Pseudo Random Binary Sequence using Lattice-based Hard Problem for Quantum Key Distribution

Anupama Arjun Pandit[1] and Arun Mishra* arunmishra@diat.ac.in[1]

[1]Department of Computer Science & Engineering, SoCE&MS, Defence Institute of Advanced Technology, Pune, India

***Abstract*—Recently, there has been a growing interest among researchers in Pseudo-Random Binary Sequences (PRBS) for Quantum Key Distribution (QKD). QKD is a cryptographic protocol that distributes secret keys between senders and receivers, relying on random numbers to determine the basis for evaluating transmitted bits. Quantum Random Number Generators (QRNG) utilizing natural resources create random bit sequences for QKD. However, QRNGs are susceptible to natural noise, which compromises their security. Additionally, post-processing of QRNG-generated sequences is necessary to reduce noise, but this process slows down the random number generation speed in QKD.**

Non-deterministic Pseudo-Random Sequence Generators (NDPRSG) may overcome the QRNG's constraints, post-processing, and speed. Lattice-based NDPRSG provides security and efficiency.

The proposed work focuses on generating a uniform PRBS with non-deterministic entropy using Learning with Errors (LWE) and LFSR. NIST Statistical Tests are performed on generated PRBS for randomness analysis, and the results are reported. The proposed PRBS generation can be used for QKD, operating up to 20.17 Mbit/s.

***Index Terms*—Lattice-Based Cryptography, Linear Feedback Shift Register, Learning with Error, NIST Statistical Test Suite, Pseudo Random Binary Sequence**

I. Introduction

Cryptography is the study and practice of secure data transmission, with a focus on maintaining integrity, secrecy, and authentication. Quantum Key Distribution (QKD) [10] is an advanced quantum communication technique aimed at establishing secure keys between the source and destination. A crucial aspect of QKD security involves the use of Pseudo-Random Binary Sequences (PRBS) [2], which both parties utilize to determine the basis for measuring transmitted bits. In contrast to PRBS generated using mathematical algorithms, Quantum Random Number Generators (QRNG) [6] produce true randomness by exploiting the inherent unpredictability of quantum mechanics, resulting in True Random Numbers (TRN) [29] that hold significance in data processing applications.

However, the presence of noise and defects in the control mechanism can compromise the uniformity of the random numbers generated by QRNG, impacting the security of quantum computers [27]. To enhance QRNG security, self-testing or device-independent (DI) QRNGs have been proposed [13], which use Bell inequality tests [1] to ensure the unpredictability of both the source and the true random numbers. However, practical implementation of Bell testing is complex and may result in insufficient random numbers.

To address these challenges and ensure quantum-safe random number generation, researchers have made substantial progress in Post Quantum Cryptography (PQC) primitives that offer security against attackers with quantum capabilities, such as Cryptography Based on Lattice [31], Random Linear Codes [25], Multivariate Quadratic Equations [5], Elliptic Curve Isogeny [9]. Among PQC primitives, Lattice-based cryptography (LBC) [24] has emerged as a popular choice due to its speed, quantum-resistance, and lightweight nature. LBC provides a quantum-safe alternative to classical cryptography, featuring small key sizes and ease of implementation. The security of lattice-based crypto is dependent on hard problems like the Closest Vector Problem (CVP) [17], the Shortest Vector Problem (SVP) [18]. However, LBC has seen limited exposure to symmetric primitives like Pseudo-random Functions (PRFs) and Pseudo-random Generators (PRNGs). Banerjee et.al. [4] proposed Learning with Rounding (LWR) problem to develop a pseudo-random function, is used to construct PRNGs [21].

A major component of the PRBS generators used in cryptographic applications is established on Linear Feedback Shift Registers (LFSR), owing to their simplicity, low implementation cost, good statistical behavior, and the ability to use a mathematical model that allows the generator to be designed for optimal performance. LFSR generates a pseudo-random binary sequence with uniform distribution, which can be used directly for QKD. Although, if the seed is strongly correlated with (at least one of) the LFSR sequences, a correlation attack on that sequence might lower the effectiveness of a brute-force attack [16]. Correlation immunity can be provided by masking the seed.

a) Our Contribution:: Therefore, a quantum-safe, non-deterministic PRBS for QKD is proposed in the presented work. The presented approach to generate PRBS is based on Learning with Errors (LWE) and uses LFSR as a fundamental element to create PRBS. Objective of the proposed research

979-8-3503-1587-5/23/$31.00 ©2023 IEEE

is to produce a uniform PRBS based on LWE, in which the initial seed has been masked using LWE and this masked seed is utilized to feed LFSRS and generate uniform PRBS. Number Theoretic Transform (NTT) [12] method is used in the proposed approach since it's extensively used for increasing the speed of matrix multiplication. The PRBS's quality is tested using the NIST Statistical Test Suite (NIST STS). The generated PRBS can be used in the QKD application.

b) Organisation of the paper:: In Section II, related work as well as fundamentals of QRNG, PRBS and LBC are discussed. The proposed methodology for the production of PRBS is outlined in Section III. Section V presents a statistical analysis of the PRBS generated using proposed approach, along with a discussion on the method's applicability in QKD. Section VI presents conclusion.

