UDC 004.056.55 DOI: 10.26565/2519-2310-2021-1-02

RESEARCH OF IMPLEMENTATION OF CANDIDATES OF THE SECOND ROUND OF NIST PQC COMPETITION FOCUSED ON FPGA XILINX FAMILY

Marina Yesina, Bogdan Shahov

V. N. Karazin Kharkiv National University, Svobody Sq., 4, Kharkiv, 61022, Ukraine
m.v.yesina@karazin.ua, bogdanshahov2000@gmail.com

Reviewer: Serhii Toliupa, Doctor of Sciences (Engineering), Full Prof., Taras Shevchenko National University of Kiev, 81 Lomonosova St., Kyiv, 03189, Ukraine.
tolupa@i.ua

Received on December 2020

Abstract: *Today, the question of the stability of modern existing cryptographic mechanisms to quantum algorithms of cryptanalysis in particular and quantum computers in general is quite acute. This issue is actively discussed at the international level. Therefore, in order to solve it, NIST USA has decided to organize and is currently holding a competition for candidates for post-quantum cryptographic algorithms NIST PQC. The result of the competition should be the acceptance for standardization of cryptographic algorithms of different types - asymmetric encryption, key encapsulation and electronic signature (at least one algorithm of each type). At the beginning of the competition for the standardization process, 82 algorithms were presented. Based on the minimum eligibility criteria defined by NIST, 69 algorithms were considered for the 1st round. Given several parameters – security, cost, performance, implementation characteristics, etc., 43 and 11 algorithms were excluded at the end of the 1st and 2nd rounds, respectively, and the other 15 algorithms were saved for the 3rd round. The algorithms left in the 2nd round can be divided into 5 different categories depending on the mathematical basis on which they are based: based on the isogeny of elliptic curves, based on algebraic lattices, based on mathematical code, based on multivariate transformations and based on hash functions. Security is the main evaluation criterion that determines competition in the NIST competition, and it is clear that candidates' software implementations are mainly focused on it. However, it is extremely important that the algorithm has an effective hardware implementation. And timely detection of hardware inefficiencies will help focus the cryptographic community's efforts on more promising candidates, potentially saving a lot of time that can be spent on cryptanalysis. This paper discusses and compares the FPGAs of Xilinx family. Data on the implementation of the candidates of the 2nd round in the process of standardization of post-quantum cryptography NIST, which are focused on the FPGA of the Xilinx family, are presented and compared.*

Keywords: *electronic signature; post-quantum cryptography; NIST PQC competition; FPGA, Xilinx.*

1 Introduction

Today, the problem of the stability of existing cryptographic defence mechanisms to quantum cryptanalysis algorithms and quantum computers in general is quite acute. This is an issue under discussion at the international level. And for its decision, NIST USA decided to organize and holds today a competition for post-quantum cryptographic algorithms NIST PQC. The result of the competition should be the adoption for standardization of algorithms such as asymmetric encryption, key encapsulation and electronic signature (*at least one algorithm from each type*).

At the time of the start of the competition for the standardization process, 82 algorithms were presented. Based on the minimum eligibility criteria defined by NIST, 69 algorithms were considered for the first round. Considering several parameters – safety, cost, productivity, implementation characteristics, etc., 43 and 11 algorithms were excluded at the end of the first and second rounds, respectively, and the remaining 15 algorithms were saved for the third round [1].

The algorithms that remained in the second round can be categorized into five different cryptographic hard problems: isogeny-based *(1 algorithm*), lattice-based (*12 algorithms*), code-based (*7 algorithms*), multivariate polynomial cryptography (*4 algorithms*) and hash-based digital signatures (*2 algorithms*) [1-2]. Security is the main evaluation criterion that determines competition in the NIST competition, and it is clear that the implementation of the candidate software is mainly focused on it. However, it is essential that the algorithm has an efficient hardware implementation. And timely identification of hardware inefficiency will help concentrate the efforts of the crypto-

graphic community on more promising candidates, potentially saving a large amount of time that can be spent on cryptanalysis [3].

2 Hardware and software

Cryptographic algorithms are routinely implemented using both software and hardware. By software, we mean implementations that can be executed using processors. These processors may vary from low-cost low-power embedded processors, such as ARM Cortex-M4, to high-performance general-purpose microprocessors, such as Intel Core i7, with Haswell microarchitecture, supporting Advanced Vector Extensions 2 (AVX2) and the AES New Instructions (AES-NI). The common feature is that all of these processors are typically programmed using high-level programming languages, such as C. Code written in these languages is portable among different processor types. Software implementations can be further optimized by using assembly language programming, involving instructions specific to a given processor (or more accurately to its Instruction Set Architecture (ISA)). Assembly language programs are not easily portable among processors based on different ISAs.

By hardware, we mean implementations that can be executed using Field Programmable Gate Arrays (FPGAs), Application-Specific Integrated Circuits (ASICs), Programmable Logic (PL) of System on Chip FPGAs (SoC FPGAs), Application-Specific Standard Products (ASSPs), etc. The common feature is that most of these implementations are developed using hardware description languages (HDLs), such as VHDL and Verilog. These languages differ substantially from high-level programming languages by introducing the concepts of an entity, connectivity, concurrency, and timing. HDL source code is transformed by a synthesis tool to a netlist composed of basic logic components and connections among these components. Because of its generic nature, HDL code can be easily ported among different technologies, such as FPGAs and ASICs. ASIC implementations are faster, use less power, and require less physical area. FPGA implementations have the advantage of less expensive development tools, much shorter design cycle, and reconfigurability, understood as an ability to change the function of all internal building blocks and connections among them, even after a given integrated circuit has been deployed in actual products.

3 FPGA

Although software implementations are likely to be dominant during the first phase of deploying PQC standards in real applications, hardware implementations will inevitably follow. They are likely to start from hardware accelerators for constrained environments, such as smart cards and Internet of Things devices. Low-cost low-power processors used in such applications may not be able to keep up with the increased demands for computational power and energy usage. Thus, these processors may need to be extended with hardware accelerators. In the medium term, high-performance security processors enhanced with new PQC standards will emerge. These processors will be optimized to process in hardware all the algorithms associated with secure communication (*such as those used in the post-quantum versions of TLS, IPSec, IKE, and WTLS/WAP protocols*) and secure storage. Finally, in the longer-term, support for new instructions, enabling the efficient and side-channel resistant implementations of PQC standards, is likely to be added to the most popular processor ISAs. Co-processors for such instructions are, effectively, hardware implementations of PQC. Taking into account that the new PQC standards are likely to remain in use for decades, all of the mentioned above use cases should be given considerable weight. In particular, the performance of a given algorithm in hardware may affect its long-term performance in software, on processors equipped with new specialized instructions. Even if Round 2 hardware implementations are not a final word in terms of the algorithm performance, they provide the first glimpse into each candidate's suitability for hardware acceleration. They establish [1] an open source-code base on which more optimized implementation and implementations protected against side-channel and fault attacks can be built in Round 3 and beyond.

Assuming the use of the same technology, hardware implementations outperform software implementations using at least one, and typically multiple metrics, such as speed, power consump-

tion, energy usage, and security against physical attacks. They also allow much higher flexibility in trading one subset of these metrics for another. From the point of view of benchmarking and ranking of candidates, such flexibility may become a curse, especially taking into account that no two metrics are likely to have a simple linear dependence on each other. A practical solution to this problem [4] is to focus during the evaluation process on two major types of implementations: high-speed and lightweight.

In high-speed implementations, the primary target is speed. For PQC schemes, this target amounts to minimizing the execution times of major operations involving the public and private key, respectively. For Key Encapsulation Mechanisms (KEMs), these operations are encapsulation and decapsulation; for digital signature schemes, signature verification and generation; for public-key encryption (PKE), encryption and decryption. The time of key generation may also play a major role in the case when a public-private key pair cannot be reused for security reasons. The resource utilization is secondary. Still, hardware designers typically aim at achieving the Pareto optimality, in which any further improvement in speed comes at the disproportionate cost in terms of resource utilization. The primary advantage of high-speed implementations is that they reveal the inherent potential of a given algorithm for parallelization. As long as the resource-utilization limit is sufficiently high, this limit does not affect the ranking of algorithms. As a result, the ranking is strongly correlated with the features of algorithms themselves and is not substantially influenced by any additional assumptions and technology choices. Additionally [4], only high-speed hardware implementations may effectively compete with optimized software implementations targeting high-performance processors with vector instructions (*e.g., AVX2*).

