничения на энергию, среднее значение наблюдаемой или минимальную энтропию при заданных физических допущениях. Данная группа включает в себя генераторы с частичной самопроверкой. Генератор периодически выполняет внутренние проверки, такие как калибровка уровня шума, тесты на симметрию или кратковременные измерения параметров источника, чтобы подтвердить работоспособность и стабильность квантового процесса. Преимущества: снижение риска длительного дрейфа параметров, умеренные накладные расходы по скорости и ресурсам. Недостатки: между проверками возможны необнаруженные сбои или атаки; безопасность зависит от частоты и полноты тестов [Jac21, Man22, Aci16].

Проведённый анализ КГСЧ позволяет сформулировать следующие выводы:

1. Многоуровневость классификации. Современные квантовые генераторы случайных чисел представляют собой технологически разнородный класс устройств. Для их комплексной оценки необходима многоаспектная классификация по взаимодополняющим критериям: физической платформе (оптика, полупроводники, атомные системы), типу измеряемой величины (дискретные/непрерывные), архитектуре верификации (интегрированные, самотестируемые) и уровню обработки данных (сырые/постобработанные). Такой подход позволяет гибко оценить баланс между криптографической надёжностью и практической реализуемостью каждого типа КГСЧ.

2. Источник надежности. Все рассмотренные технологии обеспечивают принципиально более высокий уровень истинной случайности по сравнению с классическими ГПСЧ, поскольку основаны на фундаментальной непредсказуемости квантовых процессов: туннелирования, спонтанного излучения и вакуумных флуктуаций.

3. Компромисс при выборе. Анализ выявил четкий дихотомический тренд:

- практичность для массового внедрения: наиболее перспективны полупроводниковые решения (лавинный шум, туннелирование) и интегрированные CMOS/Si-фотонные платформы. Они предлагают оптимальное сочетание высокой скорости генерации, технологической совместимости с существующей электроникой, приемлемой стоимости и устойчивости;
- максимальная криптографическая надёжность: наивысший, вплоть до device-independent, уровень доверия обеспечивают оптические и атомные системы. Однако их применение сдерживается необходимостью в сложном, дорогостоящем оборудовании и высокой чувствительностью к условиям эксплуатации.

4. Рекомендация для систем защиты информации. Для задач, где критически важна доказуемая случайность, например, генерация одноразовых паролей, предпочтительны архитектуры с дискретным бинарным измерением, например, на основе поляризации одиночных фотонов. Их сочетание с алгоритмической постобработкой (экстракция энтропии) гарантирует не только высокое качество случайности, но и практическую совместимость с программными криптографическими платформами, что подтверждается разработанным в работе прототипом на языке Q#.

Таким образом, выбор конкретной реализации КГСЧ для систем защиты информации является стратегическим решением, требующим взвешенного компромисса между требованиями к безопасности, производительностью, бюджетом и средой эксплуатации. Предложенная классификация служит основой для обоснованного принятия такого решения.

2. Прототип квантового генератора

На основе проведённого анализа существующих подходов к генерации случайных чисел и выявленных недостатков классических решений в работе разработан прототип квантового генератора паролей, реализующий принципы фундаментальной квантовой случайности. В отличие от псевдослучайных алгоритмов или аппаратных генераторов, основанных на классическом шуме, предложенный прототип использует непредсказуемость квантового измерения как источник энтропии, что обеспечивает теоретически неуязвимую основу для создания криптографически стойких паролей. Ниже представлены алгоритм программного решения, прототипа генератора и особенности его реализации.

Код программы написан на языке программирования Q# с использованием внутренних библиотек, разработанных компанией Microsoft. В качестве компилятора применяется пакет Visual Studio Code. Q# является высокоуровневым языком программирования, который позволяет внедрять квантовые алгоритмы в код.

Алгоритм работы прототипа квантового генератора состоит из 3 этапов:

- генерация отдельных квантовых битов (GeneratorBitov());
- преобразование кубитов в числа заданного диапазона (GeneratorChisel(max));
- построение символьного пароля на основе полученных чисел (password(max)).

Разработанный алгоритм квантового генератора паролей реализует фундаментальный принцип квантовой механики – непредсказуемость результата измерения кубита, находящегося в суперпозиции базисных состояний. В отличие от классических методов, основанных на детерминированных или псевдослучайных процессах, предложенный подход обеспечивает истинную случайность на уровне каждого генерируемого бита. Ниже приведено пошаговое описание алгоритма, включая инициализацию квантового состояния, выполнение измерений, преобразование битовой последовательности в читаемый пароль и обработку параметров генерации.

