Assessment of Radiation-Induced Soft Errors on Lightweight Cryptography Algorithms Running on a Resource-constrained Device

Jonas Gava, Nicolas Moura, Joaquim Lucena, Vinicius Rocha, Rafael Garibotti, Ney Calazans, Sergio Cuenca-Asensi, Rodrigo Possamai Bastos, Ricardo Reis, and Luciano Ost

***Abstract*—Most safety-critical edge-computing devices rely on lightweight cryptography (LWC) algorithms to provide security at minimum power and performance overhead. LWC algorithms are traditionally embedded as a hardware component, but with the advance of the Internet of Things (IoT), emerging firmware is more likely to support cryptography algorithms to comply with different security levels and industry-standards. This is the first work to present the soft error assessment of five cryptography algorithms executing in a low-power microprocessor running under neutron radiation, considering electronic code book (ECB) and counter (CTR) mode of operation implementations. Results obtained from two neutron radiation tests suggest that: (*i*) the NOEKEON algorithm gives the best relative soft error reliability, performance, power efficiency and memory footprint utilisation trade-offs between the five algorithms considering both ECB and CTR implementations, and (*ii*) cryptography solutions based on the counter mode of operation present better FIT rate for silent data corruption (SDC) and crash w.r.t. ECB implementations.**

***Index Terms*—Cryptography Algorithms, Mode of Operation, Neutron Radiation, Low-power Microprocessor.**

I. Introduction

With the advance of 5G technologies, a wide variety of edge solutions will emerge as part of the industrial Internet of Things (IoT). The proliferation of edge devices will have a profound impact across a variety of industrial sectors and society as well [1]. Although performing computation at the edge improves latency and privacy issues [2], edge devices still acquire first-hand sensitive information from users (e.g., financial, medical details), which might cause damage if revealed [3]. In 2017, St. Jude Medical recalled 465k pacemakers due to security vulnerabilities, including, among others, the unencrypted patient information transmission via radio-frequency (RF) communication [3]. To avoid similar security risks, edge devices are expected to implement bespoke protections for safety-critical applications and employ security mechanisms such as continuous monitoring and encryption of sensitive data. The latter is achieved through the adoption of advanced lightweight cryptography (LWC) algorithms, which must be performed at the edge by low-power microprocessors or even on the sensors themselves.

The deployment of cryptography algorithms in resource-constrained devices is challenging, since it must consider the trade-off between security level, power efficiency, memory footprint, and response time, a premium metric for real-time applications. Standardised cryptography algorithms undergo extensive security criteria analysis (e.g., selection of appropriate mode of operation [4]) and their adoption depends on the target application's criticality. For instance, emerging autonomous vehicle sensors are expected to execute machine learning inference models for object recognition and decision-making within multi-dimensional environments. Such systems are likely to employ an encryption algorithm for critical data acquisition and/or transmission (e.g., image transfer) [5]. The resulting scenario calls for lightweight and reliable cryptography algorithms capable of maintaining efficient performance, a reasonable response time and protecting a resource-constrained system against the occurrence of radiation-induced soft errors. While the soft error susceptibility of cryptography implementations on ASIC and SRAM-based FPGAs are highly explored [6], the impact of soft errors on software solutions running on microprocessors is still an open question.

In this scenario, the main *contribution* of this work is the soft error reliability assessment of five lightweight cryptography algorithms running on an Arm Cortex-M4 microprocessor under neutron radiation. The other contributions of this work are the following:

- Comprehensive relative reliability, power efficiency, performance and memory utilisation trade-off analysis of all cryptography algorithms deployed in the target microprocessor, which is highly used in smart sensor systems [7];
- Assess the impact of using different plaintext sizes on the soft error reliability of a standardised embedded cryptography algorithm, widely used worldwide;
- Investigate the soft error resilience of two modes of operation of embedded lightweight cryptography algorithms running on an Arm Cortex-M4 microprocessor under radiation effects.

This work was partially funded by: CAPES; CNPq (grants 317087/2021-5 and 407477/2022-5); FAPERGS (grant no. 22/2551-0000570-5); MultiRad (PAI project funded by Région Auvergne-Rhône-Alpes); IRT Nanoelec (ANR-10-AIRT-05 project funded by French PIA); UGA/LPSC/GENESIS platform; and PID2019-106455GB-C22 (funded by the Spanish Ministry of Science and Innovation).

J. Gava and R. Reis are with PGMicro, UFRGS, Brazil (e-mail: {jfgava, reis}@inf.ufrgs.br).

N. Moura, J. Lucena, V. da Rocha, R. Garibotti and N. Calazans are with School of Technology, PUCRS, Brazil (e-mail: {nicolas.moura, joaquim.lucena, vinicius.rocha97, rafael.garibotti, ney.calazans}@pucrs.br).

S. Cuenca-Asensi is with Universidad de Alicante, Spain. (e-mail: sergio@dtic.ua.es).

R. Possamai Bastos is with TIMA, Université Grenoble Alpes, France. (e-mail: rodrigo.bastos@univ-grenoble-alpes.fr).

L. Ost is with Loughborough University, UK (e-mail: l.ost@lboro.ac.uk).

Corresponding author: Luciano Ost (e-mail: l.ost@lboro.ac.uk).

Manuscript received October 14, 2022.

The rest of this paper is organised as follows. Section II presents related works covering the reliability of cryptography algorithms and their modes of operation. Section III describes the five cryptography algorithms successfully executed in the reference device under neutron radiation. Next, Section IV presents the radiation test flow and set-up used to assess the soft error reliability of adopted cryptography algorithms. Sections V and VI discuss the performance and soft error reliability results considering cryptography algorithms and their modes of operation. Finally, Section VII points out conclusions.

II. Related Work in Soft Error Reliability of Lightweight Cryptography Algorithms

LWC is a quite active field of research and applications, fuelled by the rising importance of edge computing, as evidenced in different surveys [8]–[10]. For instance, Mohd *et al.* [9] present forty-seven LWC dedicated to low-resource devices, while Thakor *et al.* [10] reviewed fifty-two LWC algorithms and mentioned that another fifty-seven new LWC algorithm proposals were submitted in a recent NIST competition.

Given the sheer abundance of LWC algorithms, their comparison and evaluation become a Herculean work. It goes from selecting the parameters to compare, choosing an adequate algorithm subset for the application classes, and considering items such as block and key size variations, modes of operation available, considering implementations in hardware, software or both, etc [11]–[14]. Besides the obviously relevant parameters to take into account, such as silicon area and power consumption in hardware, memory footprint in software and latency and throughput in both forms of implementing LWC algorithms, there are several other considerations for selected fields of application. Examples are the degree of security [15] and the reliability to soft errors [16], [17].