In lightweight implementations, the primary targets are typically minimum resource utilization and minimum power consumption, under the assumption that the execution time does not exceed a predefined maximum. Another way of formulating the goal is to achieve minimum execution time, assuming a given maximum budget in terms of resource utilization, power consumption, or energy usage. The maximum budget on resource utilization is related to the cost of implementation; the budget on power assures correct operation without overheating or devoting additional resources to cooling. The maximum energy usage affects how long a battery-operated device can function before the next battery recharge. In the context of the standardization process for cryptographic algorithms, the mentioned above maximum budgets are very hard to select. Any change in these thresholds may favor a different subset of candidates. With new standards remaining in use for decades, timing, cost, and power requirements of new applications are challenging to predict.

Additionally, changes in technology significantly affect which hardware architectures meet particular constraints. For example, an architecture capable of accomplishing the execution time of 0.1 seconds (*or below*), under a certain power or energy budget, may substantially change with the improvements in technology. As a result, the majority of current limits are selected somewhat arbitrarily by different designers, or left undefined in their reports. Consequently, the ranking of PQC candidates based on their lightweight implementations, especially those developed by different groups, is extremely challenging and assumption-dependent. These rankings have little to do with the parallelization allowed by each algorithm, as most of the operations must be executed sequentially due to the small resource budget. The primary feature of algorithms these implementations reveal is the number and complexity of its distinct elementary operations. Each major operation infers an additional functional unit, increasing resource utilization and power consumption. Additionally [4], lightweight hardware implementations can outperform software implementations targeting low-cost and low-power embedded processors (*for instance, such as Cortex-M4*).

In the case of FPGA implementations, resource utilization is a vector, such as (*#LUTs, #flip-flops, #DSP units, #BRAMs*). No single element of this vector can be expressed in terms of other elements. As a result, imposing a resource limit implies specifying the values of all components of this resource vector. One possible approach may be to choose the resources of the smallest FPGA of a given low-cost FPGA family. However, FPGA families and their resources change over time, so this limit has only a physical meaning during the limited time, covering the evaluation period, and may lose its significance just a few years after the standard is published and deployed. Besides, the

FPGA device may need to offset some of the costs associated with countering side-channel attacks (*moreover, this overhead and even countermeasures may remain unknown at the time of the candidates' evaluation*) [4].

4 FPGA Xilinx family

One of the major concerns is the NIST recommendation to focus on hardware benchmarking using the Xilinx Artix-7 FPGA family. This recommendation appeared in several NIST presentations related to Round 2 of the NIST standardization process, e.g., during PQCrypto 2019 in May 2019 and the Second PQC Standardization Conference in August 2019. We believe that, in its current form, this recommendation is counterproductive, and it impedes rather than supports fair and comprehensive hardware and software/hardware benchmarking [1].

FPGA family is a set of FPGA devices sharing the same internal structure and the same process technology (*also known as technology node or process node*), described by a number related to the size and density of transistors that can be fabricated using a given manufacturing process. With the steady improvements in process technology, described by Moore's Law, the maximum capacity and speed of FPGA devices have been steadily increasing while their prices have remained approximately the same. Every new generation of FPGA devices of a particular vendor receives a unique name, referred to as a family name. Every family consists of multiple devices with various distinct sizes to match the needs of different applications. All devices of a particular family share the same internal architecture and process technology but differ in terms of the number of resources of a particular type, such as Look-Up Tables (LUTs), flip-flops (FFs), block memories, and digital signal processing units (DSP units) or multipliers. Most vendors release both low-cost families (*such as Xilinx Artix-7) and high-performance families (such as Xilinx Virtex-7*). Most of them also release mid-range families, such as Xilinx Kintex-7. The maximum amount of resources available in the largest device of a low-cost family is naturally significantly smaller than the equivalent amount in the largest device of a high-performance family (*e.g., over 5 times smaller for Artix-7 vs. Virtex-7*).

Additionally, in recent years, FPGA vendors started releasing new types of programmable devices that enhance Programmable Logic of traditional FPGAs with the Processing System based on a hardwired embedded processor, such as ARM. Since this processor is custom designed, it takes full advantage of a given technological process and operates at a clock frequency significantly higher than Programmable Logic. With a fast processor and an efficient interface between this processor and Programmable Logic, these devices are ideal for software/hardware co-designs targeting high-speed. Although these types of devices appear under multiple commercial names, they are often collectively referred to as System on Chip FPGAs (SoC FPGAs). The first family of this type was Xilinx Zynq-7000, released in 2011, based on ARM Cortex-A9 embedded processors [4].

Hardware designs are described in hardware description languages. HDL code is typically identical for all FPGA families. As opposed to software, where each processor may require different optimized assembly language code, no such concepts exist for hardware. Thus, it is possible to synthesize the same HDL code targeting various FPGA families from various vendors, as long as the maximum capacity of the largest device of a given family is not exceeded.

5 General features of the FPGA Xilinx family

Today, the most modern is the series 7 FPGA Xilinx – Artix-7, Kintex-7, Virtex-7. In this series, the FPGA family with the ARM Cortex-A9 - Zynq-7000 processor core has been announced. In the new series, only Virtex-7 continues the existing line of high-performance FPGA, and the other two families – Artix and Kintex – replaced the Spartan line. FPGA Artix are designed for mass products, and are characterized by low power consumption and low cost, and Kintex is, to some extent, Spartan, specialized for digital signal processing. Until now, the Virtex series has traditionally been used in applications built around high-speed serial receivers and in projects based on digital signal processing. The Kintex-7 family «fits» well into a niche where a large number of parallel DSC units are required at a moderate price [5], and more expensive Virtex-7 are intended for systems with a large number of hardware receivers (see Table 1).

Zynq-7010 and Zynq-7020 chips are based on playable resources of the Artix family, and Zynq-7030 and Zynq-7040 are based on Kintex. This affects in the peak performance [6] of the digital signal processing subsystem (*the frequency of the lower FPGA Zynq is lower, they do not have PCI Express units and high-speed receivers*) (see Table 2).

A key feature of the next generation of FPGA is the unification of playable resources. It is assumed that for the next generation of FPGA, a quick migration between Virtex/Kintex/Artix families will be possible without adjusting the project.

Table 1 – FPGA Xilinx family, 7th series

Maximum parameters	**Artix-7**	**Kintex-7**	**Virtex-7**
Logical cells, thousand	*352*	*407*	*1955*
Block memory	*12*	*29*	*65*
DSP sections	*700*	*1540*	*3960*
Peak digital signal processing performance*, *GMAC/s*	*504*	*1965*	*5053*
Receivers	*4*	*16*	*88*
Maximum transfer rate, *Gb/s*	*3,75*	*10.325*	*28,05*
Peak capacity of receivers, *Gb/s*	*30*	*330*	*2784*
PCI Express interfaces	*Gen1x4*	*Gen2x8*	*Gen3x8*
Memory interface exchange speed, *MB/s*	*800*	*2133*	*2133*
External outputs	*450*	*500*	*1200*

** – for symmetric coefficient filters.*

Table 2 – FPGA Zynq-7000 family

Parameters	**Z -7010**	**Z -7020**	**Z -7030**	**Z -7040**
Programmable logic cells (ASIC gates)	*28 K (430 K)*	*85 K (1,3 M)*	*125 K (1,9 M)*	*235 K (3,5 M)*
Memory blocks (36 KB)	*60*	*140*	*265*	*760*
DSP sections (18x25 MACC)	*80*	*220*	*400*	*760*
Peak DSP performance*, GMAC/s	*58*	*158*	*480*	*912*
PCI Express blocks	-	-	*Gen2 x4*	*Gen2 x8*
ADC	*2x12 bits, 1 M samples/sec, 17 dif. channels*			
Encryption	*AES and SHA 256-bit*			
I/O units, 3.3 V	*100*	*195*	*100*	*200*
I/O units, 1.8 V	-	-	*150*	*150*
High speed receivers	-	-	*4*	*12*

** – for KIX with symmetric coefficients.*

6 FPGA Xilinx family details

Tables 3-6 show the characteristics of the FPGA family Xilinx, namely Spartan-7, Artix-7, Virtex-7, Kintex-7, respectively [6].

Giving preference to the Xilinx Artix-7 family has several undesired consequences summarized below:

1. Artix-7 is a low-cost FPGA family. As such, it is not very suitable for high-speed implementations. Hardware resources of even the largest device of this family are often insufficient to demonstrate the full potential for parallelizing operations a given PQC algorithm. Thus, the use of Artix-7 makes perfect sense for benchmarking lightweight implementations but may lead to suboptimal results for high-speed implementations.

