

UDC 004.387

<https://doi.org/10.32362/2500-316X-2025-13-6-25-46>

EDN XEUFSE



RESEARCH ARTICLE

Approach for identifying the optimal set of qubits of quantum computing devices based on a model for generating binary random sequences

Andrey V. Korolkov, Andrey A. Kryuchkov[@]*MIREA – Russian Technological University, Moscow, 119454 Russia**[@] Corresponding author, e-mail: kryuchkov_a@mirea.ru*

• Submitted: 23.12.2024 • Revised: 15.05.2025 • Accepted: 06.10.2025

Abstract

Objectives. The absence of error-resistant quantum computers, coupled with the challenges associated with providing unrestricted and fully operational physical access to cloud quantum computing systems, prompts a critical examination of the necessity to develop universal and independent methods for evaluating and verifying cloud quantum computers. A promising approach involves evaluating the capabilities of a quantum computer in relation to its effectiveness in addressing specific challenges encountered in the assessment of information security systems. A potential test for ascertaining the performance and computational quality of a quantum computing device (QCD) is based on a model designed to generate a random binary sequence. By analyzing this sequence, insights can be obtained into the accuracy and reliability of the quantum register under study. The paper presents a software program developed for simulating the operation of a quantum random number generator.

Methods. The software implementation for interacting with cloud quantum computers was performed using the Qiskit open-source software kit. The graphical user interface of the software package was developed using a Qt5 cross-platform set of tools and widgets for creating applications. The analysis of the generated binary sequence was performed using a set of statistical tests NIST STS¹.

Results. The developed software package provides users with a graphical interface for conducting an analysis of a cloud QCD to identify the optimal and most error-resistant set of qubits. The findings from experiments conducted on three cloud quantum computing devices are reported.

Conclusions. The proposed approach, which is constrained by limitations of computing power and duration of access to cloud-based QCD, imposes minimal demands on the productive capabilities of the quantum system. It offers clear and unequivocally interpretable insights into the technical characteristics of a cloud quantum computer, while also being reproducible, easily scalable, and universally applicable.

Keywords: quantum computer, quantum computer device, random number generator, benchmarking

¹ The National Institute of Standards and Technology, Statistical Test Suite.

For citation: Korolkov A.V., Kryuchkov A.A. Approach for identifying the optimal set of qubits of quantum computing devices based on a model for generating binary random sequences. *Russian Technological Journal*. 2025;13(6):25–46. <https://doi.org/10.32362/2500-316X-2025-13-6-25-46>, <https://www.elibrary.ru/XEUFSE>

Financial disclosure: The authors have no financial or proprietary interest in any material or method mentioned.

The authors declare no conflicts of interest.

НАУЧНАЯ СТАТЬЯ

Подход к выявлению оптимального набора кубит квантовых вычислительных устройств на примере модели генерации случайных двоичных последовательностей

А.В. Корольков, А.А. Крючков [®]

МИРЭА – Российский технологический университет, Москва, 119454 Россия

[®] Автор для переписки, e-mail: kryuchkov_a@mirea.ru

• Поступила: 23.12.2024 • Доработана: 15.05.2025 • Принята к опубликованию: 06.10.2025

Резюме

Цели. Отсутствие квантовых компьютеров, устойчивых к ошибкам, а также невозможность обеспечить беспрепятственный и полнофункциональный физический доступ к облачным квантовым системам поднимает актуальный вопрос о необходимости разработки методов оценки и верификации облачных квантовых компьютеров. Авторам видится перспективным использование подхода к анализу возможностей квантового процессора в контексте его применимости для решения некоторых задач, возникающих при оценке систем защиты информации. Одним из примеров теста квантового вычислительного устройства (КВУ) на предмет выявления уровня производительности и качества вычислений может послужить модель генерации случайной двоичной последовательности, анализ которой предоставляет информацию о корректности и надежности исследуемого квантового регистра. Цель работы заключается в разработке программного комплекса, с помощью которого представляется возможным моделирование работы КВУ в режиме квантового генератора случайных чисел.

Методы. Программная реализация по взаимодействию с облачными квантовыми компьютерами выполнена с использованием библиотеки Qiskit. Интерфейс программного комплекса реализован средствами Qt5, кроссплатформенного набора инструментов и виджетов для создания графических приложений. Анализ генерируемой двоичной последовательности выполнен статистическими тестами NIST STS².

Результаты. Разработан программный комплекс, предоставляющий пользователю в графическом режиме возможность выполнения опционального исследования некоторых облачных квантовых компьютеров на предмет выявления оптимального и наиболее устойчивого к ошибкам набора кубит. Приведены результаты экспериментов на трех облачных КВУ.

Выводы. В условиях накладываемых ограничений на вычислительные мощности и время использования облачных КВУ предложенный подход отличается минимальными требованиями к производительности устройства, предоставляет наглядные и однозначно-интерпретируемые сведения об исследуемых технических характеристиках квантового компьютера, является воспроизводимым, легко масштабируемым и универсальным.

² The National Institute of Standards and Technology, Statistical Test Suite – Национальный институт стандартов и технологий США, набор статистических тестов.

Ключевые слова: квантовый компьютер, квантовое вычислительное устройство, генератор случайных чисел, оценка производительности

Для цитирования: Корольков А.В., Крючков А.А. Подход к выявлению оптимального набора кубит квантовых вычислительных устройств на примере модели генерации случайных двоичных последовательностей. *Russian Technological Journal*. 2025;13(6):25–46. <https://doi.org/10.32362/2500-316X-2025-13-6-25-46>, <https://www.elibrary.ru/XEUFSE>

Прозрачность финансовой деятельности: Авторы не имеют финансовой заинтересованности в представленных материалах или методах.

Авторы заявляют об отсутствии конфликта интересов.

INTRODUCTION

In recent years, quantum processors have consistently demonstrated improvements in their computational and technological capabilities. This progress necessitates a straightforward and efficient method for tracking and recording the evolution of quantum computing technology, from intermediate stages to its final form. Such a mechanism, whether used alone or as part of a broader testing suite, offers a way to leverage quantum computing devices (QCD) for solving random number generation (RNG) problems. This approach provides a dependable and straightforward method for benchmarking the performance and reliability of the device under study.

QCD BENCHMARKING

Currently, there is no globally accepted standard for evaluating QCD computing power within the international scientific community. Nevertheless, the Institute of Engineers and Electrical Engineering (IEEE)³, along with the United States Defense Advanced Research Projects Agency (DARPA)⁴, are actively working on establishing such a framework. These projects, which were both initiated in 2021, anticipate releasing preliminary findings in the near future.

The absence of a standardized framework in quantum computing is primarily due to the early developmental phase of the industry; the first conceptualization of a 1-qubit cloud quantum computer occurred only in 2016. However, results that hold practical significance are continuously advancing, with QCD testing methodologies being refined, broadened, and updated each year to keep pace with current innovations and the technological viability of specific metrics. The previously established metric known as quantum volume is an unreliable indicator for modern quantum devices due to its failure to accurately reflect the true capabilities of various quantum processors. As a result, it has been replaced by newer

metrics that provide a more effective evaluation of the quality and performance of these processors. For further insights into these technologies and the progression of QCD assessment methods, please refer to [1–4].

To confirm the significance of the issue and to ascertain the applicability of the RNG model in analyzing the QCD capabilities, we will outline several key aspects of widely recognized methods for evaluating their computational potential.

Quantum computers are categorized into four epochs of technological advancement, the most recent of which is referred to as a noisy intermediate-scale quantum (NISQ) system. This stage, which represents the foundational level in the hierarchy, pertains to devices featuring a limited number of qubits within the processor. However, NISQ systems lack error correction codes and exhibit a high vulnerability to disturbances resulting from environmental interactions.

The second essential aspect of quantum computer analysis pertains to the execution technology. Contemporary quantum processors can be developed using photons, ions, superconductors, or neutral atoms. When comparing devices based on various physical principles, the foundational components of a quantum computer play a critical role. For instance, superconducting QCDs typically operate at significantly higher speeds than their ionic counterparts, which can render performance comparisons between these types of devices regarding quantum circuit speed less meaningful. Nevertheless, when assessing speed characteristics within a single physical implementation, even across different versions, such evaluations remain valuable. They demonstrate the extent to which changes to QCD affect the processing speed of a quantum circuit (Fig. 1).

The methods for studying QCD can differ based on the specific application. This may involve examining the overall system such as the circuit layer operation per second (CLOPS) [6], which measures the number of layers processed in a quantum circuit each second, or focusing on particular functional aspects, like the error per layer gate (EPLG) [7]. Analysis can additionally, be conducted at the device level by investigating independent components, such as specific gates used in conjunction with designated qubits [8]. This approach entails applying a certain set of gates to quantum states to evaluate the quality of the quantum chain nodes.

³ P7131. Standard for Quantum Computing Performance Metrics & Performance Benchmarking. IEEE. 2021. <https://sagroups.ieee.org/7131/>. Accessed October 06, 2025.

⁴ Quantifying Utility of QC, DARPA. 2021. <https://www.darpa.mil/news-events/2021-04-02>. Accessed October 06, 2025.

