Lattice-based Cryptography for IoT in A Quantum World: Are We Ready?

Ayesha Khalid, Sarah McCarthy, Maire O'Neill
The Centre for Secure Information Technologies (CSIT)
Queens University Belfast (QUB), UK
Email: a.khalid@qub.ac.uk

Weiqiang Liu
College of Electronic Information and Engineering,
Nanjing University of Aeronautics and Astronautics, China
Email: liuweiqiang@nuaa.edu.cn

***Abstract*—The impending realization of scalable quantum computers has led to active research in Post Quantum Cryptography (PQC). The challenge is harder for embedded IoT (edge) devices, due to their pervasive diffusion in today's world as well as their stricter resources (tight area and energy budgets). Amongst various classes of quantum-resistant cryptography schemes, Lattice-based Cryptography (LBC) is emerging as one of the most viable, almost half of the 'survivors' of second round of the NIST's PQC competition are lattice-based in construction. This paper surveys the practicality of deployment of these schemes. In this context, the state-of-the-art LBC implementations on the constrained devices (including low-power FPGAs and embedded microprocessors), leading in terms of low-power footprint, small area, compact bandwidth requirements and high performance is fairly evaluated and bench-marked. The work concludes by identifying a suite of some favorite LBC schemes in terms of various IoT critical performance bench-marks.**

***Index Terms*—Quantum Safe cryptography, Post quantum cryptography, IoT security**

I. Introduction

With the societal shift towards the *Internet of Things (IoTs)*, ensuring security and privacy for an increasing number of heterogeneous connected devices is fast becoming a crucial concern. The IoT has become a reality as more and more of our devices are connected to the Internet. The influence of IoT in our day to day activities is set to further increase with a projected 25 billion connected devices by 2020, according to Gartner [1], while Cisco believes that by 2020, 50 billion devices will be network-connected [2]. IoT has the potential to truly revolutionize how we interact with the world today.

Quantum computers will also have a significant impact on todays security. Quantum computers will be capable of executing Shor's algorithm, that can, in polynomial time, break the two hard mathematical problems, i.e., integer factorization and discrete logarithm problem [3], on which RSA and ECC are based. These public-key schemes are used in today's security infrastructure to provide public-key encryption and (authenticated) key exchange. Reacting to this urgency, much research is now being conducted into *quantum-resilient* or *post-quantum* cryptography. The concern is also reflected by the stance of government agencies, including National Security Agency (NSA) and Communications-Electronics Security Group (CESG) [4]–[6]. NSA's Information Assurance Directorate (IAD) announced a transition to quantum resistant public-key cryptography in the near future for their Suite B of recommended algorithms [6]. The National Institute of Standards and Technology (NIST) in the US announced a call requesting new quantum-resilient algorithm candidates to be considered for analysis, standardization and eventually industry adoption [7].

Of the various flavors of quantum-resilient cryptography proposed to-date, lattice-based cryptography (LBC) stands out for various reasons. *Firstly*, these schemes offer security proofs based on NP-hard problems with average-case to worst-case hardness. *Secondly*, in addition to being quantum-age secure, the LBC implementations are notable for their efficiency, primarily due to their inherent linear algebra based matrix/ vector operations on integers. This makes them a favorite class to be considered for the IoT applications. *Thirdly*, LBC constructions offer extended functionality for advanced security services such as identity-based encryption (IBE) [8] attribute-based encryption (ABE) and fully-homomorphic encryption (FHE) [9], in addition to the basic classical cryptographic primitives (encryption, signatures, key exchange solutions) needed in a quantum age [10].

The IoT end user entities are generally portable, with small embedded processors, usually simple in design, limited in computational power and I/O capabilities, and have minimal power requirements. Many quantum resistant algorithms are more complex than the currently deployed public-key techniques. Their key sizes tend to be much larger too, making them at times impractical for low-cost devices. This work investigates the practicality of lattice-based post quantum schemes, both for digital signatures and key exchange, based on the following bench-marks critical to IoT applications.

