



Neural Processing Unit (NPU): Accelerating neural network inference

Stéphane Emery

CSEM White Paper • January 2026

 [ASICs for the Edge](#)

This document is the property of CSEM S.A. Users may not use the work for commercial purposes, they may not redistribute it in modified form, and they must give credit to the author.

info@csem.ch • csem.ch

Table of contents

Efficient AI/ML acceleration for low-power edge processing.....	3
Edge ML acceleration challenges.....	3
NPU enabling high compute efficiency through parallelization and data reuse.....	3
Silicon-proven integration in a system-on-chip (SoC).....	5
Exploring energy scaling limits.....	6
ML integration and deployment toolchain.....	7
CSEM AI/ML IP library.....	8
References.....	8
Document Information.....	9

Efficient AI/ML acceleration for low-power edge processing

Artificial intelligence (AI) algorithms, such as neural networks, are fundamental to many advanced computer vision and signal processing capabilities, often exceeding human-level performance. These algorithms enable the automation of repetitive tasks by processing input data through millions of computations, achieving high accuracy and reliable performance. However, the significant memory and power requirements for processing these networks pose challenges for deployment in miniaturized, wearable, and other energy-constrained applications.

Local processing of AI/ML algorithms within the end node, known as edge processing, can enhance the performance of many applications. However, unless a large battery size or frequent recharging can be afforded, embedded AI/ML processing is restricted to low complexity tasks or necessitates offloading to cloud processing. This offloading incurs the costs of energy-intensive radio communications, increased latency, and additional privacy concerns.

Optimized and energy-efficient AI/ML chips address these challenges by accelerating neural network computations within a constrained power budget, thereby enhancing edge processing capabilities.

CSEM's next generation neural processing unit (NPU) has been designed to address these challenges, **enabling neural network edge processing in power- and energy-constrained embedded systems**. The NPU is a standalone AI/ML accelerator IP that delivers state-of-the-art ML acceleration performance, optimized for embedded edge processing. With nearly 200x higher measured throughput (910GOP/s) and significantly increased efficiency (3.5TOPS/W) than its predecessor, CSEM's latest AI/ML system-on-chip showcases the NPU's cutting-edge performance for low-power AI chips.

Edge ML acceleration challenges

Edge processing of ML algorithms presents several challenges. Limited power availability necessitates efficient processing approaches, while on-chip memory is constrained by area and cost limitations. Off-chip memory access, on the other hand, incurs high power consumption. These challenges are particularly pronounced in image processing tasks and other spatial data arrangements, since the data volume of images and videos is substantial. Consequently, networks must perform millions of operations with significant data movements, leading to high memory bandwidth requirements. Thus, the memory subsystem quickly becomes dominant in the overall power budget and resulting in a so-called “memory wall”.

From a computational perspective, **arithmetic operations in neural networks primarily consist of multiply-and-accumulate (MAC) computations**, typically organized as matrix-vector or matrix-matrix multiplications. These operations are highly parallelizable, but they require frequent access to input data and model parameters. Since these values are often reused across multiple operations, naive implementations can lead to excessive memory traffic unless data is buffered locally—a technique known as *data reuse*. For example, in a two-dimensional (2D) convolution, the same kernel weights are reused to compute each output channel (or feature map). Leveraging data reuse mechanisms can reduce the total number of memory accesses by several orders of magnitude per layer, significantly improving energy efficiency.

NPU enabling high compute efficiency through parallelization and data reuse

Energy-efficient computation is crucial for enabling edge processing in energy-constrained embedded systems. The NPU, illustrated in Figure 1, employs a highly parallelized architecture that facilitates the reuse of input activations and weight parameters, minimizing memory accesses and maximizing the compute utilization. Block parameters enable user to configure the degree of compute parallelism (number of operations per clock cycle), the memory bandwidth, as well as the supported arithmetic precision options.

- Parameterizable IP
 - # of Processing Elements (PE): {1, 2, 3, ...}

- # of MACs/PE: {..., 2x2, 3x3, 4x4, 5x5, ...}
- Memory port count: { 1, 2, 3, ...}
- Memory bus width: {8, 16, 32, ..., 256, ...} [support for wide memory bus interfaces]
- Memory level parallelism (# of Banks per port, # of pending memory transactions per port)
- Multiple arithmetic precision options (e.g. 8 and/or 16-bit weights and activations)

