The Gate-Level Security Gap: How Analyz-N™ raises the Bar on Security EDA

Silicon Assurance

Ensuring RTL Security Intent Survives Synthesis, Integration, and Silicon Implementation

TABLE OF CONTENTS

INTRODUCTION

Modern IP blocks, such as processors, controllers, bus fabrics, crypto engines, and debug logic, are designed with security intent at the RTL level. However, the true security of an IP block is determined not at RTL, but at the gate-level, where the final silicon-bound logic structure is created.

A critical challenge is that security verified at RTL does **not** guarantee security at the gate-level. During logic synthesis, optimization can fundamentally alter security-critical behavior. Operations such as retiming, logic minimization, gate reordering, and structural reshaping frequently modify privilege enforcement, hardening logic, cryptographic protections, and masking structures. As such, it is essential to verify at the gate-level that security countermeasures have not been degraded or removed during synthesis.

Furthermore, the availability of power and timing information from standard-cell libraries enable estimation of side-channel leakage and fault-injection susceptibility at the gate-level. Although this analysis does not model analog effects such as parasitics or wire coupling, and therefore cannot accurately capture EM leakage, detailed power signatures, or glitch propagation, it enables rapid, broad coverage for early vulnerability detection. In contrast, physical-design-level analysis offers higher accuracy but is significantly slower and is limited in scope.

Analyz-N™, Silicon Assurance's gate-level security EDA platform, is the first automated solution designed specifically to validate IP security post-synthesis. It analyzes gate-level designs, identifies security-relevant assets, generates security assertions, constructs adversarial scenarios, and produces the evidence required for post-synthesis security verification. Analyz-N™ enables teams to verify whether the security intent defined at RTL is preserved beyond synthesis and reflected in the gate-level implementation.

MESSAGE FROM OUR LEADERS

“Secure RTL” is not the same as “secure silicon.”

Security must survive the optimization and structural transformation of the logic synthesis process. Gate-level analysis, in conjunction with RTL verification, ensures robust security of an IP core.

Analyz-N™ performs gate-level analysis to verify that security intent remains intact after logic synthesis. Using automated assertion generation, adversarial analysis, and post-synthesis validation, Analyz-N™ guarantees that hardware IP comes with **provable, measurable security—not assumptions.**

> “**Analyz-N™** shifts hardware security from reactive guesswork to proactive, evidence-backed verification that proves security intent is preserved at multiple abstraction levels.”

Dr. Travis Meade – Head of R&D

Secure IP at RTL ≠ Secure IP at Gate-Level

Even when designed with robust protections, IP that appears secure at RTL can become vulnerable after synthesis. For instance, retiming can reduce the effectiveness of fault-injection countermeasures such as register hardening. Retiming changes the internal structure of a design by moving registers, thereby undermining the effectiveness of countermeasures that rely on redundant instantiation of registers.

Synthesis directives such as dont_touch can mitigate some of these issues, but only if used correctly. The effectiveness of dont_touch depends on how and where it is specified. When applied to a hierarchical module instance within a higher-level design, the directive propagates to all cells in that hierarchy. However, when assigned to the top-level module, it has no effect because the module isn't instantiated inside another. Similarly, applying dont_touch to a net affects only the mapped combinational cells directly connected to that net at the same hierarchy level. The directive does not influence nets with unmapped cells, and using it can interfere with downstream DFT insertions.

As a result, IP blocks that appear secure at RTL often exhibit weakened or compromised behavior after synthesis. Relying solely on RTL security verification is insufficient, as the synthesized gate-level implementation can diverge from the intended design in ways that directly affect security.

01 How Synthesis Breaks RTL Security

Retiming

Moving registers can create glitch windows, expose transient signal values, and disrupt carefully designed security sequences.

Logic Minimization

Optimization may remove “redundant” or “unused” logic, sometimes eliminating deliberate security checks, dummy rounds, masking logic, or power-balancing structures critical for side-channel resistance.

Gate Reordering

Masked crypto or multishare designs depend on strict separation of shares; reordering can combine sensitive signals inside individual gates, creating observable leakage.

Implementation Diversity

Two RTL implementations that are functionally identical may produce drastically different gate-level structures with different leakage profiles. Functional equivalence does **not** imply security equivalence.

02 Security Is Seldom Checked at Gate-Level

A critical, industry-wide issue is that security verification typically stops at RTL.

Once a design enters synthesis, teams focus on timing closure, area/power optimization, ECO rewiring, clock/reset integration, scan insertion, and fabric stitching, **none** of which include gate-level security checks.

