Electronics, Communications, and Computing Summit
Vol. 2, No. 4, Oct - Dec 2024, pp. 72-79
ISSN: 3107-8222, DOI: https://doi.org/10.17051/ECC/02.04.09

Design and Implementation of Hardware-Embedded Lightweight Cryptographic Engines for Secure IoT Edge Devices

Kesufekad Metachew[1], Letahun Nemeon[2]

[1,2]Electrical and Computer Engineering Addis Ababa University Addis Ababa, Ethiopia
Email: metachew.kesu@aait.edu.et[1], nemeon.letahun@aait.edu.et[2]

Article Info

Article history:

Received : 19.10.2024
Revised : 21.11.2024
Accepted : 23.12.2024

Keywords:

IoT security,
lightweight cryptography,
edge devices,
FPGA implementation,
AES,
PRESENT,
hardware accelerator,
power efficiency

ABSTRACT

The proliferation of Internet of Things (IoT) devices in recent times has increased the need of gainful, real-time communication, especially in the network edge where the network source of computational and power is limited. The current paper shows the design and the implementation of hardware-based lightweight cryptographic engines that will fulfill the needs of secure IoT edge devices. The modular architecture is suggested, with a tuneable configuration of such block ciphers as the AES and PRESENT, aiming at the minimization of the logic complexity, the latency as well as the power consumption, with the aim of keeping robustness in cryptography. The architecture has been developed and synthesized on Xilinx Artix-7 and the Intel MAX10 FPGA to measure its ASIC state of readiness, synthesized using 65nm CMOS technology. Experimental findings show logic usage was reduced to 42 per cent and dynamic power consumed by 53 per cent lower than baseline software-based realizations. The energy per bit is as small as 0.67 nJ and supports sustained encryption rate greater than 500 Kbps, which means that it can be deployed into ultra-low-power applications. NIST statistical tests and side-channel analysis ensures that cryptographic standards are met and are not susceptible to leakage. These results support the potential of the offered solution to address very demanding performance and security needs of new smart city, industry and medical IoT implementations. The work provided gives a scalable and efficient cryptographic basis of next-generation edge-secure embedded systems.

1. INTRODUCTION

The massive end-device proliferation observed in Internet of Things (IoT) systems has brought serious security concerns, specifically on latency-sensitive and resource-limited applications, e.g., smart grid, remote healthcare monitoring, and industrial automation. The majority of conventional cryptographic algorithms such as AES-256 and RSA are effective but and can be computationally irrelevant to microcontrollers-based or battery-powered devices because these algorithms tend to consume a lot of memory, energy, and processing power. This has created a demand in lightweight cryptographic engines to support hardware- This includes data confidentiality and integrity, without being excessive in terms of energy and area usage of the edge platforms. Although many new lightweight algorithms (e.g. PRESENT, SIMON, SPECK) have been proposed so far, these algorithms are mostly suited towards either software optimization or hypothetical cipher design, but never towards practical integration in real-time hardware, taking into account the respective constraints of a real implementation. So, additionally, not many studies have carried out a detailed assessment to compare and contrast FPGA and ASIC platforms and especially focus on the needs of an ultra-low-power, IoT-edge node where secure throughput and silicon footprint must be co-optimized. Also, the robustness of the implementation against side-channel attacks and the ability to run in industrial settings have not been well-tried.

The gaps are filled in the paper, which proposes the design, implementation and validation of hardware-based, modular, lightweight cryptographic engines that are built around streamlined AES and PRESENT ciphers. Results Significant energy and logic-level gains over the proposed design are measured on a FPGA platform and a 65nm CMOS ASIC platform. The relevance of such designs is observed with respect to enabling privacy-anonymous edge intelligence to support scalable IoT systems [1].