II. Related work and Preliminaries

QRNG [6] uses quantum physics to generate random numbers with a high entropy source and can be generated from radioactive sources, electronic noises, quantum state of photons, etc. QRNGs produced by radioactive sources have substantial constraints, significantly limiting their applicability in the actual world. Furthermore, the radioactivity may influence the detectors, lowering their efficiency over time. Another limitation would be the detector's dead time, induced by a buildup of ions within the detector. Following a successful detection event, semiconductor and GM tubes need some time to recover fully. It is due to the detector's dead time, which precludes it from doing so immediately. Each of these details and the appropriate post-processing steps must be considered throughout the random bit generation process. The electrical Random Number Generators (RNG) based on electronic noises mostly use electronic components like diodes and resistors as the sources of entropy. In general, noise is generated due to the quantum nature of charge carriers. However, it is hard to isolate these effects into practice; consequently, the extraction of randomness from electronic noises suffers from such a limitation [14].

QRNGs that are based on the quantum state of photons [15] are sensitive to practical mistakes corresponding with the detectors used. The requirement of establishing a structure is significantly more complicated than prior systems.

Temporal QRNGs utilize the arrival time of photons as the seed to generate randomness. It uses specialized circuitry to convert the exponential dispersion of incoming photons into a uniform distribution.

Randomness is caused by devices that involve various photon-number states and also evaluate quantum states that contain multiple photons. Macroscopic Scale Photo detection, in this kind of QRNG random numbers are generated by monitoring more traditional parameters such as amplitude, intensity, and so on rather than individual photons. The design of such QRNGs seems complicated since extra care must be given to ensure that quantum noise is the predominant source of randomness.

A. Notations

The notation $[n]$ signify the set $1, 2, 3, \cdots, n-1, n$ for a natural number n. PPT is the most often used abbreviation for probabilistic polynomial time. Let us use $s' \leftarrow S'$ for a set S' to imply that s' would be selected at random and uniformly from S'. negl() signifies an arbitrary negligible function. In each lattice $\mathcal{L}$, the length of the shortest non-zero Euclidean vector is represented by $\lambda_1(\mathcal{L})$.

B. Pseudo random sequence

Pseudo-random sequences are those that are generated by deterministic algorithms in order to simulate truly random sequences. A Pseudo-Random Binary Sequence (PRBS) is a pseudo-random sequence in the unit interval [0, 1).
The Pseudo Random Sequence (PRS) component generates a pseudo-random sequence, which produces a pseudo-random bit stream using an LFSR.

Shift register has two components: a feedback function and a shift register. The shift register is made up of a series of bits. A shift register's **period** is the duration of the output sequence before it begins to repeat.
Definition 1: A shift register is referred to as a linear-feedback shift register (LFSR) if the input bit of the register is a linear function of the state it was in before.
To create a maximal-period LFSR, every polynomial formed by a tap sequence plus the constant polynomial should be primitive polynomial mod 2. A size of a shift register would be the degree of a polynomial.

An exclusive-or (XOR) function is the most frequently used linear function over single bits. LFSR is a shift register in which the input bit is generated by Bitwise XOR of certain binary digits of a total shift register value.

- The LFSR's initial value is referred to as the seed.
- Clocking meant that every LFSR goes through a single shift operation.
- The taps are the bit locations that impact the next state.
- The tap bits are sequentially XOR'd to the output bit and afterward returned to the lower left bit.
- The output stream is the series of bits in the rightmost location.
- The input and output taps are the first and final bits respectively.

C. Lattice-based Cryptography

The quantum-resistant cryptography on classical computers could be achieved by the PQC primitives. In terms of applications, security, and advantages over other PQC techniques, the lattice-based cryptography implementation is lightweight, efficient, and based on strong mathematical problems. Lattice-based cryptography primitives use the mathematical structure of lattices discussed by [7]. The security of lattice-based cryptography relies on its hard problems such as the SVP [18], CVP [17]. The lattice-based system is easy to solve in a two-dimensional system, but difficult to solve in high dimensions, even quantum computers cannot find a solution to the problem efficiently suggested [8].

