A CMOS Electrical Non-Persistence Primitive for Single-Use Secrets

Francis X. Cunnane III
QSymbolic LLC
Virginia Beach, USA
Email: frank@qsymbolic.com

***Abstract*—We present a CMOS-compatible *Electrical Non-Persistence Primitive (ENP-P)* that enforces a strict hardware-level rule for ephemeral secrets in the *modeled digital design*: a read returns the true secret or an inert logical output—nothing in between at the interface. Each ENP-P cell stores an encoded value under basis b; a read with supplied basis b' performs a hardware comparison. A matching basis reveals the secret exactly once, while a mismatching basis immediately yields an inert, ground-referenced logical output. In both cases, comparison triggers a one-directional grounding event in the design semantics that sinks the internal encoding through a dedicated discharge path. Grounding is modeled as a monotonic electrical transition—independent of clocks, metastability, or control flow at the RTL boundary—and is *intended* in ASIC realizations to leave no node capable of retaining charge patterns correlated with the secret. Unlike synchronous clear-on-read or zeroization wrappers, there is no temporal window in which the value is first exposed at the logical interface and only later overwritten: in our implementation and model, the act of measurement and the logical destruction of the encoding are the same event. ENP-P thus eliminates remanence, replay, speculative duplication, and transient-execution leakage *at the modeled digital interface*, and is designed to provide single-use semantics for PQC decapsulation, secure boot, enclaves, and hardware security modules. ASIC-level analog grounding behavior and resistance to invasive physical attacks are design goals left to future work.**

***Index Terms*—Ephemeral secrets, hardware security, post-quantum cryptography, grounding networks, electrical non-persistence, single-use memory**

I. Introduction

Modern cryptographic systems assume that sensitive values can be safely erased once no longer needed. In practice, this assumption rarely holds. Commodity hardware retains secrets in electrically recoverable form long after software has "cleared" memory: DRAM decays slowly and preserves charge for milliseconds to minutes; classic cold-boot experiments demonstrate that DRAM contents can be imaged seconds to minutes—or longer at low temperature—after power loss [1]; and even with scrambling and modern controllers, cold-boot style attacks remain viable [2]. SRAM bitcells and on-chip arrays are also vulnerable to power-domain manipulation and freezing attacks, showing that remanence is not a DRAM-only phenomenon [6]. Caches, write buffers, coherence structures, and speculative execution units all retain replicas until overwritten. Clearing a buffer removes its *logical* value, not its physical footprint.

This mismatch has enabled cold-boot attacks, DMA and bus-snooping, Rowhammer disturbance, Spectre- and Meltdown-style transient leaks, glitch-induced metastability recovery, and debug-fabric extraction. These attacks all succeed because mainstream hardware provides no primitive ensuring that a secret ceases to exist electrically after use.

Post-quantum cryptography (PQC) magnifies the risk. Ephemeral private keys used during decapsulation are large, high-entropy, and instantiated frequently. Their compromise breaks forward secrecy and undermines protocol robustness. Recent NIST guidance (e.g., SP 1800–38 [4]) warns that ephemeral PQC keys must not remain electrically recoverable, yet current architectures lack a mechanism to enforce this. In parallel, NIST IR 8517 catalogs a wide range of hardware security failure scenarios and attack surfaces, underscoring the need for primitive-level guarantees rather than ad hoc mitigations [5].

We close this gap by introducing a CMOS-compatible *Electrical Non-Persistence Primitive (ENP-P)* providing deterministic single-use semantics at the hardware boundary of the digital design. Any read triggers an asymmetric grounding event in the modeled circuit that sinks the encoded state into a dedicated discharge path, preventing re-read, replay, or logical reconstruction. Unlike conventional memory—which leaves secrets as stable charges, metastable nodes, or microarchitectural shadows—the ENP-P is constructed so that, after grounding, no modeled node at the RTL level retains information correlated with the secret. We show that the mechanism integrates cleanly with CMOS, that it admits efficient hardware realization on FPGA for logical semantics, and that it supports practical deployment in PQC decapsulation, secure boot, enclave initialization, and secure-element workflows.

In this work, all experimentally validated guarantees refer to the *logical* behavior of the ENP-P as realized on an Intel Cyclone V FPGA: a basis-conditioned read discloses a stored value at most once and then renders the cell inert at the digital interface. ASIC-level analog grounding dynamics, transistor-level charge behavior, and resistance to invasive physical attacks are design objectives and remain future work.

Throughout, we use the term *dead circuit* to denote an inert, ground-referenced *logical* output state at the interface of the modeled design, not a claim about permanent physical damage or impossibility of analog-level probing in all conditions.

II. Background and Motivation

A. Electrical Persistence in Commodity Memory

Memory technologies store information as charge or bistable state, persisting long after logical overwrite. DRAM charge decays gradually; seminal cold-boot work by Halderman *et al.* shows that DRAM retains sufficient structure to recover cryptographic keys long after power loss [1]. Subsequent work demonstrates that DRAM scrambling and modern controllers do not fully eliminate this threat, and that cold-boot style remanence attacks remain viable on contemporary platforms [2]. SRAM nodes retain state through parasitic capacitances and can be exploited via power-domain separation and environmental manipulation [6]. Overwriting a buffer fails to immediately erase the underlying electrical pattern. NIST SP 1800–38 emphasizes that ephemeral private keys must not remain electrically recoverable, but commodity memory cannot enforce this [4].