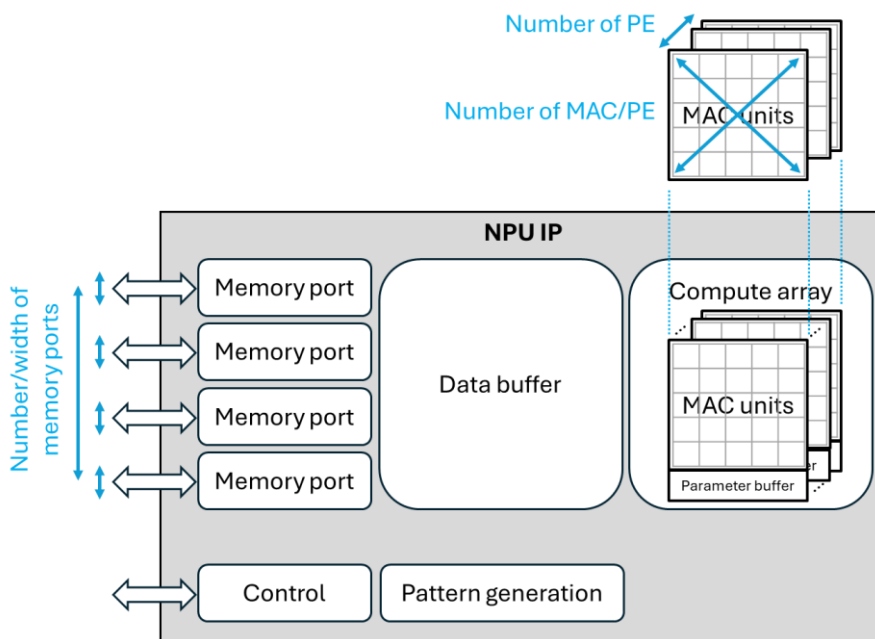


Figure 1: High-level overview of the NPU IP

- Fully programmable through compiler-generated memory-mapped settings (microcode)
- Standalone execution of common classes of neural networks e.g.
 - Convolutional neural networks (CNNs)
 - Multi-layer perceptron
 - MobileNets
 - ResNets
 - VGG
- High data reuse to minimize memory access
 - Weights: up to 100% (e.g. kernel parameters only requested once from memory and then buffered inside compute array for DS-convolution)
 - Activations: up to 100% (all inputs only requested once from memory)
- Achieving close to peak roofline [2] performance: (See Figure 3)
 - Native layer types with standalone NPU hardware acceleration support:
 - Convolutions (2D)
 - Depthwise separable (DS) convolutions (2D)
 - Dense/fully connected
 - Residual connections
 - Point-wise operations (1D/2D/3D addition, subtraction, multiplication)
 - Pooling (2D min/max/average)
 - Scaling/Shifting (including offline batch normalization)

- Biasing
- Activation function (Linear, ReLU)
- Hierarchical and conditional execution
 - Interrupt-based mechanism allows the host MCU to suspend inference early or choose branches dynamically based on intermediate results.
 - Supports early-exit and layer-wise selective execution.
- Standalone operation capability
 - Fully synchronous single-clock design, operable without a host controller.
 - Optional interrupt output to notify the host upon layer or network completion.
- Possibility of tandem operation between NPU and the host processor for layers not supported on the NPU (e.g. SoftMax, Online Batch Norm, ...)

Figure 2 illustrates example use and system connections around the NPU IP.

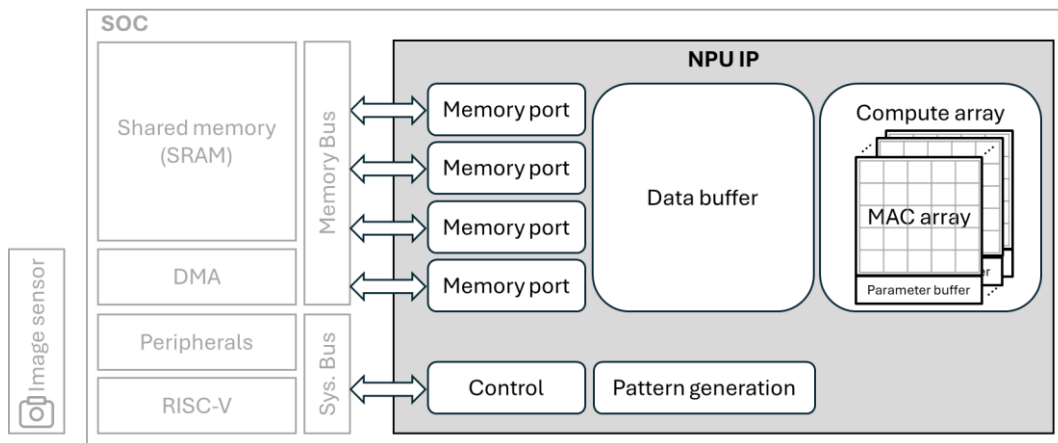


Figure 2: Example of NPU integration in a SoC.

