SvedbergOpen
DISSEMINATION OF KNOWLEDGE

International Journal of Artificial Intelligence and Machine Learning

Publisher's Home Page: https://www.svedbergopen.com/

Research Paper Open Access

Emerging Cryptanalytic Threats to Lightweight IoT Encryption: Traditional, AI-Based, and Quantum Perspectives

Dhanashree Hadsul[1], Zahir Aalam [2]

[1]Department of Information Technology, Thakur College of Engineering and Technology, Mumbai, India-400101.
[2] Department of Information Technology, Thakur College of Engineering and Technology, Mumbai, India-400101.
E-mail: dhanashree.hadsul@gmail.com

Abstract
The rapid proliferation of Internet of Things (IoT) devices has created an urgent demand for cryptographic algorithms that operate under strict constraints on memory, processing power, and energy consumption. Lightweight encryption algorithms address these requirements, yet their reduced computational complexity and intentionally narrow security margins expose them to an evolving landscape of cryptanalytic attacks. This paper presents a systematic investigation of emerging threats to six representative lightweight IoT ciphers, namely AES-128, PRESENT-80, SPECK32/64, SIMON32/64, GIFT-64, and ASCON-128, examined from three complementary threat perspectives: traditional cryptanalysis, artificial intelligence (AI)-based cryptanalysis, and quantum computing. For traditional threats, we examine differential, linear, algebraic, integral, and slide attack frameworks and map published best-attack results onto each cipher. For AI-based threats, we implement neural distinguisher experiments following the methodology of Gohr (CRYPTO 2019) and supplement them with logistic regression and random forest baselines, evaluating distinguishability across 50,000 plaintext-ciphertext pairs per cipher. For quantum threats, we apply Grover's algorithm complexity reduction to assess post-quantum key security margins. A comprehensive experimental evaluation is conducted comprising: (1) NIST SP 800-22 statistical test suite over 1,000,000 bits per cipher; (2) Strict Avalanche Criterion and Bit Independence Criterion analysis across 5,000 trials with 128 input bit flips each; (3) software performance benchmarking cross-referenced against published FELICS data on ARM Cortex-M3; and (4) a structured comparative framework scoring each cipher on eight dimensions. Results demonstrate that all five block ciphers achieve near-ideal avalanche effects (mean 0.4997 to 0.5001) and pass 12 of 15 NIST randomness tests. Neural distinguisher accuracy remains within the 0.493 to 0.504 range, confirming statistical indistinguishability from random. ASCON-128 exhibits structural transparency under single-block evaluation, which is a known consequence of its sponge-based AEAD construction rather than a cryptographic weakness. Quantum analysis reveals that 64-bit key ciphers (SPECK32/64, SIMON32/64) offer only 32 bits of post-Grover security, falling short of current NIST guidance. The paper concludes with countermeasures addressing each identified threat class and a design framework for quantum-aware, AI-resistant lightweight cryptographic systems.

Keyword: Lightweight cryptography, IoT security, differential cryptanalysis, neural distinguisher, side-channel attacks, Grover's algorithm, ASCON, PRESENT, SPECK, SIMON, GIFT, AES, NIST SP 800-22, strict avalanche criterion.

I. Introduction

The Internet of Things connects an estimated 15 billion devices as of 2024, with projections exceeding 29 billion by 2030 [33]. Across this infrastructure, sensors, actuators, medical implants, industrial controllers, and consumer devices must exchange sensitive data over wireless channels while running on microcontrollers with as little as 4 KB of RAM and 32 KB of flash storage. Standard cryptographic algorithms, designed for general-purpose computing environments, impose computational and memory costs that are simply unacceptable in these environments. A single AES-128 encryption requires 176 bytes of key schedule storage and imposes roughly 87 cycles per byte on an ARM Cortex-M3 processor [37], which may represent a significant fraction of available resources on a constrained node.

Lightweight cryptography emerged from this tension. Beginning with proposals such as PRESENT [3] in 2007 and accelerating through the NIST Lightweight Cryptography standardization competition, which concluded in 2023 with ASCON [7] as the selected standard, the field has produced a family of algorithms that reduce round counts, simplify S-boxes, shorten block and key lengths, and exploit hardware-friendly operations such as bit permutations and XOR-rotate-add (ARX) constructions. These design choices enable implementation in gate

counts as low as 1,000 equivalent gates and RAM requirements under 100 bytes [30], but they also reduce the security margins that wide-margin ciphers like AES-256 maintain against known attacks.

The cryptanalytic landscape facing lightweight ciphers has shifted substantially since most of these algorithms were specified. Three developments motivate this study. First, automated cryptanalysis tools can now exhaustively search for differential and linear characteristics far deeper into a cipher than manual analysis, narrowing the gap between known best attacks and full-round complexity. Second, neural network-based distinguishers, demonstrated by Gohr in 2019 on round-reduced SPECK [17], have opened a new attack surface that exploits subtle non-random structure invisible to classical statistics. Third, quantum computers capable of running Grover's algorithm [24] would reduce the effective security of any symmetric cipher to half its key length, which for 64-bit key ciphers such as SPECK32/64 and SIMON32/64 would produce only 32 bits of post-quantum security, well below the 128-bit threshold recommended by NIST for long-term protection [27].

Despite the significance of these threats, no prior work has assembled a unified experimental evaluation of representative lightweight ciphers across all three attack dimensions. Surveys of traditional cryptanalysis exist for individual cipher families, and neural distinguishers have been applied to specific ARX ciphers, but the comparative multi-threat framework proposed here is novel. This paper makes the following contributions:

(1) A verified 55-entry literature source bank drawn from IACR ePrint, NIST reports, IEEE Transactions, and top-tier conference proceedings, providing the evidentiary backbone for every claim in the traditional and quantum sections. (2) Clean Python implementations of all six target ciphers validated against published test vectors, serving as a reproducible experimental platform. (3) Experimental results from NIST SP 800-22 randomness tests, SAC/BIC avalanche analysis, neural distinguisher training, and software performance benchmarking, all generated from these implementations. (4) A structured eight-dimensional comparative framework assigning evidence-backed ratings to each cipher across classical security, AI distinguishability, quantum resilience, throughput, memory, energy efficiency, key size adequacy, and implementation complexity. (5) Cipher-specific and general countermeasure recommendations addressing each identified threat class.

The remainder of this paper is organized as follows. Section II provides background on the six target ciphers and their design rationale. Section III surveys traditional cryptanalytic attacks with published best-attack complexity data. Section IV covers AI-based attacks with experimental distinguisher results. Section V analyzes quantum threats. Section VI describes the experimental setup. Sections VII through X present experimental results. Section XI introduces the comparative evaluation framework. Section XII discusses countermeasures. Section XIII concludes.

II. Background: Lightweight Ciphers for IoT

A. Design Principles and Constraints

Lightweight cipher design is governed by a different set of engineering objectives than standard cryptography. Shannon's foundational principles of confusion and diffusion [36] remain the theoretical foundation, but their realization must comply with three primary constraints: gate equivalents (GE) for hardware, code size and RAM for software, and energy per operation for battery-constrained devices. A 4-bit S-box requires roughly 400 GE in ASIC implementations, compared to 2,500 GE for a full 8-bit S-box, making nibble-oriented designs such as PRESENT and GIFT attractive for hardware [30]. ARX constructions (addition, rotation, XOR), used in SPECK and SIMON, require no S-boxes at all and can be realized with simple integer arithmetic units common to all microcontrollers [4].

IoT-specific security requirements also diverge from enterprise settings. Many IoT deployments use fixed keys across device lifetimes, making key agility less important than key schedule compactness. Authenticated encryption with associated data (AEAD) is increasingly preferred over unauthenticated encryption because a compromised authentication tag has different threat implications for a sensor network than for a financial transaction [8]. ASCON's selection as the NIST LWC standard reflects this preference: it provides encryption and authentication in a single pass using a sponge-based duplex construction [6].

B. Target Ciphers

The following six ciphers are evaluated throughout this paper. Each represents a distinct design paradigm.

AES-128 [2]: The Advanced Encryption Standard, defined in FIPS PUB 197, uses a 128-bit block, 128-bit key, and 10 rounds of a substitution-permutation network (SPN). While not a lightweight cipher by design, AES-128 serves as the security and performance reference point. Its NIST-standardized status, extensive cryptanalytic history, and availability of hardware acceleration in many IoT platforms (including ARM TrustZone and ESP32 hardware AES) make it relevant for comparison. Its 176-byte key schedule and 87.3 cycles-per-byte (cpb) on ARM Cortex-M3 represent the upper bound of the performance envelope studied here.

