Poster: BlindMarket: A Trustworthy Chip Designs Marketplace for IP Vendors and Users

Zhaoxiang Liu
Kansas State University
Manhattan, Kansas, USA
zxliu@ksu.edu

Ning Luo
UIUC
Champaign, Illinois, USA
nl27@illinois.edu

Samuel Judson
Yale University
New Haven, Connecticut, USA
samuel.judson@yale.edu

Raj Gautam Dutta
Silicon Assurance
Gainesville, Florida, USA
rajgautamdutta@siliconassurance.com

Xiaolong Guo
Kansas State University
Manhattan, Kansas, USA
guoxiaolong@ksu.edu

Mark Santolucito
Barnard College, Columbia University
New York City, New York, USA
msantolu@barnard.edu

Abstract

Due to the globalization of the semiconductor supply chain, chip fabrication now involves multiple parties, including intellectual property (IP) vendors and Electronic Design Automation (EDA) tool vendors. Involving multiple entities and valuable IP naturally raises security and privacy concerns. Various frameworks and tools, such as the IEEE 1735 standard for IP protection, have been developed to mitigate the risk of theft. However, existing solutions fail to address all the threats envisioned by the zero-trust model. We propose a novel zero-trust formal verification framework that requires only two essential parties: IP users and IP vendors. This framework leverages secure multiparty computation to ensure the security and privacy of the hardware verification process. Our proposed solution allows IP users and IP vendors to independently convert the hardware design and assertions into conjunctive normal form (CNF), and then apply privacy-preserving SAT solving to verify the conformance of the design to the specification. This paper introduces a domain-specific secure decision procedure, hw-ppSAT, designed to overcome the scalability challenges of using SAT solving in hardware design verification. Our approach also leverages property-based hardware optimizations and domain-specific heuristics to enhance the verification process. We showcase the framework's effectiveness through its application to several open-source benchmarks.

CCS Concepts

• **Security and privacy** → *Formal security models.*

Keywords

Hardware Security & Piracy; IP Verification; Secure Multiparty Computation

ACM Reference Format:
Zhaoxiang Liu, Ning Luo, Samuel Judson, Raj Gautam Dutta, Xiaolong Guo, and Mark Santolucito. 2024. Poster: BlindMarket: A Trustworthy Chip Designs Marketplace for IP Vendors and Users. In *Proceedings of the 2024 ACM SIGSAC Conference on Computer and Communications Security (CCS '24), October 14–18, 2024, Salt Lake City, UT, USA.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3658644.3691378

CCS '24, October 14–18, 2024, Salt Lake City, UT, USA

ACM ISBN 979-8-4007-0636-3/24/10
https://doi.org/10.1145/3658644.3691378

1 Introduction & Motivation

The broad integration of intellectual property (IP) cores in chip designs has driven the recent semiconductor industry's remarkable growth. According to a 2021 report, ≈ 1.15 trillion devices have been equipped with chips incorporating a variety of IP cores, including soft, firm, and hard variants. Typically, these IP cores are the proprietary intellectual assets of *IP vendors*, who engage in the development of an array of IP cores. Subsequently, entities known as *IP users* purchase these IP cores. These acquired IP cores are integrated into the users' unique system-on-chip (SoC) designs that are taken through the manufacturing process and introduced into the market.

The role of IP cores in the functionality of products developed by IP users is of significant importance, and they are often very valuable. As a result, IP users must ascertain that the IP cores offered by IP vendors align not only with their functional requirements but also with considerations related to durability and safety. This careful *IP verification* process is a crucial step that usually occurs prior to the finalization of commercial agreements between the parties involved. For instance, in the soft IP business process, IP users must review the synthesizable RTL from IP vendors and validate the corresponding functional assertions against this design before finalizing the business agreement.

Motivation: Privacy concerns arise if both parties are mutually distrustful, particularly when IP design and functional requirements are disclosed for verification. *IP users may attempt to extract, reuse, overuse, or modify IP designs without complying with the designated usage rights, potentially jeopardizing the integrity of the IP.* This disclosure can have severe consequences, as evidenced by reported incidents of IP core theft from companies, resulting in substantial financial losses surpassing millions of dollars in revenue [13]. But security and privacy concerns are two-sided. On the other hand, *IP vendors might collect functional requirements from IP users and examine their assertions.* Such technical details themselves are a security and privacy risk, as they can expose the design intentions and architecture of the user's system — which may be of significant value and interest to their competitors. Meanwhile, competitors

who obtain such information might undermine their business models and gain an unfair advantage in the market.