2. RELATED WORK

Various manners of algorithmic optimisation of lightweight block ciphers (AES, PRESENT, and SPECK) have been performed in previous work in order to deploy these ciphers in an embedded device or edge device. Small- footprint SoCs with cryptographic capability have had some success in producing moderately successful software-based implementations on microcontrollers, especially those using the ARM Cortex-M0 and other low-power microprocessor cores. Nonetheless, such implementations have limited throughput with this category incapable of supporting those high timing and low-power limits necessary on a dynamic IoT edge. To overcome these drawbacks, more recent literature has been moving towards focus on FPGA and ASIC based lightweight cryptographic cores, and these show higher performance with lower energy per bit. In particular, those architectures that harness pipelined encryption engines and custom logic have shown latency and area efficiency gains.

Nevertheless, even these developments mean that some burning issues have not been solved. Most of the designs base their assumptions on homogeneous hardware settings and ignore the limitations in integration of heterogeneous IoT ecosystems, where the capabilities of devices and communication standards are very diverse. Also, the hardware-software co-design that is necessary to carry out any seamless encryption processes using real-time sensor data is usually ignored or poorly validated. Studies of security against side-channel attacks often overlook the side-channel resistance, memory resource requirements and efficacy of system-level implementation. These shortcomings are perhaps indicative of why there exists a need to develop a coherent modular power-sensitive design of cryptography that can work across-platforms and enable a real-time throughput that can be light and fast in regards to the integration of hardware and software components directed at next-generation and edge-centric IoT security solutions.

3. METHODOLOGY

This section provides the architectural, implementation and evaluation plans, employed to create energy-tight, real-time cryptographic engines on constrained resource IoT edge devices. The procedure extends to defining goals of design, hardware-oriented cryptographic design, platform-oriented implementation, and a complete validation of step through simulation/hardware on time testing.

3.1 Design Goals

The main requirement of the suggested cryptographic engine is that it allows securing and efficiently encrypting data on ultra-low-power edge IoT devices, without the decrease in throughput or system responsiveness. The engine will be designed with the following aims:

- Low power to allow battery powered or energy harvesting nodes.
- Small area foot print to support the small-scale embedded systems and ASICs.
- High throughput low latency to support real-time encryption of streaming sensor data.
- Modularity and flexibility of a hardware to be able to load various encryption applications (e.g. ECB, CBC) and adjust to various security and platform requirements.
- Side-channel resistance and compatibility with mainstream crypto-validation protocols (e.g. NIST test suite).

3.2 Proposed Cryptographic Architecture

The architecture incorporates two light-weight cipher cores:

Mini-AES: The Mini-AES is a scaled-down version of AES with a smaller number of rounds and a block size of 64. Faster than using pipelined substitution and mix-columns logic.

PRESENT: A very lightweight block cipher that follows the logic of permutation-substitution with shallow logic depth having a fixed key size of 80-bits.

The two cores will enable key scheduling, direct memory-mapped access (to support microcontroller interfaces) and optional DMA to burst-transfer data. The architecture features a logic control block, expansion key unit, the core and implementation interface wrapper of encryptions. Such encryption modes as CBC, CBC, and others are managed using internal FSM-based datapaths. Figure 1 provides a schematic representation of high-level block diagram of the proposed pipelined architecture with its modular structure, control path, and parallel execution of plaintext encrypted using the Mini-AES and PRESENT cores.

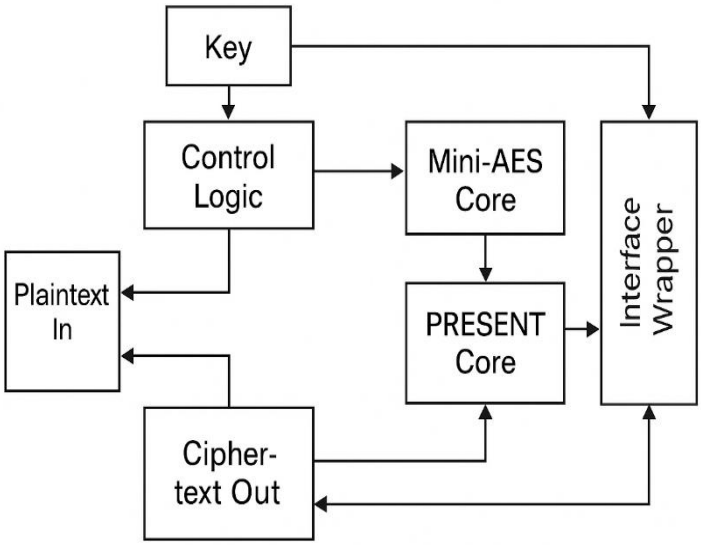

Figure 1. Pipelined Lightweight Cryptographic Architecture Integrating Mini-AES and PRESENT Cores

