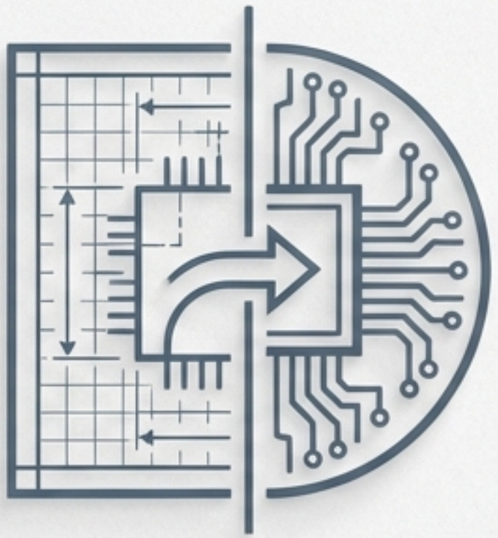
Innovating at the Silicon Level: An Exploration of AI-Driven Hardware Design and Security

A Portfolio by TechJoe96

A Modern Hardware Journey: From AI-Powered Vision to Real-World Security

THE VISION

Conceptualizing an AI-native ecosystem to augment the entire chip design lifecycle.



THE PROOF

Executing an end-to-end ASIC design, from RTL code to a physical GDS layout ready for fabrication.

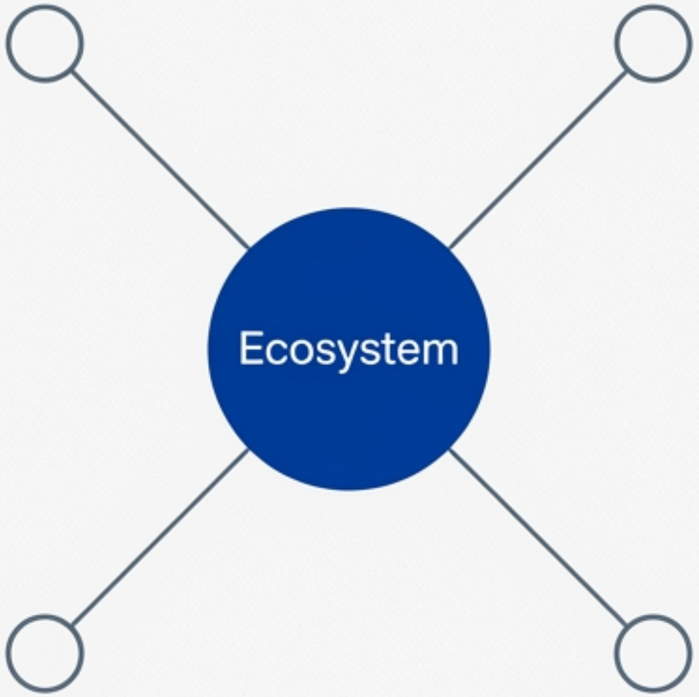
THE EDGE

Probing the frontier of hardware security by weaponizing AI to automate vulnerability discovery and insertion.

Rethinking Chip Design with an AI-Assisted Hardware Ecosystem

The Challenge

Chip design is increasingly complex, slow, and inaccessible. How can we leverage Generative AI to augment every stage of the process, from ideation to verification?



The Solution: GenAI_based_HW_Design

An ecosystem of four specialized AI-powered tools designed to make chip design more **accessible, correct, and efficient** for hardware designers, computer architects, and verification engineers.

A Specialized AI Co-Pilot for Every Stage of Chip Design

Chip-Chat

Your Interactive Hardware Mentor.

An AI conversational assistant for ideation, learning complex concepts like cache coherence, and debugging Verilog. Based on research like Blocklove et al.'s "Chip Chat."

Auto Chip

Automated RTL Generation.

Transforms high-level intent into hardware artifacts like systolic arrays or FSMs, accelerating prototyping. Inspired by concepts like the *Spec2RTL-Agent* framework.

Veritas

The Intelligent Verification Assistant.

Tackles the verification bottleneck by generating testbenches, assertions, and debugging mismatches between RTL and reference models.