- **Communication Bandwidth**: Most embedded processors are memory constrained and consequently well suited to smaller secuirty parameter set. For IoT applications with limited transmission bandwidth (e.g., wireless sensor networks), the minimum size of the ciphertext/ encapsulated key is critical in case of PKE/KEM. For digital signature schemes, a small sized public key, small digital signature and a range of supported hash output sizes is recommended.
- **Security Strength**: In the NIST call for PQC competition, the proposals invited had to classify the range

978-1-7281-0557-4/19/$31.00 ©2019 IEEE

of their algorithms security strength equivalent to the existing NIST standards in symmetric cryptography, i.e., a security strength of 1,2,3,4,5 (in order of increasing strength), which implies that any brute-force cryptanalytic effort requires computational resources comparable to (or greater than those) required for key search on a block cipher (or finding a hash collision) on an AES-128, SHA256, AES-192, SHA384, AES-256, respectively [7]. For most of the algorithms, these security levels offer a trade-off between performance (cost, resource, latency etc) and the required security; for IoT applications, higher security levels are generally less desirable due to their associated overhead, generally, the middle range NIST equivalent security (level 3) is chosen.

- **Performance**: Performance is often a function of the security level of the algorithm and the computing platform, in this context some typical low-resource IoT platform including FPGAs and microprocessors are undertaken.

The paper is outlined as follows: Section II gives a background of LBC proposals and their key components. Section III summarizes the state of the art in physical implementation reported against various constrained platforms. Section IV discusses some countermeasure challenges that still need to be addressed while Section V concludes the paper.

II. Background

A. Lattice-Based Primitives

Lattices are discrete subgroups in n-dimensional Euclidean space characterized by a regular arrangement of points. More precisely, a lattice in $\mathbb{R}^n$ generated by the basis $\mathrm{B} = \{\mathrm{b}_1, \mathrm{b}_2, \ldots, \mathrm{b}_n\}$, is defined as $\mathrm{L(B)} = \{\mathrm{Bx}, \mathrm{x} \in \mathbb{Z}^n\}$.

A number of hard mathematical problems are used to construct lattice-based schemes. The most commonly used problem is the Learning with Errors (LWE) problem which involves finding a vector $\mathbf{s}$ when given a matrix $\mathbf{A}$ and a vector $\mathbf{b} = \mathbf{As} + \mathbf{e}$ where $\mathbf{e}$ is a small (unknown) error vector. Other popular mathematical problems used to construct lattice-based schemes include the Short Integer Solution (SIS) and the NTRU assumption (associated with NTRU lattices).

There are three classes of lattices that are relevant for cryptography. Schemes that are based on LWE are **standard or random lattice-based schemes**. These schemes have in common that they require computations with large matrices that either need a lot of memory or require costly on-the-fly computations. A further issue with standard lattice-based schemes is that they require matrix-vector multiplication with quadratic complexity. **Ideal or ring lattice-based schemes** are an alternative to standard lattices. The major difference between these classes of lattices is that the matrix that is used in standard lattices is represented by a single row in ring lattices. The remaining rows are generated by cyclic shifts of the first row. Therefore ideal lattice-based schemes are more efficient as they require less memory and the main arithmetic operation is polynomial multiplication instead of matrix-vector multiplication. With the help of the number-theoretic transform (NTT) polynomial multiplication can be accelerated to have a complexity of $O(n \log n)$. In the case of ring lattices the security of the constructed schemes is based on ring variants of the original problems. Hence, the Ring-Learning with Errors (R-LWE) or Ring-Short Integer Solution (R-SIS) are the underlying problems used in these schemes.

While ideal lattice-based schemes are more efficient, the additional structure in the lattice might also be exploitable by attacks. So far no strong attack is known that exploits the ring structure or that is better than other attacks that work on standard lattices as well. To have a trade-off between the efficiency of ideal lattices and the trust in the security of standard lattices, **module lattices** were introduced. The difference between module lattices and standard lattices is that in module lattices the matrix has small dimensions and the coefficients of the matrix are no longer simple integers but entire polynomials. Therefore the number-theoretic transform can still be used for efficient polynomial multiplication. The security of module lattice-based schemes is once again based on variants of the original mathematical problems, e.g. Module-LWE or Module-SIS.

As one of the first lattice-based cryptosystems Hoffstein, Pipher, and Silverman introduced the encryption scheme NTRU [11] in 1998 which is based on ring lattices. To date the encryption scheme NTRUEncrypt has withstood cryptanalytic scrutiny provided parameters are chosen correctly, but the NTRU-based digital signature scheme is considered broken. However, a modified version of the signature scheme (pqNTRUsign) has been submitted to the NIST post-quantum call, along with many other proposals.

