Post-Quantum IoT Security

For

CMC Research Workshop – Secure IoT Hardware

Yu Zhang, Cinnati Loi, Karim Shahbazi and Seok-Bum Ko, PhD., P.Eng.
Department of Electrical and Computer Engineering
University of Saskatchewan

February 25, 2022

UNIVERSITY OF SASKATCHEWAN

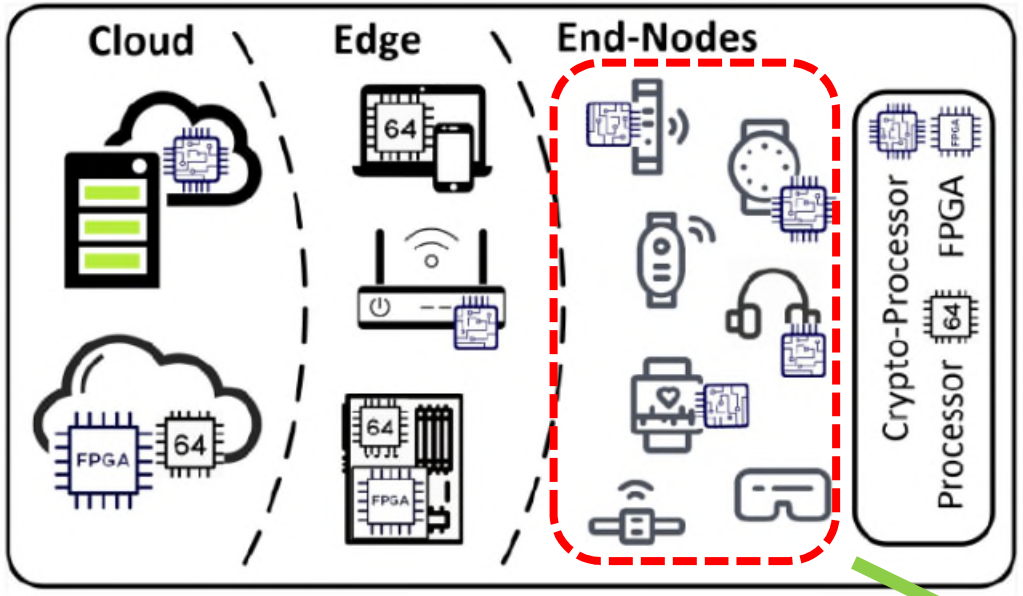
Contents

- **Introduction**
  - Cryptography in large network
  - Cryptography algorithms
  - Shor's algorithm and Post Quantum Cryptography
  - AES algorithm
  - Lattice Based Cryptography
  - Learning With Error (LWE)
  - Ring-LWE and Ring-Bin LWE
- **Proposed Design and Architecture**
  - Area-Efficient Nano-AES Implementation for Internet of Things Devices
  - Post-Quantum Cryptosystem for IoT Resource-Constrained Devices
  - Lightweight and CCA2-Secure Implementation of Ring-BinLWE
- **Conclusions**

Introduction to Cryptography

Encryption **Decryption**

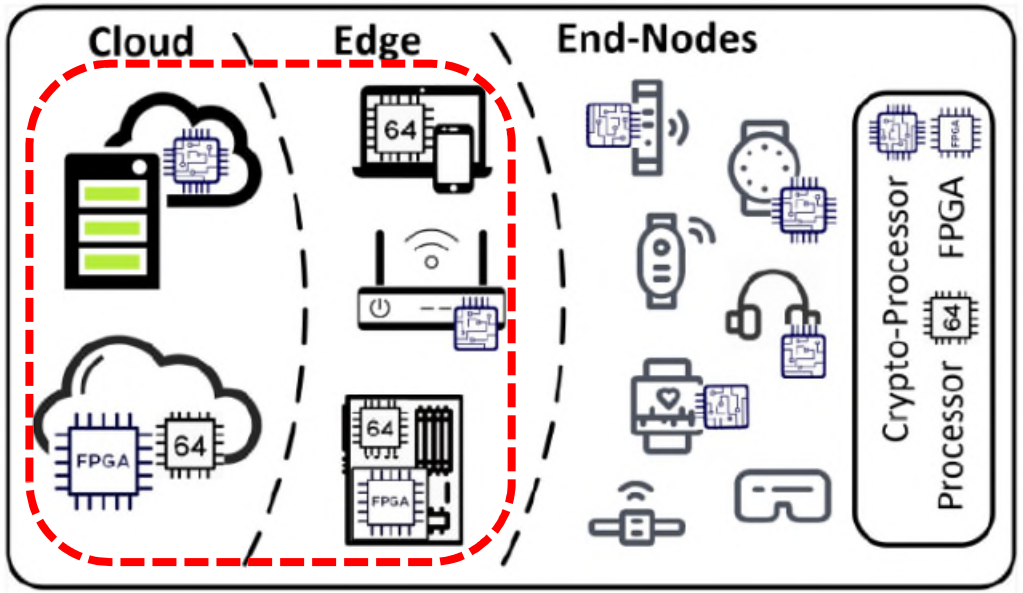
Architecture of IoT Network and Available Hardware Resources in each Layer



Number of connected devices to IoT

Ebrahimi, Shahriar, Siavash Bayat-Sarmadi, and Hatameh Mosanaei-Boorani. "Post-quantum cryptoprocessors optimized for edge and resource-constrained devices in IoT." *IEEE Internet of Things Journal* 6.3 (2019): 5500-5507.
Alam, Tanweer. "A reliable communication framework and its use in internet of things (IoT)." *CSEIT1835111| Received* 10 (2018): 450-456.

