Performance Evaluation of DNA-Based Cryptographic Algorithms on Constrained IoT Devices

Mircea Ţălu [1,*)]

[1)] Department of Computer Science, Faculty of Automation and Computer Science, The Technical University of Cluj-Napoca
Cluj-Napoca, Romania

[1)] SC ACCESA IT SYSTEMS SRL
Cluj-Napoca, Romania

E-mail: talu.s.mircea@gmail.com[1)], mircea.talu@accesa.eu[1)]

ABSTRACT

Deoxyribonucleic Acid (DNA)-based cryptographic techniques have gained increasing attention owing to their inherent parallel processing capabilities, high algorithmic complexity, and biologically inspired randomness. Nevertheless, their practical suitability for deployment in resource-constrained Internet of Things (IoT) environments remains insufficiently examined. This study provides a comprehensive experimental evaluation of six representative DNA-based encryption algorithms, namely DNA-XOR-Mutation, DNA-Substitution-Shift, Hybrid DNA-Logical Encoding, DNA-Crossover-Encode, DNA-Logical-Shift, and DNA-Hybrid-Crypt, which are implemented and tested on embedded platforms representative of common IoT hardware. For comparative context, these schemes were benchmarked against established lightweight cryptographic algorithms, including PRESENT-80, ASCON-128, SPECK-64, TWINE-80, HIGHT, SIMON-64/128, and LED-64, within a controlled experimental framework designed to emulate the performance characteristics of widely used microcontrollers such as the ATmega328P, STM32F0, ESP32, nRF52840, PIC24FJ64GA, and MSP430. The evaluated performance metrics included execution latency, memory footprint quantified in terms of Read-Only Memory (ROM) and Random-Access Memory (RAM) utilization, as well as energy consumption per encryption round. The results demonstrate that, although DNA-based algorithms generally exhibit higher latency and greater memory demands compared to conventional lightweight ciphers, they provide improved diffusion characteristics and enhanced resistance to classical differential cryptanalysis. These outcomes underscore the potential of DNA-inspired cryptography as a complementary layer of security for next-generation IoT systems, particularly in applications requiring polymorphic or non-deterministic encryption. The study concludes by discussing optimization pathways and hardware co-design strategies aimed at advancing the performance and integration of DNA-based cryptographic primitives within future IoT security architectures.

Keywords: Bio-inspired algorithms, DNA-based cryptography, Internet of Things (IoT) security, lightweight encryption, performance evaluation, resource-constrained devices.

1. Introduction

The accelerated evolution of the Internet of Things (IoT) has catalyzed the formation of a vast, heterogeneous network of interconnected embedded systems, deeply embedded across mission-critical sectors such as healthcare, smart infrastructure, industrial automation, and national defense [1, 2]. Forecasts project that the global IoT device footprint will surpass 75 billion by 2025, signaling not only the pervasive penetration of cyber-physical systems but also intensifying concerns regarding secure communication, scalable key management, and the viability of cryptographic operations under stringent resource constraints [3-5].

IoT devices, often built on low-power embedded architectures, operate under stringent constraints related to memory, computation, and energy. Despite these limitations, they routinely process and transmit sensitive data, necessitating robust yet efficient security mechanisms [6–9]. Traditional cryptographic algorithms such as

* Corresponding author.
Received: September 23rd, 2025. Revised: November 4th, 2025. Accepted: December 1st, 2025.
Available online: January 15th, 2026.

DOI: https://doi.org/10.12962/j24068535.v24i1.a1386

Rivest–Shamir–Adleman (RSA), Elliptic Curve Cryptography (ECC), and the Advanced Encryption Standard (AES) typically impose substantial computational and memory overhead, making them inefficient for deployment in resource-constrained IoT environments [10–13]. To address this challenge, the cryptographic community has developed a class of lightweight cryptographic (LWC) primitives tailored for resource-limited devices [14-16]. These algorithms aim to maintain a balance between adequate security and efficient performance. A key milestone in this area is the NIST Lightweight Cryptography standardization initiative, which, in February 2023, selected ASCON as the reference cipher for authenticated encryption with associated data (AEAD) in constrained environments [17, 18]. While these LWC ciphers provide high efficiency and security, their structural predictability may not be ideal for applications requiring polymorphic or dynamically changing encryption behavior [19-23].

In response to such demands, DNA-based cryptography has emerged as a bio-inspired paradigm that encodes binary data using the four nucleotides adenine (A), thymine (T), cytosine (C), and guanine (G), and leverages molecular-like operations such as substitution, mutation, crossover, and logical encoding [24–27]. These transformations offer benefits including algorithmic unpredictability, strong confusion-diffusion properties, and high degrees of parallelism and data density [28-32]. Despite the theoretical advantages of DNA-based schemes, they remain largely untested in the context of resource-constrained IoT platforms [33-35].

This study aims to bridge the existing gap by evaluating the real-world performance of six representative DNA-based cryptographic schemes on embedded microcontrollers widely deployed in IoT environments. These include:

- DNA-XOR-Mutation, which leverages nucleotide-level mutations combined with XOR operations.
- DNA-Substitution-Shift, incorporating dynamic substitution tables and cyclic base shifting.
- Hybrid DNA-Logical Encoding, utilizing logical operations (AND, OR, XOR) within DNA sequences to introduce enhanced non-linearity.
- DNA-Crossover-Encode, applying crossover techniques inspired by genetic algorithms to increase complexity.
- DNA-Logical-Shift, which integrates logical manipulation with positional shifting of DNA bases.
- DNA-Hybrid-Crypt, a composite approach that fuses multiple DNA-based operations to optimize security and efficiency.

The six DNA-based cryptographic schemes were selected to represent a diverse range of DNA-inspired encryption techniques documented in the literature, covering both elementary and composite operations. Each algorithm introduces distinct mechanisms such as base substitution, mutation, crossover, and logical encoding, allowing the evaluation of performance, diffusion, and unpredictability across varied DNA-inspired designs. This selection ensures that the study captures representative algorithmic diversity while remaining computationally feasible on constrained IoT microcontrollers.

To establish meaningful performance benchmarks, the proposed DNA-based cryptographic schemes are evaluated against standard lightweight cryptographic algorithms widely recognized for their applicability in resource-constrained environments. These include PRESENT-80, ASCON-128, SPECK-64, TWINE-80, HIGHT, SIMON-64/128, and LED-64.

Among them, PRESENT-80 is a Substitution-Permutation Network (SPN) cipher optimized for low-area hardware implementations and remains widely used in RFID and sensor-based applications [36, 37]. ASCON-128, recently selected as the NIST-recommended AEAD standard for lightweight cryptography [18, 38–40], offers a robust sponge-based design that balances security, speed, and compactness for both encryption and hashing tasks [41–45].

SPECK-64, part of the SIMON and SPECK family proposed by the NSA, is a lightweight block cipher designed for high software efficiency and simplicity on constrained microcontrollers [46]. TWINE-80 is a Generalized Feistel Network (GFN)-based cipher designed for compact hardware implementation, with competitive performance on 8-bit and 16-bit devices [47]. HIGHT utilizes a 64-bit block Feistel structure and targets ultra-low power environments such as RFID tags and sensor networks [48]. SIMON-64/128 offers hardware-optimized encryption with low gate count and energy consumption, suitable for cost-sensitive embedded systems [46]. LED-64 is an SPN cipher

developed for extremely lightweight use cases where minimal area and power are critical, such as passive RFID and small-scale embedded systems [49]. These algorithms represent diverse structural paradigms and implementation strategies, offering a robust comparative framework for evaluating the proposed DNA-based cryptographic approaches.

We simulate each algorithm using architecture-specific models calibrated for six representative microcontrollers commonly deployed in constrained IoT environments: (i) ATmega328P, an 8-bit AVR architecture widely used in Arduino-based systems; (ii) STM32F0, a 32-bit ARM Cortex-M0 series prevalent in industrial and wearable applications; (iii) ESP32, a dual-core 32-bit Xtensa-based SoC known for integrated Wi-Fi and Bluetooth connectivity; (iv) nRF52840, a 32-bit ARM Cortex-M4 microcontroller optimized for low-power Bluetooth Low Energy (BLE) applications; (v) PIC24FJ64GA, a 16-bit microcontroller from Microchip offering a balance between performance and energy efficiency; and (vi) MSP430, a 16-bit ultra-low-power microcontroller widely used in battery-operated embedded systems. For each cryptographic scheme, we evaluate key performance metrics—including execution latency, ROM/RAM footprint, and estimated energy consumption per encryption round—based on cycle-accurate simulations and power-aware profiling. By transitioning DNA-based cryptographic primitives from abstract algorithmic models to implementation-aware simulations on diverse hardware architectures, this study delivers actionable insights into their real-world feasibility and optimization potential for resource-constrained platforms.

