Random number generation using quantum computers

Philip Ilono[1]

[1] *Independent Scholar*

(Dated: December 11, 2024)

In this project, we experimentally generate random numbers using IBM's quantum computers and analyse their statistical properties. We begin by implementing a basic quantum circuit, which creates a superposition state followed by measurement, and evaluate the quality of the random numbers produced by this approach. Next, we characterize the noise present in the system and apply algorithmic post-processing to the measurement outcomes to enhance the quality (at-least theoretically) of the generated random numbers. After this post-processing, we generate a total of 1,608,305 random bits, which are then statistically tested, with results compared to those obtained from the initial protocol. To provide a consistent metric for comparison, we employ NIST min-entropy estimators and observe an increase in the min-entropy rate from 0.80903 and 0.824166 to 0.86219. This document serves as a technical guide to our non-novel implementation and statistical analysis methods.

I. INTRODUCTION

Random numbers are essential in various fields and applications, with cryptography being one of the most well-known. Many encryption algorithms, such as RSA and AES, utilise some variation of random numbers (e.g., prime numbers in the case of RSA) for secure key generation [1]. Secure key generation protects data and allows secure communication between a sender and the intended recipients. Another common application for which random numbers are vital is simulations. Modelling real-world scenarios often requires suitable models for randomness, as unpredictability is common in such systems. Monte Carlo simulations utilise random sampling to estimate the probabilities of various outcomes within unpredictable and complex processes. Monte Carlo simulations are commonplace in finance for option pricing, risk assessment, and more. They are also used in physics and chemistry to simulate particles and physical systems.

Such applications require random number generators (RNGs) that produce a sample that is hard to predict. Classical RNGs, such as pseudorandom number generators (PRNGs), typically rely on deterministic algorithms, with exceptions like true random number generators (TRNGs), which rely on extracting randomness from a physical environment. Random numbers generated from deterministic algorithms can be predictable if an attacker has expansive computational resources, and the random numbers generated from a TRNG are bounded by the quality of the physical source used. Quantum random number generators (QRNGs) harness the fundamental principles of quantum mechanics (particularly superposition in this case) to achieve true randomness.

Superposition is when a system can be in multiple states simultaneously until the system is measured, which then causes the system to collapse to one of the possible states [2]. In the case of randomness generation, a uniform superposition would be ideal, meaning that each of the possible states has an equal probability of being measured.

This requires the use of quantum devices to execute quantum processes using qubits. Qubits are fundamental units of quantum information and can be thought of as analogous to bits in classical computation [2]. The main separation from classical bits is that qubits can exhibit quantum processes such as superposition, entanglement, and interference.

However, even though quantum hardware has made significant progress in the last two decades, current quantum devices are still susceptible to issues that cause noise. Environmental sensitivity is one issue that can lead to noise, as qubits are vulnerable to electromagnetic field disturbances and changes in temperature. Quantum gates are used to act as a method of manipulating qubits to exhibit quantum processes. Thus, imperfections in quantum gates lead to errors in such computations. Crosstalk is another issue and arises when qubits are influenced by neighbouring qubits, typically in densely packed systems, causing errors.[2]

The presence of such noise can lead to bit flip errors or decoherence. These errors can alter the probabilities associated with the state and measurement being implemented. The manipulation of such states leads to a disruption in the unpredictability of the outcome, as the uniform distribution of possible outcome states is altered. Furthermore, due to the no-hiding theorem, information about the noise can be acquired as the information cannot be destroyed. This leads to unwanted consequences, especially in the field of cryptography and security, where malicious actors may be at work [3]. The result of an unpredictable sample is still the idealised goal, so finding methods to handle the effect of noise on the underlying unpredictability of the output has been the focus of much research.

Although the presence of noise poses a problem, there are still opportunities to enhance randomness generation. An example that has been explored is to utilise two independent sources of weak randomness, such as two quantum devices, to each, generate randomness that is bounded by the presence of noise in each device.

The two sources can then be combined to produce a single output that is more unpredictable than the two initial sources. This is implemented using two-source extractors, mathematical constructs that combine weakly random sources to extract high-quality, nearly uniform randomness.

Due to the no-hiding theorem, a good two-source extractor should be information-theoretically secure to protect against malicious attackers. Information-theoretic security ensures that the output remains unpredictable even if one of the initial sources is partially compromised, or even against an adversary with unlimited computational resources [4].