Architecture of IoT Network and Available Hardware Resources in each Layer.



Require a reliable high-speed cryptosystem

Different cryptography

Symmetric key (private key)

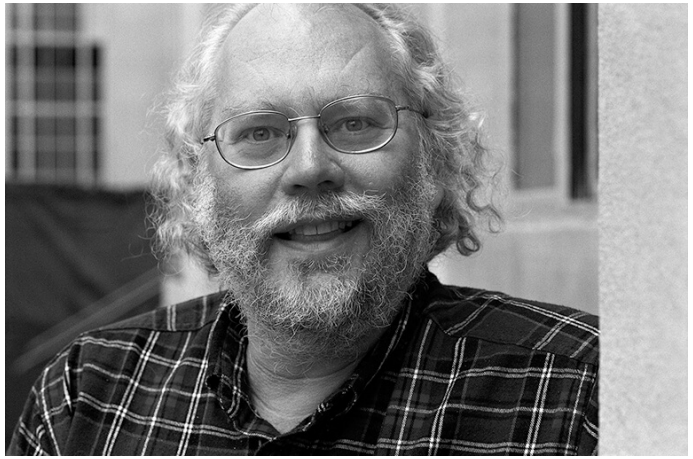
Such as: AES, IDEA, and PRESENT

Asymmetric key (public key)

Such as: RSA, Ring-LWE, and ECC

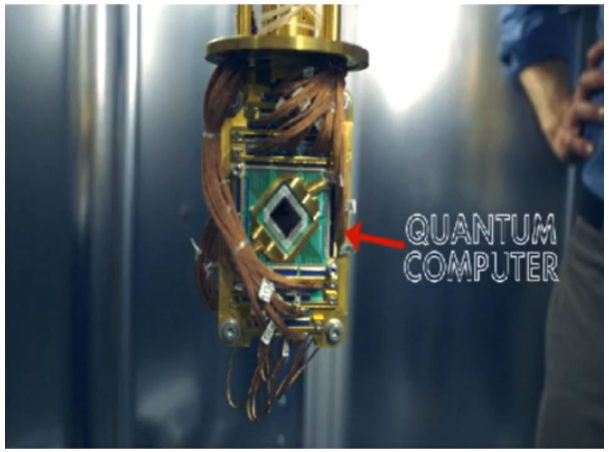
Shor's algorithm

- ✓ In 1994, Peter Shor introduced the factoring algorithm
- ✓ Shor's algorithm works by quickly finding the prime factors of any integer.



Peter Shor

- ✓ When we are using a quantum computer, we can use Shore's algorithm to make it really effective at finding these prime factors.



- ✓ Unfortunately, most encryptions rely on the prime factors of integers in order to be secure.

Shor, Peter W, "Algorithms for quantum computation: discrete logarithms and factoring," *Proceedings 35th annual symposium on foundations of computer science*. IEEE, 1994.

AES algorithm

- AES stands for Advanced Encryption Standard
- AES established by National Institute of Standards and Technology (NIST) in 2001
- AES is used in a wide range of applications
- There are no successful attacks against AES
- AES-256 is secure in quantum era