B. Microarchitectural Replication

Modern processors replicate data across caches, reorder buffers, speculation units, DMA queues, ECC logic, and coherence FIFOs. These microarchitectural shadows persist unpredictably. Software clearing cannot reach them, and many of the failure scenarios documented in NIST IR 8517 arise precisely because such internal buffers outlive their intended logical lifetime [5].

C. Exploitation of Residual State

Cold-boot attacks, DMA snooping, Rowhammer, glitch recovery, and transient-execution leaks all exploit persistent or duplicated electrical state. Each arises because hardware provides only read-many memory, even for values that must be read once. Cold-boot style attacks remain possible even in the presence of memory scrambling [2], and recent work shows that on-chip SRAM and register banks can be abused under power-domain separation [6].

D. Escalation Under PQC

PQC decapsulation uses large ephemeral keys. Their compromise breaks forward secrecy entirely. Because these keys are short-lived yet electrically persistent, any leakage channel poses serious risk. SP 1800–38 explicitly calls out the need to protect ephemeral PQC secrets in memory and microarchitectural state [4].

E. Clear-on-Read and Zeroization Wrappers

Several commercial designs and academic proposals expose a "clear-on-read" or "read-and-zeroize" behavior, typically implemented by wiring read strobes into synchronous clear signals on flip–flops or RAM macros. In these designs, the secret is first transferred into architectural state and only then overwritten on a subsequent clock edge or control event. This leaves a window in which the value can be sampled by microarchitectural paths, faulted before clearing, or revived from decay.

ENP-P differs fundamentally at the modeled interface: there is no phase in which the secret is held stable in a re-readable memory location and then asynchronously scrubbed. The same logical event that resolves the encoding under the correct basis also triggers the collapse that invalidates it in the digital design. The secret never exists as a stable, re-readable memory state in the way clear-on-read RAM does.

F. Need for a Single-Use Hardware Primitive

Commodity memory is designed for stability, not guaranteed erasure. A secure primitive must guarantee:

- single-read disclosure at the interface,
- electrical non-persistence after consumption (as a design goal),
- non-replayability,
- elimination of speculative and physical recovery channels *at the logical boundary*.

The ENP-P is designed to provide this guarantee and is intended to address remanence-driven failures of the kind catalogued in NIST IR 8517 [5].

III. Threat Model

We consider an attacker attempting to recover a secret *after* the legitimate read. Capabilities include memory observation (DMA, snooping), microarchitectural leakage (speculation, caches), physical attacks (cold-boot, EM probing, glitching), and fault injection, consistent with the classes of threats documented in NIST IR 8517 [5] and explored empirically in cold-boot and SRAM remanence work [1], [6].

We assume correct initialization, honest CMOS fabrication, one trusted read, and nominal operating conditions. The primitive does not protect against disclosure *before* the grounding-triggering read or against invasive microscopy and transistor-level modification.

The security goal is **electrical non-persistence at the primitive boundary**: after the first read, no stable or recoverable *logical* state correlated with the secret exists in the modeled digital design. Achieving full analog electrical non-persistence on fabricated ASICs is a design objective for future work.

IV. Overview of the Electrical Non-Persistence Primitive

We introduce a CMOS-compatible *Electrical Non-Persistence Primitive (ENP-P)* that enforces: *a read returns the true secret or a dead (inert) logical circuit.*

Each cell stores a secret encoded under basis b. A read with supplied basis b' performs an exact comparison:

- if $b' = b$, the secret is disclosed exactly once at the interface;
- if $b' \neq b$, the output rails immediately enter their inert, ground-referenced logical state.

In both cases, comparison triggers a one-directional *grounding event* that sinks the encoded state through a dedicated discharge network in the intended ASIC design. The transition is physically asymmetric in that design goal—unlike RAM or flip–flops, it is not *intended* to depend on synchronous clocks or microarchitectural control flow; however, robustness to glitches, fault injection, and all analog effects must be validated on silicon. In the FPGA prototype, these grounding semantics are modeled as a single-cycle collapse in the digital design.

A. High-Level Operation

The cell implements:

- $E(v, b)$ encoding that resolves only under the correct basis,
- inert output on mismatch,
- unconditional grounding of $E(v, b)$ and b for any read in the modeled circuit.

After grounding in the digital design, no modeled node retains recoverable information about the secret.

B. Grounding Mechanism

Basis comparison activates a directional discharge path tied to electrical ground in ASIC realizations. Conceptually, charge dissipates monotonically with no stable metastable points, ensuring:

- the secret need not enter registers, FIFOs, or caches,
- speculative or partial reads at the interface capture nothing beyond the single authorized disclosure,
- grounding completes before any replication is possible at the logical level.

The FPGA prototype cannot reproduce analog discharge but enforces the same logical invariant: the cell transitions to an inert state in the same event that discloses the value at the interface.

C. Security Properties

At the modeled digital interface, the ENP-P:

- eliminates remanence and replay by preventing re-read of the secret,
- prevents speculative re-use of the secret in microarchitectural structures that never see it,
- makes mismatched-basis reads indistinguishable from already-consumed cells.

These guarantees are stated for the digital design; their extension to fabricated ASICs against invasive physical attacks depends on layout, process, and measurement and is treated as future work.