Information-theoretic security requires that the sources are independent, each source has a sufficient amount of unpredictability, and a strong two-source extractor is implemented. A strong two-source extractor ensures that the unpredictability of the output post-extraction is indistinguishable from a truly random sequence of bits.

The process of evaluating unpredictability will be introduced in the background, and the quality of the extractor utilised will be reviewed in the results. Thus, this particular project aims to observe a higher min-entropy than the respective observed min-entropy values for each of the two sources.

II. BACKGROUND

The scope of this project is generating high-quality randomness. High-quality randomness can be characterised by unpredictability and a uniform distribution of outcomes. Unpredictability is a property that is quantified by min-entropy and thus, a sufficient measure of randomness. Min-entropy can be imagined as how much information is expected to be obtained from an observation of the random variable X. The formula quantifies the unpredictability of the random variable X, highlighting the worst-case situation where the observed result is as predictable as possible. To summarise the formula, min-entropy is defined as a conservative measure of the unpredictability of a random variable [5].

The min-entropy, $H_\infty(X)$, quantifies the lower bound of the unpredictability of any outcome from a random variable X. It is defined mathematically as:

$$H_\infty(X) = -\log\left(\max_x P(X=x)\right), \tag{1}$$

where $P(X=x)$ is the probability of observing the outcome x.

The higher the min-entropy, the more unpredictable the outcome.

A. Extractors

Randomness extractors are effective post-processing algorithms to transform weak sources of randomness into high-quality randomness.

A deterministic extractor generates near-perfect randomness by transforming a random variable X with specific properties into a new variable $Ext_d(X)$. The new variable approximates a uniform distribution. The effectiveness of the deterministic extractor relies on the properties of the random variable X, such as having a sufficient level of min-entropy [5].

A two-source extractor combines two weak sources of randomness to produce a near-perfect random output. The output is nearly indistinguishable from a uniform random distribution. The requirements of the two-source extractor is that the two sources must have sufficient min-entropy and be independent [5].

These extractors can be implemented in programming from scratch but are also available in open-source libraries, such as Cryptomite [6]. Cryptomite is a Python library designed to implement randomness extractors to transform weak randomness into near-perfect randomness. The library includes the extractors stated in the related works and has specific documentation to aid in choosing suitable extractors for whatever goal.

B. Related Works

Berta et.al. [7] utilises two weak sources of randomness to generate information-theoretically secure random bits. The approach utilises noisy quantum computers as the weak sources of randomness and the use of the extractors executes in quasi-liner time, $O(n\ log\ n)$. Dodis [8] implements a similar approach but introduces a new construction of a two-source extractor that can extract more bits than previously possible.

Foreman et. al [5] displays how post-processing algorithms are effective in increasing the quality of RNG outputs. The increased quality is evaluated using various statistical tests. The paper concludes that advanced extractors are able to improve the performance of some RNGs that fail initial tests.

III. METHODOLOGY

In this work, we construct three protocols with varying assumptions and compare them with one another, based on estimating the amount of entropy in the final outputs. We begin by reviewing established techniques for entropy assessment, in particular, focusing on methods that can be applied universally based on statistical testing. Following this, we outline our specific approach to entropy testing and analysis. Post-processing extractors are introduced in protocols 2 and 3 and act as an improvement upon Protocol 1.

A. Entropy Assessment

Entropy is a measure of unpredictability or randomness in a system. Min-entropy is a specific type of entropy that quantifies the most conservative measure of entropy in a system. In the context of quantum computers, entropy estimation involves quantifying the amount of randomness or unpredictability in the measurement outcomes of quantum processes. This is crucial for applications like quantum cryptography, where perfect entropy is needed for secure key generation [9]. Entropy estimation is important as it provides a method of evaluating the level of randomness and unpredictability of different bit sequences. This provides a way to assess the strength of a source of randomness and whether a two-source extractor has increased the randomness of a system post-extraction.

There are different methods to estimate the entropy of a system, each suited to specific scenarios and requirements. Selecting the most appropriate method depends on factors such as scalability, sensitivity to noise, and the ability to capture quantum correlations. In the context of this project, scalability to larger systems and compatibility with non-iid sources were key criteria for method selection.