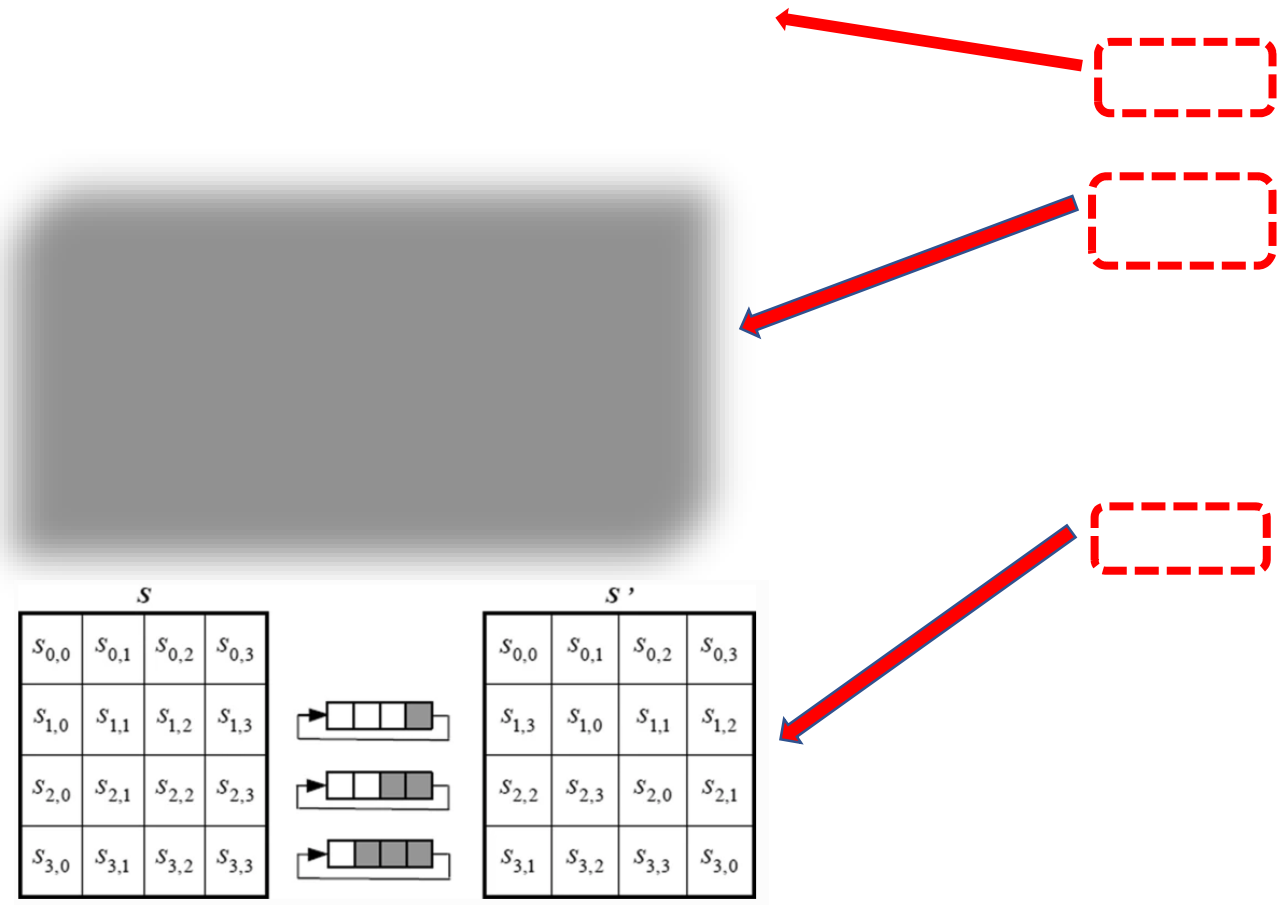
Federal Information Processing Standards Publication 197, *Advanced Encryption Standard FIPS PUB 97*, 2001, pp. 1–51
Liu, Zhe, Kim-Kwang Raymond Choo, and Johann Grossschadl. "Securing edge devices in the post-quantum internet of things using lattice-based cryptography." *IEEE Communications Magazine* 56.2 (2018): 158-162.
Karim Shahbazi and Seok-Bum Ko, "High throughput and area-efficient FPGA implementation of AES for high-traffic applications," *IET Computers & Digital Techniques* 14.6 (2020): 344-352.

Introduction to AES algorithm

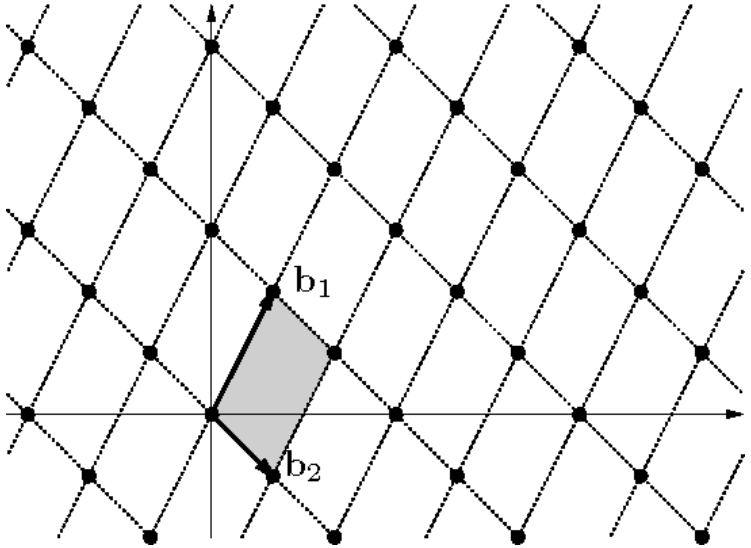
- AES contains four main blocks: Add-Round-Key, Shift-Rows, Mix-Columns, and Sub-Bytes
- The key and plaintext are 128-bit
- The number of rounds is 10

N_r		N_b		
		4	**6**	**8**
N_k	**4**	10	12	14
	6	12	12	14
	8	14	14	14

Number of rounds depending on key and input length



A lattice $L \in Z^n$ is the set of all integer linear combinations of ***n* independent basis vectors b_i**



Closest vector problem (CVP)
Shortest vector problem (SVP)

There is no algorithm classic or quantum to solve these problems (so far!)

Learning With Error:

$$(A, \vec{S}A + \vec{e})$$

It is hard to find 'S'

$\vec{e}$ is error vector chosen from a distribution

A, S $\in Z_q$

The shortest and closest vector problem are hard. **Thus, LWE are hard**.

Ring-LWE:

Key Genration :

Choose $r1, r2 \leftarrow \chi_k$ and calculate $p = r1 - a \times r2 \in Rq$ public key (a, p), and secret key $r2$.