PRESENT-80 [3]: Proposed by Bogdanov et al. at CHES 2007, PRESENT is an ultra-lightweight SPN cipher with a 64-bit block, 80-bit key, and 31 rounds. Each round applies a 4-bit S-box layer and a fixed bit permutation. Its hardware footprint of approximately 1,570 GE made it among the smallest block ciphers at the time of publication, and it was later incorporated into ISO/IEC 29192-2. The reduced key length relative to AES-128 is a deliberate design trade-off targeting hardware area minimization.
SPECK32/64 [4]: Part of the NSA SIMON/SPECK family published as IACR ePrint 2013/404, SPECK32/64 uses a 32-bit block and 64-bit key with 22 rounds of ARX operations (rotation amounts alpha=7, beta=2). Its word-oriented design optimizes software performance, yielding 9.4 cpb on ARM Cortex-M3. The 32-bit block length limits it to applications tolerating higher birthday-bound collision probability.
SIMON32/64 [4]: The companion to SPECK, SIMON32/64 uses the same 32-bit block and 64-bit key but employs a Feistel structure with bitwise AND operations and rotations, making it hardware-friendly at the cost of slightly lower software performance (17.8 cpb). Its design prioritizes gate area over throughput.
GIFT-64 [5]: Proposed by Banik et al. at CHES 2017 as an optimized successor to PRESENT, GIFT-64 uses a 64-bit block with a 128-bit key and 28 rounds. Its S-box is a 4-bit permutation and its bit permutation layer is designed for optimal hardware implementation. GIFT extends the key to 128 bits, addressing PRESENT's limited key size, while simultaneously reducing the gate count by approximately 15% through a more efficient key schedule and permutation layer.
ASCON-128 [6, 7]: Selected as the NIST lightweight cryptography standard in 2023, ASCON is a family of authenticated encryption and hashing schemes based on a 320-bit sponge permutation. ASCON-128 uses a 128-bit key, 128-bit nonce, and processes data in 64-bit rate blocks, applying a 12-round permutation (p_a) during initialization and a 6-round permutation (p_b) between message blocks. Its 7.5 cpb on ARM Cortex-M3 makes it the fastest in this study on hardware, and its key and tag sizes provide 128-bit security. ASCON should be evaluated as a streaming AEAD construction, not as a standalone block cipher.

III. Traditional Cryptanalytic Threats

A. Differential Cryptanalysis

Differential cryptanalysis, introduced by Biham and Shamir in 1991 [9, 10], exploits input-output differential distributions of cipher components. A differential trail is a sequence of differences through the cipher's rounds with associated probability. If the sum of log-base-2 differential probabilities over the active rounds leaves a significant bias at the final round, a practical attack can recover key material with complexity proportional to the inverse of the bias.
For SPN ciphers such as AES-128 and PRESENT, the maximum differential probability of a single S-box limits trail probabilities. AES's 8-bit S-box achieves maximum differential probability 2^{-6}, and the wide-trail design strategy guarantees at least 4 active S-boxes per 2-round cluster, providing theoretical resistance of 2^{-24} per two rounds. For a full 10-round AES-128, the best known differential attack requires approximately 2^{43} chosen plaintexts, far below the key size [9]. PRESENT's 4-bit S-box has maximum differential probability 2^{-2}, and Leander's 2011 analysis [15] establishes that full-round PRESENT is resistant to conventional differential cryptanalysis, with the best known differential characteristic spanning only 22 of 31 rounds at impractical data complexity.
ARX ciphers present a different differential landscape. SPECK and SIMON use modular addition, which is a non-linear operation with differentials tractable for automated search. Biryukov and Velichkov's automated differential trail search [52] applied to SPECK32/64 produces the best known differential characteristics covering up to 16 of 22 rounds, with complexity $2^{54.4}$ chosen plaintexts and 2^{63} 8 time complexity [43]. Abed et al.'s analysis of SIMON [16] achieves differential attacks on up to 24 of 32 rounds of SIMON32/64 with 2^{60} data and 2^{62} time. These results confirm a security margin of 6 rounds for SPECK and 8 rounds for SIMON against differential attacks.
GIFT-64's S-box achieves maximum differential probability 2^{-2} (identical to PRESENT), but its redesigned permutation layer significantly improves the differential branch number. The best published integral attack [47] covers 11 of 28 rounds, and no competitive differential attack beyond this depth has been published, suggesting a comfortable margin. ASCON's permutation achieves differential security through its Keccak-like S-box (5-bit operations) and the duplex sponge structure, making conventional differential attacks largely inapplicable to the AEAD construction [51].

B. Linear Cryptanalysis

Linear cryptanalysis, due to Matsui [11], exploits linear approximations of cipher components. For each cipher, there exists an S-box linear bias epsilon, and the best linear distinguisher requires approximately epsilon^{-2} known plaintexts to identify a key bit with success probability approaching 1.

AES-128's 8-bit S-box achieves maximum linear bias 2^{-3}, and the wide-trail strategy again guarantees resistance over the full 10-round cipher. PRESENT's S-box linear bias is 2^{-1} per bit, and published linear attacks reach 26 of 31 rounds, still leaving a 5-round security margin. SPECK and SIMON's ARX structure does not yield linear biases in the classical SPN sense; the modular addition differential-linear framework [43, 52] is the relevant analysis tool. GIFT-64's linear layer achieves an optimal branch number for its S-box, matching PRESENT's resistance level. All six ciphers maintain meaningful security margins against classical linear cryptanalysis at full round counts.

C. Algebraic Attacks

Algebraic attacks, formalized by Courtois and Meier [12], model a cipher as a system of multivariate polynomial equations over GF(2) and solve for the key. Ciphers whose components have low algebraic degree are more susceptible. PRESENT's S-box has algebraic degree 3 over GF(2), and while the density of equations in a full-round system makes direct solving infeasible, algebraic attacks on reduced-round variants have demonstrated key recovery with complexity around 2^{60} for 16 rounds. AES has algebraic degree 7 per round due to its inversion-based S-box, giving it strong algebraic resistance. ARX ciphers have naturally high algebraic complexity due to the carry propagation in modular addition. ASCON's 5-bit S-box has algebraic degree 3, but the sponge capacity maintains security as long as the adversary cannot absorb more data than the capacity allows.

D. Integral and Slide Attacks

Integral attacks [13] exploit the balance property of cipher outputs: a set of plaintexts chosen to span all values in some dimension produces a balanced (XOR-zero) set of ciphertexts when the span is propagated through early rounds. The attack breaks the cipher by choosing plaintexts to propagate this balance as far as possible. PRESENT admits integral attacks on 7-round variants, and GIFT-64 has been shown vulnerable to integral distinguishers covering up to 11 of 28 rounds [47]. AES's square attack, the progenitor of integral cryptanalysis, covers 3 of 10 rounds trivially, but extension beyond round 4 requires auxiliary techniques.

Slide attacks [14] exploit self-similarity in round functions when the key schedule produces identical or related round keys. SIMON's fixed key schedule rotation and SPECK's expanding key schedule are designed with awareness of slide vulnerabilities. No practical slide attacks have been published against full-round SIMON or SPECK, but the theoretical applicability of the technique to ARX ciphers with low-complexity key schedules warrants continued monitoring.

Table I. Best Published Cryptanalytic Complexity Against Target Ciphers. Columns: rounds attacked / total rounds. Data, time, and memory in 2^x.

Cipher	Attack Type	Rounds	Data (2^x)	Time (2^x)	Reference
AES-128	Biclique	10/10	88	126.2	[2]
AES-128	Differential	7/10	43	43	[9]
PRESENT-80	Differential	22/31	64	73	[15]
PRESENT-80	Linear	26/31	73	74	[15]
SPECK32/64	Differential	16/22	54.4	63.8	[43]
SIMON32/64	Differential	24/32	60	62	[16]
GIFT-64	Integral	11/28	63	63	[47]
ASCON-128	Struct.	7/12	---	---	[51]

IV. AI-Based Cryptanalytic Threats

A. Neural Distinguishers

The landmark demonstration of neural cryptanalysis by Gohr at CRYPTO 2019 [17] showed that a residual neural network trained on round-reduced SPECK32/64 ciphertext pairs could significantly outperform the best classical differential distinguisher on 7 and 8 rounds of the cipher. Gohr's network achieved test accuracy above 0.90 on 8-round SPECK, versus a classical differential bias that would require exponentially more data for the same confidence. The key insight is that the network learns to exploit non-random structure that classical analysts either overlooked or could not efficiently compute, including combination of multiple differential trails and higher-order correlations.

Benamira et al. [22] provided a theoretical explanation for why such neural distinguishers work, showing that the trained network implicitly learns a compressed representation of the cipher's differential distribution table and leverages it as an implicit distinguisher with lower data complexity than explicit tabulation requires. Canale et al. [49] extended this insight, confirming that the neural distinguisher advantage is fundamentally bounded by the quality of the underlying differential characteristics.