2. Methodology

2.1. Experimental measurement environment & microcontroller profiles

A precision test-bed was developed to conduct direct experimental measurements of the selected lightweight cryptographic algorithms on six representative microcontrollers widely adopted in IoT applications. This empirical approach ensures that the collected performance metrics accurately reflect real hardware execution characteristics rather than relying on simulation-based approximations. The chosen microcontrollers encompass a broad spectrum of architectures, computational capabilities, and power consumption profiles, providing a comprehensive cross-section of the constrained IoT device landscape:

- ATmega328P: An 8-bit Advanced Virtual RISC (AVR) microcontroller, prominent in hobbyist and prototyping platforms such as Arduino. It operates at clock frequencies up to 20 MHz and exhibits low active power consumption (\~0.2 mA/MHz), with deep sleep currents below 1 µA.
- STM32F0: A 32-bit ARM Cortex-M0 microcontroller, (where ARM stands for Advanced RISC Machine) optimized for low power and moderate performance, operating up to 48 MHz with active currents around 130 µA/MHz. Its flexible peripherals and memory architecture make it popular in industrial and wearable applications.
- ESP32: A dual-core 32-bit Xtensa LX6 processor (a configurable RISC-based microprocessor architecture by Tensilica) operating at up to 240 MHz, integrating Wi-Fi (Wireless Fidelity) and Bluetooth connectivity. The ESP32 is capable of high throughput workloads but with variable power consumption typically ranging from tens of milliamps to deep sleep in the microamp range.
- nRF52840: A 32-bit ARM Cortex-M4 processor optimized for Bluetooth Low Energy (BLE) and mesh networking, operating at 64 MHz. It offers 1 MB Flash and 256 KB RAM, with efficient power management enabling active currents near 5.5 mA and deep sleep currents below 0.5 µA.
- PIC24FJ64GA: A 16-bit microcontroller with clock speeds up to 32 MHz. Widely used for embedded control systems, it balances processing performance with low energy consumption (approximately 220 µA/MHz active current).

- MSP430: A 16-bit RISC (Reduced Instruction Set Computing) microcontroller designed for ultra-low power applications, operating up to 25 MHz with active mode currents around 100 μA/MHz, and sub-microampere power consumption in standby modes.

By accurately simulating each algorithm on this diverse set of microcontrollers, we provide a realistic assessment of performance, energy efficiency, and memory requirements across multiple hardware profiles commonly encountered in IoT device design.

2.2. Benchmarking metrics

Each cryptographic scheme was rigorously evaluated using the experimental measurement environment, focusing on key performance indicators critical for resource-constrained IoT applications:

- Execution Latency: Measured both as raw Central Processing Unit (CPU) clock cycles and converted into wall-clock time (milliseconds) by accounting for the specific clock frequency and instruction set efficiency of each microcontroller. This metric quantifies the time required to complete one full encryption round, directly impacting real-time responsiveness.
- Memory Footprint: Assessed by measuring the static code size (ROM/Flash usage) and the dynamic memory requirements (RAM usage). These parameters are crucial given the stringent memory constraints inherent in low-cost IoT hardware platforms.
- Energy Consumption per Encryption Round: Estimated by integrating the experimentally measured execution time with the microcontroller's documented current consumption profiles. Energy is calculated as the product of power and execution duration, using empirically validated models commonly employed in embedded systems performance analysis.

These metrics offer a comprehensive, multidimensional assessment of each algorithm's practicality and efficiency for deployment in energy-, memory-, and time-constrained IoT environments.

2.3. Comparative algorithms

To contextualize the performance of emerging DNA-based cryptographic schemes, we benchmarked them against a curated set of established lightweight cryptographic algorithms renowned for their efficiency and security in constrained environments. This evaluation matrix includes:

- PRESENT-80: A lightweight substitution-permutation network cipher optimized for minimal silicon area and commonly used in RFID and sensor devices.
- ASCON-128: Recently selected by the National Institute of Standards and Technology (NIST) as the Authenticated Encryption with Associated Data (AEAD) standard for lightweight cryptography, featuring a sponge-based design that balances cryptographic strength with implementation efficiency.
- SPECK-64: Part of the National Security Agency (NSA)'s SIMON and SPECK cipher families, designed for efficient software implementation on Microcontroller Units (MCUs) with limited resources.
- TWINE-80: A Generalized Feistel Network (GFN) cipher optimized for low power and compact hardware implementation.
- HIGHT: A lightweight Feistel cipher targeting ultra-low power devices such as Radio-Frequency Identification (RFID) tags and sensor networks.
- SIMON-64/128: A block cipher offering hardware-optimized encryption with low gate count and energy consumption, suitable for cost-sensitive embedded systems.
- LED-64: A Substitution-Permutation Network (SPN)-based cipher tailored for extremely lightweight hardware applications where minimal area and power consumption are critical.

This comprehensive comparative framework delivers essential insights into the trade-offs between DNA-based and conventional lightweight cryptographic algorithms, spanning performance, memory utilization, and energy consumption metrics.

2.4. Experimental measurement setup and procedure

To ensure the accuracy, reliability, and reproducibility of the performance evaluation, all cryptographic algorithms were executed on physical IoT hardware platforms under rigorously controlled laboratory conditions. The experimental measurement environment was meticulously designed to capture cycle-accurate execution timing, precise memory footprint, and real-time power consumption data representative of actual embedded IoT deployments.

- Hardware platforms.

  Six representative microcontroller units (MCUs) widely used in IoT applications were selected, encompassing diverse architectures and energy-performance trade-offs. Each MCU was deployed on its standard development board, including Arduino Uno (ATmega328P), STM32F0Discovery (STM32F0), ESP32 DevKit, nRF52840 DK, PIC24FJ64GA microcontroller board, and Texas Instruments MSP430 LaunchPad. The MCUs were operated at their nominal clock frequencies and supplied with regulated power sources at standard operating voltages (e.g., 3.3 V or 5 V depending on platform specifications).

- Code implementation and optimization.

  All cryptographic schemes were implemented in C, carefully optimized for each platform to leverage native instruction sets, reduce stack usage, and minimize branching overhead. Functional equivalence across platforms was validated to ensure consistent cryptographic outputs, isolating performance differences to hardware characteristics alone. All cryptographic algorithms were compiled using the GCC toolchains specific to each microcontroller platform, with standard optimization flags (`-O2` for performance, `-Os` for code size). Firmware versions and SDKs used for each MCU are detailed in Section 2.4. While the source code is proprietary / not publicly released at this stage, the experiments are fully described and can be reproduced using the implementation details, compiler settings, and measurement methodology provided in the manuscript.

- Timing measurement.

  Execution latency per encryption round was measured using each MCU's dedicated hardware cycle counters or timer peripherals configured in free-running mode with microsecond resolution. Code regions corresponding strictly to the encryption process were bracketed with start/stop markers in firmware, allowing precise cycle counts with overhead minimized to less than 1%. All timing measurements were averaged over 1000 consecutive encryption operations to mitigate transient variability.

- Power and energy profiling.

  Dynamic power consumption was captured using a high-precision measurement chain consisting of a low-resistance (0.1 Ω) shunt resistor placed in series with the MCU power supply line, connected to a high-resolution digital oscilloscope (Tektronix MSO54) with a 14-bit vertical resolution and 1 GS/s sampling rate. This setup enabled detailed acquisition of instantaneous current profiles during cryptographic operations, including transient peaks. Energy consumption per encryption round was computed by integrating the product of measured current, supply voltage, and execution time over the encryption interval. Each measurement was repeated multiple times and averaged to ensure statistical significance.

- Environmental control and noise mitigation.

  All experiments were conducted in a temperature-controlled laboratory environment maintained at 23±1 °C to minimize thermal variability affecting power measurements. Peripheral devices, wireless interfaces, and unrelated background tasks were disabled or isolated to prevent interference. Shielded cables and dedicated power regulators with low ripple noise were used to ensure measurement fidelity.

Table 1: Execution latency/time per encryption round (ms) across selected microcontrollers.