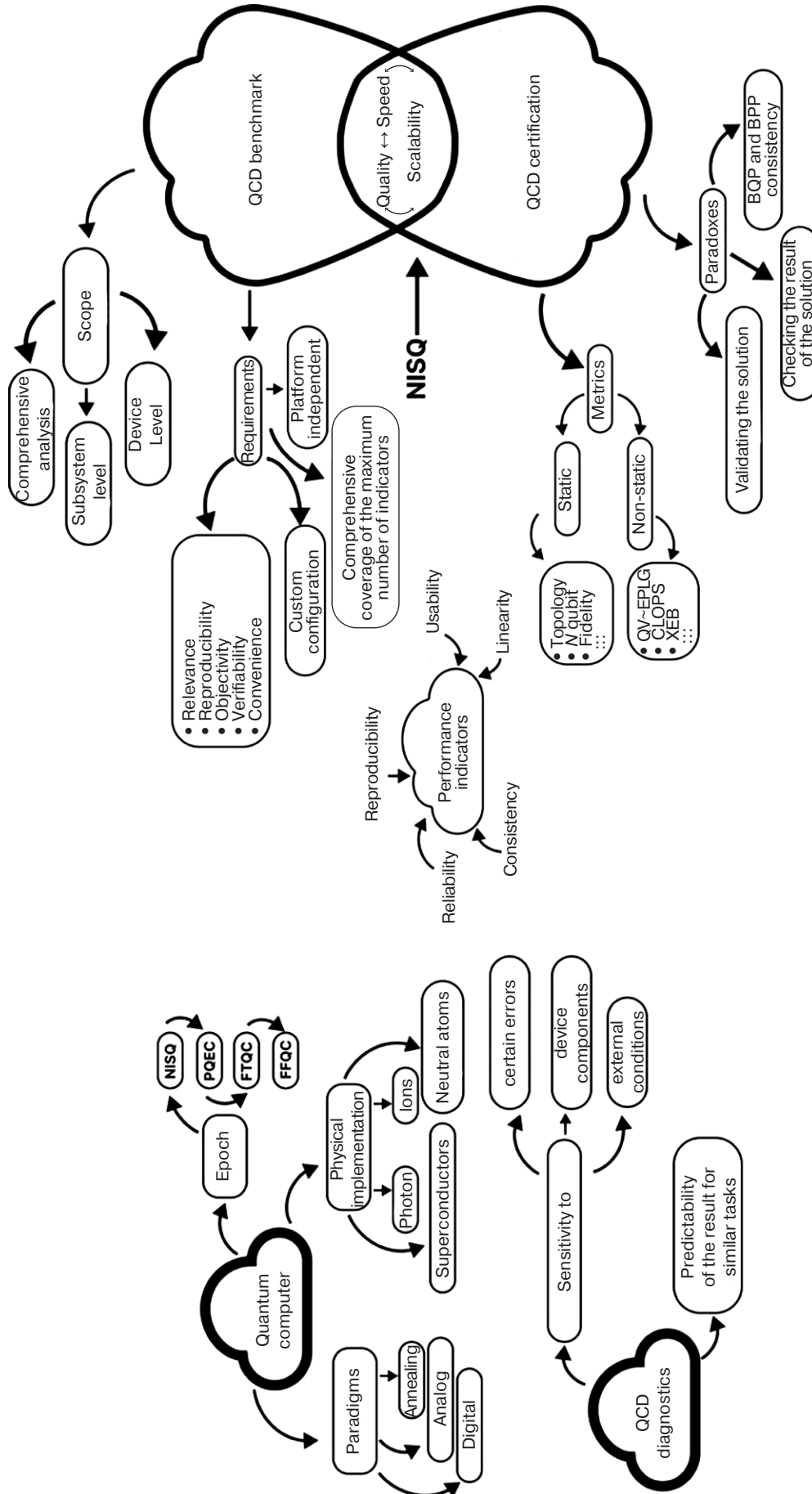


Fig. 1. Flow diagram of QCD performance tools [5]. PQEC—Partially Quantum Error Corrected, i.e., partial correction of errors that occur on the QCD; FTQC—fault tolerant quantum computers; FFQC—fully functional quantum computers; XEB—cross-entropy benchmarking

It is essential to focus on the issue of certifying quantum computers or verifying the accuracy of their computed results. The central paradox lies in the fact that, while operating within the classical realm (where Bounded-Error Probabilistic Polynomial Time denotes polynomial time with a constraint on error probability), assessing the reliability of a quantum algorithm (represented by Bounded-Error Quantum Polynomial Time) presents a complex challenge. Here the main issue involves verifying that the program executed on the quantum computer has been solved correctly by the quantum device. In other words, what methods can be employed to validate the classical results obtained through quantum computation? In certain instances, such as with the Shor algorithm [9], the expected result may be predetermined to eliminate the potential for multiple interpretations of the final outcome. However, the approach to verification becomes less straightforward in cases where the results are unique and cannot be classically simulated for comparison with expected results.

The benchmark and validation of quantum computers in the NISQ era must adhere to three interrelated imperatives: quality, speed, and scalability. For instance, the quantum volume metric has become less relevant due to the challenges involved in meeting scalability requirements. As a result, manufacturers of quantum devices no longer include it in the essential list of metrics that indicate the computational capabilities of quantum computing devices.

The final and essential aspect of analyzing quantum computers involves viewing a quantum processor as a unified device. In its canonical form, the execution of a quantum algorithm is segmented into three stages: initializing the quantum register, executing the quantum algorithm, and analyzing the results, which are presented as classical information in binary format.

Current QCDs are made accessible to regular users via cloud services. This presents a crucial challenge, since individual researchers, who are limited to observing the output of a distant quantum computer (functioning as a “black box”), cannot be certain that a classical QCD simulator is not being disguised as a cloud-based quantum system.

Upon initial observation, it appears reasonable to conclude that no conventional computing device or complex of such devices possesses the capability to effectively simulate the execution of a multi-bit quantum program featuring a substantial depth of quantum circuit operations. Nevertheless, it is important to remember that this is a working assumption rather than a definitive rule. Additionally, the absence of physical access to cloud-based QCD and the current challenges in verifying classical information generated by quantum computers create limitations and opportunities for speculation,

sometimes bordering on unfounded assumptions, regarding the characteristics and capabilities of devices marketed as quantum computers.

Given these considerations, the present authors propose a study focused on modeling the creation of binary random sequences (RSs) using cloud-based quantum computers. This study aims to provide fundamental and practical insights into the performance and technological capabilities of these devices.

METHOD FOR SIMULATING RNG ON A QUANTUM COMPUTER

Along with inherent errors stemming from hardware limitations, the accuracy and dependability of solutions derived from modern QCD applied to real-world problems are currently hampered by external factors. These hardware limitations arise from the current technological imperfections and the intricate process of tuning the entire quantum system. However, even with the current limitations of modern quantum computers, it may be worthwhile to explore the potential for analyzing the computational capabilities and quantum characteristics of QCD within the context of modeling RS generation.

The creation of binary RSs and evaluation of the technical capabilities of a quantum computer have additional practical significance due to the need to safeguard information obtained through cryptography. For this reason, the use of sequences generated on any cloud device, regardless of its type, is strictly forbidden. At the present time, the results generated by cloud quantum computers within RNGs are primarily suitable for research purposes such as simulating random processes. Nevertheless, exploring the potential of QCD in RNG applications can lay the groundwork for future research when technological advancements allow for more precise localization of quantum systems.

Contemporary RNGs are categorized into three primary types distinguished by their technical execution and implementation methods: software, physical, and combined. Software RNGs, considered the least reliable, are susceptible to vulnerabilities such as predictable output patterns (periods) in pseudo-RSs (PRS) and weaknesses stemming from the initialization vector, which, if compromised, could jeopardize the security of binary data. The randomness of output from physical and RNGs is often lacking in robust mathematical justification. Furthermore, such generators can be susceptible to external factors such as temperature fluctuations, magnetic radiation, or radio wave interference.

Although quantum random number generators (QRNG) are categorized as physical RNGs, they stand apart due to their robust mathematical

foundation for ensuring the randomness of the generated RSs. Examples of QRNG devices are described in [10–12].

When comparing a portable QRNG to a quantum computer used as an RNG, several factors come into play. An independent quantum RS generation device offers a clear advantage in terms of the security of the generated sequence, particularly when employed within a “controlled zone.” However, it remains susceptible to environmental factors, which could degrade the randomness quality of the output bit string. On the other hand, a cloud-based quantum computer operates within an ideally isolated environment to minimize the risk of external noise and interference while preserving the intrinsic properties of quantum computation. Yet, due to its remote nature, concerns about security are inevitable in such scenarios.

We will now consider the process of simulating the generation of a random bit of classical information on a quantum computer according to the assumption that there are no adverse effects on QCD caused by the hardware or the physical environment of the device. Several studies exploring this topic are referenced in works [13–16].

At the heart of any quantum processor is the qubit, which is the fundamental unit. A qubit can be represented as a unit vector in a two-dimensional complex vector space, with the basis being defined by the orthogonal vectors $|0\rangle$ and $|1\rangle$.

$$|\psi\rangle = c_1|0\rangle + c_2|1\rangle, \quad (1)$$

where c_1, c_2 are arbitrary complex numbers, the probabilities of the quantum states are given by:

$$|c_1|^2 + |c_2|^2 = 1. \quad (2)$$

To generate a random bit using a single qubit on a quantum processor, it is both necessary and sufficient to prepare the qubit in a superposition state. In this state, the probabilities of measuring either “0” or “1” will both be equal to 0.5. Achieving this requires applying the Walsh–Hadamard transform to the quantum state, which is represented by the following matrix:

$$\mathbf{H} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \quad (3)$$

Applying the Hadamard gate to a single qubit results in a quantum state superposition given by the following expression:

$$\begin{aligned} \mathbf{H}:|0\rangle &\rightarrow \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \\ \mathbf{H}:|1\rangle &\rightarrow \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle). \end{aligned} \quad (4)$$

As seen from (4), the initial state of the RNG simulation is irrelevant irrespective of whether the qubit begins in state $|0\rangle$ or $|1\rangle$. This ensures the effective creation of a superposition state to produce equal probabilities for the quantum bit to collapse into a classical state of “0” or “1” upon measurement. To generate an RS of a specific length, operation (4) should be executed iteratively for the desired number of cycles and the results stored as a single binary string.

The proposed QRNG model represents a theoretical process for generating RS under optimal conditions with the aim of producing an RS with a balanced distribution of “0” and “1.” However, achieving these ideal conditions is currently unfeasible due to the limitations of current quantum equipment, which have inherent technological imperfections. These limitations are particularly evident in the era of NISQ devices used by manufacturers of quantum processors.

Contemporary developers of quantum computers are closely focused on addressing the technical limitations arising from hardware imperfections and external environmental influences. In order to enhance the performance of quantum algorithms, it is recommended that error correction codes be implemented. These codes consolidate multiple physical qubits into a single logical qubit to ensure greater reliability and accuracy.