Encryption:

*Enc(***a, p, m** $\in \{0, 1\}^n$*):*

1. Choose $e1, e2, e3 \leftarrow \chi e$
2. $\bar{m} = \lfloor (q/2) \rfloor m \in R_q$
3. $C_1 = a \times e_1 + e_2$

   $C_2 = P \times e_1 + e_3 + \bar{m}$

Decryption:

Dec(C_1, C_2, r_2)**:**

1. Calculate $\bar{\mathbf{m}} = \mathbf{C_1} \times \mathbf{r_2} + \mathbf{C_2} \in \mathbf{R_q}$
2. $\bar{\mathbf{m}}$ should convert into {0, 1}

   if $\bar{\mathbf{m}} \in$ [(q/4) , (3q/4)) then m[i] = 1, otherwise m[i] = 0.

Ring-Bin LWE

- ✓ This new variant of the ideal-lattice based encryption scheme was introduced in 2016.
- ✓ Ring-BinLWE has a binary distribution, which is very suitable for resource-restricted devices.
- ✓ In Ring-BinLWE, the errors are sampled with binary coefficients, and thus Ring-BinLWE does not require Gaussian distribution and NTT; also, the key size is smaller.
- ✓ Multiplication and addition are two main operations.

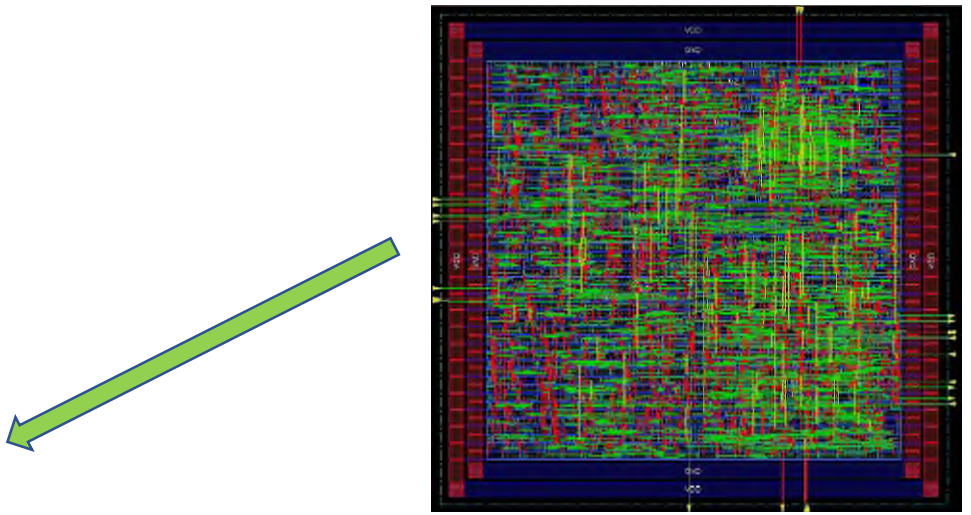
Contents

- **Introduction**
  - Cryptography in large network
  - Cryptography algorithms
  - Shor's algorithm and Post Quantum Cryptography
  - AES algorithm
  - Lattice Based Cryptography
  - Learning With Error (LWE)
  - Ring-LWE and Ring-Bin LWE
- **Proposed Design and Architecture**
  - Area-Efficient Nano-AES Implementation for Internet of Things Devices
  - Post-Quantum Cryptosystem for IoT Resource-Constrained Devices
  - Lightweight and CCA2-Secure Implementation of Ring-BinLWE
- **Conclusions**

8-Bit Nano-AES data path accelerator

The design includes:

- One Sub-Bytes for both key expansion and encryption.
- One 8-bit Mix-columns
- One register banks for storing keys (Key-Register)
- One register bank for plain-text (State-Register)
- RCON block
- Control unit



Karim Shahbazi, and Seok-Bum Ko, "Area-efficient nano-AES implementation for Internet-of-Things devices," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 29.1 (2020): 136-148.

The structure of the proposed State-Register with Shift-Rows

- Storing the main plain-text
- Executing Shift-Rows
- Storing the data that comes from Mix-Columns only in the last column
- Feeding the design by one 8-bit and storing the data at the same time

The proposed State-Register with Shift-Rows, control circuitry, and clock gating technique

The structure of the proposed Sub-Bytes

The optimized structure of combination of inverse isomorphic with Affine Transformation

(a) The modified architecture of Sub-Bytes [1] with combination of inverse isomorphic with Affine Transformation (γ) and bypass circuit
(b) The multiplication operation in GF(2^4)
(c) The multiplication operation in GF(2^2)

[1] X. Zhang and K. K. Parhi, "High-speed VLSI architectures for the AES algorithm," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 12, no. 9, pp. 957–967, 2004.