One common approach involves employing specialized algorithms that can estimate entropy from the outcomes of quantum measurements. These algorithms are diverse and often designed for specific use cases. The National Institute of Standards and Technology (NIST) has an estimator tool that can implement specialised algorithms by taking raw bits from a noise source to then perform different tests and entropy estimates of the bits. These tests are dependent on whether the data are independent and identically distributed (iid) or not. The estimation process utilises modelling, black box estimators and statistical tests. For data to be iid, the data must satisfy each respective condition of being independent and identically distributed. Independence is satisfied when the random variables have no effect on each other. Identical distribution is satisfied if the probability of an observation is constant for each event.

Another method for entropy estimation employs quantum state tomography to reconstruct the density matrix ρ and then compute the Von Neumann entropy [10]. While this tomographic approach can account for quantum correlations such as entanglement, it too suffers from poor scalability and sensitivity to noise in larger systems.

Alternatively, another method involves directly measuring the quantum system multiple times to empirically estimate probabilities and calculate the entropy. This method is simple to implement but does not scale well to larger systems.

In this project, the NIST entropy estimators were chosen because they offer scalability to larger input sizes and are well-suited for handling non-iid data, which aligns with the characteristics of our quantum measurement sources. Although these algorithms have limitations, such as not fully capturing quantum correlations, they meet the project's primary need for a reliable and scalable solution for entropy estimation.

While entropy estimation is straightforward for iid sources, most practical sources do not meet this criterion. The process in this scenario applies a vast set of black-box estimators on the data with differing constraints about the distribution of the source. The final estimate will then be the lowest out of the different methods to act as a lower bound and thus a min-entropy rate (entropy per bit).

The different methods are [11]:

- **Most Common Value Estimate (MCV)**: Uses the most common value in the sequence and computes the upper bound of the confidence interval of the value. Good estimator if there is biased noise in the sources.

- **Collision Estimate**: Measures the probability of repeated values in a sequence and is effective for biased noise.

- **Markov Estimate**: Considers the conditional probability of changing values, it captures interdependence within the sequence. Thus, a Markov Model can act as a viable estimator.

- **Compression Estimate**: Serves as a measure for the possible level of compression a dataset can undergo. Based on the principle that more random data is less compressible.

- **t-Tuple Estimate**: Examines blocks of t consecutive values (tuples) in the sequence and analyzes the frequency distribution of these tuples based on how predictable each block is.
- **Longest Repeated Substring Estimate (LRS)**: Evaluates the length of the sequence's repeating substring against the principle that longer repeating substrings correlate to less randomness.
- **Multi Most Common in Window Prediction Estimate (MultiMCW)**: A Sliding window variant of the MCV estimate.
- **Lag Prediction Estimate**: Predicts subsequent values in a sequence based on earlier values with a specified lag and identifies recurring patterns.
- **MultiMMC Prediction Estimate**: A lag variant of the Markov estimate that predicts subsequent values based on the most frequently observed transition utilising multiple Markov Models with Counting (MMC) sub-predictors.
- **LZ78Y Prediction Estimate**: Based on Bernstein's Yabba algorithm and LZ78 encoding. Updates substrings to a string dictionary from prior samples until the capacity limit is reached.

The min-entropy rate provides a useful metric for assessing the unpredictability of a source, serving as a lower bound for the source's entropy rate. For a completely unpredictable sequence of bits, the min-entropy rate is 100%. In theory, a fault-tolerant, noiseless quantum computer would produce entirely unpredictable outputs when the initial state is placed into a superposition. Consequently, the min-entropy rate can also be interpreted as an upper bound on the noise present in the source, as lower unpredictability indicates higher noise.

IV. GENERATING RANDOM NUMBERS FROM QUANTUM COMPUTERS

A. Protocol 1

One approach to generate randomness using quantum computers is to prepare n single qubits, place each in superposition and then measure each outcome. In this method, each qubit is placed in a superposition of the states $|0\rangle$ and $|1\rangle$ resulting in an equal probability of collapsing to either state upon measurement in the computational basis. We assume that the initial state, all gates and all measurements are error free.

State preparation:

$$|0\rangle^{\otimes n}$$

After applying the Hadamard gate to each qubit:

$$\frac{1}{\sqrt{2^n}}(|0\rangle + |1\rangle)^{\otimes n}$$

Measurement: Measure each qubit, collapsing the superposition into one of the 2^n possible states.