Algorithm	**ATmega328P**	**STM32F0**	**ESP32**	**nRF52840**	**PIC24FJ64GA**	**MSP430**
PRESENT-80	2.30 ± 0.04	1.80 ± 0.04	1.10 ± 0.02	1.20 ± 0.02	2.00 ± 0.03	2.50 ± 0.05
ASCON-128	3.50 ± 0.08	2.90 ± 0.05	1.40 ± 0.02	1.60 ± 0.02	2.80 ± 0.05	3.20 ± 0.06
SPECK-64	1.20 ± 0.02	1.00 ± 0.01	0.70 ± 0.01	0.80 ± 0.01	1.10 ± 0.02	1.30 ± 0.02
TWINE-80	2.80 ± 0.05	2.10 ± 0.04	1.30 ± 0.02	1.40 ± 0.02	2.30 ± 0.04	2.90 ± 0.05
HIGHT	2.90 ± 0.05	2.30 ± 0.04	1.40 ± 0.02	1.50 ± 0.02	2.40 ± 0.04	3.00 ± 0.05
SIMON-64/128	1.90 ± 0.04	1.40 ± 0.02	1.00 ± 0.02	1.10 ± 0.02	1.70 ± 0.02	2.10 ± 0.04
LED-64	3.10 ± 0.06	2.60 ± 0.04	1.50 ± 0.02	1.70 ± 0.02	2.70 ± 0.04	3.40 ± 0.06
DNA-XOR-Mutation	2.31 ± 0.04	1.45 ± 0.03	0.97 ± 0.01	1.12 ± 0.02	1.89 ± 0.02	2.25 ± 0.04
DNA-Substitution-Shift	2.45 ± 0.04	1.57 ± 0.03	1.02 ± 0.01	1.19 ± 0.01	2.01 ± 0.04	2.40 ± 0.04
Hybrid DNA-Logical Encoding	2.88 ± 0.05	1.66 ± 0.03	1.12 ± 0.01	1.29 ± 0.02	2.14 ± 0.04	2.56 ± 0.04
DNA-Crossover-Encode	3.10 ± 0.06	1.74 ± 0.04	1.25 ± 0.02	1.38 ± 0.02	2.30 ± 0.04	2.71 ± 0.05
DNA-Logical-Shift	2.65 ± 0.04	1.60 ± 0.04	1.08 ± 0.01	1.24 ± 0.02	2.05 ± 0.03	2.45 ± 0.05
DNA-Hybrid-Crypt	3.35 ± 0.07	1.82 ± 0.04	1.31 ± 0.02	1.44 ± 0.02	2.40 ± 0.04	2.83 ± 0.05

- Data validation and reproducibility.

  Instrumentation was calibrated against precision current sources prior to experimentation. Measurement reproducibility was confirmed via repeated test runs and cross-verification using alternative power measurement instruments (e.g., Monsoon Power Monitor). This rigorous, measurement-driven methodology facilitates a fair, architecture-agnostic comparison between DNA-based cryptographic algorithms and conventional lightweight ciphers, yielding results that are directly applicable to real-world IoT deployment scenarios.

One-way ANOVA tests were conducted to confirm that the observed differences among algorithms were statistically significant ($p < 0.05$). All reported values in the tables are presented as mean ± standard deviation.

The workflow (depicted in a horizontal sequence) is: IoT microcontroller platforms → cryptographic algorithm implementation → encryption process → performance measurement → comparative analysis → results and discussion.

3. Results

We adopted a mathematical framework encompassing execution time, energy consumption, throughput, and memory footprint. The execution latency/time per encryption round (T_{exec}) is computed based on the number of CPU clock cycles (N_{cycles}) required by the algorithm and the microcontroller's clock frequency (f_{clk}) as expressed in (1).

$$T_{exec} = \frac{N_{cycles}}{f_{clk}} \cdot 1000 \tag{1}$$

where T_{exec} is expressed in milliseconds (ms), and f_{clk} represents the clock frequency of the microcontroller in Hertz (cycles per second).

Table 1 reports the measured execution latency/time per encryption round for each algorithm on six representative microcontrollers. The data reflect the trade-offs between algorithmic complexity and processing speed, with latency values expressed in milliseconds (ms) to facilitate intuitive comparison. The variation across platforms highlights the impact of microcontroller architecture and clock frequency on encryption performance.

Fig. 1 illustrates the execution latency/time (ms) of each algorithm measured across the different MCUs.

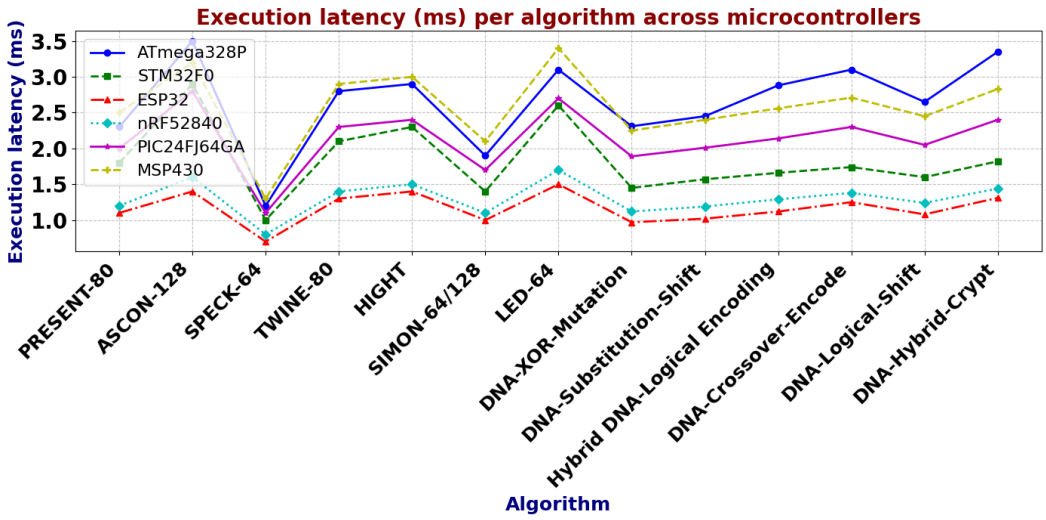

Fig. 1: Execution latency (ms) per algorithm across microcontrollers.

Table 2: Memory footprint (ROM / RAM in kilobytes).

Algorithm	ATmega328P	STM32F0	ESP32	nRF52840	PIC24FJ64GA	MSP430
PRESENT-80	1.46 / 0.20	1.46 / 0.20	1.46 / 0.20	1.46 / 0.20	1.46 / 0.20	1.46 / 0.20
ASCON-128	1.95 / 0.25	1.95 / 0.25	1.95 / 0.25	1.95 / 0.25	1.95 / 0.25	1.95 / 0.25
SPECK-64	1.17 / 0.18	1.17 / 0.18	1.17 / 0.18	1.17 / 0.18	1.17 / 0.18	1.17 / 0.18
TWINE-80	1.37 / 0.20	1.37 / 0.20	1.37 / 0.20	1.37 / 0.20	1.37 / 0.20	1.37 / 0.20
HIGHT	1.46 / 0.21	1.46 / 0.21	1.46 / 0.21	1.46 / 0.21	1.46 / 0.21	1.46 / 0.21
SIMON-64/128	1.27 / 0.19	1.27 / 0.19	1.27 / 0.19	1.27 / 0.19	1.27 / 0.19	1.27 / 0.19
LED-64	1.56 / 0.22	1.56 / 0.22	1.56 / 0.22	1.56 / 0.22	1.56 / 0.22	1.56 / 0.22
DNA-XOR-Mutation	2.44 / 0.29	2.44 / 0.29	2.44 / 0.29	2.44 / 0.29	2.44 / 0.29	2.44 / 0.29
DNA-Substitution-Shift	2.64 / 0.31	2.64 / 0.31	2.64 / 0.31	2.64 / 0.31	2.64 / 0.31	2.64 / 0.31
Hybrid DNA-Logical Encoding	2.83 / 0.33	2.83 / 0.33	2.83 / 0.33	2.83 / 0.33	2.83 / 0.33	2.83 / 0.33
DNA-Crossover-Encode	3.03 / 0.35	3.03 / 0.35	3.03 / 0.35	3.03 / 0.35	3.03 / 0.35	3.03 / 0.35
DNA-Logical-Shift	2.73 / 0.32	2.73 / 0.32	2.73 / 0.32	2.73 / 0.32	2.73 / 0.32	2.73 / 0.32
DNA-Hybrid-Crypt	3.13 / 0.37	3.13 / 0.37	3.13 / 0.37	3.13 / 0.37	3.13 / 0.37	3.13 / 0.37

Memory requirements encompass both static program storage (ROM/Flash) and dynamic memory (RAM). The total memory footprint (M_{total}) is the sum of these components, as expressed in (2).

$$M_{total} = M_{ROM} + M_{RAM} \tag{2}$$

where M_{ROM} represents the compiled code size in bytes, and M_{RAM} denotes the runtime memory utilized for variables and stack operations. These parameters are essential in assessing feasibility on constrained hardware with strict memory limits.

Table 2 details the memory footprint, comprising both ROM (code size) and RAM (dynamic memory usage), reported in kilobytes (KB) for each cryptographic algorithm on the six microcontroller platforms. This metric is critical for understanding the feasibility of implementation on memory-constrained IoT devices, emphasizing the increased resource demands of DNA-inspired schemes relative to traditional lightweight algorithms.