2.1. Генерация квантового бита (GeneratorBitov())

Данный этап состоит из подэтапов 1–6. Алгоритм работы представлен на рис. 2.

Прежде всего определяется базовая операция для генерации одного случайного бита посредством квантового измерения (1). Объявляется функция operation GeneratorBitov():

Result, которая инкапсулирует всю логику получения одного истинно случайного бита на основе квантовых принципов. Это основная строительная единица всего алгоритма.

Выделение одного кубита (2) – физического носителя квантовой информации, способного находиться в состоянии суперпозиции. Создаётся один кубит q = Qubit(), который изначально инициализируется в базовое состояние $|0\rangle$. Это обеспечивает детерминированную отправную точку перед применением квантовых операций.

Применение оператора Адамара $H(q)$ (3). К кубиту применяется унитарный оператор Адамара, переводящий его из состояния $|0\rangle$ в суперпозицию. Это создаёт равновероятную суперпозицию двух базовых состояний, лежащую в основе квантовой неопределённости.

Измерение кубита $M(q)$ (4). Результат otvet $\in \{0, 1\}$ определяется коллапсом волновой функции. Вероятности: $P(0) = P(1) = ½$. Производится проективное измерение кубита в вычислительном базисе. В результате суперпозиция коллапсирует в одно из классических состояний – 0 или 1 – с вероятностью 50% каждое. Это превращает квантовую неопределённость в классическую случайность.

Сброс квантового состояния Reset(q) (5). После измерения кубит возвращается в исходное состояние $|0\rangle$. Это необходимо для повторного использования кубита в последующих вызовах функции без влияния предыдущего состояния (предотвращение корреляций).

Рис. 2 Этап 1. Генерация отдельных квантовых бит

Возврат результата измерения (6) – одного случайного бита. С этого момента квантовая неопределённость превращается в классическую случайность. Функция завершает работу, возвращая результат измерения (otvet). Этот бит является истинно случайным, поскольку его значение определяется физическим процессом квантового измерения, а не детерминированным алгоритмом.

2.2. Генерация чисел (GeneratorChisel(max))

Данный этап состоит из подэтапов 7–15. Алгоритм работы второго этапа представлен на рис. 3. Функция генерации случайного числа в диапазоне [0, max] (7). Использует последовательность вызовов GeneratorBitov(). Объявляется функция operation GeneratorChisel(max : Int): Int, которая использует метод отбора (rejection sampling) и многократные вызовы GeneratorBitov() для получения равномерно распределённого числа в заданном диапазоне.