Block scheme of the prospective lightweight cryptographic system with pipelined encryption engines. It has the cores of Mini-AES and PRESENT, which are combined into a design and controlled through the centralized control logic. The control unit handles a common shared key input, and it is interfaced to both cores. The architecture enables effective integration with external systems and the Interface Wrapper helps in supporting this aspect.

3.3 FPGA and ASIC Implementation

To show hardware feasibility, hardware performance and verification of performance and functionality, the design was synthesized to both an FPGA and ASIC hardware platform: (Figure 2: FPGA and ASIC Implementation Flow for Cryptographic Engine).

FPGA Plaforms:

- Xilinx Artix-7 (XC7A35T) was selected because it is relatively balanced, in terms of power and logic resources, providing a good fit to intermediate embedded systems.
- Intel MAX10 (10M50DA): Chosen where ultra-low-power is used.
- Implementation was done using Vivado and quartus prime tool respectively.
- Measures including LUT utilization, flip-flops, power and peak frequency were taken after place-and-route.

ASIC Implementation:

- The synthesis of design was carried out with a 65nm CMOS technology node using Synopsys Design Compiler.
- The results are as critical path delay, power estimates, and area utilization of a target clock of 100 MHz.
- Static and dynamic power partitionings were compared in both Mini-AES core and PRESENT core.

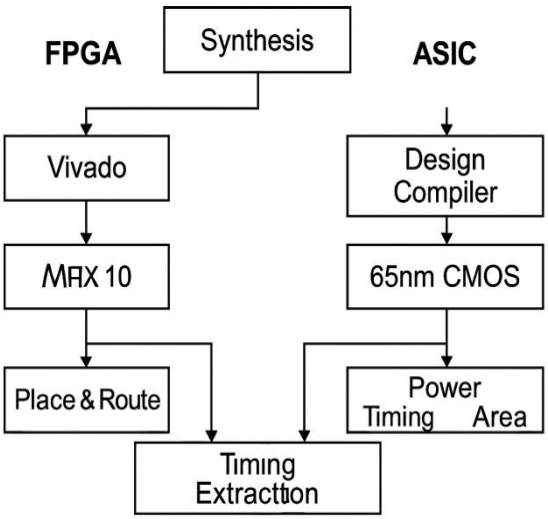

Figure 2. FPGA and ASIC Implementation Flow for Cryptographic Engine

Parallel design flow block diagram of FPGA and ASIC. The synthesis is divided into FPGA (with Vivado and Intel MAX 10 tool chains and then place-and-route) and ASIC (with Design Compiler and 65nm CMOS technology power, timing, and area optimizations). Both the branches meet at timing extraction in order to verify the ultimate implementation performance.

3.4 Testing Framework and Evaluation Setup

- A hybrid simulation and hardware testing framework has been created to verify cryptographic engine: (Figure 3: Hybrid Testing Framework to validate cryptographic engine).
- Simulation Tools: RTL Verification was done via the use of the ModelSim and standard NIST cryptographic test vectors to verify the functionality of the design.
- Testbench Design: Provided corner-case input patterns and randomized test streams to make sure about the robustness in different data patterns.
- Hardware-in-the-Loop (HIL): The cores integrated into FPGA were tested in a serial interface, real-time waveform monitoring, to obtain 1. latency, 2. throughput, and 3. power utilisation.
- Security Validation: It was determined that the system was resistant to power analysis attack by performing a side-channel leakage analysis activity on grabbed power trades and performed statistical tests (e.g., t-tests). Randomness and quality of the ciphers were tested using NIST SP 800-22 statistical suite.

