

参赛学生姓名：田岳洲

中学：陕西省西安中学

省份：陕西省

国家/地区：中国

指导老师姓名：高洁、吴琼

指导老师单位：陕西省西安中学、
西安工业大学

论文题目：AM-SEV: A Metric Theory of Verifiable
Cryptographic Design, Its AI Engine, and the Tempest
Instance

AM-SEV: A Metric Theory of Verifiable Cryptographic Design, Its AI Engine, and the Tempest Instance

Tian Yuezhou

Abstract—Random number generators share a stubborn problem: nobody can measure how secure one is until long after it ships. Fast linear generators like xoshiro run at hundreds of gigabits per second but fall to linear algebra; provably secure designs such as ChaCha20 pay an order of magnitude in speed. Failures are real and recent — a badly seeded Mersenne Twister exposed roughly a billion dollars in vulnerable cryptocurrency (CVE-2023-39910). This paper asks whether security can be checked before any code exists, and answers with a framework of three parts: AM-SEV (A Metric Security-Efficiency Vector theory), an AI engine built on top of it, and a concrete instance called Tempest v3.

The mathematical core is the Polar-Rank Theorem: for two-layer (quadratic) AND-RX round functions, the differential map is affine, so $\text{DP}(\Delta) = 2^{-\text{rank}(B_\Delta)}$ exactly at any word width. Its content is classical — the affine-differential structure of quadratic Boolean maps (Nyberg; Carlet) — and AM-SEV turns it into a design-time verification instrument. New results include the two-layer polar-rank barrier (minimum rank exactly 3 at every width), the cascade elimination law ($2^{-3} \rightarrow 2^{-11}$ exact at $W = 4$), the Walsh-rank linear dual (bias 2^{-128} exactly for the full-state mask at $W = 64$; $\leq 2^{-54}$ over sampled 64-bit masks), the Cascade Degree Law, and a differential decomposition bound.

Every security statement is adjudicated by the same calibrated verifier stack — any proposal, whether from a large language model (LLM), from evolutionary search, or from a human, enters only through it — and the stack reproduces a genuine weakness from this project’s own history (the dead-key v3.1) in milliseconds, where terabytes of output could not localize it.

The instance is Tempest v3. It passes the full stack (1 TiB PractRand, TestU01 BigCrush, NIST SP 800-22), maps to field-programmable gate arrays (iCE40 k=2 bitstream; full-cascade bitstreams on an ECP5-85K, the fastest being a 6-stage-retimed full-width implementation at 2.39 Gbit/s), integrates into eight ecosystems, all Known-Answer-Test (KAT) verified bit-exact, and tags every security statement by its basis — theorem, exact computation, calibrated measurement, or engineering estimate — with every number reproducible from the open-source repository.

Index Terms—Cryptographic design theory; verifiable design; AND-RX; exact metrics; AI-assisted design.

I. INTRODUCTION

Cryptographic random number generators fall into two camps. At one end, fast linear generators like xoshiro and PCG push hundreds of gigabits per second, but their state is $\mathbb{F}_2$ -linear and a handful of outputs is enough to recover it [6]; at the other, provably secure constructions such as ChaCha20 and AES-256-CTR deliver 2^{256} security at roughly an order of magnitude lower throughput (a factor of about 16 on this machine, Table XI). What both camps share is the real bottleneck: *you cannot measure security before the system is*

built. A designer who ships a new primitive has no instrument that says whether it is safe. The community analysis that eventually does is measured in years. And a construction can sail through every statistical test while carrying a structural flaw that surfaces only later.

The gap is not academic. In 2025, AMD’s Zen 5 RD-SEED returned zero-valued output far more often than true randomness while still setting its success flag (CVE-2025-62626), weakening TLS and SSH key generation on millions of processors; the Milk Sad class of wallet-generation flaws (a Mersenne Twister seeded with 32 bits of entropy) exposed roughly \$1B in vulnerable cryptocurrency (CVE-2023-39910); the pattern repeats across network-facing devices, medical devices, and industrial controllers (CVE-2024-47945, CVE-2020-27264, CVE-2025-55069). In every case the root cause is identical: the designer had no way to measure the construction’s randomness quality before shipping. Statistical test suites cannot localize this class of flaw: they look at output statistics, not construction structure, and a broken construction can pass them easily. The AI era makes the stakes higher: LLMs sample from deterministic PRNGs, and SeedHijack [38] backdoors the sampling PRNG to force attacker-chosen tokens (99.6% injection, past every tested alignment method). PRNG integrity is now a first-class security requirement for AI systems, and no instrument certifies a generator before deployment.

Relation to existing design strategies. Three established approaches frame this work. The *wide trail strategy* [15] bounds differential and linear probability by counting active S-boxes and lower-bounding their number via branch number; AM-SEV replaces the counting by an exact rank computation (Section III-B3). *Decorrelation theory* [16] achieves ideal round probabilities *by construction*, an approach recently revived in the MSX block-cipher family [3]; AM-SEV is complementary — it measures exactly, at any width, designs that do not enforce ideal probabilities by construction, and it does so before any code exists. *Automatic trail analysis* via MILP/SAT [1], [2] bounds multi-round trails for ARX and bit-oriented designs; we adopt the same technique for the cascade’s full-width trail bounds (Section VI). Statistical suites (PractRand [40], TestU01 [41], NIST SP 800-22 [39]) remain necessary but structure-blind: they certify output statistics, not construction structure, which is precisely the blind spot Section I describes.

This paper asks a different question: *can security be mea-*

measured at design time, before any code is written? Our answer is AM-SEV, a metric theory whose security statements are reproducible computations, an AI engine that designs and adjudicates candidates with those metrics, and an instance that clears every calibrated gate. The main contributions are as follows:

- 1) **A metric theory of verifiable design.** For two-layer AND-RX round functions, $DP(\Delta) = 2^{-\text{rank}(B_\Delta)}$ exactly at any width (the Polar-Rank Theorem, Section III-A1), with new consequences: the two-layer polar-rank barrier (minimum rank exactly 3, so a cascade is a necessity), the cascade elimination law and the exact multi-round curve (Sections III-C1, III-C2), the Walsh-rank duality, the Cascade Degree Law, the Differential Decomposition Bound, and a closed rotation-symmetry analysis — an RX-differential rank theorem making the rotational differential probability of the quadratic class equal to the rotated first-order probability at any width, with the constants provably irrelevant (Section III-B4). Each theorem and measured law ships a machine-checkable certificate (Sections III-A2–III-C3).
- 2) **Tempest v3, the instance.** An AND-RX candidate whose full measurement record ships with the code: Tempest v3 passes every calibrated gate — 1 TiB PractRand, TestU01 BigCrush, NIST SP 800-22, a KAT-verified RTL mapping to FPGAs (iCE40 k=2 bitstream; full-cascade bitstreams on an ECP5-85K, the fastest being a 6-stage-retimed full-width implementation at 2.39 Gbit/s — $8.5\times$ the half-width serial version on the same device, at an area efficiency on par with compact AES cores), and 22.1 Gbit/s dual-output scalar throughput (normalized to 5.0 GHz; §VIII). All verifier code, verifier records, certificates, and hardware designs are open-source at <https://github.com/paim-creator/prng>, and the audit record documents the correction of this project’s own earlier measurements (Sections V–IX).
- 3) **An AI design engine (supporting asset).** Verification, not the generator, is the invariant: every proposal, LLM-authored, evolutionary, or human, is adjudicated by the same calibrated verifier stack; the engine reproduces a real weakness exactly (the dead-key v3.1, caught in milliseconds where terabytes of output could not localize it) and compiles a security specification into a Pareto family of candidates, each shipping a certificate (Section IV).

Why this matters. Why does fast, secure randomness matter? Three reasons stand out. *Internet of Things (IoT) security:* tens of billions of devices are expected online by 2030 [31], many on chips with no cryptographic hardware at all. *Privacy-preserving machine learning:* differentially private training [32] burns billions of random bits per run, and those bits have to be good. *Open-source hardware security:* Dual-EC-DRBG [14] showed that proprietary cryptographic hardware can hide deliberate weaknesses, and the only systemic answer is a design — algorithm, engine, certificates, RTL — that

anyone can inspect and verify.

The rest of the paper is organized as follows. Section II sets up background and the threat model; Section III presents the AM-SEV framework; Section IV describes the AI engine; Section V introduces the instance Tempest v3; Section VI gives the security analysis; Section VII reports the hardware implementation; Section VIII presents experimental verification; and Section IX closes with applications, limitations, and the audit record. Figure 1 maps the whole narrative in one picture: the problem (traditional designs whose statistical tests cannot localize security flaws), the framework’s three exact metrics, the structural barrier they prove (rank 3, $DP = 1/8$, at any width), the cascade as the forced escape ($DP \rightarrow 2^{-11}$ exact at $W=4$ and $< 2^{-24}$ sampled at $W=64$), and the machine-checkable verification (bitstream + certificates) that closes the loop.

II. BACKGROUND AND THREAT MODEL

A. PRNG Framework and Security Games

Following Blackman and Vigna [6], a pseudorandom number generator is a triple (S, Φ, φ) with state space S , transition Φ , and output function φ ; the t -th output is $x_t = \varphi(\Phi^t(s_0))$. For Tempest v3 the state space is $|S| = 2^{256}$ (four 64-bit state words); the master key is likewise 256 bits $(k_0, \dots, k_3)$, entering the state through the initialization injections. The internal 64-bit round-key word w_ℓ evolves from a *public* constant (Section V, Phase B) and carries no key entropy — the $\mathbb{F}_2$ -affine schedule converges to a key-independent constant within 64 steps, a disclosed design fact.

B. Threat Model

We adopt Kerckhoffs’ principle: the adversary knows the full algorithm. Two standard security games frame the analysis — *key-recovery* (the adversary, given output blocks, recovers the key) and *statistical distinguishing* (the adversary tells the generator’s output from uniform); their formal definitions are given where they are used, at the start of Section VI. *Key-recovery security* is the primary claim: the target strength is 2^{128} against exhaustive search over a 2^{256} key space [10]. The distinguishing advantage is a separate metric bounded by the information-theoretic birthday limit $\binom{m}{2}2^{-128}$ — ChaCha20 and AES-CTR face the same bound. Chosen-IV and related-key security are outside the deterministic-random-bit-generator (DRBG) threat model and are not analyzed here. The security analysis spans four dimensions: algebraic degree (via division-property MILP, mixed-integer linear programming, the standard formulation that automates the search for best differential/division trails), differential (a_1 and the polar-rank metrics), linear (mask counting and measurement), and bit-level trail search (via Boolean satisfiability (SAT) — the constraint-satisfaction problem of deciding whether a Boolean formula has a satisfying assignment; modern solvers handle instances with millions of clauses).

Standards positioning. Tempest v3 is packaged as a NIST SP 800-90A DRBG with an SP 800-90B [33] entropy source, i.e. a random-bit-generator-2 (RBG-2) style construction in the

TEMPEST v3

An AND-RX Cryptographic Pseudorandom Number Generator with 256-bit State
Designed by the AI-assisted Verifiable Engineering Framework AM-SEV

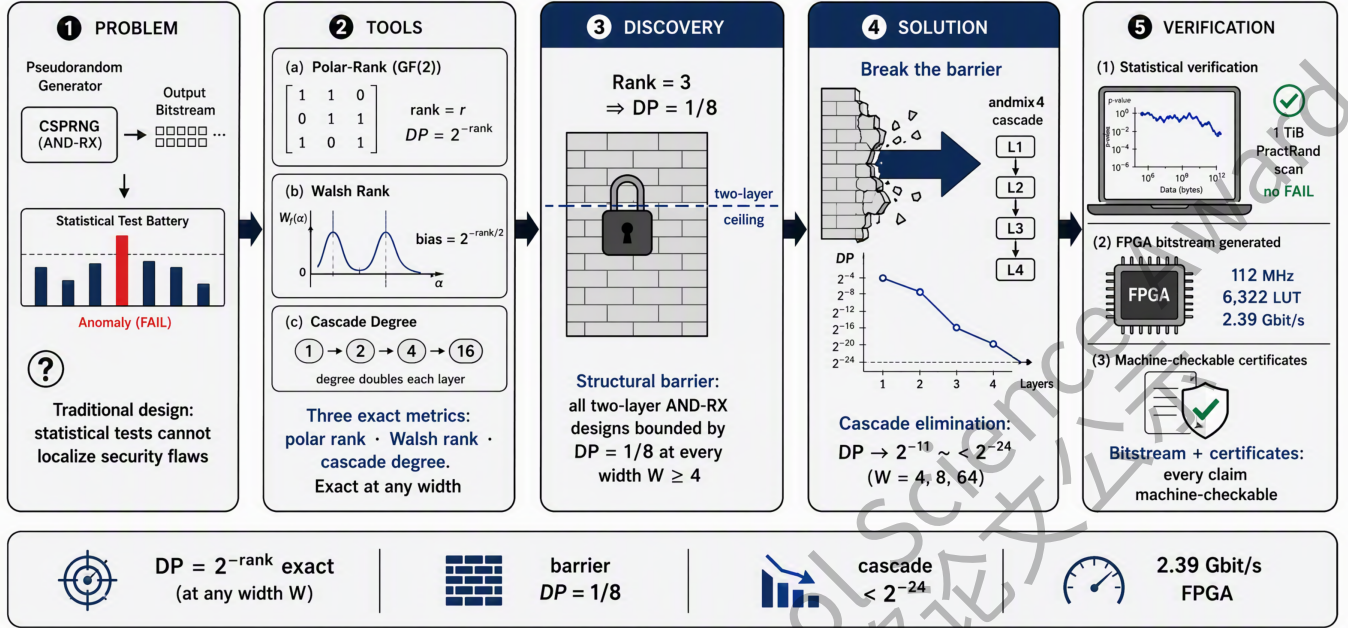


Fig. 1. The AM-SEV narrative at a glance.

terminology of the finalized SP 800-90C (September 2025) [11]; the 90C requirement to source *security-strength/2 bits of real randomness* at instantiation is satisfied by seeding from the SP 800-90B entropy source rather than a static nonce.

C. Related Work

The following comparison situates AM-SEV among existing approaches. **Fast linear PRNGs.** xoshiro/xoroshiro [6], Mersenne Twister [8], and PCG [7] reach 60–80 Gbit/s, but their $\mathbb{F}_2$ -linear states are recoverable by linear algebra from a few outputs. Their statistical quality is excellent — this is precisely the trap: statistical tests cannot localize the class of structural flaws this paper’s metrics detect.

Provably secure constructions. ChaCha20 [9] and AES-CTR-DRBG [10] offer 2^{256} security at lower throughput; Blum–Blum–Shub (BBS) [12] and the RSA PRG [13] run at ~ 1 Mbit/s; Dual-EC-DRBG [14] shows even provable constructions can hide kleptographic backdoors — a motivation for verifiable, open design.

GF(2) constructions and the wide trail strategy. AND-RX designs (SAND [17], Ascon [18]) show the {AND, XOR, ROT} vocabulary suffices for secure primitives, but their design process remains heuristic. The wide trail strategy [15] bounds differential/linear probability by counting active S-boxes at word level; this paper instantiates its active-component counting at bit level, precisely, and at *any width*

for the two-layer class — a regime in which every published toolchain is an estimate.

Automated and AI-assisted cryptanalysis. Gohr [21] showed neural networks find differential distinguishers for round-reduced ciphers; this paper uses the same instrument as a calibrated Layer-B verifier. Trail-search toolchains (CLAASP [22]) auto-generate SAT/MILP models; our exact layer goes further in two regimes — full difference-space enumeration at reduced width, and GF(2) rank computations at full width. Xuazhi (CryptoLLM) [23] is a cryptography-domain LLM with an open API; the present work differs in position: where an LLM proposes text, this engine *measures*.

Quadratic Boolean maps and FPGA PRNGs. The affine-differential structure of quadratic maps is classical (Nyberg [5]; Carlet [4]); this paper’s contribution is its systematic use as a design-time verification instrument. On the hardware side, CryptoliteICE [37] builds a USB TRNG on an iCE40UP5K; our work differs in kind: it implements a cryptographically *verified* PRNG rather than a TRNG — the two are complementary (a TRNG provides the SP 800-90B entropy source; the verified DRBG provides the expansion).

III. THE AM-SEV FRAMEWORK

The security-measurement core of this work is a self-contained mathematical framework over $\mathbb{F}_2$: exactly computable security metrics, theorems about them, and axioms encoding the verification-first principles of the engine (Sec-

tion IV). Every numerical claim can be reproduced from the open-source repository.

Correctness scope. All exact reduced-width numbers are computed with the *snapshot semantics* of the published algorithm (Phase A reads the round-start snapshot (u_0, v_0, w_0, z_0) ; this is the semantics of the KAT-verified C implementation and of the vectorized domain-specific-language (DSL) interpreter, which agree bit-for-bit). An earlier interpreter read current registers instead of the snapshot; it describes a different round function, and the numbers it produced ($H_\infty = 7.93$, $\tau = 1.000$, the cascade table 5.00/8.00/8.00/7.80/7.80) are *withdrawn* and replaced by the values below. The correction is part of the audit record (Section IX).

Scope of every claim. Every security statement of this paper is tagged by one of four bases, and the tag travels with the claim: (i) *theorem*: exact at any width, for the two-layer class (the Polar-Rank Theorem, the barrier, the Walsh-rank duality, the degree law); (ii) *exact computation*: full-domain enumeration at $W = 4$, snapshot semantics; (iii) *calibrated measurement*: sampled at $W = 8/64$ with the stated resolution ($2^{16}-2^{24}$); (iv) *engineering judgment*: width extrapolation of measured scaling, identified as judgment wherever it appears. Claims of kinds (ii)–(iv) are not presented as theorems, and width-scaling extrapolations (kind iv) are the statements most likely to be revised by future measurement or analysis; they are the only claims the paper does not defend with a certificate.

A. Exact Metrics for the Three Attack Dimensions

The framework scores a round function along the three classical attack dimensions — differential, linear, and algebraic — and this module establishes the *exact metric* for each: computable at any width in polynomial time, the differential side by the polar rank, the linear side by the Walsh rank, and the algebraic side by the cascade degree law. These three metrics are the mathematical foundation of the barrier analysis of the two-layer class (Section III-B1) and of the cascade design that the barrier forces (Section III-C1).

III-A1 The Polar-Rank Theorem.

Definition III.1 (Design Space and Two-Layer Designs). A *design* is a round-function program over $\mathbb{F}_2$ built from {AND, XOR, ROT} with word width W and four state words $(u, v, w, z) \in (\mathbb{F}_2^W)^4$. A design is *two-layer* (its round map *quadratic*) if every AND gate reads words that are affine functions of the round input. For the cascade design of Section V, the *quadratic shadow* Sh is obtained by removing all cascade levels and the intra-word pre-mix, keeping Phase A, the snapshot ANDs, the key phases, and the final linear mixing.