Table 3 – Characteristic of Spartan-7

Parameters		I/O optimization at the lowest cost and highest performance-per-watt (1.0 V, 0.95 V)					
Part number		***XC7S6***	***XC7S15***	***XC7S25***	***XC7S50***	***XC7S75***	***XC7S100***
Logic resources	Logic cells	*6,000*	*12,800*	*23,360*	*52,160*	*76,800*	*102,400*
	Slices	*938*	*2,000*	*3,650*	*8,150*	*12,000*	*16,000*
	CLB flip-flops	*7,500*	*16,000*	*29,200*	*65,200*	*96,000*	*128,000*
Memory resources	Max. distributed RAM, (Kb)	*70*	*150*	*313*	*600*	*832*	*1,100*
	Block RAM/FIFO w/ ECC (36 Kb each)	*5*	*10*	*45*	*75*	*90*	*120*
	Total Block RAM (Kb)	*180*	*360*	*1,620*	*2,700*	*3,240*	*4,320*
Clock resources	Clock Mgmt Tiles (1 MMCM + 1 PLL)	*2*	*2*	*3*	*5*	*8*	*8*
I/O resources	Max. single-ended I/O pins	*100*	*100*	*150*	*250*	*400*	*400*
	Max. differential I/O pairs	*48*	*48*	*72*	*120*	*192*	*192*
Embedded hard IP resources	DSP slices	*10*	*20*	*80*	*120*	*140*	*160*
	Analog mixed signal (AMS) / XADC	*0*	*0*	*1*	*1*	*1*	*1*
	Configuration AES / HMAC blocks	*0*	*0*	*1*	*1*	*1*	*1*
Speed grades	Commercial temp, (C)	*-1 -2*	*-1 -2*	*-1 -2*	*-1 -2*	*-1 -2*	*-1 -2*
	Industrial temp (I)	*-1 -2 -1L*	*-1 -2 -1L*	*-1 -2 -1L*	*-1 -2 -1L*	*-1 -2 -1L*	*-1 -2 -1L*
	Expanded temp (Q)	*-1*	*-1*	*-1*	*-1*	*-1*	*-1*

Package	Body area (mm)	Ball pitch (mm)	Available user I/O: 3.3 V SelectIO™ HR I/O					
CPGA196	*8x8*	*0.5*	*100*	*100*				
CSGA225	*13x13*	*0.8*	*100*	*100*	*150*			
CSGA324	*15x15*	*0.8*			*150*	*210*		
FTGB196	*15x15*	*1.0*	*100*	*100*	*100*	*100*		
FGGA484	*23x23*	*1.0*				*250*	*338*	*338*
FGGA676	*27x27*	*1.0*					*400*	*400*

2. Artix-7 is a traditional FPGA, and not a SoC FPGA. As a result, the only way to develop a single-chip software/hardware implementation using Artix-7 is the use of so-called "soft" processor cores, i.e., processors implemented using programmable logic. Soft processors compatible with Artix-7 include MicroBlaze and lightweight versions of RISC-V. All of them operate at much lower clock frequency than hardwired embedded processors of SoC FPGAs.

3. Artix-7 is unsuitable for HLS designs. Such designs typically take significantly more resources than designs based on writing code manually in HDL. As a result, assuming the Pareto optimization for high-speed, they are unlikely to fit in the largest Artix-7 FPGA.

4. Artix-7 is a relatively old FPGA family, released by Xilinx in 2010. By the time of the release of the PQC standard, this family will be at least 12 years old. While still relatively popular for low-cost applications, this family does not represent the state-of-the-art in FPGA technology.

5. It is not customary to base ranking of candidates in cryptographic contests on results obtained for a single family of a single vendor. Although Xilinx is the largest developer of FPGAs and SoC FPGAs, Intel comes a strong second, and other vendors, such as Microchip and Lattice Semiconductor, also develop FPGAs suitable for implementing cryptographic algorithms. During the SHA-3 competition, the results were reported for seven FPGA families from two major vendors, Xilinx and Altera. During the CAESAR contest, four Xilinx families and four Altera families were employed. For all of these families, results were generated based on the same HDL code. There was no need to purchase multiple tools or boards. Free or trial versions of tools were sufficient. The designs ended with the generation of post-place-and-route reports, which correctly described the worst-case performance of any particular instance of the given FPGA device.

Table 4 – Characteristic of Artix-7

				Transceiver optimization at the lowest cost and highest DSP bandwidth (1.0 V, 0.95 V, 0.9 V)							
Part number				***XC7A12T***	***XC7A15T***	***XC7A25T***	***XC7A35T***	***XC7A50T***	***XC7A75T***	***XC7A100T***	***XC7A200T***
Logic resources	Logic cells			12800	16,640	23,360	33,280	52,160	75,520	101,440	215,360
	Slices			2,000	2,600	3,650	5,200	8,150	11,800	15,850	33,650
	CLB flip-flops			16,000	2,800	29,200	41,600	65,200	94,400	126,800	269,200
Memory resources	Maximum distributed RAM (Kb)			171	200	313	400	600	892	1,188	2,888
	Block RAM/FIFO w/ ECC (36 Kb each)			20	25	45	50	75	105	135	365
	Total block RAM (Kb)			720	900	1,620	1,800	2,700	3,780	4,860	13,140
Clock resources	CMTs (1 MMCM + 1 PLL)			3	5	3	5	5	6	6	10
I/O resources	Maximum single-ended I/O			150	250	150	250	250	300	300	500
	Maximum differential I/O pairs			72	120	12	120	120	144	144	240
Embed-ded hard IP resources	DSP slices			40	45	80	90	120	180	240	740
	PCIe® Gen2			1	1	1	1	1	1	1	1
	Analog mixed signal (AMS) / XADC			1	1	1	1	1	1	1	1
	Configuration AES / HMAC blocks			1	1	1	1	1	1	1	1
	GTP Transceivers (6.6 Gb/s max rate)			2	4	4	4	4	8	8	16
Speed grades	Commercial temp (C)			-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2
	Extended temp (E)			-2L -3	-2L -3	-2L -3	-2L -3	-2L -3	-2L -3	-2L -3	-2L -3
	Industrial temp (I)			-1 -2 -1L	-1 -2 -1L	-1 -2 -1L	-1 -2 -1L	-1 -2 -1L	-1 -2 -1L	-1 -2 -1L	-1 -2 -1L
	Package	Dimen-sions (mm)	Ball pitch (mm)	Available user I/O: 3.3 V SelectIO™ HR I/O (*GTP transceivers*)							
Footprint compati-ble	**CPG236**	10x10	0.5		106(2)		106(2)	106(2)			
	CPG238	10x10	0.5	112(2)		112(2)					
	CSG324	15x15	0.8		210(0)		210(0)	210(0)	210(0)	210(0)	
	CSG325	15x15	0.8	150(2)	150(4)	150(4)	150(4)	150(4)			
	FTG256	17x17	1.0		170(0)		170(0)	170(0)	170(0)	170(0)	
	SBG484	19x19	0.8								285(4)
	FGG484	23x23	1.0		250(4)		250(4)	250(4)	285(4)	285(4)	
	FBG484	23x23	1.0								285(4)
	FGG676	27x27	1.0						300(8)	300(8)	
	FBG676	27x27	1.0								400(8)
	FFG1156	35x35	1.0								500(16)