Dedicated hardware implementations of lightweight cryptography (LWC) algorithms have been thoroughly investigated in both ASIC [6], [18] and FPGA [19], [20]. The underlying hardware solutions are susceptible to radiation-induced soft errors, i.e., a bit flip caused by a soft error can affect the encrypted message and cause problems for the communication protocol. In this sense, some works have started to assess the impact of the radiation-induced soft error on embedded (i.e. hardware-based implementations) LWC algorithms. For instance, Bandeira *et al.* [17] present the relative performance, area, and soft error reliability trade-off of XTEA and AES cryptographic algorithms under radiation-induced effects. Similarly, Dutertre *et al.* [21] and Roscian *et al.* [22] conducted laser-induced fault experiments to assess the reliability of AES implementations. Authors also investigated different mitigation techniques and resilience implementations aiming to reduce the impact of soft errors on cryptography solutions implemented on FPGAs [23]–[25].

Except for our pioneering work [16], which uses the SOFIA environment [26] to assess the soft error reliability of ten software-based LWC algorithms, this is the only work considering the execution of LWC algorithms in a resource-constrained device. This work distinguishes from the previous works in two main aspects:

- First, this is the first work to assess the soft error susceptibility of LWC algorithms considering their execution on a resource-bound microprocessor under several radiation exposure hours;
- Second, for the first time, a work investigates the effect of modes of operation on the response of a set of LWC ciphers to radiation.

III. Cryptography Algorithms

This Section starts by covering the process of selecting the ciphers to address in this work in Section III-A. Section III-B justifies the choice of modes of operation to employ in case studies. Lastly, Section III-C analyses the performance and security of the chosen LWC algorithms considering their modes of operation.

A. Cryptography Algorithm Selection

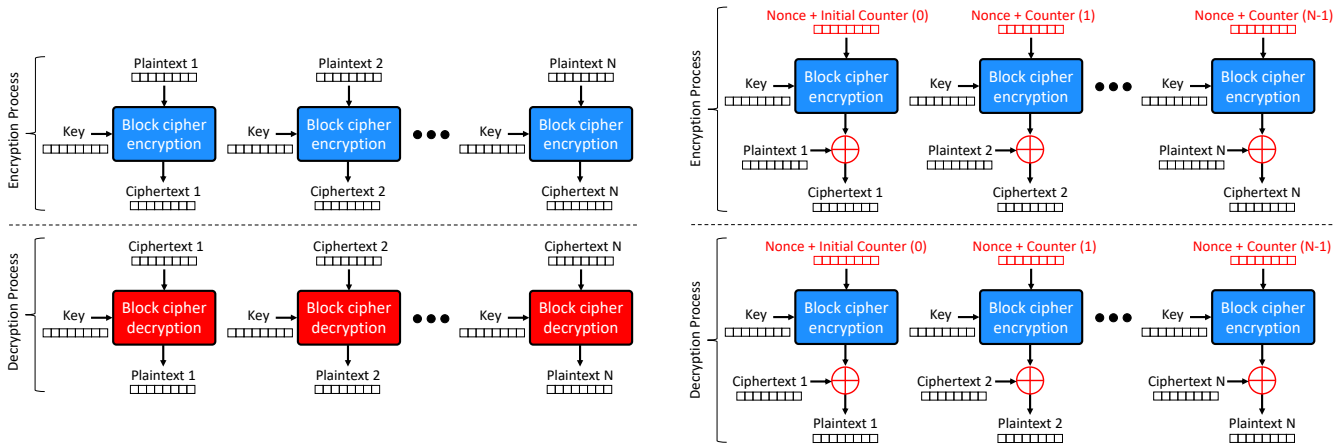
This work addresses five lightweight cryptography algorithms: *AES* [27], *ARIA* [28], *IDEA* [29], *NOEKEON* [30] and *SEED* [31]. The adopted LWC algorithms are of the symmetric type, providing better performance at a low memory footprint. Also, symmetric ciphers generally consume less power than asymmetric algorithms, such power-efficient attributes are crucial for the LWC algorithms' choice as these are the most sought-after features in upcoming embedded systems [32]. Furthermore, the selected LWC algorithms are commonly used in software-based cryptographic systems [13], which qualifies them as natural candidates to be massively used in IoT devices [17].

All addressed LWC algorithms have a fixed block size of 64 or 128 bits and up to three key length settings: 128, 192, and 256 bits. Here, all algorithms consider fixed plaintext and 128-bit keys. The resulting setting makes all LWC algorithms comparable in terms of data throughput and security.

B. Modes of Operation

Symmetric key algorithms are divided into block ciphers and stream ciphers. A block cipher is recommended for static data, where the data size is already known and can be divided by block lengths to compute how many rounds are needed to finish the encryption process. On the other hand, stream ciphers encrypt information by processing individual bits or words, while block ciphers always work over a fixed-size set of bits, usually 64 or 128 bits. A vulnerability of block ciphers is that the same entry will always generate the same output given a key, creating patterns that can be used to uncover the key. With this in mind, NIST (SP 800-38A) proposes different modes of operation to mitigate such a problem [4].

This work focuses on case studies only for the *Electronic code book* (ECB) and *Counter* (CTR) modes of operation because just these two modes work with text blocks independently of the previous ones. So these two modes are able to display errors on a block-by-block basis. If an error is found and its impact affects other blocks' processing, an error at the beginning of the process can impact all subsequent results, making soft error analysis difficult or even unfeasible.



(a) ECB – Electronic code book mode

(b) CTR – Counter mode

Fig. 1. Block diagram of ECB and CTR modes of operation.

Figure 1 shows the absence of correlation between blocks in the selected modes of operation, while illustrating the differences between these two modes, as highlighted in red. On the one hand, the ECB mode relies on a block cipher decryption. On the other hand, the CTR mode changes the encryption block entry from plaintext to ciphertext by XORing it to the ciphered Nonce[1]+Counter combination, increasing the algorithm's security. This, in fact, prevents blocks with the same structure from generating the same encrypted output.

C. Performance and Security Analysis

Figure 2 relates the performance of the five LWC algorithms implemented according to two modes of operation. Each bar represents a mode of operation (ECB and CTR) associated with the y-axis, which is the LWC algorithm execution time on a resource-constrained device (Arm Cortex-M4 board).

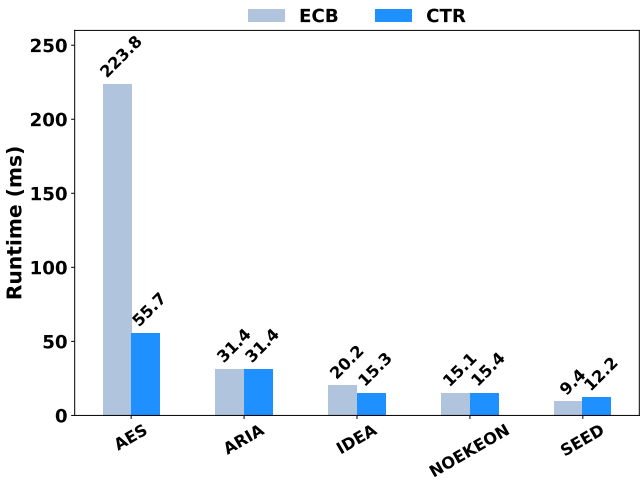


Fig. 2. Performance analysis of the five LWC algorithms.