However, a significant challenge with this approach is the presence of noise in quantum systems. Quantum computers are susceptible to various sources of noise, such as decoherence and gate errors, which can distort the superposition states and lead to biased or incorrect measurements. This noise can compromise the quality and unpredictability of the generated random numbers, necessitating robust error correction and mitigation techniques to ensure the reliability of the results.
Due to the presence of noise in quantum systems, it is unlikely that this will be the true state generated. We assume, however, that the real state generated is a tensor product of arbitrary pure states, as opposed to some mixed state. This assumption is made to facilitate analysis despite the presence of noise in the system. We write this state as

$$\otimes_{i=1}^{n} [\alpha_i |0\rangle + \beta_i |1\rangle] \tag{2}$$

for $\alpha_i, \beta_i \in \mathbb{C}$ and $|\alpha_i|^2 + |\beta_i|^2 = 1$. We note that the index i allows the pure state to depend on each qubit i.

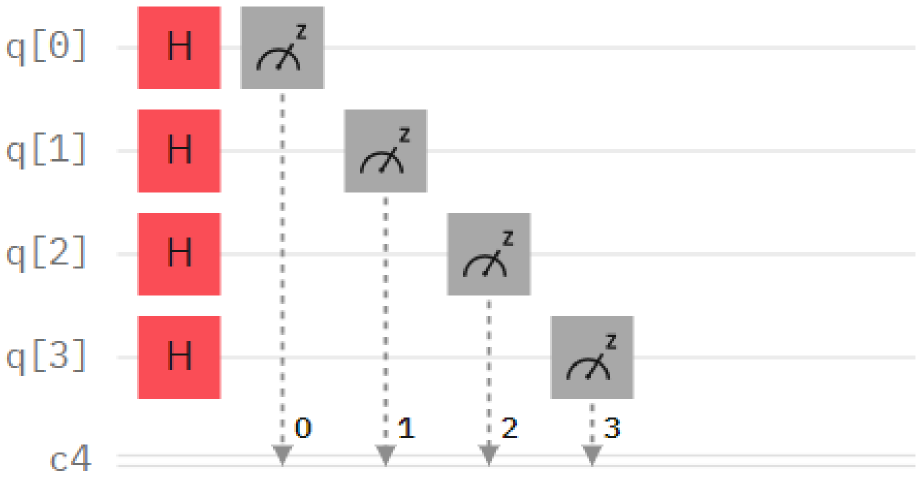

FIG. 1: Quantum circuit construction of Protocol 1 with 4 qubits (n=4)

B. Protocol 2

Protocol 2 extends Protocol 1 by employing two quantum devices, each following the same process to generate independent sources of randomness. The Von Neumann extractor is then applied to both sources independently to generate two sequences of random bits. The Von Neumann extractor takes pairs of consecutive bits and performs an operation to add to the output sequence depending on whether the pairs are equal or opposing values. If the bits are of opposing values (e.g. 01 or 10), the last value in the pair is amended to the final output sequence. If the bits are of equal value (e.g. 00 or 11) then the pair is discarded and nothing is amended to the final output sequence. This method ensures that the final sequence has an equal probability of 0s and 1s in the final sequence and is therefore unbiased. The method is also under the assumption that the initial bits are independent and have a fixed bias.

First, the min-entropy rate of each source is calculated using the NIST entropy estimator tool. After applying the Von Neumann extractor, we reassess the min-entropy rate of the outputs using the same tool.

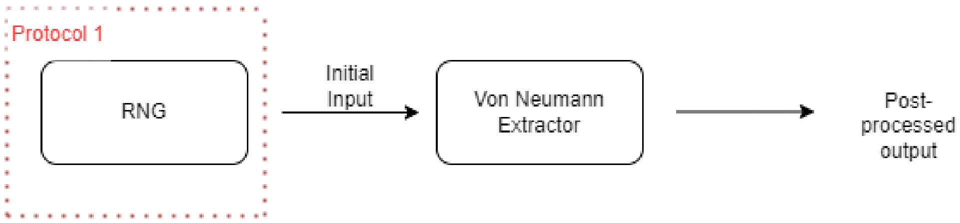

FIG. 2: This figure demonstrates the high-level setup of Protocol 2. The dashed red box represents the input to the extractor and is also the same implementation in Protocol 1

C. Protocol 3

Protocol 3 builds on Protocol 1 by utilising two quantum devices to generate independent sources of randomness. Each device implements the same process described in Protocol 1 to generate two independent sources of randomness. These two independent outputs from each device act as respective inputs for Protocol 3 which utilises a two-source extractor to combine the inputs to produce higher-quality randomness.