When working with a logical qubit constructed from three physical qubits, such as when transmitting the quantum state $|0\rangle$, the receiver should receive the encoded state $|000\rangle$ (referred to as the codeword). If one of the bits is flipped (bit-flip)—for example, the first one resulting in $|100\rangle$ —an error can be detected by the system. However, if more than two bits are flipped, it becomes impossible to identify any discrepancy in the final state. For this reason, it is essential in such scenarios to carefully determine the code distance $d = 2t + 1$, where d represents the maximum number of errors that cannot be detected, and t corresponds to the number of errors that can be identified.

The most widely recognized quantum error correction codes today include⁵ repetition-, Shor-, surface-, Steane-, and Hastings—Haah codes.

However, for the specific task at hand, employing error correction codes proves to be impractical; in order to generate a genuinely RS, direct interaction with quantum states at the physical level is required.

Contemporary quantum computers possess various parameters that enable preliminary assessment of their computational capabilities and technical attributes. The information presented in Table 1, which highlights the key characteristics specific to quantum processors

⁵ Introduction to quantum error correction. Microsoft Quantum Azure. 2024. <https://learn.microsoft.com/en-us/azure/quantum/concepts-error-correction>. Accessed October 06, 2025.

Table 1. Impact of superconducting QCD parameters on the performance of quantum circuit execution steps

Quantum circuit execution step	QCD Parameter											
	T_1	T_2	Frequency	Anharmonicity	Readout length	Readout error	meas0 prep1	meas1 prep0	(ID) error	(Pauli-X) error	(sx) error	ECR error
Qubit initialization	–	–	–	–	–	–	+	+	–	–	–	–
Quantum scheme implementation	+	+	+	+	–	–	–	–	+	+	+	+
Register measurement	–	–	–	–	+	+	–	–	–	–	–	–

Note: T_1 and T_2 represent the relaxation and dephasing times of quantum states; Frequency and Anharmonicity denote the qubit's frequency and anharmonicity (the difference in energy between the first and second excited states, expressed in Hertz); Readout error/length refers to error/time for reading the qubit state; (ID) error, (Pauli-X) error, and (sx) error represent errors in initialization, inversion, and qubit superposition, respectively. ECR error stands for a two-qubit gate error.

built using superconducting technology, serves as a foundational reference for upcoming experiments involving IBM quantum processors based on superconductors.⁶

The values presented in Table 1, which are not fully applicable to all three stages of program execution on a quantum computer, only influence specific periods of operation within the quantum circuit on a remote device. Furthermore, not all of these values need to be considered during the initial assessment of the test qubit's quality. To determine which parameters can be disregarded, it is necessary to construct a quantum scheme that clearly highlights the relevant characteristics of the selected qubits.

The RS generator on the quantum computer can be implemented as illustrated in Fig. 2. In Fig. 2a, the original quantum circuit designed by the user is presented, while Fig. 2b shows the same circuit after it has been compiled and decomposed into a version compatible with the native gates of the quantum computer executing the program. These native gates represent the fundamental operations specific to the hardware's architecture. Notably, the same circuit implemented on a QCD with a different underlying physical design and native gate set may differ substantially from the intended configuration.

Considering the quantum scheme post-compilation (Fig. 2b) alongside the available QCD parameters outlined in Table 1, the key factors to be addressed include T_1 , T_2 , Readout error, and (sx) error.

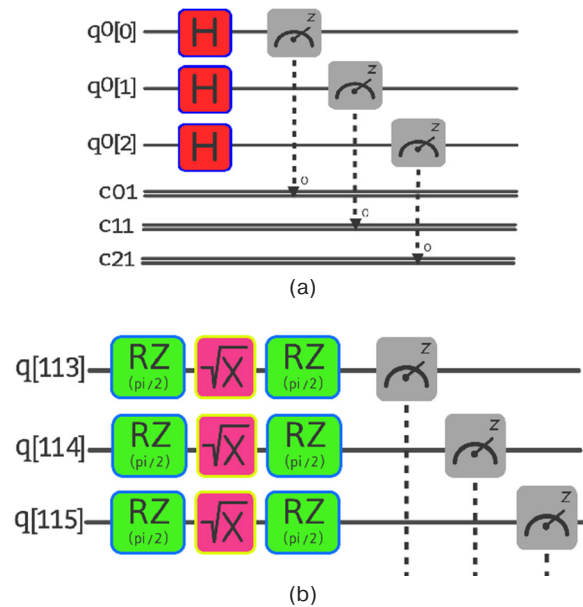


Fig. 2. Three-qubit QRNG circuit for superconducting QCD: (a) High-level representation; (b) Base gate decomposition of QCD (low-level representation). q0[0], q0[1], q0[2], q[113], q[114], and q[115] represent qubits while c11, c12, and c21 are classic bits. H are Hadamard gates; “speedometer”—qubit measurement operation; RZ—quantum state rotation at an angle of $\pi/2$; $\sqrt{X}$ —one of the Hadamard operation components [17]

Since their minimal variance across all qubits utilized in the experimental research renders them insignificant, the properties of Frequency, Anharmonicity, and Readout length are intentionally

⁶ IBM Quantum Platform. <https://quantum.ibm.com/>. Accessed May 15, 2025.

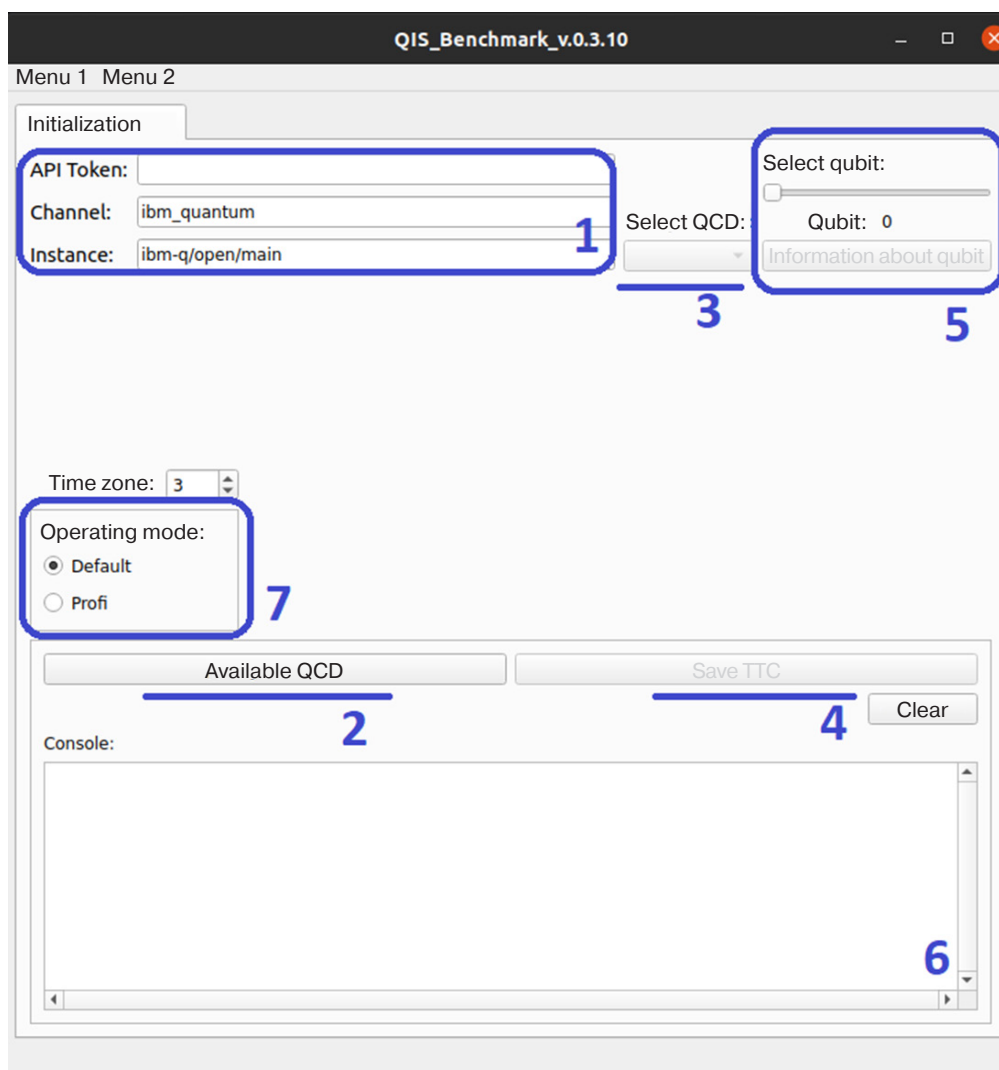


Fig. 3. QISs main window. TTC—tactical and technical characteristics

excluded. Similarly, the parameters `meas0 prep1` and `meas1 prep0` are irrelevant in this context, given that the “H” gate is applied to qubits in an idle state without prior initialization. Furthermore, as stated in Eq. (4), the operational output of the programmable circuit can be assumed to be independent of the qubit’s initial state.⁷ Among the remaining error sources affecting the relevant qubits, only (sx) error remains significant, as indicated by the diagram in Fig. 2b. According to the specifications provided by the cloud QCD manufacturer, the gate responsible for RZ quantum state rotation has a zero-error value and is can thus be omitted from consideration.

Having developed the RS generation model and identified the key parameters of the cloud QCD that require consideration when analyzing the results, we will proceed with a practical series of QRNG experiments on a cloud-based quantum computer.

EXPERIMENTAL QRNG IMPLEMENTATION ON A CLOUD-BASED QCD

In order to conduct permit the conducting of various experiments, IBM quantum computers are made accessible to users via the company’s cloud-based services in the public domain.

To enhance the efficiency, autonomy, adaptability, and ease of use when working with these quantum devices, the present authors have developed an application leveraging the open-source library Qiskit. This application is designed with the graphical interface of the Qt5 framework. The main window of the program, named *Quantum Information Security (QISs)* [18], is illustrated in Fig. 3.