Table I presents a summary of the lattice-based schemes submitted to the NIST standardization process [7] and their related classes of lattices. Out of a total of 69 submissions to the NIST call for post quantum cryptographic proposals for digital signatures and KEM/encryption schemes, 26 are lattice-based proposals. Note that some schemes base their security on multiple assumptions. There are also two submissions based on **polynomial lattices**. This class is very similar to ring lattices and for power-of-two dimensions even equivalent.

In February 2019, NIST announced the selected 26 second-round candidates from the 69 first-round PQC candidates using the evaluation criteria specified in the original (security, cost, performance, implementation characteristics of the candidate) call [12]. The lattice-based schemes make the largest group of these schemes (12 out of the 26) and the only candidate having schemes belonging to the KEM and digital signatures category. Table I presents the lattice-based second round survivors of the NIST PQC competition highligted in blue color, the constituent schemes of two merged schemes NTRU (merger of NTRUEncrypt and NTRU-HRSS-KEM) and Round5 (merger of HILA5 and Round2) are highlighted via an italics font.

III. Performance Evaluation

To enable a fair performance evaluation, it is critical to identify the right performance bench-marks including latency, data/memory usage, security level etc since they will differ

Lattice Type	Schemes	
	KEM/PKE	Signatures
Standard	FrodoKEM Odd Manhattan LOTUS Compact LWE Giophantus	DRS
Ring, Standard	Lizard *Round 2* KCL EMBELM/R. EMBELM	
Ring	NTRU Prime *NTRU Encrypt* Ding Key KINDI LIMA NewHope *HILA5* *NTRU-HRSS-KEM* Mersenne-756839	qTESLA FALCON
Ring, Module		pqNTRUsign
Module	KYBER SABER Three Bears	DILITHIUM
Polynomial	Titanium LAC	

TABLE I
LATTICE-BASED PROPOSALS SUBMITTED TO NIST POST QUANTUM CRYPTOGRAPHY CALL, ALL SURVIVORS OF ROUND 2 AND *THE MERGED SCHEMES* IN THEM ARE HIGHLIGHTED.

for various applications. For constrained environments having low memory, the communication bandwidth is evaluated. This is followed by the reported implementation on embedded microprocessors, for which both the memory stack usage and the performance latency is bench-marked. Finally the best reported round 2 finalist LBC implementations on FPGAs are discussed and fairly bench-marked for efficiency.

A. Communication Bandwidth

Figure 1 shows the communication bandwidth of parameters (in bytes) of various lattice-based digital signature schemes that have successfully made it to the round 2 of the NIST PQC competition. A post fixed number at the end of the name of the scheme shows its security level. It can be seen that in terms of communication bandwidth, Dilithium offers fairly good performance, however, it does not offer the NIST equivalent security level 5. This highest security level might not be needed for most IoT applications scenarios. The private key is shown in the Figure 1, however it is not transmitted. Consequently, Falcon has the most compact parameters.

Figure 2 shows the communication bandwidth of parameters (in bytes) of various PKE/ KEM schemes that have successfully made it to the round 2 of the NIST PQC (excluding some merged schemes). For NewHope, a lattice-based cryptosystem of KEMs, two implementations are bench-marked since it achieves both CPA and CCA security. Also for Threebears, the ephemeral use case for the three security levels it claims is additionally bench-marked. Figure 2 does not show the communication bandwidth requirements for various versions of Frodo due to their large sizes compared to rest of the schemes (the sk/pk/ciphertext for Frodo-1 and Frodo-5 is 19872/9616/9736 and 31272/15632/15768 bytes, respectively). It can be seen from that SABER has very competitive performance among all lattice-based candidates for post-quantum key exchange. It achieves one of the lowest costs for bandwidth at each security level.

