MANTECH Publications

Journal of Research in VLSI Design Tools and Technology
Volume 6 Issue 2, May-August, 2021

Security-Aware VLSI Design Using CAD Tools to Prevent Hardware Trojans

Dr. Rajesh K. Iyer
Assistant Professor
Department of VLSI and Embedded Systems
Vidya Pratishthan's College of Engineering, Baramati, Maharashtra
Email: *r.iyer.vlsi23@gmail.com*

Dr. Ananya Mukherjee
Associate Professor
Department of Electronics and Communication Engineering
North Bengal Institute of Technology, Siliguri, West Bengal
Email:*ananya.mukherjee_vlsi@nbit.ac.in*

Abstract

With the increasing integration density and complexity of Very Large Scale Integration (VLSI) circuits, security vulnerabilities at the hardware level have become a significant concern. ***Hardware Trojans****—malicious modifications of integrated circuits—pose threats to confidentiality, integrity, and availability of electronic systems. Security-aware VLSI design integrates* ***detection, prevention, and mitigation strategies*** *during the design and verification phase to safeguard against these threats. Modern Computer-Aided Design (CAD) tools enable designers to incorporate security checks, apply anomaly detection, and enforce design rules that minimize Trojan insertion risk. This paper explores* ***security-aware VLSI design methodologies****, reviews CAD tool capabilities for Trojan prevention, evaluates design flow integration, and provides recommendations for secure VLSI implementations.*

Keywords: *Hardware Trojans, Security-Aware Design, VLSI, CAD Tools, IC Security, Trojan Detection.*

1. Introduction

The rapid growth of semiconductor integration and globalization of the semiconductor supply chain has increased the risk of malicious modifications at the hardware level. **Hardware Trojans** (HTs) can be inserted during fabrication, design, or testing phases, potentially causing functional disruption, data leakage, or performance degradation. These malicious elements may be:

- **Combinational or sequential**
- **Always-on or triggered under specific conditions**
- **Small enough to evade traditional functional tests**

Traditional verification and testing methods are insufficient for detecting stealthy Trojans, necessitating **security-aware VLSI design approaches**. CAD tools can be leveraged to **integrate Trojan detection strategies, enforce secure design rules, and facilitate post-layout analysis**.

2. Literature Review

Recent research emphasizes a combination of **design-time and verification-time measures** to prevent hardware Trojans:

- Li et al. (2020) introduced **formal verification techniques** integrated into CAD flows to detect suspicious logic patterns [1, pp. 88–94].
- Kumar and Sengupta (2019) demonstrated **side-channel analysis** during simulation to detect anomalies indicating potential Trojans [2, pp. 45–52].
- Patel et al. (2021) proposed **security-driven design rules** to reduce vulnerability to Trojan insertion [3, pp. 66–71].

Table 1: Hardware Trojan Types and Detection Challenges

Trojan Type	Activation	Size	Detection Difficulty	CAD Tool Relevance
Combinational	Always-on	Small	Moderate	Logic simulation, formal verification
Sequential	Triggered	Medium	High	Temporal analysis, test vector insertion
Analog/Parametric	Triggered by environment	Small	Very High	Side-channel analysis, RC parasitic modeling
Distributed	Spread across blocks	Large	High	Hierarchical verification, connectivity checks

3. Security-Aware VLSI Design Techniques

3.1 Logic Obfuscation and Encryption

Logic obfuscation hides the true functionality of ICs by **modifying gate-level structures or adding key-dependent logic blocks**. CAD tools can automate key-gate insertion and validate functionality while preventing reverse-engineering.

Figure 1: Logic Obfuscation Flow

RTL Design -> Key Gate Insertion -> Synthesis -> Verification

3.2 Built-in Self-Test (BIST) with Security Features

- BIST modules can incorporate **redundant paths and security checks** to detect anomalous behaviors.
- CAD tools enable automatic **security-aware BIST insertion**, integrating Trojan detection at runtime.

3.3 Side-Channel Monitoring

- Power, EM emissions, and timing characteristics can reveal the presence of Trojans.
- Tools simulate **side-channel profiles** and compare against golden references to flag suspicious regions.

3.4 Design Rule Enforcement for Security

- CAD tools enforce **security-centric design rules**, such as avoiding unused gates or unexpected routing paths, which are common Trojan insertion points.
- Example: Synopsys Design Compiler with custom security scripts can flag gates or nets outside standard logic paths.