D. CMOS Compatibility

All operations—basis encoding, comparison, output resolution, grounding—are implemented using standard CMOS combinational logic and ground-tethered discharge structures. No exotic fabrication is required in principle. FPGA prototypes emulate the grounding semantics at the RTL level and validate single-use behavior and collapse timing in the digital domain.

V. ENP-P Architecture and Circuit Design

This section describes the circuit structure of the Electrical Non-Persistence Primitive (ENP-P), including its encoding function, basis comparison, one-shot output resolution, and the grounding network that is intended to ensure electrical non-persistence on ASIC. The design uses only standard CMOS devices and is intended for straightforward ASIC realization; our FPGA implementation validates the logical semantics.

A. Cell-Level Description

At a high level, each ENP-P cell consists of four components:

- an encoding network implementing $E(v, b)$,
- a basis storage latch holding the per-cell basis b,
- a basis-comparison and output-selection path,
- a collapse trigger and directional grounding network that irreversibly sinks charge in the intended ASIC realization.

On a read request, the supplied basis b' is compared against the stored b. A basis match routes the output through a reconstructive decode path that yields the true secret v exactly once. A mismatch immediately routes the output rails to their inert grounded logical state and never exposes intermediate internal nodes at the interface. In both outcomes, the comparison event sets a collapse trigger that activates the grounding network, which destroys all residual charge in the encoding network and permanently renders the cell electrically inert with respect to v in the intended ASIC design; in the FPGA prototype, the corresponding state bits are deterministically cleared at the RTL level.

B. Encoding Function $E(v, b)$*E(v,b)*

The cell stores an encoded representation of the secret such that:

1) no internal node equals v directly,
2) the mapping is reversible only under the correct basis $b' = b$,
3) the mapping is non-resolving under any mismatching basis.

A practical encoding is a per-bit masked structure:

$$E(v, b) = v \oplus b,$$

or a non-linear extension thereof. In ASIC designs, $E(v, b)$ is realized as a network of XOR/XNOR gates fed by the basis bits. Because the basis is fixed per cell, its effect propagates structurally into the encode graph and determines the grounding trajectory (Section V-E).

Crucially, the encoding network feeds exclusively into the comparison logic and grounding path; it is never latched into registers, FIFOs, or sequential logic in the intended integration.

C. Basis Storage and Comparison

The per-cell basis b is stored in a small latch or non-resetting flip-flop configured to avoid metastability on write.

The supplied basis b' is compared bitwise against b using standard CMOS comparators. A global match signal:

$$\mathsf{match} = \bigwedge_i (b_i = b_i')$$

drives both:

- the output multiplexer (selecting v vs. dead circuit), and
- the trigger for the grounding network.

Comparison is performed using purely combinational logic; no explicit timing window is introduced in which partially resolved internal states are copied into architectural state.

D. Output Resolution and Dead-Circuit State

The output network is a two-branch, one-shot structure:

- **Match path:** If $\mathsf{match} = 1$, the encoded value is resolved to v through a decode network $E^{-1}(\cdot, b)$.
- **Mismatch path:** If $\mathsf{match} = 0$, the output drivers are immediately tied to the inert grounded rail, producing a dead-circuit logical state indistinguishable from an already-consumed cell at the interface.

The decode path is combinational and lacks any sequential elements; once grounding begins, the decode network is also drained in the ASIC design, preventing any later attempt to reconstruct the value. In the FPGA model, the corresponding state bits are cleared in the same cycle.

Timing is deliberately biased in favor of grounding in the intended ASIC: the discharge network begins pulling down internal nodes as soon as comparison completes, and output resolution must occur before this discharge front reaches the decode network. This ensures that the secret is observable for exactly one read under $b' = b$ in the idealized timing model; validating this timing margin in silicon is future work.

E. Grounding Network Design

The grounding network is the mechanism that provides electrical non-persistence as a design goal. It consists of:

- directional NMOS pull-down structures tied to global ground,
- basis-dependent activation paths determining discharge trajectories,
- isolation transistors preventing back-charging or lateral propagation.

Once triggered, grounding is intended to proceed as an asymmetric, monotonic physical event:

- internal encode nodes are discharged toward ground with no stable intermediate voltages under nominal conditions,
- the discharge network provides no feedback path to restore charge,
- the output rails also transition to their inert level under mismatch,
- the encoded state cannot be held, stalled, or re-sampled in the modeled digital design.

a) Basis-Dependent Grounding Diversity.: Different basis configurations b alter the structure of $E(v, b)$ and therefore the ordering and routing of the grounding discharge front. This diversification is intended to ensure that no single measurement point suffices to capture partial information across different cells in an ASIC implementation.

b) No Sequential Recovery (Modeled).: Because all nodes in the encoding and comparison network become shorted to ground during the event in the envisioned ASIC, and are cleared in the FPGA prototype, no metastable or half-charged state persists in the modeled digital design that an attacker could subsequently amplify or probe. Physical verification of this property on silicon is future work.

F. ASIC Realization

All ENP-P components can be implemented using standard CMOS devices:

- **Encode network:** XOR/XNOR trees and small combinational S-box variants for diversity.
- **Basis storage:** Minimal latches; no clocked reset required.
- **Comparison network:** CMOS comparators and a tree-structured AND reduction.
- **Output stage:** Transmission-gate multiplexer feeding either the decoded value or a hardwired ground rail.
- **Grounding network:** NMOS pull-down transistors arranged in a directional tree with controlled activation.

