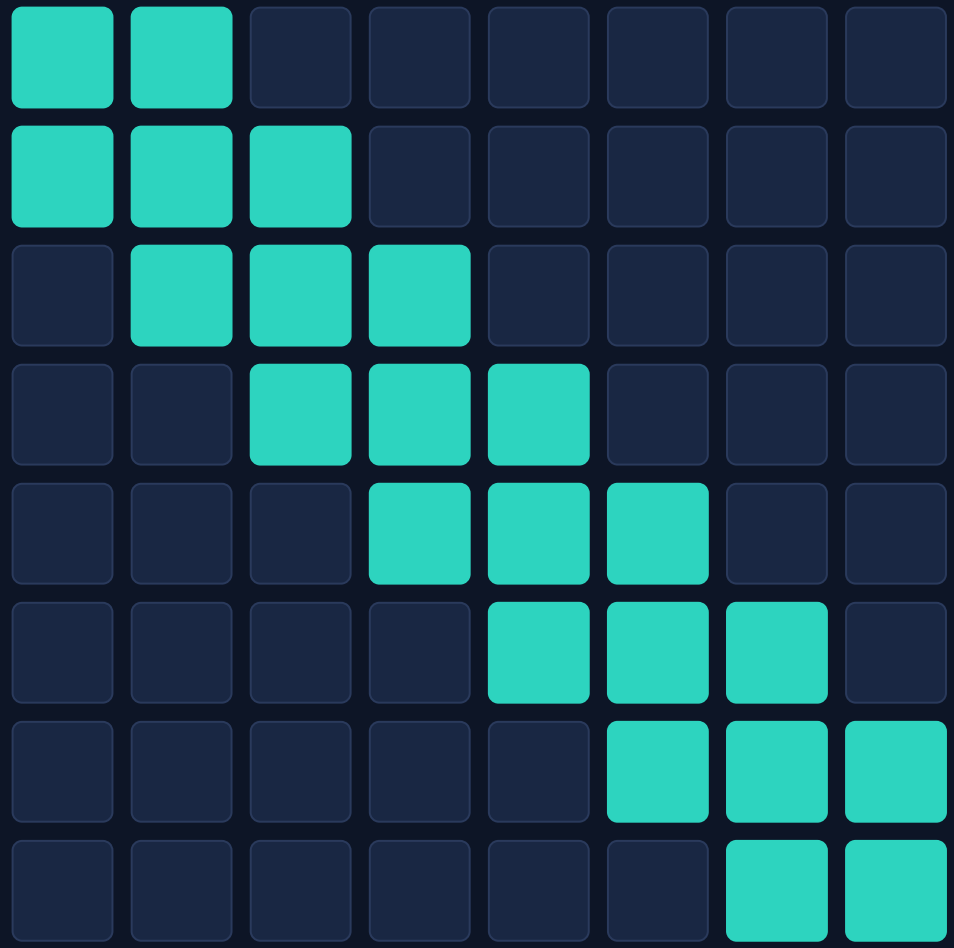
INFERENCE-TIME COMPUTE HACKATHON

Fairchild

Devin For Chip Design



Motivation

- Current LLMs, while powerful, often struggle with the rigorous syntax and logical precision required for Verilog/RTL.
- Chasing subtle timing bugs and race conditions leads to significant "token spend" and developer fatigue in manual debugging loops.
- There is a critical need for an agentic system that automates verification and repair.
- Closing the loop between generation and simulation is the only way to achieve production-grade hardware design.
- **On-premise deployment remains the primary choice for semiconductor teams to maintain IP security and control.**

What Did We Build?