Definition 2: A lattice $\mathcal{L}$ is a n-dimensional discrete additive subgroup of $\mathbb{R}^n$. Additive subgroup implies that it is a group $< \mathbb{R}^n, + >$ and for any $x, y \in \mathcal{L}$ the following properties are satisfied [24]:
$x + y \in \mathcal{L}$ Presence of an Identity Element $0 \in \mathbb{R}^n$. Presence of an Inverse element $-x \in \mathcal{L}$ is present.

Discrete nature of lattices implies that within a distance period if there is a point, it should belong to the lattice. Formally, for every $x \in \mathcal{L}$ and $\phi > 0$, the point $(x+\phi*B) \in \mathcal{L}$, where B is the basis of the lattice.

1) Average-case Hardness in the LBC: SIS and LWE are addressed as average-case problems intimately allied to cryptosystems using a random matrix A. They are at least as difficult as all cases of SIVP in dimension n when reduced from worst-case to average-case. The decision LWE problem is often used in lattice-based encryption methods. Although solving the search LWE problem instantaneously solves the related decision problem, the other way is only possible via a polynomial-time reduction. As a result, we opt to approach the decision problem since it is a simpler problem.

Search LWE (SLWE) : Assume we have an Oracle O_s^n that produces samples of the form $(A, < A, s > +e)$,
- For each sample, $A \leftarrow Z_q^n$ is picked at random.
- $s \in Z_q^n$ would be the 'secret' (the same for every sample).
- $e \leftarrow \chi$ is error picked at random for each sample based on χ .
The SLWE problem is to retrieve the secrets provided to O_s^n. The LWE$_{n,q,\chi}$ assumption states that the SLWE problem is significantly more challenging; this is specified below.
- Note: The problem would be straightforward if the 'error' bit e were not present: if we had n samples of the form $(A_1, < A_1, s >), \cdots, (A_n, < A_n, s >)$, we can solve for s using Gaussian elimination.
- LWE$_{n,q,\chi}$ assumption : It is true for every PPT algorithm Ag that :

$$\Pr_{s \leftarrow Z_q^n}[Ag^{O_s^n}(1^n) = s] = negl(n) \qquad (1)$$

One way for solving LWE by brute force is the greatest likelihood method. Assume, for the purpose of simplicity, that the error has a normal distribution and the variable q is polynomial. After $O(n)$ values are given to equations, the only correct assignment is the one that causes the equation to be fulfilled. This may be shown using a traditional argument based only on Chernoff's bound and a union bound that includes all of the $s \in Z_q^n$. The approach only requires $O(n)$ samples and can finish its task in $2nlogn$ time [26]. There are several indications that show the LWE problem will be difficult. First, since the most efficient LWE algorithms run in exponential time. To begin with, it has been shown that LWE is difficult based on particular assumptions regarding the worst-case complexity of standard lattice problems. Since the modulus q is exponential, the difficulty is established by the common assumption that estimating (decision version) SVP within polynomial factors is difficult [19]. Hardness, to be more explicit, is based on this premise.

D. Quantum Key Distribution

Quantum key distribution, also referred to as QKD, is a form of secure communication that is used to transmit encryption keys that are only known between parties who share information. The QKD protocol enables two parties to generate and share a random key, which can subsequently be put to use to encrypt and decode communications. The method of communication takes advantage of some of the properties that may be found in quantum physics in order to exchange cryptographic keys in a manner that is both verifiable and guaranteed to be secure. This was facilitated by utilizing quantum physics. In the field of QKD, the presence of randomness in the secret key is an essential prerequisite condition. In spite of the fact that others eavesdrop on the transmission medium, the randomness ensures that the communications on this channel are stored securely.

a) Working of QKD:: QKD works by sending numerous photons, or light particles, between parties over fiber optic lines. Each photon has a different quantum state, and the photons are delivered together to form a sequence of zeros and ones. Qubits are the binary equivalents of bits in this sequence of quantum states, which comprise zeros and ones. When a photon arrives at its destination, it passes through one beam splitter, forcing it to choose one of two paths into the photon collector. The receiver would then react towards the original sender, providing data describing the sequence of photons delivered, which the sender would then compare to the emitters, which would have emitted each photon. Photons in the incorrect beam collector are eliminated, leaving just a precise sequence of bits. This bit sequence could then be employed as a key to data encryption. Any mistakes or data leaks are eliminated during the error repair and other post-processing procedures. Delayed privacy amplification is another part of the post-processing phase. It gets rid of any information an eavesdropper might have gotten about the final private key.