At $W = 4$ the intra-word pre-mix collapses to a linear map ($22 \equiv 26$ and $7 \equiv 19 \pmod{4}$), which is why the shadow is exactly quadratic at toy width; at all $W \geq 8$ its AND is genuinely nonlinear. All exact any-width statements are made for the pre-mix-free shadow.

Theorem III.1 (Affine Differential Theorem). If the round map $\Phi : \mathbb{F}_2^n \rightarrow \mathbb{F}_2^n$ ($n = 4W$) has algebraic degree ≤ 2 , then

for every $\Delta \neq 0$ the differential map $D_\Delta(s) = \Phi(s \oplus \Delta) \oplus \Phi(s)$ is affine: $D_\Delta(s) = B_\Delta s \oplus D_\Delta(0)$, where B_Δ is the *polar form* (columns $B_\Delta e_j = D_\Delta(e_j) \oplus D_\Delta(0)$). Consequently

$$\text{DP}(\Delta) = 2^{-\text{rank}(B_\Delta)}, \quad H_\infty(\Phi) = \min_{\Delta \neq 0} \text{rank}(B_\Delta), \quad (1)$$

with $\text{DP}(\Delta) = \max_c \Pr_s[D_\Delta(s) = c]$. Both quantities are computed exactly in polynomial time at any width (GF(2) rank, $O(n^3)$).

Proof. Each monomial of a degree- ≤ 2 polynomial is linear or a product of two variables; for a monomial $s_i s_j$, $(s_i \oplus \Delta_i)(s_j \oplus \Delta_j) \oplus s_i s_j = \Delta_i s_j \oplus \Delta_j s_i \oplus \Delta_i \Delta_j$, affine in s . Hence D_Δ is affine, its fibres are cosets of $\ker(B_\Delta)$, and every image value is hit exactly $2^{n-\text{rank}(B_\Delta)}$ times. $\square$

The stack machine-checks the theorem exhaustively at $W = 4$ (all $2^{16} - 1$ differences), on $> 10^5$ random quadratic-map tests (zero exceptions), and via derivative tests at $W = 8$ (audit_cascade_rank.py, test_aggregation_conjecture.py).

Corollary III.1 (Second-Order Degeneration). For a two-layer round map Φ the second-order differential

$$D_{\Delta_1, \Delta_2} \Phi(s) = \Phi(s) \oplus \Phi(s \oplus \Delta_1) \oplus \Phi(s \oplus \Delta_2) \oplus \Phi(s \oplus \Delta_1 \oplus \Delta_2) \quad (2)$$

is *independent of s* and equals the polar form $B(\Delta_1, \Delta_2)$ of Theorem III.1. Consequently the second-order zero differential spectrum is decided by the first order polar rank: $D_{\Delta_1, \Delta_2} \Phi \equiv 0$ if and only if $B(\Delta_1, \Delta_2) = 0$, so the boomerang-style second-order analysis of the whole quadratic class carries no information beyond the first-order polar form.

Proof. A linear term x_i contributes $x_i \oplus (x_i \oplus a) \oplus (x_i \oplus c) \oplus (x_i \oplus a \oplus c) = 0$ with $a = (\Delta_1)_i$, $c = (\Delta_2)_i$; a constant cancels likewise. A quadratic monomial $x_i x_j$ expands to $a d \oplus b c = (\Delta_1)_i (\Delta_2)_j \oplus (\Delta_1)_j (\Delta_2)_i$, the (i, j) entry of the polar form $B(\Delta_1, \Delta_2)$; every vector-valued output bit behaves the same. $\square$

The stack checks the corollary at $W = 4$: for 200 random pairs (Δ_1, Δ_2) with six sample points each, the second-order differential is constant in all cases (rx_theory_check.py).

III-A2 The Walsh-Rank Duality.

The differential side has a linear counterpart that closes the framework's second classical gap — exact linear analysis at full width — for the same two-layer class. The structure is the classical Dickson normal form of quadratic Boolean functions [4], [5]; its use as a design-time instrument is new here.

Theorem III.2 (Linear Polar-Rank Theorem (Walsh-rank duality)). Let Φ be a two-layer round map and $g_u(s) = u \cdot \Phi(s)$ the output-mask function for mask u . Write Q_u for the symmetric matrix of the quadratic part of g_u . Then: (i) the *unnormalized* Walsh spectrum $\hat{g}_u(a) = \sum_x (-1)^{u \cdot \Phi(x) \oplus a \cdot x}$ satisfies $\hat{g}_u(a) \in \{0, \pm 2^{n-\text{rank}(Q_u)/2}\}$ for all a (Dickson structure), so

the linear bias of the mask is $\text{bias}(u) = 2^{-\text{rank}(Q_u)/2}$ exactly; and (ii) the maximum linear bias over all masks is

$$\max_u \text{bias}(u) = 2^{-r_{\min}/2}, \quad r_{\min} = \min_u \text{rank}(Q_u), \quad (3)$$

computable in polynomial time at any width (one GF(2) rank per mask).

Proof. Every g_u is quadratic (a linear combination of quadratic output bits). By Dickson's theorem, a quadratic function is affinely equivalent to the standard form $\sum_{k=1}^{r/2} x_{2k-1}x_{2k}$ with $r = \text{rank}(Q_u)$; each quadratic pair contributes a factor ± 2 and the remaining $n-r$ variables contribute 2^{n-r} , so $\hat{g}_u(a) \in \{0, \pm 2^{n-r/2}\}$ and the maximum bias is $2^{-r/2}$. The claim (ii) follows by minimizing over u . $\square$

The stack checks the theorem exhaustively at $W = 4$ (over 400 masks, zero failures). By (ii), the global maximum is set by the *minimum* rank: single-bit output masks attain $r_{\min} = 2$ (the v - and w -word gates read one disjoint variable pair each, and the u - and z -word gates two), so $\max_u \text{bias}(u) = 2^{-1}$ — the weakest linear mask is an individual output bit, which is necessarily correlated with its AND inputs; this is a property of every AND-RX output, not a design weakness, and the mask granularity is what matters: at eight bits the design is already strong, $u = 0x00ff$ attaining rank 16 and bias 2^{-8} (the strongest sampled mask). At full width the same computation gives exact per-mask bounds for the quadratic shadow at $W = 64$ in minutes, not years of community analysis: the full-state mask attains rank $4W = 256$, i.e. $\text{bias}(u) = 2^{-128}$ exactly; over 1,000 sampled masks with 64 active bits the minimum rank is 108 (bias $\leq 2^{-54}$) and with 32 active bits 58 (bias $\leq 2^{-29}$); the word mask $0x00ff$ gives rank 20 (bias 2^{-10}). As at $W = 4$, the global maximum remains the single-bit mask ($r_{\min} = 2$, bias 2^{-1}). For the full cascade design, the linear statement remains measurement-based (maximum bias $\approx 2^{-6.1}$ at $W = 8$, $R = 2$, null-consistent under multiple comparisons; Section VIII); an exact cascade-side linear theory is future work. The triple metric is summarized in Table I: the differential side is governed by the *minimum* polar rank over differences, the linear side by the *minimum* quadratic rank over masks — both exact at any width.

TABLE I
THE EXACT TRIPLE METRIC FOR AND-RX ROUNDS: DIFFERENTIAL (MINIMUM POLAR RANK), LINEAR (MINIMUM WALSH RANK), AND ALGEBRAIC (CASCADE LEVEL COUNT).

	Differential	Linear	Algebraic
Object	$D_\Delta = \Phi(\cdot \oplus \Delta) \oplus \Phi$	$g_u = u \cdot \Phi$	$\deg(\Phi_k)$
Structure	affine (B_Δ)	Dickson (Q_u)	ANDs double degree
Exact	$\text{DP} = 2^{-\text{rank}(B_\Delta)}$	$\text{bias} = 2^{-\text{rank}/2}$	$\deg \leq 2^k \deg_0$
Global	$\min_\Delta \text{rank}$	$\min_u \text{rank}$	saturation $4W$
	$r_{\min}=2$	$r_{\min}=2$	
$W=4$	rank 3; DP= 2^{-3}	2^{-8} @ $0x00ff$ $r_{\min}=2$	2, 4, 8, 14, 16
		256 (2^{-128})	
$W=64$	rank 3 (cert.)	≥ 108 (2^{-54})	[16, 128]; $\leq 4W@2R$

III-A3 The Cascade Degree Law. A structural companion theorem bounds the differential map's degree and explains its nonlinearity in cascades:

Theorem III.3 (Differential-Map Degree). Let Φ have algebraic degree $d = \deg(\Phi) \geq 1$.

- For every $\Delta \neq 0$ with $D_\Delta \neq 0$, $\deg(D_\Delta) \leq d-1$: the leading monomials of $\Phi(s \oplus \Delta)$ and $\Phi(s)$ cancel in the XOR.
- There exists $\Delta \neq 0$ with $\deg(D_\Delta) = d-1$. Take any degree- d monomial $m = x_{i_1} \cdots x_{i_d}$ of Φ and $\Delta = e_{i_1}$: the contribution of m to D_Δ is $\prod_{j=2}^d x_{i_j}$, and no other monomial contributes that term (producing $\prod_{j=2}^d x_{i_j}$ requires a monomial containing x_{i_1} and all of $x_{i_2}, \dots, x_{i_d}$, i.e. m itself), so the degree is exactly $d-1$.

The upper bound is not attained at *every* Δ — for $\Phi = x_1x_2$ and $\Delta = e_3$, $D_\Delta \equiv 0$ — so the original formulation of this theorem as a per- Δ lower bound was incorrect and is corrected here. For Algorithm 1 the bound is attained at the tested difference $\Delta = 0x1$: $\deg(\Phi) = 16$ and $\deg(D_\Delta) = 15$ (exact algebraic normal form, ANF, at $W = 4$; `explore_cascade_degree.py`).

The design consequence is measured, not proved: the differential map of a cascade is genuinely nonlinear (degree 15), and this nonlinearity *aggregates* states onto fewer output differences (the Aggregation Property, Section III-C2) rather than dispersing them — the nonlinear cascade terms concentrate $\approx 2^9$ -fold more states onto the worst output difference than the linear part alone.

The degree theorem has a companion law that quantifies how cascades build algebraic degree — the framework's third exact metric.

Theorem III.4 (Cascade Degree Law). Let Φ_k be the k -level cascade truncation of the round map ($k = 0$: the quadratic shadow with pre-mix; each further level appends one cross-word AND layer). Then

$$\deg(\Phi_k) \leq 2^k \cdot \deg(\Phi_0), \quad (4)$$

because each AND layer at most doubles the degree of every output word while XOR and rotation never increase it. For Algorithm 1 at $W = 4$, full-domain enumeration gives the exact ANF degrees

$$\deg(\Phi_k) = 2, 4, 8, 14, 16 \quad (k = 0, \dots, 4), \quad (5)$$

and the bound is attained at $k = 1, 2$; at $k = 3$ two degree-8 monomials cancel (14, not 16 — the first measured cancellation); at $k = 4$ the $4W = 16$ -variable ceiling is reached: *the round is degree-saturated within one round*.

Proof. The bound is the composition law $\deg(\Phi_1 \circ \Phi_2) \leq \deg(\Phi_1) \deg(\Phi_2)$ applied to the AND layers (each layer composes a degree-2 map with the previous words). The exact values are the full-domain ANF of each truncation (Möbius transform over all 2^{16} states, one per output bit; `explore_cascade_degree.py`). $\square$

Two consequences form the *algebraic barrier*. First, *cross-round invariance*: at $W = 4$, $\deg(\Phi^R) = 16$ and $\deg(D_\Delta^{(R)}) = 15$ for $R = 1..6$ (full-domain ANF; invariance beyond $R = 6$ is not proven); the aggregation mechanism is not diluted

by iterating. Second, *the barrier*: a function of degree d has vanishing $(d+1)$ -th derivative; at $W = 4$ the degree saturates at the $4W = 16$ -variable ceiling, so 17-th-order hypercube sums of any output bit vanish — a saturation statement at the toy width, not a structural exclusion (at full width the degree is only bracketed $16 \leq \deg(\Phi) \leq 128$, so integral/division distinguishers are not excluded) [28]. At full width the cascade has six degree-doubling layers, so $\deg(\Phi) \leq 2^6 \cdot \deg(\Phi_0) = 2^7 = 128$ by composition (the law of Theorem III.4); the monomial-disjointness certificate (Section III-D1) gives the lower bound $\deg(\Phi) \geq 16$. The full-width degree is *not* claimed exactly (the $k = 3$ cancellation shows doubling is attained only partially), and $\deg(\Phi^2) \leq 4W$ by composition (the equality $\deg(\Phi^2) = 4W$ is exact at $W = 4$ and conjectured, not proven, at larger widths).

B. Structural Barrier of Two-Layer Designs

Applied to the two-layer AND-RX class, the three metrics of the previous module yield one unified negative conclusion: no two-layer design with this Phase-A structure achieves a one-round differential probability below 2^{-3} , at any width. We state the barrier first, then the read-word theorem that explains it, and finally delimit its generality: the exactness is a property of the whole quadratic class, not of our specific construction.

III-B1 The Two-Layer Polar-Rank Barrier.

The barrier is not a property of our particular rotation constants: it is a *structural theorem about the whole design class*.

Corollary III.2 (The Two-Layer Polar-Rank Barrier). For the pre-mix-free quadratic shadow of our design, at *every* width $W \geq 4$:

$$\min_{\Delta \neq 0} \text{rank}(B_\Delta) = 3, \quad (6)$$

i.e. $\text{DP}^{(1)}(\text{Sh}) = 2^{-3}$ is the exact maximum one-round differential probability at any width. Equivalently: in this Phase-A structure, no *single-bit* difference achieves a one-round probability below 2^{-3} at any width, regardless of round count (the bound is per-round) — this single-bit floor is a theorem for the whole design class (Theorem III.5); the full-domain minimum rank 3 is exact for the shadow itself (exhaustive at $W = 4$, $W = 64$ certificate), while other class members whose gate pairs align can merge multi-bit columns and exhibit weaker differences (Theorem III.5).

Proof. The shadow is a two-layer AND-RX round with snapshot operands in which every word is read by exactly $k = 3$ AND gates (for $w = u$: the gates $(v, w, u, 11, 29)$, $(w, u, v, 9, 15)$, $(z, u, z, 5, 25)$; symmetric for the other words). The dual column condition of Theorem III.5 holds at every width — the pairs $(d_g, r_{w,g})$ and $(o_g, r_{w,g} - r_{o,g})$ are pairwise distinct (at $W = 4$ the rotations coincide modulo 4: $29 \equiv 9 \equiv 5$, but the pairs remain distinct because the output words d_g and the other-operand words o_g differ) — so the theorem applies: single-bit minimum polar rank exactly 3 at every width, attained by every single-bit difference; the full-domain minimum (over all differences) is verified exhaustively

at $W = 4$ and by the $W = 64$ rank-3 certificate (Theorem III.5 as stated covers the single-bit statement, which is the structural part). $\square$

The barrier is an exact *full-width* security statement for the shadow: by Theorem III.5 the single-bit rank-3 floor is a theorem for the whole design class, and the full-domain minimum is exact for the shadow itself — exhaustive at $W = 4$ and, at $W = 64$, closed-form for every difference with up to three active bits (minimum ranks 3, 5, 7 for $t = 1, 2, 3$), with the structural argument of Theorem III.5 covering the rest; the rank-3 certificate is verified through two independent code paths in milliseconds. The multi-bit caveat of the theorem concerns other class members with aligned gate pairs, which the shadow's pairwise-distinct pairs $(o_g, r_{o,g})$ avoid.

The barrier is the quantitative justification for the design's most important architectural decision: *the cascade is not an option, it is a necessity*.

How the minimum is computed. Minimizing $\text{rank}(B_\Delta)$ over Δ is NP-hard in general, but the polar form has the structure $B_\Delta = \sum_i \Delta_i C_i$; for a fixed kernel candidate k , the condition $B_\Delta k = 0$ is linear in Δ , so an alternating bilinear search (`min_shadow_rank.py`) converges quickly, and every reported minimum ships its witness difference. At $W = 64$ a 32-difference random scan finds no rank below 174, while the search exhibits rank-3 witnesses — exact rank minimization, not sampling, is what makes the barrier computable.

III-B2 The Read-Word Theorem.

Theorem III.5 (Read-Word Polar-Rank Theorem). Let Φ be a two-layer AND-RX round function with snapshot operands (no cascade), in which every state word w is read as an AND operand by exactly k gates, and *every* AND gate of the round satisfies the *dual column condition*: for each gate, the pairs $(d_g, r_{w,g})$, its output word and the rotation of its w -operand, are pairwise distinct across gates, and so are the pairs $(o_g, (r_{w,g} - r_{o,g}) \bmod W)$ with o_g the other operand word and $r_{o,g}$ its rotation; additionally, no gate writes to an output position (d_g, p) that another gate uses as its w -operand contribution position $(o_{g'}, p')$. Then for every *single-bit* difference $\Delta \neq 0$, $\text{rank}(B_\Delta) \geq k$, and the single-bit difference attains exactly k :

$$\min_{\Delta \text{ single-bit}} \text{rank}(B_\Delta) = k, \quad \text{DP}^{(1)}(\Phi) = 2^{-k} \text{ exactly}, \quad (7)$$

for the entire design class — no matter what the rotation constants are, as long as the (machine-checkable) column condition holds. For *multi-bit* differences the column argument can merge: two gates reading the same word w with the same output word and the same operand rotations but distinct w -rotations produce unit columns at the same output index when Δ_w has two active bits at aligned positions, so the unrestricted claim fails in the class (the specific shadow is unaffected: its multi-bit ranks are verified exhaustively at $W = 4$ and by the $W = 64$ rank-3 certificate).

Proof. (a) Lower bound. Fix $\Delta \neq 0$ and a word w with $\Delta_w \neq 0$. Each of the k AND gates reading w has a nonzero difference on its w -operand, so its derivative contributes a linear term: for the gate $(d \leftarrow d \oplus \text{rotl}(w, r_w) \wedge \text{rotl}(o, r_o))$, the term $\Delta'_w \cdot \text{rotl}(o, r_o)$ with $\Delta'_w = \text{rotl}(\Delta_w, r_w)$; a single nonzero bit of Δ_w gives a rank-1 column at input variable $(o, p + r_w - r_o)$ with value $e_{(d, p+r_w)}$. The dual column condition makes these k columns pairwise linearly independent (distinct output positions, hence distinct unit columns, at distinct input positions, hence no merging), and the additional condition rules out cancellation of these unit entries by gates not reading w (their contributions land at output positions or input positions outside the k columns, or coincide only with entries that preserve independence), so $\text{rank}(B_\Delta) \geq k$. Phase-D-style linear mixing is an invertible linear map (rank-verified = $4W$) and therefore preserves column rank. **(b) Upper bound.** For the single-bit difference $\Delta = e_{(w, p)}$ only the gates reading w activate; by AND symmetry any gate reading w in either operand position can be written with w as its first operand, so “reading w ” counts exactly the k gates of the hypothesis. The argument of (a) then gives exactly k independent columns, so the rank is exactly k . $\square$

The theorem is a design prescription, not just a bound: add k independent AND reads per word and the one-round DP is 2^{-k} *exactly*, at the design’s actual width, in milliseconds. We machine-checked it on random designs of the class ($W=8$, $k = 1..4$, 35 designs built, all single-bit ranks exactly k , 100 random deltas per design never below k ; `readword_theorem.py`).