The Vision

A fully integrated IDE

Under the Hood

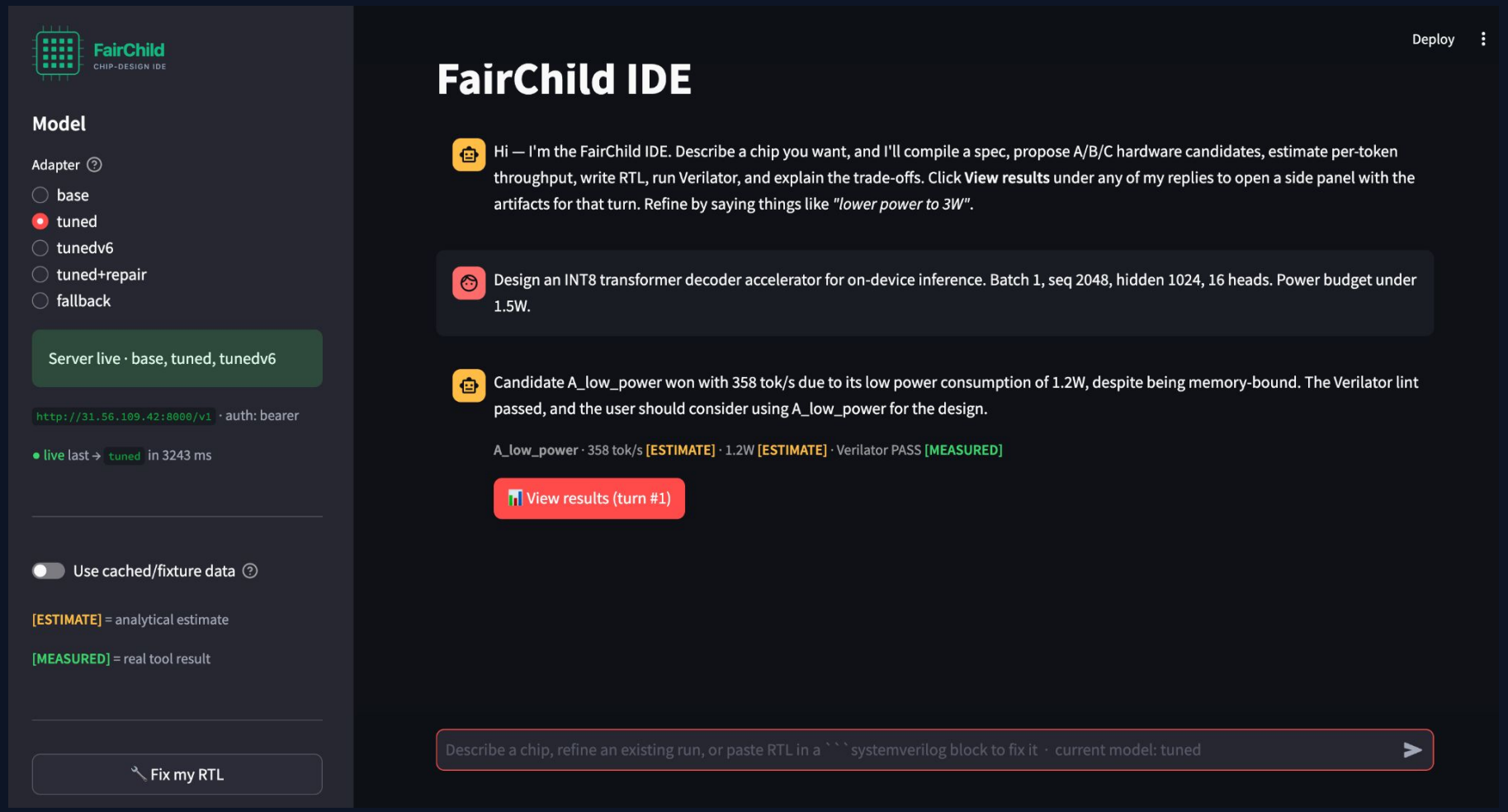
- Hackathon Scope: **RTL Generation + Verification**
- **Qwen 2.5 7B**
  Post Trained RTL Coder
- **Synthetic RTL SFT**
  Advanced Data Mix
- **Agentic Harness**
  Generation and Verification

Closing the loop between hardware design and automated simulation.

What Did We Build?