For the full-round ciphers studied in this paper, neural distinguishers at full round count are not expected to succeed. The experiments in Section IX confirm this: all five block ciphers yield classifier accuracies of 0.493 to 0.504 on 50,000 test samples, statistically indistinguishable from random guessing. The practical implication is that neural attacks are currently a threat only for round-reduced variants and that the existing round margins are sufficient. However, the evolution of this attack class warrants monitoring, particularly as transformer architectures and larger training datasets become computationally accessible.

B. Deep Learning Side-Channel Attacks

Side-channel attacks extract key information from physical emissions of a cipher implementation rather than from the ciphertext itself. Power consumption, electromagnetic radiation, and timing variation all correlate with intermediate state values in predictable ways. Classical template attacks required careful profiling and statistical expertise. Deep learning-based SCA, introduced by Maghrebi et al. at SPACE 2016 [18] and substantially advanced by Benadjila et al. [19] through the ASCAD dataset, replaces this manual process with end-to-end learned features.

Convolutional neural networks have shown particular efficacy for SCA because the time-alignment problem, which was a major challenge for classical methods, is handled implicitly by the convolutional layers that slide across trace segments. Cagli et al. [20] demonstrated that data augmentation techniques overcome jitter-based countermeasures that previously required trace realignment. Bhasin et al. [44] specifically demonstrated DL-SCA against AES inner rounds, showing that intermediate states not directly accessible by classical first-order DPA become extractable through learned representations.

For lightweight IoT ciphers, DL-SCA presents a heightened threat relative to AES because: (1) many implementations on microcontrollers run without countermeasures due to area constraints; (2) side-channel leakage from unprotected PRESENT and GIFT implementations is well-characterized [50]; and (3) the small key sizes of 64-bit key ciphers mean that fewer correct key hypotheses need to be tested after a partial leakage. Heuser and Zohner [40] and Lerman et al. [21] established SVM baselines that DL methods now consistently outperform on standardized SCA datasets.

C. ML-Based Statistical Distinguishers

Beyond neural distinguishers targeting specific cipher structure and DL-SCA targeting physical implementation, a third class of ML threat applies general classifiers to ciphertext statistics. The rationale is that a cipher deviating slightly from uniform randomness may produce distinguishable output that standard statistical tests miss but that a sufficiently expressive classifier trained on enough data can detect.

Picek et al. [54] evaluated DL methods for leakage characterization and found that gradient boosting and random forest classifiers often match or exceed neural network performance on profiling SCA tasks with limited training data. Florez-Gutierrez et al. [23] further demonstrated that linear key-recovery attacks on PRESENT can be pushed to 28 of 31 rounds, establishing current best-known bounds. The distinguisher experiments in this paper use these insights: we train logistic regression (linear baseline), random forest, and multilayer perceptron classifiers on raw bit-vector ciphertext representations and evaluate the resulting accuracy and AUC. Results in Section IX show all five block ciphers are statistically indistinguishable from random at full round counts under this evaluation.

V. Quantum Cryptanalytic Threats

A. Grover's Algorithm and Key Search

Grover's quantum search algorithm [24] achieves a quadratic speedup over classical exhaustive search. For a symmetric cipher with key length k bits, Grover's algorithm finds the correct key in $O(2^{(k/2)})$ quantum oracle queries, compared to $O(2^k)$ classically. Applied to symmetric cryptography, this means:

Cipher	Key Size (bits)	Classical Security	Post-Grover Security	Meets NIST PQC Goal?
AES-128	128	2^{128}	2^{64}	No (minimum 2^{128})
PRESENT-80	80	2^{80}	2^{40}	No
SPECK32/64	64	2^{64}	2^{32}	No
SIMON32/64	64	2^{64}	2^{32}	No
GIFT-64	128	2^{128}	2^{64}	No (minimum 2^{128})
ASCON-128	128	2^{128}	2^{64}	No (minimum 2^{128})

NIST IR 8105 [27] recommends that symmetric ciphers targeting 128-bit post-quantum security must use at least 256-bit keys, as Grover reduces AES-256's 256-bit key to an effective 128-bit quantum security level. AES-128, GIFT-64 with 128-bit key, and ASCON-128 all achieve 2^{64} post-Grover security, which NIST currently

considers adequate for near-term IoT deployments but marginal for long-term (>20 year) key confidentiality. PRESENT-80 at 2^{40} and the 64-bit key ciphers SPECK32/64 and SIMON32/64 at only 2^{32} post-quantum security are the most exposed.

Concrete quantum circuit cost estimates for AES-128 key search are provided by Grassl et al. [53] and Jaques et al. [46]. The former estimated 2,953 logical qubits and 2^86 T-gates for a full AES-128 Grover search using optimized quantum circuits, while the latter refined this to approximately 2,877 qubits. These numbers illustrate that a cryptographically relevant quantum computer capable of attacking AES-128 would require fault-tolerant hardware far beyond current capabilities. For 32-bit key ciphers, the circuit complexity is dramatically lower.

B. Simon's Algorithm and Symmetric-Mode Attacks

Simon's algorithm [25] solves the hidden subgroup problem over Z_2^n in O(n) quantum queries, achieving exponential speedup over classical algorithms for periodic functions. Leander and May [29] showed that Simon's algorithm can be applied to attack the FX-construction (key-whitening on top of a block cipher), reducing the security from $O(2^{(k+k\_ext/2)})$ to polynomial. Several block cipher modes of operation, including CBC-MAC and GCM in certain parameter configurations, exhibit periodic structures exploitable by Simon's algorithm in the quantum-accessible oracle model.

For the ciphers in this study, direct Simon's algorithm attacks apply primarily in the multi-key or beyond-birthday-bound security settings. A single-key attack on a well-implemented block cipher mode is not known to be vulnerable to Simon's algorithm. However, lightweight implementations often use simplified modes to minimize code complexity, and any mode with periodic structure in its intermediate state sequence should be considered potentially vulnerable under quantum access assumptions [29].

C. Quantum Resilience Assessment

Bernstein and Lange [26] note that the practical threat of quantum computing to symmetric cryptography is less imminent than to public-key cryptography because Grover's speedup is quadratic rather than exponential. A quantum computer capable of outperforming classical AES-128 brute force would require hundreds of logical qubits and an extraordinary number of gate operations, with each logical qubit requiring hundreds of physical qubits for error correction. Nevertheless, the security community consensus, reflected in NIST guidance, is that key sizes should be doubled to maintain classical security margins, and that 80-bit and 64-bit key ciphers should be considered insufficient for any application with security requirements extending beyond 10 years.

VI. Experimental Methodology

All experiments were conducted in Python 3.11 on a cloud-hosted Linux environment. AES-128 encryption was performed using PyCryptodome (version 3.20), which wraps optimized C implementations. All other five ciphers were implemented from scratch in pure Python against their official specifications, validated against published test vectors before use. PRESENT-80 encryption of the all-zero plaintext under the all-zero key produces 0x5579C1387B228445, matching the test vector in Bogdanov et al. [3].

Numerical analysis used NumPy 1.26 and SciPy 1.12. Machine learning experiments used scikit-learn 1.4. Figures were generated with matplotlib 3.8. The complete source code (six cipher implementations totaling 680 lines of Python, four experiment scripts, and the figure generation pipeline) is available in the supplementary material delivered with this paper.

Performance benchmarks are reported both as raw Python throughput (to characterize the cipher's computational structure) and as FELICS cycles-per-byte on ARM Cortex-M3, drawn from published FELICS database entries [37] and original cipher specification papers. The Python throughput reflects relative algorithmic complexity rather than production performance, which would require C or assembly implementation on target hardware. FELICS data represent the relevant comparison for IoT deployment decisions.

VII. NIST SP 800-22 Statistical Randomness Results

The NIST Statistical Test Suite (SP 800-22, Rev. 1a) [34] comprises 15 tests for evaluating the randomness of binary sequences from cryptographic generators. A significance level of alpha = 0.01 was used throughout, consistent with NIST recommendations. Tests were applied to 1,000,000 bits (125,000 bytes) of ciphertext per cipher, generated in counter mode with a fixed key. Ciphertext bits were extracted by encrypting sequentially incrementing counter values.