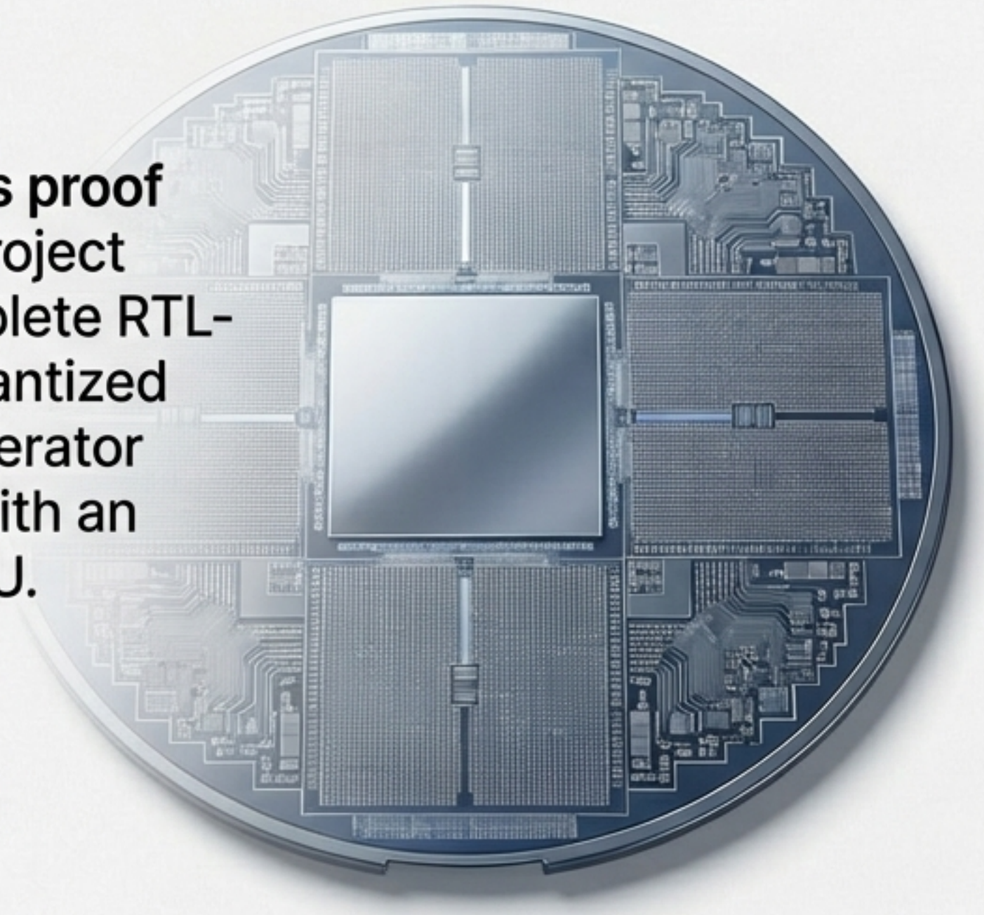
VeriThoughts

Formal Reasoning & Explainability.