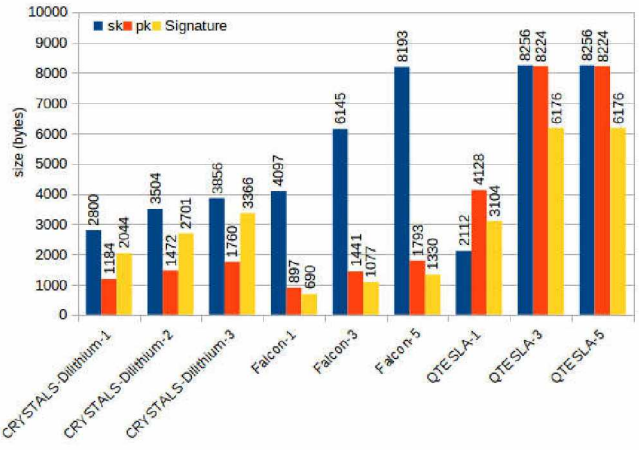

Fig. 1. Communication bandwidth parameter comparison for various flavors of NIST round 2 lattice-based signature contestants.

B. Reported Implementations on Embedded Microprocessors

1) Post-quantum crypto library for the ARM Cortex-M4 (PQM4): The PQM4 library is an open-source bench-marking and testing framework [13], started by the EU H2020 funded PQCRYPTO project [14]. The ARM Cortex-M4 processor is a family of high performance embedded processors, offering high-efficiency processing with the low-power, low-cost and ease-of-use benefits [15], it is the NISTs official recommended platform for microcontroller implementations. The PQM4 framework targets the STM32F4 Discovery board featuring an ARM Cortex-M4 CPU, 1MB of Flash, and 192KB of RAM.

PQM4 has of 10 post-quantum KEM implementations currently, *all except one of them are lattice-based in their construction*. Figure 3 shows the stack usage of some of the most efficient KEM implementations from PQM4. These implementations have been optimized in assembly using techniques specific to Cortex-M4. The implementations target NIST equivalent security (level 3), unless no level-3 parameters for that algorithm are available or if level-3 parameters exceed the development board's resources (in particular RAM) [13]. . It is easy to see that CRYSTALS-Kyber and SABER give the most competitive stack sizes here.

Figure 4 gives the average cycle counts of the KEM implementations from PQM4. Hence for Kyber-1 requiring 7269/9879/10189 clock cycles for Key Gen. /Enc./Dec., respectively, on an ARM Cortex-M4 CPU running on a 24MHz, generates 3302/2429/2355 operations per second. It is important to note here that Kyber scheme is between 2 and 4 orders

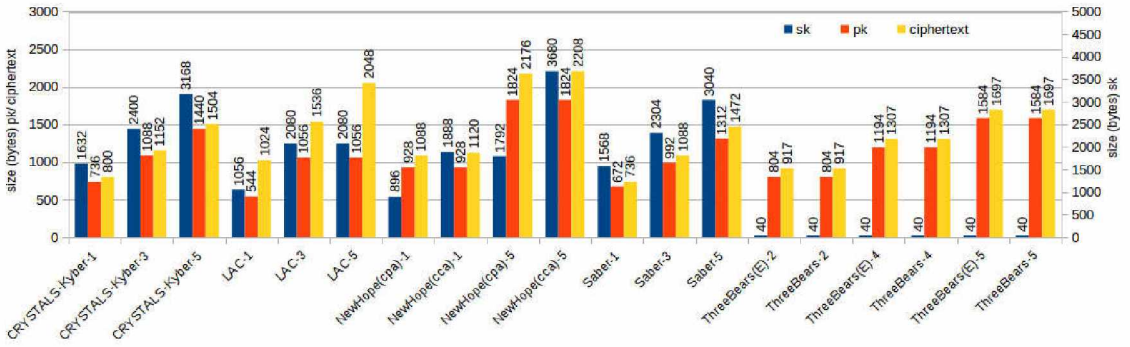

Fig. 2. Communication bandwidth parameter comparison for various flavors of NIST round 2 lattice-based KEM contestants.