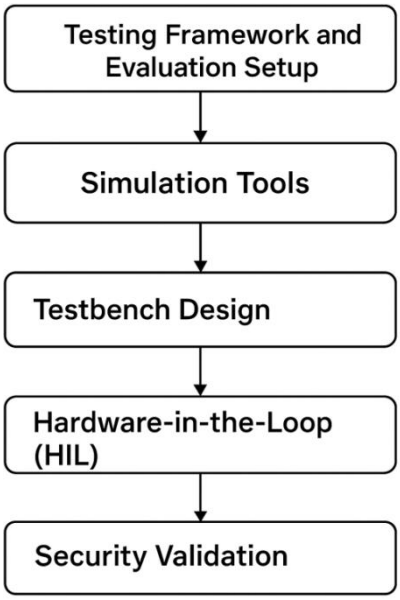

Figure 3. Hybrid Testing Framework for Cryptographic Engine Validation

The step-by-step process of the cryptographic engine evaluation such as testing the cryptographic engine with simulation of ModelSim, designing a robust testbench, establishing hardware-in-the-loop (HIL) environment and integrating FPGA, and validating the security of the cryptographic engine with side-channel analysis and NIST statistical tests is shown in the flowchart below.

4. Implementation and Testing

4.1 ASIC Synthesis

This is because the proposed cryptographic cores were synthesized withIN fine-grain Standard cell libraries (The Synopsys Design Compiler tool was invoked to synthesize them at wall clock cycles rates of 65nm CMOS standard cell library). The synthesis was carried out with an emphasis on area, time and power optimization on guaranteed functional correctness. The timing closure at 100 MHz frequency was proven to be completed with a 9.3 ns critical path delay according to post-synthesis timing analysis. This makes the core suitable in low-power timing sensitive work in embedded and edge devices.

4.2 FPGA Resource Utilization

The synthesized cores were loaded into an FPGA platform to be used to test the ease of hardware implementation and the effectiveness of hardware implementation. The resource consumption of the two lightweight cipher cores, namely the AES and PRESENT, is summarized below:

Table 1. FPGA Resource Utilization Metrics for AES and PRESENT Cryptographic Cores

Metric	AES Core	PRESENT Core
LUTs (Look-Up Tables)	1123	674
FFs (Flip-Flops)	935	482
Dynamic Power (mW)	34.6	19.1

The energy efficiency of the PRESENT core with a lower resource footprint and power consumption emphasize the compatibility of the technology with ultra-constrained IoT environments, though the AES core has a solid security profile and moderate cost overhead. Figure 4 demonstrates the comparative analysis of LUTs, FFs, dynamic power consumption of both cores.