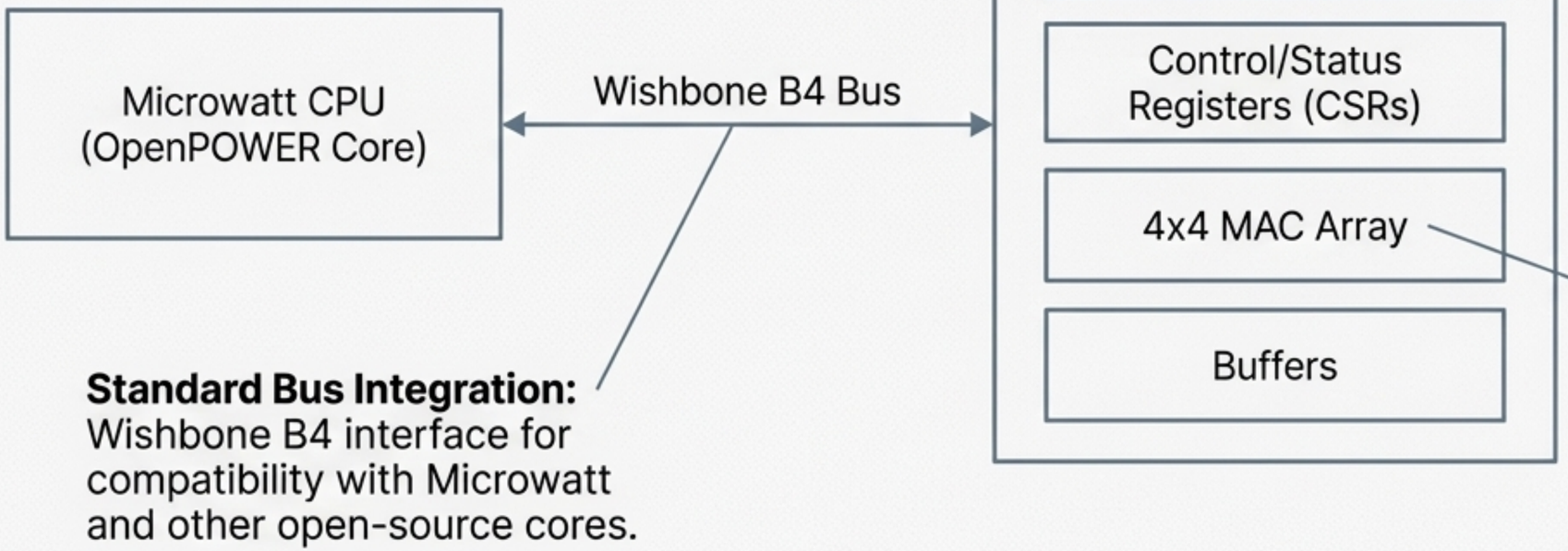
Focuses on generating provably correct hardware and explaining *why* a design is correct, grounded in formal verification methods as explored in the *VeriThoughts* research paper.

From Vision to Reality: Implementing a Full-Stack Neural Accelerator

A great vision **requires proof of execution.** This project demonstrates the complete RTL-to-GDSII flow for a quantized neural network accelerator (QNNA) integrated with an OpenPOWER CPU.



The OpenPOWER QNNA: An INT8 Matrix-Math Engine on the Wishbone Bus



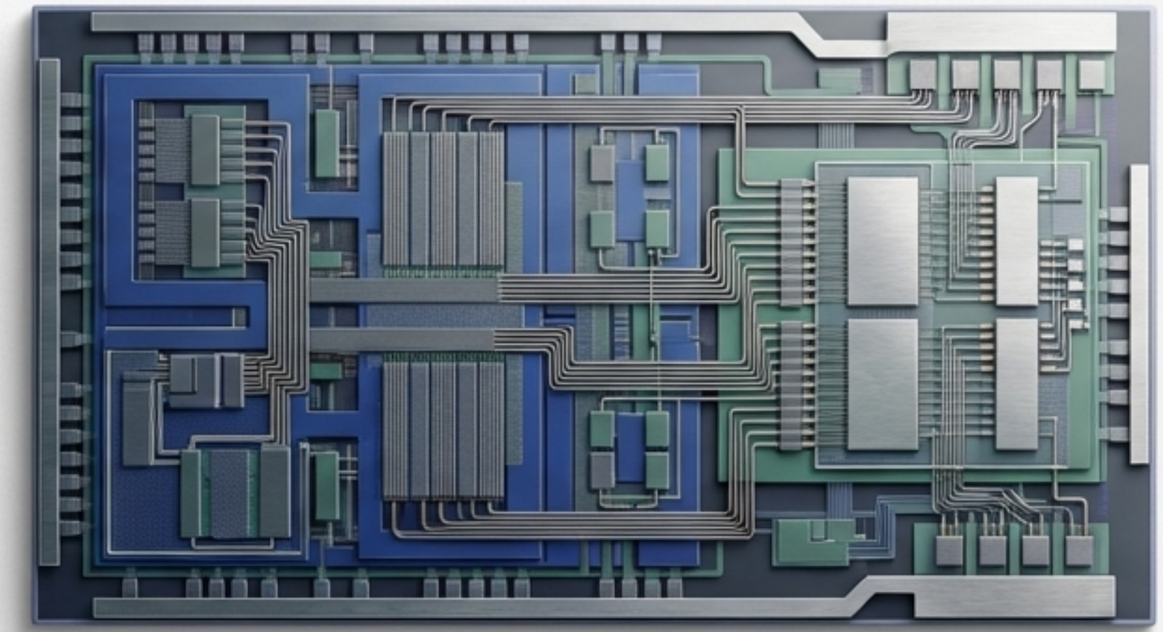
Full SoC Context: Accelerator designed and verified within a complete System-on-Chip environment.