Initially, the randomness of the outputs from the quantum devices is assessed using the NIST entropy estimator tool to calculate the min-entropy rate for each source, which serves as a benchmark for the final output of Protocol 3.

Next, the respective outputs from the quantum devices are then fed into two-source cryptomite extractors. These extractors process the inputs to produce a single output with improved randomness.

Finally, the output of these extractors is assessed using the min-entropy rate measurement from the same NIST entropy estimator. In principle, the min-entropy rate after the extractor implementation should be higher than the min-entropy rate of both sources pre-implementation.

Thus, a post-extractor min-entropy rate above the benchmark signifies success in regards to Protocol 3 as it validates the protocol has enhanced the randomness of the initial two outputs from the quantum devices.

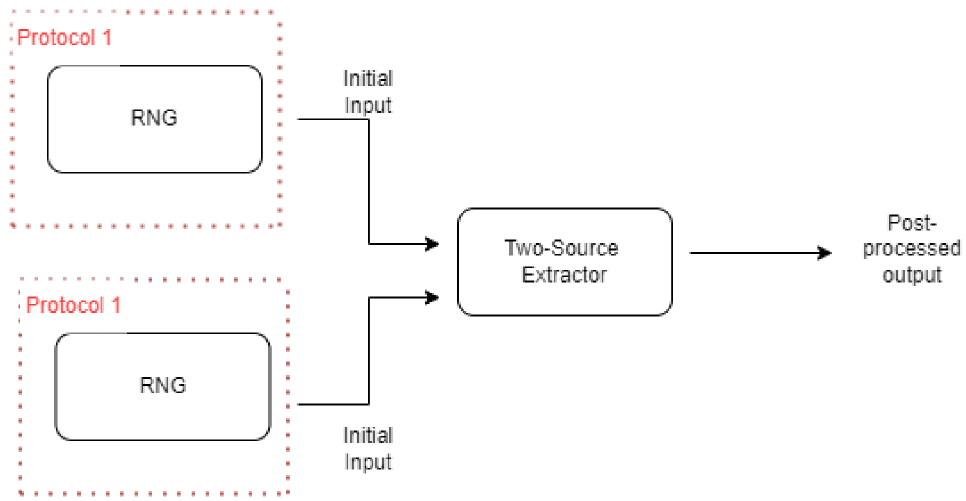

FIG. 3: This figure demonstrates the high-level setup of Protocol 3. The two dashed red boxes represent the input to the two-source extractor and are also the same implementation in Protocol 1

V. DODIS EXTRACTOR

The Dodis extractor is a method used to extract high-quality randomness from weakly random sources, often used in conjunction with quantum random number generators. The Dodis extractor in the Cryptomite library [6] is derived from the extractor utilised in Dodis [8]. Implementation of this extractor has some necessary conditions:

- The length of both input vectors n must be a prime number with a primitive root of 2. In number theory, a prime number p with a primitive root b is a prime number that all powers of b must generate all possible real numbers from 1 to $(p-1)$.
- The input vectors cannot be solely comprised of 0 or solely comprised of 1. This condition is intuitive, the input vector having insignificant randomness is not ideal.
- Matrices A_i are circulant matrices of size $n \times n$ where A_0 is the identity matrix and i represents the numerical horizontal index shift of each element within the matrix relative to A_0. The rank of any subset of such matrices is at least $n - r$ where r should be small (typically 1). High relative rank ensures the required information retention of the input vector.

A matrix-vector multiplication between the $(n \times n)$ circulant construction of one of the inputs and the $(n \times 1)$ initial vector of the other input generates a $(n \times 1)$ output vector. Each element in the output vector has a *modulo* 2 operation performed to normalise the output to just 0s and 1s. The first m bits of this vector are the resulting output of the extractor.

$$Ext(x,y) = \left[\begin{pmatrix} x_0 & x_1 & x_2 & \cdots & \cdots & \cdots & x_{n-1} \\ x_{n-1} & x_0 & x_1 & \cdots & \cdots & \cdots & x_{n-2} \\ x_{n-2} & x_{n-1} & x_0 & \cdots & \cdots & \cdots & x_{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots \\ x_2 & x_3 & x_4 & \cdots & \cdots & x_0 & x_1 \\ x_1 & x_2 & x_3 & \cdots & \cdots & x_{n-1} & x_0 \end{pmatrix} \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ \vdots \\ y_{n-1} \end{pmatrix} \right] \mod 2 \tag{3}$$