Table 5 – Characteristic of Kintex-7

				Optimized for best price-performance (1.0 V, 0.95 V, 0.9 V)						
Part number				***XC7K70T***	***XC7K160T***	***XC7K325T***	***XC7K355T***	***XC7K410T***	***XC7K420T***	***XC7K480T***
Logic resources	Slices			10,250	25,350	50,950	55,650	63,550	63,150	74,650
	Logic cells			65,600	162,240	326,080	356,160	406,720	416,960	477,760
	CLB flip-flops			82,000	202,800	407,600	445,200	508,400	521,200	597,200
Memory resources	Maximum distributed RAM, (Kb)			838	2,188	4,000	5,088	5,663	5,938	6,778
	Block RAM/FIFO w/ ECC (36 Kb each)			135	325	445	715	795	835	955
	Total block RAM, (Kb)			4,860	11,700	16,020	25,740	28,620	30,060	34,380
Clock resources	CMTs (1 MMCM + 1 PLL)			6	8	10	6	10	8	8
I/O resources	Maximum single-ended I/O			300	400	500	300	500	400	400
	Maximum differential I/O pairs			144	192	240	144	240	192	192
Integrated IP resources	DSP48 slices			240	600	840	1,440	1,540	1,680	1,920
	PCIe® Gen2			1	1	1	1	1	1	1
	Analog mixed signal (AMS) / XADC			1	1	1	1	1	1	1
	Configuration AES / HMAC blocks			1	1	1	1	1	1	1
	GTX transceivers (12.5 Gb/s max rate)			8	8	16	24	16	32	32
Speed grades	Commercial Temp, (C)			-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2
	Extended temp (E)			-2L -3	-2L -3	-2L -3	-2L -3	-2L -3	-2L -3	-2L -3
	Industrial temp (I)			-1 -2 -2L	-1 -2 -2L	-1 -2 -2L	-1 -2 -2L	-1 -2 -2L	-1 -2 -2L	-1 -2 -2L
	Package	Dimensions (mm)	Ball pitch (mm)	Available user I/O: 3.3 V HR I/O, 1.8 V HP I/Os (*GTX*)						
Footprint compatible	**FBG484**	23x23	1.0	185, 100 (4)	185, 100 (4)					
	FBG676	27x27	1.0	200, 100 (8)	250, 150 (8)	250, 150 (8)		250, 150 (8)		
	FFG676	27x27	1.0		250, 150 (8)	250, 150 (8)		250, 150 (8)		
	FBG900	31x31	1.0			350, 150 (16)		350, 150 (16)		
	FFG900	31x31	1.0			350, 150 (16)		350, 150 (16)		
	FFG901	31x31	1.0				300, 0 (24)		380, 0 (28)	380, 0 (28)
	FFG1156	35x35	1.0						400, 0 (32)	400, 0 (32)

6. Based on the authors' experiences, multiple reviewers of papers devoted to implementations of Round 2 PQC candidates treated the NIST's choice of Artix-7 as an absolute requirement. Submissions not complying with this requirement were subject to rejection or requests for major revisions. As a result, a noble goal of making the results more comparable with one another was turned into a reason for suppressing or delaying the publication of relevant results.

Table 6 – Characteristic of Virtex-7

		Optimized for highest system performance and capacity (1.0 V)										
Part number		***XC7V585T***	***XC7V2000T***	***XC7VX330T***	***XC7VX415T***	***XC7VX485T***	***XC7VX550T***	***XC7VX690T***	***XC7VX980T***	***XC7VX1140T***	***XC7VH580T***	***XC7VH870T***
Logic Resources	Slices	91,050	305,400	51,000	64,400	75,900	86,600	108,300	153,000	178,000	90,700	136,900
	Logic cells	582,720	1,954,560	326,400	412,160	485,760	554,240	693,120	979,200	1,139,200	580,480	876,160
	CLB flip-flops	728,400	2,443,200	408,000	515,200	607,200	692,800	866,400	1,224,000	1,424,000	725,600	1,095,200
Memory Resources	Maximum distributed RAM (Kb)	6,938	21,550	4,388	6,525	8,175	8,725	10,888	13,838	17,700	8,850	13,275
	Block RAM/FIFO w/ ECC (36 Kb each)	795	1,292	750	880	1,030	1,180	1,470	1,500	1,880	940	1,410
	Total block RAM (Kb)	28,620	46,512	27,000	31,680	37,080	42,480	52,920	54,000	67,680	33,840	50,760
Clocking	CMTs (1 MMCM + 1 PLL)	18	24	14	12	14	20	20	18	24	12	18
I/O Resources	Maximum single-ended I/O	850	1,200	700	600	700	600	1,000	900	1,100	600	300
	Maximum differential I/O pairs	408	576	336	288	336	288	480	432	528	288	144
Integrated IP Resources	DSP slices	1,260	2,160	1,120	2,160	2,800	2,880	3,600	3,600	3,360	1,680	2,520
	PCIe® Gen2	3	4	-	-	4	-	-	-	-	-	-
	PCIe Gen3	-	-	2	2	-	2	3	3	4	2	3
	Analog mixed signal (AMS) / XADC	1	1	1	1	1	1	1	1	1	1	1
	Configuration AES / HMAC blocks	1	1	1	1	1	1	1	1	1	1	1
	GTX transceivers (12.5 Gb/s max rate)	36	36	-	-	56	-	-	-	-	-	-
	GTH transceivers (13.1 Gb/s max rate)	-	-	28	48	-	80	80	72	96	48	72
	GTH transceivers (28.05 Gb/s max rate)	-	-	-	-	-	-	-	-	-	8	16
Speed Grades	Commercial temp, (C)	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2
	Extended temp (E)	-2L -3	-2L -2G	-2L -3	-2L -3	-2L -3	-2L -3	-2L -3	-2L	-2L -2G	-2L -2G	-2L -2G
	Industrial temp (I)	-1 -2	-1	-1 -2	-1 -2	-1 -2	-1 -2	-1 -2	-1	-1	-	-

Continuation of Table 6

			Optimized for highest system performance and capacity (1.0 V)										
Part number			***XC7V585T***	***XC7V2000T***	***XC7VX330T***	***XC7VX415T***	***XC7VX485T***	***XC7VX550T***	***XC7VX690T***	***XC7VX980T***	***XC7VX1140T***	***XC7VH580T***	***XC7VH870T***
Package	Dimen-sions (mm)	Ball Pitch (mm)	Available user I/O: 3.3 V HR I/O, 1.8 V HP I/Os (*GTX*, *GTH*)									1.8 V HP I/O (*GTH*, *GTZ*)	
FFG1157	35x35	1.0	0, 600 (20, 0)		0, 600 (20, 0)	0, 600 (20, 0)	0, 600 (20, 0)		0, 600 (20, 0)				
FFG1761	42.5x42.5	1.0	100, 750 (36, 0)		50, 650 (0, 28)		0, 700 (28, 0)		0, 850 (0, 36)				
FHG1761	45x45	1.0		0, 850 (36, 0)									
FLG1925	35x35	1.0		0, 1200 (16, 0)									
FFG1158	45x45	1.0				0, 350 (0, 48)	0, 350 (0, 48)	0, 350 (0, 48)	0, 350 (0, 48)				
FFG1926		1.0							0, 720 (0, 64)	0, 720 (0, 64)			
FLG1926		1.0									0, 720 (0, 64)		
FFG1927		1.0				0, 600 (0, 48)	0, 600 (56, 0)	0, 600 (0, 80)	0, 600 (0, 80)				
FFG1928		1.0								0, 480 (0, 72)			
FLG1928		1.0									0, 480 (0, 96)		
FFG1930		1.0					0, 700 (24, 0)		0, 1000 (0, 24)	0, 900 (0, 24)			
FLG1930		1.0									0, 1100 (0, 24)		
FLG1155	35x35	1.0										400 (24, 8)	
FLG1931	45x45	1.0										600 (48, 8)	
FLG1932		1.0											300 (72, 16)

Taking these concerns into account, our recommendation for Round 3 is to encourage reporting results for at least the following FPGA families:

1. For lightweight hardware implementations and lightweight software/hardware implementations based on soft processor cores: Xilinx Artix-7 (f*or compatibility with Round 2 results*) and Intel Cyclone 10 LP.

2. For lightweight software/hardware implementations based on the use of hard processor cores: Xilinx Zynq 7000-series and Intel Cyclone V SoC FPGAs.

3. For high-speed hardware and high-speed software/hardware implementations: Zynq Xilinx UltraScale+ and Intel Stratix 10 SoC.

One of the reasons for selecting Zynq Xilinx UltraScale+, even for pure hardware implementations that do not require SoC capabilities, is the support for these devices by the free version of the Xilinx toolset, called Vivado HL WebPACK, which is sufficient to generate all required benchmarking results. Xilinx Virtex-7 UltraScale+ FPGAs, which could be considered as a natural candidate, are not supported by the same free version of tools. The Zynq Xilinx UltraScale+ family is also recommended for high-speed software/hardware implementations based on the use of hard processor cores because of moderate cost of suitable prototyping boards and the availability of a free Benchmarking Setup for Software/Hardware Implementations of PQC Schemes, developed at George Mason University [7].

7 FPGA-focused implementations

In Tables 7-8, summarize implementations targeting Xilinx Artix-7 FPGAs and related Xilinx Zynq-7000 SoC FPGAs (*indicated with the superscript* ^{Z}). For the security level 1, six candidates – Classic McEliece, CRYSTALS-Kyber, FrodoKEM, NewHope, SIKE, and Saber – have implementations of all three operations reported. The preliminary implementations of BIKE focused on key generation only. For security level 3, NewHope does not have a variant. For security level 5, the results are missing for Classic McEliece.