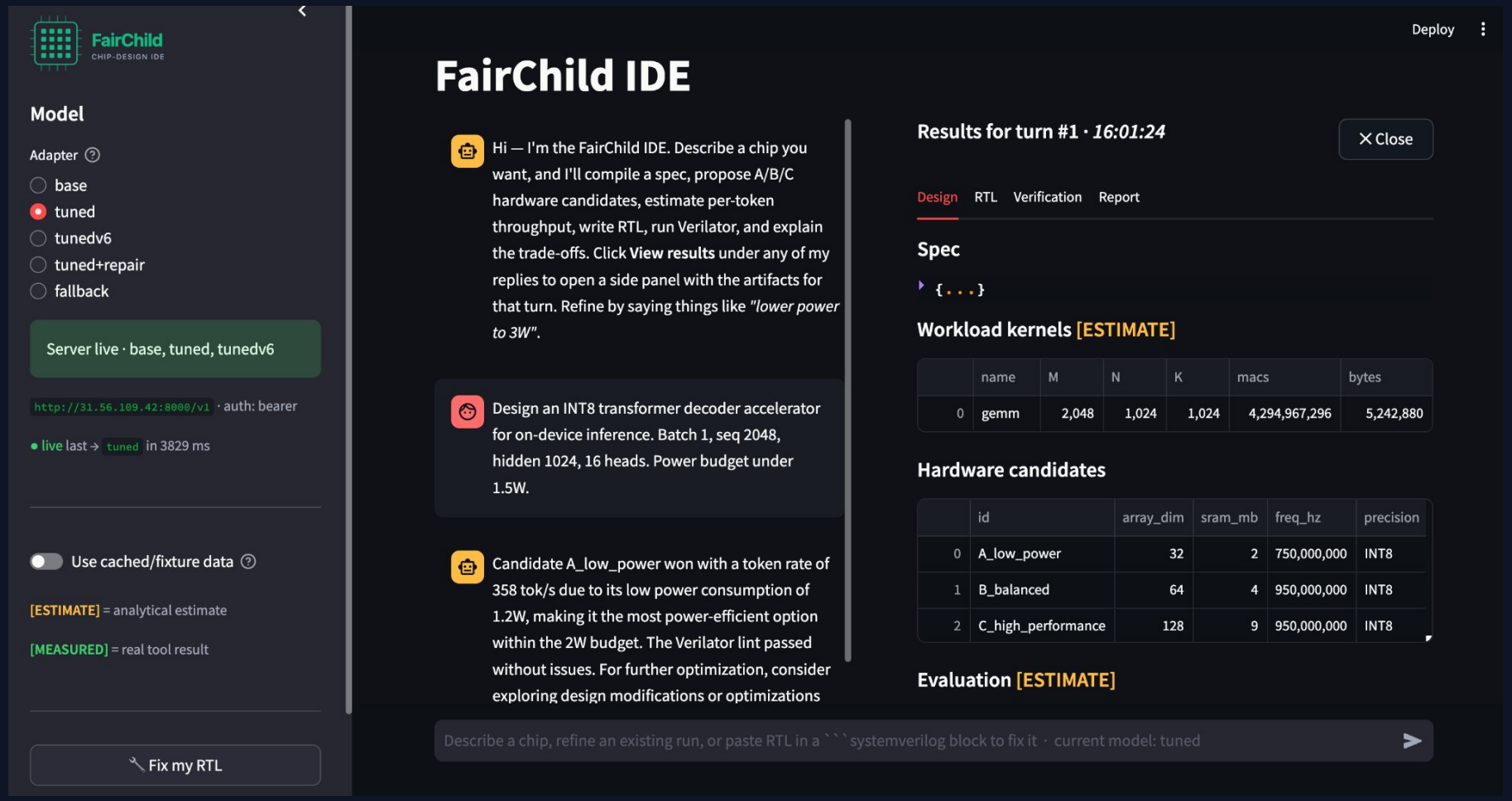
Closing the loop between hardware design and automated simulation.

What Did We Build?