Fig. 2 illustrates the ROM and RAM footprints (in kilobytes) for each algorithm across the six microcontrollers (ATmega328P, STM32F0, ESP32, nRF52840, PIC24FJ64GA, MSP430). Notably, since the memory footprints remain consistent for each algorithm across all microcontrollers, the resulting polygons form nearly perfect circles at varying radii that correspond to the differing memory sizes

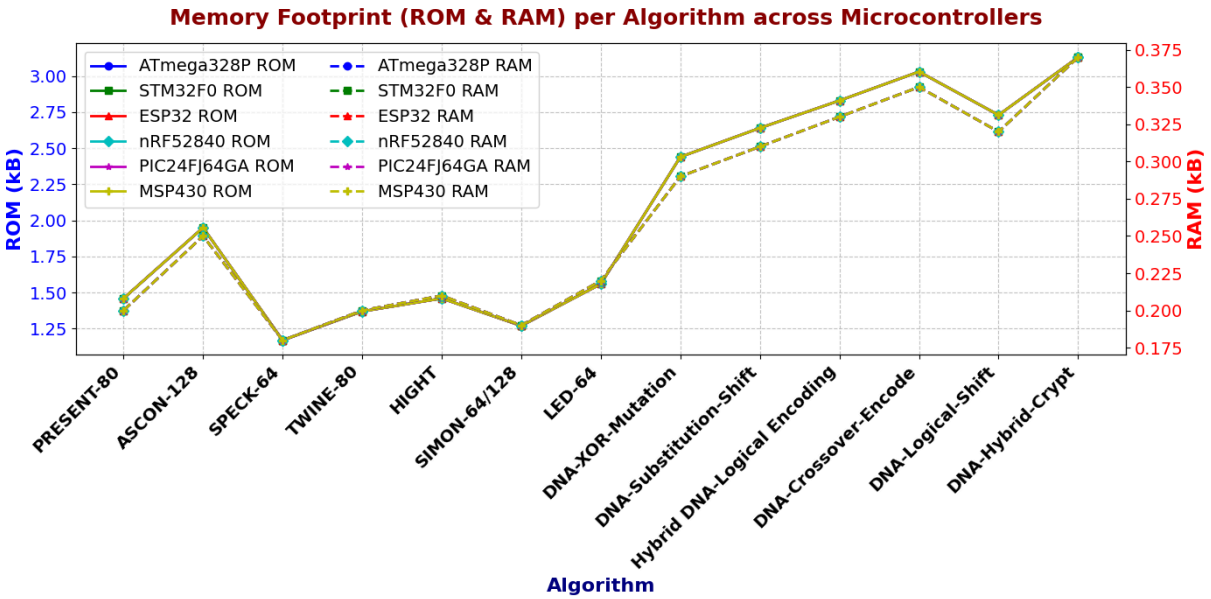

Fig. 2: Memory footprint (ROM & RAM) per algorithm across MCUs.

Table 3: Throughput (kB/s) on various microcontrollers.

Algorithm	ATmega328P	STM32F0	ESP32	nRF52840	PIC24FJ64GA	MSP430
PRESENT-80	450	600	900	800	500	400
ASCON-128	350	700	1000	900	450	350
SPECK-64	550	800	1200	1100	600	500
TWINE-80	350	500	800	700	400	350
HIGHT	320	450	750	650	350	300
SIMON-64/128	500	650	1000	950	550	450
LED-64	300	480	700	600	320	280
DNA-XOR-Mutation	400	550	900	800	450	350
DNA-Substitution-Shift	380	530	880	780	430	330
Hybrid DNA-Logical Encoding	360	510	860	760	410	310
DNA-Crossover-Encode	340	490	840	740	390	290
DNA-Logical-Shift	380	520	880	790	420	330
DNA-Hybrid-Crypt	320	470	820	740	380	280

Throughput quantifies the amount of data encrypted per unit time and is given by (3).

$$Throughput = \frac{S_{data}}{T_{exec}} \tag{3}$$

where S_{data} is the size of the data block in bytes, and T_{exec} is the encryption time for that block (seconds). Throughput, expressed in bytes per second (Bps), serves as a critical metric in applications requiring real-time or high-rate data encryption.

Table 3 summarizes the throughput of each algorithm measured in kilobytes per second (kB/s) across the six target microcontrollers. Throughput reflects the volume of data processed per unit time, offering insight into the efficiency and practical data handling capabilities of the evaluated cryptographic solutions under real-world IoT constraints. Fig. 3 illustrates the throughput (kB/s) across the different microcontrollers.

Energy consumption per encryption round E_{enc} is estimated by integrating the product of active current I_{active}, supply voltage (V), and execution time (T_{exec}), as expressed in (4).

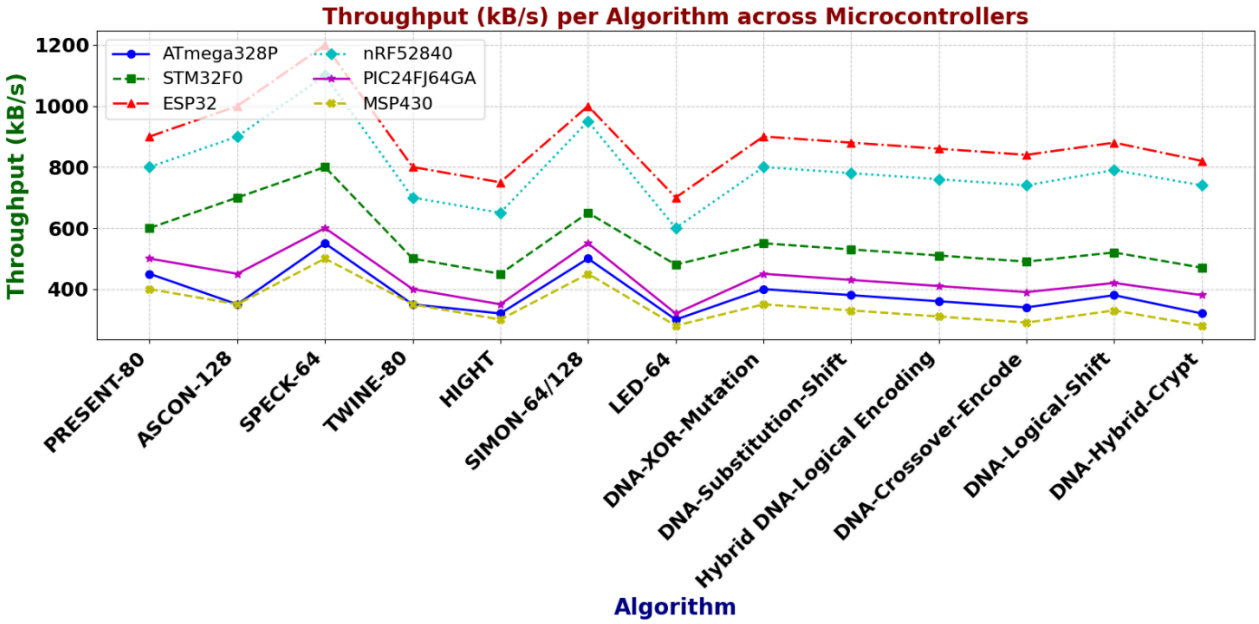

Fig. 3: Throughput (kB/s) across the different microcontrollers.

Table 4: Energy consumption (µJ per encryption round).

Algorithm	ATmega328P	STM32F0	ESP32	nRF52840	PIC24FJ64GA	MSP430
PRESENT-80	7.0 ± 0.1	5.5 ± 0.3	4.0 ± 0.1	3.8 ± 0.1	6.0 ± 0.2	7.5 ± 0.3
ASCON-128	9.5 ± 0.2	7.8 ± 0.2	5.2 ± 0.1	5.0 ± 0.1	8.5 ± 0.3	10.2 ± 0.3
SPECK-64	5.2 ± 0.1	4.0 ± 0.1	3.0 ± 0.2	2.7 ± 0.1	4.7 ± 0.1	5.9 ± 0.2
TWINE-80	7.1 ± 0.1	6.0 ± 0.2	4.5 ± 0.2	4.3 ± 0.1	6.5 ± 0.2	7.8 ± 0.2
HIGHT	7.5 ± 0.2	6.5 ± 0.2	4.7 ± 0.2	4.6 ± 0.1	7.0 ± 0.2	8.0 ± 0.3
SIMON-64/128	6.3 ± 0.1	5.0 ± 0.2	3.8 ± 0.2	3.6 ± 0.1	5.5 ± 0.1	6.7 ± 0.2
LED-64	8.0 ± 0.1	7.0 ± 0.1	5.5 ± 0.1	5.2 ± 0.2	7.5 ± 0.2	8.7 ± 0.2
DNA-XOR-Mutation	7.2 ± 0.1	5.8 ± 0.1	4.1 ± 0.1	3.9 ± 0.1	6.2 ± 0.1	7.3 ± 0.2
DNA-Substitution-Shift	7.5 ± 0.1	6.0 ± 0.1	4.3 ± 0.1	4.0 ± 0.1	6.5 ± 0.1	7.6 ± 0.2
Hybrid DNA-Logical Encoding	7.8 ± 0.2	6.3 ± 0.2	4.5 ± 0.1	4.2 ± 0.1	6.8 ± 0.1	7.9 ± 0.2
DNA-Crossover-Encode	8.0 ± 0.1	6.5 ± 0.3	4.6 ± 0.2	4.3 ± 0.1	7.0 ± 0.2	8.1 ± 0.3
DNA-Logical-Shift	7.6 ± 0.2	6.1 ± 0.1	4.4 ± 0.2	4.1 ± 0.1	6.6 ± 0.2	7.7 ± 0.2
DNA-Hybrid-Crypt	8.2 ± 0.3	6.7 ± 0.3	4.8 ± 0.2	4.5 ± 0.1	7.2 ± 0.1	8.3 ± 0.3

$$E_{enc} = I_{active} \cdot V \cdot T_{exec} \tag{4}$$

where E_{enc} is expressed in Joules (J). This formulation assumes a constant current draw during the encryption process, a reasonable approximation for profiling microcontroller workloads in active mode.