Note that the execution time corresponds to the full algorithm execution cycle, which includes both the encryption and decryption processes. The results show that AES has the most significant performance gain when using the CTR mode. This is because AES block cipher decryption (ECB mode) takes longer to finish than block cipher encryption. The CTR mode only uses encryption in its decryption process (see Figure 1.b), which eliminates the AES block cipher decryption execution time cost. The CTR implementations of IDEA and ARIA also incur better performance w.r.t. their ECB mode versions. In the reference implementation, both have an extra function called `Generatedecryptionkeys()`, which makes the decryption process slower than the encryption process. However, they are lighter than AES, hence this smaller difference between the two modes of operation. For NOEKEON, the execution time was very similar for both modes of operation. On the one hand, the block cipher decryption calls the function `Theta()` twice in ECB mode. On the other hand, CTR mode has an extra XOR operation. Lastly, SEED has the same runtime for both block ciphers (encryption and decryption), making ECB faster than CTR mode as it does not have the extra XOR operation cost.

To analyse the security provided by the modes of operation, a parallel can be made with the birthday paradox [33], where the days of the year (i.e., category size) would be the plaintext length for ECB mode. In this context, the ECB mode security is precisely the plaintext length because the same input will generate the same output. In this regard, the probability of having duplicate values increases as the number of blocks to be encrypted increases, facilitating an attack on this mode of operation. On the other hand, CTR mode generates different outputs for the same plaintext thanks to the counter (see Figure 1.b), reinforcing the security of the cryptography algorithm regardless of the number of processed plaintexts.

IV. Adopted Radiation Test Methodologies

This Section describes the methodology used to collect and show the results obtained with a 14-MeV neutron generator, which has been used to assess the soft error reliability of five cryptography algorithms implemented according to two different modes of operation executing on a microprocessor under neutron radiation.

[1]A *cryptographic nonce* is a random or pseudo-random number used here as a prefix to a plain counter. A nonce is used only once with a given key.

A. Radiation Test Flow

Figure 3 shows the adopted radiation test flow schematic. First, the flux is calibrated remotely to fit a proper operation for the device under test (DUT), which is connected to a control computer (CC) outside the radioactive chamber through a USB cable. Then, the Universal Asynchronous Receiver Transmitter (UART)-based communication between the DUT and the CC is verified via checkers (i.e., checksum) to isolate radiation-induced failures in the UART peripheral and the data communication channel between the DUT and the CC. Note that the output checker is highlighted in Figure 3.

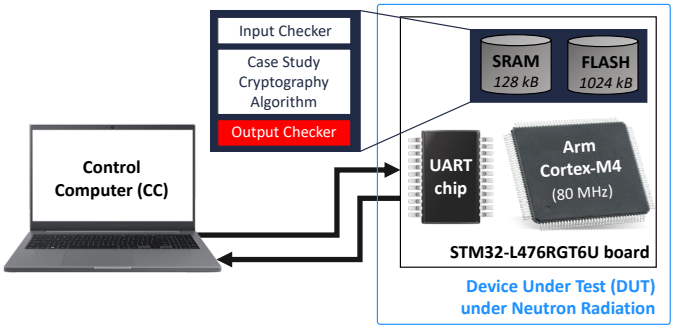


Fig. 3. The radiation test flow schematic, highlighting the output checker included in the DUT (*extended version*) to improve observability.

In the *original version*, this output checker did not exist, making it impossible to distinguish between faults inherent to the algorithm execution and faults resulting from the communication channel between the DUT and the CC. On the other hand, in the *extended version*, the output checker was included in the DUT to improve observability, making it possible to identify faults in the algorithms' execution and the ciphertext communication.

The steps to run the radiation tests are shown below:

1) Board programming using the Open On-Chip Debugger (openocd);
2) Synchronise both DUT and CC devices before the algorithm main function execution, i.e., send a message from CC to DUT and awaits the correct response. If the DUT takes more than 5 seconds to respond, the board is then reprogrammed;
3) Check the input checksum. The checksum is calculated in the DUT and sent to the CC for checking against the golden reference. If there is a mismatch in the input data, the board is reprogrammed. If the reprogramming fails, the relay is actioned, and a power cycle is done;
4) Algorithm main function execution;
5) Synchronise DUT and CC after the algorithm computation, i.e., send a message from CC to DUT and awaits the correct response;
6) Send output from DUT to CC;
7) (*extended version* only) The output checksum is calculated in the DUT and sent to CC. If there is a mismatch with the output, a fault can be observed (i.e., identified) in the output communication.

Note that the developed setup enables to isolate the algorithm execution as well as eliminate possible sources of errors inherent to the test environment, which facilitates the results analysis. While the communication time and control takes up to 20ms in total for each run, the checksum computation requires less than 1ms.

B. Radiation Test Set-Up

Two 14-MeV neutron radiation test campaigns were performed at the Laboratory of Subatomic Physics & Cosmology (LPSC, Grenoble, France), the first in February 2022 and the second in July 2022. The two radiation test campaigns used the GENEPI2 (GEnerator of NEutrons Pulsed Intense) neutron source, a neutron generator that delivers 14-MeV neutron beam with a maximum flux that exceeds the natural 14-MeV neutron flux at 40,000 ft by a factor of 10^{10}. Note that a total fluence $\geq 9.1 \times 10^{10}$ $neutron/cm^2$ was chosen to achieve statistical significance for the experiment by accumulating event counts from multiple runs. The average flux during the experiment in February 2022 was 3.5×10^6 neutron/cm^2/s, while the average flux during the radiation tests in July 2022 was 7.4×10^6 neutron/cm^2/s.

Figure 4 illustrates the set-up assembled at the LPSC. The STM32-L476RGT6U board [34] was selected as the target resource-constrained device to execute the five cryptography algorithms with two modes of operation, which were compiled using GCC 9.3.1 with O1 optimisation level. For radiation tests performed in February 2022, the DUT was placed in the second boards row of the neutron generator at a distance of 155 mm, as highlighted in Figure 4 (right above). On the other hand, for radiation tests performed in July 2022, the DUT was placed in the first boards row of the neutron source. However, in both cases, the flux were calibrated remotely to fit the proper operation of the DUTs, which have been connected to a CC outside the radioactive chamber through a USB cable. Note that the whole system (CPU, memory, communication peripherals) is under the beam.

C. Adopted Fault Classification and Reliability Metrics

This work classifies the radiation results into two different event types. The first event is *SDC*, where the algorithm execution normally occurs without error indication. However, there is a mismatch between the encrypted output message and its golden reference. The second event is a *crash*, where the algorithm suffers from abnormal termination or application hang. The communication between CC and DUT is lost during the algorithm execution, indicating that radiation effects have upset the DUT. In this situation, the board must be restarted.

