FlexNPU

Compiler-Integrated FPGA Platform for Plug-and-Play AI Accelerator Research

Yimin Gao · Liangtao Dai · Junting Huo · Mircea Stan

University of Virginia · HPLP Lab · OSCAR 2026

THE PROBLEM

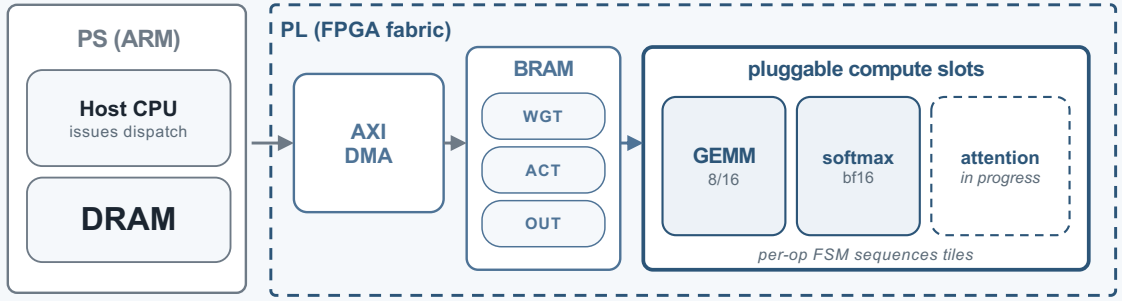
Every new compute tile is easy to design but hard to evaluate end-to-end: the compiler, runtime, data path, and FPGA shell are all tied to one microarchitecture.

THE CONTRIBUTION

A compiler-defined hardware slot contract. MLIR op categories map to typed slots behind a frozen ABI, conformance-checked against reference math, so any conforming tile runs compiled models with no compiler, runtime, or binary change.

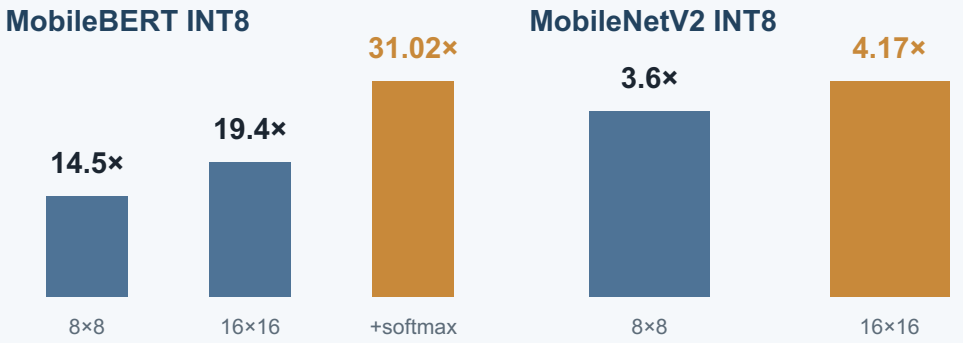
IREE *makes op categories reachable as named ops;* **FlexNPU** *turns them into a checkable hardware interface.*

ON-CHIP DATAPATH · ZYNQ / KRIA KV260



ABI-stable contract: *frozen ports · types · handshake — any conforming tile drops into a slot.*

EVALUATION — MEASURED ON KRIA KV260



vs CPU · bigger array, then softmax slot

vs CPU · second model class (CNN), same binary, bit-exact

Same binary runs both model classes — transformer and CNN.

One compiled .vmfb, one shell; only a runtime dispatch parameter changes to match the active array.

Both models run correctly. MobileNetV2 is bit-exact to the reference (top-1 = 845); MobileBERT predictions match, with bf16 logit drift across 24 layers (top-1 preserved).

Design a compute tile, not an accelerator system.

Open-sourcing the shell RTL, IREE pass, and runtime · github.com/hplp/FlexNPU