To initiate the program for establishing a remote connection with the cloud QCD in Area 1, the operator needs to input the initialization data,

⁷ As a rule, the quantum register is “reset” to the state $|00...0\rangle$ by default.

including a personal API⁸-token that is uniquely tied to each user. Following this, the operator can proceed to retrieve details about the accessible cloud QCD in Area 2. Currently, IBM provides the capability to run quantum circuits on 12 devices, 9 of which cater to commercial usage. However, users without a paid subscription are limited to conducting research on only three cloud systems. Once information about the user's accessible QCD is obtained, the drop-down menu in Area 3 will populate to enable access to both general information in Area 4 and more specific details in Area 5 about the chosen QCD or individual qubits.

The text window serves to display the program's progress (Area 6). For less experienced users, the operating mode selection (Area 7) is tailored to simplify interaction with both the application and the cloud-based quantum computer.

Upon switching to Profi mode, the program unveils additional windows, as illustrated in Fig. 4.

The INIT-IDLE mode⁹ (Fig. 4, left) is designed to permit the effective configuration of study qubits. Once the QCD is selected (Area 3), the user specifies the qubits to utilize (Area 4) and configures the initialization gate (Area 5) along with the Hadamard gate settings (Area 6). Subsequently, instructions are set for the quantum computer, and the circuit is submitted for execution (Area 7). If it is not feasible to wait for the QCD to complete its online computations, users can download the program results using the JobID (Area 8). A text window provides real-time updates on the program progress (Area 9).

The QRNG mode (Fig. 4) offers multiple functionalities (Area 1). Among these, the cloud QCD selection is only required for the configuration unit responsible for QRNG operations (Area 3). As part of the training process, users can generate quantum circuits and analyze corresponding data by executing a QRNG simulation on a quantum computer simulator (Area 4). Area 5 focuses on testing the generated RS and PRS to detect any statistical dependencies within their binary strings. Additionally, the QRNG tab allows results to be stored offline (Area 7). Each step of the application workflow is detailed within the text box located in Area 6.

A key parameter is Alpha, the significance level. Within the RNG process, the sequence produced by each qubit undergoes analysis to verify its adherence to a uniform distribution of "0" and "1." To achieve this, the frequency bit test, representing a component

of the NIST STS¹⁰ suite of statistical tests designed for evaluating binary sequences¹¹, is employed. After testing, a P_{value} is provided, calculated using statistical methods. This value ranges from 0 (indicating a strong bias) to 1 (suggesting an evenly probable distribution between "0" and "1"). It should be noted that P_{value} does not always fall at the extremes, meaning the user's judgment plays a role in determining randomness or uniformity. The user independently selects the significance level Alpha, which serves as a benchmark when comparing statistical metrics.

The NIST STS test documentation suggests setting Alpha within a range of 0.001 to 0.01. When P_{value} exceeds the chosen significance level ($P_{\text{value}} > \alpha$), the sequence is considered random with confidence levels of 99.9% and 99%, respectively. The selected Alpha value directly affects type I errors, leading to false positives and the incorrect rejection of the null hypothesis. For instance, with $\alpha = 0.01$, approximately one in every 100 RSs will be mistakenly rejected.

In the QISs application, a frequency bit test serves as a method for filtering quantum states into two categories: "bad" and "good" qubits. This classification is based on a threshold value, termed "Alpha," which is specified by the user to evaluate the suitability of quantum states. While the separation of qubits is subject to conditional constraints, in certain scenarios, it becomes both evident and essential.

To conduct a bitwise frequency test on a binary sequence $\{x\}_{i=1}^n$, the statistical distribution of "0" and "1" values is determined according to the following rule:

$$s_{\text{obs}} = \frac{|S_n|}{\sqrt{n}}, \quad (5)$$

where at $X_i = 2x_i - 1$:

$$S_n = \sum_{i=1}^n X_i. \quad (6)$$

Based on the results obtained, an additional error function is used to calculate the P_{value} . The error function has the following form:

$$\text{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} e^{-t^2} dt. \quad (7)$$

⁸ Application Programming Interface.

⁹ INIT is initialization; IDLE stands for on "idle" (initialization). The "INIT-IDLE" mode is not considered in the paper.

¹⁰ The National Institute of Standards and Technology, Statistical Test Suite.

¹¹ NIST SP 800-22. <https://csrc.nist.gov/projects/random-bit-generation/documentation-and-software>. Accessed May 15, 2025.



Fig. 4. Interface of the INIT-IDLE and Q/SS tabs. QKD—quantum key distribution

P_{value} is then calculated using the following equation:

$$P_{\text{value}} = \text{erfc}\left(\frac{s_{\text{obs}}}{\sqrt{2}}\right). \quad (8)$$

When the calculated P_{value} is smaller than the chosen significance level α , the RS fails to meet the randomness criteria.¹² Conversely, if P_{value} exceeds the significance level, the RS is deemed to be of good quality, indicating that the qubit is stable and well-suited for use in RNG tasks.

After outlining the description and fundamental operations of the *QIS* software and establishing the method for identifying stable and reliable qubits among the quantum states of the QCD register, we will proceed to generate RS on a cloud-based quantum computer. To achieve this, the following steps are carried out:

1. Navigate to the QRNG tab, select QCD ibm_brisbane from the drop-down list, and wait for the connection to be established with the cloud device.
2. Using the slider interface element, note the entire quantum register of QCD. The ibm_brisbane has 127 qubits, so select all 127 quantum states for the QRNG.
3. Set the significance level “Alpha” to 0.01.
4. The quantum scheme is set to repeat 20,000 times, representing the maximum available during the

research period. Consequently, each qubit will generate an RS measuring $2 \cdot 10^4$ bits in length. Considering a total of 127 qubits per QCD, with the same maximum repetition count, the final length of the potential key will reach $254 \cdot 10^4$ bits—equivalent to approximately 0.3 MB.

5. The optimization level of the quantum scheme is deemed irrelevant for the task under consideration, so its parameter remains fixed at 0.
6. With all configurations finalized, the QRNG process can now commence.

ANALYSIS AND INTERPRETATION OF RESULTS

Step 1. Statistics Collection

Once the quantum computer has finished processing, the results will be stored in the user’s home folder (Fig. 5).

Directories labeled “1” and “2” are designated to store RS files generated from each qubit, which are categorized into “good” and “bad” quantum states following the filtering process. Directories “3” and “4” hold files resulting from the concatenation of all RS files in directories “1” and “2,” respectively. The authors aim to derive the final random binary key using qubits identified as stable and appropriate for the QRNG process. Additionally, a final “bad” key is created to enable comparison and assessment of the generated

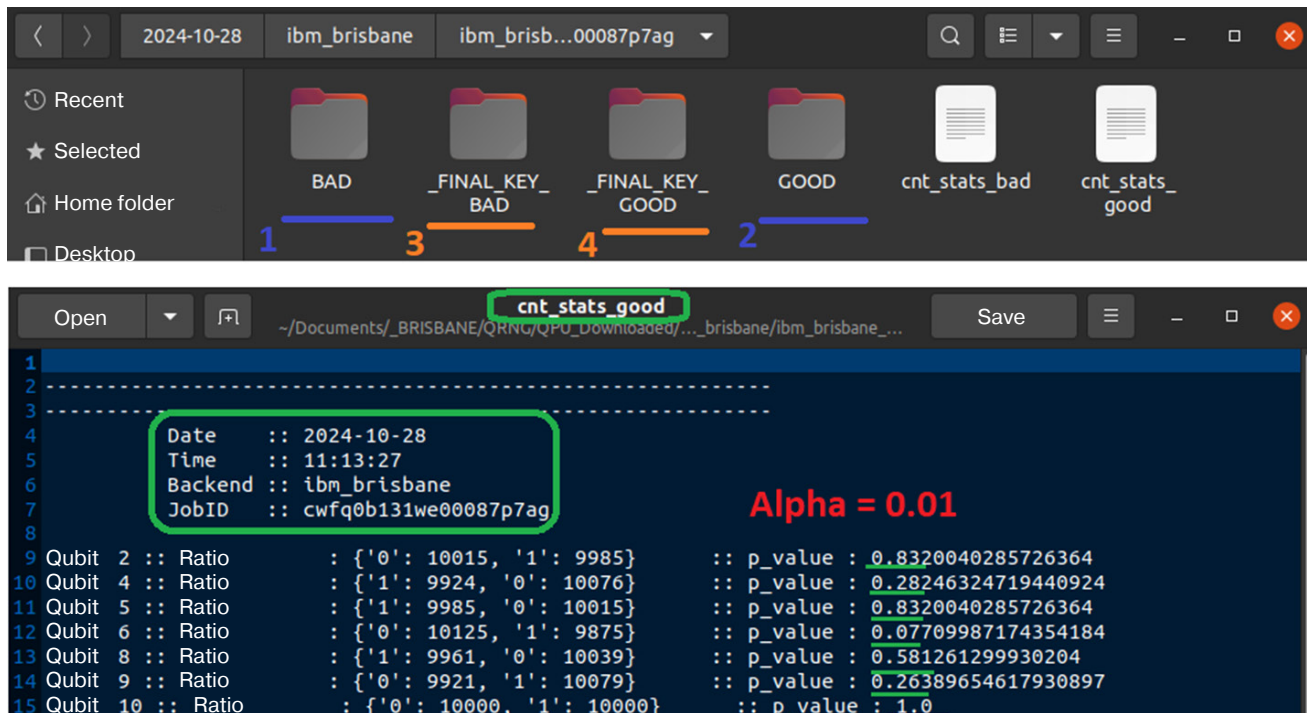


Fig. 5. Q/IS results (QRNG mode)

¹² The frequency test is clearly an essential step in assessing the randomness of the RS, though on its own, it remains insufficient to fully establish randomness.

keys along with their corresponding qubits against each other. Figure 5 below illustrates the contents of a file containing the final statistics of a single experiment, derived from the “good” qubits in the QCD.

Due to the probabilistic nature of the computing processes in quantum computers, it becomes evident that a single execution may not always suffice. To address this, the *QISs* software incorporates mechanisms for accumulating statistical data from prior completed projects. Figure 6 illustrates the information gathered from executing the QRNG on two IBM QCD systems.