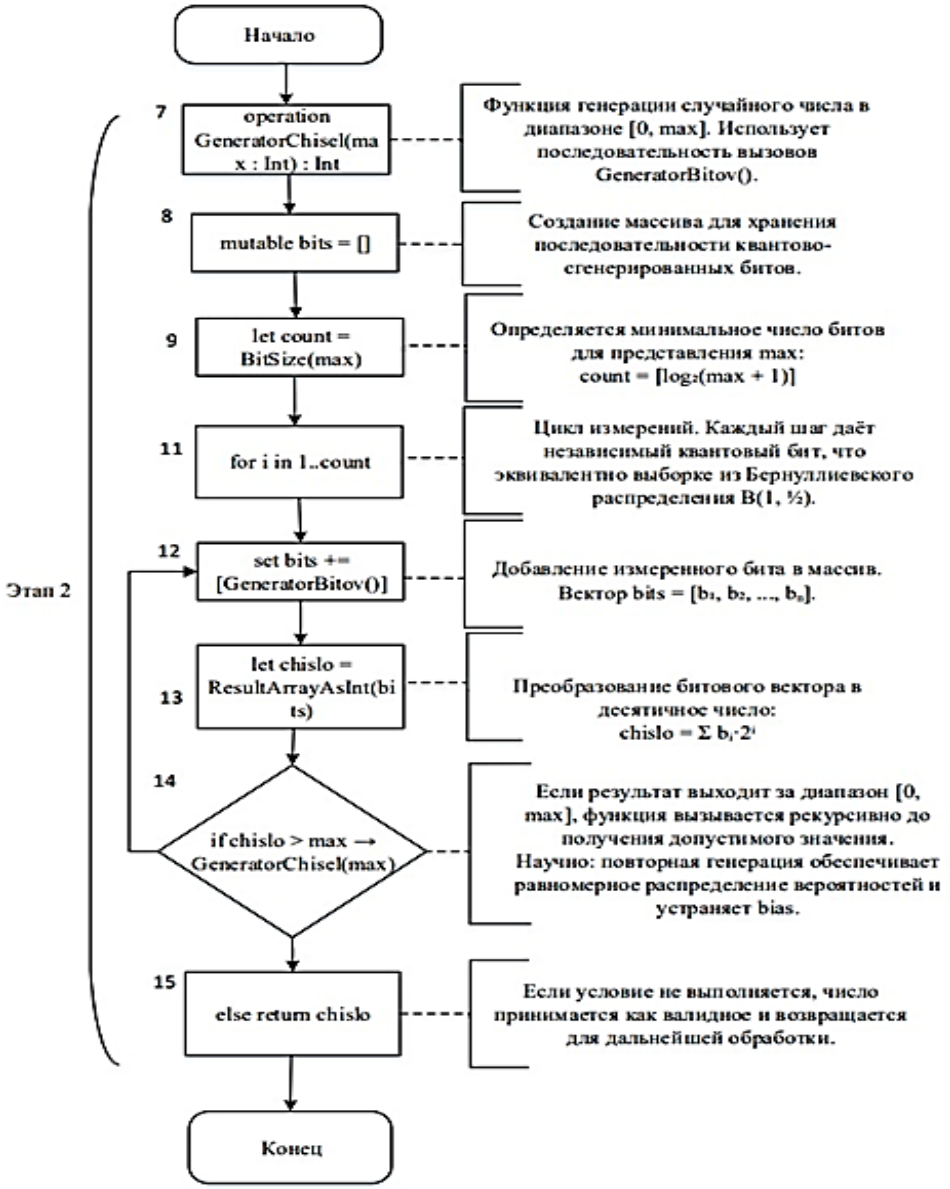

Рис. 3 Этап 2. Генерация чисел

Создание массива для хранения последовательности квантово-сгенерированных битов (8). Инициализируется изменяемый массив mutable bits = [], предназначенный для временного хранения последовательности битов, полученных в ходе цикла генерации. Это позволяет собрать полное двоичное представление числа.

Определение минимального числа битов для представления max (9). Вычисляется количество битов, необходимое для однозначного представления любого числа в диапазоне [0, max]. Формула гарантирует, что все возможные значения будут покрыты без потери точности.

Цикл измерений (10). Каждый шаг даёт независимый квантовый бит, эквивалентный выборке из Бернуллиевского распределения. Запускается цикл for i in 1..count, в котором каждый раз вызывается GeneratorBitov(). Каждый бит является статистически независимым и равновероятным, что соответствует идеальной модели Бернуллиевского распределения.

Добавление измеренного бита в массив (11). Вектор bits = [b_1, b_2, ..., b_n]. Полученный бит добавляется в конец массива bits. Порядок сохраняется, что позволяет при последующем преобразовании интерпретировать биты как старшие/младшие разряды числа.

Преобразование битового вектора в десятичное число (12). Выполняется преобразование массива bits в целое число с помощью функции ResultArrayAsInt(bits). Это стандартная операция перевода двоичной последовательности в десятичное представление.

Проверка условия: если полученное число > max, то происходит рекурсивный вызов функции (13). Иначе – возврат числа. Если сгенерированное число выходит за пределы диапазона [0, max], оно отбрасывается, и функция вызывается рекурсивно. При успешном результате число возвращается как выход функции.

Если условие выполняется (chislo > max), функция вызывается рекурсивно до получения допустимого значения (14). Это ключевой момент метода отбора: повторная генерация гарантирует, что ни одно число не будет иметь повышенной вероятности, что устраняет смещение (bias) и обеспечивает математически чистое равномерное распределение.

Если условие не выполняется (chislo ≤ max), число принимается как валидное и возвращается для дальнейшей обработки (15). Успешно сгенерированное число передаётся дальше по цепочке. Это число будет использовано для формирования символа пароля.

2.3. Формирование пароля (password(max))

Алгоритм 3-го этапа состоит из подэтапов 16–21 (рис. 4).