The structure of the proposed Mix-Columns

$2.a + a$

$$a'_0 = 2.a_0 + 3.a_1 + a_2 + a_3$$
$$a'_1 = a_0 + 2.a_1 + 3.a_2 + a_3$$
$$a'_2 = a_0 + a_1 + 2.a_2 + 3.a_3$$
$$a'_3 = 3.a_0 + a_1 + a_2 + 2.a_3$$

$$(2.a)\text{mod}(x^8 + x^4 + x^3 + x + 1) = \begin{cases} x_7 = a_6 \\ x_6 = a_5 \\ x_5 = a_4 \\ x_4 = a_7 \oplus a_3 \\ x_3 = a_7 \oplus a_2 \\ x_2 = a_1 \\ x_1 = a_7 \oplus a_0 \\ x_0 = a_7. \end{cases}$$

The timing diagram of the proposed Mix-Columns

The proposed architecture of Mix-Columns with clock gating technique and bypass circuit

Implementation results and comparison

Design	#Sub-bytes	#Clock Cycles	Frequency (MHz)	Area ($\times 10^3$ μm^2)	Power (μW)	Normalized power (μW)
Our Design*	1	527	100	11.7	245.60 @ 1.1 V	245.6
[1]	1	210	127.2	13	97.9 @ 0.55 V	307.862
[2][2]	1	242	200	10	3460 @1.2 V	1453.6
[3][3]	2	186	10	>10	10.01 @ 0.9 V	149.53
[4][1]	2	-	11	12	14.6 @ 0.5 V	642.4
[5][1]	1	1142	0.322	18	0.85 @ 0.4V	1996.31
[6][2]	2	160	-	14.4	0.38 (μW/MHz)	-

[1] The reported area is chip area.
[2] The reported area is core with power rings
[3] The reported area is core without power rings
* The area of the proposed design contains $5.4 \times 10^3 \mu m^2$ and $7.7 \times 10^3 \mu m^2$ for core without power rings and core with power rings, respectively.

Note: The references are written on the next slide

References

[1] V.-P. Hoang, V.-L. Dao, and C.-K. Pham, "Design of ultra-low power AES encryption cores with silicon demonstration in SOTB CMOS process*," Electronics Letters*, vol. 53, no. 23, pp. 1512–1514, 2017.
[2] A. Shreedhar, K.-S. Chong, N. Lwin, N. Kyaw, L. Nalangilli, W. Shu, J. Chang, and B.-H. Gwee, "Low Gate-Count Ultra-Small Area Nano Advanced Encryption Standard (AES) Design," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2019, pp. 1–5.
[3] H. K. Kim and M. H. Sunwoo, "Low Power AES Using 8-Bit and 32- Bit Datapath Optimization for Small Internet-of-Things (IoT)," *Journal of Signal Processing Systems*, vol. 91, no. 11-12, pp. 1283–1289, 2019.
[4] W. Zhao, Y. Ha, and M. Alioto, "AES architectures for minimum-energy operation and silicon demonstration in 65nm with lowest energy per encryption," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2015, pp. 2349–2352.
[5] C. Hocquet, D. Kamel, F. Regazzoni, J.-D. Legat, D. Flandre, D. Bol, and F.-X. Standaert, "Harvesting the potential of nano-CMOS for lightweight cryptography: an ultra-low-voltage 65 nm AES coprocessor for passive RFID tags," *Journal of Cryptographic Engineering*, vol. 1, no. 1, pp. 79–86, 2011.
[6] Van-Phuc, V.-L. Dao, and C.-K. Pham, "An ultra-low power AES encryption core in 65nm SOTB CMOS process," in *International SoC Design Conference (ISOCC)*, 2016, pp. 89–90.

The straightforward method of multiplication for Ring-Bin LWE

- ✓ Each single coefficient of multiplier is multiplied by all coefficients of multiplicand to generate partial products.
- ✓ The anti-circular rotation and reduction occur in each row for partial products (the yellow coefficients).
- ✓ Partial products are added into the intermediate sum, followed by modular reduction.
- ✓ The final result of the polynomial multiplication is computed by accumulating the intermediate sum, followed by modular reduction.

The traditional method for multiplication for Ring-Bin LWE

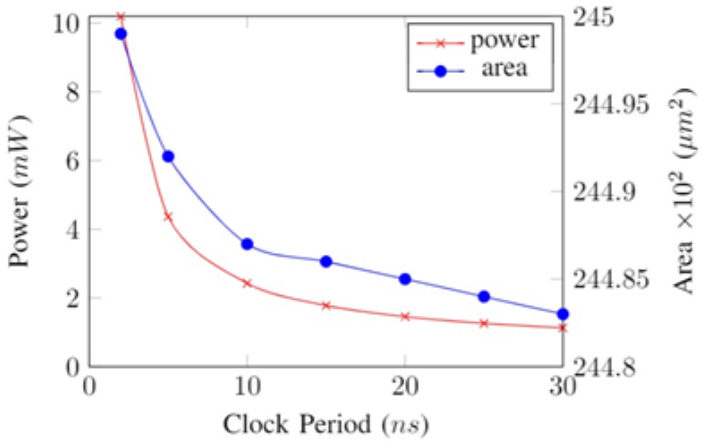
Karim Shahbazi, and Seok-Bum Ko, "Area and power efficient post-quantum cryptosystem for IoT resource-constrained devices," *Microprocessors and Microsystems* (2021): 104280.