For most KEMs, the time of decapsulation is longer than the time of encapsulation. Table entries are ordered according to the time of decapsulation in μs (*and, if needed, according to the decapsulation time in clock cycles*).

The ranking of candidates listed in Tables 7 and 8 is very challenging to determine based on available results. First, it may be unfair to compare pure hardware implementations with software/hardware implementations. Secondly, it is hard to compare lightweight implementations with high-speed implementations, as they are optimized with different primary metrics in mind. Third, software/hardware implementations based on different processors are very challenging to compare with one another. Finally, even for implementations using exactly the same type of implementation (*software/hardware*) and the same type of processor (RISC-V), the comparison may be unintentionally biased. Significantly different hardware support was provided for algorithms that can take advantage of the Number Theoretic Transform – Kyber and NewHope – vs. the algorithm that cannot – Saber. An additional, relatively minor factor is that several results for Classic McEliece and NewHope concern their IND-CPA-secure PKEs rather than IND-CCA-secure KEMs.

Taking all these factors into account, almost the only ranking that is quite clear from Tables 7 and 8 is the ranking of candidates that have results available for pure hardware implementations targeting high-speed. In this specific category, the ranking for the security level 1 is: 1. NewHope, 2. Classic McEliece, 3. FrodoKEM. If we assume that a software/hardware implementation of SIKE with a custom processor is almost as efficient as a pure hardware implementation, then we can also add SIKE at position 4. At level 3, NewHope does not have a variant, and at level 5, Classic McEliece and FrodoKEM, do not have high-speed pure hardware implementations reported.

In Tables 9 and 10, summarize implementations targeting Xilinx Virtex-7 FPGAs. Unfortunately, the only conclusion that can be drawn from these tables is an advantage of Classic McEliece over SIKE in terms of all performance metrics other than the number of LUTs and flip-flops.

In Table 11, we compare results reported by our own group at the end of 2019 in [4, 8-9], with results reported by other groups for Saber and NewHope, respectively. All results were obtained using the same SoC FPGA, Zynq UltraScale+. The software/hardware implementation of Round5 was very close to the pure hardware implementation. The same was not the case for the software/hardware implementation of Saber, were a significant percentage of the execution time was devoted to functions remaining in software and to the transfer of data and control between software and hardware. As a result, the most accurate comparison between Round5 and Saber is possible at the security level 3, for which the pure hardware implementation of Saber was reported in [3, 5]. Based on this implementation Saber outperforms Round5 by a small margin in terms of the execution times for encapsulation and decapsulation.

Table 7 – Level 1 KEMs and PKEs on Artix-7 (*default*) & Zynq-7000

Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation		Encaps. / Enc.^{cpa}		Decaps. / (Dec.+Enc.)^{cpa}	
									cycles	μs	cycles	μs	cycles	μs
Security Level 1														
NewHope-512^{cpa}	HW	HS	200	6,780	4,026	–	2	7.0	4,200	21.0	6,600	33.0	9,100	45.5
mceliece348864^{cpa}	HW	HS	106	81,339	132,190	–	0	236.0	202,787	1,920.3	2,720	25.8	12,743	120.7
mceliece348864^{cpa}	HW	HS	108	25,327	49,383	–	0	168.0	1,599,882	14,800.0	2,720	25.2	18,358	169.8
Kyber-512	SW/HW^{RV}	LW	–	23,925	10,844	–	21	32.0	150,106	–	193,076	–	204,843	–
FrodoKEM-640	HW	HS	172	2,587	2,994	855	16	0	–	–	–	–	–	–
			171	5,796	4,694	1,692	16	0	204,766	1,190.5	207,269	1,212.1	209,867	1,408.5
16x			149	6,881	5,081	1,947	16	12.5	–	–	–	–	–	–
Kyber-512	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	74,519	2,980.8	131,698	5,267.9	142,309	5,692.4
NewHope-512	SW/HW^{RV}	LW	–	23,925	10,844	–	21	32.0	123,860	–	207,299	–	226,742	–
NewHope-512	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	97,969	3,918.8	236,812	9,472.5	258,872	10,354.9
LightSaber	SW/HW^{RV}	LW	–	23,925	10,844	–	21	32.0	366, 837	–	526, 496	–	657,583	–
Kyber-512	SW/HW^{RV}	LW	59	1,842	1,634	–	5	34.0	710,000	11,993.2	971,000	16,402.0	870,000	14,695.9
NewHope-512	SW/HW^{RV}	LW	59	1,842	1,634	–	5	34.0	904,000	15,270.3	1,424,000	24,054.1	1,302,000	21,993.2
SIKEp434	SW/HW^{c}	HS	162	22,595	11,558	7,491	162	37.0	1,474,200	9100	2,494,800	15,400.0	2,656,800	16,400.0
SIKEp503	SW/HW^{c}	HS	162	22,595	11,558	7,491	162	37.0	1,733,400	10,700.0	2,932,200	18,100.0	3,126,600	19,300.0
FrodoKEM-640	HW	LW	191	971	433	290	1	0						
			190	4,246	2,131	1,180	1	0	3,237,288	16,949.2	3,275,862	17,241.4	3,306,122	20,408.2
1x			162	4,446	2,152	1,254	1	12.5	–	–	–	–	–	–
	SW/HW^{c}								–	–	–	–	–	–
SIKEp434		LW	143	10,976	7,115	3,512	57	21.0	2,187,902	15,300.0	3,718,004	26,000.0	3,946,804	27,600.0
SIKEp503	SW/HW^{c}	LW	143	10,976	7,115	3,512	57	21.0	2,602,603	18,200.0	4,390,104	30,700.0	4,676,105	32,700.0
FrodoKEM-640	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	11,453,942	458,157.7	11,609,668	464,386.7	12,035,513	481,420.5
BIKE-1 Level 1^{cs}	HW	HS	165	1,907	1,049	608	0	7.0	95,500	578.0	–	–	–	–
BIKE-3 Level 1^{cs}	HW	HS	170	1,397	925	453	0	4.0	98,500	579.0	–	–	–	–
BIKE-2 Level 1^{cs}	HW	HS	160	3,874	2,141	1,312	0	10.0	2,150,000	13,437.0	–	–	–	–
BIKE Level 1	HW	HS	135	1,865	589	590	0	4.0	7,370,429	54,540.0	–	–	–	–

Table 8 – Level 3 & 5 KEMs and PKEs on Artix-7 (*default*) & Zynq-7000

Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation		Encaps. / Enc.^{cpa}		Decaps. / (Dec.+Enc.)^{cpa}	
									cycles	μs	cycles	μs	cycles	μs
Security Level 3														
mceliece460896^{cpa}	HW	HS	107	38,669	74,858	–	0	303.0	5,002,044	46,704.4	3,360	31.4	31,005	289.5
FrodoKEM-976	HW	HS	169	2,869	3,000	908	16	0	–	–	–	–	–	–
16x			168	6,188	4,678	1782	16	0	476,056	2,816.9	479,993	2,857.1	483,073	3,076.9
			157	7,213	5,087	2042	16	19.0	–	–	–	–	–	–
Saber^{Z}	SW/HW^{A9}	HS							–	–	–	–	–	–
			125	7,400	7,331	–	28	2.0	–	3,273.0	–	4,147.0	–	3,844.0
Kyber-768	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	111,525	4,461.0	177,540	7,101.6	190,579	7,623.2
SIKEp610	SW/HW^{c}	HS	162	22,595	11,558	7,491	162	37.0	2,916,000	18,000.0	5,443,200	33,600.0	5,508,000	34,000.0
FrodoKEM-976	HW	LW	189	1,243	441	362	1	0	–	–	–	–	–	–
1x			187	4,650	2,118	1,272	1	0	7,560,000	40,000.0	7,480,000	40,000.0	7,714,286	47,619.0
			162	4,888	2,153	1,390	1	19.0	–	–	–	–	–	–
SIKEp610	SW/HW^{c}								–	–	–	–	–	–
		LW	143	10,976	7,115	3,512	57	21.0	4,347,204	30,400.0	8,108,108	56,700.0	8,208,208	57,400.0
FrodoKEM-976	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	26,005,326	1,040,213.0	29,749,417	1,189,976.7	30,421,175	1,216,847.0
BIKE Level 3	HW	HS	135	1,884	557	593	0	5	30,447,947	231,400.0	–	–	–	–