The matrix-vector multiplication has a time complexity of $O(n^2)$, so the problem scales very poorly for large input lengths that are typically required for cryptography. As stated in [5], this method can also be rewritten in a way to allows the original extractor from [8] to be written as a circulant matrix. This is the circular convolution of a reverse operation on one input vector and computing its inner product with the other input vector.

$$Ext(x,y)_i = \sum_{j=0}^{n-1} R(x)_{i-j} \cdot y_j \tag{4}$$

Here, $R(x)$ denotes an input vector x that has been reversed so the indices are essentially inverted. Each element of the output is calculated as a sum of element-wise products when we compute the circular convolution of the two input vectors. The circular convolution occurs due to negative indices resulting in cyclically looping around the vector after traversing to the end. An additional segment of the operation is that if $(i-j)$ exceeds n then the index will be $i \ mod \ n$. It is additionally computed with modulo arithmetic like the previous method to again normalise each element to $\{0,1\}^n$.

The method of circular convolution allows the use of the Number Theoretic Transform (NTT) by way of the convolution theorem. This is advantageous in this context as utilisation of NTT or the Fast Fourier Transform (FFT) gives a time complexity of $O(n \log n)$ which can be referred to as quasi-linear time and scales much better than the matrix-vector multiplication method.

When implementing the Dodis extractor, using the cryptomite.dodis version, the method employs the NTT instead of the FFT. The NTT is a discrete transform used to perform polynomial multiplications efficiently in the context of modular arithmetic. The FFT is a powerful tool for fast polynomial multiplication, but it relies on floating-point arithmetic, which can introduce small numerical errors due to the finite precision of floating-point representations. In cryptographic applications, even tiny errors can compromise the integrity and security of the system. The NTT, by using integer arithmetic, ensures exact calculations and thus maintains the integrity of the data. The NTT's reliance on integer arithmetic prevents errors that could arise from floating-point representations, ensuring the correctness required for secure cryptographic computations.

VI. RESULTS

Min-entropy has already been introduced as the chosen measure of randomness. After generating two bit-strings, each with a length of 2.54 million bits, the two sources are measured using an open-source entropy estimator tool provided by NIST. Then, for Protocol 3, a two-source extractor is used to distil the two sources into a single source of high-quality randomness. We perform this extraction with the Dodis and Toeplitz extractor, both from the cryptomite library. We present and compare the results of the min-entropy estimators for the initial sources and the extracted outputs. An extractor is then implemented on the sources to generate higher-quality randomness, which will be realised through an increase in the min-entropy. All the extractors in protocols 2 and 3 are implemented using the cryptomite library. For the purposes of more granularity, the results of entropy estimates from the other tests will be shown rather than just the min-entropy. It is worth mentioning that the IBM quantum computers were utilised as the choice of quantum computers due to IBM's cloud service allowing easy availability to execute quantum processes. For specificity, IBM Osaka and Kyoto were the IBM devices utilised due to being the most available devices at the point of use. Table I summarises the min-entropy rates obtained from different estimation methods applied to the input sources and outputs of the extractors.

No.	Estimation Test	P1:IBM-Osaka	P1:IBM-Kyoto	P2:IBM-Osaka	P2:IBM-Kyoto	P3: Toeplitz	P3: Dodis
1	MCV	0.987355	0.958350	0.994368	0.993671	0.995983	0.996245
2	Collision	0.936331	0.919683	**0.877650**	0.907937	0.896711	0.916378
3	Markov	0.990858	0.962268	0.995313	0.998212	0.997038	0.999298
4	Compression	0.897618	**0.824166**	0.934227	**0.808045**	**0.8314**	**0.862191**
5	t-Tuple	0.924425	0.917361	0.928347	0.928381	0.934852	0.936704
6	LRS	**0.809031**	0.995517	0.994387	0.993435	0.987351	0.977965
7	MultiMCW	0.995704	0.963393	0.994193	0.997629	0.996157	0.999207
8	Lag Prediction	0.981424	0.979918	0.991054	0.919597	0.997075	0.996625
9	MultiMMC	0.987896	0.958435	0.995848	0.919597	0.997841	0.997446
10	LZ78Y	0.987506	0.958398	0.994370	0.885345	0.996419	0.996713

TABLE I: Min-entropy rate for the 2 input bit-strings, and the output bit-string of the Toeplitz and Dodis extractors for each NIST entropy estimation method. The first input source of weak randomness is represented by P1:IBM-Osaka and the second input of weak randomness is represented by P1:IBM-Kyoto. P2:IBM-Osaka and P2:IBM-Kyoto represent the final output sequence from the von Neumann extraction applied on P1:IBM-Osaka and P1:IBM-Kyoto respectively. The values highlighted in bold in each column represent the respective lowest entropy estimation.