In-place Reduction and anti-circular Rotation Column-based Multiplication

The hardware design of the proposed multiplication with the related coefficients of the first column

The Proposed method of multiplication for Ring-Bin LWE

Hardware design and implementation result



Area-delay curve and Power-delay curve of the proposed design

Results and comparison for cryptosystem implementation on TSMC-65nm

REF	Freq. (MHz)	Area		#CC Enc/Dec	Time (μm) Enc/Dec	Power (mW)
		μm²	GE			
This work	33.33	4637.51	3.2 K	132929/66337	3987.87/1990.11	0.196
[1]	33.33	11 K	7.9 K	131840/65792	3.8k/1.9k	0.38

[1] S. Ebrahimi, S. Bayat-Sarmadi, and H. Mosanaei-Boorani, "Post-Quantum Cryptoprocessors Optimized for Edge and Resource-Constrained Devices in IoT," *IEEE Internet of Things Journal*, 2019.

Post-Quantum Cryptosystem for IoT Resource-Constrained Devices

University of Saskatchewan

Results and comparison for cryptosystem implementation on FPGA

Ref.	Scheme	Devices	Freq. (MHz)	LUT/FF/Slice	DSP/ BRAM	#CC Enc/Dec	Time (µs) Enc./Dec.	ENS	AT Enc./Dec.
This work	R-BLWE	Virtex-7	434.32	380/640/165	0	133k/66k	307.94/153.67	165	50.81k/25.35k
		Spartan-6	279.64	444/642/146	0	133k/66k	477.98/238.93	146	69.78k/34.88k
[1]	R-BLWE	Virtex-7	540/560	2k/2k/652	0	512/256	0.95/0.46	652	619.4/299.92
[2]	R-BLWE	Spartan-6	-/135	57/30/19	0/2	-/65.79k	-/487.4	131	-/63.84k
[3]	R-LWE	Virtex-6	313	1349/860/-	1/2(18k)	6.3k/2.8k	20.1/9.1	-	-
[4]	R-LWE	Kintex-7	275	1381/1179/479	2/2(8k)	35.45k/17.73k	129/64	795.8	102.65k/50.93k

ENS stands for Equivalent Number of Slices. Based on [5], One DSP equivalent to 102.4 Slices, one 8k BRAM equivalent to 56 Slices, and one 18k BRAM equivalent to 116.2.
AT=number of Slices×Time in microsecond ($Slices \times \mu sec$).

Note: The references are written on the next slide

References

[1] Ebrahimi, Shahriar, Siavash Bayat-Sarmadi, and Hatameh Mosanaei-Boorani, "Post-quantum cryptoprocessors optimized for edge and resource-constrained devices in IoT," *IEEE Internet of Things Journal* 6.3 (2019): 5500-5507.
[2] Aysu, Aydin, Michael Orshansky, and Mohit Tiwari, "Binary Ring-LWE hardware with power side-channel countermeasures," *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*. IEEE, 2018.
[3] Roy, Sujoy Sinha, et al, "Compact ring-LWE cryptoprocessor," *International workshop on cryptographic hardware and embedded systems*. Springer, Berlin, Heidelberg, 2014.
[4] Zhang, Yuqing, et al, "An efficient and parallel R-LWE cryptoprocessor," *IEEE Transactions on Circuits and Systems II: Express Briefs* 67.5 (2020): 886-890.
[5] Liu, Weiqiang, et al, "Optimized schoolbook polynomial multiplication for compact lattice-based cryptography on FPGA," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems* 27.10 (2019): 2459-2463.

Fault injection attacks:
The data can be manipulated by injecting an error into a memory or a signal to obtain the faulty execution of the system and try to extract message or private key by evaluating the results.

There are three different fault attracts over cryptosystems:

Randomization:

some part of memory is set to random values. For Ring-BinLWE, randomization fault does not have any impact over key generation and encryption phases

Skipping:

is avoiding to run certain operations in the algorithm, such as addition and multiplication.

Zeroing

contains setting some parts or the entire value of a coefficient to zero.

Skipping and zeroing attacks have a high impact on Ring-BinLWE.