Three metrics are used to assess the soft error reliability of the adopted cryptography algorithms. The *Failure in Time* (FIT) metric shows how many failures occur in a billion hours. It depends on both the device sensitivity and the particle flux to which it will be exposed. The specification that defines standard requirements and procedures for terrestrial soft-error-rate testing of integrated circuits and reporting of results (JEDEC) suggests to uses $13n/cm^2/h$ as flux at sea level [35]. The second metric adopted in this work is the *Mean Time Between Failures* (MTBF), which divides the number of SDCs and crashes by the board radiation exposure time. Lastly, this

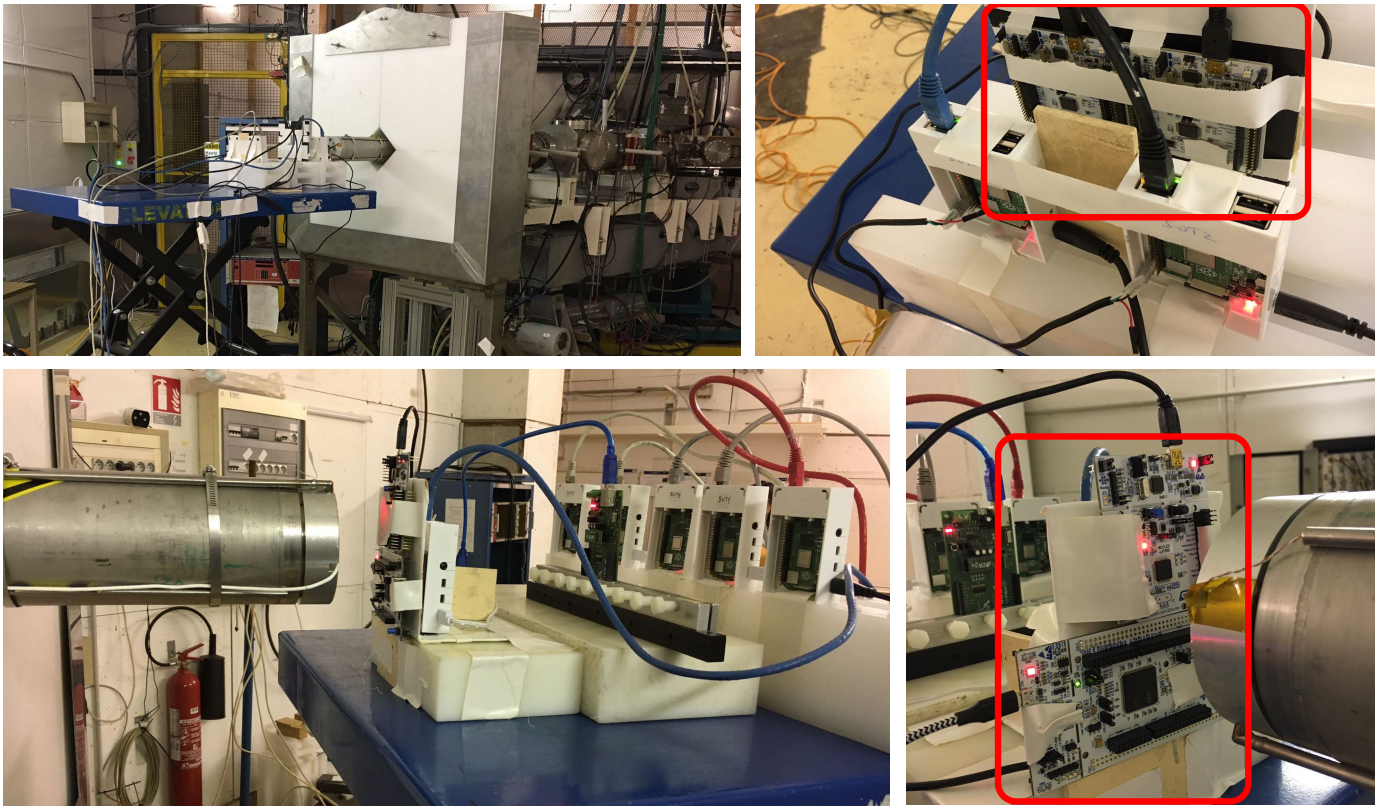


Fig. 4. SUTs mounted at the LPSC facility in February 2022 (top photos) and July 2022 (bottom photos).

work also uses the *Mean Work to Failure* (MWTF), which captures the trade-off between reliability improvement and runtime overhead. Equation (1) shows how it is calculated, where the σ symbol stands for cross section.

$$MWTF = \frac{1}{(\sigma \times flux \times execution\ time)} \quad (1)$$

V. Radiation Results: Case-Study A

This Section explores the soft error reliability of five LWC algorithms with ECB mode under neutron radiation. In this sense, Section V-A details the ECB mode results using the original version of the radiation test flow obtained from radiation test campaigns conducted in February 2022. Section V-B analyses a larger plaintext input's impact on the soft error reliability using the extended version of the radiation test flow assessed in July 2022. Finally, Section V-C analyses the relative soft error reliability, performance, power efficiency and memory footprint trade-offs for the experiments conducted in February 2022.

A. ECB Mode Radiation Results

Figure 5 shows the ratio of radiation-induced event types considering the five cryptography algorithms with ECB mode. Results show that crashes are higher than SDCs in all cases. This occurs due to the communication and control errors resulting from the entire board's exposure. While AES presents an SDC ratio of 43% (worst-case scenario), ARIA and SEED algorithms showed an SDC ratio of 25%. However, to make a fair comparison, it is necessary to relate the events to the radiation exposure time. Figure 6 shows the radiation test results regarding the FIT metric for all cryptography algorithms, which captures the relationship between radiation exposure time and the number of events. Each bar represents a FIT rate value for SDC (grey) and crash (blue).

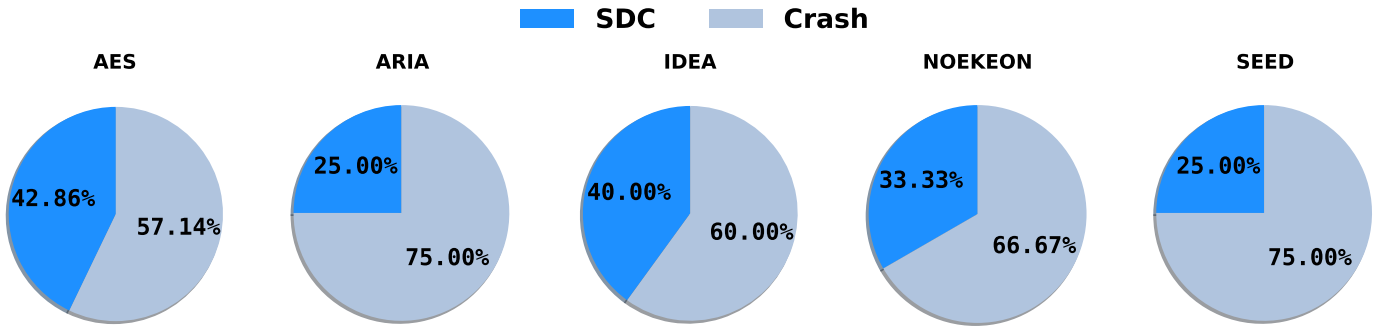


Fig. 5. Ratio of radiation-induced event types observed in each ECB mode case-study during the radiation test campaigns conducted in February 2022.