III. Proposed Method for Pseudo Random Binary Sequence

As a QKD scheme is one of the applications of a PRBS based on hard problems in LBC, it is essential to take care of some important requirements like: fast generation, uniform distribution of the sequence, non repeatability, seed in-dependency, and security. To meet the requirements of QKD and overcome the challenges, we develop PRBS. The PRBS is generated in two parts: Seed masking using LWE and then sequence of uniform bit generation using LFSR. The proposed approach first uses the LWE algorithm on the seed in order to avoid attacks based on seed recovery. Next, LFSRs are initiated by generating long sequences of random bits with the secure seed. This approach is a basic implementation of a realistic non-deterministic PRBS generation based on a Lattice-based hard problem. This approach takes advantage of LWE's non-deterministic nature to generate PRBS which can inherit non-determinism.

A. Seed Masking using LWE

In order to ensure the safety of the seed, the architecture makes use of a lattice-based hard problem LWE. The Seed Masking with LWE (hard function) step is included at the beginning of the proposed PRBS generation process. The Gaussian error of the LWE problem allows for the necessary nondeterminism to be established, which is required for applications such as QKD. In addition, LWE concealing is the primary component that must be present for the suggested PRBS generation to have adequate levels of security.
Current work proposed the LWE-based seed masking function: Masking of Seed (seed r). Let $q, m,$ and n be integers. To mask a seed $r \in \{0,1\}^n$, sample a secret $s \in Z_q^n$. Select a uniform $A \leftarrow Z_q^{m\times n}$ and sample an error $er \in Z_q^m$. Finally, compute $seedb = A \cdot s + er + \frac{q}{2} \cdot r$ and output $seedb$, the seed hidden beneath the SeedMask function. The SeedMask function is described in pseudo-code below. **Non-determinism in Seed Masking:** Discrete Gaussian sampling is a significant element of Lattice-Based Cryptography. The Discrete Gaussian sampling increases security by including an error term in the matrix-vector multiplication $A \times s$. The matrix-vector multiplication requires large number of computations. In the proposed approach Number-theoretic transform (NTT) [12] has been used for fast multiplication as NTT performed multiplication in $O(nlogn)$ time while school-method performs same multiplication in $O(n^2)$ time. The distortion of the vector $A \cdot s$ contributes to the LWE samples being indistinguishable from random samples. Without the error, LWE would reveal sensitive information. Gaussian sampling across lattice vector s is used to generate the errors. It takes short vector s and adds them to the $A \cdot s$, yielding $A \cdot s + er$. Each lattice vector in the bell curve is sampled by giving a probability to it. As a result, we obtain different errors for different occurrences of time, resulting in a non-deterministic seed masking and $A \cdot s$ as a result.
The output of the Seed Masking process is $seedb$, which is non-deterministic. $seedb$ is used further for the initial feed of the LFSR to generate PRBS.

Algorithm 1 SeedMask(r) function

```
Choose uniform $s \leftarrow Z_q^n$
Sample $A \leftarrow Z_q^{m\times n}$
Sample Error vector $er \leftarrow Z_q^m$
Mask seed $r$ : $seedb = A \cdot s + er + r \cdot \frac{q}{2}$
Masked seed: $(seedb)$
return $(seedb)$
```

B. Sequence of binary bit generation using LFSR

According to the QKD requirements, perfect secrecy is achieved by using n random bits to generate n key bits. This means that the PRBS for QKD applications should be able to create millions of bits without running out of resources. To do this, the second part of this proposed method uses Linear Feedback Shift registers to generate sequences indefinitely. The masked seed - $seedb$ obtained by the seed masking procedure consisting of one polynomial is fed into the LFSR. Each $seedb$ polynomial has 256 coefficients, and each coefficient has 32 bits. As a result, the total amount of bits in $seedb$ is $256 \times 32 = 8192$. Running 4096 bits around LFSR could significantly influence PRBS speed because 4096 bits would have to be moved for every bit of shifting. This will result in an unnecessary delay in bit creation. As a result, we only use 130 bits of $seedb$. LFSR comprises single polynomial $seedb$ coefficients. We use the LSB coefficients to feed into our LFSR.The bit generation includes following steps:

- Step1: Choose a primitive polynomial as per the requirement of period. Proposed work uses 129 degree primitive polynomial with tap bits 129, 5, 0, and 75^{th} bit as a clocking bit.
- Step2: A linear shift register (LFSR) is initialized to zero, refer Fig. 1 .

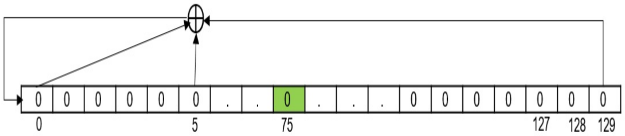

Fig. 1. LFSR initialized to zero

- Step3: Register clocked 130 times ignoring irregular clocking, refer Fig. 2 .