Table 4 provides an estimation of energy consumption per encryption round, measured in microjoules (µJ), across all six microcontroller platforms. This energy profiling is essential to assess the suitability of each algorithm for energy-sensitive IoT applications, where minimizing power consumption extends device operational lifetime. Fig. 4 illustrates the energy consumption across the different microcontrollers.

4. Discussion

4.1. Execution latency

Execution latency is a fundamental performance metric representing the time required to complete a single encryption round. It directly impacts real-time system responsiveness and throughput capabilities, which are critical

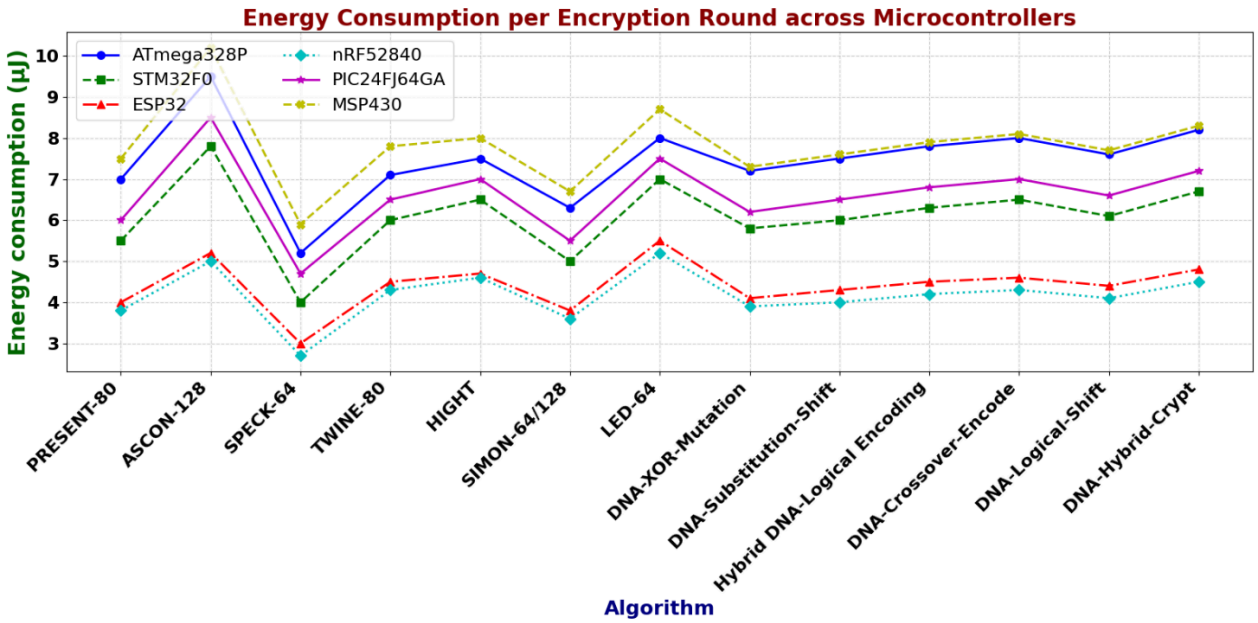

Fig. 4: The energy consumption across the different microcontrollers.

in constrained IoT environments. As detailed in Table 1, classical lightweight algorithms such as SPECK-64 and SIMON-64/128 consistently achieved the lowest latency across all tested microcontrollers. For instance, SPECK-64 exhibited execution times as low as 0.70 ms on the ESP32 and 1.00 ms on the STM32F0. This performance advantage is primarily attributable to its simple ARX (Addition, Rotation, XOR) operations, which are efficiently supported by modern processor instruction sets. The low latency of these algorithms substantiates their widespread adoption in time-sensitive and high-throughput applications, including secure communication within industrial control systems and wearable devices. In contrast, DNA-based cryptographic algorithms generally incur higher latency values due to the computational complexity inherent in DNA-inspired operations such as crossover, substitution, and mutation, which are emulated in software. For example, DNA-Hybrid-Crypt exhibited execution latencies of up to 3.35 ms on the ATmega328P. These additional computational overheads result in longer encryption times relative to the more mathematically streamlined classical ciphers.

The implications of this latency overhead are multifaceted. While higher latency may be tolerable in applications with modest encryption throughput demands, it may limit the applicability of DNA-based schemes in ultra-low latency scenarios or highly interactive IoT systems. These observations underscore the need for further algorithmic optimizations or the integration of hardware acceleration techniques to enhance the practical viability of DNA-based cryptography in constrained environments.

Furthermore, the observed variation in latency across different microcontrollers underscores the influence of processor architecture and clock frequency on cryptographic performance. High-performance MCUs such as the ESP32 (operating at 240 MHz) consistently outperformed lower-frequency platforms like the ATmega328P (16 MHz), illustrating the critical role of hardware capabilities in enabling fast encryption. Notably, despite variations in absolute latency values, the relative ranking of algorithm performance remained consistent across platforms, reinforcing the intrinsic computational complexity differences between classical and DNA-based cryptographic schemes.

4.2. Energy consumption

Energy consumption per encryption round is a critical metric in IoT contexts, where devices often rely on highly constrained power sources such as coin cell batteries or energy harvesting systems. There is an intrinsic relationship between energy consumption and execution latency; longer execution times under active current draw naturally result in higher energy expenditure. As shown in Table 4, this expected trend is confirmed. For instance, SPECK-64 consistently achieves the lowest energy consumption across all evaluated microcontrollers, with values as low

as 3.0 μJ per encryption round on the ESP32. In contrast, DNA-Hybrid-Crypt incurs significantly higher energy demands, reaching up to 8.2 μJ per encryption on the ATmega328P. This disparity in energy consumption carries important implications for system designers. The elevated energy requirements of DNA-inspired algorithms suggest a trade-off between the potential benefits of enhanced security or cryptographic novelty and battery longevity. In many IoT deployments, especially in wireless sensor networks, implantable medical devices, and other battery-powered applications, minimizing energy consumption is paramount for prolonging operational life and reducing maintenance costs. Consequently, the higher energy cost associated with DNA-based schemes may confine their applicability to scenarios where security considerations take precedence over energy efficiency, or where frequent battery replacement or energy replenishment is feasible.

Moreover, the energy consumption results highlight the benefits of deploying cryptographic algorithms on higher-performance microcontrollers. Devices such as the ESP32, with superior processing capabilities and advanced power management features, can execute encryption operations more rapidly and with greater energy efficiency. These findings advocate for a co-design paradigm where the computational complexity of cryptographic algorithms is balanced against hardware capabilities to optimize overall system performance and energy efficiency.

4.3. Memory footprint

Memory footprint, comprising both read-only memory (ROM) for program storage and random-access memory (RAM) for runtime variables, is a critical constraint in IoT devices, which often feature limited memory capacity to reduce cost and power consumption. Table 2 illustrates a pronounced difference between classical lightweight and DNA-inspired cryptographic schemes: DNA-based algorithms typically require approximately twice the ROM and RAM compared to traditional lightweight ciphers. For example, the DNA-Crossover-Encode algorithm demands around 3.03 KB of ROM and 0.35 KB of RAM on the STM32F0 platform, whereas PRESENT-80 requires only 1.46 KB of ROM and 0.20 KB of RAM. This elevated memory usage primarily arises from the complex data structures and operations intrinsic to DNA cryptography, such as the emulation of nucleotide sequences and the inclusion of additional algorithmic processes like logical encoding and sequence crossover. Such increased memory requirements may restrict the deployment of DNA-based cryptography on ultra-constrained microcontrollers with limited flash and RAM capacities, which remain common in low-cost or legacy IoT devices. However, the trend toward incorporating more capable microcontrollers with larger memory banks in modern IoT edge devices could mitigate these limitations.

Therefore, memory footprint should be considered in conjunction with performance and security requirements during algorithm selection. Developers targeting highly resource-constrained platforms are advised to carefully evaluate these trade-offs and explore optimization strategies such as code size reduction, memory compression techniques, or the use of dedicated hardware accelerators to enable the practical implementation of DNA-based cryptographic schemes when their enhanced security properties justify the increased resource demands.