Related Work & Limitation: Researchers have proposed various techniques to address these privacy concerns, though each has its limitations. For instance, IP watermarking [6] is designed for authorship protection instead of privacy, while [7] helps the IP vendor to gain control post-chip fabrication, but without protecting design leakage prior to fabrication. Recent cryptographically-enhanced works [4, 9] either require a trusted third party or limit verification to simulation stages. Presently, the industry relies on non-disclosure agreements and other legal mechanisms to mitigate the risks of IP theft and the improper utilization of IP designs. However, this approach requires significant human resources and still carries inherent risks. In 2014, the IEEE Design Automation Standards Committee proposed the IEEE 1735 standard to promote IP protection. This standard has been widely adapted by major IP market participants, such as AMD, Intel, etc. However, the standard assumes that the EDA tool developers and IP vendors are trusted parties, with only IP users considered as potential attackers attempting to exploit the IP design. Such an assumption is overly strong and could enable those other parties, if malicious, to attack unencumbered. Our work supports zero-trust formal logic-based hardware verification without relying on a trusted party.

Contribution: To establish a more secure and private IP marketplace, this paper proposes a zero-trust hardware verification framework, we summarize our contribution as follows:

1. A zero-trust architecture for 3P/COTS IP verification: we design an architecture in BlindMarket for 3P/COTS IP verification under zero-trust. BlindMarket addresses the limitations of existing tools by adopting a zero-trust model, eliminating the need for third-party involvement and ensuring that no party has to trust another or share confidential information.

2. An implementation of BlindMarket for the hardware domain using cryptographically-secure SAT solving: we securely automate the entire zero-trust verification procedure. We develop a specialized parser to translate Register-Transfer Level (RTL) Intellectual Property (IP), specifically soft IP designs and functional requirements, into SAT representations in Conjunctive Normal Form (CNF). Furthermore, we improve verification speed by incorporating property-based design optimization techniques. We also introduce a hardware-oriented, privacy-preserving SAT solver variant, `hw-ppSAT`, which is a hardware-oriented enhancement of `ppSAT`.

3. Benchmarking with various real-world IP designs and assertions: We benchmark the proposed framework using open-source IP cores and assertions, demonstrating the readiness of our tool for deployment in real-world IP verification tasks.

2 Overview & Approach

We propose BlindMarket, a zero trust verification framework to address the privacy concern in the IP market. Figure 1 depicts the top-level workflow, illustrating the process whereby IP users seek to purchase IP designs from IP vendors. Before making a purchase, IP users need to ensure the target design offered by IP vendors satisfies functional correctness as per the specified assertions. To prevent information leakage, both parties need to follow BlindMarket protocol.

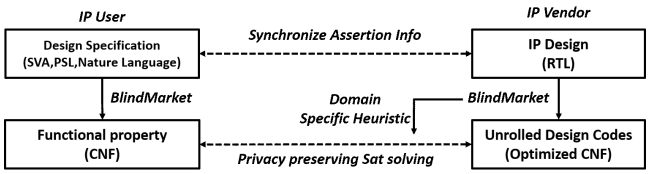

Figure 1: BlindMarket's system overview: 1)IP users synchronize signal information with IP vendors 2)Both generate CNF locally 3)Both jointly compute satisfaction results.

The protocol begins with assertion information synchronization, i.e., IP users announce the candidate signal set $\mathbb{A}$ (the group of assertion variable names) and the time bound b to IP vendors. Meanwhile, on the IP users' side, the design specifications expressed in SystemVerilog Assertion (SVA) are processed by BlindMarket into Conjunctive Normal Form (CNF). The functional property in CNF is the IP users' final input for `hw-ppSAT`.