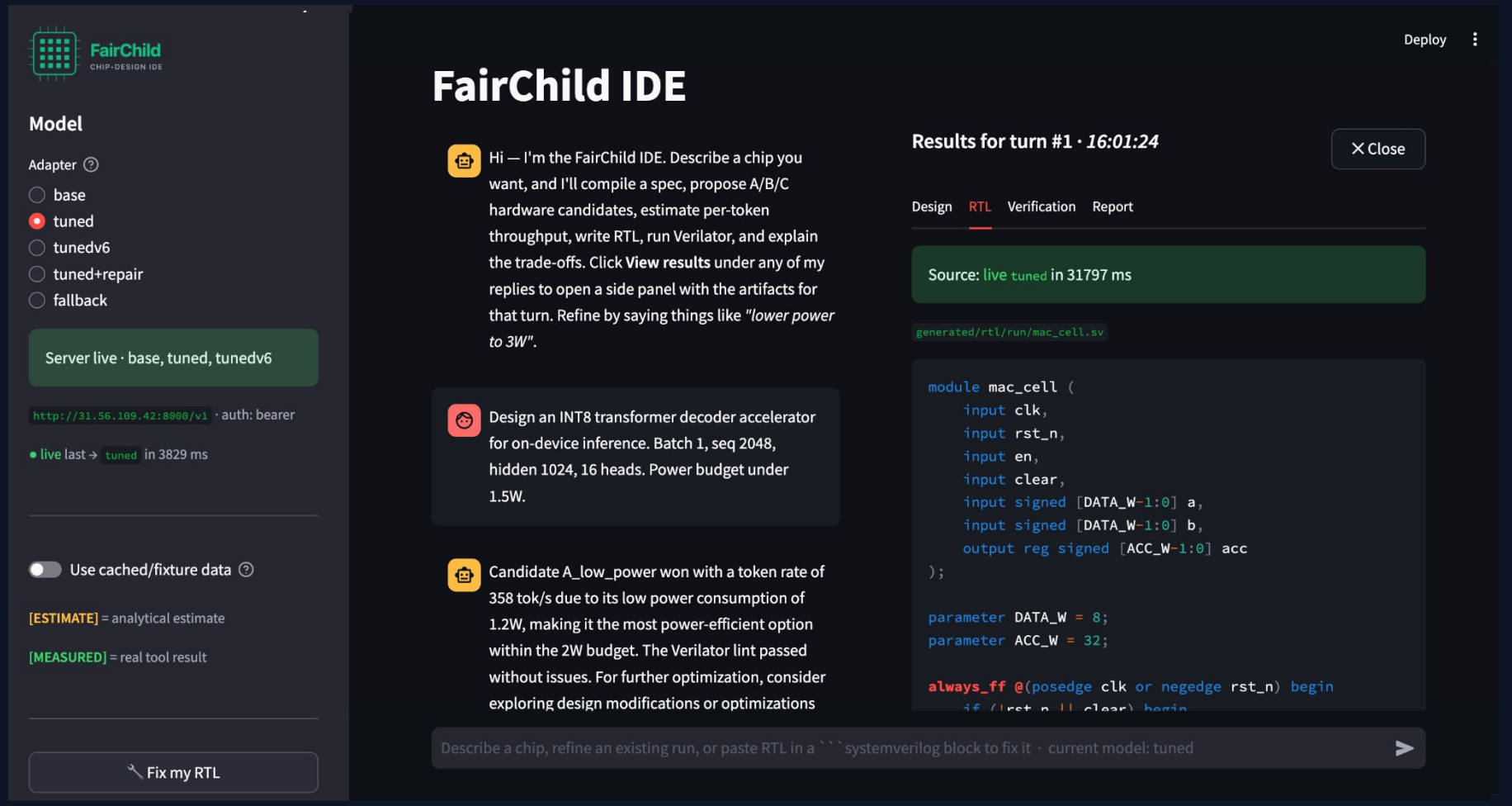
Closing the loop between hardware design and automated simulation.

What Did We Build?