Silicon-proven integration in a system-on-chip (SoC)¹

The NPU’s standalone operability allows for easy integration into a system-on-chip, eliminating the need for complex ISA extensions. Besides the clock and reset signals, the NPU only requires an Advanced Peripheral Bus (APB) for configurations and a generic number of memory ports with Open-bus Interface (OBI). The standard APB bus enables the host controller to configure and monitor the NPU’s status. All layer settings and parameters are autonomously retrieved by the NPU from the memory through its own memory port(s).

CSEM’s NPU is silicon-proven on a CSEM ML system-on-chip, implemented a 22nm technology node, offering detailed performance measurements and demonstrating straightforward system integration. The ML SoC called [FIBONACCI](#), features two NPU instantiations with different configurations listed in Table 1.

Table 1 Sample implementation scenarios for NPU

Implementation Scenario	NPU_A	NPU_B
Voltage	0.65V	0.80V
NPU number of PEs	4	16

¹ For information related to the IP version please refer to the (Document Information) section.

Implementation Scenario	NPU_A	NPU_B
NPU number of memory ports	2	4
Precision	16b	8b
Total MAC Units	100	800
Memory bus interface width	256b	256b
Max Frequency	200 MHz	600 MHz
Silicon area (excl. memory)	840 kGE ²	3820 kGE

Figure 3 presents energy-performance results for different scenarios on the implemented silicon demonstration.

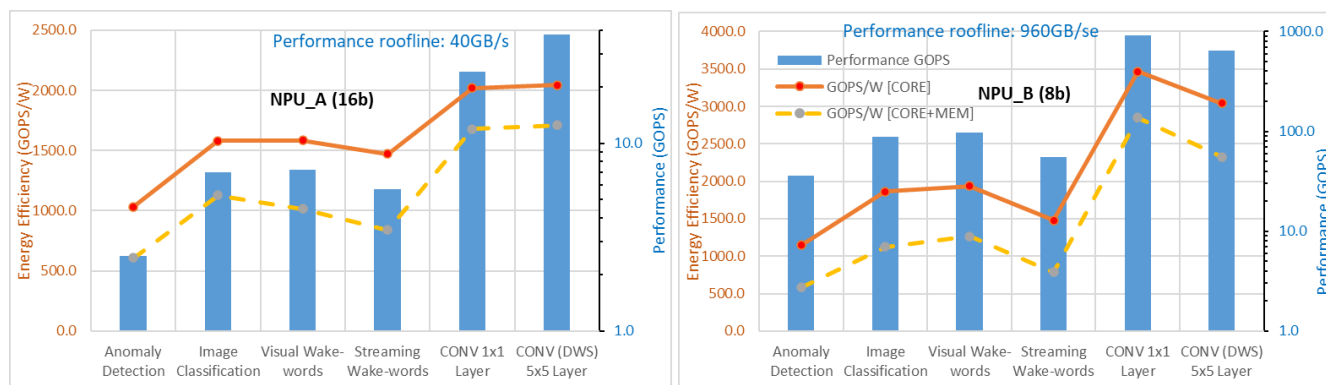


Figure 3: Measured energy-performance results for different applications (including MLPerf-Tiny benchmarks [1]) using randomly distributed inputs and weights on the silicon demonstrator.

Exploring energy scaling limits

Figure 4 illustrates the impact of voltage scaling on performance and energy efficiency of the NPU. The results are shown for the core of NPU_B (8b) @22nm technology running CONV 1x1. The X-axis shows peak achievable frequency under that condition, and the Y-axis presents TOPS/W.

² Gate-equivalent area is calculated as post-synthesis gate-area divided by the area of the smallest NAND2 gate.