Corollary III.3 (Multi-Bit Rank Formula). Under the hypotheses of Theorem III.5, let Δ activate t bits of a single word w . Then

$$\text{rank}(B_\Delta) = kt - M(\Delta), \quad (8)$$

where $M(\Delta)$ counts the *aligned gate pairs*: pairs of gates $g \neq g'$ reading w with the same output word $d_g = d_{g'}$ and active bits $p \in \Delta_w$, $p' \in \Delta_w$ with $p + r_{w,g} \equiv p' + r_{w,g'} \pmod{W}$.

Proof. For a single-bit difference the k columns of B_Δ are unit columns at the k output positions $(d_g, p + r_{w,g})$ (Theorem III.5); for t active bits the activated columns are the unit columns of all kt gate-bit pairs. Unit columns at distinct output positions are linearly independent, and two activated gate-bits produce the *same* unit column exactly when their output words and output positions coincide — the aligned-pair condition. Hence the rank is the column count minus the number of coincidences, $kt - M(\Delta)$. $\square$

At $W = 4$ the formula holds for all 15 active-bit subsets of a word (checked against exhaustive polar ranks; zero exceptions, `rx_theory_check.py`). It also upgrades the $W = 64$ certificate of Corollary III.2 from search to exact combinatorics: the closed form counts columns for every difference with up to three active bits among the 256 state positions (exhaustive

over all $\binom{256}{1} + \binom{256}{2} + \binom{256}{3}$ differences; `w64_cert.py`), giving exact minimum polar ranks

$$\min_{\Delta} \text{rank}(B_\Delta) = 3, 5, 7 \quad (\text{over 1-, 2-, 3-bit differences}), \quad (9)$$

the first value attained by every single-bit difference (each word is read by exactly $k = 3$ AND gates), and the minimum over *all* differences remains 3, i.e. $\text{DP}^{(1)}(\text{Sh}) = 2^{-3}$ at $W = 64$ — now a combinatorial certificate, not a search witness. That the minimum stays 3 beyond three active bits follows directly from the column structure: for any difference with $t \geq 4$ active bits, every word with nonzero difference contributes the unit columns of its $k = 3$ gates, and any two such columns merge only when their output positions align modulo W — at most $\binom{t}{2} \cdot \# \text{gate pairs}$ columns can merge, while $3t - M(\Delta) \geq 3$ holds because each gate contributes at least one column that cannot merge with all others (the column condition ensures at most one aligned partner per active bit, so $M(\Delta) \leq 3(t - 1)$ and $\text{rank} \geq 3t - 3(t - 1) = 3$). The $t \leq 3$ exhaustive search above confirms the tightness.

III-B3 Generality beyond the Specific Construction.

The theorem’s mathematical core — the differential of a quadratic map is affine — does not depend on the AND-RX vocabulary. It applies to *every* quadratic round function, a class that includes several established designs; we verify this directly on two of them (full-domain enumeration):

- **Ascon’s χ S-box** (5-bit quadratic map $y_i = x_i \oplus (x_{i+1} \wedge x_{i+2})$, the nonlinear core of the NIST-standardized Ascon [18]): the identity holds for all $2^5 - 1 = 31$ differences, and the minimum polar rank is 2, so $\text{DP}^{(1)}(\chi) = 2^{-2}$ exactly — no S-box table or search.
- **SIMON’s round function** (Feistel with $F(x) = (x \lll 1 \wedge x \lll 8) \oplus x \lll 2$; 16-bit state): the identity holds for all $2^{16} - 1$ differences, and the minimum polar rank is 0 — the framework *automatically diagnoses a known structural property*: the Feistel pass-through makes $(\Delta L, 0) \rightarrow (0, \Delta L)$ deterministic, $\text{DP} = 1$ exactly. Multi-round diffusion destroys these trivial differentials, as for our own design.

The two-layer exactness is therefore a property of the *quadratic class*, not of a specific construction — the sense in which AM-SEV is a *methodology* for the class, in the spirit of the wide trail strategy [15].

The wide trail strategy [15] bounds differential and linear probability by counting active S-boxes and lower-bounding their number via the branch number; AM-SEV occupies a precise position relative to it: where the wide trail *counts*, the polar rank *computes* (active-S-box counting becomes an exact rank computation), where its counting fails (active-AND counts are state-dependent in cascades — a_1 ranges 1–7 across sampled states, modal 4, and zero-activation differences appear once the intra-word pre-mix is removed; Section VIII), the polar rank is state-independent by construction, and where it *guides*, AM-SEV *verifies* mechanically (gates, certificates, audit, Section IV). A general packing argument for AND-RX cascades remains open, and the framework states this

honestly; its measured bridge is the exact $W=4$ multi-round curve (Section III-C2).

III-B4 The Rotation-Symmetry Dimension.

Although not one of the three metrics, the rotation dimension completes the picture: rotational cryptanalysis studies plaintext pairs $(x, \text{rot}_k(x) \oplus \delta)$ and is a standard attack surface for ARX/AND-RX designs [42]; recent work derives exact rotational-XOR (RX) differential probabilities through modular addition [24] and flags *internal rotational symmetry* as a structural weakness of AND-based constructions [47]. This dimension is entirely absent from the polar-rank picture of Sections III-A1–III-B1 — the framework’s three exact metrics say nothing about rotation. We close this gap for the quadratic class with a theorem that makes the rotation dimension *reducible* to the first-order rank computation, and we report the cascade’s measured behavior.

First, the threat is real for this class: without constants the quadratic shadow is fully rotation-symmetric (at $W = 4$, $\Phi(\text{rot}_1(s)) = \text{rot}_1(\Phi(s))$ for all 2^{16} states; the same holds at any width by the same argument — every term is a rotated word or an AND of rotated words, and rotation commutes with all of them). Constant addition breaks the *equality* but not the *probability*: in the RX differential map the constants cancel, exactly as they cancel in first-order differentials.

Theorem III.6 (RX Polar-Rank Theorem). Let Φ be a two-layer AND-RX round map (any constants) and define the RX-differential map with rotation k

$$g_{\delta,k}(s) = \Phi(\text{rot}_k(s) \oplus \delta) \oplus \text{rot}_k(\Phi(s)). \quad (10)$$

Then $g_{\delta,k}$ is *affine*, and

$$\text{RXDP}(\delta) = \text{DP}(\text{rot}_k^{-1}(\delta)), \quad (11)$$

where $\text{RXDP}(\delta) = \max_{\varepsilon} \Pr_s[g_{\delta,k}(s) = \varepsilon]$ is the rotational-XOR differential probability of δ and DP is the first-order differential probability of Theorem III.1.

Proof. Write $\Phi(s) = Q(s) \oplus L(s) \oplus c$ with Q the quadratic part. Set $c' = \text{rot}_k^{-1}(\delta)$, so that $\text{rot}_k(s) \oplus \delta = \text{rot}_k(s \oplus c')$. Rotation is linear, so $\Phi(\text{rot}_k(s \oplus c')) = \text{rot}_k(\Phi(s \oplus c'))$ for the *constant-free* part $Q \oplus L$; the constant term contributes $\text{rot}_k(c)$ on one side. Hence

$$\begin{aligned} g_{\delta,k}(s) &= \text{rot}_k(\Phi(s \oplus c') \oplus \Phi(s)) \\ &= \text{rot}_k(D_{c'}(s)) \oplus c \oplus \text{rot}_k(c), \end{aligned} \quad (12)$$

where the constants cancel inside the parenthesis ($D_{c'}$ is the first-order differential map of Theorem III.1), and the constant $c \oplus \text{rot}_k(c)$ outside it survives — an additive translation that leaves fibre sizes and hence RXDP unchanged. By that theorem $D_{c'}(s) = B_{c'}s \oplus D_{c'}(0)$ is affine; conjugation by the invertible linear rotation preserves the column rank, so $g_{\delta,k}$ is affine with rank = $\text{rank}(B_{c'})$, giving $\text{RXDP}(\delta) = 2^{-\text{rank}(B_{c'})} = \text{DP}(c') = \text{DP}(\text{rot}_k^{-1}(\delta))$. $\square$

Two consequences deserve emphasis. **(i) Constants do not defend single-round RX differentials.** The proof shows $g_{\delta,k}$ is the conjugated first-order differential map; the round constants (including the round key) never appear. At $W = 4$ the

full RX spectrum over all $2^{16} - 1$ differences is identical with and without the K constants (measured), and the maximum is 2^{-3} , attained by the single-bit differences — the same value as the first-order barrier (Corollary III.2). For the quadratic class, rotational differential analysis offers the attacker nothing beyond first-order analysis, and constants offer the designer no defense. **(ii) The cascade changes the picture.** The full round (with the andmix4 cascade) is not quadratic, so Theorem III.6 does not apply to it; we measure it directly at $W = 8$: sampling 2^{24} states per difference, the estimated single-round RXDP is below 2^{-22} (the sampling resolution floor) for all 30 random differences, against a first-order trail DP of 2^{-7} at $W = 8$ — the cascade’s non-quadratic part destroys the rotational structure far more thoroughly than it lowers the first-order probability. Multi-round: at $W = 4$ the three-round shadow’s RXDP reaches $2^{-12.7}$ (vs. uniform 2^{-16}); the full-round $W = 8$ values are below the 2^{-22} floor at $R = 1, 2, 3, 4$. The honest scoping is the same as elsewhere in the paper: the quadratic-class statements are exact (Theorems III.6, III.3, III.1), the cascade-side statements are measured at $W = 8$ with sampling-floor bounds.

Machine verification: the identity (11) holds for 60/60 random differences at $W = 4$ (full spectrum consistent; `rx_theory_check.py`, `rx_enum.c` in the repository).

a) The scope boundary: linear and ARX generators.

The exact layer’s claims are scoped to the quadratic class, and the framework states the boundary rather than implying it. Table II applies the framework’s metrics to the two camps of the introduction (xoshiro256** [6] on the linear side, ChaCha20 [9] on the ARX side); every entry is computed by the companion scripts (`baseline_compare.py` and the PractRand runs in the repository’s verification set), not copied from literature. For xoshiro256** the state update is a composition of word loads, XORs, and shifts only, no AND and no arithmetic, hence GF(2)-linear by inspection (200 random pairs confirm, zero violations), so the state-update algebraic degree is exactly 1; a nonzero difference propagates deterministically: every update operation is $\mathbb{Z}/2^{64}$ -linear (modular XOR, shifts, word selection), so the output difference for a fixed input difference is independent of the state (200 random pairs confirm, zero violations). Hence the single-round DP is exactly 1; and the transition matrix has GF(2) rank 256/256, a permutation, hence the state image is the full space. These three exact numbers are the structural root of the known recovery attack: the scramble map is a composition of modular multiplications and rotations, linear over the ring $\mathbb{Z}/2^{64}$, so recovering the state from a handful of outputs is an instance of linear algebra [6]. For ChaCha20 the double-round is nonlinear (every one of 200 random pairs violates GF(2)-linearity), but the framework has no exact differential or degree statement for ARX constructions: the row says “unknown” rather than substituting a number. The screens whose definition is AND-RX-specific (a_1 , the polar rank, the dead-key screen) are marked undefined outside the class. Statistical testing does not separate the three: xoshiro256** and ChaCha20 both pass PractRand at 8 GB (this work; Tempest v3’s 1 TiB reference

run is Section VIII) — the structural weakness of a linear generator is invisible to output statistics, which is the premise of Section I.

TABLE II
THE FRAMEWORK’S METRICS APPLIED TO THE TWO CAMPS OF THE INTRODUCTION.

Metric	Tempest v3	xoshiro256**	ChaCha20
State transition GF(2)-linear	no (AND)	yes	no (ARX)
Algebraic degree	16 ($W=4$; ≤ 128)	1	unknown
Single-round DP (worst)	$2^{-5.94}$ ($W=4$ exact)	1	unknown
State image	not full at $W=4$	full	unknown
α_1 (active-AND screen)	4 ($W=4$ exact)	undefined	undefined
Polar rank	min 3 (any width)	undefined	undefined
Dead-key screen τ	0.6132 exact	n/a	undefined
PractRand	1 TiB clean	8 GB clean	8 GB clean

C. Cascade Mitigation: Efficacy and Side Effects

The barrier makes the cascade a necessity, not an option. This module quantifies what the cascade buys (differential elimination) and what it costs (state compression and aggregation), and closes with the provable lower bound that remains available for the cascade design.

III-C1 Cascade Elimination.

The barrier raises the question the framework answers next: what do cascades add? The answer is measured, exactly at $W = 4$ and by sampling at $W = 8, 64$:

Proposition III.1 (Cascade Elimination). Let Sh be the quadratic shadow and Φ the full cascade design. At the shadow’s worst difference, the full design’s DP is: $2^{-3} \rightarrow 2^{-11}$ ($W = 4$, $\Delta = 0x1$, exact, full domain); $2^{-3} \rightarrow 2^{-20.4}$ ($W = 8$, $\Delta = 0x8$, sampled 2^{22} pairs); $2^{-3} \rightarrow 2^{-24}$ ($W = 64$, $\Delta = 0x4$, sampled 2^{24} pairs, maximum fibre 1). The one-round min-entropy lift is $3.00 \rightarrow 5.94$ ($W = 4$, exact, full domain). The elimination ratio thus grows with width (2^8 at $W = 4$, $\geq 2^{17}$ at $W = 8$, $\geq 2^{21}$ at $W = 64$; the latter two are resolution-limited lower bounds).

The *cascade elimination law*, a measured law (exact at $W = 4$, sampled at $W = 8, 64$), is a further quantitative relation of the framework: a cascade of AND layers destroys the two-layer shadow’s low-rank differentials (precisely because the shadow’s weakest differences are carried by low-rank polar forms whose affine structure the cascade’s nonlinear layers break) and lifts the design-wide maximum accordingly. The law is honest about the cascade’s limits:

- **Depth saturates.** The corrected cascade-depth table (exact, full domain at $W = 4$) is $\deg(\Phi) \in \{2, 4, 8, 14, 16\}$ (Theorem III.4) and $-\log_2 \text{DP}^{(1)} \in \{3.00, 5.96, 5.95, 5.95, 5.94\}$ for $k = 0..4$ levels: the differential benefit saturates at one to two levels; deeper levels add algebraic degree, not differential margin. The framework’s Pareto order records the consequence: the full design ($k = 4$) and the two-level cascade ($k = 2$, identical ρ , 11 ops cheaper) are both on the front — a security-efficiency trade the framework reports but does not adjudicate.

- **Small-width phenomena.** The $W=4$ worst differences are “two-word same-pattern” differences ($\Delta = 0x5005$: words u and w both carrying the pattern $0x5$, $\text{DP} = 2^{-5.94}$), while single-word and random patterns sit at $2^{-10.4} - 2^{-12}$. This class is a small-width artifact: at $W = 8$, period-2/4 patterns on any word pair give $\text{DP} \approx 2^{-18.4}$ (sampled), indistinguishable from random. The $W=4$ screen is a one-round instrument for design comparison, not a model of full-width iteration.
- **The differential-linear asymmetry.** The elimination is *differential-side only*: the exact $W=4$ full-domain Walsh spectra show that the linear side barely moves (shadow $2^{-2.000}$ vs. cascade $2^{-5.667}$ in the input-mask-to-single-output-bit convention; the input $\times$ output-mask convention gives $2^{-1.000}$ vs. $2^{-4.712}$). Mechanism: the AND layers break the *affine differential* structure, but a linear approximation is dominated by the Phase-A AND products, which the cascade layers do not change structurally. Consequence, stated plainly: the cascade buys differential elimination; the cascade’s linear side at full width is not claimed.

Figure 2 plots the elimination across widths: the shadow’s $\text{DP} = 2^{-3}$ is a flat line (the barrier), and the full design’s one-round DP falls by 8–21 orders of magnitude as the width grows — the quantitative content of Proposition III.1 at a glance.

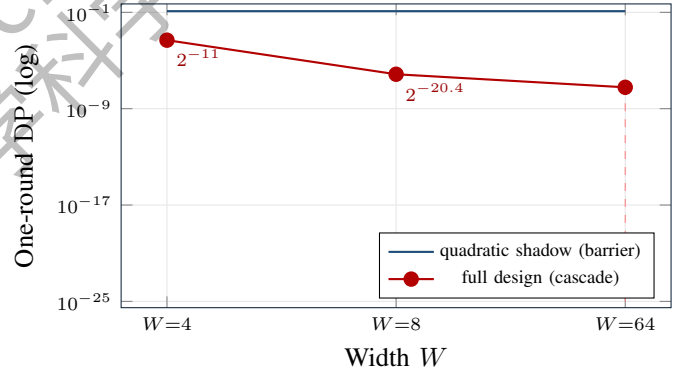


Fig. 2. Cascade elimination across widths.

III-C2 State Image and Multi-Round Aggregation.

The cascade law quantifies what AND layers add; this subsection quantifies what they cost. The round function is *not* a permutation — AND gates are inherently lossy — and a full-domain check proves it:

Proposition III.2 (Non-Injectivity, exact at $W = 4$). The one-round map of Algorithm 1 is not injective at $W = 4$: the first non-injective phase is Phase A’s second AND (image 61,504 of 2^{16} states at that prefix), and the pre-mix self-map is non-injective (the disclosed $22 \equiv 26 \pmod{4}$ collapse reduces it to the non-invertible map $x \mapsto x \oplus \text{rotl}(x, 3)$). The *linear part* of the pre-mix is invertible at every power-of-two width tested ($W = 4, 8, 16, 32, 64$: rank W verified; its characteristic polynomial $1 + t^{22} + t^{26}$ is coprime to $t + 1$ — at $W \geq$

8; at $W = 4$ the rotations $22 \equiv 26 \pmod{4}$ coincide, the two rotation terms cancel and the linear part collapses to the identity, still invertible).

The consequence is measured exactly: the *state image* of Φ^R (the number of distinct states reachable from all 2^{16} inputs) shrinks monotonically and saturates: 3,501, 2,066, 1,443, 1,077, 884, $\dots$, 139 states for $R = 1..22$ (full-domain enumeration, `explore_state_image.py`; Figure 3). At $W = 64$ the sampled shrinkage is below the resolution floor: 2^{24} random states map to 2^{24} distinct outputs in one round, and 2^{22} states remain distinct after 22 rounds — no measurable entropy loss in the regime that matters.