We emphasize that the FPGA evaluation in this paper validates only the single-use and collapse semantics at the RTL level; ASIC tapeout and physical characterization are required to confirm the analog grounding behavior and resistance to invasive attacks.

VI. Security Analysis

We analyze the ENP-P under the threat model of Section III, focusing on adversaries who attempt to recover, replay, or determine the prior existence of secrets *after* the triggering read has occurred. The central invariant of the primitive is that any basis comparison—match or mismatch—immediately initiates a one-directional grounding event that eliminates the encoded state from the modeled digital design. After this transition, no logical element retains information correlated with the secret or with the basis under which it was stored.

A. Post-Read Non-Recoverability

Once a read occurs, ENP-P cells become logically inert with respect to the stored secret. A correct-basis read discloses the secret exactly once; a mismatching basis produces the inert dead-circuit output; but in both cases the grounding event removes all pre-read logical state. Because the FPGA prototype implements grounding as a deterministic collapse within the RTL, the following modeled properties hold:

- **No re-read:** A second read of the same cell yields only the inert output; the original encoding is no longer present.

- **No replay:** No metastable or partially-decayed encoding exists in the synthesized digital design that could be revived.
- **No modeled physical reconstruction:** After collapse, the design contains no logical node whose value remains correlated with the secret.

These guarantees apply to the digital model; ASIC-level charge measurements require future physical evaluation.

B. Indistinguishability of Mismatch and Consumed States

ENP-P deliberately ensures that the mismatch case ($b' \neq b$) is indistinguishable from a post-consumption cell. Both produce identical inert outputs and trigger the same collapse behavior. An adversary with post-read access cannot determine:

- whether a secret was ever stored,
- whether the basis was incorrect, or
- whether the cell was never initialized.

This indistinguishability prevents adversaries from using oracle-style probing to detect the prior presence of secret material.

C. Microarchitectural Non-Replication

Because ENP-P grounding completes before any architectural or microarchitectural structure can observe the secret, no replicas of v exist outside the cell. In particular:

- **No cache residency:** The secret is not staged into load/store paths or CPU caches.
- **No register-file or pipeline shadows:** The value does not enter rename tables, reorder buffers, forwarding networks, or speculation units.
- **No DMA or coherence exposure:** The secret does not appear in coherence queues, DMA descriptors, or I/O buffers.

Thus, the ENP-P acts as a hardware boundary preventing secret replication into any microarchitectural path.

D. Timing and Constant-Time Considerations

It is important to clarify that *ENP-P does not, by itself, guarantee constant-time behavior*. The term "atomic" in this work refers to the logical coupling of disclosure and collapse in a single modeled transition, not to uniformity of timing or power characteristics under all inputs or across all physical observables.

Several timing-related considerations arise:

- The basis-comparator is standard CMOS logic whose propagation delay may vary with input patterns.
- The decode and mismatch paths may have different switching activity or power signatures.
- In ASIC realizations, the analog dynamics of grounding may emit data-dependent EM or power traces unless further masked.

Therefore, although the ENP-P ensures *single-read semantics* and *post-read non-persistence*, it does *not* provide inherent protection against timing or side-channel leakage during the read/collapse event itself. Conventional countermeasures—balanced comparators, masking, reshuffling, delay randomization, and EM-hardening—remain complementary and may be needed for high-assurance deployments.

E. Physical and Fault Attacks

The ENP-P eliminates the logical residual state that underlies remanence-driven attacks, but it does not eliminate all physical vectors. In the modeled design:

- **Cold-boot and decay attacks:** No persistent charge exists in registers or RAM paths, as secrets never enter those structures.
- **Clock/voltage glitching:** Because collapse is not staged through a sequential scrub, glitching the clock cannot restore a secret after collapse in the FPGA model.
- **Metastability revival:** The encode network contains no partially-held state after grounding in the RTL.

However, ASIC-level validation is required to characterize:

- sub-cycle glitch effects on grounding trajectories,
- EM or power leakage during the collapse event,
- transient analog behavior in transistor-level discharge.

These are left to future fabrication and lab evaluation.

F. Security Scope

Within the modeled digital semantics, the ENP-P guarantees:

- **Single-use disclosure:** The secret is revealed exactly once under a correct basis.
- **Post-read non-persistence:** No logical reconstruction is possible after collapse.
- **Mismatch indistinguishability:** Mismatched reads are indistinguishable from consumed cells.
- **No microarchitectural replication:** No architectural or microarchitectural structure receives the secret.

The ENP-P does *not* guarantee:

- constant-time execution,
- power or EM side-channel resistance during collapse,
- protection against invasive physical attacks,
- retroactive protection against pre-read leakage.

These limitations define the boundary of the primitive's guarantees and motivate continued ASIC-level investigation.

VII. Formalization of ENP-P Security

We formalize the core post-read property of the Electrical Non-Persistence Primitive (ENP-P): an adversary with arbitrary access *after* the first read cannot determine whether the cell previously stored a valid secret or was never valid to begin with, based on the modeled digital behavior. This captures the "true-secret-or-dead-circuit" invariant at the interface.