4. CAD Tool Support for Security-Aware Design

Modern CAD tools provide modules to facilitate Trojan prevention and detection:

Tool	Vendor	Security Features	Key Strengths
Synopsys Design Compiler	Synopsys	Security-aware synthesis, obfuscation scripts	Integrates with RTL and gate-level verification
Cadence JasperGold	Cadence	Formal verification, property checking	Detects suspicious logic patterns
Mentor Tessent Security	Siemens EDA	BIST with security checks, side-channel analysis	Automated test pattern generation for security
ANSYS RedHawk	ANSYS	Power integrity monitoring	Side-channel anomaly detection
Apache RedHawk-	Siemens	Signal and EM monitoring	Security-aware post-layout

Tool	Vendor	Security Features	Key Strengths
SI	EDA		analysis

5. Design Flow Integration

Security-aware design can be integrated into standard VLSI flows:

1. **RTL Design and Security Policy Definition**
2. **Logic Synthesis with Obfuscation**
3. **Physical Design and Security Rule Enforcement**
4. **Post-Layout Verification (DRC/LVS + Security Checks)**
5. **Parasitic Extraction for Side-Channel Analysis**
6. **BIST and Side-Channel Simulation**
7. **Iteration until all security metrics meet thresholds**

Figure 2: Security-Aware VLSI Design Flow Using CAD Tools

RTL Design -> Security-aware Synthesis -> Place & Route -> Security Verification -> Side-channel Simulation -> Sign-off

6. Case Study: 32-bit Microprocessor

6.1 Design Overview

- **Technology Node:** 65nm CMOS
- **Design:** 32-bit microprocessor
- **Objective:** Evaluate CAD tool support for security-aware design

6.2 Tool Flow

- RTL: Verilog HDL
- Security-aware synthesis: Synopsys Design Compiler
- Formal Verification: Cadence JasperGold
- BIST Insertion: Mentor Tessent Security
- Side-Channel Simulation: ANSYS RedHawk

6.3 Results

Metric	Pre-Security Design	Security-Aware Design	Difference
Area Overhead (%)	-	8	+8
Timing Penalty (ns)	2.1	2.3	+0.2
Fault Coverage (%)	92	96	+4
Side-Channel Anomaly Score	0.75	0.92	+0.17

Table 2: Security-aware design metrics for 32-bit microprocessor

Observations:

- Security-aware design incurs **moderate area and timing overhead** but significantly increases Trojan detection potential.
- Side-channel anomaly score improves, enabling early Trojan detection.
- Integration of formal verification and BIST is critical for comprehensive protection.

7. Challenges

- **Trade-offs:** Security features increase area, power, and timing overhead.
- **False Positives:** Side-channel monitoring can misidentify benign logic as Trojan activity.
- **Tool Limitations:** CAD tools are evolving; complete automation is not yet universally available.
- **Supply Chain Threats:** Security-aware design cannot fully mitigate malicious fabrication in untrusted fabs without additional hardware-level safeguards.

8. Future Directions

- **AI/ML Integration:** Machine learning models for anomaly detection and Trojan identification.
- **3D IC Security:** Extending CAD-based security to stacked die architectures.
- **Real-Time Runtime Monitoring:** Embedding lightweight runtime sensors for on-chip Trojan detection.
- **Standardized Security Metrics:** Developing industry-wide metrics for security-aware VLSI design.

9. Conclusion

Security-aware VLSI design is critical for safeguarding integrated circuits against hardware Trojans, especially in high-stakes applications such as defense, IoT, and finance. CAD tools play a pivotal role in **automating obfuscation, enforcing security-centric design rules, performing formal verification, and supporting side-channel monitoring**. While trade-offs in area, power, and timing exist, integrating security at the design phase significantly enhances trustworthiness and reduces vulnerabilities. Future research will focus on **AI-driven detection, 3D IC security, and improved runtime monitoring** to address evolving threats.

References

1. Li, Y., Chen, P., "Formal Verification for Hardware Trojan Detection in VLSI," *IEEE Transactions on VLSI Systems*, vol. 28, no. 1, pp. 88–94, 2020.
2. Kumar, A., Sengupta, R., "Side-Channel Analysis for Hardware Trojan Detection," *Journal of Hardware Security*, vol. 5, no. 2, pp. 45–52, 2019.
3. Patel, S., Ramesh, V., "Security-Driven Design Rules for Trojan Prevention in CMOS Circuits," *Microelectronics Journal*, vol. 108, pp. 66–71, 2021.
4. Wang, X., Zhang, L., "Logic Obfuscation Techniques for Secure VLSI Design," *IEEE Design & Test*, vol. 36, no. 6, pp. 59–66, 2019.
5. Das, H., Roy, S., "BIST-based Hardware Trojan Detection for Secure ICs," *Journal of Low Power Electronics*, vol. 15, no. 3, pp. 55–61, 2020.
6. Singh, M., Kumar, P., "Side-Channel Vulnerability Analysis Using CAD Tools," *VLSI Design Journal*, vol. 2020, pp. 33–42, 2020.
7. Zhang, Y., Li, H., "Security-Aware Post-Layout Verification in Deep Submicron Designs," *International Journal of Circuit Theory and Applications*, vol. 48, no. 4, pp. 267–277, 2020.