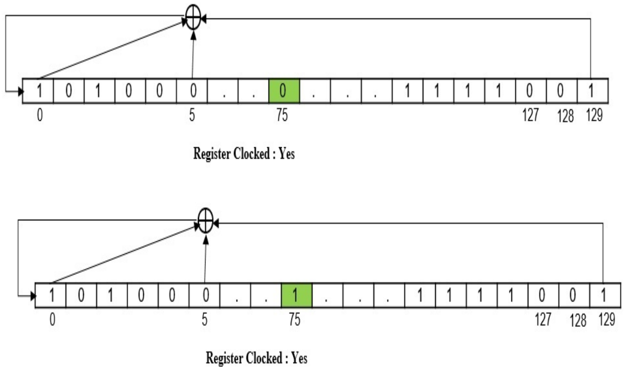

Fig. 2. Register clocked ignoring irregular clocking

In this step, 130 Key bits of a key $seedb$ are consecutively Xored with feedback refer Fig. 3 . Every cycle j $(0 <= j < 130)$ the bit $seedb[j]$ gets XORed with register's input bits and saved in the register's LSB.

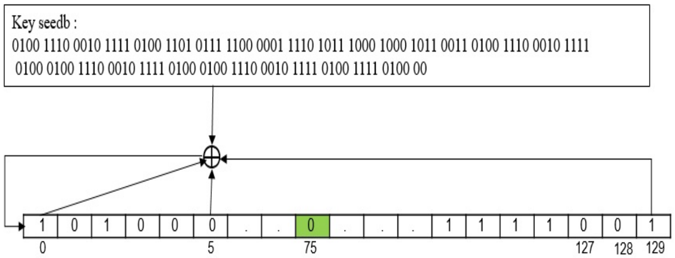

Fig. 3. seedb Xored with feedback

- Step4: Register clocked 200 times with irregular clocking (refer Figur 4): Irregular clocking follows the clocking rule, if clocking bit is one, the register will clock. The output of the register is ignored this step is called as warm-up phase. Here key gets diffused in the LFSR.

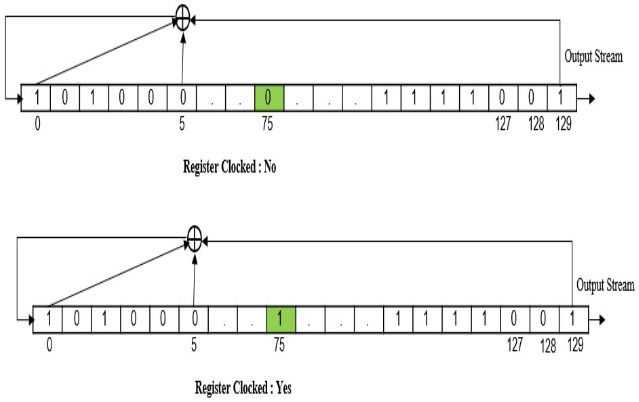

Fig. 4. Register clocked with irregular clocking

- Step5: Register clocked n times with irregular clocking and outputs n binary bits are stored as a PRBS in a file buffer.

C. Implementation details

This section discusses the seed masking implementation specifics. The seed masking using LWE is computed for suggested security [30] with modulus $q = 8380417$ and m, $n = 4$. As a result, matrix A contains a total of 16 elements. Every element of A is a polynomial in $Z_q^{m\times n}$ (Z_q^n for s). The polynomials are stored in a 256 word structure, with each word consisting of 32 bits representing a polynomial coefficient. Seed masking returns the masked seed $seedb$ as a m-row matrix.

Seed masking implementation consists of two essential computations: i) sampling matrix A and ii) polynomial multiplication. The proposed method employs a symmetric scheme SHAKE-128 [22] to produce the samples of matrix A. The short vectors s are sampled using rejection sampling, and the error er is created using a Gaussian distribution in the range $-1, 1$, ensuring that the disturbance caused by mistake is brief and does not reveal any secrets.

Number Theoretic Transform (NTT) [12] contributed by [12] has been used for low complexity up to $O(n\,log\,n)$ for the primary algebraic operation - multiplication of matrix A whose components are polynomials in $Z_q[X]/(x^{256}+1)$ by the secret vector s. NTT is an FFT variant that operates on the finite field Z_q rather than the complex numbers. In our scenario, the schoolbook method would require $4 \times 4 = 16$ polynomial multiplication. NTT simplifies multiplication to point-wise multiplication, which is particularly efficient for polynomials like those employed in this study.

IV. Security of the proposed PRBS

Ensuring the security of a PRBS is of paramount importance, especially in cryptographic applications. In this section, we discuss the security aspects of the proposed PRBS.

One of the key security features of the proposed PRBS is its resilience against quantum attacks. The scheme is built upon a lattice-based hard problem and utilizes LWE technique, which is known for its quantum resistance. The hardness of the LWE problem serves as a foundation for the security of the PRBS, making it resistant to attacks by both classical and quantum computers. This ensures that the generated random bit sequences remain secure even in the presence of powerful quantum adversaries. This is a crucial aspect, as traditional cryptographic methods may become vulnerable to quantum attacks, making the PRBS's post-quantum security a significant advantage.

