AiNEKKO

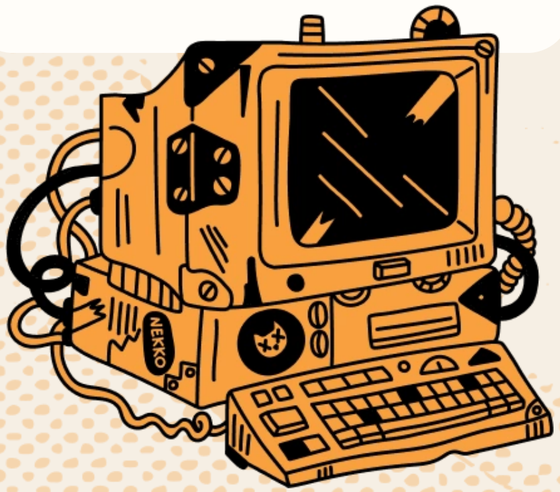
Developing an Open AgenticSOC

TANYA DADASHEVA
Co-founder, CEO
AiNEKKO, AIfoundry.org

RISC-V®
SUMMIT EUROPE 2026
BOLOGNA | 8-12 JUNE, 2026

QUESTIONS

Let’s talk about the agents



Have you fed your design into foundational models?

Documentation? RTL? Microachitecture?

Do you wish you could?

What stops you? Legal? Copyright? Token cost?

What is agentic systems in your mind?

2 agents? 5? 10?

Why now?

Three forces are converging

→ Desired workloads are overwhelmingly **AI inference**. Inference is a fixed compute graph with known data flow, precision, and latency requirements. The model becomes the **design specification for silicon**

→ Model builders are becoming synonyms for **a function, not an architecture**. Models absorb every required block into predictable releases, like software. **Models ship on a schedule now** - the hardware has to follow

→ AI agents and foundational models recently made a **huge leap forward** - what was previously possible only in SW space now entered HW, what took teams of dozens of engineers years can be explored in hours

SW? Integration? New models?

THE PROBLEMS FROM CUSTOMER'S VIEWPOINT

Integration burden

- Models must be converted, quantized, and partitioned per hardware target
- Deployment requires hardware-aware optimization across the stack
- Integration is complex and multidisciplinary – stalling go-to-market and requiring bigger teams/3rd party vendors

No Customization

Off-the-shelf silicon wasn't designed for your workload. You pay for features you'll never use while lacking the ones your application actually requires

Scaling the wrong bottleneck

- Power, thermals, latency, wake-up time, battery life, and BOM should be architecture constraints, not afterthoughts
- Data-center chips were optimized for dense, synchronous, high-batch workloads. Physical AI is sparse, asynchronous, and bursty

No agents or 3rd parties allowed

- Proprietary SW stacks, toolchains, and closed documentation under NDA significantly hamper the usage of coding assistants and agentic design systems.
- Customers are locked in the ecosystem of a single vendor that can't adopt new models fast enough and requires vendor field engineering support.

Hardware dictates the model – not the other way around. The customer deals with all the complexity

AI agents don't need to respect human boundaries in system design

> *"If you can vibe-code directly in assembly, you don't need a compiler. If you can reliably vibe-code RTL, you don't need the layers above it"*

Models are fixed (not turing-complete) compute graphs which allows agents to reason about model structure, quantization, data movement, memory layout, compute topology, scheduling, microarchitecture, and RTL as **coupled choices**