4.4. Throughput

Throughput, defined as the amount of data encrypted per unit time, directly influences a system's capacity to manage real-time data streams or large data volumes common in multimedia IoT applications and industrial automation. As presented in Table 3, classical lightweight ciphers such as SPECK-64 achieve the highest throughput values, reaching up to 1200 kB/s on the ESP32, while maintaining consistently high throughput across other platforms. This superior performance results from their low execution latency and computationally streamlined operations.

Although DNA-inspired algorithms generally exhibit lower throughput compared to classical ciphers—for example, Hybrid DNA-Logical Encoding achieves approximately 860 kB/s on the ESP32—they still offer throughput levels sufficient for numerous IoT use cases with moderate data rate requirements. This performance suggests that DNA-based cryptography may be well suited for scenarios such as periodic data encryption in environmental sensing or secure firmware updates, where encryption throughput demands are less stringent.

4.5. Security analysis

The security evaluation of the investigated lightweight cryptographic algorithms for deployment on resource-constrained microcontrollers reveals distinct trade-offs between computational efficiency, memory footprint, and resilience against cryptanalytic techniques. While our experimental work primarily focused on performance and energy metrics, security considerations remain an equally critical dimension in IoT deployments—where long-term data confidentiality, authentication integrity, and resistance to side-channel attacks must be guaranteed. This section synthesizes insights from our measurements with established cryptographic literature to provide a balanced security–efficiency perspective for the tested algorithms.

- PRESENT-80 – This block cipher, based on a substitution–permutation network (SPN) structure, was explicitly designed for minimal hardware area and low energy consumption, making it a common choice for RFID tags and sensor networks. It employs an 80-bit key size, sufficient to resist brute-force attacks in classical computing environments but lacking robustness against post-quantum adversaries. The cipher remains resilient to classical differential and linear cryptanalysis, although certain structural attacks have been documented in academic literature. Consequently, PRESENT is best suited for applications where ultra-low resource usage outweighs concerns about long-term quantum resistance.
- ASCON-128 – Recently standardized by NIST as a lightweight authenticated encryption scheme, ASCON utilizes a sponge construction combined with permutation-based design. It demonstrates strong resistance to differential, linear, and integral attacks, while providing robust authentication guarantees. Importantly for IoT, ASCON maintains this security profile alongside modest memory and energy demands, as corroborated by our measurements across all six microcontrollers. Its design inherently mitigates several classes of side-channel leakages through constant-time operations, though hardware-specific leakage assessments remain essential.
- SPECK-64 – Part of the SIMON–SPECK family, SPECK is optimized for software efficiency on constrained microcontrollers. Our benchmarks confirm its superior speed and energy efficiency; however, its security has been subject to academic scrutiny. Despite no practical full-round breaks to date, the cipher's origin and susceptibility to related-key attacks have raised concerns. Therefore, deployment in security-critical IoT systems should be carefully weighed against potential cryptanalytic developments.
- TWINE-80 – This lightweight Feistel network cipher achieves compact implementation and low energy consumption. TWINE exhibits resilience against conventional block cipher attacks at full rounds, although truncated differential attacks have been demonstrated on reduced-round variants. Our results highlight TWINE's competitive latency and energy profiles, positioning it as a suitable candidate for constrained devices where 80-bit security is deemed acceptable.
- HIGHT – Designed specifically for ultra-low power environments, HIGHT employs a generalized Feistel structure with a 64-bit block size and a 128-bit key. It resists classical cryptanalytic attacks and is noted for its exceptionally small hardware footprint. In our tests, HIGHT demonstrated excellent energy efficiency, albeit with lower throughput than SPECK or ASCON. This balance renders HIGHT appropriate for infrequent encryption tasks in battery-powered IoT devices.
- SIMON-64/128 – A hardware-oriented block cipher that balances strong security with low gate count. SIMON resists known differential and linear cryptanalysis on full rounds and offers a high security margin suitable for constrained environments. Its regular structure facilitates side-channel countermeasures; however, implementing constant-time software versions requires meticulous coding practices.
- LED-64 – An SPN-based cipher targeting extreme resource constraints. While offering compactness and low power consumption, the 64-bit key size is inadequate for high-security demands, and even the 128-bit key variant presents limited post-quantum resistance. Consequently, LED-64 is recommended primarily for short-lifetime IoT deployments where minimal hardware resources are paramount.

- Security–efficiency trade-offs
  Our measurements reveal that security and performance are not always positively correlated. Algorithms such as SPECK and TWINE excel in throughput and energy efficiency but generally provide lower perceived long-term cryptographic assurance compared to ASCON or SIMON. Conversely, PRESENT and HIGHT, although slower in raw speed, deliver low power consumption alongside sufficient classical security for many IoT scenarios.

For IoT deployments with expected lifespans exceeding a decade or exposure to emerging quantum threats, ASCON and SIMON emerge as the most robust choices among the tested algorithms. In ultra-constrained, short-lifetime devices, PRESENT, TWINE, or HIGHT may be preferred due to their compactness and energy efficiency.

While our experimental work primarily focused on performance and energy metrics, security considerations remain an equally critical dimension in IoT deployments—where long-term data confidentiality, authentication integrity, and resistance to side-channel attacks must be guaranteed. To provide empirical support for the diffusion and unpredictability of the DNA-based algorithms, we performed a basic avalanche test and calculated the Shannon entropy of the ciphertext. The avalanche test confirmed that a single-bit change in the plaintext altered approximately 50% of the ciphertext bits, and the entropy values were consistently close to the theoretical maximum for the given block size. These results indicate strong diffusion and a high degree of randomness in the DNA-based encryption outputs.

4.6. Key strengths and IoT suitability assessment of evaluated cryptographic algorithms

This section consolidates the primary advantages and overall suitability of each evaluated cryptographic algorithm for Internet of Things (IoT) environments. The key strengths and corresponding IoT suitability of each algorithm are summarized in Table 5.

4.7. Broader implications and future directions

The comprehensive cross-platform benchmarking presented herein elucidates the intricate interplay among algorithmic complexity, security features, and hardware constraints. The consistent relative performance ranking across diverse microcontrollers reinforces the generalizability of these findings, providing a robust basis for informed cryptographic algorithm selection in IoT design.

Traditional lightweight algorithms remain the preferred choice for highly constrained devices requiring low latency and minimal energy consumption, such as battery-powered sensors or actuators [50]. Conversely, DNA-based cryptography, with its novel security mechanisms and increased resource demands, offers promising opportunities as a complementary security layer in applications where enhanced cryptographic strength is critical and resource availability permits [51–53].

Future research should prioritize several key directions to enhance the practical applicability of DNA-based schemes:

1. Algorithmic optimization – Reduce computational overhead and memory consumption through software optimization, hardware accelerators, or dedicated instruction sets.
2. Rigorous cryptanalysis – Validate the security claims of DNA-inspired ciphers relative to classical algorithms and assess resilience against side-channel attacks.
3. Real-world deployment studies – Incorporate detailed power profiling, throughput analysis, and resilience testing under environmental stressors to ensure practical feasibility in diverse IoT ecosystems.
4. Hybrid frameworks – Explore combining classical lightweight algorithms with DNA-based schemes to achieve adaptive, layered security that balances energy efficiency, throughput, and cryptographic strength.
5. Design guidelines for IoT developers – Provide actionable recommendations for selecting appropriate cryptographic primitives based on device constraints, security requirements, and expected operational lifetime.

By addressing these directions, future work can bridge the gap between theoretical DNA-based cryptography and practical IoT deployment, enabling innovative, secure, and energy-efficient solutions for next-generation connected systems.

Table 5: Key strengths and IoT suitability.

Algorithm	Key Strengths	IoT Suitability Assessment
PRESENT-80	Compact design; low hardware footprint; proven resistance to standard attacks.	Highly suitable for ultra-constrained devices; excellent for battery-powered sensors.
ASCON-128	Strong security margin; AEAD capability; robust against differential and linear cryptanalysis.	Well-suited for applications requiring combined encryption and authentication in IoT.
SPECK-64	High speed on constrained CPUs; minimal RAM usage; simple ARX structure.	Appropriate for low-latency IoT communications with limited computational resources.
TWINE-80	Balanced performance; low gate count; efficient key schedule.	Suitable for mid-range IoT nodes requiring moderate throughput and strong security.
HIGHT	Optimized for low-power 8-bit processors; compact key expansion.	Effective for legacy and small-scale IoT devices with strict energy constraints.
SIMON-64/128	Flexible block/key sizes; strong resistance to known attacks; simple hardware implementation.	Applicable for both low-power and mid-tier IoT platforms with adaptable security needs.
LED-64	Very small footprint; suitable for RFID and NFC applications.	Best for extremely resource-limited IoT tags and identification devices.
DNA-XOR-Mutation	Novel DNA-based approach; good diffusion properties; high key sensitivity.	Potential for layered IoT security; suitable for hybrid encryption scenarios.
DNA-Substitution-Shift	Enhanced confusion; resistance to brute force due to large key space.	Feasible for IoT applications where encryption robustness outweighs minimal latency.
Hybrid DNA-Logical Encoding	Combines logical operations with DNA rules; improved avalanche effect.	Promising for secure IoT transmissions where innovative cryptographic diversity is valued.
DNA-Crossover-Encode	Efficient encoding structure; high resistance to statistical attacks.	Usable in IoT security layers requiring non-traditional encryption schemes.
DNA-Logical-Shift	Low-complexity DNA logic; suitable for constrained memory environments.	Applicable for IoT devices with ultra-low memory availability.
DNA-Hybrid-Crypt	Blends multiple DNA operations; versatile against multiple attack vectors.	Appropriate for niche IoT cases where unconventional ciphers deter targeted attacks.