Lightweight and CCA2-Secure Implementation of Ring-BinLWE

University of Saskatchewan

Pseudo-code 1: The proposed CCA2-Secure Ring-BinLWE

A: The proposed Ring-BinLWE Encryption with CCA2-Secure

Data: r_2, a, m

Output: c_1, c_2, c_3, c_4

begin

$v \leftarrow RNG$

$seed \leftarrow H(v, m)$

$e_1 \leftarrow HH(seed)$

$e_2 = HH(e_1)$

$e_3 = HH(e_2)$

$c_1 = a.e_1 + e_2$, $c_2 = p.e_1 + e_3 + \overline{v}$

$c_3 = G(v) \bigoplus m, c_4 = HH(v)$

end

B: The proposed Ring-BinLWE Decryption with CCA2-Secure

Data: $c_1, c_2, c_3, c_4, a, r_2$

Output: m

begin

$\overline{v} = c_1.r_2 + c_2$, $v = decode(\overline{v})$

$m = G(v) \bigoplus c_3$

$seed \leftarrow H(v, m)$

$e_1 \leftarrow HH(seed)$

$e_2 = HH(e_1)$

$e_3 = HH(e_2)$

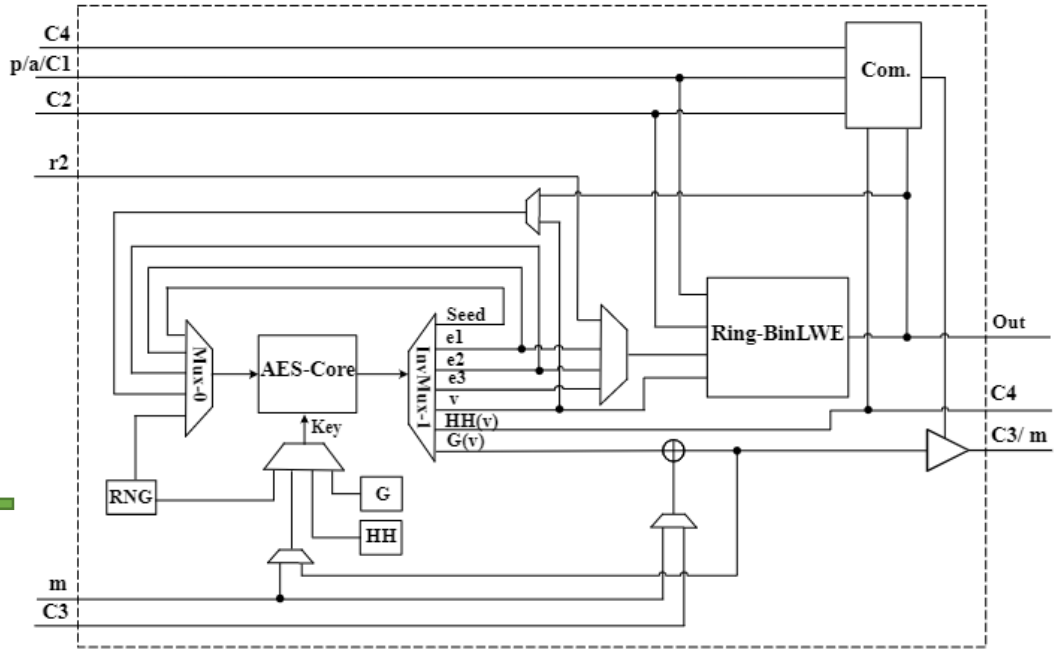
$c_1^{'} = a.e_1 + e_2$, $c_2^{'} = p.e_1 + e_3 + \overline{v}, c_4^{'} = HH(v)$

if $c_1^{'} == c_1$ and $c_2^{'} == c_2$ and $c_4^{'} == c_4$ **then**

return m

end

end



The hardware design of the proposed fault resilient Ring-BinLWE

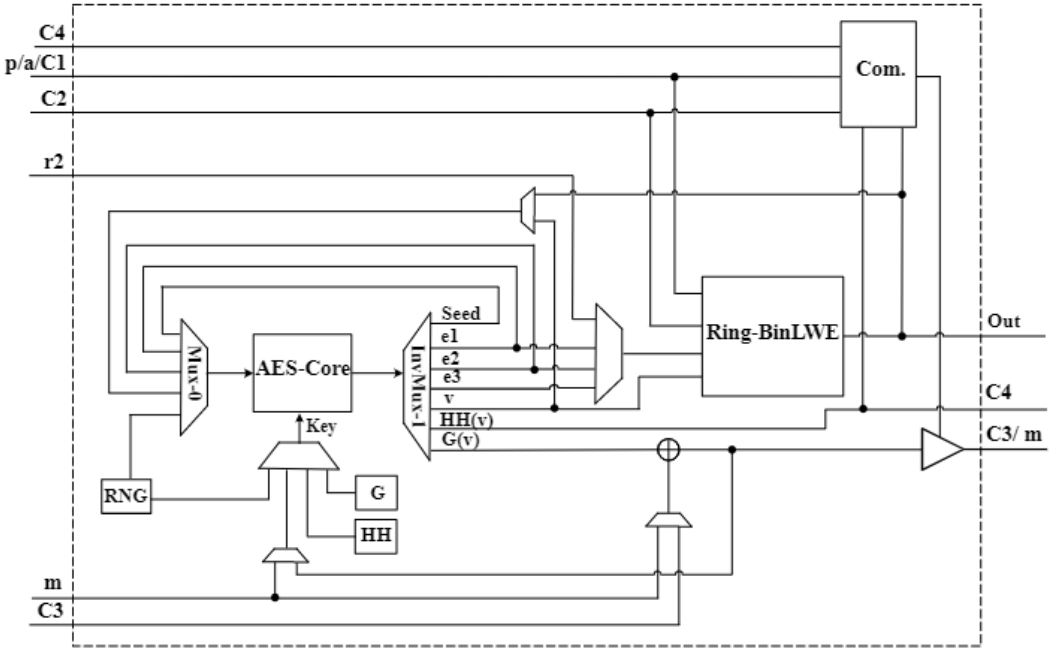
K. Shahbazi and S.-B. Ko, "Lightweight and CCA2-Secure Hardware Implementation of Ring Binary LWE," accepted to *ISCAS 2022*.