Continuation of Table 8

Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation		Encaps. / Enc.^{cpa}		Decaps. / (Dec.+Enc.)^{cpa}	
									cycles	μs	cycles	μs	cycles	μs
Security Level 5														
NewHope-1024^{cpa}	HW	HS	200	6,781	4,127	–	2	8.0	8,000	40.0	12,500	62.5	17,300	86.5
NewHope-1024^{cpa}	HW	HS	190	13,244	8,272	–	24	18.0	–	–	34,000	178.0	30,600^{KD}	160.0^{KD}
Kyber-1024	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	148,547	5,941.9	223,469	8,938.8	240,977	9,639.1
NewHope-1024	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	97,969	3,918.8	236,812	9,472.5	258,872	10,354.9
Kyber-1024	SW/HW	LW	–	23,925	10,844	–	21	32.0	349,673	–	405,477	–	424,682	–
NewHope-1024	SW/HW	LW	–	23,925	10,844	–	21	32.0	235,420	–	392,734	–	450,541	–
NewHope-1024^{cpa}	SW/HW	HS	25	26,606	26,303	–	32	1.0	357,052	14,282.1	589,285	23,571.4	756,932	30,277.3
FireSaber	SW/HW	LW	–	23,925	10,844	–	21	32.0	1,300,272	–	1,622,818	–	1,898,051	–
Kyber-1024	SW/HW^{RV}	LW	59	1,842	1,634		5	34.0	2,203,000	37,212.8	2,619,000	44,239.9	2,429,000	41,030.4
SIKEp751	SW/HW^{c}	HS	162	22,595	11,558	7,491	162	37.0	3,742,200	23,100.0	6,188,400	38,200.0	6,658,200	41,100.0
NewHope-1024	SW/HW^{RV}	LW	59	1,842	1,634		5	34.0	1,776,000	30,000.0	2,742,000	46,317.6	2,528,000	42,702.7
SIKEp751	SW/HW^{c}	LW	143	10,976	7,115	3,512	57	21.0	7,965,108	55,700.0	13,156,013	92,000.0	14,185,614	99,200.0
FrodoKEM-1344	SW/HW^{RV}	LW	25*	14,975	2,539	4,173	11	14.0	67,994,170	2,719,766.8	71,501,358	2,860,054.3	72,526,695	2,901,067.8

At the same time, even the fastest reported implementation of Saber uses 1.6x fewer LUTs than Round5, with the same number of BRAMs and DSP units. FrodoKEM is demonstrated to be by far slower than Saber and Round5 for all security levels.

Somewhat differently, for the security level 5, the pure hardware implementation of NewHope, reported in [9], is not fast enough to outperform the software/hardware implementation of Round5 from [10]. However, the comparison is somewhat complicated by the fact that, in [9], the results are reported the IND-CPA-secure PKE (*rather than the IND-CCA-secure KEM*), and only the sum of the key generation and decryption (rather than the decryption itself) is reported in the paper.

In Tables 12, summarize results available for the implementations of digital signatures. The implementations targeting FPGAs are considered first in Table 9.

Table 9 – Level 1 KEMs on Virtex-7 (default) & Virtex-6 (*indicated with the superscript* ^{V6})

Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation		Encap./Enc.^{cpa}		Decaps./Dec.^{cpa}	
									cycles	μs	cycles	μs	cycles	μs
Security Level 1														
SIKEp503	HW	HS	171	25,094	26,971	9,514	264	34.0	640,000	3,738.3	1,120,000	6,542.1	1,210,000	7,067.8
SIKEp434	SW/HW	HS	142	21,210	13,657	7,408	162	38.0	981,180	6,900.0	1,677,960	11,800.0	1,777,500	12,500.0
SIKEp503	SW/HW	HS	142	21,210	13,657	7,408	162	38.0	1,166,040	8,200.0	1,976,580	13,900.0	2,104,560	14,800.0
LEDAkem-128^{o,cpa,V6}	HW	LW	235	104	53	33	0	1.0	–	–	712,000	3,029.8	2,620,000	18,714.3
SIKEp434	SW/HW	LW	152	10,937	7,132	3,415	57	21.0	2,191,781	14,400.0	3,713,851	24,400.0	3,957,382	26,000.0
SIKEp503	SW/HW	LW	152	10,937	7,132	3,415	57	21.0	2,602,740	17,100.0	4,383,562	28,800.0	4,672,755	30,700.0

^{cpa} - Design of a KEM variant resistant against Chosen-Plaintext Attack (CPA)
^{V6} - Design implemented on Virtex-6
^{o} - Design for an old parameter set changed by the submitters on March 19th, 2020

Unfortunately, multiple results available for qTESLA concern heuristic parameter sets that have been withdrawn by submitters on Aug. 20, 2019. Among the remaining designs, for Artix-7, the ranking of candidates for the security level 1 is 1. Picnic, 2. Dilithium, and 3. qTESLA. The differences among these candidates in terms of the execution time for the signature generation (*more critical*) and signature verification are very significant. At the same time, only the implementation of Picnic is a high-speed and pure hardware implementation. The remaining implementations are software/hardware implementations based on RISC-V. Additionally, the number of LUTs for Picnic is approximately 6 times larger than for Dilithium, and the number of BRAMs, 3.75 times larger. At the same time, compared to Picnic, the execution time for signature generation is 12 times longer for Dilithium-I and 16 times longer for Dilithium-II.

Table 10 – Level 3 & 5 KEMs & PKEs on Virtex-7

Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation		Encap./Enc.^{cpa}		Decaps./Dec.^{cpa}	
									cycles	μs	cycles	μs	cycles	μs
Security Level 3														
mceliece460896^{cpa}	HW	HS	131	109,484	168,939	–	0	446.0	515,806	3,943.5	3,360	25.7	17,931	137.1
SIKEp610	SW/HW	HS	142	21,210	13,657	7,408	162	38.0	1,962,360	13,800.0	3,654,540	25,700.0	3,711,420	26,100.0
SIKEp610	SW/HW	LW	152	10,937	7,132	3,415	57	21.0	4,353,120	28,600.0	8,097,412	53,200.0	8,219,178	54,000.0
Security Level 5														
mceliece6960119^{cpa}	HW	HS	130	116,928	188,324	–	0	607.0	974,306	7,500.4	5,413	41.7	25,135	193.5
mceliece6688128^{cpa}	HW	HS	137	122,624	186,194	–	0	589.0	1,046,139	7,658.4	5,024	36.8	29,754	217.8
mceliece8192128^{cpa}	HW	HS	130	123,361	190,707	–	0	589.0	1,286,179	9,901.3	6,528	50.3	32,765	252.2
mceliece6960119^{cpa}	HW	HS	141	44,154	88,963	–	0	563.0	11,179,636	79,570.4	5,413	38.5	46,141	328.4
mceliece6688128^{cpa}	HW	HS	136	44,345	83,637	–	0	446.0	12,389,742	91,034.1	5,024	36.9	52,333	384.5
mceliece8192128^{cpa}	HW	HS	134	45,150	88,154	–	0	525.0	15,185,314	113,154.4	6,528	48.6	55,330	412.3
SIKEp751	HW	HS	167	45,893	50,390	17,53	512	43.5	1,240,000	7,407.4	2,170,000	12,963.0	2,330,000	13,918.8
SIKEp751	SW/HW	HS	142	21,210	13,657	7,408	162	38.0	2,516,940	17,700.0	4,166,460	29,300.0	4,479,300	31,500.0
SIKEp751	SW/HW	LW	152	10,937	7,132	3,415	57	21.0	7,960,426	52,300.0	13,150,685	86,400.0	14,185,693	93,200.0