It is observable that the Dodis has the highest min-entropy rate of the two-source extractors, 0.862191, and exceeds that of the Toeplitz output by 3.7%. Both extractors also show an increase in the min-entropy compared to both initial sources. However, the Von Neumann extractor does have a higher min-entropy rate, 0.87765, when applied to the first initial source of weak randomness, P1:IBM-Osaka. When applied to the second initial source, P1:IBM-Kyoto, the min-entropy rate is below that of both two-source extractors and also below that of one of both the initial sources. Thus, the variance performance of the von Neumann extractor does show that it can improve randomness significantly in some instances but is ineffective in others.

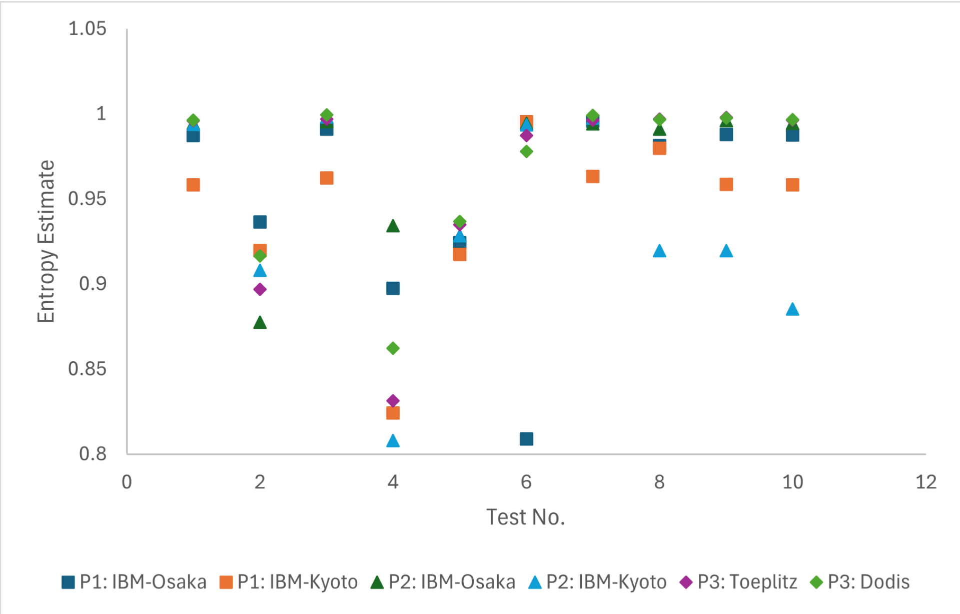

FIG. 4: Min-entropy rate against test number of the respective NIST entropy estimation test in Table 1 for the 2 input bit-strings, and the output bit-string of the Toeplitz and Dodis extractors.

VII. DISCUSSION

In this work, we have presented three protocols in order to generate high-quality randomness. The first protocol uses a quantum device to generate superstition states of qubits then measure each to generate a random bit-string. The second protocol builds on the first by using a Von Neumann extractor to extract randomness from the random bit-string in order to increase the unpredictability of the output. The third builds further by using a two-source extractor to combine two sources of weak randomness in order to generate a single output with improved unpredictability.

Following the results observed in this project, it can be surmised that Protocol 3 acts as an effective approach to generating high-quality random numbers. By developing a simple Hadamard circuit with two quantum devices to generate superposition states and measurements, two independent strings of bits with weak randomness were successfully generated. The random bits of these outputs are said to be weak due to the presence of noise in each quantum device so entropy estimation tools were used to act as a viable measure of unpredictability of these outputs and a benchmark for improvement. Classical post-processing was then executed using a Dodis two-source extractor to combine both weak sources to generate a single string of bits with more unpredictability. This enhanced randomness was validated by the min-entropy rate of the final output exceeding that of the benchmark of the two initial sources. The min-entropy rate increased from 0.81 and 0.82 of each respective source to 0.86 in the final output. Thus, Protocol 3 can be asserted as a more viable approach for high-quality randomness generation than Protocol 1 as the min-entropy rate of Protocol 1 would be equivalent to the rate of the two initial sources in Protocol 3 pre-extractor.
Protocol 2 showed some promising results but the failure to increase min-entropy on the second initial source does not suggest that it is more effective than Protocol 3. Reasons for the variance in Protocol 2 may be due to the Von Neumann extractor subsampling the size of the output significantly but further analysis would be required.