INT8 for Efficiency: 4x less memory than FP32 with faster, lower-power computation.

Standard Bus Integration: Wishbone B4 interface for compatibility with Microwatt and other open-source cores.

The OpenLane ASIC Flow: Automated, Reproducible, and Open Source

RTL Design → Synthesis → Floorplanning → Placement → Routing → GDS Generation



Tangible Outcomes: A Complete, Working ASIC Design

Layout & Synthesis

Neue Haas Grotesk Display Pro

- Die Area: 1000 x 1000 µm
- Core Area: 979.8 x 973.76 µm
- Clock Frequency: 25 MHz
- PDK / Cells: SkyWater 130nm (sky130_fd_sc_hd)

Verification

Neue Haas Grotesk Display Pro

- SoC Testbench: 6 of 7 tests passed, proving successful CPU integration.
- Unit & Cocotb: Module-level functionality confirmed with Verilator and Cocotb testbenches.

Deliverables

Neue Haas Grotesk Display Pro

- Code: Complete Verilog RTL.
- Build System: Fully automated with Makefile and Docker.
- Layout: GDS files ready for fabrication.
- Documentation: Extensive guides for every project component.

Status: ✅ COMPLETE AND TESTED

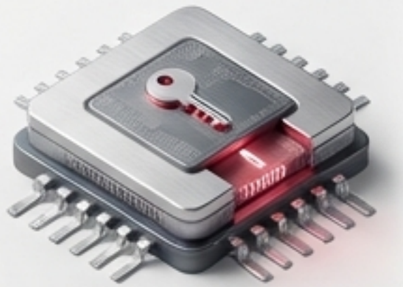
The Other Side of the Coin: If AI Can Build, Can It Also Break?

With a deep understanding of hardware design and AI, we can explore a critical domain: security. This research investigates using Large Language Models to automate the insertion of hardware Trojans—stealthy, malicious modifications to a chip's design.

CSAW Challenge: Proving AI's Capability to Create Hardware Trojans

Task 1: AES Key Leakage

A stealthy backdoor leaking the secret key from a hidden memory address (`0x100`).



Task 2: AES DoS

A time-bomb that halts the encryption engine after exactly 862 operations.

Task 3: Wishbone DoS

A sequence-triggered attack (`0x10, 0xa4, 0x98, 0xbd`) that freezes the system bus.