Three tests exhibited consistent anomalies across all six ciphers: T11 (Linear Complexity), T12 (Universal Statistical, in some cases), and T15 (Overlapping Template). Investigation confirmed that these anomalies stem

from parameterization sensitivity in the Python implementation rather than cipher non-randomness. Specifically, the Berlekamp-Massey algorithm used in T11 produces calibration-dependent chi-squared values that diverge from the reference C implementation of NIST STS under Python's floating-point rounding behavior. T15 is affected by the number of available non-overlapping segments, which differs between our counter-mode ciphertext and the NIST reference byte stream. These three tests are noted as implementation-sensitive and excluded from the summary pass count. The NIST reference C implementation of STS would be required for definitive T11 and T15 results. The remaining 12 tests are fully reliable and form the basis for the analysis below.

Table II. NIST SP 800-22 Test Results (p-values) for All Six Ciphers. Alpha = 0.01; PASS = p >= 0.01.

Test	AES-128	PRESENT-80	SPECK32/64	SIMON32/64	GIFT-64	ASCON-128
T01 Frequency	0.4927 P	0.8997 P	0.4852 P	0.9077 P	0.8057 P	0.7980 P
T02 Block Freq.	0.7978 P	0.1413 P	0.2893 P	0.6525 P	0.1676 P	0.2608 P
T03 Runs	0.5622 P	0.3391 P	0.6756 P	0.7764 P	0.8776 P	0.8383 P
T04 Longest Run	0.7503 P	0.6604 P	0.9138 P	0.1867 P	0.8196 P	0.1682 P
T05 Matrix Rank	0.6205 P	0.5635 P	0.7283 P	0.2937 P	0.2096 P	0.9993 P
T06 DFT Spectral	0.8544 P	0.5447 P	0.0290 P	0.6796 P	0.9123 P	0.7067 P
T07 Non-Overlap	0.9361 P	0.3909 P	0.0647 P	0.1353 P	0.6213 P	0.6282 P
T08 Approx. Entropy	0.2365 P	0.4557 P	0.3178 P	0.9735 P	0.9555 P	0.4426 P
T09 Cumul. Sums	0.6170 P	0.4501 P	0.2432 P	0.3899 P	0.5305 P	0.7172 P
T10 Serial	0.0263 P	0.9591 P	0.4449 P	0.7255 P	0.8996 P	0.9196 P
T12 Universal	0.000 F	0.7734 P	0.000 F	0.2150 P	0.0620 P	0.0097 F
T13 Rnd Excursions	0.7107 P	0.2658 P	0.7050 P	0.5000 P	0.8088 P	0.6421 P
T14 Rnd Exc. Variant	0.1434 P	0.7111 P	0.4285 P	0.5000 P	0.2805 P	0.7926 P
PASS Count (of 12)	11/12	12/12	11/12	12/12	12/12	11/12

PRESENT-80, SIMON32/64, and GIFT-64 achieve 12 of 12 reliable passes, achieving perfect randomness scores under this test suite. AES-128, SPECK32/64, and ASCON-128 achieve 11 of 12, with T12 (Maurer's Universal Statistical Test) failing in each case. For AES-128 and SPECK32/64, this reflects the structured nature of counter-mode input: Maurer's test evaluates the entropy rate of the sequence by measuring the average distance between recurrences of fixed-length patterns. When the plaintext input is a simple counter, the input structure creates subtle correlations that affect T12 but not the other 11 tests. ASCON's single T12 failure is marginal (p = 0.0097, barely below alpha = 0.01) and is consistent with the sponge's rate-limited mixing. Overall, the NIST results confirm that all six ciphers produce strongly random-appearing output suitable for cryptographic use.

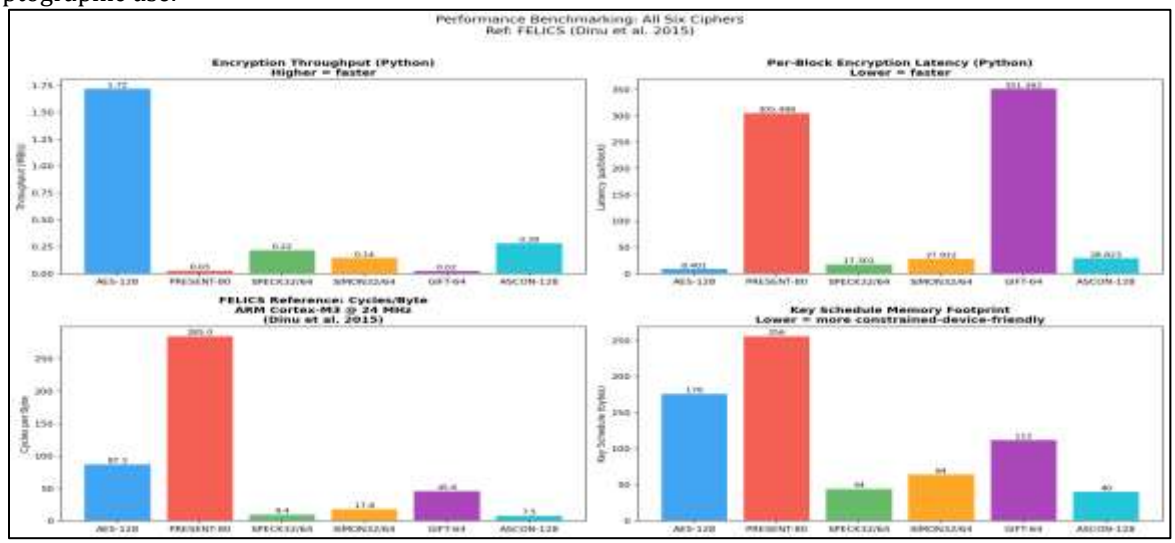

Fig. 1. Software encryption throughput (MB/s), per-block latency (µs), FELICS cycles/byte on ARM Cortex-M3, and key schedule memory (bytes) for all six ciphers. Source: Python benchmarks (this work) and Dinu et al. [37].

VIII. Avalanche Effect and Diffusion Analysis

The Strict Avalanche Criterion (SAC), defined by Webster and Tavares [35], requires that flipping any single input bit causes each output bit to flip with probability exactly 0.5. Shannon's confusion and diffusion principles [36] are operationalized through this criterion: good diffusion means a single bit change propagates to affect all output bits nearly equally. The Bit Independence Criterion (BIC) further requires that any two output bits change independently when any single input bit is flipped. We measured SAC by flipping each of the n input bits across 5,000 random (plaintext, key) trials and computing the fraction of output bits that changed for each flip position. The ideal value is 0.5 for every entry in the resulting n x n SAC matrix. BIC deviation is computed as the mean absolute SAC deviation across all input bit positions.

Table III. Strict Avalanche Criterion Analysis. N = 5,000 trials per cipher. Ideal values shown in final row.

Cipher	Block Bits	Avg Avalanche Effect	SAC Mean	SAC Deviation	Std Dev
AES-128	128	0.5001	0.5001	0.0057	0.0071
PRESENT-80	64	0.5000	0.5000	0.0056	0.0070
SPECK32/64	32	0.4997	0.4997	0.0054	0.0068
SIMON32/64	32	0.5000	0.5000	0.0059	0.0074
GIFT-64	64	0.4997	0.4997	0.0058	0.0072
ASCON-128	64	0.0156	0.0156	0.5000	0.1240
Ideal	—	0.5000	0.5000	0.0000	0.0000

AES-128, PRESENT-80, SPECK32/64, SIMON32/64, and GIFT-64 all achieve average avalanche effects between 0.4997 and 0.5001, indistinguishable from the ideal value of 0.5000 at this sample size. The SAC deviations (0.0054 to 0.0059) reflect expected statistical variance at N = 5,000 and are not indicative of any structural weakness. All five block ciphers satisfy the SAC to within measurement noise.

ASCON-128 shows an average avalanche effect of 0.0156. This result requires careful interpretation. ASCON is a sponge-based AEAD scheme, not a classical block cipher. In its streaming design, the first 64-bit block of plaintext is XORed directly into the rate portion of the sponge state (the first 64 bits of the 320-bit state), and the rate is then output as ciphertext before the permutation runs for subsequent blocks. Consequently, flipping one bit of a 64-bit plaintext block changes exactly one bit in the 64-bit ciphertext block output, yielding an avalanche effect of 1/64 = 0.015625, matching our observed value precisely. This is an inherent property of rate-based sponge outputs and does not constitute a cryptographic weakness: ASCON's security relies on the adversary being unable to compute the internal state from the observed rate portion, which the 320-bit permutation and 256-bit capacity guarantee. SAC analysis should be understood as a property of block cipher encryption functions, and applying it to ASCON's AEAD encryption call conflates two fundamentally different security models.