There are limitations to Protocol 3; the approach assumes that the two initial sources of weak randomness are single qubit systems and entanglement is not present in each. Future work could explore extractors that can generalize to systems of entangled qubits. Entangled qubit systems offer intrinsic benefits such as correlations amongst qubits that may provide some benefit to extracting randomness.

Returning to the limitations, the assertion that the unpredictability of the output increased post-extractor is also bound by the entropy estimation technique used. The approach was conservative with the use of a lower bound calculated on multiple entropy estimation algorithms, but an even more conservative extension would be to test for an increase in observed metrics from all entropy estimation algorithms. The results do show that the increase is seen across most estimation tests but the entropy rate of one of the initial sources is higher in Collision and Compression tests. Further work could be applied in using alternate entropy estimation techniques as there has been progress in machine learning based approaches that could offer benefits.

To add, the increase in entropy from Protocol 1 to P3:Dodis does not provide statistical significance across all entropy tests when subjected to statistical analysis using t-tests. For P1: IBM-Osaka to P3: Dodis, the p-value is 0.47 and for P1: IBM-Kyoto to P3: Dodis, the p-value is 0.27. This is a limitation of the project and further work should aim to attain results that show statistical significance across all tests.

Overall, the main takeaway from the project is that two-source extractors act as a viable method for handling the common issue of noise in quantum devices. Noise is an ever-present issue in quantum cryptography and hardware so methods that provide any way of mitigating or circumventing the issue are promising.

[1] A. Ezz-Eldien, M. Ezz, A. Alsirhani, A. M. Mostafa, A. Alomari, F. Alserhani, and M. M. Alshahrani, "Computational challenges and solutions: Prime number generation for enhanced data security," *PLOS ONE*, vol. 19, no. 11, pp. 1–28, 11 2024. [Online]. Available: https://doi.org/10.1371/journal.pone.0311782

[2] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010. [Online]. Available: https://doi.org/10.1017/CBO9780511976667

[3] S. L. Braunstein and A. K. Pati, "Quantum information cannot be completely hidden in correlations: Implications for the black-hole information paradox," *Physical Review Letters*, vol. 98, no. 8, Feb. 2007. [Online]. Available: http://dx.doi.org/10.1103/PhysRevLett.98.080502

[4] S. P. Vadhan, *Pseudorandomness*. Now Publishers Inc., 2012. [Online]. Available: https://people.seas.harvard.edu/~salil/pseudorandomness/pseudorandomness-published-Dec12.pdf

[5] C. Foreman, R. Yeung, and F. J. Curchod, "Statistical testing of random number generators and their improvement using randomness extraction," *arXiv preprint arXiv:2403.18716*, 2024.

[6] C. Foreman, R. Yeung, A. Edgington, and F. J. Curchod, "Cryptomite: A versatile and user-friendly library of randomness extractors," *arXiv preprint arXiv:2402.09481*, 2024.

[7] M. Berta and F. Brand˜ao, "Robust randomness generation on quantum computers," 2021.

[8] Y. Dodis, A. Elbaz, R. Oliveira, and R. Raz, "Improved randomness extraction from two independent sources," in *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, K. Jansen, S. Khanna, J. D. P. Rolim, and D. Ron, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2004, pp. 334–344.

[9] P. Austrin, K.-M. Chung, M. Mahmoody, R. Pass, and K. Seth, "On the impossibility of cryptography with tamperable randomness," *Algorithmica*, vol. 79, no. 4, p. 1052–1101, Dec. 2017. [Online]. Available: https://dblp.uni-trier.de/db/journals/algorithmica/algorithmica79.html#AustrinCMPS17

[10] P. Boes, J. Eisert, R. Gallego, M. P. Müller, and H. Wilming, "Von neumann entropy from unitarity," *Phys. Rev. Lett.*, vol. 122, p. 210402, May 2019. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.122.210402

[11] M. S. Turan, E. Barker, J. Kelsey, K. A. McKay, M. L. Baish, M. Boyle *et al.*, "Recommendation for the entropy sources used for random bit generation," *NIST Special Publication*, vol. 800, no. 90B, p. 102, 2018.