The proposed PRBS leverages non-deterministic entropy generated through the hard lattice problem LWE using random error. This leads to the generation of truly random and unpredictable sequences of bits, enhancing the overall security of the PRBS. The incorporation of entropy from the LWE problem adds a layer of randomness that is difficult for attackers to predict, further bolstering the security of the generated random sequences.

The security of the PRBS is also evaluated through comprehensive statistical analysis and testing. The generated random bit sequences undergo rigorous testing using established cryptographic standards, including the NIST statistical tests. These tests ensure that the generated sequences exhibit the desired statistical properties of randomness, providing evidence of the PRBS's security.

V. Results and discussion

One of the most important testing suites for conducting randomness analysis is the NIST Statistical Test Suite (NIST STS) [28]. It is commonly used for the aim of getting formal certificates or approvals. It is a collection of randomness tests. 10^6 bits are selected from one sequence for each test.

The proposed PRBS has been evaluated according to all 15 test standards defined by NIST. We stored more than 1 gigabyte worth of output bits into a file, which was then used as input to the NIST test suite so that we could validate the randomness of the sequence. Table I clearly shows that the proposed PRBS passed 15 NIST tests. According to Intel® VTuneTM Profiler, the CPU time required to complete the proposed seed masking is 74.01 μs, and binary bits are generated at 20.17 Mbit/s.

As generated sequence has passed NIST 15 randomness tests which imply that generated sequence is uniformly distributed.

TABLE I
RESULTS OF NIST STS TEST FOR PROPOSED PRBS

Sl. No.	Test name	Proportion	Result
1.	Frequency Test (Mono-bit)	94.00	Passed
2.	Frequency within a Block Test	94.00	Passed
3.	Approximate Entropy Test	94.00	Passed
4.	Serial Test	94.00	Passed
5.	Maurer's "Universal Statistical" Test	94.00	Passed
6.	Discrete Fourier Transform (Spectral) Test	96.00	Passed
7.	Linear Complexity Test	96.00	Passed
8.	Overlapping Template Matching Test	94.00	Passed
9.	Non-overlapping Template Matching Test	94.00	Passed
10.	Cumulative Sums Test	98.00	Passed
11.	Random Excursions Test	93.00	Passed
12.	Random Excursions Variant Test	95.65	Passed
13.	Binary Matrix Rank Test	96.00	Passed
14.	Tests for the Longest-Run-of Ones in a Block	94.00	Passed
15.	Runs Test	94.00	Passed

The proposed technique accomplishes non-determinism by including the LWE function in our seed *seedb*.
To support non-repeatability, the proposed approach uses the LFSR using primitive polynomial with degree 130 which can generate non repeated sequence up to a maximum length $2^{130} - 1$. Further, Learning with Errors (LWE) problem's hardness is based on lattice based one functions SVP [18] and CVP [17] which are non solvable in polynomial time using quantum computers. Therefore, proposed approach is secured from a quantum attack.

1) Performance analysis: : As the majority of PRBS generators are deployed in real-time applications, and in certain circumstances in integrated devices with limited hardware, the performance and speed of a PRBS generating method are critical. We employ LFSR with LWE samples with enhanced pseudorandom features in the proposed PRBS generation. Table II includes the speed in Mbits/s as measured using time stamping.

A. Comparison of proposed PRBS with recent PRBS

The proposed PRBS is compared to existing approaches (on classical computers) in terms of randomness, security, and performance. The proposed approach is ingrained in a quantum-safe problem LWE & an LFSR which can produce a sequence of random bits. Proposed PRBS has an advantage over other PRBS schemes in that it holds satisfactory statistical properties & behaves complete randomly. The proposed PRBS scheme has the important advantage of being based on the PQC hard problem, which gives security against quantum adversaries. The suggested strategy is compared to other comparable approaches in table II. To analyze PRBS, the table employs criteria like statistical testing, speed, and post quantum security. The table depicts the evaluated features of several approaches, as well as the number of characteristics examined in each approach. The data show that proposed PRBS cleared most of the tests.

TABLE II
COMPARISON OF THE PROPOSED PRBS WITH EXISTING SCHEMES

Features	[3]	[23]	[20]	[11]	Proposed PRBS
Technique used	LFSR	LFSR+ Memristor	Chaotic map	Avalanche noise	LWE + LFSR
NIST SP 800-22	✓	✓	✓	✓	✓
Speed Analysis	-	✓	-	✓	✓
Quantum Safe	-	-	-	-	✓
Bit generation speed in Mbit/s	17.0	1.50	1.70	0.008	20.17

Remark: ✓denotes " achieved" and - denotes there is no result reported.