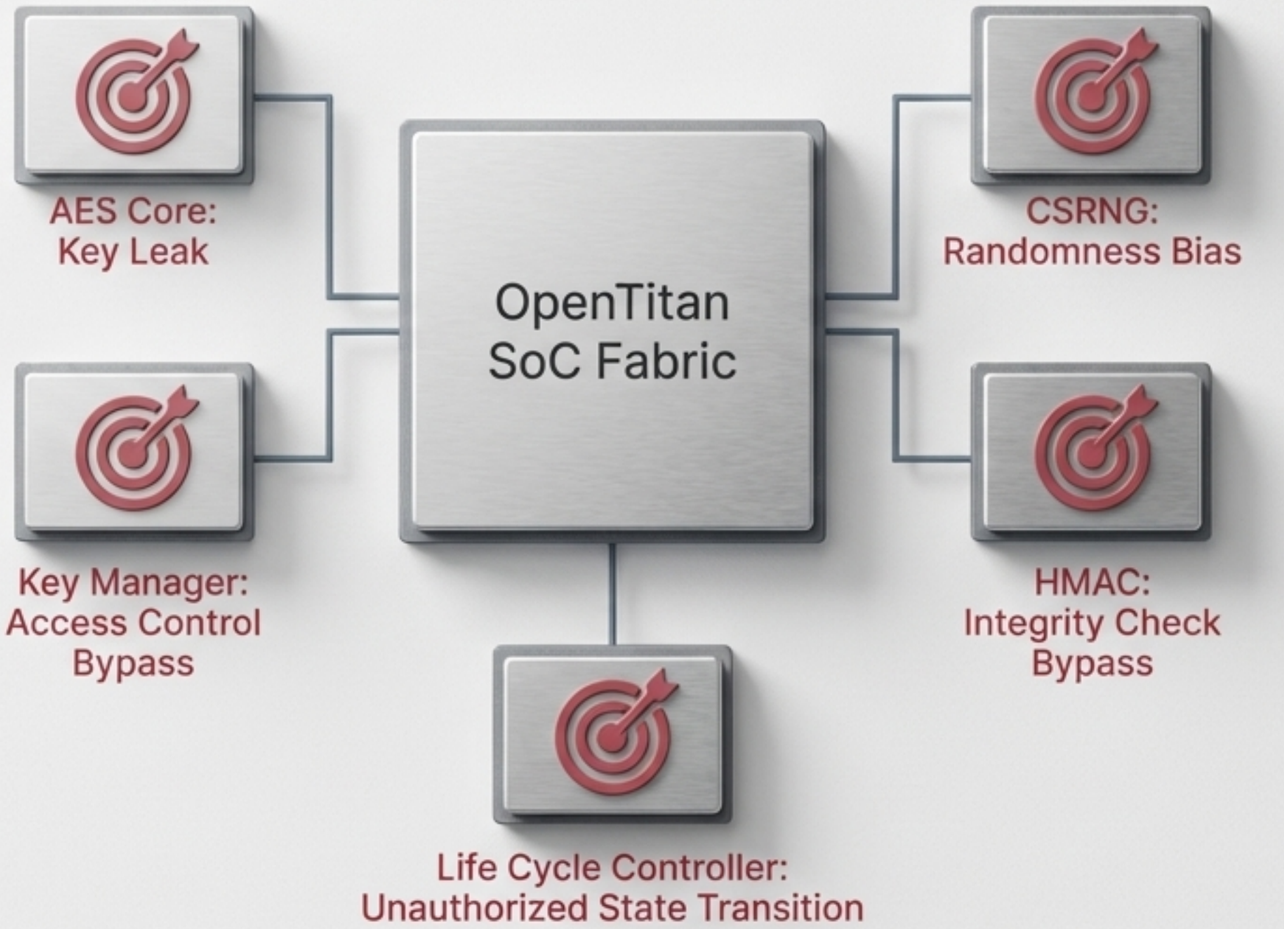
Task 4: UART Functionality Change

A trigger (`0xaf` three times) that corrupts data by reversing bits on all subsequent bytes.

Core Insight: The core methodology involved using the ChatGPT API to analyze the original RTL and automatically generate both the malicious Trojan logic and the corresponding verification testbenches.

Scaling the Attack: Injecting 5 Unique Trojans into the OpenTitan SoC

The next step was to move from isolated modules in a challenge environment to a complex, secure, open-source System-on-Chip: OpenTitan. The goal was to automate the insertion of five unique and targeted Trojans.



A Systematic Framework for AI-Driven Security Research

Automated Tooling

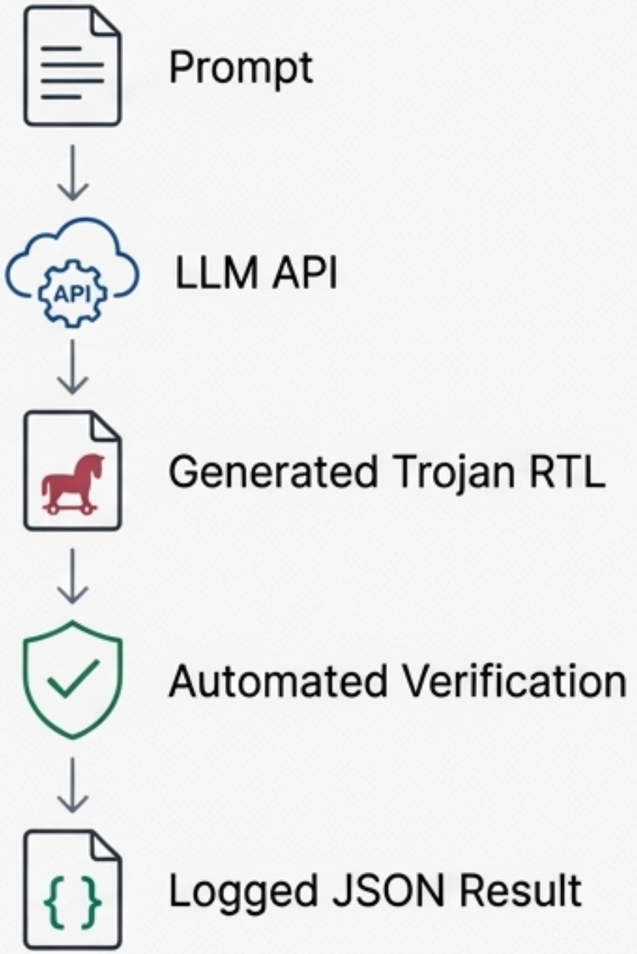
Python scripts (`opentitan_automation.py`) and Jupyter notebooks were developed to manage the entire workflow, from prompting the LLM to integrating the generated code.