On the IP Vendors' side, upon reception of $\mathbb{A}$ and b. the candidate signal set $\mathbb{A}$ is used for RTL pruning. Bound b is used for design unrolling. The control signal set $\mathbb{C}$ is extracted from the RTL, the heuristic guidance for the `hw-ppSAT`. Then, the design is translated into CNF with the bound of b using *Bit-blasting* and *Tseitin transformation*. The design in CNF and control signal set $\mathbb{C}$ is the final input to (`hw-ppSAT`) from IP vendors' side. With the provided design CNF, property CNF, and heuristic input $\mathbb{C}$, SAT solver (`hw-ppSAT`) conducts a two-party computation to accomplish the design verification. The verification result is announced to both parties without sensitive information leakage.

Assertion-guided RTL Pruning: Design Prunning is an optimization technique when analysis focuses on a specific component or aspects of a system. In the hardware domain, RTL Prunning is particularly beneficial for resource-intensive analysis like formal verification as the irrelevant state variables and transition function are excluded in data encoding. BlindMarket utilizes Verilog parser [11] to translate RTL into Abstract Syntax Tree (AST), and then traverses this AST to generate a directed cyclic graph(DAG). Based on the given assertion signal set $\mathbb{A}$, a backward propagation starting from a signal $a \in \mathbb{A}$ gathers all dependent signals, representing the minimal design logic associated with the given assertion. This pruned design is then translated into CNF for verification.

Hardware Privacy Preserving SAT Solving: The `ppSAT` protocol [8] solves the two-party SAT-solving problem by taking the classic DPLL decision procedure for Boolean propositional logic [2, 3] and lifting it into a secure protocol through a data-oblivious reformulation of the core reasoning loop compiled using boolean garbled circuits. To extend the concept of oblivious DPLL, our `hw-ppSAT` confines the search set of the decision step of DPLL to a specific subset of hardware signals, namely, the control signals. `hw-ppSAT` receives a ranked literal list from IP vendors as a hint in the DPLL decision routine. As a deterministic method, our experiment shows it is better than `ppSAT` in the hardware domain.

RTL	Source	Bound	Design			Design_Prune				Result
			Var	Cls	DLIS	Var	Cls	DLIS	HW	
b01	[10]	4	255	994	2834.4s*	210	762	**1815.8s***	**56.8s**	UNSAT
b02-1	[10]	3	162	565	24.2s	143	478	**16.6s**	**11.3s**	UNSAT
b02-2	[10]	3	161	570	35.9s	161	570	36.9s	37.8s	UNSAT
b02-3	[10]	3	159	570	45.3s	159	570	45.1s	**26.4s**	UNSAT
b03	[10]	1	226	760	25.9s	226	760	25.7s	**16.7s**	UNSAT
Rrobin	[10]	1	38	77	0.6s	38	77	0.7s	**0.6s**	UNSAT
Prep3	[12]	3	875	2617	25193.8s*	770	2392	**20263.1s***	**11174.6s***	N/A
RS232	[5]	5	1486	5200	>8hrs*	224	708	**84.2s**	**48.9s**	SAT

* timeout

Table 1: Evaluation of zero trust verification framework.

In the hardware domain, control signals uniquely determine signal propagation, which in turn defines the hardware microarchitecture state, inspired by control flow guided SMT solving method[1]. We leverage control flow in hardware to significantly reduce the search space in the DPLL decision step, resulting in a smaller total time cost in oblivious DPLL. We conduct control signal extraction on RTL by maintaining a set $\mathbb{C}$ to store the control signal with its rank. A higher rank of the control signal indicates a deeper level of the control signal within the circuit. Our heuristic picks the lowest rank signal first as it dominates the nested control signals which can be excluded from consideration[1].

3 Evaluation & Conclusion

Table 1 displays the experimental results obtained using BlindMarket to verify the IP designs and associated assertions sourced from Opencore [12], ANSI-C benchmarks [10] and TrustHub [5]. We list the RTL design and its corresponding source in **Source** column. The **Bound** *b* is the estimated clock period sent by IP users to guide the circuit unrolling.

We evaluate our framework in three cases: the plain design (column **Design**) with ppSAT in-built DLIS heuristics, the pruned design(column **Desing_Prune**) via RTL Pruning with in-built DLIS heuristics, and the pruned design with hardware domain-specific heuristic(column **HW**). For each method, we record its time cost and the timeout limit is set to 5000 iterations.