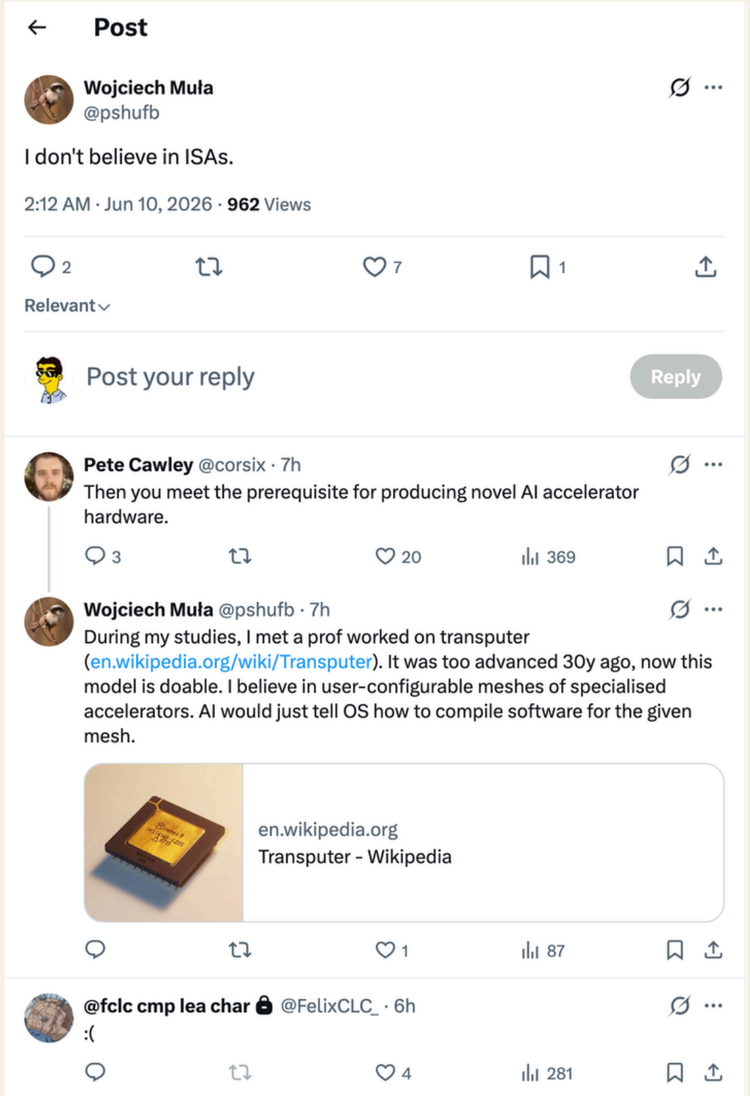
This collapses hardware/software co-design into an **end-to-end optimization problem**: given a workload and constraints, find the best implementation across the software-hardware boundary

What is required for it to work?

You can't ask AI to create chip design from scratch — that's an infinite search problem. Agents need a real, open, silicon-proven **substrate** they can reason about, compose, and specialize

RISC-V's importance is in how uniportant it must become

Wrongly attributed to Dave Ditzel



Post

Wojciech Mula
@pshufb

I don't believe in ISAs.

2:12 AM · Jun 10, 2026 · **962** Views

2 7 1

Relevant

Post your reply

Reply

Pete Cawley @corsix · 7h
Then you meet the prerequisite for producing novel AI accelerator hardware.

3 20 369

Wojciech Mula @pshufb · 7h
During my studies, I met a prof worked on transputer (en.wikipedia.org/wiki/Transputer). It was too advanced 30y ago, now this model is doable. I believe in user-configurable meshes of specialised accelerators. AI would just tell OS how to compile software for the given mesh.

en.wikipedia.org
Transputer - Wikipedia

1 87

@fclc cmp lea char @FelixCLC_ · 6h
:(

4 281

Agentic search for the best design

Model
Graph, operators, precision, latency targets

AGENTIC SEARCH

PLANNER
Decomposes model into sub-tasks

IMPLEMENTER
Writes SW, RTL, config per layer

COMMITTER
Merges validated changes

REVIEWER
Validates correctness & perf

Search across layers:

- Quantization & precision tuning
- Compiler & kernel mapping
- Microarch & memory hierarchy
- RTL generation & tile composition

MoM Design
SW + silicon co-optimized for your model

Erbium Core
RISC-V core + iRAM
Compute & local memory

Tile
Cluster of cores + shared NoC
Basic composable unit

Composable Fabric
8 to 4,000+ cores
Scale to workload

Model-on-Module
SW + HW co-optimized
Ready for deployment

CORE-ET: open, composable, silicon-proven IP building blocks

🤖 Minion Core

Tightly optimized, dual-threaded RISC-V core. Low-power. Small enough to replicate aggressively, predictable enough to compose into larger fabrics. Vector and tensor operations on up to 256-bit FP or 512-bit integer data per clock cycle