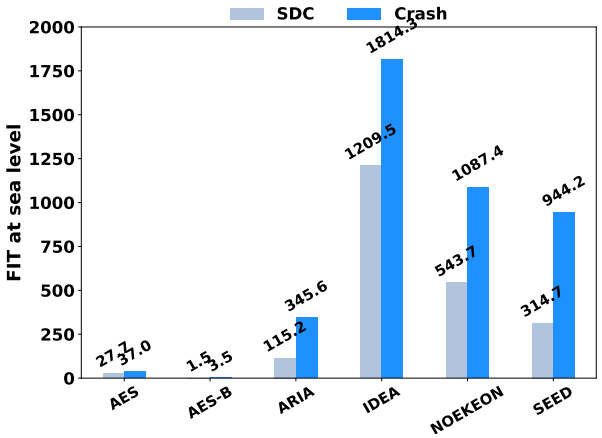


Fig. 6. Radiation-induced failures considering the FIT metric for the five ECB mode cryptography algorithms.

As the FIT rate uses the radiation exposure time, it is possible to determine which algorithm is the most reliable when executed in the adopted resource-constrained device. The IDEA algorithm had the worst result with more than 1200 FIT_{SDC}, while AES, with the highest number of SDC events, had only 27 FIT_{SDC}. Also, the FIT_{crash} rate ranges from 37 in the best case (AES) to 1814 in the worst case (IDEA). The FIT results suggest that AES has the best resilience against SDCs and crashes for the ECB mode.

Table I shows a summary of data collected from case study A. The number of runs for each cryptography algorithm ranged from 120k to 130k. On the other hand, an effective fluence ranging from 0.21×10^8 neutrons/cm^2 to 14.00×10^8 neutrons/cm^2 is observed even with a tiny difference in the number of runs. This occurs due to the execution time of each run, which varies from 0.05ms to 3.07ms. Some LWC algorithms run so fast that the time needed to control each run exceeds the algorithm computation time dozens of times. Note that the execution time of the LWC algorithms has a direct impact on the energy consumption, which was measured by a USB current tester [36] and shown Table I. For this reason, metrics that consider reliability and exposure time are necessary to make a fair comparison. Table I also shows the MTBF metric values considering both SDC and Crash occurrences. Regarding the SDC events, there is a variation from 1.8 million hours (IDEA) to 77.9 million hours (AES) per failure when scaling to the level of terrestrial radiation flux. In turn, for crash events, the metric values range from 1.2 million hours (IDEA) to 58.4 million hours (AES).

This first neutron radiation test campaign show that AES and IDEA algorithms have significantly different behaviours. To understand the architectural difference of the two that made them have the biggest variances in terms of soft error reliability, Figure 7 shows the dynamic executed instructions for both algorithms classified according to the following three categories: memory access, control, and data processing.

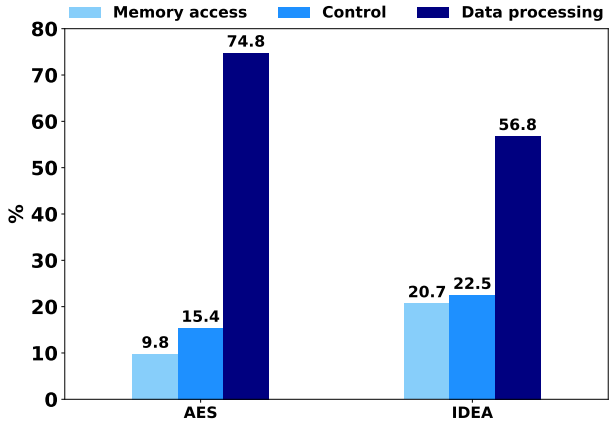


Fig. 7. Dynamic executed instructions for the AES and IDEA algorithms.

Note that AES spends almost 75% of the time executing data processing instructions, while IDEA has more than 10.9% of memory access and 7.1% of control instructions w.r.t. AES, which may explain the increase in crashes due to wrong branches and illegal memory accesses.

B. The Plaintext Size Impact

Due to the prominence of the AES results, a more in-depth analysis was performed, investigating the impact of the plaintext input size on the soft error reliability in this cryptographic algorithm. In the February 2022 radiation campaign, a simple plaintext input with a 128-bits size was used. This caused an algorithm execution time of a few milliseconds. On the other hand, the communication and control time takes much longer. This, in addition to the original version of the radiation test flow where it was impossible to distinguish faults between computation and ciphertext communication, can lead to inaccurate radiation results. For the July 2022

TABLE I
SUMMARY OF PERFORMANCE AND NEUTRON RADIATION TEST RESULTS OF THE FIVE LWC ALGORITHMS WITH ECB MODE.

Case-Study Scenarios	Runs	Runtime	Fluence [10^8 neutrons/cm^2]	FIT [Failures/10^9 h]		MTBF [10^6 h]		Memory Utilisation [kB]		Energy [mWs]
				SDC	Crash	SDC	Crash	RAM	Flash	
AES-B*	5525	1.34s	260.29	1.5	3.5	669	289	3.15	165.94	283
AES	130k	3.07ms	14.00	27	37	77.9	58.4	3.15	46.75	0.64
ARIA	125k	0.26ms	1.11	115	345	18.7	6.2	2.81	47.44	0.054
IDEA	124k	0.05ms	0.21	1209	1814	1.8	1.2	2.81	45.18	0.010
NOEKEON	123k	0.06ms	0.23	543	1087	3.9	1.9	2.82	44.71	0.012
SEED	122k	0.10ms	0.42	314	944	6.8	2.3	2.81	48.77	0.020

*AES-B version has a large plaintext input and was evaluated in the July 2022 radiation campaign.

radiation campaign, the plaintext input size has been increased to 48kB and the radiation test flow has been extended to implement a checksum on the DUT to distinguish computation and communication faults. Figure 8 illustrates the overhead communication time spent controlling each run of the cryptography algorithm. Note that for the first approach (AES), communication time and control of each run takes about 21ms and 3ms for the algorithm's computation, while the second approach (AES-B) uses the same communication and control time for a computation time of 1.34s.

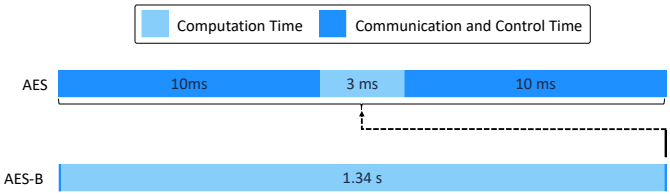


Fig. 8. AES timeline running on Arm Cortex-M4 for two plaintext sizes.

Table I also compares the two plaintext input sizes for the AES with ECB mode. Note that AES had 130k runs, while AES-B had only 5.5k runs. However, due to the control time overhead, the effective fluence was more than 18x higher for AES-B. Moreover, even with a reasonable longer radiation exposure time, fewer events were observed. This may indicate that many events captured in February 2022 resulted from communication errors that corrupted the output data or desynchronised the connection with the board going into a hang. Finally, the extended version of the radiation test flow provided better observability in separating the communication faults from those of the computation. For this reason, it is used in the following case study that considers LWC algorithms implemented with a different mode of operation.