Table 11 – All KEMs and PKEs on Zynq Ultrascale +

Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation		Encapsulation		Decapsulation	
									cycles	*μs*	cycles	*μs*	cycles	*μs*
Security level 1														
R5ND_1KEM_0d	SW/HW	HS	260	55,442	82,341	10,627	0	2	–	–	–	19.0	–	24.0
LightSaber	SW/HW	HS	322	12,343	11,288	1,989	256	3.5	–	–	–	53.0	–	56.0
FrodoKEM-640	SW/HW	HS	402	7,213	6,647	1,186	32	13.5	–	–	–	1,223.0	–	1,319.0
Security level 3														
Saber	HW	HS	250	45,895	18,705	–	0	2	4,320	17.3	5,231	20.9	6,461	25.8
Saber	HW	HS	250	25,079	10,750	–	0	2	5,435	21.8	6,618	26.5	8,034	32.1
R5ND_3KEM_0d	SW/HW	HS	249	73,881	109,211	14,307	0	2	–	–	–	24.0	–	33.0
Saber	SW/HW	HS	322	12,566	11,619	1,993	256	3.5	–	–	–	60.0	–	65.0
FrodoKEM-976	SW/HW	HS	402	7087	6693	1190	32	17	–	–	–	1,642.0	–	1,866.0
Security level 5														
R5ND_5KEM_0d	SW/HW	HS	212	91,166	151,019	18,733	0	2	–	–	–	32.0	–	42.0
NewHope-1024^{cpa}	HW	HS	406	13,961	8,149	–	25	18	–	–	34,000	83.0	30,600^{KD}	75.0^{KD}
FireSaber	SW/HW	HS	322	12,555	11,881	2,341	256	3.5	–	–	–	74.0	–	80.0
FrodoKEM-1344	SW/HW	HS	417	7,015	6,610	1,215	32	17.5	–	–	–	2,186.0	–	3,120.0

Table 12 – Digital Signature Schemes on Artix-7, Kintex-7 and Virtex-7

Algorithm	Type	Target	Max. Freq.	LUT	FF	Slice	DSP	BRAM	Key Generation		Signature Verification		Signature Generation		Family
									cycles	us	cycles	us	cycles	us	
Security Level 1 & 2															
Picnic-L1-FS	HW	HS	91	90,535	23,516	25,160	0	52.5	–	–	29,600	325.6	31,300	344.3	
qTESLA-I [o2]	SW/HW	LW	25*	14,975	2,539	4,173	11	14.0	4,846,949	193,878.0	38,922	1,556.9	168,273	6,730.9	
Dilithium-I	SW/HW	LW	25*	14,975	2,539	4,173	11	14.0	95,202	3,808.1	142,576	5,703.0	376,392	15,055.7	Artix-7
Dilithium-II	SW/HW	LW	25*	14,975	2,539	4,173	11	14.0	130,022	5,200.9	184,933	7,397.3	514,246	20,569.8	
qTESLA-p-I	SW/HW	LW	121	7,212	4,378	2,438	15	139.0	925,431	7,648.2	946,520	7,822.5	4,165,160	34,422.8	
Rainbow-Ic [o1]	HW	HS	90	52,895	32,476	15,112	0	67.0	–	–	–	–	979	10.9	
Rainbow-Ia	HW	HS	111	27,712	27,679	8,939	0	59.0	–	–	–	–	1,980	17.8	Kintex-7
Picnic-L1-FS	HW	HS	125	90,037	23,105	–	0	52.5	–	–	29,600	237.0	31,300	250.0	
Rainbow-Ic [o1]	HW	HS	167	52,721	32,475	15,976	0	67.0	–	–	–	–	979	5.9	Virtex-7
Rainbow-Ia	HW	HS	181	27,556	27,675	7,065	0	59.0	–	–	–	–	1,980	10.9	
Security Level 3															
qTesla-III-speed [o2]	SW/HW	LW	25*	14,975	2,539	4,173	11	14.0	11,898,241	475,929.6	67,712	2,708.5	317,083	12,683.3	
qTesla-III-size [o2]	SW/HW	LW	25*	14,975	2,539	4,173	11	14.0	11,479,190	459,167.6	69,154	2,766.2	348,429	13,937.2	Artix-7
Dilithium-III	SW/HW	LW	25*	14,975	2,539	4,173	11	14.0	167,433	6,697.3	229,481	9,179.2	634,763	25,390.5	
qTESLA-p-III	SW/HW	LW	121	7,475	4,518	2,473	15	147.0	2,305,220	19,051.4	2,315,950	19,140.1	7,745,088	64,009.0	
Security Level 4 & 5															
Picnic-L5-FS	HW	HS	125	167,530	33,164	–	0	98.5	–	–	146,600	1,173.0	154,500	1,236.0	Kintex-7
Dilithium-IV	SW/HW	LW	25*	14,975	2,539	4,173	11	14.0	223,272	8,930.9	276,221	11,048.8	815,636	32,625.4	Artix-7

For security level 3, no implementation of Picnic is available. The implementations of Dilithium-III and qTESLA-p-III are comparable in terms of type, target, and resource utilization. At the same time, the implementation of Dilithium is an order of magnitude more efficient. The implementations of digital signature schemes targeting Kintex-7 and Virtex-7 are summarized in the same table. For the Kintex-7 implementations, Rainbow substantially outperforms Picnic at the security level 1. For all remaining families and security levels, only one candidate with the up-to-date parameter set is reported.

7 Conclusions

In this paper, we first reviewed the previous work on hardware and software/hardware implementations of Round 2 PQC schemes. Out of 26 candidates, six – NewHope, CRYSTALS-Kyber, FrodoKEM, Saber, Round5, and SIKE – received the highest coverage in terms of the number of implementations and related publications. All of them have high-speed and simplified implementations reported. Applied software/hardware co-design to high-speed rather than lightweight implementations, which led to the choice of Xilinx Zynq UltraScale+, a state-of-the-art SoC FPGA family, as our primary platform. What matters is that this platform includes a «hardwired» ARM Cortex-A53 processor operating at the frequency of 1.2 GHz and a significant amount of programmable logic supporting hardware accelerators operating at the clock frequencies up to 500 MHz.

For each candidate, an attempt was made to offload as many as possible operations to hardware. For 50% of investigated KEMs, this percentage reached 100%. Thus, the corresponding implementations could be treated as hardware implementations, assuming that a random seed (*of 16, 24, or 32 bytes*) was transferred to the hardware module during encapsulation. KEMs implemented using this approach included Kyber, LAC (v3a and v3b), NewHope, and Round5 (*with and without error-correcting code*). Their code was benchmarked using Artix-7 and Virtex-7 FPGAs.

In terms of both the execution times and resource utilization, Round5 with an error-correcting code (R5ND_5d) outperformed Round5 without an error-correcting code (R5ND_0d). Similarly, LAC-v3b appeared superior over LAC-v3a in terms of both speed and use of FPGA resources. Then, when the best representatives of four candidates – Kyber, LAC, NewHope, and Round5 – were compared, the following conclusions could be drawn. The execution times of these candidates were extremely close to one another. For encapsulation, the execution times were within 10% from one another at the security level 5, within 22% at the security level 3, and within 32% at the security level 1. For decapsulation, the largest differences were 26% at level 5, 22% at level 3, and 48% at level 1. In multiple instances, just a change of an FPGA family from low-cost Artix-7 to high-performance Virtex-7 caused a significant change in the rankings, even though the HDL code remained exactly the same. As a result, we must conclude that the differences among these candidates in terms of speed are too small to give preference to any particular candidate. These results contradict one of the earlier reports placing LAC well behind NewHope and Kyber.

In terms of resource utilization, a small advantage belongs to NewHope and Kyber. Both of them use fewer LUTs and flip-flops than LAC and Round5, and their use of DSP units and BRAMs, although slightly higher, is very moderate. Additionally, both NewHope and Kyber use almost the same amount of resources independently of the security level. In the case of both LAC and Round5, resource usage increases sharply with the increase in security level. The former property appears to be an advantage for applications requiring support for the highest or all security levels. In particular, the k-in-1 designs, which support all k security levels and allow modifying them at run time, typically have only slightly higher resource utilization than that for the maximum security level. Thus, the flat dependence of the resource utilization on the security level implies a potential for very cost-effective k-in-1 designs. At the same time, this potential should still be confirmed through complete designs.

A detailed characterization of the FPGA Xilinx family was also provided. Each particular FPGA should be used based on purpose, expected cost, and performance.

References

[1] J.-S. Coron, A. Joux, Cryptanalysis of a provably secure cryptographic hash function, Cryptology ePrint Archive Report 2004/013, 2004. http://eprint.iacr.org/2004/013

[2] Post-quantum cryptography, round 2 submissions. [Електронний ресурс]. – Режим доступу: https://csrc.nist.gov/projects/post-quantum-cryptography/round-2-submissions.

[3] Malik Imran A Systematic Study of Lattice-based NIST PQC Algorithms: from Reference Implementations to Hardware Accelerators / Malik Imran, Zain Ul Abideen, Samuel Pagliarini //. – Режим доступу: https://arxiv.org/pdf/2009.07091.pdf.