Gate-level simulation (GLS), if run at all, is limited to reset sequencing or DFT validation and offers **no coverage** of privilege flows, lifecycle integrity, or information-flow guarantees. As a result, gate-level security is rarely verified, even though this is where security most often breaks.

Because verification ends at RTL, several essential behaviors are never validated post-synthesis:

- Privilege enforcement at the gate-level
- Correctness of cryptographic masking

- FSM security behavior after structural reshaping
- Leakage paths created by reordering and gate-level restructuring

Gate-level is exactly where security breaks, but traditionally, it is never checked.
This gap leads directly to post-silicon security failures, often discovered only after hardware is fabricated, deployed, or attacked.

03 Gate-Level Security EDA: The Missing Layer

Gate-Level Security EDA fills the missing verification layer by validating post-synthesis behavior:

- Ensuring RTL security intent survives gate-level transformation
- Preserving privilege, lifecycle, and cryptographic enforcement
- Detecting gate-level hazards, leakage, and unintended path
- Providing security assertions for both RTL and gate-level
- Delivering audit-ready security evidence for certification

“Analyz-N™ is the industry’s first gate-level security assurance EDA Platform for IP, helping teams assess whether RTL security intent holds after synthesis and into silicon.”

Security-Critical IP Families and How Analyz-N™ Secures Them

Gate-level security validation is essential for all security-relevant IP categories. Even minor synthesis-induced structural shifts can compromise privilege boundaries, lifecycle protections, or cryptographic behavior.

IP Family	Role in Security	Analyz-N™ Enables	Prevents
Bus & Interface IP	AXI/AHB fabrics govern data movement.	• Channel integrity • Correct error-handling • Protocol consistency • Safe state transitions •	Unintended access paths; privilege leaks
Controllers (Lifecycle, OTP, FSMs)	Define secure state behavior.	• Correct lifecycle progression • Proper OTP fuse usage • Valid debug/test entry • Lifecycle-based enable gating	Rollback; provisioning errors; unauthorized debug access
Microcontrollers	Control protected registers.	• Stack isolation • Instruction boundary correctness • Sensitive register protection • Hardened interrupts	Privilege misuse
Processors	Enforce execution domains.	• MPU/MMU partitioning • Secure-mode entry/exit • CSR access rules • Speculative boundary integrity	Domain crossing; privilege escalation
Security IP (Crypto, HMAC, RoT)	Provide security of data.	• Dataflow consistency • FSM-based crypto control • Secure boot anchoring	Leakage; cryptographic misuse
Test & Debug IP	Enables verification of IP blocks.	• Lifecycle-gated debug • Exclusion of secrets from scan • Secure debug unlock • Detection of unsafe modes	Debug backdoors

INSIDE ANALYZ-N™: THE SECURITY EDA WORKFLOW

Analyz-N™ performs structural, behavioral, and adversarial verification of gate-level IP through an automated pipeline.

01 Asset Identification

Keys, fuses, privilege bits, memory regions, configuration registers, FSM states, lifecycle states, secure buffers, and crypto datapaths.

02 Behavior Classification

Leakage, escalation, unauthorized access, incorrect state transitions, zeroization failures, debug bypass, and out-of-bounds access.

03 SytemVerilog Assertion Generation

High-quality SVAs for both RTL and gate-level security intent.

04 SVA Relevance Checking

Reduces false positives by ensuring each assertion is structurally and semantically meaningful.

05 Attack Point Generation

Constructs adversarial scenarios derived from assertions.

06 Testbench Generation

Automatically builds simulation-ready security testbenches.

07 Simulation

Assertions inserted in RTL and analyzed on any commercial simulator for validation

08 Evidence Collection

Logs, waveforms, coverage, assertion outcomes, and security analysis reports for audit-ready sign-off.

OUTCOMES: What Security Verification Teams gain

01 Evidence-based Gate-level Security Assurance for IP

02 Evidence for defense, government, and internal audits

03 Early detection of synthesis-induced vulnerabilities

04 Reduced risk of post-silicon escapes and costly re-spins

05 Standardized, repeatable verification across all security-critical IP

About Silicon Assurance

Silicon Assurance is a hardware security company dedicated to developing EDA software solutions that excel at detecting, assessing, and mitigating security vulnerabilities in silicon chips designed by semiconductor and system companies. Our security assurance platform, Analyz-N, enables design and security verification teams to identify threats at the gate-level of design, ensuring security implementations are preserved post-synthesis.

For more information, go to https://siliconassurance.com/

CONTACT

https://siliconassurance.com
info@siliconassiurance.com
https://www.linkedin.com/company/silicon-assurance

Silicon Assurance

747 SW 2nd Ave, Suite 258, IMB #30,
Gainesville, FL, 32601, USA