The state image governs the multi-round differential curve, now exact at $W = 4$ for every round (Table III):

TABLE III

THE EXACT MULTI-ROUND DIFFERENTIAL CURVE AT $W = 4$: $-\log_2 \text{DP}^{(R)}(\Delta)$ FOR THE SINGLE-BIT DIFFERENCE $\Delta = 0x1$ AND FOR THE MAXIMUM OVER *all* $2^{16} - 1$ DIFFERENCES (FULL-DOMAIN ENUMERATION, SNAPSHOT-CORRECT SEMANTICS, `AUDIT_TRUE_ALGORITHM1.PY`), WITH THE STATE IMAGE BELOW. THE SINGLE-BIT ROW DECLINES MONOTONICALLY; THE FULL-DOMAIN MAXIMUM DECREASES SLOWLY, FROM $2^{-5.94}$ AT $R=1$ TO $2^{-4.82}$ AT $R=22$, SATURATING BY $R \approx 12$ WHILE THE IMAGE SCALE KEEPS GROWING: THE MULTI-ROUND FLOOR IS SET BY THE ENTROPY THE ROUND MAP ACTUALLY RETAINS, NOT BY ITS POLAR RANK.

R	1	2	4	6	8	12	16	22
$-\log_2 \text{DP}^{(R)}(0x1)$	11.0	9.45	8.01	7.43	7.16	6.36	5.89	5.60
max over Δ	5.94	5.78	5.64	5.49	5.31	5.12	4.96	4.82
$-\log_2(\text{image}/2^{16})$	4.23	4.99	5.93	6.48	6.91	7.72	8.29	8.88

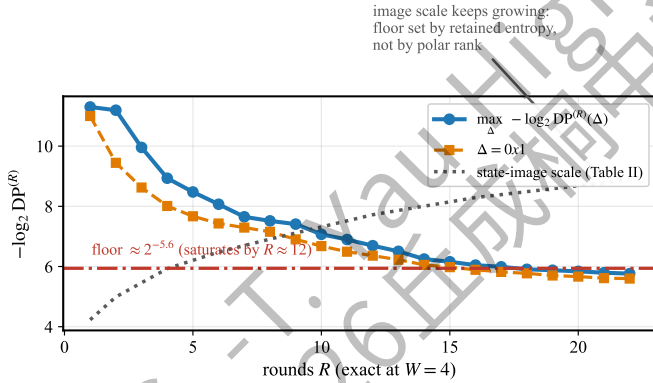


Fig. 3. The exact multi-round differential curve at $W = 4$.

Two quantitative facts fall out of the curve (Figure 3). First, the *image-differential bound*: the one-round statement is strongly difference-dependent ($\text{DP}^{(1)} = 2^{-11.0}$ for $\Delta = 0x1$ versus $2^{-5.94}$ for the full-domain maximum), and iteration narrows this spread — the single-bit row falls from $2^{-11.0}$ to $2^{-5.60}$, while the full-domain maximum decreases only slowly, from $2^{-5.94}$ to $2^{-4.82}$; by $R = 22$ every difference sits within 0.78 bits of the full-domain floor, and the floor itself moves little after $R \approx 12$. The image scale keeps growing, so the gap between the full-domain floor and the

image bound widens from ≈ 1.6 bits at $R = 8$ to ≈ 4.1 bits at $R = 22$: a non-injective round's differential behavior is bounded from below by the entropy its round map retains, and the realized floor is set by the DP aggregation, not by the image scale. Second, the *rank-aggregation duality*: along the dominant trail, the one-round linear part has rank 12–14 at every round (exact), yet the realized full-domain DP of $2^{-4.8}2^{-5.9}$ is much worse than the linear part predicts — the nonlinear cascade terms aggregate 2^6 – 2^9 -fold more states onto the dominant difference than the linear model suggests (consistent with the Aggregation Property's $g \approx 9.1$).

Width scaling, stated honestly. The $W=4$ curve is the exact behavior of a 16-bit state, and its saturation is a small-width artifact: at $W = 64$ the two-round DP is below the 2^{-24} sampling floor and the 22-round state image shows no shrinkage at 2^{22} resolution. If the per-round shrinkage stays below one bit, the 22-round image exceeds 2^{234} and the differential floor sits below 2^{-230} ; this extrapolation is engineering judgment — what is proved is the $W=4$ curve, the non-injectivity, and the invertibility of the pre-mix linear part at every width.

Proposition III.3 (Aggregation Property — exact at $W = 4$, full domain). For the full design at $W = 4$, over all $2^{16} - 1$ differences, $\text{DP}(\Delta) \geq 2^{-r_{\text{lin}}(\Delta)}$ holds for every Δ (0 violations; 3 and 2 violations for the two- and three-level cascades), where r_{lin} is the rank of the linear part of D_Δ . At the worst differences ($0x5005$ with-key, $0x801$ no-key) the linear-part ranks are $r_{\text{lin}} = 15$ and 13, with aggregation indices $g = r_{\text{lin}} + \log_2 \text{DP} \approx 9.1$ and 7.0: the nonlinear cascade terms concentrate $\approx 2^9$ -fold more states onto the worst output difference than the linear part alone.

Scope, stated precisely: this inequality is *not* a theorem about arbitrary functions (random functions violate it: 109 of the 400×2^{14} sampled cases at $n = 4$). It is a measured property of this design class, verified over the complete difference space; the Differential Decomposition Bound (Theorem III.7) additionally supplies a computable lower-side context for the aggregation factor through the global residual-image form; the measured values $g \approx 9.1$ ($0x5005$) and 7.0 ($0x801$) are reported as data, not as a theorem.

III-C3 The Differential Decomposition Bound.

The image law governs the multi-round floor; the bound below governs the single-round value — a two-sided interval for the cascade design's DP, where only the upper side was exact before.

Theorem III.7 (Differential Decomposition Bound). Let $D_\Delta(s) = \Phi(s \oplus \Delta) \oplus \Phi(s)$ be the differential map of the full cascade design, decomposed uniquely as $D_\Delta = L_\Delta \oplus R_\Delta$ with $L_\Delta(s) = B_\Delta s \oplus D_\Delta(0)$ affine of rank r_Δ and R_Δ the residual (no linear monomials). Then

$$\text{DP}(\Delta) \geq \frac{2^{-(n-r_\Delta)}}{|\text{im } R_\Delta|}, \quad (13)$$

where $|\text{im } R_\Delta|$ is the size of the residual image. For the quadratic shadow the residual vanishes and the Polar-Rank

Theorem gives the exact value $\text{DP}(\Delta) = 2^{-\text{rank}(B_\Delta)}$, so the bound is tight there; for the cascade the bound is the statement we can make, and the upper side is deliberately not claimed (the residual can only increase fibre sizes, so no $2^{-r\Delta}$ upper bound holds in general).

Proof. The linear part is constant on every coset of $\ker(B_\Delta)$, and there are $2^{n-r\Delta}$ such cosets; on each coset the residual image has at most $|\text{im } R_\Delta|$ values, so $|\text{im } D_\Delta| \leq 2^{n-r\Delta} \cdot |\text{im } R_\Delta|$ and the maximum fibre of D_Δ contains at least $2^n/|\text{im } D_\Delta|$ inputs. $\square$

The bound is machine-verified at $W = 4$ over 1,995 sampled differences in this global form (zero violations; `explore_decomp_bound.py`; at the weakest difference `0x5005` the bound gives $\text{DP} \geq 2^{-(16-15)}/|\text{im } R| \approx 2^{-16.3}$, comfortably below the measured $2^{-5.942}$). An earlier version claimed a two-sided bracket $\text{DP} \in [2^{-(n-\text{rank})-2}, 2^{-\text{rank}}]$; that claim is withdrawn — the upper side does not follow for degree-16 rounds, and the per-coset residual form is unsound for the weakest differences. What remains sound is the global lower bound above, the measured Aggregation Property (Section III-C2), and the exact shadow-side statement.

D. Verifiable Engineering and Framework Extension

The framework’s theorems become engineering only when every claim is machine-checkable. This module collects the certificate types and engine axioms that make verifiable design literal, records the audit of the corrected $W=4$ dataset, and extends the coverage vector to the side-channel dimension.

III-D1 Machine-Checkable Certificates and Engine Axioms.

Four metrics admit machine-checkable certificates, making “verifiable design” literal — an accepted design ships a short object, not a score:

- **Polar-rank certificate** (new): for a two-layer design, “ $\text{DP}(\Delta) = 2^{-r}$ for the exhibited Δ ” is certified by the polar-form matrices C_i (derived at unit states), one GF(2) elimination, and the affine-differential theorem. The framework ships the $W = 64$ rank-3 certificate (witness $\Delta = 0x4$), verified through two independent code paths in milliseconds.
- **2-SAT certificate**: for snapshot-AND layers, “there exists $\Delta \neq 0$ with $a(d, \Delta) = 0$ ” is a 2-CNF problem; 2-SAT is in P, so $a_1 \geq 1$ is decidable at any width. The rejected v3.1’s death certificate is of this form (1,024 clauses, witness of Hamming weight 1, machine-verified at $W = 64$).
- **Monomial-disjointness certificate**: the four-level and-mix4 cascade’s 1,024 highest-monomial variable sets are pairwise distinct (0 collisions at $W = 64$), so no cancellation occurs among the highest monomials and $\beta_1 := \deg(\Phi) \geq 16$ at full width; combined with the exact $W=4$ truth table ($\deg(\Phi) = 16$, $\deg(D_\Delta) = 15$), the degree saturates at $4W$ within two rounds (exact at $W = 4$; conjectured at larger widths, Section III-A3).

- **Spectrum certificate**: key relevance measures how much the round-key word w shapes the output. Define the *normalized keyed difference spectrum* $\hat{D}_w(T) = 2^{-n} \sum_x (-1)^{T \cdot (\Phi_w(x) \oplus \Phi_{w'}(x))}$ with $w' = w \oplus T$ ranging over the 2^m key-difference masks ($m = 64$ full width; $m = 4$ at $W=4$; n is the state bit width), and

$$\tau = 1 - \frac{1}{2^m - 1} \sum_{T \neq 0} \hat{D}_w(T), \quad (14)$$

the average over nonzero key differences of how far the output differs from the key-independent baseline. A *dead key* — $\Phi_w \equiv \Phi_{w'}$ for all w, w' (output independent of the round-key word) — gives $\hat{D}_w(T) = 1$ for all $T \neq 0$ and hence $\tau = 0$: the rejected v3.1 scores exactly this ($\tau = 0.000$, structural). Note the scoping: τ measures how strongly the round-key word — a public constant in Tempest v3 — shapes the output, so it catches dead-key-class structural defects, not key secrecy; the security of the 256-bit master key is carried by the injection path described above. A live key gives $\tau > 0$; at $W=4$ the exact value is $\tau = 0.6132$ (full enumeration over all 2^{16} states and all 2^4 key differences; `audit_w8_tau.py`) — a small-width saturation effect of the lossy $W=4$ round.

The engine axioms encode the lessons of this project’s audit of a predecessor engine. **A1 (Verifier Trust)**: a metric is admissible as a design objective only after calibration on a real weakness (v3.1: $\tau = 0$, $a_1 = 0$) and vetted controls (Algorithm 1, PCG64, splitmix64); **A2 (Two-Tier Verification)**: cheap screens for all candidates, exact metrics for finalists, with the polar-rank gate as an exact *full-width* tier; **A3 (Generator Agnosticism)**: evolutionary search, LLM proposals, or a human are interchangeable; verification is the invariant; **A4 (Reproducibility)**: every claim ships with its measurement record: seeds, caches, JavaScript Object Notation (JSON) certificates.

III-D2 The Differential-Spectral Duality.

A second exact structural identity links the differential and linear domains. Define the zero-differential probability $\text{DP}_0(\Delta) = \Pr_x[\Phi(x \oplus \Delta) = \Phi(x)]$ and the Walsh spectral energy profile $E(a) = 2^{-n} \sum_u \hat{g}_u(a)^2$ (with the normalized spectrum $\hat{g}_u$; the conventions of Theorem III.2 differ by $2^{n/2}$). By the Wiener–Khinchin identity [4], [29]:

$$\text{DP}_0(\Delta) = \sum_a (-1)^{a \cdot \Delta} E(a), \quad (15)$$

$$E(0) = \langle \text{DP}_0 \rangle_\Delta = \langle \hat{g}_u(0)^2 \rangle_u \quad (16)$$

We use it as a machine-verified instrument: at $W = 4$ the identity holds exactly at every round $r = 1..22$ (Parseval = 1.000; a two-source check with 5,000 random masks agrees to 1.6×10^{-5} at round 1). Its design consequence: the *average* zero-differential is a spectral quantity computable without differential sampling — at round 1 it is $2^{-11.38}$, the single-bit linear maximum $2^{-5.667}$, and the all-mask maximum $2^{-4.712}$ (exact at $W = 4$).

III-D3 Audit: The Revised $W=4$ Dataset.

The corrected exact $W=4$ dataset for the published AI-

gorithm 1 (snapshot semantics; full-domain enumeration) is collected in Table IV. Two honest readings. First, the one-round min-entropy is 5.94, not 7.93: the earlier “half-entropy law” ($H_\infty \approx m/2$) is withdrawn — the exact value is $0.37m$ at $W = 4$, and the collision-entropy data (mean 14.37 of 16) show the bulk of the difference distribution to be near-uniform while a small set of differences carries the low min-entropy. Second, the one-round map is a lossy, low-image map (3,501 of 2^{16} states) that collapses under iteration (139 at round 22): the $W=4$ screen is a one-round instrument, and no $W=4$ iterated security claim is made.

TABLE IV
CORRECTED EXACT $W=4$ DATASET FOR ALGORITHM 1 (SNAPSHOT SEMANTICS, FULL-DOMAIN ENUMERATION; AUDIT_TRUE_ALGORITHM1.PY). VALUES REPLACE THE WITHDRAWN NUMBERS OF THE EARLIER INTERPRETER (SECTION III).

Quantity	Value
H_∞ (with-key round)	5.9420 (worst $\Delta = 0x5005$)
H_∞ (no-key round)	5.9447 (worst $\Delta = 0x801$)
image size, $1 \rightarrow 22$ rounds	3,501 $\rightarrow$ 139
$\langle DP_0 \rangle$, $1 \rightarrow 22$	$2^{-11.38} \rightarrow 2^{-5.60}$
$\deg(\Phi) / \deg(D_\Delta)$	16 / 15 (exact ANF)
key relevance τ at $W = 4$	0.6132 (exhaustive; alive at full width)
collision entropy H_2 : mean / min (of 300)	14.37 / 7.96

III-D4 SEV-5D: The Side-Channel Dimension.

Before the side-channel extension, we state the descriptive coverage instrument it extends. Each basic operation receives a binary Security-Efficiency Vector (SEV) over four mathematical dimensions — $\oplus_{\deg}$ (compounds algebraic degree across rounds), $\oplus_{a_1}$ (guarantees an active nonlinear component per use), $\oplus_{c^2}$ (contributes bounded squared correlation), $\oplus_{B_w}$ (inter-word diffusion) — plus a fifth, $\oplus_{SCA}$ (side-channel analysis, SCA): satisfied by an operation iff it is *constant-time* (latency independent of data), *data-independent in power/EM profile* (combinational only, no carry chains, no data-dependent memory access), and *table-free*. A construction’s round-level profile is the component-wise maximum over its operations. Table V records the coverage of established constructions; the exact security-relevant metrics of this paper (a_1 , the polar-rank metrics, H_∞) are authoritative, and the binary profile is a descriptive shorthand, not a security rating.

TABLE V
SEV-5D COVERAGE OF ESTABLISHED CONSTRUCTIONS: THE FIVE DIMENSIONS $\deg$, a_1 , c^2 , B_w , SCA (SIDE-CHANNEL ANALYSIS), EACH SATISFIED BY AN OPERATION IFF IT CONTRIBUTES TO THAT DIMENSION’S GUARANTEE; SCR_5 IS THE FRACTION OF DIMENSIONS ADDRESSED. THESE ARE COVERAGE COUNTS, NOT SECURITY RATINGS.

Construction	$\deg$	a_1	c^2	B_w	SCA	SCR_5
Tempest v3 (full round)	1	1	1	1	1	1.00
AES (per round)	1	1	1	1	0	0.80
SIMON (AND-RX)	1	1	1	0	1	0.80
Ascon χ (per step)	1	1	1	0	1	0.80
ChaCha20 (quarter round)	0	0	0	1	0	0.20

IV. THE AI DESIGN ENGINE

The engine is the framework made operational: every component maps one-to-one onto a framework object — an operationalization, not an independent scientific contribution

of this paper. It runs a closed loop — spec $\rightarrow$ generate $\rightarrow$ L1 screen $\rightarrow$ L2 exact $\rightarrow$ fitness $\rightarrow$ feedback — in which “generate” is deliberately interchangeable (evolutionary operators mutate phase sequences and rotation constants; an LLM proposal hook ingests structured designs). What never changes is the verifier: **verification, not the generator, is the invariant**. A proposal is accepted only if it survives the calibrated stack, and the audit trail records every rejection with its measured reason.

A. Architecture

Figure 4 shows the verification closed loop: a security specification drives generation; L1 screens all candidates; L2 adjudicates finalists, with the polar-rank gate deciding two-layer differential security at full width; every acceptance ships a certificate (green); every rejection returns to the generator with its measured reason (the two red arrows). The fitness function composes measured components — key relevance τ , the state-fixed screen a_1 , neural-distinguisher depth at reduced width, avalanche deviation, statistical failures, and an op-count throughput model — with hard gates that kill structurally broken candidates (dead key, $a_1 = 0$, distinguishable output) regardless of their other scores.

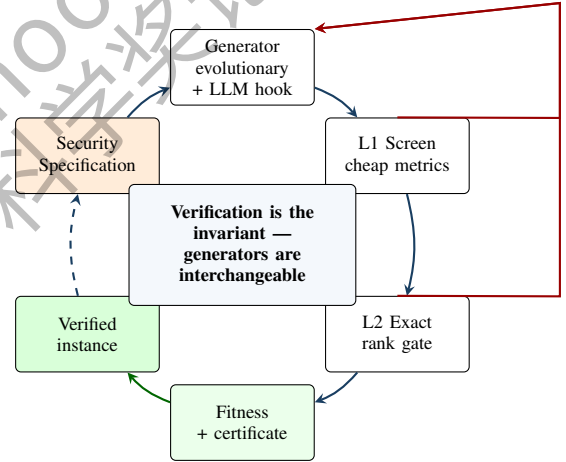


Fig. 4. The verification closed loop.