A. Indistinguishability Game

We define experiment $\mathsf{IND\text{-}ENP}$ between a challenger $\mathcal{C}$ and adversary $\mathcal{A}$:

1) **Setup.** $\mathcal{C}$ samples a secret $v \leftarrow \{0,1\}^\lambda$ and basis $b \leftarrow \{0,1\}^k$.

2) **Challenge Bit.** $\mathcal{C}$ chooses $c \leftarrow \{0,1\}$.
3) **Initialization.**
   - If $c = 1$: initialize the cell with (v, b).
   - If $c = 0$: either (a) initialize with (v, b') for $b' \neq b$, or (b) leave the cell uninitialized.
4) **Single Read.** $\mathcal{C}$ performs a single read using basis b: outputting v if $b' = b$ (and thus $c = 1$), or the inert grounded state otherwise. Grounding is now complete in the model.
5) **Post-Read Access.** $\mathcal{A}$ receives:
   - the final grounded cell hardware (as an abstract interface),
   - full probing/timing/EM observation ability at the interface,
   - unlimited post-read interface interactions.
6) **Guess.** $\mathcal{A}$ outputs c' and wins if $c' = c$.

ENP-P achieves *post-read indistinguishability* in this model if for all polynomial-time $\mathcal{A}$:

$$\left|\Pr[c' = c] - \tfrac{1}{2}\right| \leq \mathsf{negl}(\lambda).$$

This ensures that grounded cells reveal no evidence of past secrets through their modeled digital behavior. It does not by itself capture physical side channels such as instantaneous power or EM emissions during the collapse event.

VIII. Security Claims for the ENP-P

We now state the key ENP-P security properties in lemma form. Each proof sketch appeals to the modeled grounding semantics and idealized ASIC design. None of these claims should be interpreted as proven for all physical realizations without further lab evaluation.

A. Single-Read Security

Lemma 1 (Single-Read Property): After the first invocation of $\mathsf{read}(b')$ in the modeled design, no subsequent operation—including repeated reads, probing through the specified interface, or fault injection on modeled signals—can output v again.

Proof Sketch: Grounding sinks all encode nodes to inert potentials in the idealized design and clears the corresponding state bits in the FPGA model. No stable or metastable state correlated with v remains at the digital boundary. ■

B. Indistinguishability of Consumed and Mismatched Cells

Lemma 2 (Indistinguishability of Consumed and Mismatched Cells): A grounded cell produced from $b' \neq b$ is indistinguishable, at the modeled interface, from a grounded cell that previously contained (v, b).

Proof Sketch: Both yield the same inert logical output rails and identical grounding behavior in the model. Probing exposed signals shows only settled ground-referenced nodes without dependence on v. ■

C. Absence of Microarchitectural Shadows

Lemma 3 (No Microarchitectural Shadow Copies): Under correct integration that prevents ENP-P outputs from being written into ordinary memories, the secret v never enters RAM, caches, register files, reorder buffers, or DMA paths.

Proof Sketch: Output resolution and grounding occur in the same transition at the ENP-P boundary. No clock edge in the surrounding microarchitecture is required to capture v into any structure; avoiding explicit software copy or bus transfer in system design prevents such exposure. ■

D. Robustness Against Physical and Fault Attacks (Modeled)

Lemma 4 (Non-Recoverability Under Physical Attack (Idealized)): In the idealized ASIC design with correctly functioning grounding network, cold-boot recovery, glitching, EM probing, and decay analysis reveal no information about v once grounding completes.

Proof Sketch: Encode nodes are tied to ground with no remaining charge gradients in the idealized physical model; in the FPGA model, state bits are cleared. A uniform ground-referenced state contains zero extractable entropy about v. Demonstrating this behavior under real lab conditions is future work. ■

IX. Integration With PQC Decapsulation (ML-KEM)

Ephemeral secrets in ML-KEM decapsulation are large and extremely sensitive. Standard implementations store them in RAM or microarchitectural buffers, producing many leakage channels.

A. Integration Strategy

Using ENP-P cells, the ephemeral private key $\mathbf{s}$ is:

- written once when parsing the ciphertext,
- never stored in RAM, cache, or register files if integration avoids those paths,
- consumed coefficient-by-coefficient through ENP-P reads,
- grounded immediately after each retrieval in the modeled design.

Each coefficient s_i is retrieved via exactly one call to $\mathsf{read}(b_i)$; the cell becomes permanently inert after that call.

B. Security Benefits

This removes the dominant *logical* persistence channel for $\mathbf{s}$:

- no DRAM/SRAM persistence of $\mathbf{s}$ in the modeled integration,
- no cache-line and transient-execution leakage for values that never reach the microarchitecture,
- no DMA or peripheral extraction via ordinary memory paths,
- no cold-boot or rowhammer recovery on conventional memory arrays,
- no mis-speculated reuse of key material through standard load/store paths.

The ENP-P enforces the protocol-level requirement that PQC decapsulation keys are truly ephemeral at the hardware boundary, aligning with the objectives of SP 1800–38 [4]. Physical side channels during the collapse event itself remain in scope for conventional masking and countermeasures.

X. Comparison With Existing Memory Primitives

Table I compares ENP-P against conventional DRAM, SRAM/flip–flops, and typical HSM buffers along key dimensions such as electrical persistence, replay, speculative leakage, and basis-gated access. All ENP-P entries refer to the *modeled digital behavior*; ASIC-level physical behavior is a design goal to be validated separately.