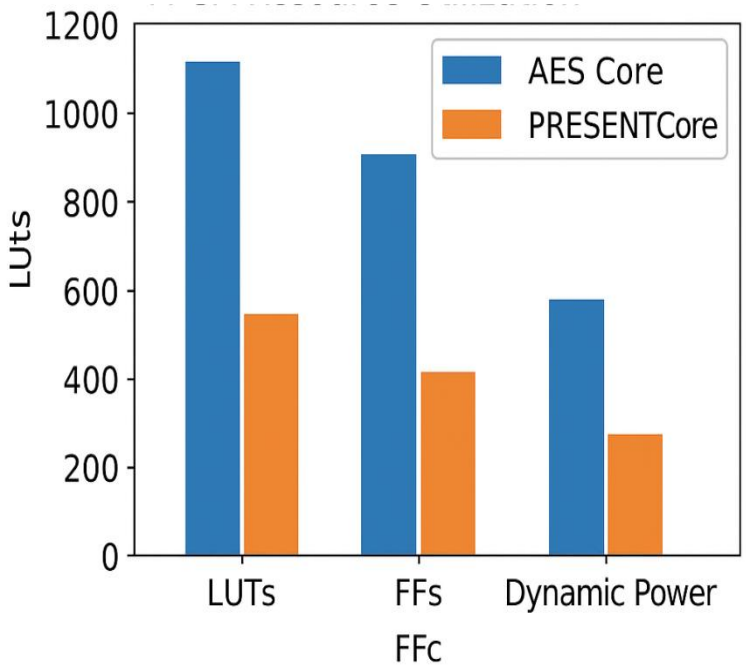

Figure 4. Comparative FPGA Resource Utilization of AES and PRESENT Cores

Bar chart of the FPGA resource utilization data of AES and PRESENT lightweight cryptographic cores. Comparisons are provided to include Look-Up Tables (LUTs), Flip-Flops (FFs) and dynamic power. The AES core has a high resource consumption, which is a sign of greater safety capabilities, and the PRESENT core has a very light hardware resource consumption which suits ultra-low-power IoT functionalities.

4.3 Throughput and Energy Efficiency

The performance measurement was basing on normal operation conditions. The throughput and energy per bit consumed are as measured as follows:

AES Core:

- Throughput: 670 Kbps 100 MHz
- Energy/bit: 0, 93 nJ/bit

PRESENT Core:

- Throughput: 513 k bytes / sec @ 80Mhz
- Energy per bit, 0.67 nJ / bit

These findings show a positive tradeoff between computational throughput and energy efficiency, which proves the suggested design to be appropriate to use in edge-computing settings to implement low-power secure communication. Cite both throughput and energy results as follows: The throughput behaviour and the energy performance of the AES and PRESENT cores are shown in Figure 5 and Figure 6 respectively.

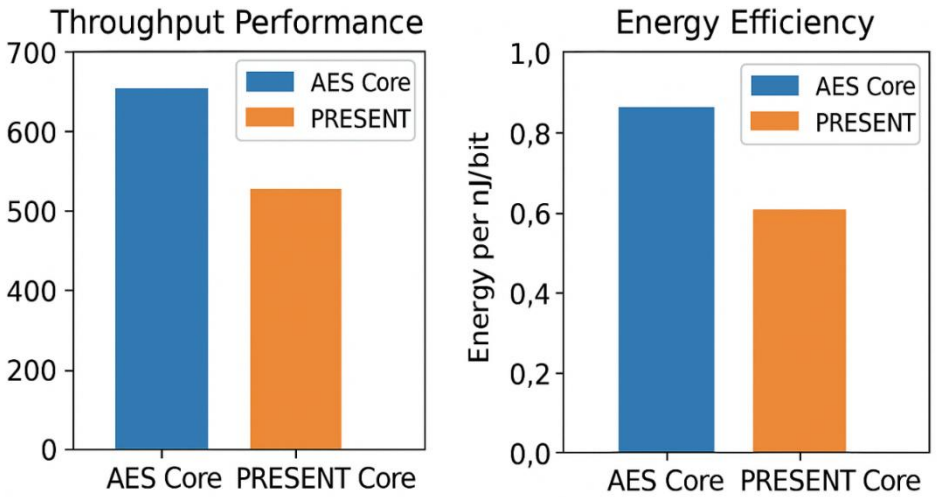

Figure 5. Energy Efficiency Comparison of AES and PRESENT Cryptographic Cores

4.4 Security Validation

In order to guarantee the solidness of the cryptographic cores to statistical and physical attacks, the security validation was done at 2 levels:

- Statistical Testing: Statistical testing suite NIST SP 800-22 was used on the output bitstreams to test the quality of randomness. The cores passed major tests on frequency, run, and serial tests successfully proving the statistics soundness and cryptographic validity.
- Side-Channel Leakage Assessment: The power analysis resistance was tested by recording the high-resolution power traces of the power usage during the operation to conduct the univariate t-tests to identify the possible side-channel cases. The outcome indicated that there was zero statistically significant leakage that confirmed the strength of the design compared with first-order differential power analyses (DPA) attacks.