Applications of PRBS in QKD: To speed up the process sender Alice might create random numbers using the proposed PRBS and assign her bits on a random basis based on the PRBS sequence. Scenario is shown in Fig. 5.

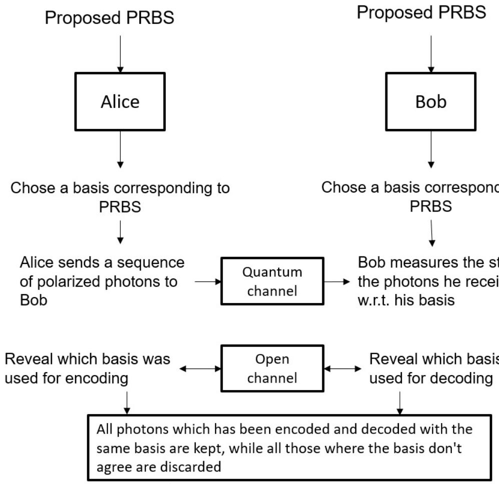
Proposed PRBS
Proposed PRBS
Alice
Bob
Chose a basis corresponding to PRBS
Chose a basis corresponding to PRBS
Alice sends a sequence of polarized photons to Bob
Quantum channel
Bob measures the state of the photons he receives w.r.t. his basis
Reveal which basis was used for encoding
Open channel
Reveal which basis was used for decoding
All photons which has been encoded and decoded with the same basis are kept, while all those where the basis don't agree are discarded

Fig. 5. QKD with proposed PRBS

Alice transmits a photon sequence to recipient Bob. Every photon in a polarisation state is equal to one /zero, but with a randomly selected basis.
Similarly, Bob may randomly generate sequence from the proposed PRBS in predicting Alice's basis.
Bob measures the state of the photons he gets, with each state assessed randomly.
An open channel connects Alice and Bob. Bob and Alice disclose which basis was utilized in decoding and encoding each photon. All photons decoded and encoded with a similar basis are retained, whereas those that did not agree with the basis are discarded. The proposed PRBS technique is an algorithm that feeds 'seed'into a well before mathematical model and produces a secure pseudo-random binary sequence at an incredibly rapid pace, with the statistical properties of the sequence guaranteed by the algorithm which can be used for QKD directly without any post processing.

VI. CONCLUSION

We propose a new approach for generating PRBS for QKD in this work. The proposed technique in the first part generates an initial uniform and random seed using quantum-safe, most significant basic LWE problem rather than classical mathematical hard problems. The second part uses LFSR to produce an indefinitely long random binary sequence that fits the LWE problem's theoretical security criterion. The proposed PRBS was generated at a rate of 20.17 Mbit/s. NIST statistical testing is being performed on the proposed PRBS. So, in future work, the improvement of PRBS generation will be considered by designing a more secure PRBS generator algorithm using more than one LFSR.

References

[1] Agresti, I., Poderini, D., Polacchi, B., Miklin, N., Gachechiladze, M., Suprano, A., Polino, E., Milani, G., Carvacho, G., Chaves, R., et al.: Experimental test of quantum causal influences. Science advances **8**(8), eabm1515 (2022)

[2] Ahrens, J.H., Dieter, U., Grube, A.: Pseudo-random numbers: A new proposal for the choice of multiplicators. Computing **6**(1-2), 121–138 (1970)

[3] AL-khatib, M.A.S., Lone, A.H.: Acoustic lightweight pseudo random number generator based on cryptographically secure lfsr. International Journal of Computer Network and Information Security **12**(2), 38 (2018)

[4] Banerjee, A., Peikert, C., Rosen, A.: Pseudorandom functions and lattices. In: Advances in Cryptology–EUROCRYPT 2012: 31st Annual International Conference on the Theory and Applications of Cryptographic Techniques, Cambridge, UK, April 15-19, 2012. Proceedings 31. pp. 719–737. Springer (2012)

[5] Berbain, C., Billet, O., Gilbert, H.: Efficient implementations of multivariate quadratic systems. In: Selected Areas in Cryptography: 13th International Workshop, SAC 2006, Montreal, Canada, August 17-18, 2006 Revised Selected Papers 13. pp. 174–187. Springer (2007)

[6] Bierhorst, P., Knill, E., Glancy, S., Zhang, Y., Mink, A., Jordan, S., Rommal, A., Liu, Y.K., Christensen, B., Nam, S.W., et al.: Experimentally generated randomness certified by the impossibility of superluminal signals. Nature **556**(7700), 223–226 (2018)

[7] Birkhoff, G.: Lattice theory, vol. 25. American Mathematical Soc. (1940)