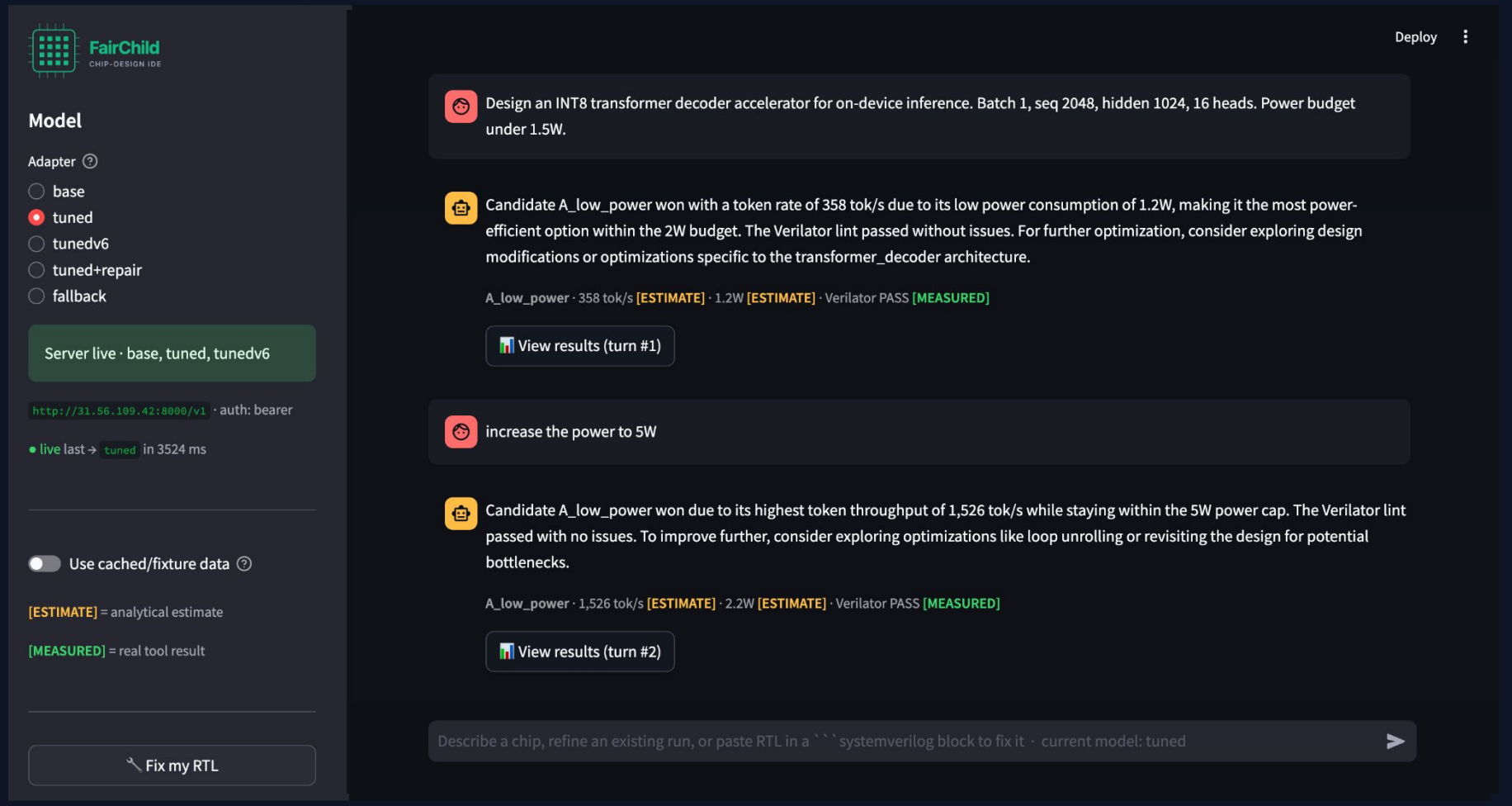
Closing the loop between hardware design and automated simulation.

What Did We Build?



Closing the loop between hardware design and automated simulation.

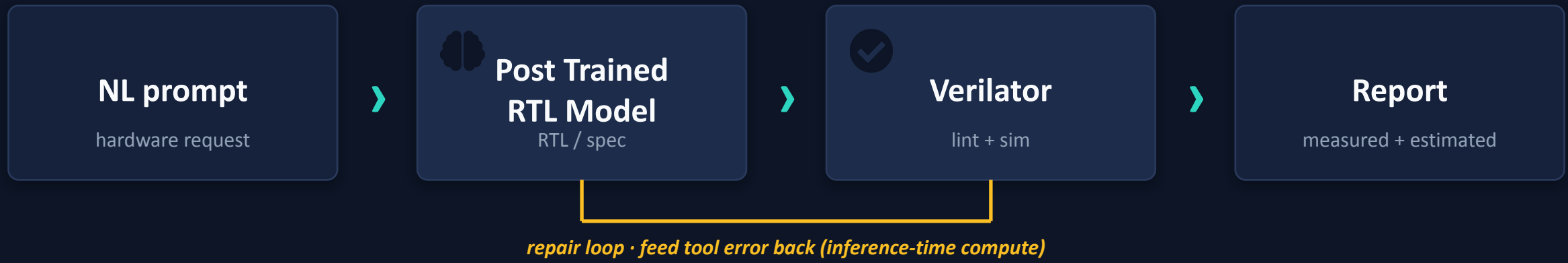
What Did We Build?



Closing the loop between hardware design and automated simulation.

Under The Hood: Overview

The model proposes. The tools evaluate. The IDE explains.



LAYER 1 — the contribution

Post-trained 7B chip model + a reproducible benchmark harness over a held-out set.