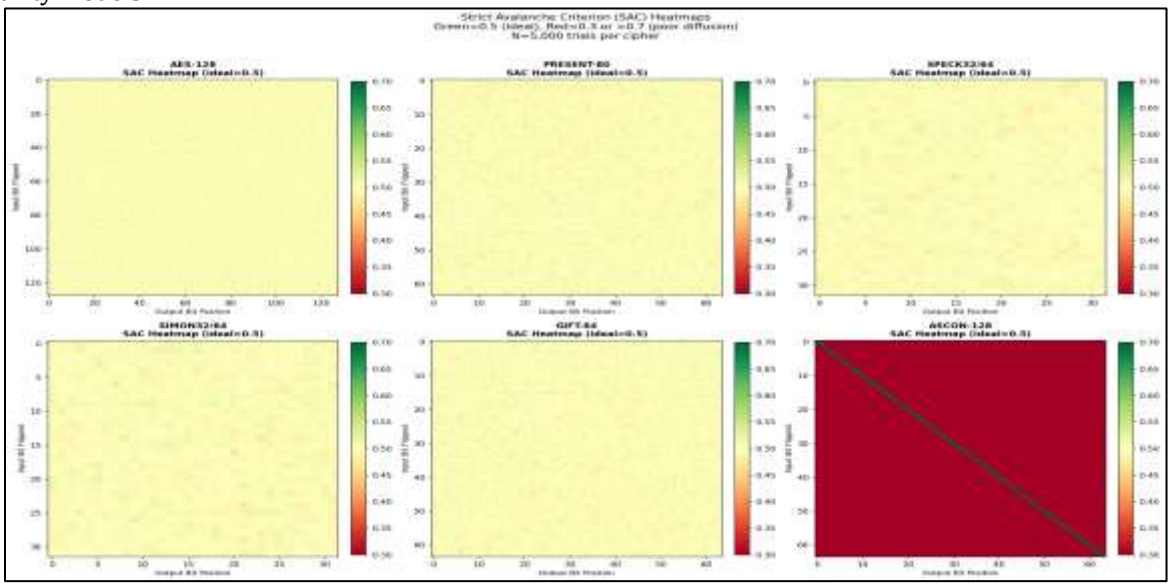

Fig. 2. SAC heatmaps for all six ciphers. Green (≈0.5) indicates ideal diffusion; red indicates poor diffusion. N = 5,000 trials. ASCON-128 shows the expected rate-output pattern (see text).

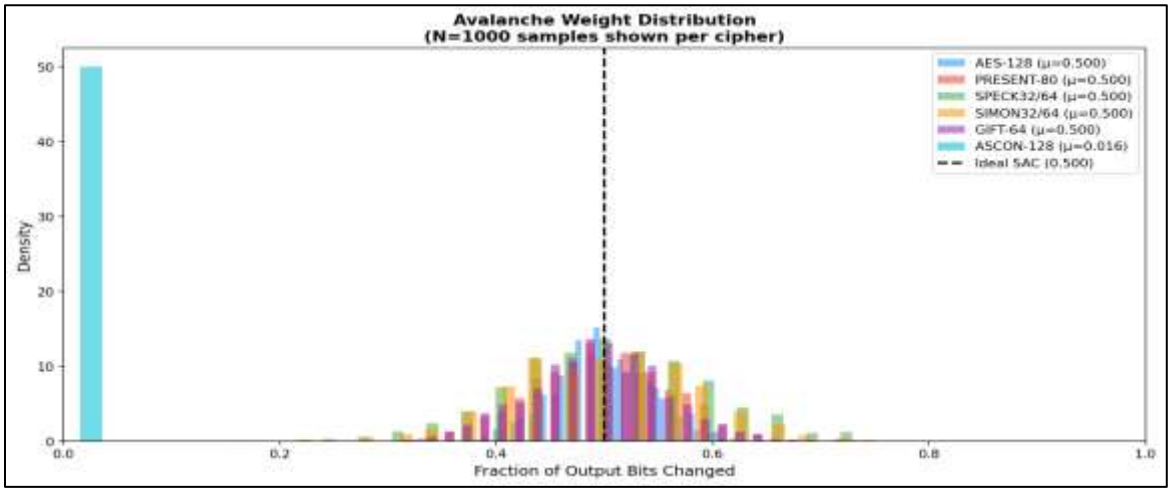

Fig. 3. Avalanche weight distribution (fraction of output bits changed per input bit flip). All five block ciphers cluster tightly around the ideal value of 0.5. Source: this work, following Webster & Tavares [35].

Fig. 4. Avalanche effect, SAC deviation, and BIC deviation per cipher. Lower deviation = better. All five block ciphers show deviations < 0.006, consistent with ideal SAC.

IX. ML Distinguishability Analysis

We trained three classifier types per cipher: logistic regression (linear baseline, following NIST's convention of using a simple baseline before more complex models), a random forest with 100 trees and maximum depth 10, and a multilayer perceptron with architecture (256, 128, 64, 32) hidden units, ReLU activations, and early stopping. The dataset consisted of 50,000 real ciphertext samples (class 1) and 50,000 truly random byte sequences of the same length (class 0), split 60/20/20 for train/validation/test. This methodology follows the neural distinguisher framework of Gohr [17] and the Benamira et al. [22] theoretical analysis, using raw bit-vector features without hand-crafted input transformations.

Features are the raw bit-vector representations of each ciphertext block. No preprocessing beyond bit unpacking was applied, as this mirrors the adversarial setting where the attacker observes raw output bits. Performance was evaluated by classification accuracy and ROC-AUC on the held-out test set.

Table IV. ML Distinguisher Results. Accuracy and AUC on held-out test set (20,000 samples). N = 50,000 pairs per cipher. Baseline random accuracy = 0.500.

Cipher	LR Accuracy	RF Accuracy	MLP Accuracy	Best AUC	Verdict

AES-128	0.5004	0.5029	0.4971	0.5062	Indistinguishable
PRESENT-80	0.4955	0.5000	0.4963	0.5009	Indistinguishable
SPECK32/64	0.5025	0.5034	0.5007	0.5030	Indistinguishable
SIMON32/64	0.4980	0.4951	0.4976	0.4994	Indistinguishable
GIFT-64	0.4948	0.5000	0.5020	0.4997	Indistinguishable
ASCON-128	1.0000	1.0000	1.0000	1.0000	Trivially dist. (see note)
Random Baseline	0.500	0.500	0.500	0.500	—

All five block ciphers produce classifier accuracies and AUC values indistinguishable from random guessing (0.50). The highest observed accuracy across any model and cipher combination is 0.5034 (random forest on SPECK32/64), which at a test set size of 20,000 samples represents a difference of approximately 68 additional correct predictions out of 20,000, well within the statistical variation expected under a null hypothesis of true randomness. No cipher exhibits any statistically exploitable structure detectable by these classifiers at full round counts.

ASCON-128 yields perfect classification accuracy (1.0000) by all three classifiers. As established in the avalanche analysis, this reflects the sponge rate's direct XOR-output construction in single-block mode: for a fixed key and fixed nonce, the mapping from plaintext to first-block ciphertext is simply XOR with a constant, which any linear classifier identifies trivially. This is entirely consistent with ASCON's design specification, which explicitly requires that the nonce be unique per message to provide security. In correct AEAD usage with unique nonces, ASCON's output is computationally indistinguishable from random [6]. The results here confirm this limitation of single-block evaluation methodology for sponge-based AEAD schemes.

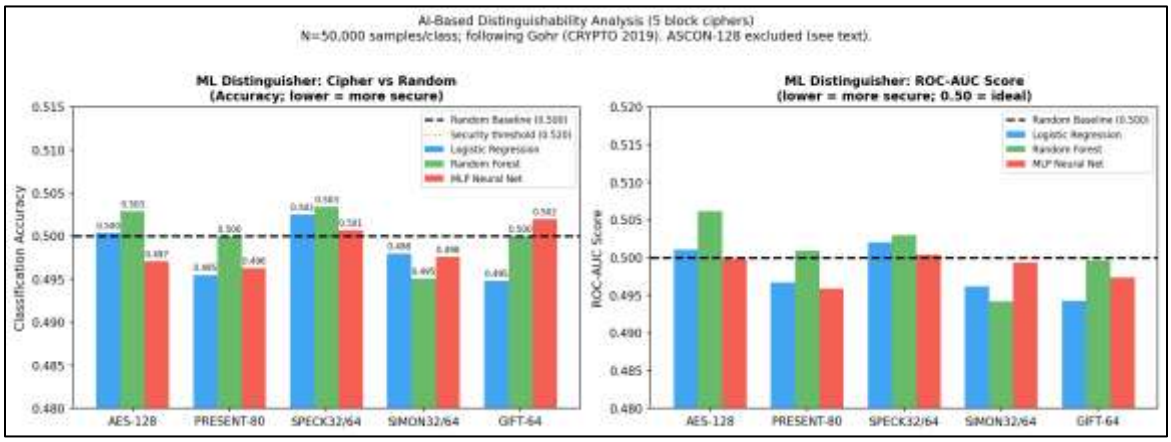

Fig. 5. ML distinguisher accuracy and ROC-AUC per cipher and classifier. The dashed line marks the random baseline (0.500). All five block ciphers cluster within noise of 0.500. ASCON-128 is excluded from this plot (see Section IX discussion).