C. Relative Trade-off Analysis for ECB Implementations

Figure 9 shows the relative trade-off between reliability, power efficiency, performance and memory footprint overhead for each LWC algorithm using the ECB mode of operation. Values, obtained from radiation experiments conducted in February 2022, are normalised between scores of 1 and 5. This correlates the MWTF for SDC and crash events with the following metrics: (*i*) lowest runtime cost (LRC) that ranks the best performing algorithms; (*ii*) memory footprint saving (MEM) which is related to lower RAM usage; and (*iii*) energy saving (EN) that ranks the algorithms with the lowest power consumption for each run.

The importance of the MWTF metric is revealed when comparing AES with IDEA (i.e., best and worst FIT_{SDC}). If, on the one hand, AES has almost 98% less FIT_{SDC}. On the other hand, IDEA has 40% more $MWTF_{SDC}$. These results show that the IDEA algorithm has better soft error reliability than AES when considering the time to perform a 128-bits encryption. Furthermore, ARIA has the highest $MWTF_{SDC}$, followed by NOEKEON, SEED, IDEA, and AES. Note that ARIA had a $MWTF_{SDC}$ 2.9× higher than AES. Regarding the crashes, NOEKEON has the highest $MWTF_{crash}$, followed by ARIA, IDEA, SEED, and AES. The difference between NOEKEON and AES results is 1.89×. It is noteworthy that the total power consumption is directly linked to the algorithm runtime since the board resource usage is similar for all algorithms.

VI. Radiation Results: Case-Study B

Section VI-A exploits the soft error reliability of four LWC algorithms implemented with the CTR mode, considering a neutron radiation test held in July 2022 at the LPSC. Section VI-B analyses the trade-off between reliability and performance for the experiments conducted in July 2022.

A. CTR Mode Radiation Results

Figure 10 shows the radiation results regarding the FIT metric for the ARIA, IDEA, NOEKEON and SEED algorithms. Due to some issues during the radiation campaign, it was not possible to run the AES algorithm using CTR mode.

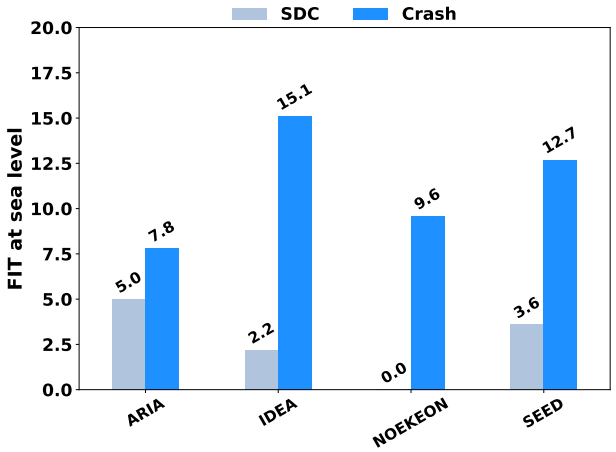


Fig. 10. Radiation-induced failures considering the FIT metric for four LWC algorithms with CTR mode.

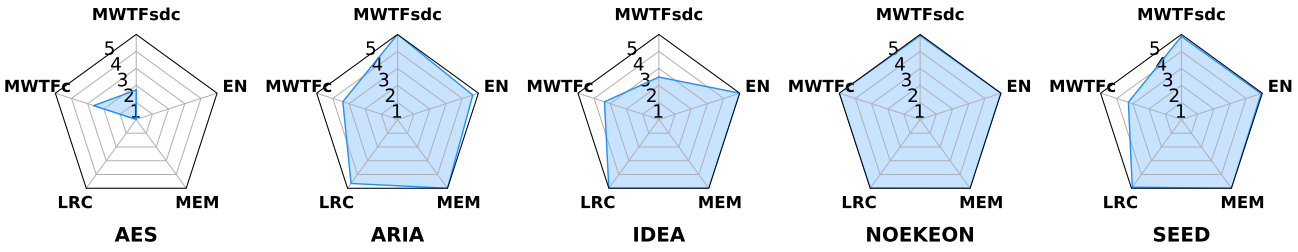


Fig. 9. Normalised profile of each LWC algorithm using ECB mode for the radiation experiments conducted in February 2022, considering MWTFsdc (Mean Work to Failure for SDC), MWTFc (Mean Work to Failure for crash), LRC (lower runtime cost), MEM (memory footprint saving), and EN (energy saving).

TABLE II
SUMMARY OF PERFORMANCE AND NEUTRON RADIATION TEST RESULTS OF FOUR LWC ALGORITHMS WITH CTR MODE.

Case-Study Scenarios	Runs	Runtime	Fluence [10^8 neutrons/cm^2]	FIT [Failures/10^9 h]		MTBF [10^6 h]		Memory Utilisation [kB]		Energy [mWs]
				SDC	Crash	SDC	Crash	RAM	Flash	
ARIA	6044	0.86s	182.74	5.0	7.8	201	128	4.00	219.35	181
IDEA	5514	0.31s	60.09	2.2	15.1	463	66	4.00	219.35	65
NOEKEON	5507	0.21s	40.66	-	9.6	-	104	4.00	219.35	44
SEED	5617	0.36s	71.09	3.6	12.8	274	78	4.00	219.35	76

As expected, results show that crashes are still higher than SDCs in all cases. More than that, this proportion has increased. For example, the IDEA algorithm presented 6.8× more crashes than SDCs while NOEKEON had only crashes.

Table II summarises the neutron radiation test results for the CTR mode solutions. Note that the number of runs for each CTR cryptography algorithm implementation, ranges from 5.5k to 6k, which is considerable lower than their respective ECB versions (i.e., from 122k to 130k runs). This difference results from the increase in the plaintext size used in the July 2022 radiation campaign, which directly affects the execution time of cryptography algorithms (see column 3 in Tables I and II). As expected, this increase in execution time led to a rise in energy consumption, as shown in column 11 of Table II. The choice for a larger plaintext size was to increase the algorithms' radiation exposure time, enabling a more adequate soft error evaluation. Unlike ECB mode, the increase of plaintext size does not weaken the security of CTR mode. The results show that the plaintext increased had little effect on the behaviour of the events, as shown in Figure 11. On the other hand, the radiation fluence varied widely, ranging from $40 \times 10^8 n/cm^2$ (NOEKEON) to $182 \times 10^8 n/cm^2$ (ARIA), resulting in a total fluence of $6.14 \times 10^{10} n/cm^2$. This CTR mode greater fluence brings a notable gain in the statistical significance of the radiation experiments performed.