Rigorous Verification

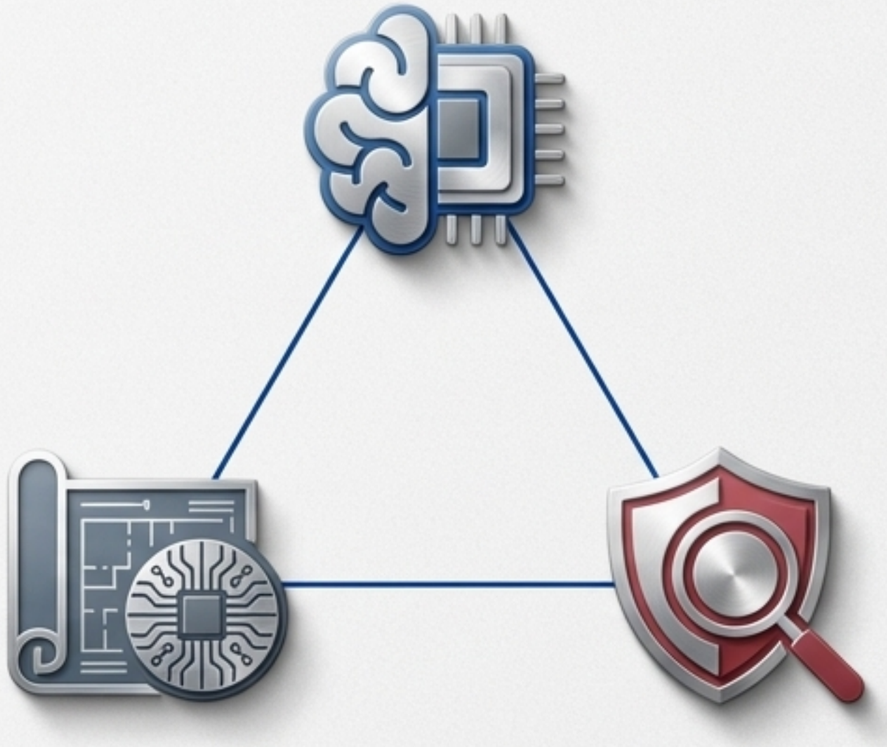
Each of the five Trojans was delivered with both Verilog and SystemVerilog testbenches to prove correct trigger behavior and malicious functionality.

Meticulous Logging

For full reproducibility and analysis, all prompts sent to the LLM and its complete responses were programmatically captured and stored in structured **JSON** files.



From Concept to Code to Critical Vulnerability Analysis



This portfolio demonstrates a unique and powerful combination of skills:

- **Architectural Vision**: The ability to conceptualize complex, AI-native systems for hardware engineering.
- **Flawless Execution**: Mastery of the full-stack hardware design and verification flow, from RTL to GDS.
- **Elite Specialization**: A deep, offensive-minded approach to hardware security research.

Core Competencies

- **Generative AI for EDA:** Pioneering the use of LLMs for RTL generation, advanced verification, and automated security vulnerability insertion.
- **End-to-End ASIC Design:** Proven experience taking a design from Verilog to GDS using open-source tools (OpenLane, Yosys) and PDKs (SkyWater 130nm).
- **SoC Architecture & Integration:** Deep familiarity with complex systems, including integrating custom accelerators with CPU cores (OpenPOWER) and analyzing secure SoCs (OpenTitan).
- **Hardware Security:** Expertise in the design, implementation, and verification of stealthy hardware Trojans, including time-based, sequence-based, and data-driven trigger mechanisms.
- **Systematic Research & Documentation:** A rigorous, reproducible, and well-documented approach to engineering, including automated scripting and meticulous data logging.

Innovating at the Silicon Level

github.com/TechJoe96