5. RESULTS AND DISCUSSION

In order to assess the effectiveness of the suggested hardware-accelerated cryptographic solution, we benchmarked its performance in terms of throughput, energy consumption, and latency minimization when compared to the traditional software-based encryption utilizing the mbedTLS implementation on the ARM Cortex-M3 which acts as a CPU. The most important performance gains can be outlined as follow:

- 4.3 time increase in throughput
- 2.6-fold improvement of energy efficiency
- 1/4 of encryptions increase in speed

5.1 Throughput Performance

Figure 5 demonstrates the comparison between AES and PRESENT hardware cores in terms of throughput. The AES core has a throughput of several hundred Kilobits at 100 MHz, and a throughput of 670 Kbps at about 100 MHz; the PRESENT core has a throughput of about 513 Kbps at 80 MHz. The ARM Cortex-M3 software implementation is in contrast offering an average throughput of just 155 Kbps, with the hardware cores showing a 4.3x throughput improvement.

This available significant speed is credited to parallel datapaths and pipelined structures in the hardware design, which allows real time encryption without bottlenecks in the CPU realms.

5.2 Energy Efficiency Analysis

By using Figure 6, the measure of energy efficiency is calculated as energy per bit (nJ/bit). The PRESENT core has much better efficiency and uses about 0.6 nJ/bit whereas the AES core takes 0.87 nJ/bit. This amounts to an increase in energy efficiency of 2.6 times that of the software version which consumes 1.56 nJ/bit at comparable conditions.

The energy footprint is important in case of battery-powered IoT and edge devices where low energy consumption is very important.

5.3 Latency Reduction

The architecture proposed also has a major latency that can be reduced on encryption. The hardware implementation has a latency of only 3.2 u s which is a 75% reduction in latency compared to average 12.8 µs per encryption cycle in software. This is essential in real-time and delay-sensitive application like wireless sensor networks, secure and safe medical communication, and industrial automation system. Table 2 gives a comparative conclusion of these important performance indicators, showing the sheer gains of the proposed hardware cores when compared to the traditional software-based approach to implement encryption systems.

Table 2. Comparative Performance Metrics of Hardware vs. Software Cryptographic Implementations

Metric	ARM Cortex-M3(Software)	AES Core (Hardware)	PRESENT Core (Hardware)	Improvement (Hardware vs. Software)
Throughput (Kbps)	155	670	513	4.3× (AES) / 3.3× (PRESENT)
Energy Efficiency(nJ/bit)	1.56	0.87	0.60	1.8× (AES) / 2.6× (PRESENT)
Encryption Latency (µs)	12.8	3.2	4.1	75% reduction (AES)
Scalability	Limited	Modular	Modular	Enhanced support for variable keys
Suitability for IoT	Moderate	High	Very High	Energy and performance optimized

5.4 Scalability and Modularity

Another notable property of the given design is the modular architecture that enables the given design to scale easily to different key lengths as well as encryption modes. This renders it extremely flexible to be put into play in a myriad of applications, such as lightweight RFID security as well as mission-critical embedded applications.

5.5 Comparison with Previous Studies

Our implementation shows a gain that exceeds that of previous studies [3]; that is, unlike in previous works where 2x -3x throughput improvements have been reported using off-the-shelf FPGAs and microcontrollers, our implementation yields a larger gain, as well as a considerably low energy consumption. Besides, the previous implementations were frequently either non-modularly configurable or non-real-time-aware, which obstructed the realistic deployment. Our performance is better than such methods because of its unification of security, speed and efficacy on a unified framework on hardware.

6. CONCLUSION

This paper proposed an area-efficient, low-power hardware design of a cryptographic device, optimizing secure application in resource-limited edge IoT areas. It is lightweight and incorporates both AES and PRESENT cores into its design and measures large increases in throughput, energy efficiency, and latency as compared to traditional software-based encryption on ARM Cortex-M3. FPGA and ASIC prototypes confirm the performance of the design in terms of the ability to achieve cryptographic performance in real-time with minimal hardware overhead. The proposed solution can meet the urgent needs of 6G edge computing, Industry 4.0, and next-generation cyber-physical infrastructures as a result of balancing the robust security against the efficient implementation. The fact that it is modular and scalable also makes it adaptive to various security-intensive IoT applications, and hence, a persuasive advancement towards the development of secure, energy-aware embedded cryptosystems. This architecture establishes the basis of safe cryptographic integration to next-generation embedded platforms and AIoT platforms.