[8] Chen, L., Chen, L., Jordan, S., Liu, Y.K., Moody, D., Peralta, R., Perlner, R.A., Smith-Tone, D.: Report on post-quantum cryptography, vol. 12. US Department of Commerce, National Institute of Standards and Technology … (2016)

[9] De Feo, L., Jao, D., Plût, J.: Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. Journal of Mathematical Cryptology **8**(3), 209–247 (2014)

[10] Gisin, N., Ribordy, G., Tittel, W., Zbinden, H.: Quantum cryptography. Reviews of modern physics **74**(1), 145 (2002)

[11] Lampert, B., Wahby, R.S., Leonard, S., Levis, P.: Robust, low-cost, auditable random number generation for embedded system security. In: Proceedings of the 14th ACM conference on embedded network sensor systems CD-ROM. pp. 16–27. New York, NY, USA (2016)

[12] Longa, P., Naehrig, M.: Speeding up the number theoretic transform for faster ideal lattice-based cryptography. In: Cryptology and Network Security: 15th International Conference, CANS 2016, Milan, Italy, November 14-16, 2016, Proceedings 15. pp. 124–139. Springer (2016)

[13] Lunghi, T., Brask, J.B., Lim, C.C.W., Lavigne, Q., Bowles, J., Martin, A., Zbinden, H., Brunner, N.: Self-testing quantum random number generator. Physical review letters **114**(15), 150501 (2015)

[14] Ma, X., Xu, F., Xu, H., Tan, X., Qi, B., Lo, H.K.: Postprocessing for quantum random-number generators: Entropy evaluation and randomness extraction. Physical Review A **87**(6), 062327 (2013)

[15] Ma, X., Yuan, X., Cao, Z., Qi, B., Zhang, Z.: Quantum random number generation. npj Quantum Information **2**(1), 1–9 (2016)

[16] Meier, W., Staffelbach, O.: Fast correlation attacks on certain stream ciphers. Journal of cryptology **1**, 159–176 (1989)

[17] Micciancio, D.: The hardness of the closest vector problem with preprocessing. IEEE Transactions on Information Theory **47**(3), 1212–1215 (2001)

[18] Micciancio, D.: On the hardness of the shortest vector problem. Ph.D. thesis, Massachusetts Institute of Technology (1998)

[19] Micciancio, D., Peikert, C.: Hardness of sis and lwe with small parameters. In: Advances in Cryptology–CRYPTO 2013: 33rd Annual Cryptology Conference, Santa Barbara, CA, USA, August 18-22, 2013. Proceedings, Part I. pp. 21–39. Springer (2013)

[20] Murillo-Escobar, M., Cruz-Hernández, C., Cardoza-Avendaño, L., Méndez-Ramírez, R.: A novel pseudorandom number generator based on pseudorandomly enhanced logistic map. Nonlinear Dynamics **87**, 407–425 (2017)

[21] Pandit, A.A., Kumar, A., Mishra, A.: Lwr-based quantum-safe pseudo-random number generator. Journal of Information Security and Applications **73**, 103431 (2023). https://doi.org/https://doi.org/10.1016/j.jisa.2023.103431

[22] Pardo, J.L.G., Gómez-Rodríguez, C.: The sha-3 family of cryptographic hash functions and extendable-output functions. In: Maple Document (2015)

[23] Rai, V.K., Tripathy, S., Mathew, J.: Memristor based random number generator: Architectures and evaluation. Procedia Computer Science **125**, 576–583 (2018)

[24] Regev, O.: Lattice-based cryptography. In: Advances in Cryptology-CRYPTO 2006: 26th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 2006. Proceedings 26. pp. 131–141. Springer (2006)

[25] Regev, O.: On lattices, learning with errors, random linear codes, and cryptography. Journal of the ACM (JACM) **56**(6), 1–40 (2009)

[26] Regev, O.: The learning with errors problem. Invited survey in CCC **7**(30), 11 (2010)

[27] Renner, R.: Security of quantum key distribution. International Journal of Quantum Information **6**(01), 1–127 (2008)

[28] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E.: A statistical test suite for random and pseudorandom number generators for cryptographic applications. Tech. rep., Booz-allen and hamilton inc mclean va (2001)

[29] Stipčević, M., Koç, Ç.K.: True random number generators. Open Problems in Mathematics and Computational Science pp. 275–315 (2014)

[30] Wunderer, T.: On the security of lattice-based cryptography against lattice reduction and hybrid attacks (2018)

[31] Zia, U., McCartney, M., Scotney, B., Martinez, J., Sajjad, A.: A novel pseudo-random number generator for iot based on a coupled map lattice system using the generalised symmetric map. SN Applied Sciences **4**, 1–17 (2022)