This method is intended to enable users to visually monitor variations in both the quantity and quality of qubits within the quantum register of a cloud-based device. However, caution is necessary when interpreting the results. Minor fluctuations in the ratio of “0” and “1” may mistakenly classify a qubit as “unreliable” (false positive), even when the actual distribution of zeros and ones remains nearly unchanged. Therefore, it is essential to clarify the specific objective the user aims to achieve:

1. Generating a random binary key with optimal properties, leveraging the capabilities of the QCD. This involves strict criteria for the Alpha parameter and improving its value based on the quality of the resulting RS.
2. Evaluating the QCD quantum register to identify the most stable qubits in order to ensure the selection of an optimal set of quantum states for future experiments. Given Alpha’s requirements, it may be acceptable to reduce the stability parameters to a necessary level.

Based on the above, it can be concluded that careful consideration should be given to selecting an appropriate level of significance when filtering qubits by frequency. Additionally, it is essential to verify the final statistical results of individual experiments, as illustrated in Fig. 5, which records all the “0” and “1” values for each qubit. In certain scenarios, adjustments may be required to relax the Alpha parameter. However, caution is advised, as overly lenient modifications might compromise the reliability of the RS by amplifying minor deviations from individual qubits into significant discrepancies between “0” and “1” values in the concatenated final key derived from the processed files.

Step 2. Correlation analysis

Performing a preliminary correlation analysis on the obtained results is crucial before taking any further steps. Even with the use of “good” qubits, concatenating generated files containing binary sequences without such analysis is not recommended.

Essentially, submitting a QRNG implementation scheme to a quantum computer implies that the QCD

is not a standalone RNG. In our case, the quantum computer itself is divided into numerous smaller RS generators, which correspond to individual qubits in the cloud-based device.

A natural question then arises: can each qubit in this context be considered an independent source of random data? The answer has a direct impact on whether there are valid justifications for combining small RSs to form a single final key consisting of two or more qubits.

The RS correlation test aims to establish if chosen qubits act as independent sources of random numbers, both in terms of their own states and when compared to other states in the quantum register.

The correlation between two randomly generated binary sequences $\{x\}_{i=1}^n$, $\{z\}_{i=1}^n$ of length $n \in \mathbb{N}$ is described by the following function:

$$F_{\text{corr}(i)} = \frac{1}{n} \sum_{j=1}^n s_j, i = (0, n), \quad (9)$$

where $s_j = 1$ for $\overline{x \oplus z(i)} = 1$ and $s_j = -1$ for $\overline{x \oplus z(i)} = 0$.

In other words, s_j performs a bitwise addition modulo 2 operation on the two sequences. However, when it comes to binary string comparison, this analysis is done with a cyclic shift. The results of the comparison are visually represented using inversion, with “1” being displayed when the two sequences are identical.

Although the current version of the *QISs* application does not include an automatic preliminary analysis of received RS for correlation, the results of this analysis were presented at the RADIOINFOCOM conference held in November 2024 [19]. The results of the preliminary studies show that the correlation between “good” qubits does not exceed a value of 10^{-3} , while qubits producing less stable and reliable sequences have a certain offset in the correlation function, ranging up to $4 \cdot 10^{-2}$.

Step 3. Verification of statistical independence

The final step in verifying the generated RS is to test them for statistical independence. One possible tool for this verification is the NIST Statistical Test Suite (STS), which is included in the *QISs* software for automated and user-defined RS verification (Fig. 4, on the right, Area 5).

Here, the focus lies in analyzing the final RS produced when merging binary strings with “good” qubits. However, since examining a single sequence may not always be revealing, a comparison of the final sequences generated over a defined timeframe enables us to observe the characteristics of the quantum device over time (such as the length of the final key, the number of successful and unsuccessful tests, etc.).

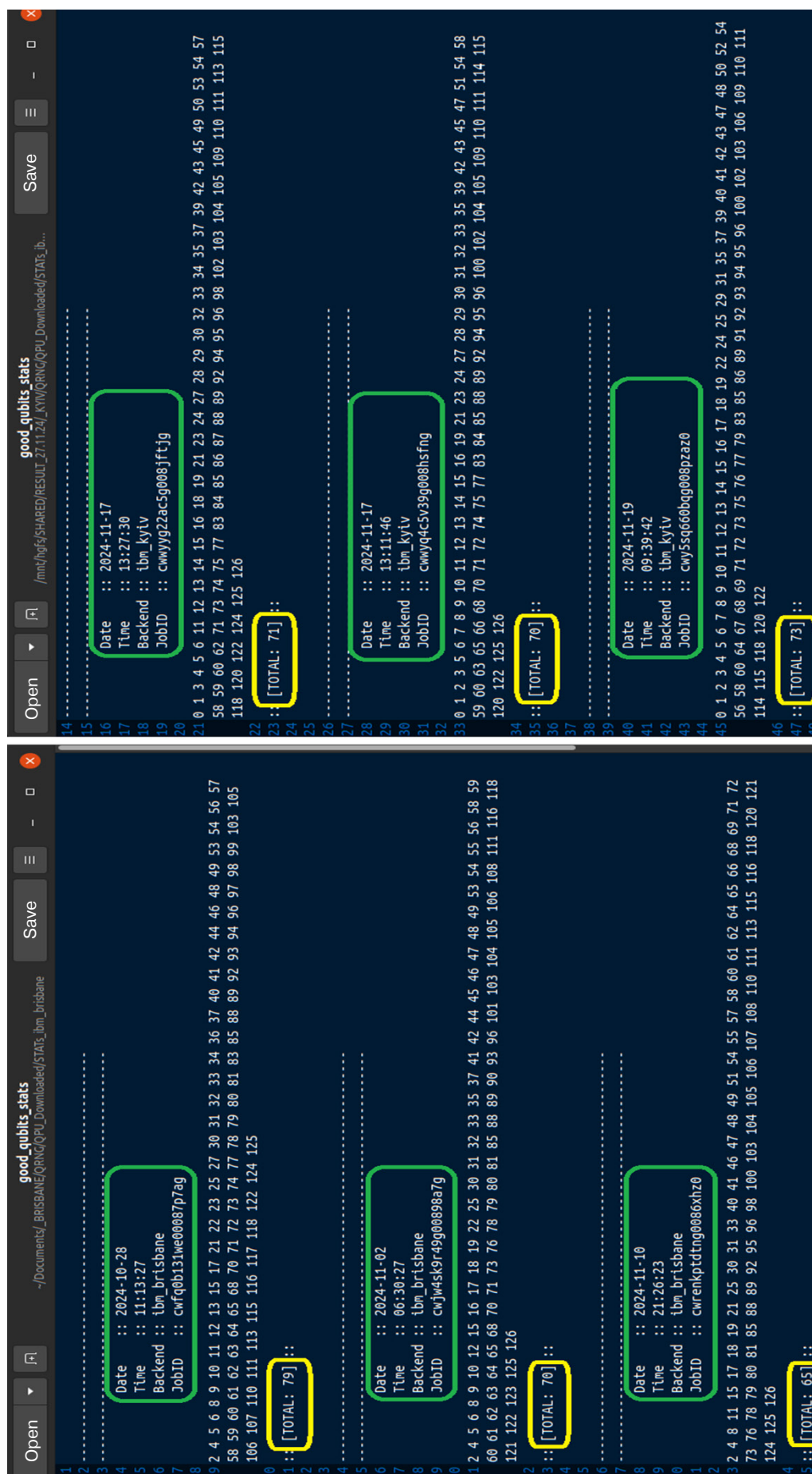


Fig. 6. Statistics on qubits of three-cloud QCD in QRNG tasks

The data presented in Fig. 7 is derived from the analysis of RS measurements conducted on the *ibm_brisbane* quantum computer, with a one-month time interval between data points.

Although the number of stable qubits varies slightly over the course of the experiments, it consistently starts and ends each month at 79. Notably, the results of the tests are largely consistent, except for the test measuring the length of consecutive “1” strings, which demonstrates success on the second attempt.

Similar studies are conducted on the *ibm_kyvi* QCD, with a two-week interval between each (Fig. 8a) and on the *ibm_sherbrooke* QCD, with a seven-day gap (Fig. 8b).

The nature of passing the tests for two QCD shown in Fig. 8 deserves special attention. When separately RS is applied to each “good” qubit in the selected QCD, with a length of $2 \cdot 10^4$ bits, the bit test passes successfully. However, when these strings are concatenated into final RSs (slightly larger than 0.1 MB), identical tests related to the search for “0” and “1” fail. The reason for this phenomenon is the “run-in” difference between the

levels of zeros and ones in the final binary string due to the low requirements for the significance level of the primary filtering of qubits ($\alpha = 0.01$). Meanwhile, the *ibm_brisbane* QCD, which has similar significance levels, shows better results.

The experimental results are presented in Table 2.

The minimum and maximum ratios of stable qubits observed over the research period, with a significance level of 0.01 applied to each QCD, are as follows: *ibm_brisbane* at 65/81, *ibm_kyvi* at 59/73, and *ibm_sherbrooke* at 52/62, respectively.

While these findings are not exhaustive and do not offer full comprehension of the computational capabilities of the investigated quantum devices, the results serve as a specific case study to confirm the possibility of utilizing quantum computers as RNGs to unveil crucial aspects of a computing system. The described approach, which offers valuable starting points for initial QCD evaluations, is of value in assessing the strengths of a selected computer and in identifying the most desirable features of quantum states for effectively handling user-defined applications.