Furthermore, the CTR mode implementations provide a notable soft error reliability improvement, leading to better MTBF results. For SDC events, the values ranged from 201 million (ARIA) to 463 million hours to failure (IDEA). For crash events, the values ranged from 66 million (IDEA) to 128 million hours to failure (ARIA). According to Table I, the CTR mode implementation of NOEKEON presents the best soft error reliability regarding the number of SDC events. However, this cryptography algorithm spent significantly less time exposed to neutron radiation, which may have led to no occurrence of SDC (see Figure 11). Note that a single SDC event would increase the NOEKEON's FIT to 3.2, making it less reliable than the IDEA algorithm.

B. Trade-off Between Reliability and Performance

Figure 12 shows the normalised profile results for four LWC algorithms using CTR mode for the radiation experiments performed in July 2022, ranging from 1 to 5 where more is better.

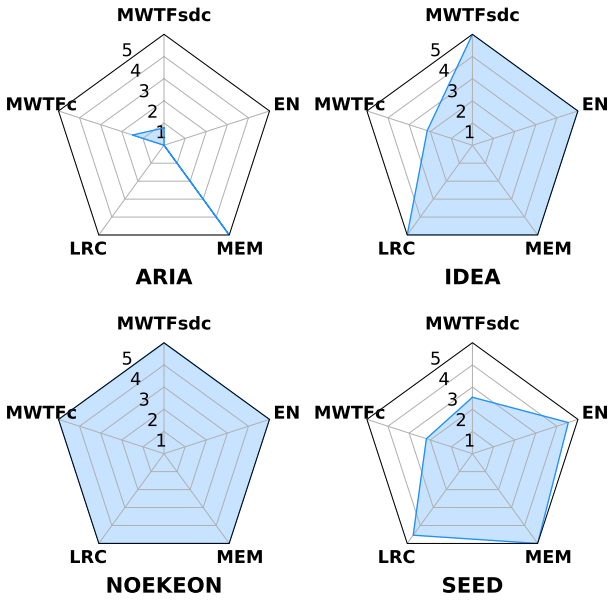


Fig. 12. Normalised profile of each LWC algorithm using CTR mode for the radiation experiments conducted in July 2022, considering MWTFsdc (Mean Work to Failure for SDC), MWTFc (MWTF for crash), LRC (lower runtime cost), MEM (memory footprint saving), and EN (energy saving).

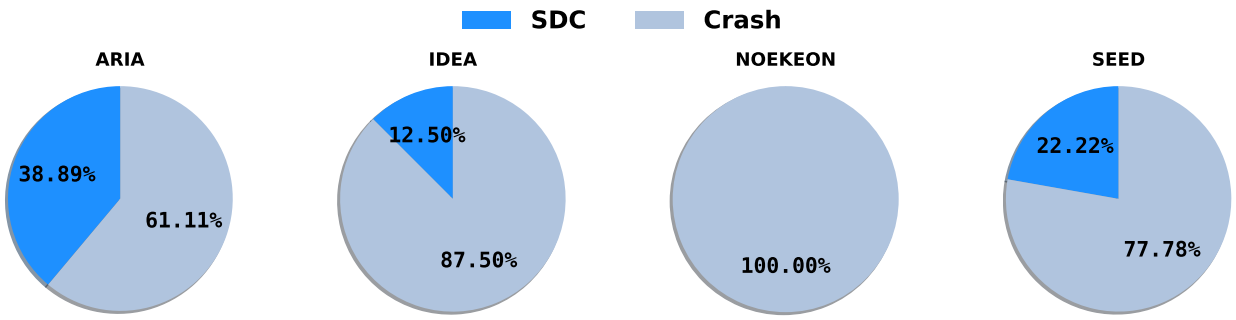


Fig. 11. Ratio of radiation-induced event types observed in each CTR mode case-study during the radiation test campaigns conducted in July 2022.

This Section focuses on the MWTF metric, similar to Section V-C, which correlates MWTF with lowest runtime cost (LRC), memory footprint saving (MEM), and energy saving (EN). Note that the NOEKEON algorithm did not have SDC-type failures (see Figure 11). Therefore, for comparison purposes, this parameter was set to one. Results show that the NOEKEON algorithm presents the best $MWTF_{crash}$ and $MWTF_{SDC}$, which is tied with the IDEA on $MWTF_{SDC}$. The $MWTF_{SDC}$ for both algorithms are 6× better than ARIA and 2× better than SEED $MWTF_{SDC}$. Results also show that NOEKEON has an $MWTF_{crash}$ 3.3× better than ARIA and about 2.3× better than IDEA and SEED. Interestingly, the NOEKEON algorithm had the best result considering the trade-off between MWTF and performance, memory usage, and energy saving for both modes of operation.

VII. Conclusion

This paper presents the soft error reliability assessment of five LWC algorithms considering two modes of operation and their execution on a resource-constrained device under high-energy neutron radiation. Results suggest that ARIA has higher reliability against SDCs and NOEKEON against crashes when considering the MWTF metric using the ECB mode implementation. On the other hand, NOEKEON stood out by presenting the best resilience against SDCs and crashes considering the MWTF metric for the CTR mode implementation. Results also show that the NOEKEON algorithm gives the best relative soft error reliability, performance, power efficiency and memory footprint utilisation trade-offs between the five algorithms considering both ECB and CTR implementations.

Future works include the soft error reliability analysis of different LWC algorithms taking into account wide-energy spectrum neutron and other Arm and RISC-V microprocessor architectures. We also intend to make a more in-depth comparison between different modes of operation, considering not only different LWC algorithms but also various plaintext sizes.

References

[1] Z. Guo, N. Karimian, M. M. Tehranipoor, and D. Forte, "Hardware Security Meets Biometrics for the Age of IoT," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2016, pp. 1318–1321.

[2] P.-E. Novac, G. B. Hacene, A. Pegatoquet, B. Miramond, and V. Gripon, "Quantization and Deployment of Deep Neural Networks on Microcontrollers," *Sensors*, vol. 21, no. 9, p. 2984, April 2021.

[3] I. Arghire, "St. Jude Medical Recalls 465,000 Pacemakers Over Security Vulnerabilities," 2017. [Online]. Available: https://www.securityweek.com/st-jude-medical-recalls-465000-pacemakers-over-security-vulnerabilities

[4] M. Dworkin, "Recommendation for Block Cipher Modes of Operation: Methods and Techniques," U.S. Government, Tech. Rep., 2001. [Online]. Available: https://doi.org/10.6028/NIST.SP.800-38A

[5] R. Gao, S. Li, Y. Gao, and R. Guo, "A lightweight cryptographic algorithm for the transmission of images from road environments in self-driving," *Cybersecurity*, vol. 4, no. 3, pp. 1–11, February 2021.

[6] M. A. Bahnasawi, K. Ibrahim, A. Mohamed, M. K. Mohamed, A. Moustafa, K. Abdelmonem, Y. Ismail, and H. Mostafa, "ASIC-oriented comparative review of hardware security algorithms for internet of things applications," in *IEEE International Conference on Microelectronics (ICM)*, 2016, pp. 285–288.