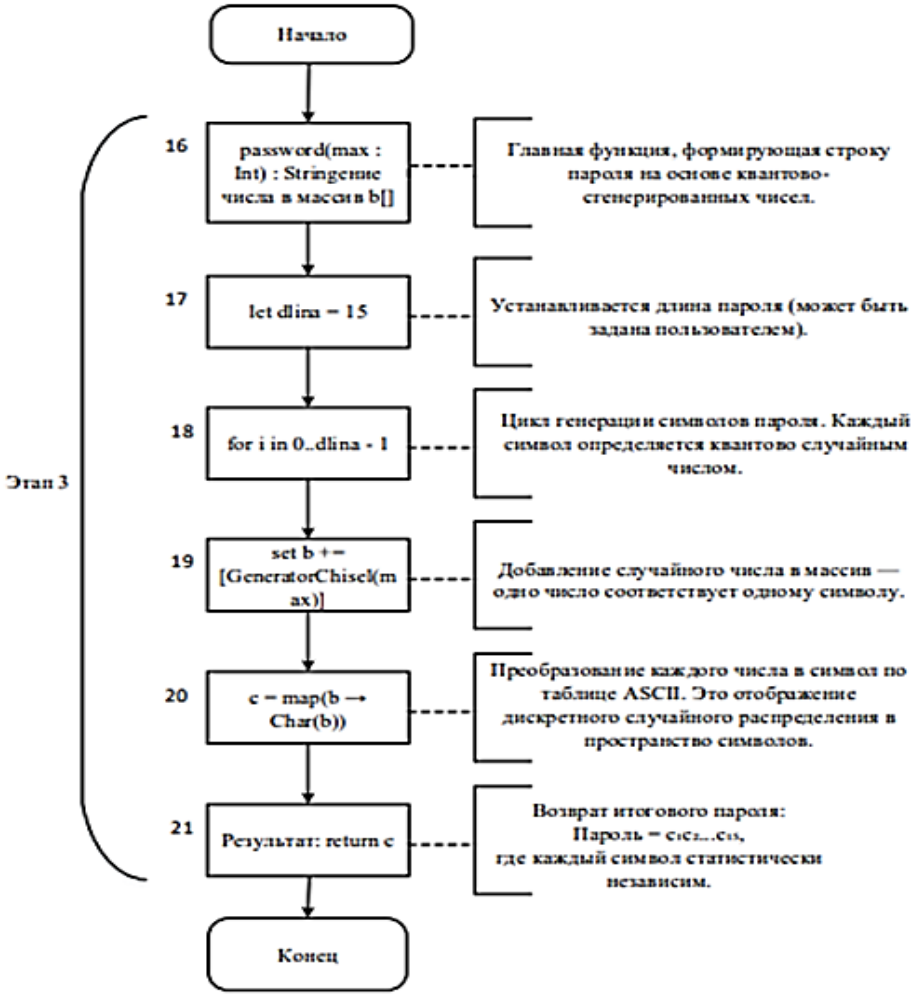

Рис. 4 Этап 3. Формирование пароля

Вызывается главная функция, формирующая строку пароля на основе квантово-сгенерированных чисел (16). Затем устанавливается длина пароля (например, let dlina = 15) (17). Определяется количество символов в итоговом пароле. Это может быть жёстко задано или передано пользователем как параметр. Длина напрямую влияет на энтропию и криптостойкость.

Цикл генерации символов пароля (18). Каждый символ определяется квантово-случайным числом. Запускается цикл for i in 0.dlina - 1, в котором для каждого символа пароля вызывается GeneratorChisel(max). Это обеспечивает независимую генерацию каждого символа.

Добавление случайного числа в массив – одно число соответствует одному символу (19). Полученное число добавляется в массив b [], где каждый элемент будет затем преобразован в символ. Это связывает числовую и символьную модель пароля.

Преобразование каждого числа в символ по таблице ASCII (20). Это отображение дискретного случайного распределения в пространство. Каждое число из массива b [] отображается в символ с помощью функции map($b \rightarrow \text{Char}(b)$), где Char() интерпретирует целое число как код символа в таблице ASCII (или другом наборе). Это обеспечивает однозначное соответствие между числами и символами.

Возврат итогового пароля, пароль = $c_0, c_1, ..., c_n$, где каждый символ статистически независим (21). Все сгенерированные символы объединяются в единую строку, которая возвращается как итоговый квантово-случайный пароль высокой энтропии и криптографической стойкости.