Table 2. Results of the QRNG process on cloud-based QCD

Date	QCD	Number of “good” qubits / RS length	Failed NIST STS tests ¹³	Circuit operation time, s	Best / Worst “0” to “1” ratio across entire QCD register
28.10.2024	<i>ibm_brisbane</i>	79 / $158 \cdot 10^4$	1 “Runs” test 8 “RandomExcursions” tests 18 “RandomExcursionsVariant” tests	7	10.000–10.000 / 12.684–7.316
26.11.2024	<i>ibm_brisbane</i>	79 / $158 \cdot 10^4$	8 “RandomExcursions” tests 18 “RandomExcursionsVariant” tests	8	9.999–10.001 / 7.498–12.502
05.11.2024	<i>ibm_kyvi</i>	59 / $118 \cdot 10^4$	1 “Frequency” test 2 “CumulativeSums” tests	13	9.998–10.002 / 13.029–6.971
24.11.2024	<i>ibm_kyvi</i>	64 / $128 \cdot 10^4$	1 “Frequency” test 2 “CumulativeSums” tests 8 “RandomExcursions” tests 18 “RandomExcursionsVariant” tests	14	9.999–10.001 / 15.143–4.857
06.11.2024	<i>ibm_sherbrooke</i>	52 / $104 \cdot 10^4$	1 “Frequency” test 2 “CumulativeSums” tests 8 “RandomExcursions” tests 18 “RandomExcursionsVariant” tests	9	9.997–10.003 / 13.404–6.596
12.11.2024	<i>ibm_sherbrooke</i>	52 / $104 \cdot 10^4$	1 “Frequency” test 2 “CumulativeSums” tests 8 “RandomExcursions” tests 18 “RandomExcursionsVariant” tests	8	9.984–10.016 / 12.596–7.404

¹³ Maurer’s universal test is not taken into account due to a lack of adequate information.

28.10.24 // 1.580.000 Bit	
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES	
generator is </home/and/Documents/QISs/NIST/ResultTest_2024-11-29_16:20:15/KEY.txt>	
C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 P-VALUE PROPORTION STATISTICAL TEST	
3 1 2 1 1 0 1 0 1 0 0.534146 8/10	Frequency
4 1 0 2 1 0 0 1 0 1 0.122325 9/10	BlockFrequency
4 1 0 2 1 1 0 0 1 0 0.122325 8/10	CumulativeSums
4 1 1 2 1 0 1 0 0 0 0.122325 8/10	CumulativeSums
9 0 0 1 0 0 0 0 0 0 0.000000 * 3/10	Runs
4 2 1 0 2 0 1 0 0 0 0.066882 10/10	LongestRun
1 2 0 1 1 1 1 0 2 1 0.911413 10/10	Rank
0 3 2 1 0 0 1 1 1 1 0.534146 10/10	FFT
3 1 1 2 1 0 1 0 1 0 0.534146 10/10	NonOverlappingTemplate
2 1 2 2 1 0 0 1 1 0 0.739918 10/10	NonOverlappingTemplate
1 1 1 2 0 0 1 2 0 2 0.739918 10/10	NonOverlappingTemplate
1 1 3 3 0 1 0 1 0 0 0.213309 10/10	NonOverlappingTemplate
2 1 2 1 1 0 0 1 1 1 0.911413 10/10	NonOverlappingTemplate
1 0 0 1 1 0 2 3 1 1 0.534146 10/10	NonOverlappingTemplate
1 0 1 1 1 2 1 0 1 2 0.911413 9/10	NonOverlappingTemplate
2 2 2 1 1 1 1 0 0 0 0.739918 10/10	NonOverlappingTemplate

26.11.24 // 1.580.000 Bit	
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES	
generator is </home/and/Documents/QISs/NIST/ResultTest_2024-11-29_16:21:10/KEY.txt>	
C1 C2 C3 C4 C5 C6 C7 C8 C9 C10 P-VALUE PROPORTION STATISTICAL TEST	
8 3 2 1 1 0 0 1 0 1 0.534146 8/10	Frequency
9 1 2 2 0 1 0 2 2 0 0.534146 10/10	BlockFrequency
10 3 3 0 1 0 1 0 2 0 0.122325 8/10	CumulativeSums
11 5 0 0 0 1 0 1 1 2 0 0.008879 8/10	CumulativeSums
12 4 2 0 2 0 1 0 0 1 0 0.066882 8/10	Runs
13 3 1 1 2 1 0 2 0 0 0 0.350485 9/10	LongestRun
14 1 1 1 2 0 2 1 0 2 0 0.739918 10/10	Rank
15 0 0 2 1 3 1 1 0 1 0 0.534146 10/10	FFT
16 0 1 1 1 3 2 2 0 0 0 0.350485 10/10	NonOverlappingTemplate
17 2 2 2 1 0 1 1 0 0 1 0.739918 10/10	NonOverlappingTemplate
18 1 2 1 2 0 2 1 0 0 1 0.739918 10/10	NonOverlappingTemplate
19 0 1 0 2 2 1 2 0 0 2 0.534146 10/10	NonOverlappingTemplate
20 0 0 2 0 0 1 3 1 2 1 0.350485 10/10	NonOverlappingTemplate
21 1 0 0 2 0 2 1 1 2 1 0.739918 10/10	NonOverlappingTemplate
22 2 0 1 1 1 2 3 0 0 0 0.350485 10/10	NonOverlappingTemplate
23 2 0 1 1 0 3 1 0 2 0 0.350485 10/10	NonOverlappingTemplate

Fig. 7. Results of verifying the final RS using the ibm_brisbane QCD

05.11.24 // 1,180,000 Bit														
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES														
generator is </home/amd/Documents/QISS/NIST/ResultTest_2024-11-29_16:24:08/KEY.txt>														
C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE	PROPORTION	STATISTICAL TEST		
9	0	1	0	0	0	0	0	0	0	0.000000	*	Frequency	8/10	
5	2	0	0	1	0	1	1	0	0	0.008879		BlockFrequency	10/10	
8	1	1	0	0	0	0	0	0	0	0.000000	*	CumulativeSums	8/10	
8	1	0	1	0	0	0	0	0	0	0.000000	*	CumulativeSums	7/10	
3	4	0	1	0	0	1	1	0	0	0.035174		Runs	10/10	
2	1	0	0	0	1	2	0	2	2	0.534146		LongestRun	9/10	
1	0	0	1	1	3	0	1	1	2	0.534146		Rank	10/10	
2	2	1	0	2	0	0	0	2	1	0.534146		FFT	9/10	
2	2	1	1	1	2	0	0	0	0	0.739918		NonOverlappingTemplate	9/10	
1	1	0	2	2	1	1	0	1	1	0.911413		NonOverlappingTemplate	10/10	
4	1	0	1	1	0	1	0	0	2	0.122325		NonOverlappingTemplate	10/10	
1	0	2	1	2	0	1	0	2	1	0.739918		NonOverlappingTemplate	10/10	

(a)

24.11.24 // 1,280,000 Bit														
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES														
generator is </home/amd/Documents/QISS/NIST/ResultTest_2024-11-29_16:24:58/KEY.txt>														
C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE	PROPORTION	STATISTICAL TEST		
8	7	1	2	0	0	0	0	0	0	0.000001	*	Frequency	4/10	
9	3	1	1	1	0	1	1	1	0	0.739918		BlockFrequency	9/10	
10	7	0	1	1	1	0	0	0	0	0.000003	*	CumulativeSums	4/10	
11	7	2	0	1	0	0	0	0	0	0.000001	*	CumulativeSums	4/10	
12	5	1	0	0	0	1	1	0	0	0.008879		Runs	8/10	
13	1	0	2	1	0	1	1	3	1	0.534146		LongestRun	10/10	
14	0	3	0	1	1	0	1	2	1	0.534146		Rank	10/10	
15	2	2	0	0	0	2	0	0	3	0.213309		FFT	10/10	
16	1	1	0	1	0	3	2	0	1	0.534146		NonOverlappingTemplate	10/10	
17	3	0	2	1	2	0	0	0	1	0.350485		NonOverlappingTemplate	10/10	
18	4	1	2	0	1	1	0	1	0	0.122325		NonOverlappingTemplate	9/10	
19	0	1	0	1	3	1	0	2	0	0.350485		NonOverlappingTemplate	10/10	

(a)

12.11.24 // 1,040,000 Bit															
RESULTS FOR THE UNIFORMITY OF P-VALUES AND THE PROPORTION OF PASSING SEQUENCES															
generator is </home/amd/Documents/QISS/NIST/ResultTest_2024-11-29_16:23:19/KEY.txt>															
C1	C2	C3	C4	C5	C6	C7	C8	C9	C10	P-VALUE	PROPORTION	STATISTICAL TEST			
8	7	1	1	0	0	1	0	0	0	0.000003	*	Frequency	5/10		
9	2	1	1	0	1	0	2	0	2	1	0.739918		BlockFrequency	10/10	
10	7	0	0	1	1	0	0	1	0	0.000003	*	CumulativeSums	5/10		
11	9	0	0	1	0	0	0	0	0	0.000000	*	CumulativeSums	6/10		
12	1	1	0	0	0	1	3	1	2	1	0.534146		Runs	9/10	
13	1	1	2	1	0	1	0	2	2	0.739918		LongestRun	10/10		
14	2	0	1	0	4	1	2	0	0	0.066882		Rank	10/10		
15	2	1	2	0	0	1	0	3	0	1	0.350485		FFT	10/10	
16	2	5	1	0	1	0	0	1	0	0.008879		NonOverlappingTemplate	10/10		
17	1	2	1	1	0	1	1	1	1	0.991468		NonOverlappingTemplate	10/10		
18	1	1	2	1	2	0	1	0	0	0.739918		NonOverlappingTemplate	10/10		
19	0	1	3	0	2	1	0	2	1	0	0.350485		NonOverlappingTemplate	10/10	

RESULTS AND DISCUSSION

Based on the findings of the study, the following observations, which may serve as a basis for future research, can be made.

1. Given the QCD technical specifications that influence the quantum state and consequent RS generation outcome, there is a need to determine a formal mathematical approach to the degree of correlation between the current calibration parameters of the quantum processor and the expected outcomes of the QRNG process. For instance, the characteristics of the best and the worst qubits within the entire quantum register for each experiment detailed in Table 2 are presented in Table 3.

Although some results are not immediately apparent, it is proposed that a portion of the qubit may produce suboptimal outcomes, including when neighboring quantum states are engaged, which has a secondary effect on the targeted qubit. Verification of this hypothesis, as well as the development of a mathematical model for the correlation between the tactical and technical properties of the QCD and the outcomes of the QRNG system, are potential areas for future investigation.