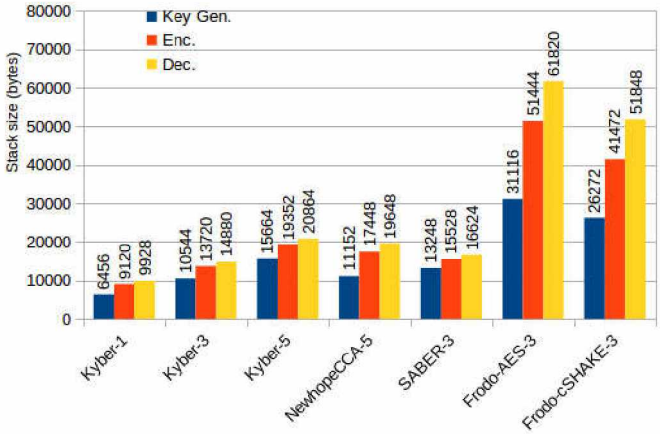

Fig. 3. Stack usage for various KEM implementations currently included in PQM4 [13].

of magnitude faster in generating keys and performing the encapsulation/decapsulation of the secret key. Kyber keys are however larger than the SIKE keys (a supersingular isogeny based KEM scheme). Kyber private keys are about four times the size of the SIKE private keys Kybers public keys and ciphertext are twice the size of the SIKE keys; however, the SIKEp751 reference implementation submitted to PQM4 [13] is much slower (orders of magnitude) than the lattice-based schemes, requiring 3525M, 5712M, 6139M for key generation, encapsulation and decapsulation, respectively.

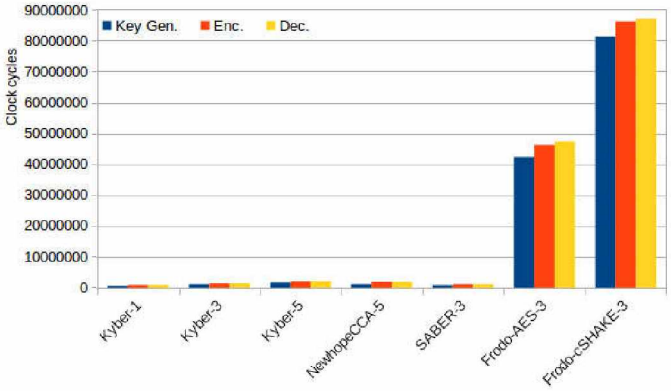

Fig. 4. Execution clock cycles taken by various KEM implementations currently included in PQM4 [13].

PQM4 library currently contains 3 post-quantum signature schemes targeting the ARM Cortex-M4 family of microcontrollers.

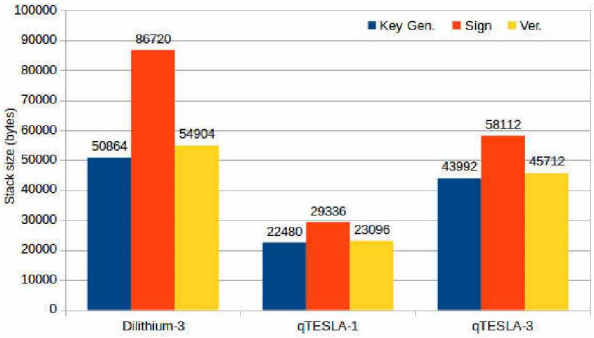

Fig. 5. Stack usage for various signatures schemes implementations currently included in PQM4 [13].

Figure 5 and Figure 6 give the stack usage and the average cycle counts of some digital signature schemes for PQM4, respectively. For Dilithium-3 requiring 2322955/9978000/2322765 clock cycles for Key Gen. /Signing/Verification, respectively, on an ARM Cortex-M4 CPU running on a 168MHz requires 14/60/14 ms for each of these operations, respectively.

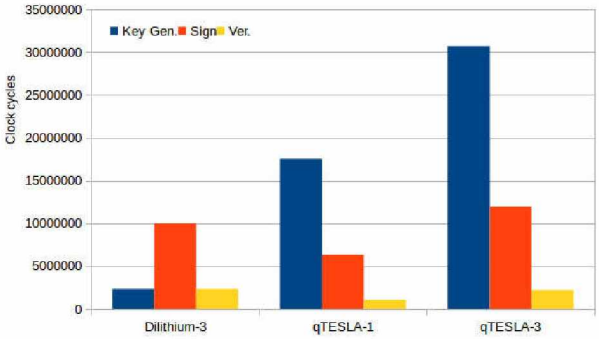

Fig. 6. Execution clock cycles taken by various signatures schemes implementations currently included in PQM4 [13].

2) More Optimized Implementations on ARM Cortex-M4: Though PQM4 [13] contains the most comprehensive collection of the LBC schemes on of ARM Cortex-M4, we now discuss some implementations not yet been included in it. Table II shows some of the most competitive results of lattice-

based signature schemes, that have made it to the second round of NIST PQC competition.