Для иллюстрации программной реализации разработанного квантового генератора паролей ниже представлен фрагмент исходного кода на языке Q#, реализующий ключевой этап алгоритма – генерацию случайного бита путём измерения кубита в суперпозиционном состоянии.

Приведённый код отражает прямое применение постулатов квантовой механики в программной среде и демонстрирует простоту и прозрачность архитектуры прототипа.

Фрагмент кода программы, описывающий этапы работы генератора, представлен ниже (листинг).

Листинг

Квантовый генератор паролей

```
000 operation GenerateRandomBit() : Result {
001   use qubit = Qubit();
002   H(qubit);
003   let result = M(qubit);
004   Reset(qubit);
005   return result;
006 }
007 operation GenerateRandomBitsArray(bitCount : Int) : Result[] {
008   mutable bits = new Result[bitCount];
009   for i in 0..bitCount - 1 {
010         set bits w/= i <- GenerateRandomBit();
011   }
012   return bits;
013 }
014 operation ResultArrayAsInt(bits : Result[]) : Int {
015 mutable result = 0;
016   for i in 0..Length(bits) - 1 {
017         if bits[i] == One {
018               set result += 1 <<< (Length(bits) - 1 - i);
019         }
020   }
021   return result;
022 }
023 operation GenerateQuantumPassword(length : Int) : String {
024     let     charset     =     "ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstu-
vwxyz0123456789!@#$%^&*()";
025 let charsetLength = Length(charset);
026 mutable password = "";
027 let bitsNeeded = BitSizeI(charsetLength - 1);
028 for i in 0..length - 1 {
029 mutable index = -1;
030 repeat {
031   let randomBits = GenerateRandomBitsArray(bitsNeeded);
032   let candidate = ResultArrayAsInt(randomBits);
033   if candidate < charsetLength {
034         set index = candidate;
035   }
```

```
036 } until (index >= 0);
037 set password += charset[index].ToString();
038 }
039 return password;
040 }
050 @EntryPoint()
051 operation Main() : Unit {
052 let passwordLength = 12;
053 let password = GenerateQuantumPassword(passwordLength);
054 Message($"Сгенерированный квантовый пароль: {password}");
055 }
```

Программным кодом, представленным на листинге, реализованы следующие задачи:

- создание массива, для хранения последовательности битов;
- измерение количества битов, необходимых для создания максимального числа, вводимого пользователем с клавиатуры с помощью встроенной функции BitSizeI;
- запуск цикла, направленного на заполнение массива битами до заданного значения;
- перевод массива в десятичное число с использованием функции ResultArrayAsInt;
- проверка на соответствие числа допустимому значению.

Далее происходит непосредственно квантовая генерация пароля:

- задается длина пароля;
- создается массив случайных чисел;
- создается переменная, где будет генерироваться пароль;
- запускается цикл, который заполняет созданный массив числами;
- в конечной операции числа массива будут преобразовываться согласно таблице ASCII, тем самым генерируя пароль.

Заключение

В данной работе систематизированы современные подходы к генерации квантовой случайности. Авторами была разработана и применена многоуровневая классификация квантовых генераторов случайных чисел (КГСЧ) по ключевым критериям: физической природе явления, типу измеряемой переменной (дискретная/непрерывная), уровню доверия и методам верификации, а также архитектуре реализации.

На основе проведённого сравнительного анализа по параметрам криптографической надёжности, скорости генерации, практической реализуемости и устойчивости ко внешним воздействиям был сделан вывод о перспективности дискретного подхода с бинарным измерением (аналогичного измерению поляризации одиночного фотона). Данный подход представляет собой баланс между высокой степенью истинной случайности, простотой интерпретации и возможностью программной реализации на доступных квантовых симуляторах.

Практическим результатом исследования стала разработка прототипа квантового генератора паролей на языке программирования Q#. Прототип реализует принцип фундаментальной непредсказуемости квантового измерения кубита в суперпозиционном состоянии, что позволяет генерировать криптографически стойкие пароли, принципиально защищённые от уязвимостей классических генераторов псевдослучайных последовательностей.

Предложенный алгоритм и его программная реализация демонстрируют практическую возможность использования принципов квантовой механики для задач защиты информации уже на современном этапе развития квантовых технологий.