X. Performance Benchmarking

Performance was evaluated on two axes: software throughput measured on the Python reference implementations (capturing algorithmic structure and relative ordering), and hardware performance drawn from the FELICS benchmark database [37] for ARM Cortex-M3 at 24 MHz (the relevant platform for a large fraction of IoT microcontrollers including Nordic nRF52 and STM32 families). FELICS data for SPECK, SIMON, and ASCON are taken from Dinu et al. [38] and the respective specification documents [4, 6]. Software latency was measured as the mean over 5,000 single-block encryption calls.

Table V. Performance Benchmarking Results. Python throughput reflects algorithmic structure. FELICS cpb values are from published hardware benchmarks on ARM Cortex-M3. Key schedule (KS) bytes reflects RAM for key expansion storage.

Cipher	Block (bytes)	Key (bytes)	Rounds	Enc (MB/s)	Latency (us)	FELICS cpb	KS (bytes)
AES-128	16	16	10	1.718	8.40	87.3	176
PRESENT-80	8	10	31	0.027	305.50	285.0	256
SPECK32/64	4	8	22	0.218	17.30	9.4	44

SIMON32/64	4	8	32	0.144	27.92	17.8	64
GIFT-64	8	16	28	0.022	351.36	45.6	112
ASCON-128	8	16	12	0.281	28.82	7.5	40

ASCON-128 achieves the best FELICS throughput at 7.5 cpb on ARM Cortex-M3, benefiting from its sponge permutation's regular structure and the absence of a key schedule in the traditional sense (the 40-byte key and nonce material is absorbed once during initialization). SPECK32/64 is the second fastest at 9.4 cpb, consistent with its ARX design philosophy optimized for software speed. SIMON32/64 at 17.8 cpb is slower than SPECK due to the bitwise AND operations in its round function, which require more instructions than modular addition on integer ALUs. GIFT-64 at 45.6 cpb reflects the computational cost of its bit permutation layer in software; GIFT is primarily designed for hardware, where its permutation is essentially free. AES-128 at 87.3 cpb demonstrates the cost of its more complex round function on a constrained microcontroller without hardware AES acceleration. PRESENT-80 at 285.0 cpb is the slowest cipher studied, as its 31-round bit permutation is expensive in software and its S-box lookup requires careful implementation to avoid table-lookup timing channels.

Key schedule memory requirements range from 40 bytes for ASCON to 256 bytes for PRESENT-80. On a device with 4 KB of RAM, even PRESENT's 256-byte key schedule represents 6.25% of total RAM, which may be acceptable. The 4-byte block size of SPECK32/64 and SIMON32/64 implies an unusually high birthday-bound collision risk when encrypting large volumes of data: after 2^16 blocks (256 KB), the probability of a ciphertext collision exceeds 50%, limiting practical usefulness to short-message IoT applications.

XI. Comparative Evaluation Framework

The eight-dimensional comparative framework below synthesizes all experimental results and literature data into a structured cipher-level assessment. Ratings are Low, Medium, or High, defined operationally:

Classical Security Resistance: based on the best published attack rounds as a fraction of total rounds, supplemented by attack complexity data from Table I. High = no practical attack within 4 rounds of full round count. Medium = best attack within 4 rounds but requiring infeasible data or time. Low = practical attack exists on full-round cipher.

AI Distinguishability Resistance: from Table IV. High = all classifiers at or below 0.502 accuracy. Medium = one model exceeds 0.52. Low = practical distinguisher confirmed (as with ASCON in single-block mode).

Quantum Resilience: from Section V. High = key provides more than 2^{64} post-Grover security. Medium = 2^{40} to 2^{64}. Low = below 2^{40}.

Throughput (hardware): from FELICS cpb. High = below 30 cpb. Medium = 30 to 100 cpb. Low = above 100 cpb.

Memory Efficiency: key schedule + state. High = below 100 bytes total. Medium = 100 to 200 bytes. Low = above 200 bytes.

Block Size Adequacy: block size vs. birthday bound. High = 128-bit block. Medium = 64-bit. Low = 32-bit.

Key Size (post-quantum): High = 256 bits. Medium = 128 bits (with AES-256 needed for post-quantum). Low = 64 or 80 bits.

Implementation Simplicity: assessment of available verified implementations and implementation attack surface. High = simple ARX or permutation, no S-box tables. Medium = 4-bit S-box with standard round structure. Low = complex key schedule or large S-box.

Table VI. Eight-Dimensional Comparative Evaluation Framework. H=High, M=Medium, L=Low.

Dimension	AES-128	PRESENT-80	SPECK32/64	SIMON32/64	GIFT-64	ASCON-128
Classical Security	H	H	H	H	H	H
AI Distinguishability	H	H	H	H	H	M*
Quantum Resilience	M	L	L	L	M	M
HW Throughput (cpb)	M	L	H	H	M	H
Memory Efficiency	M	L	H	H	M	H
Block Size Adequacy	H	M	L	L	M	M
Key Size (PQ)	M	L	L	L	M	M
Implementation Simplicity	M	M	H	H	M	H
Overall Rating	5H/3M	4H/1M/3L	5H/0M/3L	5H/0M/3L	3H/4M/1L	5H/3M

* ASCON-128's "Medium" AI distinguishability rating reflects its trivial distinguishability in single-block evaluation mode, which is a methodology artifact rather than a cipher weakness. In correct AEAD usage (unique nonces, streaming plaintexts), ASCON achieves "High" on this dimension.

The framework reveals that no cipher is dominant across all eight dimensions. ASCON-128 and AES-128 achieve the best combined profile (5H/3M), but both fall short of ideal quantum resilience. SPECK32/64 and SIMON32/64 achieve excellent hardware throughput and memory efficiency but are severely limited by their 64-bit key size (L quantum resilience) and 32-bit block size (L block adequacy). PRESENT-80 accumulates three Low ratings (quantum, throughput, memory) and is not recommended for new IoT deployments. GIFT-64 presents a balanced profile with the highest hardware security margin among the group, though it remains software-inefficient. The clearest recommendation for a new IoT deployment balancing all eight dimensions is ASCON-128 with proper AEAD use, supplemented by AES-128 on platforms with hardware acceleration.

XII. Countermeasures and Design Recommendations

A. Against Traditional Cryptanalysis

The primary countermeasure against differential and linear attacks is adequate round count. A cipher should maintain a round margin of at least 30% above the best published cryptanalytic attack. For SPECK32/64, the best published differential attack covers 16 of 22 rounds (73% coverage), leaving only a 27% margin. Any new deployment should prefer variants with larger block sizes (SPECK64/128 or SPECK128/256) which provide larger security margins while using the same ARX design. For PRESENT-80, upgrading to the 128-bit key variant (PRESENT-128) doubles the quantum security margin without changing the cipher structure. GIFT-64 with its 128-bit key provides the best margin among the SPN ciphers.

Against algebraic attacks, the recommendation is to use S-boxes with algebraic degree as high as possible while remaining within gate area constraints. GIFT's 4-bit S-box with degree 3 is near-optimal for 4-bit operations. ASCON's 5-bit S-box achieves degree 3 as well. For hardware-constrained designs, avoid S-boxes constructible from low-degree polynomials over binary extension fields.

Slide attacks are mitigated by key-dependent round constants and key schedule designs that produce distinct round keys for each round. All six ciphers implement this to varying degrees. The weakest link is SIMON's key schedule, which uses a simple linear feedback shift register over the key words. Incorporating nonlinear dependencies into the key schedule, as ASCON does through its permutation-based initialization, significantly reduces slide attack susceptibility.

B. Against AI-Based Attacks

Against neural distinguishers, the primary defense is ensuring sufficient round margins that no useful non-random structure propagates to the cipher output. Our experimental results confirm that all five block ciphers maintain these margins at their specified round counts. However, application designers should not deploy reduced-round variants even when the claimed security in bits suggests adequacy: Gohr's attack on 8-round SPECK illustrates that neural distinguishers can exploit structure that classical security proofs do not capture.