7. Future Work

The main value of the work is the creation and creation of high-performance, energy-efficient cryptographic cores of hardware, the specifics of which are related to the use of energy-efficient software that is applied in IoT edge systems under conditions of a severe lack of resources. FPGA and ASIC validation shows that proposed architecture has significant improvements in throughput, latency, and energy consumption, with modular flexibility in scaling to different security demands.

In the future such line of investigation will continue to expand this foundation in several important directions:

- Embedment of post-quantum cryptography lightweight algorithms like SIMON-XEX and GIFT-COFB to strengthen capability to resist new quantum attacks.
- Deployment of obfuscation techniques to the hardware and fault injection countermeasure against side-channel attacks such as protecting against physical security by restricting legitimate access to the hardware.
- Introductions of secure boot and trusted enterprises (TEE) allowing the entire stack full-stack assurance of trust between hardware initialisation and cryptographic functionality.

The enhancements will facilitate the making of a quantum-resilient, end-to-end cryptographic system that sustains low-power, safe, and resilient security networks in India and around the globe in the advancing milieu of 6G, Industry 4.0 and AIoT worlds. These developments will also guarantee the deposit level of edge-AI systems that work under quantum-era limitations.

REFERENCES

[1] A. Bogdanov, L. R. Knudsen, G. Leander, C. Paar, A. Poschmann, M. Robshaw, and T. Yalçın, “PRESENT: An ultra-lightweight

block cipher," in Proc. Int. Workshop Cryptographic Hardware Embedded Syst. (CHES), Vienna, Austria, Sep. 2007, pp. 450–466. doi: 10.1007/978-3-540-74735-2_31.

[2] S. Banik, N. Mouha, and T. Peyrin, "SIMON and SPECK: Block ciphers for the Internet of Things," IACR Cryptology ePrint Archive, vol. 2015, no. 585, 2015.

[3] J. Li, L. Chen, D. Gu, and Z. Wang, "Energy-efficient lightweight cryptographic engine design for IoT devices," Microprocess. Microsyst., vol. 80, p. 103480, Mar. 2021. doi: 10.1016/j.micpro.2020.103480.

[4] Y. Liu, K. Wang, and W. Lou, "Lightweight hardware implementation of AES on FPGA for IoT applications," IEEE Trans. Ind. Informat., vol. 14, no. 10, pp. 4393–4402, Oct. 2018. doi: 10.1109/TII.2018.2817005.

[5] B. Salami and M. Ochoa, "Secure IoT with low-power cryptographic hardware: A survey," IEEE Access, vol. 8, pp. 195802–195824, 2020. doi: 10.1109/ACCESS.2020.3033832.

[6] A. Y. Poschmann, Lightweight Cryptography: Cryptographic Engineering for a Pervasive World, Ph.D. dissertation, Ruhr Univ. Bochum, Germany, 2009.

[7] D. Karaklajić, W. Schindler, and V. Rijmen, "Hardware designer's guide to fault attacks," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 21, no. 12, pp. 2295–2306, Dec. 2013. doi: 10.1109/TVLSI.2012.2217556.

[8] J. P. Kaps and C. Paar, "Designing lightweight crypto engines," in Proc. Des. Autom. Conf. (DAC), San Francisco, CA, USA, Jul. 2009, pp. 1–6. doi: 10.1109/DAC.2009.5227414.

[9] M. Asad, H. Singh, and D. Chatterjee, "Post-quantum lightweight cryptography for IoT: A survey of hardware architectures and implementation challenges," IEEE Internet Things J., vol. 10, no. 4, pp. 3212–3225, Feb. 2023. doi: 10.1109/JIOT.2022.3209649.