СПИСОК ЛИТЕРАТУРЫ | REFERENCES

[Aci16] Acín A., Masanes L. Certified randomness in quantum physics // Nature. Vol. 540, no. 7632. Pp. 213–219. Dec. 2016. DOI: 10.1038/nature20119.

[Dyn08] Dynes J. F., Yuan Z. L., Sharpe A. W., and Shields A. J. A high speed, postprocessing free, quantum random number generator Appl // Phys. Lett. Vol. 93, no. 3. P. 031109. Jul. 2008. DOI: 10.1063/1.2961000. EDN: HZPIOU.

[Her17] Herrero-Collantes M., Garcia-Escartin J. C. Quantum random number generators // Rev. Mod. Phys. Vol. 89, no. 1. P. 015004. Jan. 2017. DOI: 10.1103/RevModPhys.89.015004. EDN: YYBPPP.

[Jac21] Jacak M.M., Jóźwiak P., Niemczuk J. et al. Quantum generators of random numbers // Sci Rep 11. 16108. (2021). DOI: 10.1038/s41598-021-95388-7. DSN: CFWGXV.

[Jof11] Jofre M. et al. True random numbers from amplified quantum vacuum // Optics Express. 19(21). 20665–20672. DOI: 10.1364/oe.19.020665.

[Man22] Mannalath V., Mishra S., Pathak A. A comprehensive review of quantum random number generators: Concepts, classification and the origin of randomness. 2022. arXiv. https://arxiv.org/abs/2203.00261.

[Max16] Ma X., et al. Quantum random number generation // npj Quantum Information. 2016. 2. 16021. DOI: 10.1038/npjqi.2016.21. EDN: XRQVBC.

[Kai20] Kaiser F. et al. Integrated photonic quantum random number generator // APL Photonics. Vol. 5, no. 11. P. 116104. Nov. 2020. DOI: 10.1063/5.0022362.

[Kel98] Kelsey J., Schneier B., Wagner D., Hall C. Cryptanalytic Attacks on Pseudorandom Number Generators // Fast Software Encryption. FSE 1998. Lecture Notes in Computer Science. Vol. 1372. Springer, Berlin, Heidelberg. P. 168–188. DOI: 10.1007/3-540-69710-1_12.

[Son16] Turan M. S., et al. Recommendation for the entropy sources used for random bit generation (Draft), NIST Special Publication (SP) 800-90B, National Institute of Standards and Technology, Gaithersburg, MD, 2016. DOI: 10.6028/NIST.SP.800-90B.

[Wan22] Wang Y. et al. Spin-based quantum random number generation using NV centers in diamond // Quantum Sci. Technol.. Vol. 7, no. 3. P. 035021. Jul. 2022. DOI: 10.1088/2058-9565/ac6d9e.

[Вил25] Вилаков Н. В., Бочаров М. И. Способы защиты криптовалютных блокчейн-систем и систематизация угроз // СИИТ. 2025. Т. 7, № 2(21). С. 143-154. DOI: 10.54708/2658-5014-SIIT-2025-no2-p147. EDN: CXNIYP. [[Vilakov N. V., Bocharov M. I. Methods of protecting cryptocurrency blockchain systems and systematization of threats // SIIT. 2025. Vol. 7, No. 2(21). P. 143-154. (In Russian).]]

[Жиз23] Жизан Н. Квантовая случайность: Нелокальность, телепортация и другие квантовые чудеса. Альпина нон-фикшн, 2023. ISBN: 978-5-9614-4088-1. [[Zhizan N. Quantum Randomness: Nonlocality, Teleportation, and Other Quantum Miracles. Alpina Non-Fiction, 2023. (In Russian).]]

[Куч24] Кучкарова Н. В. Оценка актуальных угроз и уязвимостей объектов критической информационной инфраструктуры с использованием технологий интеллектуального анализа текстов // СИИТ. 2024. Т. 6, № 2(17). С. 50-65. DOI: 10.54708/2658-5014-SIIT-2024-no2-p50. EDN: NLDWBE. [[Kuchkarova N. V. Assessment of current threats and vulnerabilities of critical information infrastructure objects using text mining technologies // SIIT. 2024. Vol. 6, No. 2(17). P. 50-65. (In Russian).]]

[Сле17] Слеповичев И. И. Генераторы псевдослучайных чисел: Учебное пособие. Саратов: СГУ, 2017. Т. 117. [[Slepovichev I.I. Pseudorandom number generators (Study Guide). Saratov: Saratov State Univ., 2017, vol. 117. (In Russian).]]