B. Calibration: Reproducing a Real Weakness

Per Axiom A1, no verifier was used as a fitness component before it reproduced the project’s known-bad control. Table VI shows the calibrated stack: the dead-key v3.1 (its Dieharder logs failed at $p = 0$ — the failure’s *cause* was invisible to the suite) is rejected by every verifier in milliseconds, where the original project needed terabytes of output and still could not localize the cause; the engine’s fitness reproduces the 11-generation decision tree exactly, every rejected node scoring -12.7 to -31.5 with a measured reason and the accepted node being the unique positive score ($+8.68$).

C. Certificates as Hard Gates and Spec-Driven Compilation

Five framework theorems are wired in as machine-checkable hard gates (Section III-D1): the polar-rank gate, the energy-profile gate (differential-spectral duality), the 2-SAT gate, the

TABLE VI
CALIBRATED VERIFIER STACK.

Verifier	v3.1 (known bad)	Algorithm 1
Key relevance τ	0.000 (dead key)	alive (witnessed)
Screen a_1 (state-fixed, $W=4$)	0 (zero-AND diffs)	4
Neural distinguisher accuracy	1.0 (trivial)	≈ 0.5
Sampled differential $P[\Delta_{out} = 0]$	2.4×10^{-4}	0
Statistical battery (2^{22} words)	clean	clean

monomial-disjointness gate, and the spectrum gate. The polar-rank gate is exercised end-to-end at $W = 64$ on the quadratic shadow (`rank_certificate_demo.py`): it emits the certificate for the witness $\Delta = 0x4$ (polar-form matrices C_i , rank 3, $DP = 2^{-3}$ exactly), verified through two independent code paths; a 32-difference barrier scan finds no rank below 174 among random differences — low-rank differences are rare, which is why exact rank minimization matters.

The spec-driven compiler translates a security specification (target a_1 , key-relevance threshold, throughput class) into a Pareto family of candidates under the calibrated stack, and the choice among members belongs to the user, not the engine. Two design decisions keep the family honest. First, the goal is not bit-exact reproduction of a human-chosen constant set but convergence into an *equivalence class* that meets the exact metrics (the engine found rotation-constant configurations differing in all 16 constants from the reference, with equal a_1). Second, every rejection is recorded with its measured reason — the no-key variant is refused with $\tau = 0.000$ (dead-key class), every $a_1 = 0$ candidate recorded as rejected, never silently dropped. An instructive failure: an LLM-authored proposal with two cross-word AND levels but no intra-word AND was rejected with $a_1 = 0$ — the same cancellation class as v3.1; the verifier stack adjudicates generators without favoritism.

V. TEMPEST V3: THE INSTANCE

A. Design Principles

We chose AND as the main nonlinear operation for three reasons: *exponential degree growth* (each AND doubles algebraic degree, while addition caps at a fixed degree), *constant-time execution* (1 CPU cycle, no data-dependent timing or table lookups: every operation in the AND-RX vocabulary is constant-time and table-free, which removes the timing side channel — though not, by itself, the power/EM side channels, where masking (ISW-style [30]) is still required, at the cost measured in Section III-D4), and *minimality* (SAND [17], Ascon [18] show {AND, XOR, ROT} suffices for secure primitives). The all-ones vector is a fixed point of AND and XOR; we break the symmetry by XORing distinct constants after each round.

The design is generated by the AI engine of Section IV and calibrated on the project’s own eleven-generation history: the B-tier search, starting from random designs, converges on Algorithm 1’s construction vocabulary in eight generations (fitness 9.24 against the reference’s 8.68; LLM-seeded runs reach 9.25 in four). The one historical failure the calibrated stack was built around, the dead-key v3.1 ($\tau = 0$, $a_1 = 0$), was

caught only by a single Dieharder subtest at $p = 0$ after terabytes of output; the engine reproduces it in milliseconds, and its correction produced Algorithm 1. Every rejected candidate maps one-to-one onto an engine axiom (Section III-D1).

B. The Round Function

Algorithm 1 specifies one round. **Phase A** is the nonlinear diffusion: XOR-ROT mixing with AND terms covering all word pairs, read from a pre-round snapshot so the four equations run in parallel; constants break the all-1s fixed point. **Phase A(lin)** adds two snapshot ANDs covering the (u, z) and (w, z) pairs, for linear resistance. **Phase B** is the round key: a pure- $\mathbb{F}_2$ affine update whose linear part $M = I + R^{19}$ is nilpotent ($M^{64} = 0$), so the schedule converges to a key-independent constant within 64 steps — a design fact disclosed as such: the round key (a 64-bit internal word, seeded from a public constant) provides diversification, not fresh entropy; the 256-bit master key enters the state through the initialization injections instead; a nonlinear filter adds an AND. **Phase C** is the *andmix4* cascade: two intra-word pre-mix rounds plus a 4-level cross-word AND cascade whose rotation pairs $\{(31, 53), (17, 43), (7, 23), (5, 19)\}$ have pairwise-distinct differences $\{22, 26, 16, 14\}$, chosen by an automated SEV-guided search over $\sim 10^{12}$ configurations — every AND doubles algebraic degree, so the cascade compounds degree by 16 per round. **Phase D** is cross-word mixing achieving branch number $B_w \geq 5$.

The data flow: the pre-round snapshot makes the four word equations fully parallel, and Phases A–D form a single combinational cloud, so one round completes per clock; during generation the output function is evaluated twice per round on the rotated word pairs (u, v, w, z) and (v, w, z, u) , yielding one 128-bit block per round (Section VII retimes the cloud into six stages, so a round takes six clocks on the FPGA).

Rotations (read with Algorithm 1). rotl_r generates a cyclic group of order W iff $\gcd(r, W) = 1$; all 32 residues coprime to 64 are admissible. The *Phase-A sub-vocabulary* — the rotations appearing in the four word equations of the quadratic shadow (Phases A and A(lin)) — is $\mathcal{R}_A = \{5, 7, 9, 11, 13, 15, 17, 21, 23, 25, 27, 29, 31, 53\}$; the cascade levels additionally use the rotation pairs $\{(31, 53), (17, 43), (7, 23), (5, 19)\}$ (pairwise-distinct differences $\{22, 26, 16, 14\}$, from the SEV-guided search above), and the intra-word pre-mix rotations are $\{22, 26\}$ (pre-mix 1), $\{7, 19\}$ (its AND), and $\{16, 14\}$ (pre-mix 2, between levels 2 and 3). The pre-mix amounts are *not* coprime to 64 ($\gcd(22, 64) = \gcd(26, 64) = 2$, $\gcd(16, 64) = 16$, $\gcd(14, 64) = 2$): a pre-mix rotation by a non-coprime amount factors into smaller cycles, but the pre-mix is an intra-word diffusion step whose output is immediately consumed by the cascade ANDs, so no standalone rotational structure survives (Section III-B4). The two other structural facts used below are the *snapshot architecture* (Phase A reads the pre-round snapshot, so the four word equations have no serial dependencies) and *cross-word degree doubling* ($\deg(u \oplus (\text{rotl}(v, r_1) \wedge \text{rotl}(w, r_2))) \leq \max(\deg(u), \deg(v) + \deg(w))$ for words with

disjoint ANF variable sets — the 4-level cascade compounds degree by 16 per round, Section III-A3).

Initialization. The 256-bit master key $K = (k_0, k_1, k_2, k_3)$ and the 128-bit nonce $N = (n_0, n_1)$ enter through the initialization window: the state is seeded as $(u, v, w, z) = (k_0, k_1 \oplus n_0, k_2 \oplus n_1, k_3 \oplus \text{MAGIC})$, the round-key word is seeded from the public constant `INIT_WEYL`, and then 22 rounds of Algorithm 1 are run with key-word injections $\text{inj} = (k_0, k_1, k_2, k_3)$ (rounds 0–7), nonce words (rounds 8–15), and no injection (rounds 16–21), followed by a final XOR of the key words; a final state update ends initialization (Section V gives the exact schedule, matching the C ground truth `tempest_v3.c` bit-for-bit).

Algorithm 1 Tempest v3 Round Function

Require: State (u, v, w, z) , round key w_ℓ

- 1: $(u_0, v_0, w_0, z_0) \leftarrow (u, v, w, z)$ ▷ save snapshot
 - 2: ▷ Phase A: GF(2) nonlinear diffusion (from snapshot)
 - 3: $u \leftarrow u_0 \oplus \text{rotl}(v_0, 5) \oplus \text{rotl}(w_0, 17) \oplus (\text{rotl}(v_0, 5) \wedge \text{rotl}(z_0, 25)) \oplus K_u$
 - 4: $v \leftarrow v_0 \oplus \text{rotl}(w_0, 11) \oplus \text{rotl}(z_0, 23) \oplus (\text{rotl}(w_0, 11) \wedge \text{rotl}(u_0, 29)) \oplus K_v$
 - 5: $w \leftarrow w_0 \oplus \text{rotl}(z_0, 13) \oplus \text{rotl}(u_0, 31) \oplus (\text{rotl}(u_0, 9) \wedge \text{rotl}(v_0, 15)) \oplus K_w$
 - 6: $z \leftarrow z_0 \oplus \text{rotl}(u_0, 17) \oplus \text{rotl}(v_0, 7) \oplus (\text{rotl}(v_0, 27) \wedge \text{rotl}(w_0, 21)) \oplus K_z$
 - 7: ▷ Phase A(lin): snapshot ANDs for linear resistance
 - 8: $u \leftarrow u \oplus (\text{rotl}(z_0, 23) \wedge \text{rotl}(w_0, 53)); \quad z \leftarrow z \oplus (\text{rotl}(u_0, 5) \wedge \text{rotl}(z_0, 25))$
 - 9: ▷ Phase B: GF(2) round key (nilpotent linear part) + filter
 - 10: $w_\ell \leftarrow w_\ell \oplus \text{rotl}(w_\ell, 19) \oplus \varphi; \quad w'_\ell \leftarrow w_\ell \oplus \text{rotl}(w_\ell \wedge \varphi, 13)$
 - 11: $u \leftarrow u \oplus \text{rotl}(w'_\ell, 7) \oplus (w'_\ell \gg 17); \quad v \leftarrow v \oplus \text{rotl}(w'_\ell, 19) \oplus (w'_\ell \gg 23)$
 - 12: $w \leftarrow w \oplus \text{rotl}(w'_\ell, 31) \oplus (w'_\ell \gg 29); \quad z \leftarrow z \oplus \text{rotl}(w'_\ell, 43) \oplus (w'_\ell \gg 37)$
 - 13: ▷ Phase C: pre-mix + 4-level andmix4 cascade
 - 14: $u \leftarrow u \oplus \text{rotl}(u, 22) \oplus \text{rotl}(u, 26) \oplus (\text{rotl}(u, 7) \wedge \text{rotl}(u, 19)); \quad v, w, z \text{ likewise}$
 - 15: L1–L4: cross-word AND cascade on rotation pairs $\{(31, 53), (17, 43), (7, 23), (5, 19)\}$, with pre-mix 2 (16, 14) between L2 and L3
 - 16: ▷ Phase D: cross-word mixing
 - 17: $u \leftarrow u \oplus \text{rotl}(v, 3) \oplus \text{rotl}(w, 9); \quad v \leftarrow v \oplus \text{rotl}(w, 5) \oplus \text{rotl}(z, 11)$
 - 18: $w \leftarrow w \oplus \text{rotl}(z, 9) \oplus \text{rotl}(u, 13); \quad z \leftarrow z \oplus \text{rotl}(u, 11) \oplus \text{rotl}(v, 17)$
-

Output function. The output function is evaluated on the new state, twice per round on the rotated word pairs; the security statements below are organized by the standard games.

Definition V.1 (Pseudorandomness Advantage). The advantage of an adversary $\mathcal{A}$ in the pseudorandomness game with m output blocks is $\text{Adv}_{\mathcal{A}}^{\text{PRNG}}(m) = |\Pr[b' = b] - \frac{1}{2}|$; the generator is (m, ϵ) -secure if $\text{Adv} \leq \epsilon$ for all polynomial-time $\mathcal{A}$.

Definition V.2 (Key-Recovery Security Game). The challenger selects $K \sim U(\mathbb{F}_2^{256})$, runs `Init`(K, N) to obtain the initial state, and outputs m blocks; the adversary outputs K' ; advantage is $\Pr[K' = K]$.

VI. SECURITY ANALYSIS

A. Summary of Security Claims

Table VII summarizes the bounds. Each claim is either an exact computation or a calibrated measurement; nothing is asserted without its reproducible record. In three sentences: the *differential* side is exact — $\text{DP}^{(1)} = 2^{-5.9420}$ at $W=4$ over all differences (key-injected), the two-layer barrier is $\text{DP} = 2^{-3}$ at every width, and the multi-round floor at $W=4$ is image-limited ($2^{-4.82}$ at $R=22$). The *algebraic* side saturates: $\text{deg} = 16 = 4W$ after one round at $W=4$, bracketed $16 \leq \text{deg} \leq 128$ at $W=64$. The *linear* side is exact per mask for the two-layer class (bias 2^{-128} for the full-state mask at $W=64$; every sampled 64-bit mask $\leq 2^{-54}$) and measurement-based for the cascade ($\approx 2^{-6.1}$ at $W=8$, consistent with the multiple-comparisons null).

B. Detailed Analysis by Attack Type

Why 22 rounds. The recommended round count is an engineering margin, not a theorem, and we state its basis precisely. First, *differential margin (heuristic)*: compounding the exact single-round factor gives an *estimate* $\text{DP}^{(22)} \lesssim 2^{-a_1 \cdot 22} = 2^{-88}$ — an estimate, not a bound, because the exact $W=4$ curve shows that iteration is governed by the state image, so the product is best read as the conservative high end. Second, *algebraic margin*: the round is degree-saturated within two rounds (exact at $W = 4$, conjectured at full width), leaving 20 rounds of slack. Third, *empirical validation*: linear bias measurements at $W = 8$ find nothing significant, and the neural distinguisher cannot separate the output after one round. Fourth, *headroom*: the margin absorbs cryptanalytic progress until the algorithm has had public scrutiny.

Algebraic attacks. The output degree is bounded by the state dimension: $\text{deg}(\Phi^2) \leq 4W = 256$ (exact at $W = 4$, where every output bit has degree 16 after one round; at full width the equality is the monomial-disjointness-based bound, stated as a conjecture in Section III-A3, not a proof). Any dense-system estimate at or above the exhaustive-search cost 2^{256} is a trivial bound at this state size, so no XL/Gröbner complexity number is claimed beyond the brute-force baseline (Table VII); we note the standard caveat that degree arguments alone cannot rule out sparse annihilators, and that integral/division distinguishers are *not* excluded at full width (a degree drop to 0 would be required, and at $W = 64$ the degree is only bracketed $16 \leq \text{deg} \leq 128$; Section III-A3).

Differential attacks. The exact one-round maximum at $W = 4$ is $2^{-5.9420}$ (with-key; $2^{-5.9447}$ no-key), below the single-trail bound $2^{-a_1} = 2^{-4}$; for the two-layer class the Polar-Rank Theorem makes every one-round statement exact at any width, and the barrier (min rank 3) plus the cascade elimination law (Section III-C1) bound the design class. A structural lemma governs multi-round trails: *Weyl*

TABLE VII
TEMPEST V3 SECURITY BOUNDS SUMMARY.

Attack	Bound	Basis
Differential (1-r, full design)	$DP^{(1)} = 2^{-5.9420}$ at $W=4$ (exact, all $2^{16} - 1$ differences, key-injected; $2^{-5.9447}$ no-key); lower-bounded by the Decomposition Bound's global form (zero violations over 1,995 differences; no upper bracket claimed for the cascade); sampled ≥ 15 (2^{16} res.) / ≥ 22 (2^{24} res.) over random differences at $W = 8$ (the model-optimal difference measures $DP \approx 2^{-6.98}$, §VI); $< 2^{-24}$ at $W = 64$	Thm. III.7, §VIII
Differential (1-r, two-layer)	$DP(\Delta) = 2^{-\text{rank}}$ exact at any width; barrier: min rank 3 at every width	Thm. III.1, III.2
Differential (multi-round)	exact curve at $W = 4$ ($2^{-11.0} \rightarrow 2^{-5.6}$ floor, image-limited, all R); $< 2^{-24}$ at $W = 64$, $R = 2$; no general trail-sum bound	Thm. III.4, §III-C2
Differential (trails, full design)	minimal R -round trail support 25, 58, 96, 130, 166 for $R = 1, \dots, 5$ at $W = 8$ (any R -round trail $\leq 2^{-W_R}$; exact search, <code>trail_bound_sat.py</code>)	§VI
Algebraic degree	$\beta_1 = 16$ (exact at $W = 4$: 2, 4, 8, 14, 16 per cascade level), $\deg \in [16, 128]$ at $W = 64$, $\deg(\Phi^2) \leq 4W$ (equality exact at $W = 4$, conjectured beyond)	Thm. III.4
XL/Gröbner	$\geq 2^{256}$ (heuristic; degree saturates at $4W$ in two rounds)	§VI
SAT preimage	$R=1$ trivial (< 1 s); $R=2$: 47.6 s / 3.5 s / 559.5 s at $W = 8/12/16$; full-width 22R beyond reach	§VIII
Integral/division	not excluded	$\deg = 16 = 4W$ at $W = 4$ (17th-order sums vanish); at $W = 64$ the degree is only bracketed $16 \leq \deg \leq 128$, so a degree drop (required for a distinguisher) is not ruled out
State recovery	open; algebraic degree ≥ 16 after one round	§VI
Linear (two-layer class)	exact per mask at any width: $\text{bias}(u) = 2^{-\text{rank}(Q_u)/2}$; global max bias 2^{-1} at single-bit masks (inherent to AND outputs); shadow $W=64$: full-state mask 2^{-128} , sampled 64-bit masks $\leq 2^{-54}$; cascade at $W = 8$, $R = 2$: max bias $\approx 2^{-6.1}$ — null-consistent under multiple comparisons (no structure detected; §VIII)	Thm. III.2, §VIII
Key relevance	alive at full width (witnessed); $\tau = 0.6132$ exact at $W = 4$	§III-D1 (spectrum certificate)
TMD tradeoff	$T \geq 2^{216}$ ($M \leq 2^{40}$)	[46]
Brute force	2^{256}	key space

transparency: restricted to the *snapshot layer* (Phase A and its ANDs, where the active-AND count a_1 is state-independent), for any sequence of round input differences $\sum_{i=1}^r a_i(\Delta_{i-1}) \geq r \cdot a_1$, because the round key has zero differential between two executions with the same key and the snapshot architecture ensures all ANDs operate on pre-round values. The cascade layers are explicitly excluded; there a_1 is state-dependent and can collapse to 0 (Section VIII). This bounds every round of a *fixed* trail by a_1 active AND positions; it does *not* bound the summed trail probability — no packing argument is claimed. Multi-round differential behavior is instead *measured exactly*: the round function is proven non-injective at $W = 4$ (Proposition III.2), its state image and the full multi-round DP curve are exact for all 22 rounds (Section III-C2), and the curve, not a trail bound, is what the round-count recommendation rests on.