TABLE II
DYNAMIC MEMORY USAGE (IN BYTES) AND THE CLOCK CYCLE COUNTS FOR VARIOUS LEADING LATTICE-BASED PQC NIST SECOND ROUND KEM CONTESTANTS ON AN ARM CORTEX-M AT 168 MHZ.

Scheme	Ref.	Operation	Cycles	Time (ms)	Stack (Bytes)
Lattice-based PQC KEMs					
Saber	[16]	Key Gen	1147000	7	13883
(speed)		Enc.	1444000	9	16667
		Dec.	1543000	9	17763
Saber	[16]	Key Gen	1165000	7	6931
(memory)		Enc.	1530000	9	7019
		Dec.	1635000	10	8115
Kyber-1	[13]	Key Gen	726921	4	6456
		Enc.	987864	6	9120
		Dec.	1018946	6	9928
Kyber-3	[13]	Key Gen	1200291	7	10544
		Enc.	1446284	9	13720
		Dec.	1477365	9	14880
Kyber-5	[13]	Key Gen	1771729	11	15664
		Enc.	2142912	13	19352
		Dec.	2188917	13	20864
NewHopeCCA-5	[13]	Key Gen	1243729	7	11152
		Enc.	1963184	12	17448
		Dec.	1978982	12	19648
FrodoKEM	[17]	Key Gen	101273066	603	35484
-AES-3		Enc.	106933956	637	63484
		Dec.	107393295	639	63628
FrodoKEM	[17]	Key Gen	187070653	1114	33800
-cSHAKE-3		Enc.	253735550	1510	57968
		Dec.	254194895	1513	58112
Lattice-based PQC signatures					
Falcon-1	[18]	Key Gen.	114546135	682	63652
		Sign	80503242	479	63653
		Verify	530900	3	63654
Falcon-5	[18]	Key Gen.	365950978	2178	120596
		sign	165800855	987	120597
		verify	1046700	6	120598
Dilithium-3	[19]	Key Gen.	2320362	14	50488
		Sign	8348349	50	86568
		Verify	2342191	14	54800
qTESLA-3	[13]	Key Gen	30720411	183	43992
		Sign	11987079	71	58112
		Verify	2225296	13	45712
Classical schemes					
ECC-256	[20]	Key Gen.	12713277	76	-
		Sign	13102239	78	-
		Verify	24702099	147	-
RSA-2048	[20]	Key Gen.	-	-	-
		Sign	228068226	1358	-
		Verify	61951481	369	-

Out of the various lattice-based post-quantum key encapsulation schemes, Saber stands out both in terms of its resource-constrained nature for a small memory foot print but also in terms of throughput performance. In [16], the authors claim to exploit a memory efficient Karatsuba and just-in-time strategy for generating the public matrix of the module lattice to reduce the memory footprint; consequently *speed efficient* and *memory efficient* versions are reported in Table II. The speed-optimized implementation of Saber is faster than NewHope-CCA and Frodo in all aspects. Saber is faster than Kyber-3 in key generation and encapsulation, but marginally slower in decapsulation [13]. Frodo is much slower than Kyber/ NewHope since they are based on module/ideal lattices exploiting NTT for polynomial multiplication. Hence any decently optimized ideal lattices based scheme will always be faster than the standard lattices based schemes, targeting a similar security level [17].

The Falcon signature scheme offers 3 levels of NIST equivalent security and has the smallest public key and signature sizes among all lattice-based signature scheme submissions (as shown in Figure 1). The large Falcon tree used in the fast Fourier sampling in the signature generation of Falcon is the major bottle neck for memory usage and the authors of [18] tried to reduce the memory footprint by merging the tree generation and the fast Fourier sampling step into a single algorithm. This results in a compact implementation, the performance for the level-1 and level-5 is shown in Table II. For CRYSTALS-Dilithium, the NTT of the reference implementation is optimized at assembly level by merging of two of the eight stages of the NTT to reduce memory accesses [19]. CRYSTALS-Dilithium takes the lead here in terms of better overall throughput performance compared to both qTESLA and Falcon while qTESLA reference implementation from [13] has smaller stack requirements. Reference to classical schemes is given for comparison.