[Фай16] Файер М. Абсолютный минимум. Изд. дом «Питер», 2016. 384 с. ISBN 978-5-496-01069-6. [[Fayer M. The Absolute Minimum. Piter, 2016. ISBN 978-5-496-01069-6. (In Russian).]]

[Хай25] Хайруллин Э. Р., Вульфин А. М., Васильев В. И., Мандовен С. А. Нейросетевая система обнаружения сетевых атак // СИИТ. 2025. Т. 7, № 1(20). С. 105-112. DOI: 10.54708/2658-5014-SIIT-2025-no1-p105. EDN: GVIQLO. [[Khairullin E. R., Vulfin A. M., Vasiliev V. I., Mandoven S. A. Neural network system for detecting network attacks // SIIT. 2025. Т. 7, No. 1(20). pp. 105-112. (In Russian).]]

ОБ АВТОРАХ | ABOUT THE AUTHORS

КУЧКАРОВА Наиля Вакилевна
Уфимский университет науки и технологий, Россия.
nailya_kuchkarov@mail.ru.
Доц. каф. вычислительной техники и защиты информации. Дипл. магистр по информатике и выч. технике (Уфимск. гос. авиац. техн. ун-т, 2020), канд.техн.наук (Уфимск. ун-т науки и технологий, 2024). Иссл. в обл. информационной безопасности.

ЕРОШКИН Андрей Дмитриевич
Уфимский университет науки и технологий, Россия.
aderoshkin2005@gmail.com.
Студент спец. «Информационная безопасность».

KUCHKAROVA Nailya Vakilevna
Ufa University of Science and Technology, Russia.
nailya_kuchrarov@mail.ru.
Assoc. Prof., Dept. of Computer Science and Information Security. Master's degree in computer science and computational engineering (Ufa State Aviation Technical University, 2020), Ph.D. (Eng.), Ufa University of Science and Technology, 2024. Research in the field of information security.

EROSHKIN Andrey Dmitrievich
Ufa University of Science and Technology, Russia.
aderoshkin2005@gmail.com.
Student of the specialty "Information security".

МЕТАДАННЫЕ | METADATA

Заглавие: Применение квантовых генераторов случайных чисел в системах защиты информации: анализ и прототипирование.

Авторы: Кучкарова Н. В., Ерошкин А. Д.

Title: Application of Quantum Random Number Generators in Information Security Systems: Analysis and Prototyping l.

Authors: Kuchkarova N.V., Eroshkin A.D.

Аннотация: В представленной работе проведен анализ существующих квантовых генераторов случайных чисел (КГСЧ) в контексте их использования в средствах защиты информации. Проведена классификация КГСЧ по трем основным критериям: природе квантового явления, уровню доверия и проверки, архитектуре и уровню обработки данных. Сделаны выводы о возможности применения определенных видов КГСЧ в задачах обеспечения информационной безопасности. В качестве примера реализации КГСЧ разработан прототип квантового генератора паролей, обеспечивающий криптографически стойкую случайность за счёт фундаментальной непредсказуемости квантовых измерений. Алгоритм генерации реализован на языке квантового программирования Q# и использует суперпозицию кубитов с последующим проективным измерением в вычислительном базисе.

Ключевые слова: Генератор псевдослучайных чисел; квантовый компьютер; средства защиты информации; квантовый генератор случайных паролей.

Язык: Русский.

Статья поступила в редакцию 26 декабря 2025 г.

Abstract: The study conducts an analysis of contemporary quantum random number generators (QRNGs) concerning their potential use in data protection means. A threefold classification of QRNGs is established, focusing on the core quantum physical principle, the trust and verification paradigm, and the implementation architecture coupled with the data processing stage. This analysis yields findings on the feasibility of deploying specific QRNG types for information assurance objectives. To exemplify a QRNG implementation, a prototype quantum-powered password generator is engineered. Its operation ensures cryptographically robust randomness by harnessing the inherent unpredictability of quantum measurement outcomes. The core algorithm, coded in the Q# quantum programming language, employs the preparation of qubits in a superposition state and their subsequent projective measurement in the computational basis.

Key words: Pseudorandom Number Generator (PRNG); Quantum Computer; Information Security Tools; Quantum Random Password Generator (QRPG).

Language: Russian.

The article was received by the editors on 26 December 2025.