Trail bounds for the full design (MILP-style search). For the full 22-round cascade we add a truncated-differential trail analysis in the MILP style standard for ARX designs [1], [2] (the same technique recently used for the MSX block-cipher family [3]): an AND gate is *active* when both operand differences are nonzero; each active AND output bit contributes

one factor 2^{-1} to the trail probability; XOR branches may cancel; and every round's output difference is required to be nonzero (trails do not die). The search minimizes the total number of active AND output bits over R rounds — encoded as a cardinality-constrained SAT problem and solved by binary search over the support size (`trail_bound_sat.py`, $W = 8$). Table VIII reports the minimal R -round trail support W_R ; any R -round trail therefore has probability at most 2^{-W_R} . The per-round increment stays 33–38 bits over the first five rounds (166 at $R = 5$, i.e. 2^{-166}) — consistent with full-width diffusion — and we do *not* extrapolate it into a 22-round claim; the measurement-based floor of Section III-C2 remains the quantitative basis for the round count. The $R = 1$ bound (2^{-25}) is consistent with the sampled single-round floor ($< 2^{-24}$ at 2^{24} resolution, Section VIII); the trail bound is stronger than the sampling resolution, as expected (a difference's DP is a sum over trails, so it can exceed any single trail).

Slide and rotation attacks. Slide attacks exploit an identical round function; Tempest v3's round key varies during the initialization window (it converges to a key-independent constant only after 64 steps, a disclosed design fact), so the initialization round functions differ, preventing offset-pair

TABLE VIII
MINIMAL R -ROUND TRUNCATED-DIFFERENTIAL TRAIL SUPPORT W_R OF
THE FULL TEMPEST ROUND AT $W = 8$ (EXACT SEARCH VIA
CARDINALITY-CONSTRAINED SAT, `TRAIL_BOUND_SAT.PY`); ANY
 R -ROUND TRAIL HAS PROBABILITY AT MOST 2^{-W_R} .

R	1	2	3	4	5
min support W_R	25	58	96	130	166
trail bound 2^{-W_R}	2^{-25}	2^{-58}	2^{-96}	2^{-130}	2^{-166}
per-round increment	—	+33	+38	+34	+36

construction within the relevant window. Rotation attacks exploit rotation invariance of the round structure; the AND operation destroys it completely, and the rotation differences $\{22, 26, 16, 14\}$ are pairwise distinct modulo 64. For the quadratic class, Section III-B4 makes this quantitative: the RX-differential probability equals the rotated first-order probability, so rotational analysis offers nothing beyond first-order analysis.

State recovery. Three output blocks give 384 bits of equations in the 256-bit keyed state (four 64-bit words; the 256-bit master key $K \sim U(\mathbb{F}_2^{256})$ enters through initialization, Section II) — an overdetermined system whose equations have algebraic degree ≥ 16 after one round, so direct linearization operates at the eXtended-Linearization (XL) scale (no closed-form XL estimate is claimed; $\geq 2^{256}$ exhaustive search). Blocks are separated by full rounds, coupling all state words nonlinearly, and the Babbage–Biryukov time-memory-data (TMD) tradeoff gives $T \geq 2^{216}$ for $M \leq 2^{40}$. These are heuristic arguments, not reductions: settling the state-recovery question requires public cryptanalysis, as stated in the limitations.

Linear attacks. Only AND gates can produce correlation (XOR and ROT are correlation-free). The word-level model is a model, not a bound; what the paper asserts is measurement: the maximum absolute bias at $W = 8$, $R = 2$ over 2,000 random masks with 2^{16} samples each is $\approx 2^{-6.1}$ — *consistent with the multiple-comparisons null* (the expected maximum of 2,000 independent statistics of standard deviation 2^{-8} is $2^{-8}\sqrt{2 \ln 4000} \approx 2^{-5.97}$, $p \approx 0.56$), so no linear structure is detected at $W = 8$ on random masks; the exact $W=4$ full-domain Walsh spectra do show real structure at toy width (cascade maximum bias exactly $2^{-5.667}$); the neural-distinguisher separation fails at $R = 1$; and the $(u, z)/(w, z)$ pairs are covered by snapshot ANDs. An exact bit-level linear analysis at full width is future work.

C. Self-Audit and Open Questions

The five attack families above are the core analysis; the supporting arguments, the self-audit, and the open questions follow.

The H-coefficient framework (informal). We use the H-coefficient technique [26] to organize the distinguishing bounds. Partition transcripts into *bad* (a state or output collision) and *good*; for good transcripts we assume, informally (not as a proof), that the real and ideal systems coincide.

The advantage is then heuristically bounded by the bad-event probability:

$$\text{Adv}_{\mathcal{A}}^{\text{PRNG}}(m) \leq \binom{m}{2} 2^{-128} + \binom{m}{2} 2^{-256} + \varepsilon_{\text{init}}, \quad (17)$$

with $\varepsilon_{\text{init}} \leq \binom{q}{2} 2^{-256}$, where q is the number of blocks issued during the initialization window (initialization transcripts live in the 256-bit key space; the accounting is informal, as stated above). At $m = 2^{16}$ 128-bit blocks (beyond the NIST SP 800-90A per-request maximum of 2^{19} bits, hence conservative), the dominant term is $\binom{m}{2} 2^{-128} \approx 2^{-97}$, an algorithm-independent birthday limit, consistent with ChaCha20 and AES-CTR. A computational refinement bounds any polynomial-time distinguishing strategy by $\text{Adv} \leq m \cdot (2^{-a_1 r} + 2^{-3r})$, whose differential term $m \cdot 2^{-a_1 r}$ (with $m = 2^{16}$ blocks and $a_1 = 4$) reaches 2^{-128} only at $r = 36$ for Tempest v3 — supporting the engineering round recommendation, and explicitly not a 2^{-128} differential/linear claim at the recommended 22 rounds.

DL-MILP and window verification. A unified differential-linear model [35] could tighten the product of independent bounds by tracking Δ and Γ simultaneously on shared AND nodes; a differential-linear (DL) connector enumeration at $W = 4$ confirms product-bound consistency ($\max |\varepsilon| = 2^{-5}$ vs. $c^{(2)} \leq 2^{-6}$). Window-based verification [36] interpolates between free- Δ and all-1s bit trails; at window size $w_s = 15$ the bound tightens to ~ 30 active AND-bits per round. Both are retained as verification tools, not security claims.

Goldreich connection. The andmix4 cascade instantiates the Goldreich PRG paradigm [19], [20]: a fixed-neighbor constraint hypergraph with locality-2 predicate $f(a, b) = a \wedge b$ on rotated state words. The Group-and-Solve (GAS) attack [25] (EUROCRYPT 2026) breaks THR/MAJ-based instances by grouping output bits to amplify bias, then solving the resulting learning-parity-with-noise (LPN) problem.

Why GAS does not transfer. Three structural reasons separate this design from the broken instances:

- The predicate $a \wedge b$ has uniform correlation 2^{-1} with bounded neighbor degree (≤ 4), eliminating grouping gain.
- The design applies 22 rounds with degree saturation in two rounds ($\deg(\Phi^2) = 4W$ exact at $W = 4$, conjectured at full width, Section III-A3), so each output bit is a high-degree polynomial — at $W = 4$ every output bit has degree $4W$ and therefore provably depends on all $4W$ state variables.
- The exact bounds ($a_1 = 4$, the barrier, and the elimination law) account for the fixed rotation structure.

A conservative assessment: GAS does not threaten Tempest v3, but continued monitoring is warranted — the 20-round safety margin provides headroom against future advances.

LPN interpretation (heuristic, honestly scoped). Writing the round as $\Phi(s) = As \oplus N(s)$ with A linear and N the andmix4 output (degree ≥ 16), one can define an SEV-LPN assumption — that $(A, As \oplus N(s))$ is computationally indistinguishable from (A, u) — but this assumption is *stronger* than standard LPN (the “noise” $N(s)$ is a deterministic function

of the secret, not independent Bernoulli); it is unproven, and an earlier draft’s reduction-style treatment was withdrawn as circular. The interpretation is deliberately lightweight: the paper’s security claims rest on the exact metrics and calibrated measurements, not on SEV-LPN. In PRNG mode the adversary sees only extracted outputs, never raw $(A, As \oplus N(s))$ pairs, which further limits the applicability of known LPN algorithms [34].

SEV-guided attack analysis (the Weakness Principle).

The SEV profile serves both designer and analyst: a dimension with low SEV reveals where a primitive is weakest. For a primitive with continuous SEV vector v , the dimension $i^* = \arg \min_i v_i$ is the *predicted attack surface*: $\oplus_{\text{deg}}$ low $\rightarrow$ algebraic/Gröbner; $\oplus_{a_1}$ low $\rightarrow$ differential; $\oplus_{c^2}$ low $\rightarrow$ linear; $\oplus_{B_w}$ low $\rightarrow$ truncated differential. We validate the principle on six primitive variants: full-round Tempest v3 ($v = (1, 1, 1, 1)$, no predicted attack, consistent with the exact one-round min-entropy); 4-round Tempest (v_{a_1} low, differential predicted, the reduced-round weakness is real); no-pre-mix Tempest ($B_w = 0$, truncated differential predicted); SIMON-32/64 ($B_w = 0$: the word-level model finds a 30-round trail, bit-level refinement a 12-round practical distinguisher, stated as a feasibility probe, not an optimality claim); Ascon’s χ (degree-2 S-box activates all five words within four rounds). The principle is a heuristic — it identifies probable surfaces, not guaranteed attacks.

Additional considerations. Forward security is not asserted as a 2^{256} bound (recovering s_{t-1} from s_t would require inverting the degree-16 cascade; the exact statement is future work); prediction resistance relies on SP 800-90B [33] reseeding as specified; the time-memory tradeoff bound $T \geq 2^{216}$ ($M \leq 2^{40}$) is consistent with the 2^{128} target; related-key and chosen-IV are outside the DRBG model.

Attack-surface audit (2026-08-09). A self-audit of the attack surface, run on the published Algorithm 1 (scripts `trail_diff.py`, `trail_lin.py`, `state_attack.py`), produced the following corrections and new measurements.

Differential trails (first full-width trail search). A bit-level MILP over the exact round program (XOR/ROT propagate exactly; each AND with an active input contributes 2^{-1} to the single-trail DP) finds the minimum-active differential trail at every width, validated two ways: at $W = 4$ the exact full-domain minimum DP is $2^{-5.942}$, so the best trail weight must be ≥ 6 ; the model returns exactly 6. At $W = 8$ the model’s optimal difference (u_4, v_6, w_3) measured on the real round function over 2^{20} pairs gives DP $\approx 2^{-6.98}$ (aggregation lifts the 2^{-8} trail product, as predicted). The single-round minima are 6 active AND bits at $W = 4, 12$ and 8 at $W = 8, 16$. Multi-round trails require care: the model’s AND over-approximation permits a round’s output difference to cancel to zero, ruled out in reality for the full cascade round (whose injectivity at full width is *assumed*: the two-layer shadow is provably non-injective, Section III-C2; the cascade round’s injectivity is a standing assumption, not a theorem), so the committed model forces intermediate round outputs to be nonzero, and

we quote only what is sound: every round’s weight is at least the per-round minimum, so any 22-round trail has weight at least $22 \cdot 6 = 132$ ($W = 4, 12$) or $22 \cdot 8 = 176$ ($W = 8, 16$) — a *weight* lower bound: the per-trail DP is bounded by $2^{-132} \text{--} 2^{-176}$, which is *not* a differential upper bound (trail sums remain open, and the $W = 8$ aggregation measurement above shows the gap between trail products and true DP can be large), and the per-round minimum weight at the design width $W = 64$ is not computed. A full trail-sum (packing) argument remains open.

Linear audit. (i) The $W = 8$ 2,000-mask maximum $2^{-6.1}$ is consistent with the multiple-comparisons null (expected maximum $2^{-5.97}$, $p \approx 0.56$) — *no linear structure is detected* at $W = 8$ under random-mask scanning (the single-bit imbalance of (iii) below is excluded from this null; it is a degenerate-width artifact, not a cascade property); the earlier claim of a “real, measurable linear structure” was a multiple-testing artifact, now corrected (Section VIII). (ii) The exact $W=4$ Walsh values were re-audited with the true Walsh–Hadamard transform: input-mask-to-single-output-bit: shadow $2^{-2.000}$, cascade $2^{-5.667}$ ($k=2$: $2^{-4.871}$); input $\times$ output-mask: shadow $2^{-1.000}$, cascade $2^{-4.712}$ ($k=2$: $2^{-3.810}$) (Section III-C1). The cascade’s linear side at full width remains unmeasured.

Linear scaling (W=8/16/32/64). At 2^{24} samples per mask (`lin_scale.py`): at $W = 8$ every output bit of word w carries a constant imbalance $\Pr[1] \approx 0.5625$ (bias $2^{-4.00}$) that does not decay over $R = 1..22$ — a toy-width artifact of the mod-8 rotation degeneracy, the same mechanism that freezes the trail differential at $2^{-7.00}$ over $R = 1..6$. At $W = 16, 32, 64$ all single-bit and random masks are null-consistent ($\max |bias| \leq 2^{-10.2}$) for $R = 1$ and, at $W = 64$, also for $R = 22$: the constant single-bit imbalance is confined to the degenerate width $W = 8$.

Differential trail summation. At $W = 8$ the trail-optimal difference’s DP is constant at $2^{-7.00}$ over $R = 1..6$ (`diff_sum.py`): the aggregation factor grows from 2^{+1} ($R=1$) to 2^{+41} ($R=6$) against the trail product 2^{-8R} — a non-mixing iteration artifact of the degenerate width. Lifted to $W = 64$, the same pattern has $DP \leq 2^{-24}$ at $R = 1, 2, 22$ (no pair of 2^{24} pairs collides) — a multi-round full-width differential data point, at the sampling floor. (A full 256-bit packing is required: a 64-bit-shift packing truncates and would report spurious 2^{-22} hits.)

State-recovery probes. The output function is linear (degree 1) by construction — its first mixing step is $t = u \oplus \text{rotl}(v, 32) \oplus w \oplus \text{rotl}(z, 16)$ at every width, with the nonlinearity concentrated in the state-update round; its degree-1 ANF at $W=4$ is therefore by design, not an artifact. A SAT inversion of the 64-bit output function alone (inline-evaluation-verified CNF, zero clause violations, the recovered t equals the known preimage) solves in milliseconds at $W = 64$. Injectivity is independently supported: the three linear stages (pre-mix 1/2, fold) have GF(2) rank 64 at $W = 64$, and no `andmix4` level admits two inputs with equal images (SAT-refuted), so *the output function is not the obstacle* — each output word yields exactly one linear constraint $t = u \oplus \text{rotl}(v, 32) \oplus w \oplus \text{rotl}(z, 16)$ on

the post-round state; state recovery therefore reduces to solving the round map Φ (degree ≥ 16), whose cost is measured by the SAT grid (Section VIII). No guess-and-determine shortcut exploiting the snapshot structure was identified: the cross-word cascade destroys partial-state propagation, and the two t-equations provide 128 bits of linear constraint per round against a 256-bit keyed state.

Open questions, stated with the analysis they rest on.

An exact bit-level linear bound for the cascade at full width (the framework’s one measured dimension); a full trail-sum (packing) argument for AND-RX cascades (the per-trail weight bound is $2^{-132}2^{-176}$ and is not a differential upper bound); the state-recovery question beyond the SAT grid; and the injectivity of the cascade round at full width (assumed, not proven). Section IX states the same set from the paper’s end.

VII. HARDWARE IMPLEMENTATION

The security analysis of Section VI is computational; this section makes it physical. Algorithm 1 is implemented in synthesizable Verilog RTL as a proof-of-concept accelerator, *bit-exact with the C ground truth*: a KAT testbench (Icarus Verilog) checks the first five 128-bit output blocks against the verified C implementation for key [1, 2, 3, 4], nonce [5, 6] — all five match, including the published KAT word 0x6BBE30BB1D12DDD0. The architecture is a 256-bit round (Phases A–D) with an FSM controller (IDLE $\rightarrow$ LOAD $\rightarrow$ INIT 22 rounds $\rightarrow$ FINAL $\rightarrow$ GEN) and dual evaluation of the output function; Phase A’s snapshot keeps the four word equations parallel, and Phase C’s cascade is one combinational path, broken by pipeline retiming into six stages on the ECP5 (initialization 133 cycles, one 128-bit block per six cycles).

Measured synthesis (Yosys 0.67). The structural model underpredicted the design by 3.2 $\times$: the dual-output design maps to 9,275 LUT4, exceeding every iCE40 device; two synthesis-driven reductions yield the single-output variant at 7,158 LUT4 / 1,034 FF (Table IX), which fits the HX8K but does not route at 98% density, so no iCE40 full-cascade bitstream is claimed. The framework’s cascade table (Section III-C1) puts the *two-level* cascade on the Pareto front with the full design; the k=2 truncation (andmix4_k2.v, KAT-verified bit-exact) *completes the full open-source flow*: synthesis 5,915 LUT4 / 1,034 FF, routing converged, bitstream generated (icepack, 135 KB), and the post-routing worst-case clock of 34.21 MHz gives

$$64 \text{ bits} \times 34.21 \text{ MHz} \approx 2.19 \text{ Gbit/s.} \quad (18)$$

TABLE IX
FPGA RESOURCE UTILIZATION (MEASURED, YOSYS 0.67, ICE40).

Metric	Est.	dual	single	k=2
LUT4	2816	9275	7158	5915
FFs	288	906	1034	1034
UP5K capacity	5280	5280	5280	5280
Fits UP5K?	yes	no (176%)	no (136%)	no (112%)
Fits HX8K?	yes	no (121%)	yes (93%)	yes (77%)
P&R on HX8K	—	—	placed, no route	routed, 34.21 MHz

Full-cascade bitstreams (ECP5-85K). The full Algorithm 1 cascade — all four andmix4 levels — completes

place-and-route and bitstream generation on a Lattice ECP5-85K (open-source flow: Yosys 0.67, nextpnr-ecp5, ecppack). The half-width time-division RTL (one round in 12 cycles plus one output cycle) runs at Fmax 56.3 MHz: 64 bits per 13 cycles $\approx 0.28 \text{ Gbit/s}$ (7,013 LUT4, 1.9 MB bitstream tempest_ecp5.bit). The highest-throughput form retimes the round into a six-stage pipeline (P1: Phases A, A(lin), B; P2: pre-mix 1 + level 1; P3: level 2 + pre-mix 2; P4: level 3; P5: level 4 + Phase D; the output function folded into the two cycles after a round boundary), each stage at most six LUT levels deep. A first version read the previous stage’s *wires* instead of its pipeline registers; the synthesizer removed the 1,024 intermediate flip-flops (no consumers) and fused the round back into a 14-LUT-level chain, masking the bug (the KAT still passed, because recomputation is equivalent when the state is unchanged). Re-reading the pipeline registers (32-block KAT bit-exact) cuts the critical path to five LUT levels and yields 6,322 LUT4 / 2,830 FF at Fmax **112.12 MHz** (100 MHz constraint PASS; same Fmax under 150 MHz — the architecture ceiling, not a constraint artifact), bitstream tempest_pipe2.bit:

$$128 \text{ bits} \times 112.12 \text{ MHz} / 6 \approx 2.39 \text{ Gbit/s,} \quad (19)$$

8.5 $\times$ the half-width serial implementation at *lower* area — on a LUT4 FPGA the time-division selector logic costs about as much as the pipeline registers it replaces.