[7] STMicroelectronics, "STEVAL-STLKT01V1," 2022. [Online]. Available: www.st.com/en/evaluation-tools/steval-stlkt01v1.html

[8] P. K. Kushwaha, M. P. Singh, and P. Kumar, "A Survey on Lightweight Block Ciphers," *International Journal of Computer Applications*, vol. 96, no. 17, pp. 1–7, June 2014.

[9] B. J. Mohd, T. Hayajneh, and A. V. Vasilakos, "A survey on lightweight block ciphers for low-resource devices: Comparative study and open issues," *Journal of Network and Computer Applications*, vol. 58, pp. 73–93, December 2015.

[10] V. A. Thakor, M. A. Razzaque, and M. R. A. Khandaker, "Lightweight Cryptography Algorithms for Resource-Constrained IoT Devices: A Review, Comparison and Research Opportunities," *IEEE Access*, vol. 9, pp. 28 177–28 193, January 2021.

[11] C. A. Lara-Nino, A. Diaz-Perez, and M. Morales-Sandoval, "Lightweight Hardware Architectures for the Present Cipher in FPGA," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 9, pp. 2544–2555, September 2017.

[12] D.-H. Bui, D. Puschini, S. Bacles-Min, E. Beigné, and X.-T. Tran, "AES Datapath Optimization Strategies for Low-Power Low-Energy Multisecurity-Level Internet-of-Things Applications," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 12, pp. 3281–3290, December 2017.

[13] L. P. I. Ledwaba, G. P. Hancke, H. S. Venter, and S. J. Isaac, "Performance Costs of Software Cryptography in Securing New-Generation Internet of Energy Endpoint Devices," *IEEE Access*, vol. 6, pp. 9303–9323, January 2018.

[14] B. J. Mohd and T. Hayajneh, "Lightweight Block Ciphers for IoT: Energy Optimization and Survivability Techniques," *IEEE Access*, vol. 6, pp. 35 966–35 978, June 2018.

[15] W. Iqbal, H. Abbas, P. Deng, J. Wan, B. Rauf, Y. Abbas, and I. Rashid, "ALAM: Anonymous Lightweight Authentication Mechanism for SDN-Enabled Smart Homes," *IEEE Internet of Things Journal*, vol. 8, no. 12, pp. 9622–9633, June 2021.

[16] V. da Rocha, N. Moura, J. Gava, V. Bandeira, L. Ost, R. Reis, and R. Garibotti, "Soft Error Reliability Assessment of Lightweight Cryptographic Algorithms for IoT Edge Devices," in *IEEE International Symposium on Circuits and Systems (ISCAS)*, 2022, pp. 457–460.

[17] V. Bandeira, J. Sampford, R. Garibotti, M. G. Trindade, R. P. Bastos, R. Reis, and L. Ost, "Impact of radiation-induced soft error on embedded cryptography algorithms," *Microelectronics Reliability*, vol. 126, p. 114349, September 2021.

[18] J. W. Yu and M. D. Aagaard, "Benchmarking and optimizing AES for lightweight cryptography on ASICs," in *Lightweight Cryptography Workshop (LCW)*, 2019, pp. 1–12.

[19] H. Zodpe and A. Sapkal, "An efficient AES implementation using FPGA with enhanced security features," *Journal of King Saud University - Engineering Sciences*, vol. 32, no. 2, pp. 115–122, February 2020.

[20] X. Zhang, M. Li, and J. Hu, "Optimization and Implementation of AES Algorithm Based on FPGA," in *IEEE International Conference on Computer and Communications (ICCC)*, 2018, pp. 2704–2709.

[21] J. Dutertre, A. Mirbaha, D. Naccache, A. Ribotta, A. Tria, and T. Vaschalde, "Fault Round Modification Analysis of the advanced encryption standard," in *IEEE International Symposium on Hardware-Oriented Security and Trust (HST)*, 2012, pp. 140–145.

[22] C. Roscian, J. Dutertre, and A. Tria, "Frontside laser fault injection on cryptosystems - Application to the AES' last round -," in *IEEE International Symposium on Hardware-Oriented Security and Trust (HOST)*, 2013, pp. 119–124.

[23] G. Bertoni, L. Breveglieri, I. Koren, P. Maistri, and V. Piuri, "Error analysis and detection procedures for a hardware implementation of the advanced encryption standard," *IEEE Transactions on Computers*, vol. 52, no. 4, pp. 492–505, April 2003.

[24] R. Banu and T. Vladimirova, "Fault-Tolerant Encryption for Space Applications," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 45, no. 1, pp. 266–279, January 2009.

[25] K. Wu, R. Karri, G. Kuznetsov, and M. Goessel, "Low cost concurrent error detection for the advanced encryption standard," in *IEEE International Test Conference (ITC)*, 2004, pp. 1242–1248.

[26] J. Gava, V. Bandeira, F. Rosa, R. Garibotti, R. Reis, and L. Ost, "SOFIA: An automated framework for early soft error assessment, identification, and mitigation," *Journal of Systems Architecture*, vol. 131, p. 102710, October 2022.

[27] NIST, "Specification for the Advanced Encryption Standard (AES)," Federal Information Processing Standard (FIPS) Publication 197, November 2001.

[28] D. Kwon, J. Kim, S. Park, S. H. Sung, Y. Sohn, J. H. Song, Y. Yeom, E.-J. Yoon, S. Lee, J. Lee, S. Chee, D. Han, and J. Hong, "New Block Cipher: ARIA," in *Information Security and Cryptology (ICISC)*, 2004, pp. 432–445.

[29] X. Lai, "On the design and security of block ciphers," Ph.D. dissertation, ETH Zürich, 1992.

[30] J. Daemen, M. Peeters, G. Assche, and V. Rijmen, "Nessie proposal: NOEKEON," in *NESSIE*, 2000, pp. 213–230.

[31] H. Lee, S. Lee, J. Yoon, D. Cheon, and J. Lee, "The SEED Encryption Algorithm," *RFC4269*, pp. 1–16, December 2005.

[32] R. Garibotti, L. Ost, A. Butko, R. Reis, A. Gamatié, and G. Sassatelli, "Exploiting memory allocations in clusterised many-core architectures," *IET Computers & Digital Techniques*, vol. 13, no. 4, pp. 302–311, April 2019.

[33] IEEE, "IEEE Standard Specification for Public Key Cryptographic Techniques Based on Hard Problems over Lattices," *IEEE Std 1363.1-2008*, pp. 1–81, March 2009.

[34] STMicroelectronics, "STM32 Nucleo L476RG," 2022. [Online]. Available: https://www.st.com/en/evaluation-tools/nucleo-l476rg.html

[35] C. Slayman, "JEDEC Standards on Measurement and Reporting of Alpha Particle and Terrestrial Cosmic Ray Induced Soft Errors," in *Soft Errors in Modern Electronic Systems*, 2011, pp. 55–76.

[36] Keweisi, "Keweisi KWS-MX18 USB tester," 2023. [Online]. Available: https://elektro.turanis.de/html/prj125/Keweisi%20KWS-MX18%20-%20User%20Manual.pdf