2. Using the entire quantum processor, the maximum RS length can be $254 \cdot 104$ bits, which results in a sample size of approximately 0.3 MB with a bit

depth of 127 qubits and $\text{max_shots} = 20.000$ (the number of times the quantum circuit is run). In this scenario, given the processing time of the quantum circuit, the *ibm_brisbane* (7 seconds) is the fastest QCD, which ideally provides an RNG speed of approximately 44 kbit/s. Obviously, this speed places limitations on the application of QCD as an RNG. Furthermore, experiments have shown that approximately half of the qubits may not be suitable for obtaining a stable RS. However, by scaling up the bit depth of quantum processors and addressing existing equipment errors and interferences, the current limitations can be overcome, opening up new opportunities for using QCD in information security applications.

3. Introduced in 2010, NIST STS tests represent an older approach to verifying the RS quality. The present authors believe it is crucial to further investigate RS generated by contemporary methods, as these can not only be used to assess the statistical independence of the data but also to delve into the inherent characteristics and nature of the RNG itself, ultimately determining how well it mimics the properties of a truly physical QRNG.

4. The proposed method could be incorporated into the expanded suite of tests and programs used to evaluate the technical capabilities of a given QCD. Using this approach, preliminary insights into the execution speed of quantum circuits on the chosen QCD

Table 3. Characteristics of IBM's "extreme" cloud QCD qubits

Date	QCD/qubit	«0» to «1» Ratio	$T_1, \mu\text{s}$	$T_2, \mu\text{s}$	Readout, err	(sx), err
28.10.2024	brisbane/10	10.000–10.000	290.18	276.91	0.018	$1 \cdot 10^{-4}$
	brisbane/86	12.684–7.316	90.85	115.17	0.113	$1.45 \cdot 10^{-2}$
26.11.2024	brisbane/55	9.999–10.001	243.83	120.25	0.01	$2 \cdot 10^{-4}$
	brisbane/24	7.498–12.502	226.79	98.47	0.14	$2 \cdot 10^{-4}$
05.11.2024	kyvi/5	9.998–10.002	347.03	331.96	0.008	$2 \cdot 10^{-4}$
	kyvi/90	13.029–6.971	43.91	22.84	0.114	$3.3 \cdot 10^{-3}$
24.11.2024	kyvi/122	9.999–10.001	211.81	173.1	0.002	$2 \cdot 10^{-4}$
	kyvi/65	15.143–4.857	201.52	126.32	0.092	$2 \cdot 10^{-3}$
06.11.2024	sherbr-ke/23	9.997–10.003	220.73	24.85	0.009	$4 \cdot 10^{-4}$
	sherbr-ke/56	13.404–6.596	173.46	16.9	0.033	$3 \cdot 10^{-4}$
12.11.2024	sherbr-ke/63	9.984–10.016	217.35	133.31	0.035	$2 \cdot 10^{-4}$
	sherbr-ke/9	12.596–7.404	469.69	70.2	0.035	$2 \cdot 10^{-4}$

can be rapidly obtained. Additionally, the QRNG output offers valuable information regarding the most efficient qubit configuration, which users can then leverage when designing more intricate and resource-intensive quantum circuits.

5. Many researchers are actively exploring ways to address the decoherence challenge faced by quantum states [20–22]. The prevailing strategies often involve implementing error correction techniques and utilizing a conceptual entity known as a logical qubit. A logical qubit can be conceptualized as a group of physical qubits that, when considered together, effectively represent the behavior of a single quantum state. Nevertheless, employing correction codes or shifting to logical qubits within the framework of the QRNG task proves unsuitable since the rationale behind generating an RS on a quantum computer hinges solely on utilizing qubits functioning as independent physical RNGs.

The authors plan to incorporate a QRNG task within cloud-based QCD as part of a suite of tests designed to evaluate the computational capabilities of quantum computers for cybersecurity experts. This software package will encompass three core tests: QRNG, a simulation of the QKD protocol, and the execution of a quantum Fourier transform. The developed algorithm should be characterized by its speed, simplicity, accessibility, reliance on QCD principles, reproducibility, scalability, and unambiguous interpretability.

Both the latest version of the *QISs* program code and the associated research findings are accessible within the project repository on GitHub.¹⁴

CONCLUSIONS

Within this research, the authors have developed an application enabling the design and remote execution of quantum circuits leveraging RNG on the cloud-based QCD platform. In order to assess any potential unintended connections between quantum states functioning as independent QRNGs, the application also provides tools for analyzing correlations within the generated RSs.

Utilizing a frequency bit test and the user-defined significance level, the application delivers insights into the most and least reliable qubits, assessing their stability and suitability for quantum Walsh–Hadamard state applications.

To assess the randomness of the binary patterns (zeros and ones) produced in the generated sequences, the program offers the functionality for the RS evaluation using a suite of statistical tests from the NIST STS.

The approach proposed in the present paper can serve as a useful tool for researchers in solving several problems, such as identifying the optimal set of qubits in a quantum processor, generating binary RSs on quantum computers, and assessing some technical capabilities of studied quantum devices.

ACKNOWLEDGMENTS

The authors thank K.E. Komogorov, a student at RTU MIREA, for his prompt assistance in preparing the bash script, initial testing of the *QISs* application software capabilities, and providing his own GitHub repository for hosting the developed program in the public domain.

Authors' contribution

All authors contributed equally to the research work.

¹⁴ GitHub / *QISs*. <https://github.com/cyberavenman/QISs>. Accessed May 15, 2025.

REFERENCES

1. Proctor T., Young K., Baczewski A.D., Blume-Kohout R. Benchmarking quantum computers. *arXiv*. 2024. arXiv:2407.08828. <https://doi.org/10.48550/arXiv.2407.08828>
2. Amico M., Zhang H., Jurcevic P., et al. Defining Standard Strategies for Quantum Benchmarks. *IBM Publications*. 2023. Available from URL: <https://research.ibm.com/publications/defining-standard-strategies-for-quantum-benchmarks>. Accessed May 15, 2025.
3. Acuaviva A., Aguirre D., Pena R., Sanz M. Benchmarking Quantum Computers: Towards a Standard Performance Evaluation Approach. *arXiv*. 2024. arXiv:2407.10941. <https://doi.org/10.48550/arXiv.2407.10941>
4. Eisert J., Hangleiter D., Walk N., et al. Quantum certification and benchmarking. *Nat. Rev. Phys.* 2020;2:382–390. <https://doi.org/10.1038/s42254-020-0186-4>
5. Kryuchkov A.A. On the need to adopt a single standard for evaluating the performance and certification of quantum computers. *Informatsionno-ehkonomicheskie aspekty standartizatsii i tekhnicheskogo regulirovaniya = Information and Economic Aspects of Standardization and Technical Regulation*. 2024;6(81):43–49 (in Russ.).
6. Wack A., Paik H., Javadi-Abhari A., Jurcevic P., Faro I., Gambetta J.M., Johnson B.R. Scale, Quality, and Speed: three key attributes to measure the performance of near-term quantum computers. *arXiv*. 2021. arXiv:2110.14108. <https://doi.org/10.48550/arXiv.2110.14108>
7. McKay D.C., Hincks I., Pritchett E.J., Carroll M., Govia L.C.G., Merkel S.T. Benchmarking Quantum Processor Performance at Scale. *arXiv*. 2023. arXiv:2311.05933. <https://doi.org/10.48550/arXiv.2311.05933>
8. Amico M., Zhang H., Jurcevic P., Bishop L.S., Nation P., Wack A., McKay D.C. Defining Standard Strategies for Quantum Benchmarks. *arXiv*. 2023. arXiv:2303.02108 <https://doi.org/10.48550/arXiv.2303.02108>
9. Shor P.W. Algorithms for quantum computation: discrete logarithms and factoring. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. IEEE; 1994. P. 124–134. <https://doi.org/10.1109/SFCS.1994.365700>
10. Balygin K.A., Kulik S.P., Molotkov S.N. Implementation of a Quantum Generator of Random Numbers: Extraction of Provably Random Bit Sequences from Correlated Markov Chains. *Jetp. Lett.* 2024;119(7):538–548. <https://doi.org/10.1134/S0021364024600575>
[Original Russian Text: Balygin K.A., Kulik S.P., Molotkov S.N. Implementation of a Quantum Generator of Random Numbers: Extraction of Provably Random Bit Sequences from Correlated Markov Chains. *Pis'ma v Zhurnal eksperimental'noi i teoreticheskoi fiziki (Pis'ma v ZHEHTF)*. 2024;119(7):533–544 (in Russ.). <https://doi.org/10.31857/S1234567824070115>]
11. Gaidash A.A., Goncharov R.K., Kozubov A.V., Yakovlev P.V. Mathematical model of random number generator based on vacuum fluctuations. *Vestnik Sankt-Peterburgskogo universiteta. Prikladnaya matematika. Informatika. Protsessy upravleniya = Vestnik of Saint Petersburg University. Applied Mathematics. Computer Science. Control Processes*. 2024;20(2):136–153 (in Russ.). <https://doi.org/10.21638/spbu.10.2024.202>
12. Petrenko A.A., Kovalev A.V., Bougrov V.E. Random number generation with arrays of coupled quantum-dot micropillar lasers. *Nauchno-tekhnicheskii vestnik informatsionnykh tekhnologii, mekhaniki i optiki = Scientific and Technical Journal of Information Technologies, Mechanics and Optics*. 2021;21(6):962–968 (in Russ.). <https://doi.org/10.17586/2226-1494-2021-21-6-962-968>
13. Orlov M.A., Nechaev K.A., Reznichenko S.A. Evaluation of statistical properties and cryptographic strength of random sequences obtained by an IBM quantum computer. *Bezopasnost' informatsionnykh tekhnologii = IT Security (Russia)*. 2023;30(1):14–26 (in Russ.). <http://doi.org/10.26583/bit.2023.1.01>
14. Li Y., Fei Y., Wang W., et al. Quantum random number generator using a cloud superconducting quantum computer based on source-independent protocol. *Sci Rep*. 2021;11:23873. <https://doi.org/10.1038/s41598-021-03286-9>
15. Salehi R., Razaghi M., Fotouhi B. Hybrid Hadamard and Controlled-Hadamard Based Quantum Random Number Generators in IBM QX. *Physica Scripta*. 2022;97(6):065101. <https://doi.org/10.1088/1402-4896/ac698b>
16. Yadav A., Mishra S., Pathak A. Partial loopholes free device-independent quantum random number generator using IBM's quantum computers. *Physica Scripta*. 2024;99(11):115103. <https://doi.org/10.1088/1402-4896/ad7c02>
17. Feynman R.P. Quantum Mechanical Computers. *Optics News*. 1985;11(2):11–20. Available from URL: https://www.optica-opn.org/home/articles/on/volume_11/issue_2/features/quantum_mechanical_computers/. Accessed May 15, 2025.
18. Kryuchkov A.A. *QISs_v.0.3.9: Computer Program RU2025613655 RF*. Publ. 13.02.2025 (in Russ.).
19. Kryuchkov A.A., Komogorov K.E. Simulation of the random number generation process on quantum computing devices. In: *Proceedings of the 8th Scientific and Practical Conference "Actual Problems and Prospects of Radio Engineering and Infocommunication Systems."* Moscow: RTU MIREA; 2024. P. 501–506 (in Russ.).
20. Acharya R., Abanin D.A., Aghababaie-Beni L., et al. Quantum error correction below the surface code threshold. *Nature*. 2025;638:920–926. <https://doi.org/10.1038/s41586-024-08449-y>