TABLE III
RESOURCE CONSUMPTION AND PERFORMANCE FOR VARIOUS LEADING LATTICE-BASED PQC NIST SECOND ROUND SIGNATURE CONTESTANTS ON VARIOUS XILINX FPGA DEVICES.

Scheme, Ref., Device	Op.	LUT/FF/Slice	DSP/BRAM Freq. (KHz)	Clock Cycles	Op.s /sec
Lattice-based PQC Signatures					
FrodoKEM-640	K.Gen	6621/3511/1845	1/6/167	3276800	51
(cSHAKE)	Enc.	6745/3528/1855	1/11/167	3317760	50
	Dec.	7220/3549/1992	1/16/162	3358720	48
FrodoKEM-976	K.Gen	7155/3528/1981	1/8/167	7620608	22
[17], Artix-7	Enc.	7209/3537/1985	1/16/167	7683072	22
	Dec.	7773/3559/2158	1/24/162	7745536	21
Lattice-based PQC KEMs					
NewHope	Client	5142/4452/-	2/4/125	171124	730
[21], Artix-7	Server	4498/4635/-	2/4/117	179292	653

C. Reported Implementations on FPGAs

Table III shows the only two FPGA implementations for various LBC KEM schemes that have made it successfully to NIST's PQC competition's second round reported (no LBC signature schemes hardware reported till date). In [21], authors implement FrodoKEM on a low-cost FPGA. Since Frodo is based on standard lattices, their associated large parameters make them an unpopular choice for embedded devices implementation. This work breaks this myth by undertaking conservative post-quantum cryptography practical on small devices and also contributes to the practicality in the evaluation of a post-quantum standardization candidate.

The FPGA design targets a balance between area consumption and throughput performance; a single DSP multiplier is used, operational parallelism is exploited whenever possible, BRAMs are re-used to reduce overall memory consumption. On a Xilinx Artrix-7, a single FrodoKEM-640 (cSHAKE) decapsulation operation (the computationally most expensive operation) needs 7,220 look-up tables (LUTs), 3,549 flip-flops (FFs), a single DSP, and only 16 block RAM modules. The maximum clock frequency is 162 MHz and it takes 20.7 ms for the execution of the decapsulation.

The reported NewHope FPGA implementation [21] on a Xilinx Aritx-7 caries out an NTT based polynomial multiplication and uses a Binomial sampler to generate error polynomials. It maintains a low-area footage, with a decent superior performance.

IV. Challenges - looking forward

Following two areas need immediate attention of the PQC researchers!

- **Instruction Set Extension (ISE) Exploration:** Performance bottlenecks for some established LBC schemes should be targeted for achieving acceleration via design space exploration for specialized ISE and the associated area overheads bench-marked, no such work is reported till date. The most efficient ISE recommendations can serve as a road map to be taken up by other computing platforms.
- **Side channel analysis attacks for LBC are understudied:** LBC constructions are relatively new and a comprehensive analysis of their resistance against physical attacks is of utmost importance before their widespread deployment [22]. There is a wealth of useful techniques to learn from traditional physical attack-resistant cryptographic designs used today but as new lattice-based designs emerge and the volume of their deployment increases, further new attacks will most likely surface and this will continue to be an important area of research going forward.

V. Conclusion

Lattice-based cryptography shows a promise as a quantum-safe alternative to existing public-key cryptosystems. They easily become the best fit in terms of key sizes compactness and simplicity of implementation, when compared against other quantum-safe alternative schemes. However, compared to the traditional Public key schemes, the performance of LBC schemes suffer with associated large public key sizes, which is a challenge for real world systems. This work surveys the state-of-the-art LBC implementations on the constrained devices (including FPGAs and embedded microprocessors) to give an idea how much is being achieved already. In this context, the road map to have schemes with inherent side channel attacks (SCA) resilience and a thorough study of ISE extension of current embedded processors for further performance enhancement needs to be done.

References

[1] V. M. J. Rivera and R. Gartner, "4.9 billion connected things will be in use in 2015," *The Washington Post*, Feb 2016. [Online]. Available: http://www.gartner.com/newsroom/id/2905717

[2] Cisco, "Internet of things (IoT)," *The Washington Post*, July 2015. [Online]. Available: http://www.cisco.com/web/solutions/trends/iot/portfolio.html

[3] P. W. Shor, "Algorithms for quantum computation: Discrete logarithms and factoring," in *Proceedings 35th annual symposium on foundations of computer science*. Institute of Electrical & Electronics Engineers (IEEE), 1994, pp. 124–134.