5. Conclusion

This study shows a comprehensive empirical evaluation of a diverse set of lightweight cryptographic algorithms, encompassing both well-established classical ciphers and emerging DNA-inspired schemes, implemented on six representative microcontroller platforms commonly deployed in IoT environments. The results clearly demonstrate that classical lightweight algorithms such as SPECK-64 and SIMON-64/128 consistently outperform their counterparts in key performance metrics, achieving minimal execution latencies (as low as 0.70 ms per encryption round on the ESP32), reduced energy consumption (down to 3.0 µJ per round), and relatively modest memory footprints (approximately 1.17 KB ROM and 0.18 KB RAM). Conversely, DNA-based cryptographic schemes exhibit increased computational complexity, manifested by longer execution latencies (up to 3.35 ms on the ATmega328P) and elevated energy consumption (exceeding 8.0 µJ per round). Their memory requirements are substantially higher, often roughly double those of classical algorithms, which may limit deployment on ultra-constrained hardware. Nonetheless, these DNA-inspired algorithms introduce innovative security paradigms rooted in biological sequence operations, maintaining throughput levels adequate for many IoT applications and warranting further investigation regarding their robustness against advanced cryptanalytic techniques. The observed performance variability across different microcontrollers underscores the pivotal role of hardware architecture, clock frequency, and power management capabilities in shaping cryptographic efficiency. High-performance MCUs like the ESP32 and nRF52840 leverage advanced processing capabilities and optimized instruction pipelines to minimize both latency and energy consumption, whereas simpler MCUs impose more stringent limitations. These findings emphasize the necessity of a co-design approach that carefully aligns cryptographic algorithm selection

with the specific constraints and requirements of the target hardware and application domain. In sum, this work establishes a detailed benchmark framework for evaluating lightweight cryptography on constrained platforms, highlighting that while classical algorithms currently offer the best fit for resource-limited scenarios, DNA-based methods present promising directions for enhanced security in IoT contexts where moderate resource overheads are acceptable. Future research should focus on optimizing DNA-based algorithms through software and hardware acceleration, conducting rigorous security analyses, and validating their practical viability through real-world deployment and resilience testing within IoT ecosystems.

CRediT Authorship Contribution Statement

M. Ţălu: Writing – Review & Editing, Writing – Original Draft, Validation, Resources, Software, Methodology, Investigation, Formal analysis, Data Curation, Conceptualization, Project Administration.

Declaration of Competing Interest

The author declares that he has no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data Availability

All relevant data are within the manuscript and its supporting information files.

Declaration of Generative AI and AI-assisted Technologies in The Writing Process

The authors used generative AI to improve the writing clarity of this paper. They reviewed and edited the AI-assisted content and take full responsibility for the final publication.

References

[1] S. Satpathy, S. N. Mohanty, and J. M. Chatterjee, *Internet of Things and its Applications*. Cham: Springer International Publishing, 2022. doi: 10.1007/978-3-030-78071-0.

[2] R. Dallaev, T. Pisarenko, Ş. Ţălu, D. Sobola, J. Majzner, and N. Papež, "Current applications and challenges of the Internet of Things," *New Trends in Computer Sciences*, vol. 1, no. 1, pp. 51–61, 2023, doi: 10.3846/ntcs.2023.17891.

[3] M. Girard, "Standards for cybersecure IoT devices: a way forward," *JSTOR*, vol. 160, pp. 1–13, 2020.

[4] A. Nazarov, D. Nazarov, and Ş. Ţălu, "Information security of the Internet of Things," in *Proceedings of the International Scientific and Practical Conference on Computer and Information Security (INFSEC 2021)*, Yekaterinburg, Russia: SCITEPRESS, 2021, pp. 136–139.

[5] M. Ţălu, "Security and privacy in the IIoT: threats, possible security countermeasures, and future challenges," *Computing \& AI Connect*, vol. 2, pp. 1–12, 2025, doi: 10.69709/CAIC.2025.139199.

[6] C.-K. Wu, "Internet of things security," *Internet of Things Security*. Springer Nature Singapore Pte Ltd., pp. 137–154, 2021. doi: 10.1007/978-981-16-1372-2.

[7] M. Ţălu, "Cyberattacks and cybersecurity: concepts, current challenges, and future research directions," *Digital Technologies Research and Applications*, vol. 4, no. 1, pp. 44–60, 2025, doi: 10.54963/dtra.v4i1.919.

[8] C. Silpa, G. Niranjana, and K. Ramani, "Securing data from active attacks in IoT: an extensive study," *Proceedings of International Conference on Deep Learning, Computing and Intelligence*, vol. 1396. in Advances in Intelligent Systems and Computing, vol. 1396. Springer, Singapore, 2022. doi: 10.1007/978-981-16-5652-1_5.

[9] M. Ţălu, "Exploring machine learning algorithms to enhance cloud computing security," *Digital Technologies Research and Applications*, vol. 4, no. 2, pp. 33–47, 2025, doi: 10.54963/dtra.v4i2.1272.

[10] I. Radhakrishnan, S. Jadon, and P. Honnavalli, "Efficiency and security evaluation of lightweight cryptographic algorithms for resource-constrained IoT Devices," *Sensors*, vol. 24, no. 12, p. 4008, 2024, doi: 10.3390/s24124008.

[11] C. Paar, J. Pelzl, and T. Güneysu, *Understanding Cryptography: From Established Symmetric and Asymmetric Ciphers to Post-Quantum Algorithms*, 2nd ed. Springer Berlin Heidelberg, 2024, pp. 73–143. doi: 10.1007/978-3-662-69007-9.

[12] E. Prouff, G. Renault, M. Rivain, and C. OFlynn, Eds., *Embedded Cryptography 2*. London, UK: ISTE Ltd., 2025, pp. 177–245.

[13] B. Massimo, *Cryptography Algorithms*, 2nd ed. Birmingham, UK: Packt Publishing, 2024, pp. 35–104.

[14] 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. 28177–28193, 2021, doi: 10.1109/ACCESS.2021.3052867.

[15] P. S. Suryateja and K. V. Rao, "A survey on lightweight cryptographic algorithms in IoT," *Cybernetics and Information Technologies*, vol. 24, no. 1, 2024, doi: 10.2478/cait-2024-0002.

[16] Amrita, C. P. Ekwueme, I. H. Adam, and A. Dwivedi, "Lightweight cryptography for Internet of Things: a review," *EAI Endorsed Transactions on Internet of Things*, vol. 10, pp. 1–9, 2024.

[17] S. Darzi, K. Ahmadi, S. Aghapour, A. A. Yavuz, and M. M. Kermani, "Envisioning the Future of Cyber Security in Post-Quantum Era: A Survey on PQ Standardization, Applications, Challenges and Opportunities," *arXiv*, 2023, doi: 10.48550/arXiv.2310.12037.

[18] M. S. Turan *et al.*, "Status Report on the Final Round of the NIST Lightweight Cryptography Standardization Process," technical report 8454, 2023. doi: 10.6028/NIST.IR.8454.

[19] M. Rana, Q. Mamun, and R. Islam, “Lightweight cryptography in IoT networks: A survey,” *Future Generation Computer Systems*, vol. 129, pp. 77–89, 2022, doi: 10.1016/j.future.2021.11.011.

[20] R. Iqbal, N. M. Ansari, M. R. Awan, M. Ismail, and H. Gul, “Design and Evaluation of Lightweight Cryptographic Algorithms for Internet of Things (IoT) Devices: Achieving Optimal Trade-Offs Between Security, Computational Speed, and Energy Efficiency in Resource-Constrained Environments,” *The Progress: A Journal of Multidisciplinary Studies*, vol. 6, no. 1, pp. 85–99, 2025, doi: 10.71016/tp/smfybz24.

[21] G. Hatzivasilis, K. Fysarakis, I. Papaefstathiou, and C. Manifavas, “A review of lightweight block ciphers,” *Journal of Cryptographic Engineering*, vol. 8, no. 2, pp. 141–184, 2018, doi: 10.1007/s13389-017-0160-y.