21. Verma S., Kumari S.S., Kumar R.S. Topological quantum error correction with semions. *Int. J. Phys. Math.* 2024;6(2):44–47. <https://doi.org/10.33545/26648636.2024.v6.i2a.95>
22. Webster M., Browne D. Engineering Quantum Error Correction Codes Using Evolutionary Algorithms. *IEEE Trans. Quantum Eng.* 2025;6:3100514. <https://doi.org/10.1109/TQE.2025.3538934>

СПИСОК ЛИТЕРАТУРЫ

1. Proctor T., Young K., Baczewski A.D., Blume-Kohout R. Benchmarking quantum computers: *arXiv*. 2024. arXiv:2407.08828. <https://doi.org/10.48550/arXiv.2407.08828>
2. Amico M., Zhang H., Jurcevic P., et al. Defining Standard Strategies for Quantum Benchmarks. *IBM Publications*. 2023. URL: <https://research.ibm.com/publications/defining-standard-strategies-for-quantum-benchmarks>. Дата обращения 15.05.2025.
3. Acuaviva A., Aguirre D., Pena R., Sanz M. Benchmarking Quantum Computers: Towards a Standard Performance Evaluation Approach: *arXiv*. 2024. arXiv:2407.10941. <https://doi.org/10.48550/arXiv.2407.10941>
4. Eisert J., Hangleiter D., Walk N., et al. Quantum certification and benchmarking. *Nat. Rev. Phys.* 2020;2:382–390. <https://doi.org/10.1038/s42254-020-0186-4>
5. Крючков А.А. О необходимости принятия единого стандарта по оценке производительности и сертификации квантовых вычислительных устройств. *Информационно-экономические аспекты стандартизации и технического регулирования. Сборник научных трудов участников I Научно-практической конференции «Стандартизация: траектория науки», посвященной 100-летию деятельности ФГБУ «Институт стандартизации»*. 2024;6(81):43–49.
6. Wack A., Paik H., Javadi-Abhari A., Jurcevic P., Faro I., Gambetta J.M., Johnson B.R. Scale, Quality, and Speed: three key attributes to measure the performance of near-term quantum computers. *arXiv*. 2021. arXiv:2110.14108. <https://doi.org/10.48550/arXiv.2110.14108>
7. McKay D.C., Hincks I., Pritchett E.J., Carroll M., Govia L.C.G., Merkel S.T. Benchmarking Quantum Processor Performance at Scale. *arXiv*. 2023. arXiv:2311.05933. <https://doi.org/10.48550/arXiv.2311.05933>
8. Amico M., Zhang H., Jurcevic P., Bishop L.S., Nation P., Wack A., McKay D.C. Defining Standard Strategies for Quantum Benchmarks. *arXiv*. 2023. arXiv:2303.02108 <https://doi.org/10.48550/arXiv.2303.02108>
9. Shor P.W. Algorithms for quantum computation: discrete logarithms and factoring. In: *Proceedings 35th Annual Symposium on Foundations of Computer Science*. IEEE; 1994. P. 124–134. <https://doi.org/10.1109/SFCS.1994.365700>
10. Балыгин К.А., Кулик С.П., Молотков С.Н. Реализация квантового генератора случайных чисел: экстракция доказуемо случайных битовых последовательностей из коррелированных марковских цепочек. *Письма в ЖЭТФ*. 2024;119(7):533–544. <https://doi.org/10.31857/S1234567824070115>
11. Гайдаш А.А., Гончаров Р.К., Козубов А.В., Яковлев П.В. Математическая модель квантового генератора случайных чисел на основе флуктуации вакуума. *Вестник Санкт-Петербургского университета. Прикладная математика. Информатика. Процессы управления*. 2024;20(2):136–153. <https://doi.org/10.21638/spbu10.2024.202>
12. Петренко А.А., Ковалев А.В., Бугров В.Е. Генерация случайных чисел с использованием массива связанных лазеров на основе микростолбиков с квантовыми точками. *Научно-технический вестник информационных технологий, механики и оптики*. 2021;21(6):962–968. <https://doi.org/10.17586/2226-1494-2021-21-6-962-968>
13. Орлов М.А., Нечаев К.А., Резниченко С.А. Оценка статистических свойств и криптографической стойкости случайных последовательностей, полученных квантовым компьютером IBM. *Безопасность информационных технологий*. 2023;30(1):14–26. <http://doi.org/10.26583/bit.2023.1.01>
14. Li Y., Fei Y., Wang W., et al. Quantum random number generator using a cloud superconducting quantum computer based on source-independent protocol. *Sci Rep*. 2021;11:23873. <https://doi.org/10.1038/s41598-021-03286-9>
15. Salehi R., Razaghi M., Fotouhi B. Hybrid Hadamard and Controlled-Hadamard Based Quantum Random Number Generators in IBM QX. *Physica Scripta*. 2022;97(6):065101. <https://doi.org/10.1088/1402-4896/ac698b>
16. Yadav A., Mishra S., Pathak A. Partial loopholes free device-independent quantum random number generator using IBM's quantum computers. *Physica Scripta*. 2024;99(11):115103. <https://doi.org/10.1088/1402-4896/ad7c02>
17. Feynman R.P. Quantum Mechanical Computers. *Optics News*. 1985;11(2):11–20. URL: https://www.optica-opn.org/home/articles/on/volume_11/issue_2/features/quantum_mechanical_computers/. Дата обращения 15.05.2025. Accessed May 15, 2025.
18. Крючков А.А. *QISs_v.0.3.9: свидетельство о государственной регистрации Программы для ЭВМ RU 2025613655 РФ*. Заявка № 2025611456; заявл. 28.01.2025; опубл. 13.02.2025. Бюл. № 2.
19. Крючков А.А., Комогоров К.Е. Моделирование процесса генерации случайных чисел на квантовых вычислительных устройствах. В сб.: *Материалы VIII НПК «Актуальные проблемы и перспективы радиотехнических и инфокоммуникационных систем»* (18–22 ноября 2024 г., Москва). М.: РТУ МИРЭА; 2024. С. 501–506.

20. Acharya R., Abanin D.A., Aghababaie-Beni L., et al. Quantum error correction below the surface code threshold. *Nature*. 2025;638:920–926. <https://doi.org/10.1038/s41586-024-08449-y>
21. Verma S., Kumari S.S., Kumar R.S. Topological quantum error correction with semions. *Int. J. Phys. Math.* 2024;6(2):44–47. <https://doi.org/10.33545/26648636.2024.v6.i2a.95>
22. Webster M., Browne D. Engineering Quantum Error Correction Codes Using Evolutionary Algorithms. *IEEE Trans. Quantum Eng.* 2025;6:3100514. <https://doi.org/10.1109/TQE.2025.3538934>

About the Authors

Andrey V. Korolkov, Cand. Sci. (Eng.), Corresponding Member of the Academy of Cryptography of the Russian Federation, Corresponding Member of the A.M. Prokhorov Academy of Engineering Sciences of the Russian Federation, Head of the Department of Information Security, Institute of Artificial Intelligence, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow, 119454 Russia). E-mail: korolkov@mirea.ru. RSCI SPIN-code 3849-6868, <https://orcid.org/0009-0003-4862-4816>

Andrey A. Kryuchkov, Senior Lecturer, Department of Information Security, Institute of Artificial Intelligence, MIREA – Russian Technological University (78, Vernadskogo pr., Moscow, 119454 Russia). E-mail: kryuchkov_a@mirea.ru. RSCI SPIN-code 7117-7238, <https://orcid.org/0009-0002-4750-6204>

Об авторах

Корольков Андрей Вячеславович, к.т.н., член-корреспондент Академии криптографии Российской Федерации, член-корреспондент Академии Инженерных наук им. А.М. Прохорова Российской Федерации, заведующий кафедрой информационной безопасности, Институт искусственного интеллекта, ФГБОУ ВО «МИРЭА – Российский технологический университет», (119454, Россия, Москва, пр-т Вернадского, д. 78). E-mail: korolkov@mirea.ru. SPIN-код РИНЦ 3849-6868, <https://orcid.org/0009-0003-4862-4816>

Крючков Андрей Андреевич, старший преподаватель, кафедра информационной безопасности, Институт искусственного интеллекта, ФГБОУ ВО «МИРЭА – Российский технологический университет» (119454, Россия, Москва, пр-т Вернадского, д. 78). E-mail: kryuchkov_a@mirea.ru. SPIN-код РИНЦ 7117-7238, <https://orcid.org/0009-0002-4750-6204>

Translated from Russian into English by K. Nazarov

Edited for English language and spelling by Thomas A. Beavitt