[4] CNSS, "Use of public standards for the secure sharing of information among national security systems," Committee on National Security Systems: CNSS Advisory Memorandum, Information Assurance 02-15, July 2015.

[5] CESG, "Quantum key distribution: A CESG white paper," February 2016. [Online]. Available: https://www.cesg.gov.uk/white-papers/quantum-key-distribution

[6] National Security Agency, "Commercial national security algorithm suite," August 2015. [Online]. Available: https://www.iad.gov/iad/programs/iad-initiatives/cnsa-suite.cfm

[7] D. Moody, "Post-quantum cryptography: NIST's plan for the future," Talk given at PQCrypto Conference, February 2016. [Online]. Available: https://pqcrypto2016.jp/data/pqc2016_nist_announcement.pdf

[8] T. Güneysu and T. Oder, "Towards lightweight identity-based encryption for the post-quantum-secure internet of things," in *18th International Symposium on Quality Electronic Design, (ISQED)*. IEEE, 2017, pp. 319–324. [Online]. Available: https://doi.org/10.1109/ISQED.2017.7918335

[9] T. Pöppelmann, M. Naehrig, A. Putnam, and A. Macías, "Accelerating homomorphic evaluation on reconfigurable hardware," in *Cryptographic Hardware and Embedded Systems (CHES)*, 2015, pp. 143–163.

[10] J. Howe, T. Pöppelmann, M. O'Neill, E. O'Sullivan, and T. Güneysu, "Practical lattice-based digital signature schemes," *ACM Transactions on Embedded Computing Systems (TECS)*, vol. 14, no. 3, p. 41, 2015.

[11] J. Hoffstein, J. Pipher, and J. H. Silverman, "NTRU: A ring-based public key cryptosystem," in *Algorithmic Number Theory, 1998*, 1998, pp. 267–288.

[12] NIST, "Status report on the first round of the NIST post-quantum cryptography standardization process," February 2019. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/ir/2019/NIST.IR.8240.pdf

[13] PQM4, "Post-quantum cryptography on ARM Cortex-M4 family of microcontrollers," February 2018. [Online]. Available: https://github.com/mupq/pqm4

[14] PQCRYPTO, "Post-quantum cryptography for long-term security PQCRYPTO ICT-645622," February 2015. [Online]. Available: https://pqcrypto.eu.org/

[15] ARM, "The ARM Cortex-M4 processor," February 2018. [Online]. Available: https://developer.arm.com/ip-products/processors/cortex-m/cortex-m4

[16] A. Karmakar, J. M. B. Mera, S. S. Roy, and I. Verbauwhede, "Saber on ARM," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 243–266, 2018.

[17] J. Howe, T. Oder, M. Krausz, and T. Güneysu, "Standard lattice-based key encapsulation on embedded devices," *IACR Transactions on Cryptographic Hardware and Embedded Systems*, pp. 372–393, 2018.

[18] T. Oder, J. Speith, K. Höltgen, and T. Güneysu, "Towards practical microcontroller implementation of the signature scheme Falcon," in *International Conference on Post Quantum Cryptography*. Springer, 2019, pp. 1–17.

[19] T. Güneysu, M. Krausz, T. Oder, and J. Speith, "Evaluation of lattice-based signature schemes in embedded systems," in *25th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*. IEEE, 2018, pp. 385–388.

[20] UM0586, "STM32 cryptographic library," February 2018. [Online]. Available: https://www.st.com/resource/en/user_manual/cd00208802.pdf

[21] T. Oder and T. Güneysu, "Implementing the NewHope-simple key exchange on low-cost FPGAs," in *International Conference on Cryptology and Information Security in Latin America*. Springer, 2017.

[22] A. Khalid, T. Oder, F. Valencia, M. O'Neill, T. Güneysu, and F. Regazzoni, "Physical protection of lattice-based cryptography: Challenges and solutions," in *Proceedings of the Great Lakes Symposium on VLSI*. ACM, 2018, pp. 365–370.