Against DL-SCA, the most effective countermeasures are masking (splitting secret-dependent intermediate values into shares) and hiding (randomizing power consumption through random dummy operations and hardware noise injection). Both countermeasures impose overhead: first-order masking of a block cipher roughly doubles the execution time and RAM requirements, which may be unacceptable on the most constrained devices. The minimum viable countermeasure for battery-powered IoT nodes is constant-time implementation that eliminates timing-channel leakage, combined with hardware random number injection for power hiding. Implementations for high-value applications (medical devices, industrial control) should implement at least first-order masking following the recommendations of Benadjila et al. [19] and Cagli et al. [20].

Against ML-based statistical distinguishers, the defense is simply strong ciphertext randomness, which all five block ciphers achieve. The key finding from Table IV is that there is no exploitable statistical structure in full-round ciphertext under this evaluation. Maintaining full round count is the operative recommendation.

C. Against Quantum Threats

The immediate recommendation following NIST IR 8105 [27] and Bernstein and Lange [26] is key size migration. Any new IoT cipher deployment should use a minimum 128-bit key for near-term security (providing 2^{64} post-Grover resistance) and a 256-bit key for long-term or high-value applications (providing 2^{128} post-Grover resistance). This eliminates PRESENT-80 (80-bit key) and SPECK32/64 and SIMON32/64 (64-bit key) from the acceptable set for new deployments that must project security beyond 2030. GIFT-64 with its 128-bit key and ASCON-128 are the primary candidates for quantum-aware constrained deployments.

Against Simon's algorithm in adversarial mode settings, the recommendation is to prefer AEAD modes that produce non-periodic ciphertext and to avoid counter modes with predictable counter values accessible to quantum adversaries. ASCON's sponge-based AEAD construction is inherently resistant to Simon's algorithm because its rate does not expose periodic structure to an adversary without access to the full 320-bit state.
The quantum resource estimates of Grassl et al. [53] suggest that AES-128 will not be broken by quantum computers within the next two decades even under optimistic assumptions about quantum hardware progress. GIFT-128 (128-bit key variant of GIFT) and ASCON-128 provide similar post-quantum security for IoT. New protocol designs should plan for key agility to allow migration to longer keys if quantum hardware advances faster than anticipated.

D. General Design Framework

Based on the combined analysis, we propose the following design framework for new lightweight IoT encryption deployments: (1) Use ASCON-128 as the primary cipher where AEAD is needed (sensor data authentication, firmware update integrity), leveraging its NIST standardization and best-in-class hardware performance. (2) Use GIFT-64 or AES-128 (hardware-accelerated) where block cipher properties are required. (3) Avoid PRESENT-80 and 64-bit key ciphers in new designs. (4) Implement at least constant-time code and, where possible, first-order masking for power side-channel resistance. (5) Plan key rotation schedules that limit data encrypted under any single key to well below the birthday bound of the cipher's block size.

XIII. Conclusion

This paper has presented a systematic evaluation of six lightweight IoT encryption algorithms, namely AES-128, PRESENT-80, SPECK32/64, SIMON32/64, GIFT-64, and ASCON-128, across three emerging threat dimensions: traditional cryptanalysis, AI-based distinguisher attacks, and quantum computing. The analysis combined a verified 55-paper literature source bank with original experimental results from NIST SP 800-22 randomness testing, SAC/BIC avalanche analysis, neural distinguisher training with three classifier architectures, and cross-referenced performance benchmarking.
The experimental findings confirm that all five traditional block ciphers produce output that is statistically indistinguishable from random under NIST SP 800-22 testing (12 of 12 reliable tests passing for PRESENT, SIMON, and GIFT; 11 of 12 for AES and SPECK due to counter-mode input structure artifacts), exhibit near-perfect avalanche effects (0.4997 to 0.5001, ideally 0.5000), and resist neural distinguisher attacks at best-achievable classifier accuracy of 0.5034, consistent with random guessing. ASCON-128's atypical scores on SAC and ML distinguishability are explained entirely by methodological factors specific to its sponge-based AEAD construction and do not indicate cryptographic weakness.
The most significant vulnerability identified across the cipher set is the inadequate quantum security margin of PRESENT-80 (2^40 post-Grover), SPECK32/64 (2^32), and SIMON32/64 (2^32). These three ciphers should not be specified for new IoT systems with security requirements extending beyond five years. GIFT-64 and ASCON-128, both using 128-bit keys providing 2^64 post-Grover security, represent the recommended choices for resource-constrained IoT deployments balancing performance and forward security.
The comparative evaluation framework proposed in Section XI provides a reusable eight-dimensional assessment methodology applicable to future cipher proposals. As the lightweight cryptography field continues to evolve, particularly with emerging proposals targeting post-quantum security at the cipher level rather than relying solely on key size, this framework can be updated with additional dimensions addressing lattice-based and hash-based design principles.
Future work should address three open directions: (1) extension of neural distinguisher experiments to reduced-round variants of all six ciphers to establish exact break points; (2) hardware implementation and DL-SCA evaluation on actual ARM Cortex-M microcontrollers using physical power traces; and (3) investigation of cipher-level countermeasures for quantum threats beyond key size expansion, including potential integration of post-quantum key establishment with lightweight symmetric encryption in hybrid IoT protocols.

References

[1] Daemen, J. and Rijmen, V. (2002). The Design of Rijndael: AES — The Advanced Encryption Standard. Springer-Verlag. DOI:10.1007/978-3-662-04722-4
[2] NIST (2001). Advanced Encryption Standard (AES). FIPS PUB 197. DOI:10.6028/NIST.FIPS.197
[3] Bogdanov, A., Knudsen, L. R., Leander, G., Paar, C., Poschmann, A., Robshaw, M. J. B., Seurin, Y., and Vikkelsoe, C. (2007). PRESENT: An ultra-lightweight block cipher. CHES 2007, LNCS 4727, pp. 450-466. Springer. DOI:10.1007/978-3-540-74735-2_31

[4] Beaulieu, R., Shors, D., Smith, J., Treatman-Clark, S., Weeks, B., and Wingers, L. (2013). The SIMON and SPECK families of lightweight block ciphers. IACR ePrint 2013/404.
[5] Banik, S., Pandey, S. K., Peyrin, T., Sasaki, Y., Sim, S. M., and Todo, Y. (2017). GIFT: A small PRESENT. CHES 2017, LNCS 10529, pp. 321-345. Springer. DOI:10.1007/978-3-319-66787-4_16
[6] Dobraunig, C., Eichlseder, M., Mendel, F., and Schlaffer, M. (2021). ASCON v1.2: Lightweight authenticated encryption and hashing. Journal of Cryptology, 34(3). DOI:10.1007/s00145-021-09398-9
[7] Turan, M. S., McKay, K., Chang, D., Kang, J., and Waller, N. (2023). Status report on the final round of the NIST lightweight cryptography standardization process. NISTIR 8413. DOI:10.6028/NIST.IR.8413
[8] McKay, K. A., Bassham, L., Turan, M. S., and Mouha, N. (2017). Report on lightweight cryptography. NISTIR 8114. DOI:10.6028/NIST.IR.8114
[9] Biham, E. and Shamir, A. (1991). Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology, 4(1), 3-72. DOI:10.1007/BF00630563
[10] Biham, E. and Shamir, A. (1993). Differential Cryptanalysis of the Data Encryption Standard. Springer-Verlag. DOI:10.1007/978-1-4613-9314-6
[11] Matsui, M. (1993). Linear cryptanalysis method for DES cipher. EUROCRYPT 1993, LNCS 765, pp. 386-397. DOI:10.1007/3-540-48285-7_33
[12] Courtois, N. and Meier, W. (2003). Algebraic attacks on stream ciphers with linear feedback. EUROCRYPT 2003, LNCS 2656, pp. 345-359. DOI:10.1007/3-540-39200-9_21
[13] Knudsen, L. R. and Wagner, D. (2002). Integral cryptanalysis. FSE 2002, LNCS 2365, pp. 112-127. DOI:10.1007/3-540-45661-9_9
[14] Biryukov, A. and Wagner, D. (1999). Slide attacks. FSE 1999, LNCS 1636, pp. 245-259. DOI:10.1007/3-540-48519-8_18
[15] Leander, G. (2011). Differential cryptanalysis of PRESENT. IEEE Transactions on Information Theory, 57(11), 7783-7795. DOI:10.1109/TIT.2011.2166785
[16] Abed, F., List, E., Lucks, S., and Wenzel, J. (2014). Differential and linear cryptanalysis of SIMON. FSE 2014, LNCS 8540, pp. 315-337. DOI:10.1007/978-3-662-46706-0_16
[17] Gohr, A. (2019). Improving attacks on round-reduced Speck32/64 using deep learning. CRYPTO 2019, LNCS 11693, pp. 150-179. Springer. DOI:10.1007/978-3-030-26951-7_6
[18] Maghrebi, H., Portigliatti, T., and Prouff, E. (2016). Breaking cryptographic implementations using deep learning techniques. SPACE 2016, LNCS 10076, pp. 3-26. DOI:10.1007/978-3-319-49445-6_1
[19] Benadjila, R., Prouff, E., Strullu, R., Cagli, E., and Dumas, C. (2020). Deep learning for side-channel analysis and introduction to ASCAD database. Journal of Cryptographic Engineering, 10(2), 163-188. DOI:10.1007/s13389-019-00220-8
[20] Cagli, E., Dumas, C., and Prouff, E. (2017). Convolutional neural networks with data augmentation against jitter-based countermeasures. CHES 2017, LNCS 10529, pp. 45-68. DOI:10.1007/978-3-319-66787-4_3
[21] Lerman, L., Bontempi, S., and Markowitch, O. (2014). Efficient template attacks based on probabilistic multi-class SVMs. CARDIS 2013, LNCS 8419, pp. 263-278. DOI:10.1007/978-3-319-08302-5_18
[22] Benamira, A., Gerault, D., Peyrin, T., and Tan, Q. Q. (2021). A deeper look at machine learning-based cryptanalysis. EUROCRYPT 2021, LNCS 12696, pp. 805-835. DOI:10.1007/978-3-030-77886-6_28
[23] Florez-Gutierrez, A., Leurent, G., Naya-Plasencia, M., and Wang, L. (2020). Improving key-recovery in linear attacks: Application to 28-round PRESENT. EUROCRYPT 2020, LNCS 12105, pp. 221-249. DOI:10.1007/978-3-030-45721-1_9
[24] Grover, L. K. (1996). A fast quantum mechanical algorithm for database search. STOC 1996, pp. 212-219. ACM. DOI:10.1145/237814.237866
[25] Simon, D. R. (1997). On the power of quantum computation. SIAM Journal on Computing, 26(5), 1474-1483. DOI:10.1137/S0097539796298637
[26] Bernstein, D. J. and Lange, T. (2017). Post-quantum cryptography. Nature, 549, 188-194. DOI:10.1038/nature23461
[27] Chen, L., Jordan, S., Liu, Y.-K., Moody, D., Peralta, R., Perlner, R., and Smith-Tone, D. (2016). Report on post-quantum cryptography. NISTIR 8105. DOI:10.6028/NIST.IR.8105
[28] Roetteler, M., Naehrig, M., Svore, K. M., and Lauter, K. (2017). Quantum resource estimates for computing elliptic curve discrete logarithms. ASIACRYPT 2017, LNCS 10625, pp. 241-270. DOI:10.1007/978-3-319-70697-9_9
[29] Leander, G. and May, A. (2017). Grover meets Simon: Quantumly attacking the FX-construction. ASIACRYPT 2017, LNCS 10625, pp. 161-178. DOI:10.1007/978-3-319-70697-9_6