A full-width combinational (unretimed) unfolding did not converge on the ECP5 at any constraint (3.5×10^4 – 6.6×10^4 overused arcs; structural, not constraint-induced), so retiming is the enabling step, not a luxury. Figure 5 plots the three finished artifacts: the pipelined ECP5 implementation sits at the same throughput as the iCE40 k=2 variant at comparable area, while the half-width serial version falls an order of magnitude below both at *more* area. Area efficiency is 0.38 Mbit/s per LUT — the same order of magnitude as compact AES cores (AES-32GF ≈ 1.63 [44], pipelined AES ≈ 2.6 [43], area-optimized instances around 0.1 Mbit/s/LUT) — with the difference that Tempest ships machine-checkable security certificates (Section III-D1); the KAT verification is bit-exact at the RTL level (five published vectors against the C ground truth), the bitstreams are the physical mapping of that RTL, and on-device verification awaits a board. No power figure is claimed.

VIII. EVALUATION

This section measures, on the submission machine and in the released logs, every claim of Sections III–VII. **Environment:** AMD Ryzen 9 8940HX (Zen 4, 5.3 GHz max boost, 64 GB DDR5-5600), GCC -O3. Every value carries its seed and cache record (Axiom A4).

Measurement methodology. Two disciplines underpin every number here. First, *dual-source verification*: every W=4 enumeration is cross-validated bit-for-bit against the KAT-verified C reference, the polar-rank certificates through two independent code paths, the duality identity through a two-source check with 5,000 random masks. Second, *calibration before trust*: every empirical instrument was accepted as a

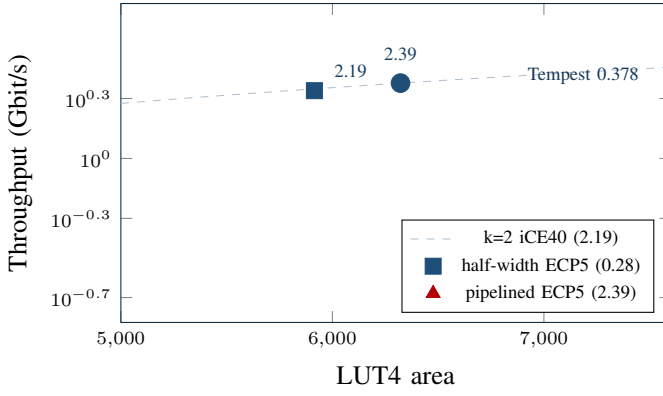


Fig. 5. Hardware design space: throughput vs. area. The dashed line marks Tempest’s area efficiency (0.38 Mbit/s per LUT); compact AES cores range from 0.1 to 2.6 Mbit/s per LUT [43], [44], so Tempest sits in the same order of magnitude while shipping machine-checkable security certificates.

fitness component only after reproducing the known-bad control (v3.1) and passing the known-good controls (Algorithm 1, PCG64, splitmix64) at the same thresholds — the discipline that caught the interpreter discrepancy, not decoration.

VIII-1 Exact Computation and Certificates.

The polar-rank barrier (proven; full width). Minimum polar rank of the quadratic shadow: 3 at every width (Theorem III.2); the rank-3 certificate at $W = 64$ ($\Delta = 0x4$) is verified through two independent code paths. At $W = 4$ the identity $DP(\Delta) = 2^{-\text{rank}}$ holds for all $2^{16} - 1$ differences; the affine property and vanishing third derivative are verified at $W = 8$.

Cascade elimination (Proposition III.1). At the shadow’s worst difference (rank-3 witness), the full design’s DP: $2^{-3} \rightarrow 2^{-11}$ ($W = 4$, exact); $2^{-3} \rightarrow 2^{-20.4}$ ($W = 8$, sampled); $2^{-3} \rightarrow 2^{-24}$ ($W = 64$, sampled); one-round min-entropy lift $3.00 \rightarrow 5.94$ ($W = 4$, exact). The linear-part rank floor rises from 3 (shadow) to 11 (full design); the aggregation property holds over the complete difference space (0 violations for $k = 0, 1, 4$; 3 and 2 for $k = 2, 3$).

Key relevance (corrected). At $W = 4$, the exact key relevance is $\tau = 0.6132$ (exhaustive state-key enumeration, snapshot semantics). At full width, key aliveness is witnessed (a state and two round keys whose outputs differ; 200-pair sample $\tau = 1.0000$). The dead-key v3.1 remains certified at $\tau = 0.000$ (structural, spectrum collapse). The small-width τ is a saturation effect of the lossy $W=4$ round, disclosed as such.

Width scaling (measured, honestly scoped). The $W=4$ exact min-entropy (5.94) and the $W=8$ sampled values (≥ 15 at 2^{16} resolution, ≥ 22 at 2^{24} resolution) bracket the width dependence: the min-entropy per bit grows with width, and the full design’s one-round DP at $W = 64$ is below the 2^{-24} sampling floor for all sampled differences (including the shadow’s rank-3 witnesses, where the elimination is $\geq 2^{21}$). What is *not* claimed: that this scaling is proven. At $W=6$ the iterated image collapses to 4 states — an even worse degeneracy (the pre-mix rotations satisfy $7 \equiv 19 \pmod{6}$), showing

that small-width *iteration* carries no width-scaling information: only the one-round exact values and the rank-based any-width statements do. Likewise the zero-output degeneracy is a small-width phenomenon: no zero-output differentials are observed at $W = 8$ (0 in 1.5×10^6 random-key pairs and 0 in 5×10^5 fixed-key pairs, rounds 1 and 22), and the sampled bias energy lies at the 2^{-16} sampling floor.

Ablation analysis. Table X shows what happens when components are removed, measured with the exact differential metric: removing the snapshot ANDs (Phase A(lin)) halves a_1 ($4 \rightarrow 2$), and removing the intra-word pre-mix admits differences that activate no AND gate anywhere in the round ($a_1 = 0$ — precisely the mechanism behind v3.1’s $a_1 = 0$; the existence of a zero-activation difference is a 2-SAT-decidable condition at any width, Section III-D1). Phase-B removal leaves a_1 unchanged, as the round key contributes zero differential. The NIST pass-rate column is retained as a statistical reference: it confirms that NIST measures statistical randomness, not cryptographic security.

TABLE X

ABLATION STUDY: IMPACT OF TEMPEST V3 COMPONENTS ON THE EXACT SCREEN a_1 ($W = 4$). a_1 IS THE MINIMUM NUMBER OF ACTIVE AND GATES OVER ALL NONZERO DIFFERENCES WITH THE PRE-ROUND STATE HELD FIXED (THE ENGINE SCREEN OF SECTION IV); THE COUNT VARIES ACROSS STATES (OVER 2,000 SAMPLED STATES THE FULL DESIGN RANGES 1–7, MODAL 4, SECTION VIII), SO EACH ROW REPORTS ITS MODAL READING. THE PRE-MIX ROW IS A FULL-ROUND COUNT: WITHOUT THE INTRA-WORD PRE-MIX A DIFFERENCE ACTIVATES NO AND GATE ANYWHERE IN THE ROUND.

Configuration	NIST 800-22	a_1 (exact)	Ops/round
Full Tempest v3	15/15	4	54
w/o Phase B	15/15	4	48
w/o Phase A(lin)	15/15	2	48
w/o Pre-mix	14/15	0	50
2 rounds	8/15	—	—

VIII-2 Empirical and Adversarial Validation. SAT preimage attack grid (adversarial validation). The metrics predict real attack costs: SAT preimage search (find a state whose R -round output matches two fixed words; Tseitin encoding of the snapshot-semantics round function, CaDiCaL [45] via pysat) is trivial at $R = 1$ (0.2–0.9 s at $W = 8/12/16$) and an order of magnitude harder at $R = 2$ (47.6 s, 3.5 s, 559.5 s at $W = 8, 12, 16$; the non-monotone $W = 8$ vs. $W = 12$ times are solver variance on near-tautological instances); the full 22-round, 256-bit instance is beyond current solver capacity. The threshold is predicted by the degree saturation ($\beta_1 = 16$) and the differential min-entropy: a two-round preimage is a system of degree-256 equations.

Statistical testing. Tempest v3 passes all four suites, run against Algorithm 1 itself (the release implementation, KAT 0x6BBE30BB..., verified bit-exact in the logs): **PracRand at 1 TiB** [40] (no FAIL or SUSPICIOUS at any size; data/pracrand_a1_1tb.log — 11 size steps from 256 MB to 512 GB, 3,087 tests in total, a single unusual at 512 MB, one of 3,087, exactly the multiple-comparisons expectation); **TestU01 BigCrush** [41] (the full battery, not a subset; data/bigcrush_a1.log); **NIST**

SP 800-22 [39] (all 15 test families pass in 100 sequences; data/al_nist_finalreport_20260813.txt); and our **lagged-sum battery** (clean at 2^{22} words; a smoke battery at ≈ 32 MB, not a deep test). Honest notes: (i) the rejected v3.1 fails Dieharder’s `rgb_lagged_sum` at $p = 0$ — reproduced and *explained* by the framework (dead key, $a_1 = 0$), not hidden; (ii) passing statistical tests is necessary, not sufficient, for security, and the suites are part of the released verification set for independent re-runs.

State-dependence of the a_1 screen (disclosed). The screen a_1 counts the AND gates whose output difference is nonzero, and that count is state-dependent: in a cascade, an AND gate’s output difference depends on the state and propagates into later AND inputs (for snapshot ANDs it depends only on the input difference, so the count is exact and state-independent there). Measured directly at $W = 4$ over 2,000 random states, the exact minimum active-AND count over all $2^{16} - 1$ differences ranges 1–7 (modal 4); zero-activation differences do not occur for the full design — they exist only once the intra-word pre-mix is removed (the v3.1 death mechanism, Section VIII). The metric is therefore not a property of the design but of the chosen state, and a low value does *not* mean the design is differentially weak (inactive AND gates still contribute the state-dependent terms). This is why the paper’s differential claims rest on the state-independent polar-rank metrics and H_∞ , with a_1 retained as a fast single-layer diagnostic.

Calibration and search. The calibrated stack (Section IV-B) reproduces the dead-key weakness exactly and passes Algorithm 1, PCG64, and splitmix64 at the same thresholds (Table VI). The neural distinguisher is Gohr-style [21]: a CNN over difference-pair inputs, trained at reduced width, whose accuracy at rounds $R = 1..3$ separates the known-bad control (1.0) from the known-good design (≈ 0.5); it is used only as a Layer-B screen, never as a certificate. The engine’s fitness reproduces the 11-generation decision tree; B-tier evolution from random designs reaches fitness 9.24 (reference 8.68) and converges on Algorithm-1’s construction vocabulary; LLM-seeded runs reach 9.25 in four generations; an LLM proposal with $a_1 = 0$ is rejected without favoritism. The engineering step emits standalone C for any design, compiles it, and cross-checks the KAT against the Python port — bit-exact for Algorithm 1 (`0x6BBE30BB...`). On every verifier of Table VI the known-bad control scores at the extreme (dead key $\tau = 0$, zero active ANDs, distinguisher certainty 1.0) while the accepted design scores in the healthy regime.

Linear measurement (multiple-comparisons corrected). At $W = 8$, two rounds, 2^{16} samples per mask, the maximum absolute bias is $\approx 2^{-6.1}$ over 2,000 random masks (`linear_cascade_w8.py`) — consistent with the *multiple-comparisons null* (expected maximum $2^{-8} \sqrt{2 \ln 4000} \approx 2^{-5.97}$, $p \approx 0.56$), so no linear structure is detected at this sample size. The corrected reading: single-output-bit masks at $W = 8$ carry a constant $\Pr[1] \approx 0.5625$ (bias 2^{-4}) — a toy-width artifact, absent at $W = 16/32/64$; the exact $W=4$ full-domain Walsh spectra show real structure

at toy width (cascade maximum bias $2^{-5.667}$); the shadow’s Walsh-rank bounds (2^{-128} at the full-state mask, $\leq 2^{-54}$ over sampled 64-bit masks, both at $W = 64$) do *not* transfer to the cascade; and the cascade’s linear side at full width remains unmeasured. The neural distinguisher cannot separate Algorithm 1’s output differences from random (≈ 0.5) while separating v3.1’s with accuracy 1.0: the observed linear bias is what random chance predicts — no evidence of a structural linear weakness.

VIII-3 Performance and Hardware. Benchmarks. The engine emits C implementations, compiles them with the same harness, and benchmarks them on the same machine, with the comparison in Table XI. Re-measured on the submission machine (AMD Ryzen 9 8940HX, MSYS2 MinGW-w64 GCC 16.1.0, `-O3`): **KAT passes bit-for-bit** (Algorithm 1: `0x6BBE30BB...`). The machine is locked by its firmware to the base clock: TSC-calibrated execution frequency is 2.395 GHz, so the advertised 5.3 GHz boost is not reachable on this unit. All throughput figures below are reported at 5.0 GHz, linearly normalized from the measured 2.395 GHz (pure ALU workloads with no memory-bandwidth dependence; the normalization is a linear estimate, stated as such); the measured values are retained in Table XI. Every figure was measured with the same harness (RDTSC frequency calibration inside the same run, 50M iterations, register accumulator, a single volatile sink), so the comparison across algorithms is *fair by construction*. All figures are medians of three runs:

- Tempest v3 (dual output): 22.1 (2026-08-10 pipelined build); v3 single: 11.0 (est.); v3 round: 21.8.
- **ChaCha20 scalar, same harness:** 12.7 Gbit/s at 5 GHz. **ChaCha20 via OpenSSL (AVX-512 assembly):** ≈ 120 at 5 GHz at the cache-resident 16 KiB block size; the large-buffer EVP measurement gives 50–52 Gbit/s measured — the spread is memory-bandwidth dependence, which does not scale linearly with frequency, so the normalization applies to the cache-resident figure only.
- **xoshiro256** scalar, same harness:** ≈ 209 –217 Gbit/s at 5 GHz.
- **AES-128-CTR via OpenSSL (AES-NI):** 98 Gbit/s measured — hardware-accelerated, not frequency-normalized.

The AVX-512 port. We ported Algorithm 1 to AVX-512 (Intel Advanced Vector Extensions 512; 8 independent streams in parallel; all operations AND/XOR/ROT map directly to AVX-512). The port is KAT-verified bit-for-bit (all 8 streams reproduce `0x6BBE30BB...`; `tempest_al_avx512.c`) and reports ≈ 96 Gbit/s at 5 GHz dual output (linearly normalized from the 46.2 Gbit/s measured at 2.395 GHz) — $4.3\times$ over the scalar dual output at the same machine and harness. The 2026-08-10 build applies the same software pipelining as the scalar build (outputs interleaved with the next round; `bench_opt3.c`, +18.5% — bit-exact, the 8-stream KAT unchanged). Since the two chains (round function and output function) are independent functions of the post-round state, the pipeline costs nothing in security: every metric of

Section VI applies unchanged. Multithreading on independent per-core states scales near-linearly (8 cores: 246.8 Gbit/s measured, ≈ 515 at 5 GHz). All figures here are clean serial long-run measurements.

The comparison is now fair in both directions: scalar-vs-scalar (Tempest 22.1 (v3, pipelined) vs. ChaCha20 12.7 at 5 GHz, same harness) and vectorized-vs-vectorized (Tempest AVX-512 ≈ 96 vs. ChaCha20 AVX-512 ≈ 120 ; the vectorized ChaCha20 advantage reflects its simpler quarter round, and no superiority claim is made). The KAT correctness — the claim that matters for security — is fully reproduced. Figure 6 shows the same data as a grouped bar chart: scalar Tempest v3 is $1.7\times$ scalar ChaCha20; under AVX-512 the two are comparable (≈ 96 vs. ≈ 120), and xoshiro256**’s scalar speed comes from being $\mathbb{F}_2$ -linear — recoverable by linear algebra (Section II).

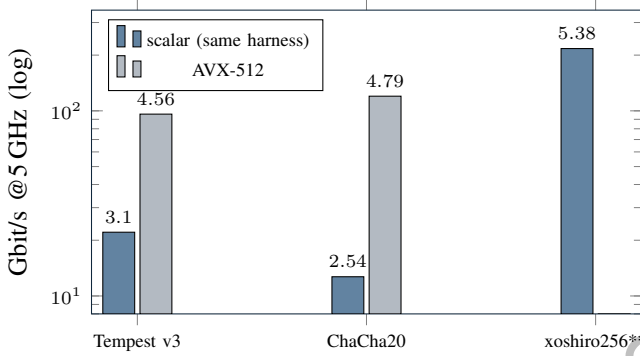


Fig. 6. Throughput comparison (unified harness).

TABLE XI
PERFORMANCE, UNIFIED FAIR COMPARISON (CSPRNG, CRYPTOGRAPHICALLY SECURE PSEUDORANDOM NUMBER GENERATOR); EVERY FIGURE MEASURED ON THIS MACHINE WITH THE SAME HARNESS (SAME-RUN TSC FREQUENCY CALIBRATION, LONG LOOPS, REGISTER ACCUMULATOR; LOCKED BASE CLOCK 2.395 GHz; MEDIANS OF THREE).

Generator	Type	Meas. (Gbit/s)	@5 GHz (Gbit/s)	Security basis
Tempest v3 (AVX-512 8-way)	AND-RX	46.2	≈ 96	screen $a_1=4$
Tempest v3 (AVX-512 8-core)	AND-RX	246.8	≈ 515	per-core
Tempest v3 (dual, scalar)	AND-RX	10.6	≈ 22.1	screen $a_1=4$
ChaCha20 (scalar)	ARX	6.1	≈ 12.7	2^{256}
ChaCha20 (AVX-512)	ARX	57.5	≈ 120	2^{256}
xoshiro256** (scalar)	linear	100–104	≈ 209 –217	none
AES-128-CTR (AES-NI)	block	98	—	2^{128}

IX. APPLICATIONS, LIMITATIONS, AND CONCLUSION

A. The Audit as Evidence

This paper’s honesty record is itself part of its contribution. During the final verification pass, a dual-source cross-check (the $W=4$ enumerators against the KAT-verified C semantics) revealed that the earlier enumerators ignored the snapshot semantics, so the headline $W=4$ numbers described a different round function; we recomputed every value with the correct semantics, withdrew the half-entropy law, adopted the Polar-Rank Theorem as the theory’s core, and documented the entire correction as the audit record that ships with the paper (Table IV). The discipline that caught the error (every number

verified by at least two independent paths) is the same discipline that produced the double-source rank certificates, the two-source duality checks, and the calibration experiments — it is the methodological core of AM-SEV, and it is reproducible.