XI. Implementation and Evaluation

This section presents our hardware prototype, validation methodology, and measured results. All experiments were performed on an Intel Cyclone V SoC (5CSEBA6U23I7) using Quartus Prime 25.1. The FPGA implementation models the *logical semantics* of the ENP-P: a basis-conditioned read resolves the stored value exactly once and immediately collapses the cell into an inert dead-circuit state at the digital interface. No sequential clearing, scrubbing, or overwrite logic is involved beyond the collapse event.

In this implementation, ENP-P is realized as a 64×256-bit *Atomic Memory* array, which we refer to as *ROOM* (Read Only-Once Memory). In the remainder of this section we use ENP-P and ROOM interchangeably for this concrete instantiation. Unlike "clear-on-read" RAM macros that first stage the secret into a flip–flop and only later zeroize it, the read gate and the logical destruction of the encoded state are the same event in this design: there is no clock edge or firmware action in which the value exists as a re-readable memory word that can then be scrubbed.

A. Prototype Setup

The prototype instantiates:

- a 64-cell collapse bank (each cell 256 bits),
- basis-gated read/resolve logic,
- an Avalon-MM wrapper for register-level access,
- a System Console test harness for deterministic stimulation.

The FPGA cannot replicate the *analog grounding dynamics* of the ASIC design, but it faithfully enforces the timing and logical semantics of atomic collapse: *a read event consumes the encoding in the same cycle that reveals it*. There is never a phase in which the secret is held in a stable sequential element and only later cleared by a separate control path in the modeled digital design.

B. Functional Correctness

We exercised the complete 64×256-bit array using randomized System Console scripts to validate correct-basis disclosure, wrong-basis behavior, and post-consumption inertness. Each test trial performs:

1) writes a full 256-bit value (split across eight 32-bit INIT words),
2) writes an 8-bit basis,
3) performs **Read 1** using either:
   - the *correct basis* (expected to disclose the secret exactly once), or
   - a *wrong basis* (expected to immediately return the inert consumed-state output),
4) performs **Read 2** (expected in all cases to return the inert state),
5) performs optional additional reads (expected to remain inert).

Across all trials, the following properties held in the FPGA prototype:

- **Correct-basis first reads** always returned the exact 256-bit stored value.
- **Wrong-basis first reads** always returned the inert "dead" output— *bitwise identical* to the output of a cell that had already been consumed or had never been provisioned.
- **Second and subsequent reads** always produced the inert state, regardless of basis.
- **No cell ever re-disclosed** after its first access.
- **Mismatch, consumed, and uninitialized cells** were indistinguishable at the interface, confirming the ENP-P indistinguishability property in the modeled design.

Representative correct-basis behavior:

```
INIT value:
  0xB35A53D130FB11FE4AED89F052E92E86
  CB8FE0B34E93D52A18E898FC65431C14

READ #1 (correct basis):
  0xB35A53D130FB11FE4AED89F052E92E86
  CB8FE0B34E93D52A18E898FC65431C14

READ #2:
  0x00000000000000000000000000000000
  00000000000000000000000000000000
```

Representative wrong-basis behavior:

```
READ #1 (wrong basis):
  0x00000000000000000000000000000000
  00000000000000000000000000000000

READ #2:
  0x00000000000000000000000000000000
  00000000000000000000000000000000
```

These experiments confirm that the FPGA prototype enforces *single-disclosure*, *wrong-basis inertness*, and *irreversible consumption* at the digital interface, and that its inert output is indistinguishable from a cell that has already been consumed or never provisioned—precisely the required ENP-P semantics in the model.

Property	DRAM	SRAM/FF	HSM Buffers	ENP-P (modeled)
Electrical persistence	ms–min	indefinite	vendor-specific	**none (grounded, modeled)**
Re-read allowed	yes	yes	conditional	**never (digital)**
Replay / revival	feasible	feasible	limited	**not possible (digital)**
Speculative leakage	yes	yes	partial	**none at ENP-P boundary**
Fault-based revival	feasible	feasible	limited	**not possible (digital)**
Basis-gated access	no	no	partial	**yes**
Post-read indistinguishability	no	no	partial	**yes (modeled)**
ASIC simplicity	high	high	medium	**high**

TABLE I
COMPARISON OF STORAGE PRIMITIVES FOR EPHEMERAL SECRETS. ENP-P IS THE ONLY PRIMITIVE IN THIS COMPARISON PROVIDING DETERMINISTIC SINGLE-USE SEMANTICS WITH GUARANTEED POST-READ NON-PERSISTENCE AT THE MODELED DIGITAL INTERFACE. ASIC-LEVEL BEHAVIOR MUST BE VALIDATED ON SILICON.

TABLE II
FPGA RESOURCE UTILIZATION (CYCLONE V 5CSEBA6U23I7)

Resource	Used	Available	Utilization
Adaptive Logic Modules	10,268	41,910	24%
Total Registers	17,892	–	–
Block RAM bits	512	5,662,720	$<0.01\%$
M10K RAM blocks	1	553	0.18%
DSP blocks	0	112	0%

C. Resource Utilization

The ROOM array occupies roughly one quarter of the fabric and stores *all* secrets in logic—not RAM—underscoring that it is a single-use measurement primitive rather than a conventional memory macro.