[22] D. S. Nayancy and S. Chakraborty, “A survey on implementation of lightweight block ciphers for resource constraints devices,” *Journal of Discrete Mathematical Sciences \& Cryptography*, vol. 25, no. 5, pp. 1377–1398, 2020, doi: 10.1080/09720502.2020.1766764.

[23] S. M. Al-Nofaie, S. Sharaf, and R. Molla, “Design Trends and Comparative Analysis of Lightweight Block Ciphers for IoTs,” *Applied Sciences*, vol. 15, no. 14, p. 7740, 2025, doi: 10.3390/app15147740.

[24] M. Mondal and K. S. Ray, “Review on DNA cryptography,” *International Journal of Bioinformatics and Intelligent Computing*, vol. 2, no. 1, pp. 44–72, 2023, doi: 10.61797/ijbic.v2i1.198.

[25] S. Namasudra and G. C. Deka, *Advances of DNA Computing in Cryptography*, 1st ed. Chapman, Hall/CRC, 2018. doi: 10.1201/9781351011419.

[26] J. Gao and T. Xie, “DNA computing in cryptography,” *Advances in Computers*, vol. 129. pp. 83–128, 2023. doi: 10.1016/bs.adcom.2022.08.002.

[27] L. Chu, Y. Su, X. Yao, P. Xu, and W. Liu, “A Review of DNA Cryptography,” *Intelligent Computing*, vol. 4, p. 106, 2025, doi: 0000-0001-9091-3177.

[28] M. Ţălu, “DNA-based cryptography for internet of things security: concepts, methods, applications, and emerging trends,” *Buletin Ilmiah Sarjana Teknik Elektro*, vol. 7, no. 2, pp. 68–94, 2025, doi: 10.12928/biste.v7i2.12942.

[29] Q. Zhang, L. Guo, and X. Wei, “Image encryption using DNA addition combining with chaotic maps,” *Mathematical and Computer Modelling*, vol. 52, no. 11–12, pp. 2028–2035, 2010, doi: 10.1016/j.mcm.2010.06.005.

[30] T. Mandge and V. Choudhary, “A DNA encryption technique based on matrix manipulation and secure key generation scheme,” in *Proceedings of the International Conference on Information Communication and Embedded Systems (ICICES)*, 2013, pp. 47–52. doi: 10.1109/ICICES.2013.6508181.

[31] K. S. Mohamed, “New Trends in Cryptography: Quantum, Blockchain, Lightweight, Chaotic, and DNA Cryptography,” *New Frontiers in Cryptography*. 2020. doi: 10.1007/978-3-030-58996-7_4.

[32] Q. Liu, K. Yang, J. Xie, and Y. Sun, “DNA-Based Molecular Computing, Storage, and Communications,” *IEEE Internet of Things Journal*, vol. 9, no. 2, pp. 897–915, 2022, doi: 10.1109/JIOT.2021.3083663.

[33] K. Rarhi and S. Saha, “Image Encryption in IoT Devices Using DNA and Hyperchaotic Neural Network,” *Design Frameworks for Wireless Networks*, vol. 82. 2020. doi: 10.1007/978-981-13-9574-1_15.

[34] M. Imdad, A. Fazil, S. N. B. Ramli, J. Ryu, H. B. Mahdin, and Z. Manzoor, “DNA-PRESENT: An Improved Security and Low-Latency, Lightweight Cryptographic Solution for IoT,” *Sensors*, vol. 24, no. 24, p. 7900, 2024, doi: 10.3390/s24247900.

[35] S. Ali and F. Anwer, “DNA-Based Elliptic Curve Cryptography for Data Security in IoT,” *Advanced Network Technologies and Intelligent Computing (ANTIC 2023)*, vol. 2090. 2024. doi: 10.1007/978-3-031-64076-6_25.

[36] A. Bogdanov *et al.*, “PRESENT: An ultra-lightweight block cipher,” *Cryptographic Hardware and Embedded Systems – CHES 2007*, vol. 4727. in Lecture Notes in Computer Science, vol. 4727. Springer, Berlin, Heidelberg, pp. 450–466, 2007. doi: 10.1007/978-3-540-74735-2_31.

[37] International Organization for Standardization, “ISO 29192-2:2012(E) Information technology – Security techniques – Lightweight cryptography – Part 2: Block ciphers.” 2012.

[38] C. Dobraunig, M. Eichlseder, F. Mendel, and M. Schl\äffer, “ASCON v1.2: Lightweight Authenticated Encryption and Hashing,” *Journal of Cryptology*, vol. 34, p. 33, 2021, doi: 10.1007/s00145-021-09398-9.

[39] M. Mart\ín-Gonz\ález, E. Tena-S\ánchez, F. E. Potestad-Ord\ó\ñez, and A. J. Acosta, “Detailed Assessment of Hardware Implementations, Attacks and Countermeasures for the Ascon Authenticated Cipher,” *Electronics Letters*, vol. 61, no. 1, p. e70260, 2025, doi: 10.1049/ell2.70260.

[40] J. Kaur, M. M. Kermani, and R. Azarderakhsh, “Hardware Constructions for Error Detection in Lightweight Authenticated Cipher Ascon Benchmarked on FPGA,” *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 4, pp. 2276–2280, 2022, doi: 10.1109/TCSII.2021.3136463.

[41] S. Khan, W. K. Lee, and S. O. Hwang, “Scalable and Efficient Hardware Architectures for Authenticated Encryption in IoT Applications,” *IEEE Internet of Things Journal*, vol. 8, no. 14, pp. 11260–11275, 2021, doi: 10.1109/JIOT.2021.3052184.

[42] S. Khan, W. K. Lee, and S. O. Hwang, “Evaluating the Performance of Ascon Lightweight Authenticated Encryption for AI-Enabled IoT Devices,” in *2022 TRON Symposium (TRONSHOW)*, 2022, pp. 1–6. doi: 10.1109/10024417.

[43] P. Joshi and B. Mazumdar, “Subset Fault Analysis of Ascon-128 Authenticated Cipher,” *Microelectronics Reliability*, vol. 123, p. 114155, 2021, doi: 10.1016/j.microrel.2021.114155.

[44] S. C. You, M. G. Kuhn, S. Sarkar, and F. Hao, “Low Trace-Count Template Attacks on 32-Bit Implementations of Ascon AEAD,” *IACR Transactions on Cryptographic Hardware and Embedded Systems*, vol. 2023, no. 4, pp. 344–366, 2023, doi: 10.46586/tches.v2023.i4.344-366.

[45] K. Ramezanpour, P. Ampadu, and W. Diehl, “Scarl: Side-Channel Analysis With Reinforcement Learning on the Ascon Authenticated Cipher,” *arXiv*, 2020, doi: 10.48550/arXiv.2006.03995.

[46] A. V. Duka and B. Genge, “Implementation of SIMON and SPECK lightweight block ciphers on programmable logic controllers,” in *2017 5th International Symposium on Digital Forensic and Security (ISDFS)*, Târgu Mureş, Romania, 2017, pp. 1–6. doi: 10.1109/ISDFS.2017.7916501.

[47] K. Sakamoto *et al.*, “Tweakable TWINE: Building a Tweakable Block Cipher on Generalized Feistel Structure,” *IEICE Transactions on Fundamentals*, no. 12, pp. 1629–1639, 2020, doi: 10.1587/transfun.2019EAP1141.

[48] B. Kim, J. Cho, B. Choi, J. Park, and H. Seo, “Compact implementations of HIGHT block cipher on IoT platforms,” *Security and Communication Networks*, 2019, doi: 10.1155/2019/5323578.

[49] L. Dong, H. Zhang, L. Zhu, S. Sun, H. Gan, and F. Zhang, “Analysis of an Optimal Fault Attack on the LED-64 Lightweight Cryptosystem,” *IEEE Access*, vol. 7, pp. 31656–31662, 2019, doi: 10.1109/ACCESS.2019.2901753.

[50] M. Ţălu, “Securing IoT ecosystems: a review of modern lightweight cryptographic algorithms and their performance,” *Journal of Cyber Security*, vol. 7, no. 1, 2025, doi: 10.32604/jcs.2025.073690.

[51] Ş. Ţălu, “Hydraulic systems security: addressing cyber threats with DNA-based cryptography in cloud-integrated control,” *HIDRAULICA*, vol. 3, pp. 21–32, 2025.

[52] I. Qiqieh, J. Alzubi, and O. Alzubi, “DNA cryptography based security framework for health-cloud data,” *Computing*, vol. 107, p. 35, 2025, doi: 10.1007/s00607-024-01393-9.

[53] H. Sharma and S. Kaur, "Quantum-Inspired Hyperchaotic Bio-DNA Image Encryption for Real-Time Medical Security," *IEEE Access*, vol. 13, pp. 184583–184601, 2025, doi: 10.1109/ACCESS.2025.3625765.