Lightweight and CCA2-Secure Implementation of Ring-BinLWE

University of Saskatchewan

References		This Work	[1][1]	[2]	[3][2]
Platform		Virtex-7	Virtex-7	AVR/ARM	Virtex-7
Frequency (MHz)		210.5	-	-	434.32
LUT		1206	1234	-	380
FF		1241	792	-	640
Slices		423	-	-	165
Clock Cycles	Enc	138k	-	2691k	133k
	Dec	204k	-	4037k	66k
Time (µs)	Enc	655.60	21.67	80200	307.94
	Dec	969.12	-	120300	153.67
AT (Slices ×µSec)	Enc	277.31k	-	-	50.81k
	Dec	409.93k	-	-	25.35k

[1] The result of this design is only for encryption phase
[2] This design does not contain fault resilient



The hardware design of the proposed fault resilient Ring-BinLWE

[1] A. Sarker, M. M. Kermani, and R. Azarderakhsh, "Fault Detection Architectures for Inverted Binary Ring-LWE Construction Benchmarked on FPGA," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 4, pp. 1403–1407, April 2021.
[2] Ebrahimi, Shahriar and Bayat-Sarmadi, Siavash, "Lightweight and Fault Resilient Implementations of Binary Ring-LWE for IoT Devices," *IEEE Internet of Things Journal*, 2020.
[3] K. Shahbazi and S.-B. Ko, "Area and power efficient post-quantum cryptosystem for IoT resource-constrained devices," *Microprocessors and Microsystems*, p. 104280, 2021.

Conclusions

- The number of connected devices to IoT is increasing day by day. Most of the connected devices to IoT are tiny devices with limited resources, providing end-to-end security is very important.
- With the quantum computer, most of the current cryptosystems are endangered.

- AES is a secure symmetric cryptography algorithm with a high level of security, which is widely used in many applications and networks.
- LBC is one of the promising PQC methods. Ring-Bin LWE is a suitable algorithm for resource-constraint devices. Ring-LWE is a secure algorithm that mainly uses in high-traffic applications.

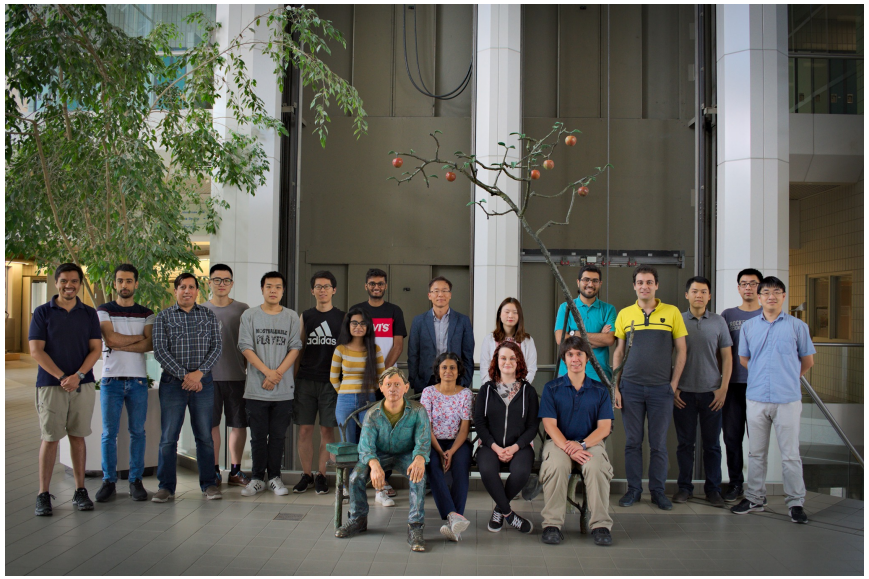
- We proposed a lightweight AES architecture and a lightweight Ring-Bin LWE for resource-constrained IoT devices.
- The design of AES had 8-bit data-path and included two specified register banks for storing plain-text, keys, and intermediate results.
- The AES core area with power rings was improved by **22.1%** over the best similar implementation.

Conclusions

- We proposed a novel Column-Based multiplication for Ring-Bin LWE.
- The proposed Ring-Bin LWE design has improved the area by **57.8%** and power by **48.42%** over the sate-of-the-art design.
- According to the result and NIST report, **the proposed lightweight AES and Ring-Bin LWE designs are suitable for resource-constrained devices and can be supplied by low-power devices.**

- Fault attack is one major threat to end-node IoT devices. The adversary tries to gain the plain-text and secret key by manipulating the data.
- To increase the security and resistance against fault attack, fault resilient countermeasure is added to the proposed Ring-Bin LWE.
- The FPGA implementation achieved a maximum frequency of 210.5MHz on Virtex-7 and occupied 423 Slices, which is only 1% of total available slices.

Acknowledgements



NSERC CRSNG

XILINX

Mitacs

intel

Jeonbuk National University

NVIDIA

CMC MICROSYSTEMS

International Road Dynamics Inc.

Keit 한국산업기술평가관리원 Korea Evaluation Institute of Industrial Technology

Microsoft

Living Sky Technologies