LAYER 2 — the demo

Agentic IDE that runs the model inside the loop — and keeps measured vs estimated strictly separate.

*Same pattern as an agentic SWE tool (Cognition) — act, read tool output, repair — **applied to RTL.***

The model

Post-training Qwen2.5-Coder-7B for RTL

Base coding LLMs write plausible-looking but often non-synthesizable Verilog. We domain-adapt with LoRA SFT (bf16) on a three-task RTL mixture.

Base `Qwen/Qwen2.5-Coder-7B-Instruct` Adapter `LoRA r=16, bf16`

SFT mixture — 39,151 examples

spec2rtl

32.2k 82%

Public corpora — description → code.
MG-Verilog (MIT) 16.2k + RTL-Coder Resyn27k 16.1k.

repair

5.8k 15%

inject bug → run tool → capture real error → diff fix.
Lint (Verilator) + semantic (iverilog) buckets.

SYNTHETIC · tool-verified

explain

1.2k 3%

Code → natural-language description.
Inverted from MG-Verilog modules.

Decontaminated against the 362-task held-out set (VerilogEval + RTLLM) — 0 source-hash overlaps. Resyn27k is model-generated / non-commercial — flagged in the report.

Training results

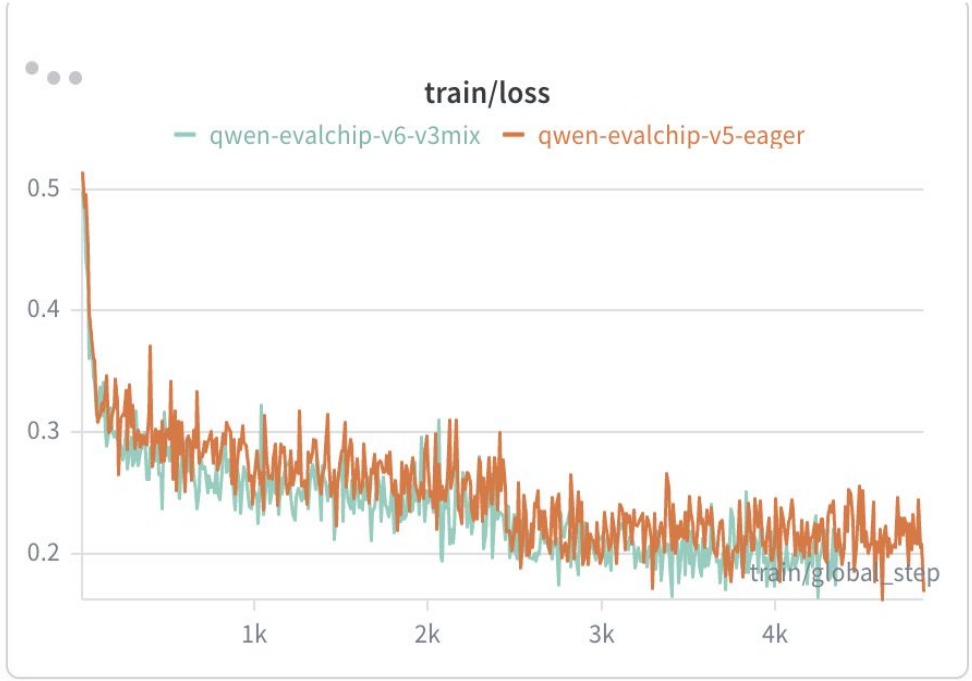
Held-out SFT validation: base vs tuned

model	eval loss	perplexity	token acc
base — Qwen2.5-Coder-7B	0.490	1.63	88.6%
tuned — v5 (LoRA)	**0.245**	**1.28**	**92.7%**

-50%
held-out eval loss

+4.1 pp
next-token accuracy

1× H100 80GB · bf16 LoRA · ~2 epochs (~4.8k steps) · ≈ 4 h
~4 runs to land a clean one — H100 bf16 attention-backward NaN, fixed with eager attention.