[4] Viet Ba Dang Implementation and Benchmarking of Round 2 Candidates in the NIST Post-Quantum Cryptography Standardization Process Using Hardware and Software/Hardware Co-design Approaches / Viet Ba Dang, Farnoud Farahmand, Michal Andrzejczak, Kamyar Mohajerani, Duc Tri Nguyen, Kris Gaj //. – Режим доступу: https://eprint.iacr.org/2020/795.pdf.

[5] И. Тарасов ПЛИС Xilinx и Цифровая обработка Сигналов Особенности, преимущества, перспективы. – Режим доступу: https://www.electronics.ru/files/article_pdf/2/article_2788_434.pdf.

[6] Xilinx. 7 Series Product Selection Guide. [Електронний ресурс]. – Режим доступу: https://www.xilinx.com/support/documentation/selection-guides/7-series-product-selection-guide.pdf.

[7] Farnoud Farahmand et al. Software/Hardware Codesign of the Post Quantum Cryptography Algorithm NTRUEncrypt Using High-Level Synthesis and Register-Transfer Level Design Methodologies. In: 29th International Conference on Field Programmable Logic and Applications, FPL 2019. Barcelona, Spain: IEEE, Sept. 2019, pp. 225–231. ISBN: 978-1-72814-884-7. DOI: 10.1109/FPL.2019.00042.

[8] Kris Gaj Challenges and Rewards of Implementing and Benchmarking Post-Quantum Cryptography in Hardware. In: 2018 Great Lakes Symposium on VLSI, GLSVLSI 2018. Chicago, IL, USA: ACM Press, 2018, pp. 359–364. ISBN: 978-1-4503-5724-1. DOI: 10/ggbscs.

[9] Jens-Peter Kaps et al. Lightweight Implementations of SHA-3 Candidates on FPGAs. In: 12th International Conference on Cryptology in India, Indocrypt 2011. Vol. 7107. LNCS. Chennai, India, Dec. 2011, pp. 270–289. ISBN: 978-3-642-25577-9 978-3-642-25578-6. DOI: 10.1007/978-3-642-25578-6_20. – Режим доступу: https://2011.indocrypt.org/slides/gurung.pdf.

[10] Viet B Dang et al. Implementing and Benchmarking Three Lattice-Based Post-Quantum Cryptography Algorithms Using Software/Hardware Codesign. In: 2019 International Conference on Field Programmable Technology, FPT 2019. Tianjin, China: IEEE, Dec. 9-13, 2019, pp. 206–214. DOI: 10.1109/ICFPT47387.2019.00032.

Рецензент: Сергій Толюпа, д.т.н., проф., Київський національний університет імені Т. Шевченка, м. Київ, Україна.
E-mail: tolupa@i.ua

Надійшло: Грудень 2020.

Автори:

Марина Єсіна, к.т.н., доцент кафедри безпеки інформаційних систем і технологій, Харківський національний університет імені В. Н. Каразіна, майдан Свободи 6, м. Харків, 61022, Україна.
E-mail: m.v.yesina@karazin.ua

Богдан Шахов, студент кафедри безпеки інформаційних систем і технологій, Харківський національний університет імені В.Н. Каразіна, майдан Свободи 4, м. Харків, 61022, Україна.
E-mail: bogdanshahov2000@gmail.com

Дослідження реалізацій кандидатів другого раунду конкурсу NIST PQC, що орієнтовані на сімейства FPGA Xilinx.

Анотація. Сьогодні досить гостро постає питання щодо стійкості сучасних існуючих криптографічних механізмів до квантових алгоритмів криптоаналізу зокрема та квантових комп'ютерів взагалі. Ця проблема активно обговорюється на міжнародному рівні. Тому, задля її вирішення, NIST США вирішив організувати та проводить на сьогоднішній день конкурс на кандидатів на постквантові криптографічні алгоритми NIST PQC. Результатом конкурсу повинне стати прийняття до стандартизації криптографічних алгоритмів різного типу – асиметричне шифрування, інкапсуляція ключів та електронний підпис (як мінімум по одному алгоритму з кожного типу). На момент початку конкурсу на процес стандартизації було представлено 82 алгоритми. На основі критеріїв мінімальної прийнятності, визначених NIST, для 1-го раунду було розглянуто 69 алгоритмів. Враховуючи декілька параметрів – безпеку, вартість, продуктивність, характеристики реалізації тощо, 43 і 11 алгоритмів були виключені при завершенні 1-го і 2-го раундів відповідно, а інші 15 алгоритмів були збережені для 3-го раунду. Алгоритми, які залишилися у 2-му раунді можна розділити на 5 різних категорій залежно від математичного базису, на якому вони засновуються: на основі ізогеній еліптичних кривих, на основі алгебраїчних решіток, на основі математичного коду, на основі багатовимірних перетворень і на основі геш-функцій. Безпека є основним критерієм оцінки, що визначає конкуренцію в конкурсі NIST, і, зрозуміло, що реалізації програмного забезпечення кандидатів в основному зосереджені на ній. Однак, вкрай важливо аби алгоритм мав й ефективну апаратну реалізацію. А своєчасне виявлення апаратної неефективності допоможе сконцентрувати зусилля криптографічної спільноти на більш перспективних кандидатах, потенційно заощадивши велику кількість часу, що може бути витрачена на криптоаналіз. У даній роботі розглядаються та порівнюються між собою FPGA сімейства Xilinx. Наводяться та порівнюються між собою дані щодо реалізацій кандидатів 2-го раунду в процесі стандартизації постквантової криптографії NIST, що орієнтовані на FPGA сімейства Xilinx.

Ключові слова: електронний підпис; постквантова криптографія; конкурс NIST PQC; FPGA, Xilinx.

Рецензент: Сергей Толюпа, д.т.н., проф., Киевский национальный университет имени Т. Шевченко, г. Киев, Украина.
E-mail: tolupa@i.ua

Поступила: Декабрь 2020.

Авторы:
Марина Есина, к.т.н., доцент кафедры безопасности информационных систем и технологий, ХНУ им. В. Н. Каразина, пл. Свободы 6, Харьков, 61022, Украина.
E-mail: m.v.yesina@karazin.ua
Богдан Шахов, студент факультета компьютерных наук, ХНУ им. В. Н. Каразина, пл. Свободы, 4, Харьков, 61022, Украина.
E-mail: bogdanshahov2000@gmail.com

Исследование реализаций кандидатов второго раунда конкурса NIST PQC, ориентированных на семейства FPGA Xilinx.

Аннотация. Сегодня достаточно остро стоит вопрос о стойкости современных существующих криптографических механизмов к квантовым алгоритмам криптоанализа в частности и квантовым компьютерам вообще. Эта проблема активно обсуждается на международном уровне. Поэтому, для ее решения, NIST США решил организовать и проводит на сегодняшний день конкурс на кандидатов на постквантовые криптографические алгоритмы NIST PQC. Результатом конкурса должно стать принятие к стандартизации криптографических алгоритмов разного типа – асимметричное шифрование, инкапсуляция ключей и электронная подпись (как минимум по одному алгоритму с каждого типа). На момент начала конкурса на процесс стандартизации было представлено 82 алгоритмы. На основе критериев минимальной приемлемости, определенных NIST, для 1-го раунда было рассмотрено 69 алгоритмов. Учитывая несколько параметров – безопасность, стоимость, производительность, характеристики реализации и т.п., 43 и 11 алгоритмов были исключены при завершении 1-го и 2-го раундов соответственно, а остальные 15 алгоритмов были сохранены для 3-го раунда. Алгоритмы, которые остались во 2-м раунде можно разделить на 5 различных категорий в зависимости от математического базиса, на котором они основываются: на основе изогений эллиптических кривых, на основе алгебраических решеток, на основе математического кода, на основе многомерных преобразований и на основе хеш-функций. Безопасность является основным критерием оценки, определяет конкуренцию в конкурсе NIST, и, понятно, что реализации программного обеспечения кандидатов в основном сосредоточены на ней. Однако, крайне важно, чтобы алгоритм имел и эффективную аппаратную реализацию. А своевременное выявление аппаратной неэффективности поможет сконцентрировать усилия криптографического сообщества на более перспективных кандидатах, потенциально сэкономив большое количество времени, которое может быть потрачено на криптоанализ. В данной работе рассматриваются и сравниваются между собой FPGA семейства Xilinx. Приводятся и сравниваются между собой данные по реализаций кандидатов 2-го раунда в процессе стандартизации постквантовой криптографии NIST, ориентированные на FPGA семейства Xilinx.

Ключевые слова: электронная подпись; постквантовая криптография; конкурс NIST PQC; FPGA, Xilinx.