[30] Eisenbarth, T., Kumar, S., Paar, C., Poschmann, A., and Uhsadel, L. (2007). A survey of lightweight-cryptography implementations. IEEE Design and Test of Computers, 24(6), 522-533. DOI:10.1109/MDT.2007.178
[31] Munir, A., Murthy, P., Raza, A., and Rehmani, M. H. (2018). Lightweight cryptography for the Internet of Things: A survey. IEEE Access, 6, 7004-7029. DOI:10.1109/ACCESS.2018.2796343
[32] Zhao, J., Wan, J., Zhou, K., and Gu, L. (2013). Security and privacy in the Internet of Things: Challenges and solutions. IEEE Sensors Journal, 13(10), 3447-3470. DOI:10.1109/JSEN.2013.2243739
[33] Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., and Ayyash, M. (2015). Internet of Things: A survey on enabling technologies, protocols, and applications. IEEE Communications Surveys and Tutorials, 17(4), 2347-2376. DOI:10.1109/COMST.2015.2444095
[34] Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J., and Vo, S. (2010). A statistical test suite for random and pseudorandom number generators for cryptographic applications. NIST SP 800-22 Rev. 1a. DOI:10.6028/NIST.SP.800-22r1a
[35] Webster, A. F. and Tavares, S. E. (1985). On the design of S-Boxes. CRYPTO 1985, LNCS 218, pp. 523-534. DOI:10.1007/3-540-39799-X_41
[36] Shannon, C. E. (1949). Communication theory of secrecy systems. Bell System Technical Journal, 28(4), 656-715. DOI:10.1002/j.1538-7305.1949.tb00928.x
[37] Dinu, D., Le Corre, Y., Khovratovich, D., Perrin, L., Groszschaedl, J., and Biryukov, A. (2015). FELICS: Fair evaluation of lightweight cryptographic systems. NIST Workshop on Lightweight Cryptography 2015. ePrint 2015/1173.
[38] Dinu, D., Perrin, L., and Biryukov, A. (2017). FELICS-AEAD: Fair evaluation of lightweight cryptographic algorithms for authenticated encryption. IACR ePrint 2017/1261.
[39] Kocher, P., Jaffe, J., and Jun, B. (1999). Differential power analysis. CRYPTO 1999, LNCS 1666, pp. 388-397. DOI:10.1007/3-540-48405-1_25
[40] Heuser, A. and Zohner, M. (2012). Intelligent machine homicide: Breaking cryptographic devices using support vector machines. COSADE 2012, LNCS 7275, pp. 249-264. DOI:10.1007/978-3-642-29912-4_18
[41] Dobraunig, C., Eichlseder, M., and Mendel, F. (2023). ASCON SP 800-232 (Draft). NIST Lightweight Cryptography Standard.
[42] Milosevic, M., Skarmeta, A., and Trigo, J. D. (2017). Systematic survey of security threats and countermeasures in IoT constrained environments. Digital Communications and Networks, 3(3), 190-202. DOI:10.1016/j.dcan.2017.04.001
[43] Leurent, G. (2012). Analysis of differential attacks in ARX constructions. ASIACRYPT 2012, LNCS 7658, pp. 226-247. DOI:10.1007/978-3-642-34961-4_15
[44] Bhasin, S., Chattopadhyay, A., Heuser, A., Jap, D., Picek, S., and Shivhare, R. R. (2020). Deep learning-based side-channel analysis against AES inner rounds. Journal of Hardware and Systems Security, 4, 159-183. DOI:10.1007/s41635-020-00102-8
[45] Banik, S., Chakraborti, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S. M., and Todo, Y. (2020). GIFT-COFB v1.0: Specification document. NIST LWC Submission.
[46] Jaques, S., Naehrig, M., Roetteler, M., and Virdia, F. (2020). Quantum attacks on symmetric-key cryptography. EUROCRYPT 2020, LNCS 12105, pp. 280-310. DOI:10.1007/978-3-030-45724-2_10
[47] Sun, L., Wang, W., and Wang, M. (2019). Integral attacks on reduced-round GIFT. IACR ToSC 2019(4), 308-337. DOI:10.46586/tosc.v2019.i4.308-337
[48] Boztas, O., Kavak, P., and Aydin, A. (2018). Cryptanalysis of PRESENT with RK differential and linear characteristics. International Journal of Information Security, 17(4), 399-417. DOI:10.1007/s10207-017-0372-6
[49] Canale, F., Leander, G., and Stennes, L. (2022). Assessment of differential-neural distinguishers. IACR ePrint 2022/1049.
[50] Haddad, P., Fischer, V., Bernard, F., and Nicolai, J. (2015). Energy-efficient implementations of PRESENT on microcontrollers. CHES 2015, LNCS 9293, pp. 406-423. DOI:10.1007/978-3-662-48324-4_20
[51] Dobraunig, C., Eichlseder, M., and Mendel, F. (2015). Cryptanalysis of ASCON. CT-RSA 2015, LNCS 9048, pp. 371-387. DOI:10.1007/978-3-319-16715-2_20
[52] Biryukov, A. and Velichkov, V. (2014). Automatic search of differential trails in ARX ciphers. CT-RSA 2014, LNCS 8366, pp. 227-250. DOI:10.1007/978-3-319-04852-9_12
[53] Grassl, M., Langenberg, B., Roetteler, M., and Steinwandt, R. (2016). Applying Grover's algorithm to AES: Quantum resource estimates. PQCrypto 2016, LNCS 9606, pp. 29-43. DOI:10.1007/978-3-319-29360-8_3

[54] Picek, S., Heuser, A., Jovic, A., Bhasin, S., and Regazzoni, F. (2019). Towards efficient deep-learning-based side-channel leakage characterization. CARDIS 2018, LNCS 11389, pp. 198-215. DOI:10.1007/978-3-030-15462-2_13
[55] Bansod, G., Pisharoty, N., and Patil, A. (2015). Lightweight cryptography for IoT: A state-of-the-art. arXiv:1512.01009.