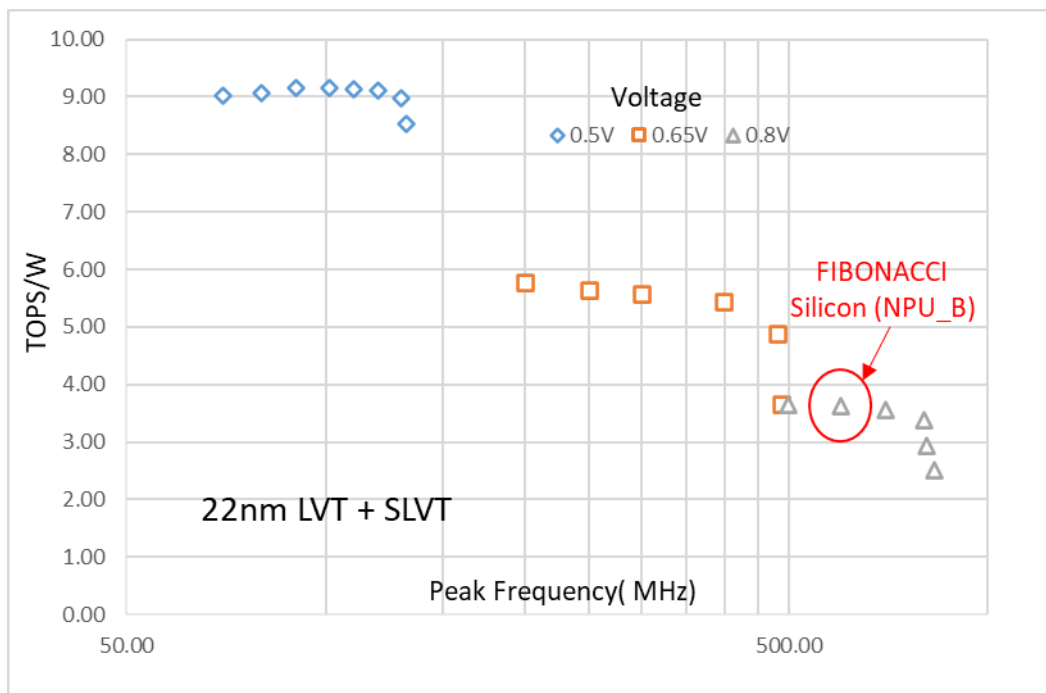


Figure 4: Impact of voltage scaling on performance and energy efficiency of NPU_B

ML integration and deployment toolchain

CSEM delivers end-to-end integration support for the NPU, complemented by a comprehensive software stack for model deployment, fine-tuning, bit-accurate validation and emulation, and high-performance inference. The toolchain, illustrated in Figure 5, orchestrates the complete deployment pipeline and automatically transforms trained models into optimized, deployable memory images for seamless integration into the target system.

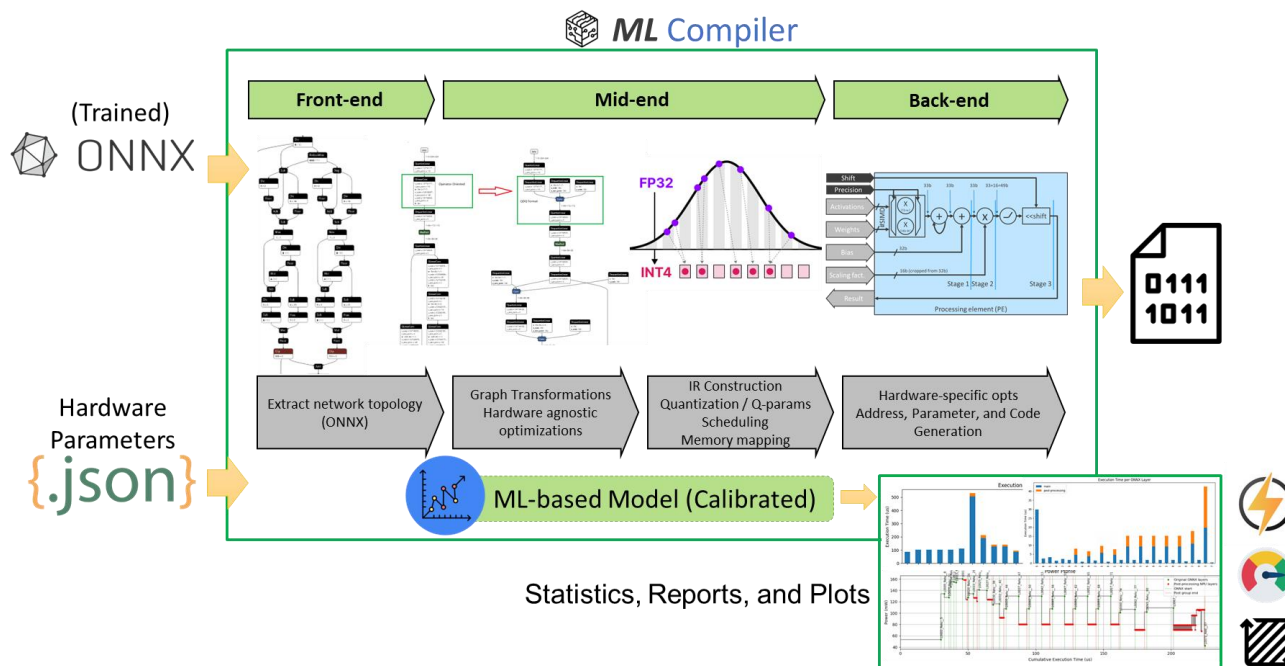


Figure 5: Deployment of trained model using ML compiler toolchain.