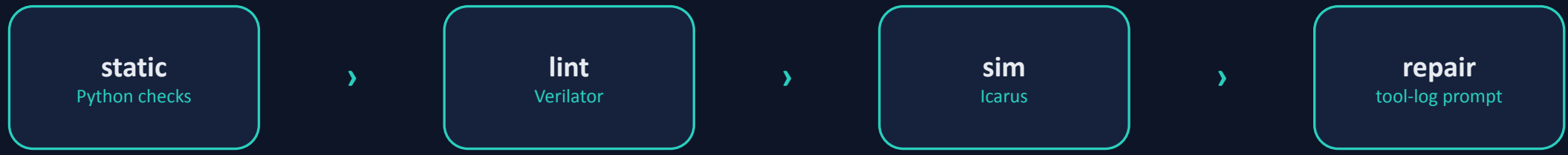
train/loss over the run — both healthy LoRA runs converge ~0.5 → ~0.2

This is the SFT fit metric (teacher-forced, assistant-masked LM loss on the same val split) — necessary but not sufficient for pass@k, which the Verilator harness scores separately. v5 measured at ~0.7 epoch.

An Agentic Harness For Generation and Verification

Implemented harness: static checks, Verilator lint, Icarus simulation, and repair re-score.

IMPLEMENTED HARNESS **fail-fast, cached, each stage timed**



↺ **Repair loop: lint/sim failure → tool log + RTL → repair prompt → regenerated RTL → re-score**

AROUND THE PIPELINE

Tool scoring API

Verilator/Icarus returns compile/test pass plus raw logs.

Repair benchmark

Failed greedy outputs are re-prompted with tool logs, then re-scored.

Typed result contract

Each run saves pass/fail, tool, errors, metrics, raw RTL, and logs.

Benchmark outputs

Harness writes table.md/csv, per-sample RTL, eval.json, and raw completions.

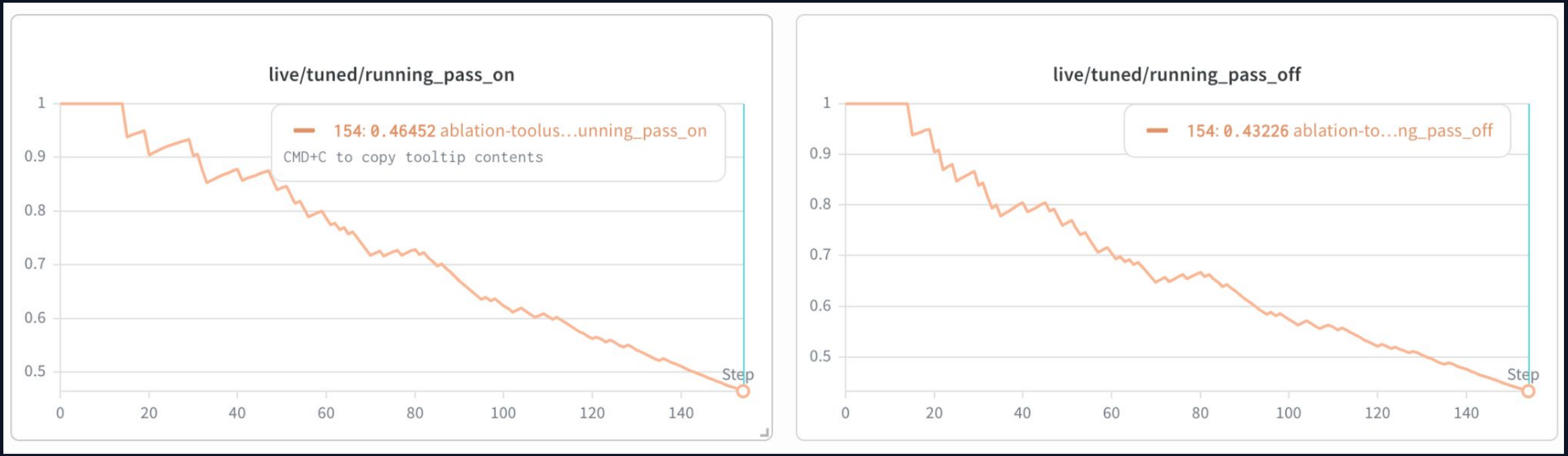
*Honesty contract: a result is marked **measured** only when a real tool actually ran.*

An Agentic Harness For Generation and Verification - Eval Results

Validation set: 155 held-out VerilogEval spec-to-RTL tasks (data/eval_tasks.jsonl), scored with Icarus/Verilator testbenches.

model	compile	test	pass@1	pass@3
base	0.81	0.35	0.35	0.46
v5	**0.89**	**0.38**	**0.38**	**0.50**

Ablation - tool use(w/o the harness)



The running_pass_off is one shot generation

A