D. Timing Analysis

We constrained the design at 50 MHz (20 ns). Table III reports the achieved $f_{\max}$ across worst-case PVT corners.

TABLE III
MEASURED f_{MAX} ACROSS PVT CORNERS

Corner	$f_{\max}$
Slow 1.1 V, 100 °C	63.22 MHz
Slow 1.1 V, −40°C	63.90 MHz

The design meets timing at the target frequency with more than 26% margin. Setup and hold slack are shown in Table IV.

TABLE IV
SETUP AND HOLD SLACK (SLOW 1.1 V)

Corner	Setup Slack	Hold Slack
100 °C	4.157 ns	0.215 ns
−40°C	4.351 ns	0.309 ns

No negative slack or timing violations were observed.

E. Metastability Considerations

The design contains no ring oscillators, entropy sources, or multi-phase CDC logic. Quartus identified 17 two-stage synchronizers, all associated with Avalon-MM handshaking. Estimated MTBF exceeds 10^9 years under slow–hot PVT. No metastability paths appear in the collapse datapath, which is entirely synchronous in the FPGA model.

F. Summary

The FPGA evaluation demonstrates that:

- 256-bit values are disclosed exactly once at the interface,
- collapse is atomic, deterministic, and irreversible in the modeled digital behavior,
- post-collapse reads reveal no residual information in the design,
- timing closure is robust across all corners,
- resource usage is modest and scales cleanly.

Although the FPGA cannot model analog grounding, it faithfully validates the *electrical non-persistence semantics at the digital boundary* of the proposed ENP-P primitive. ASIC fabrication and lab characterization are required to extend these claims to physical node behavior.

XII. APPLICATIONS

The ENP-P is intended for settings requiring strict single-use semantics and elimination of electrical persistence after a read at the interface.

A. PQC Decapsulation

In ML-KEM decapsulation, the ephemeral private key $\mathbf{s}$ is both large and security-critical. An ENP-P array stores each coefficient s_i under an independent basis b_i, and the decapsulation logic retrieves s_i exactly once. The key never needs to reside in RAM, cache, register files, DMA buffers, or speculative structures, removing the dominant logical persistence channel for PQC implementations and meeting the spirit of NIST's PQC migration guidance [4].

B. Secure Boot and Firmware Verification

Verification secrets used during secure boot must not persist after signature checks. ENP-P cells provide one-shot disclosure of these values and prevent post-boot extraction or replay by compromised firmware or DMA engines, provided the secrets are not copied elsewhere.

C. Enclaves and Attestation

Enclave launch keys and attestation seeds can be provisioned in ENP-P cells, ensuring they are consumed exactly once during enclave initialization and cannot be exfiltrated by microarchitectural leakage or physical probing after launch via standard memory channels.

D. Hardware Security Modules and Secure Elements

HSMs frequently manipulate transient keys whose lifecycle is short. ENP-P semantics allow an HSM to enforce irreversible consumption of internal derivations at the hardware boundary, preventing post-mortem extraction even in the presence of firmware compromise, as long as internal buses do not rematerialize the secret.

XIII. Limitations and Non-GOALS

While ENP-P provides strong guarantees for post-read security at the digital boundary, it does not address all attack vectors.

A. Pre-Read Leakage

ENP-P does not mitigate leakage that occurs *before* the grounding-triggering read. If the value is compromised prior to consumption, ENP-P offers no retroactive protection.

B. Invasive Physical Attacks

Our model assumes no invasive fabrication-level adversary. Focused ion-beam modification, backside laser probing, or delayering may bypass the grounding network by altering transistor-level behavior. We do not claim resistance to such attackers in this work.

C. Long-Term Storage

ENP-P is not intended for long-term non-volatile key storage. It complements but does not replace secure NVM, TPMs, or fuses.

D. Deployment Constraints

True grounding semantics require ASIC implementation. FPGA prototypes approximate grounding logically but cannot reproduce physical discharge behavior or analog effects.

E. Side-Channel Leakage During Collapse

We do not claim to eliminate power or EM side-channels during the read/grounding event itself. ENP-P guarantees that no *post-read* electrical state persists in the modeled design, but an attacker who can observe the instantaneous analog behavior of the collapse may still learn partial information about the secret. Conventional masking and implementation-level side-channel countermeasures remain necessary at that boundary and are complementary to the non-persistence guarantees provided by the primitive.

XIV. Related Work

A. Cold-Boot and Memory Erasure

Cold-boot mitigation techniques target DRAM decay but do not eliminate electrical persistence or microarchitectural shadows. Halderman *et al.* gave the classic demonstration of cold-boot attacks on encryption keys, showing practical key recovery from DRAM images after power loss [1]. Yitbarek *et al.* later showed that memory scramblers and modern DRAM controllers do not fundamentally prevent such attacks [2]. ENP-P instead removes the underlying persistent state at the primitive boundary, rather than relying on scrambling or best-effort erasure.

B. Amnesiac and Self-Destructive Memories

Farheen *et al.* propose Amnesiac Memory, a self-destructive polymorphic mechanism that erases memory under certain environmental conditions to resist cold-boot attacks [3]. Their design emphasizes environment-triggered destruction and polymorphism, while ENP-P focuses on *use-triggered* single-read semantics and monotonic grounding at the circuit level in the intended ASIC.