🏘️ Compute Tile (8 cores)

Atomic unit of design. First level of cache hierarchy. ~1 TOPS at 0.5W. Coordinated, efficient, shaped for systolic-array-like architectures

🌐 Composable Fabric

Scale from edge (8-core) to embedded (256-core) to datacenter-class (4096 core).
NEMI NoC: mesh with Static XY routing. AXI/APB support. Native broadcast

🧠 iRAM Memory Advantage

	iRAM	SRAM	LPDDR5	HBM3
Bandwidth	**0.25-1 TB/s**	~1 TB/s (mid-compute)	0.25 TB/s	1 TB/s
Latency	**6-9 ns**	3-4 ns	70-100 ns	100-150 ns
Power	**0.2-0.6W active/ 1mW standby**	12-60W	1-2W	5-9W

💻 Software Building Blocks

- Clock-accurate ET-CORE simulator
- Minion runtime & SP firmware
- TPA transputer-style process runtime
- Neuralizer transpiler framework
- LLVM/GCC support for ET-CORE SIMD
- Linux kernel driver & qemu emulator
- Host runtime & device abstraction
- General compute kernel library
- llama.cpp/ggml inference engine
- YOLO & LFM inference pipelines

FOUNDATION OPENHW™ PROVEN PROCESSOR IP

Results

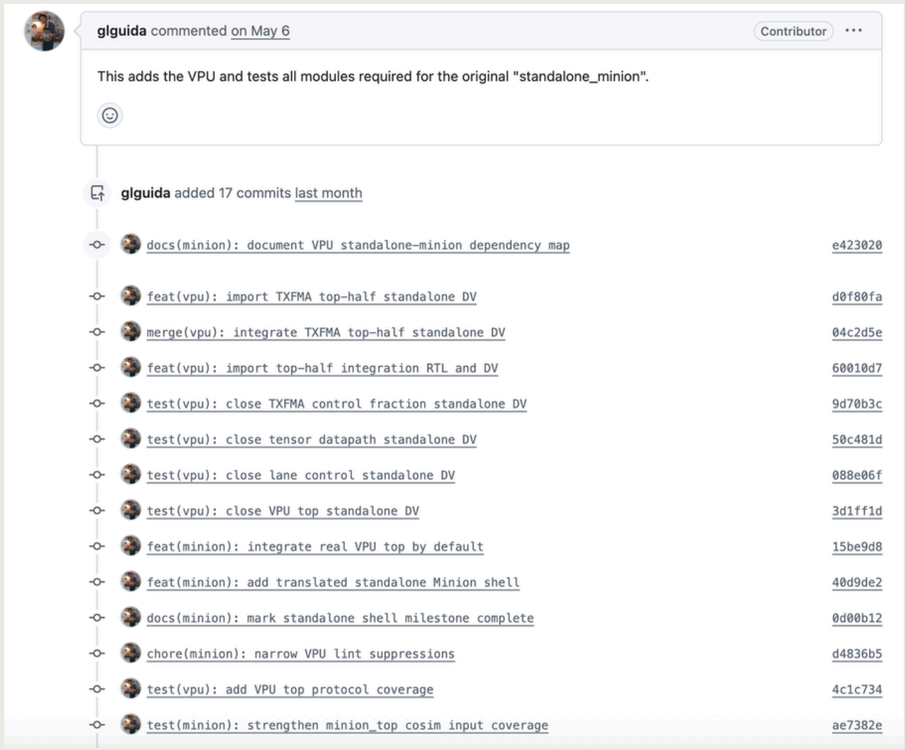
Models ported in the matter of hours

- Whisper
- YOLOv8n
- DnCNN3 (and its proto-GPU applications for ray-tracing)
- XFeat
- DepthAnything

Model-to-silicon

Join us at open hackathon on Hugging Face, coming soon!

Fully autonomus pull requests



AINEKKO.

THANK YOU!

DISCUSS

CONTRIBUTE

OWN YOUR = OWN AI.