Design with DLIS vs Design_Prune with DLIS Assertion-based optimization techniques result in a time reduction of 4/8 design scales. The other four remain unchanged due to the tight coupling between hardware and its assertions. This optimization method yields significant improvement as the data-oblivious algorithm forces a linear scan on each literal and each clause in its DPLL subroutine. The structural pruning on RTL could remove redundant design so the generated CNF is smaller than the original. Experiment on designs *b01* and *Prep3* are timed out because the pruned design is still too large. However, the time cost for reaching the timeout limit is still quicker than the original method. As a result, the loosely connected hardware design, assertion-based pruning is a deterministic optimization method.

Design Prune with DLIS vs Design Prune with HW By applying hardware heuristic to the pruned design, we observe improvement on 5/8 design. In every oblivious DPLL decision step, the search scope of the candidate signals is reduced from the entire available literals to the control signals. However, this improvement comes with a tradeoff-making the control signal set an external input to hw-ppSAT could potentially expose the control signals to the IP vendors. Precisely, IP vendors expose the number of control signal bits to IP users. Because from IP users' side, IP vendors input some random numbers into hw-ppSAT. hw-ppSAT itself runs the oblivious scan on this set of control signals, but the underlying hardware structure remains undisclosed.

Our experiment shows that with the help of high-level abstraction optimization and feature extraction, BlindMarket enables to address of various IP verification problems in a privacy-preserving way.

Acknowledgements

This work was partially supported by National Science Foundation (NSF-2419880, NSF-2340949).

References

[1] Jianhui Chen and Fei He. 2018. Control flow-guided SMT solving for program verification. In *Proceedings of the 33rd ACM/IEEE International Conference on Automated Software Engineering*. 351–361.
[2] Martin Davis, George Logemann, and Donald Loveland. 1962. A Machine Program for Theorem-Proving. *Commun. ACM* 5, 7 (1962), 394–397.
[3] Martin Davis and Hilary Putnam. 1960. A Computing Procedure for Quantification Theory. *Journal of the ACM (JACM)* 7, 3 (1960), 201–215.
[4] Mohammad Hashemi, Steffi Roy, Fatemeh Ganji, and Domenic Forte. 2022. Garbled EDA: Privacy Preserving Electronic Design Automation. In *2022 IEEE/ACM International Conference On Computer Aided Design (ICCAD '22)*. 1–9. To appear..
[5] Mark Tehranipoor Hassan Salmani. [n. d.]. Trusthub. https://www.trust-hub.org/.
[6] Andrew B Kahng and et al. 2001. Constraint-based Watermarking Techniques for Design IP Protection. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* 20, 10 (2001), 1236–1252.
[7] Hadi Mardani Kamali, Kimia Zamiri Azar, Farimah Farahmandi, and Mark Tehranipoor. 2022. Advances in Logic Locking: Past, Present, and Prospects. *IACR Cryptol. ePrint Arch.* (2022).
[8] Ning Luo, Samuel Judson, Timos Antonopoulos, Ruzica Piskac, and Xiao Wang. 2022. ppSAT: Towards Two-Party Private SAT Solving. In *31st USENIX Security Symposium (USENIX Security 22)*. 2983–3000.
[9] Dimitris Mouris and Nektarios Georgios Tsoutsos. 2020. Pythia: Intellectual Property Verification in Zero-Knowledge. In *2020 57th ACM/IEEE Design Automation Conference (DAC '20)*. 1–6. https://doi.org/10.1109/DAC18072.2020.9218639
[10] Rajdeep Mukherjee, Michael Tautschnig, and Daniel Kroening. 2016. v2c – A Verilog to C Translator Tool . In *Tools and Algorithms for the Construction and Analysis of Systems (TACAS '16)*. Springer, 580–586.
[11] Shinya Takamaeda-Yamazaki. 2015. Pyverilog: A Python-based Hardware Design Processing Toolkit for Verilog HDL. In *Applied Reconfigurable Computing: 11th International Symposium, ARC 2015, Bochum, Germany, April 13-17, 2015, Proceedings 11*. Springer, 451–460.
[12] Various Contributors. 2021. OpenCores.org.
[13] Debby Wu. 2020. Engineers Found Guilty of Stealing Micron Secrets for China. Bloomberg.