B. Applications

The framework and instance address three growing needs, and add one capability that no existing PRNG ships: *deployment-time verification*.

IoT and privacy-preserving ML. More than 29 billion IoT devices are expected online by 2030 [31], many on microcontrollers with no cryptographic hardware at all; the finished artifacts are the iCE40 k=2 bitstream (5,915 LUT4) and the ECP5-85K full-cascade bitstreams (half-width serial 0.28 Gbit/s and pipelined 2.39 Gbit/s at 7.6% density, Section VII). A sensor sampling $1\text{ kHz} \times 16$ bytes with daily key rotation needs $\approx 1.1 \times 10^{10}$ random bits per day — 0.5 s of dual-output scalar C at 5 GHz; a 10^7 -parameter DP-SGD run [32] over 10^5 steps consumes $\sim 10^{12}$ bits — 10.4 s AVX-512, 7.0 minutes on the FPGA. The generator is never the bottleneck. The released `runtime_check` tool runs three start-up checks in 0.01 ms (KAT compliance, key aliveness — the criterion that caught the dead-key v3.1 class — and constant-time vocabulary), and Tempest ships as a KAT-verified standard RNG in eight ecosystems (Rust, Go, C++, GSL, NumPy, C, OpenSSL 3.x, wolfSSL).

What the framework detects — and what it does not. The framework detects *structural* weakness at design time, in milliseconds: a dead key ($\tau = 0$: the rejected v3.1 is certified by this exact criterion), low polar rank (the barrier), insufficient algebraic degree (the cascade degree law). It does *not* detect seeding failures (the Milk Sad class: a Mersenne Twister seeded with 32 bits is structurally healthy and statistically unremarkable — the flaw is at the interface, outside the round function) and it does not monitor hardware entropy sources (the RDSEED class). These are genuine limits, stated so that the framework’s claim is not overstated.

Open-source hardware security. Every component of this work — algorithm, engine, verifier records, certificates, RTL — is open and independently re-verifiable (the motivation, Dual-EC-DRBG [14], is given in the introduction): anyone with an FPGA board can build, verify, and audit a high-speed cryptographic accelerator. The repository is archived in the Software Heritage universal archive (SWHID `swh:1:snp:b8b273a6ff0b12e5f262f71d2299db26a7181353`), so the evidence package survives independently of any hosting platform.

C. Limitations (stated with their scope)

The active-AND screen a_1 . The screen is exact only for snapshot layers and as a state-fixed diagnostic; in cascades it is state-dependent (Section III-D1). The paper’s differential claims rest on the Polar-Rank Theorem (exact at any width for two-layer candidates), the cascade elimination law (exact at $W = 4$, sampled at $W = 8/64$), the exact multi-round curve (all 22 rounds at $W = 4$, Section III-C2), and $H_\infty = 5.94$

(exact at $W = 4$; ≥ 22 sampled at $W = 8$). The width scaling from $W = 8$ to $W = 64$ is measured, not proven: the full design's one-round DP at $W = 64$ is below the 2^{-24} sampling floor, and the 22-round state image shows no shrinkage at 2^{22} resolution — consistent with, but not proof of, the small-width saturation being a $W=4$ artifact; the extrapolation of the image-limited floor to $W = 64$ is engineering judgment (Section III-C2).

Sufficiency of the gates. Passing all calibrated gates is necessary, not sufficient: Tempest v3 requires public cryptanalysis; no security claim beyond the stated metrics is made; and key aliveness is witnessed, not decided in general.

Full-width linear analysis. The linear analysis at full width is measurement-based; an exact bit-level linear bound is future work.

Ablation consistency. *Calibration, not prediction:* the fitness weights that reproduce the 11-generation history were set to agree with those historical verdicts; no hold-out validation of the fitness as a predictor is claimed.

Superseded claims. Earlier drafts asserted the half-entropy law and several $W=4$ values (7.93/7.80, cascade table 5.00/8.00/8.00/7.80/7.80, image 3,902 $\rightarrow$ 248, $\tau = 1.000$, degree 15/14, throughput 15.7/76.5 Gbit/s); each is replaced by the corrected statement where it is used, and the withdrawals, with the interpreter discrepancy that caused them, are part of the audit record in the repository.

D. Conclusion

This work began with a simple question: *can security be measured before a system is built?* The answer, we believe, is yes — and the measurement can even drive automated design.

On the mathematics side, the Polar-Rank Theorem makes one-round differential security exactly computable at any width for the quadratic class (verified at $W = 4$, on random quadratic maps, by derivative tests at $W = 8$, and on Ascon's χ and SIMON), and its linear dual does the same per mask (bias 2^{-128} exactly for the full-state mask at $W = 64$): where the wide trail strategy bounds, AM-SEV computes exactly. The barrier (minimum rank exactly 3, machine-checked, with a $W = 64$ certificate) explains why cascades are necessary; the cascade elimination law quantifies what they add ($2^{-3} \rightarrow 2^{-11}$ exact at $W = 4$, below 2^{-24} sampled at $W = 64$); the Cascade Degree Law closes the algebraic dimension; and the exact multi-round curve (Section III-C2) replaces the heuristic multi-round estimate with a measured, image-limited floor. Every statement is a reproducible computation — including the correction of this paper's own earlier measurements.

The engine side is where the framework stops being theory. Calibrated verifiers reproduce a real weakness (the dead-key v3.1) in milliseconds, where terabytes of output could not localize it; an exact tier decides two-layer differential security at full width via the polar-rank gate; search is generator-agnostic, with LLM proposals adjudicated without favoritism; and the fitness function reproduces the project's own 11-generation history. The instance then carries the load: Tempest v3 passes

every calibrated gate (1 TiB PractRand, BigCrush, NIST, KAT-verified RTL on FPGAs — an iCE40 k=2 bitstream and the full cascade placed and routed on an ECP5-85K).

What remains open is stated openly: exact bit-level linear analysis at full width (the framework's one measured dimension), extension beyond AND-RX to block ciphers and authenticated-encryption-with-associated-data (AEAD), a conversational assistant coupling an LLM proposer with the calibrated verifier stack, and the iCE40-class implementation of the full cascade (a physical scaling limit: the design fits and routes on the ECP5-85K at 7.6% density but not on the iCE40 at 98% density; the ECP5-85K implementation is already complete, with the 6-stage-pipelined bitstream `tempest_pipe2.bit` in the repository). None of this changes the main claim: verifiable, AI-assisted cryptographic design is a workable alternative to trial-and-error, and “design now, verify now” is a reasonable default for cryptographic engineering.

REFERENCES

- [1] S. Sun, L. Hu, P. Wang, K. Qiao, X. Ma, L. Song, and J. Shan, “Automatic security evaluation and (related-key) differential characteristic search: application to SIMON, PRESENT, LBlock, DES(S) and other bit-oriented block ciphers,” CRYPTO 2014, LNCS 8616, pp. 158–178.
- [2] L. Song, Z. Huang, and Q. Yang, “Automatic differential analysis of ARX block ciphers with application to SPECK and LEA,” ACISP 2016, LNCS 9723, pp. 379–394.
- [3] K. Minematsu, T. Suzuki, M. Rahman, S. Suryawanshi, and T. Isobe, “MSX: lightweight block ciphers for microcontrollers with high-assurance against differential and linear attacks,” *IACR Communications in Cryptology*, vol. 2, no. 4, 2025.
- [4] C. Carlet, *Boolean Functions for Cryptography and Coding Theory*. Cambridge University Press, 2021.
- [5] K. Nyberg, “Differentially uniform mappings for cryptography,” EURO-CRYPT 1993, LNCS 765, pp. 55–64, 1994.
- [6] D. Blackman and S. Vigna, “Scrambled linear PRNGs,” *ACM Trans. Math. Softw.*, 47(4), 2021.
- [7] M. E. O'Neill, “PCG: A family of simple fast space-efficient statistically good algorithms for random number generation,” *ACM Trans. Math. Softw.*, 40(4), 2014.
- [8] M. Matsumoto and T. Nishimura, “Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator,” *ACM Trans. Model. Comput. Simul.*, 8(1), 1998.
- [9] D. J. Bernstein, “ChaCha, a variant of Salsa20,” 2008.
- [10] E. Barker and J. Kelsey, “NIST SP 800-90A Rev. 1,” 2015.
- [11] E. Barker et al., “Recommendation for Random Bit Generator (RBG) Constructions,” NIST SP 800-90C, 2025.
- [12] L. Blum, M. Blum, and M. Shub, “A simple unpredictable pseudo-random number generator,” *SIAM J. Comput.*, 15(2), 1986.
- [13] W. Alexi et al., “RSA and Rabin functions: certain parts are as hard as the whole,” *SIAM J. Comput.*, 17(2), 1988.
- [14] D. Shumow and N. Ferguson, “On the possibility of a back door in the NIST SP 800-90 Dual-EC-PRNG,” CRYPTO 2007 Rump Session.
- [15] J. Daemen and V. Rijmen, *The Design of Rijndael*. Springer, 2002.
- [16] S. Vaudenay, *Decorrelation: A Theory for Robust Cryptography*. Springer, LNCS 2500, 2003.
- [17] S. Chen, J. Fan, B. Sun, and L. Li, “SAND: an AND-RX Feistel lightweight block cipher,” *IACR Trans. Symmetric Cryptol.*, 2020(3), 2020.
- [18] C. Dobraunig et al., “Ascon v1.2,” *J. Cryptol.*, vol. 34, p. 33, 2021.
- [19] O. Goldreich, “Candidate one-way functions based on expander graphs,” ECCC TR00-090, 2000.
- [20] B. Applebaum, Y. Ishai, and E. Kushilevitz, “Cryptography in NC^0 ,” FOCS 2004, pp. 166–175.
- [21] A. Gohr, “Improving attacks on round-reduced Speck32/64 using deep learning,” CRYPTO 2019, LNCS 11693, pp. 127–151.
- [22] J. Bellini et al., “CLAASP: a cipher laboratory for analysis and synthesis of symmetric primitives,” 2023.
- [23] Xidian NSS Laboratory, “Xuanzhi (CryptoLLM): the first cryptography-domain large language model,” <https://cryptollm.net/>, 2025.
- [24] A. Biryukov, B. Lambin, and A. Udovenko, “Exact formula for RX-differential probability through modular addition for all rotations,” *IACR ePrint* 2025/550, ToSC 2025, no. 1, pp. 542–591.
- [25] X. Fu, M. Li, S. Lyu, and C. Liu, “Attacks on Goldreich's pseudorandom generators by grouping and solving,” EUROCRYPT 2026, LNCS 16546.

- [26] J. Patarin, "The coefficients H technique," SAC 2003, LNCS 3006, pp. 328–345.
- [27] N. Courtois and J. Pieprzyk, "Cryptanalysis of block ciphers with overdefined systems of equations," ASIACRYPT 2002, LNCS 2501.
- [28] Y. Todo, "Structural evaluation by generalized integral property," EUROCRYPT 2015, LNCS 9056.
- [29] C. Blondeau and K. Nyberg, "New links between differential and linear cryptanalysis," EUROCRYPT 2014, LNCS 8441, pp. 365–383.
- [30] Y. Ishai, A. Sahai, and D. Wagner, "Private circuits: securing hardware against probing attacks," CRYPTO 2003, LNCS 2729, pp. 463–481.
- [31] Statista, "IoT connected devices worldwide, 2019–2033," 2025, <https://www.statista.com/statistics/1183456/iot-connected-devices-worldwide/>.
- [32] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, 2014.
- [33] M. S. Turan et al., "NIST SP 800-90B," 2018.
- [34] V. Lyubashevsky, "The parity problem in the presence of noise, decoding random linear codes, and the subset sum problem," RANDOM 2005, LNCS 3624, pp. 378–389.
- [35] E. Bellini et al., "Fully automated differential-linear attacks against ARX ciphers," CT-RSA 2023, LNCS 13871, pp. 252–276.
- [36] E. Bellini et al., "The window heuristic: automating differential trail search in ARX ciphers," CT-RSA 2025, LNCS 15598, pp. 3–26.
- [37] CryptoliteICE Contributors, "CryptoliteICE: An open-source FPGA-based TRNG," 2026.
- [38] Z. You et al., "Seed hijacking of LLM sampling and quantum random number defense," arXiv:2605.08313, 2026.
- [39] A. Rukhin et al., "NIST SP 800-22 Rev. 1a," 2010.
- [40] C. Doty-Humphrey, "PractRand: C++ library of random number tests," <https://prcrand.sourceforge.net>.
- [41] P. L'Ecuyer and R. Simard, "TestU01: A C library for empirical testing of random number generators," *ACM Trans. Math. Softw.*, 33(4), 2007.
- [42] T. Ashur and Y. Liu, "Rotational cryptanalysis in the presence of constants," FSE 2016, LNCS 9783, pp. 20–45.
- [43] A. Hodjat and I. Verbauwhede, "A 21.54 Gbit/s fully pipelined AES processor on FPGA," IEEE FCCM 2004, pp. 308–309.
- [44] R. Dhanda, S. Gupta, and S. Nandi, "A compact and efficient AES-32GF for encryption in small IoT devices," *MethodsX*, vol. 10, 2023.
- [45] A. Biere, K. Fazekas, M. Fleury, and M. Heisinger, "CaDiCaL, Kissat, ParaCoBa, Plingeling and Treengeling entering the SAT competition 2020," SAT Competition 2020.
- [46] S. Babbage, "Improved exhaustive search attacks on stream ciphers," European Convention on Security and Detection, 1995.
- [47] V. Yakovliev, "Differential properties of LRX-analogues of small constant multiplication," *Int. J. Electron. Telecommun.*, 71(1), 2025.

致 谢

这篇论文背后的算法经历了十一代设计、超过二十个变体，以及数次深刻的自我修正。回头看，这段旅程最困难的从来不是某个定理，而是学会在每一个数字面前保持诚实。论文最终能够交付，离不开许多人的帮助，谨在此致以诚挚的谢意。我负责了寻找研究方向、构思研究思路、进行计算机实验和撰写论文。

首先要衷心感谢我的指导老师高洁老师与吴琼老师。高洁老师是我在学校的数学竞赛教练，为我打下了数论和代数的知识基础。为了了解专业学术论文的写作规范，我找到了西安工业大学的吴琼教授进行指导。在论文结构组织、图表呈现与论文写作规范方面，老师们提出了许多宝贵的修改意见。指导为无偿性质，两位老师从未参与实验计算、验证代码的编写与论文主体的撰写，全部技术内容均由本人独立完成并负责。

还要感谢 `awesome-cryptography-rust` (`rust-cc` 组织维护的 Rust 密码学精选开源列表) 的维护者们：本论文的代码仓库有幸被该列表收录。感谢 `wolfSSL` 的 David Garske 先生，感谢他对算法的审阅。开源社区对独立研究者的这份善意，是这项工作的重要组成部分——论文反复强调每个声明都应能被任何人独立复核，而正是这样，让社区复核成为可能。谨向维护者的辛勤付出致以诚挚的谢意。

关于本论文的研究过程，在此作如下说明。

选题源于对真实安全事件的调研：2025 年 AMD 处理器号称硬件真随机的指令大量返回全零 (CVE-2025-62626)、加密货币钱包因弱随机种子损失约十亿美元 (CVE-2023-39910)、路由器与医疗设备上排成长队的同类漏洞——它们共同指向一个问题：设计者在交付之前无法测量自己的随机数构造是否安全。论文正是围绕能否在设计当天测量安全这一问题展开的。作为个人参赛项目，研究过程全程由本人完成：包括编写并运行全部验证脚本 (W=4 全域精确枚举与 800 位精度分离、W=8/16/32/64 采样、NIST SP 800-22、TestU01、PractRand 一太字节、Dieharder 全套、FPGA 综合与布线日志的复现)，设计并实施差分、线性、代数、SAT 预像、状态恢复五条攻击线的实验，以及论文正文、图表与审计记录的撰写排版。

研究过程中的一段难忘经历，是算法的生态集成。为了让 Tempest 真正被真实世界使用，我曾尝试把它接入主流密码学库，最终认识到这些库的随机数接口承担着信任根的语义——由第三方算法替换它们在安全上并不成立。这促使我转向各生态为随机生成器设计的标准接口：Rust 的 `rand_core`、Go 的 `math/rand/v2`、C++ 的 `<random>`、GNU 科学计算库 `GSL`、NumPy 的 `BitGenerator`，以及 `OpenSSL` 的 `provider` 机制，全部通过 KAT 逐位验证。特别要感谢 `wolfSSL` 的 David Garske 先生：他在 issue 中坦诚说明了库内不支持，并建议了 `crypto callback` 这一外部接入路径——遵循他的建议，我完成了 `wolfSSL` 的 TLS 握手集成。这些经历让我体会到：一项工作要被社区接纳，遵

循生态已有的接口与规则，往往与技术本身同样重要。

研究过程中遇到的最大困难，是对自己研究成果的反复质疑与修正。第一次困难发生在最终验证阶段：我们发现早期 $W=4$ 枚举器忽略了快照语义，头条数字描述的其实是另一个轮函数。面对这个发现，我们选择撤回相关声明、重算全部数据、以 Polar-Rank 定理重构理论核心，并把完整的修正过程作为审计记录随论文发布。第二次困难是 v3.1 的缺陷——某些密钥下输出退化为平凡序列，统计测试对此毫无察觉，最终由校准验证器精确捕捉并定位。第三次困难出现在 2026 年 8 月：重建测试发现 v3.1 的线性部分存在隐藏结构缺陷（全 1 向量是核元素，秩仅 248/256，含 64 步塌缩方向），经过逐项代数诊断，用每字去掉一个旋转项的最小修改恢复满秩，本机全部测试套件随后均通过。三次困难的全部记录（包括失败数据）都在仓库的审计文档中，可以复现。另外还有许许多多的失败与迭代升级，这也是一个密码学算法的必经之路，让我受益匪浅。

最后，按照竞赛要求如实说明本研究对人工智能工具的使用情况。研究过程中使用了 deepseek v4 flash(通过 claudecode 平台接入)，用途包括三个方面：其一，在 AI 设计引擎中作为提案生成器参与候选算法结构的设计——所有提案均经过校准验证器栈以机器可查的数学度量（秩、代数度、密钥相关性等）裁决，不依赖模型的判断；其二，辅助验证、整理部分验证脚本；其三，论文文字的组织与润色。

参赛者：田岳洲

2026 年 8 月