The ML Compiler is an end-to-end software toolchain that maps ONNX models onto the NPU architecture. It performs graph parsing, shape and precision inference, and hardware-aware analysis to

generate an optimized intermediate representation aligned with the NPU's compute, memory, and dataflow constraints. An advanced modeling framework integrated into the ML Compiler delivers fast and accurate performance and energy estimates for a given ONNX network across selected architectural configurations. Leveraging ML-based techniques trained on cycle-accurate simulations, the framework enables automated design-space exploration and Neural Architecture Search (NAS), providing early and actionable feedback to designers. This approach helps identify performance and efficiency bottlenecks, supports informed architecture and system-level design decisions from the outset, and enables rapid exploration of neural network architectures well before full backend implementation and final silicon availability.

CSEM AI/ML IP library

The NPU is part of an IP library for the acceleration of edge AI/ML. This library, summarized in Figure 6, offers a wide selection of hardware IPs for the design of modular and flexible SoCs that enable end-to-end inference on miniaturized systems. Available IP categories include ML accelerators, dedicated memory systems, a RISC-V based 32-bit microcontroller (icyflex-V), pre-processing blocks (focused on audio application such as feature extraction block) and peripherals.

These ML accelerators enable parallel computing for dedicated ML tasks, from computer vision to time-series signals classification. The available memory systems are optimized for the accelerators, based on either SRAM, register files, or NVM. Thus, the best matching memory solution can be chosen based on the tradeoff among power consumption, memory access, and storage density. A wide range of peripherals enable seamless integration with many external devices.

Tailoring the solutions offered to customers' needs is our priority at CSEM. These IPs allow for design customization and flexible programmability (e.g., for size and precision). The modular nature of this IP library allows for fast and simple integration in any system.

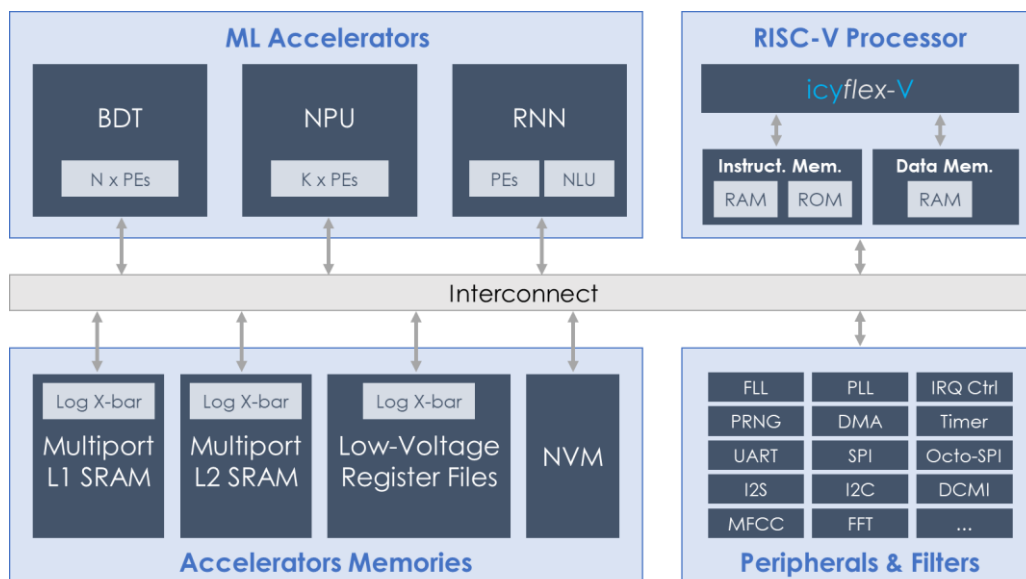


Figure 6: The CSEM IP library for the acceleration of edge AI/ML

References

- [1] MLPerf Tiny : <https://mlcommons.org/benchmarks/inference-tiny/>
- [2] Roofline: an insightful visual performance model for multicore architectures, <https://doi.org/10.1145/1498765.1498785>

Document Information

- Document version: 2.05
- IP version discussed: CSEM NPU v2.1
- Silicon Demonstration: FIBONACCI
- Date: January 2026