C. On-Chip SRAM and Power-Domain Attacks

Mahmod and Hicks show that on-chip SRAM and register-like structures are vulnerable to power-domain separation and “freezing” attacks, exposing secrets thought to be safely held in logic [6]. These results motivate primitives like ENP-P that ensure secrets do not persist electrically even in logic-based storage.

D. Hardware Security Guidance and Failure Scenarios

NIST SP 1800–38 provides practical guidance on PQC migration and emphasizes the importance of protecting ephemeral private keys in memory and hardware [4]. NIST IR 8517 catalogs hardware security failure scenarios, including those stemming from remanent states, microarchitectural shadows, and incomplete erasure [5]. ENP-P can be viewed as a primitive-level response to this class of failures.

E. PUFs, One-Time Memories, and Countermeasures

Physical Unclonable Functions derive keys but do not guarantee post-read non-persistence. Prior one-time memory designs rely on e-fuses, antifuses, or flash-based destructive writes—none provide single-cycle CMOS grounding behavior at the interface. Masking, fault sensors, dual-rail logic, and redundancy protect against specific classes of attacks but do not guarantee that a value ceases to exist electrically after use.

F. Positioning ENP-P

The ENP-P differs fundamentally: it provides a CMOS-grounded primitive that eliminates electrical persistence and enforces true single-use semantics at the digital boundary, directly addressing the cause—not the symptoms—of remanence in the modeled design.

XV. Artifact Availability

To support reproducibility of the FPGA evaluation in Section XI, we provide a minimal hardware artifact containing the exact RTL and bitstream used to obtain the functional, timing, and resource-utilization results reported in this paper, together with System Console scripts for exercising single-disclosure and post-consumption inertness.

The artifact is publicly available at:

https://github.com/fcunnane/ENP-P

A. Contents

The repository contains the following components:

- `collapse_cell.sv` — the 256-bit ENP-P collapse cell implementing basis-gated single-use disclosure and irreversible destruction in the modeled design;
- `collapse_bank.sv` — a 64-cell array wrapper with address decode and collapse control;
- `256.sof` — the Quartus Prime 25.1 FPGA bitstream (Cyclone V 5CSEBA6U23I7) corresponding exactly to Tables II–IV;
- `scripts/` — System Console TCL scripts for deterministic testing of correct-basis reads, wrong-basis reads, and multi-read inertness.

The FPGA bitstream and scripts require no additional build steps.

B. Scope and Limitations

The artifact enforces the logical semantics of ENP-P (single-disclosure and irreversible consumption at the interface). It does not emulate ASIC-level analog grounding dynamics or discharge physics.

C. Licensing

The artifact is released for academic use only. Commercial licensing for ASIC instantiation or incorporation into commercial systems is available from QSymbolic LLC.

XVI. Conclusion

Ephemeral secrets routinely outlive their intended lifetime in modern systems, persisting in RAM cells, microarchitectural structures, decay paths, and speculative engines. As post-quantum cryptography increases the size and sensitivity of ephemeral keys, this persistent misalignment becomes untenable.

We introduced the Electrical Non-Persistence Primitive (ENP-P), a CMOS-compatible grounding-based mechanism that guarantees, at the modeled digital boundary: (1) a secret is disclosed exactly once under a matching basis; (2) all mismatches yield an inert, ground-referenced logical output; and (3) every read irreversibly grounds the internal encoding in the digital design.

Unlike RAM or flip–flops, ENP-P collapse is modeled as a physical, one-directional electrical event that leaves no recoverable traces of the secret within the synthesized design. We formalized post-read indistinguishability, provided security claims, demonstrated functional FPGA prototypes, and described integration pathways for PQC decapsulation, secure boot, enclaves, and HSMs.

Future work includes ASIC tapeout, formal analog modeling of grounding dynamics, integration into PQC co-processors, and co-design with microarchitectural isolation domains, as well as full physical-security and side-channel evaluation.

References

[1] J. A. Halderman, E. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Calandrino, A. J. Feldman, J. Appelbaum, and E. W. Felten, "Lest We Remember: Cold Boot Attacks on Encryption Keys," in *Proc. USENIX Security Symp.*, 2008.

[2] S. Yitbarek, M. T. Aga, R. Das, T. Austin, and A. Gershman, "Cold Boot Attacks are Still Hot: Security Analysis of Memory Scramblers in Modern Processors," in *Proc. IEEE Int. Symp. High Performance Computer Architecture (HPCA)*, 2017.

[3] T. Farheen, S. Roy, A. Cannon, J. Di, S. Tajik, and D. Forte, "Amnesiac Memory: A Self-Destructive Polymorphic Mechanism Against Cold Boot Data Remanence Attack," in *Proc. Great Lakes Symp. VLSI (GLSVLSI)*, Workshop, 2024.

[4] W. Newhouse, M. Souppaya, W. Barker *et al.*, "Migration to Post-Quantum Cryptography," NIST Special Publication 1800–38, 2023.

[5] P. Mell and I. Bojanova, "Hardware Security Failure Scenarios," NIST Interagency Report 8517, 2024.

[6] J. Mahmod and M. Hicks, "Exploiting Power Domain Separation to Steal On-Chip Secrets: SRAM Has No Chill," in *Proc. 27th Int. Conf. Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, 2022.