



FETA: a Flexible Low-Power AI/ML Accelerator for Time Series Signals

Stéphane Emery

CSEM White Paper • May 2025

 [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

Enabling time-series AI/ML functionalities with ultra-low power consumption

The large variety of data acquired by sensors on mobile, wearable, and IoT devices has enabled numerous new applications, such as long-term medical monitoring, fitness tracking, predictive maintenance, and speech processing. ML algorithms such as neural networks (NNs) are often used to process time-dependent sensor data (time series) from these sensors. However, exploitation in edge devices is still limited due to the inefficient processing of vast amounts of sensor data.

Today, ultra-low-power devices are rarely enabled with ML functionalities. The numerous operations required by ML algorithms are typically offloaded to the cloud, at the cost of power-hungry radio communication, long latency, and privacy risks. Thus, the design of low-power NN accelerators is key to enabling ML features in any battery-powered device.

The development of optimized, yet flexible, accelerators for NNs can unlock significant power consumption savings. Thanks to the design of these circuits, the execution of computing-intensive algorithms can be made possible for any portable device and create unprecedented use cases for edge devices.

Table of contents

| | |
|---|---|
| Time series analysis with NNs..... | 3 |
| FETA: a low-power flexible RNN accelerator for any time-series signals..... | 3 |
| Example of System-on-Chip based on FETA..... | 4 |
| Easy to use and flexible for integration | 5 |
| References | 7 |

Time series analysis with NNs

In time-series applications, recurrent neural networks (RNNs) are commonly used because they can identify temporal correlations within the input signal. Most of the RNNs operations consist of matrix-vector multiplication (MVM), a highly parallelizable workload based on multiply-and-accumulate (MAC) operators.

RNNs, however, cannot exploit data-reuse the way CNN accelerators typically do, and they rely on non-linear functions such as sigmoid and hyperbolic tangent which require more complex hardware compared to the activation function like the popular Rectifier Linear Unit (ReLU) used in CNNs. This makes CNN accelerators unsuitable to RNNs and calls for specialized RNN accelerators. Needless to say, flexibility and parametrization are key requirements for supporting a descent range of application scenarios and use cases.

FETA: a low-power flexible RNN accelerator for any time-series signals

FETA is a stand-alone accelerator that fully offloads the computations from a CPU. FETA implements an efficient dataflow and supports dynamic configuration for MVM operation with the following main features:

- 1) Exploiting MAC parallelization by using N_{PE} parallel processing elements (PEs)
- 2) Data orchestration optimized for Matrix-Vector Multiplication: spatial reuse through input vector broadcasting and temporal reuse through output stationary accumulation.
- 3) Support for 8-bit / 16-bit arithmetic¹.

FETA also contains optimized hardware implementation for other operations of an RNN which can take a significant number of cycles and energy if not properly performed:

- 4) Dedicated CORDIC-based non-linear unit (NLU) capable of performing the operation *sigmoid* and *tanh* efficiently on 8 or 16-bit variable input vector length, overcoming CPU and LUT-based approaches. The *ReLU* activation function is also supported.
- 5) Hardware re-use for vector-vector element-wise addition and multiplication.
- 6) *Max* and *ArgMax* implementations with interrupt generation capability as a low-cost alternative to *SoftMax*.

Maximum flexibility is provided to the user by offering programmable operations for RNN and beyond:

- 7) Operation sequence to implement a given RNN or layer requiring MVM operation is defined with a micro-code.
- 8) Loop and jump operation are supported.
- 9) Specific operations to start and/or resume processing on external events (hardware and software) are available.
- 10) Specific operations can generate hardware events to external components.

Straight-forward SoC integration:

- 11) Micro-code, input/output and parameters can be stored in the system memory (no parallel bus for micro-code and data).
- 12) Single clock fully synchronous design.

¹ Linear algebra computation (MVM, ...) with flexible support for 8-b / 16-b arithmetic, while the Non-linear Unit (NLU) works with 16-b arithmetic only. Rounding and padding operations are provided for smooth conversion of precision.

LSTM², GRU³ and dense layers can be supported by updating the FETA micro-code. Unidirectional and bidirectional modes, with support for peephole connections and different numbers of layers can also be implemented.

Example of System-on-Chip based on FETA

In recent years, several ML-based solutions focusing on audio and voice processing have emerged. They can perform several tasks, such as Voice Activity Detection (VAD), Keyword Spotting (KWS), noise cancelling and speaker identification. This section presents the features and performance of a CSEM system-on-chip (SoC) based on the FETA accelerator (illustrated in Figure 1), along with projections to different technology flavours and application scenarios. In the rest of the document, we use the networks defined in the HelloEdge paper ^[a] as a reference and give some performance results based on those.

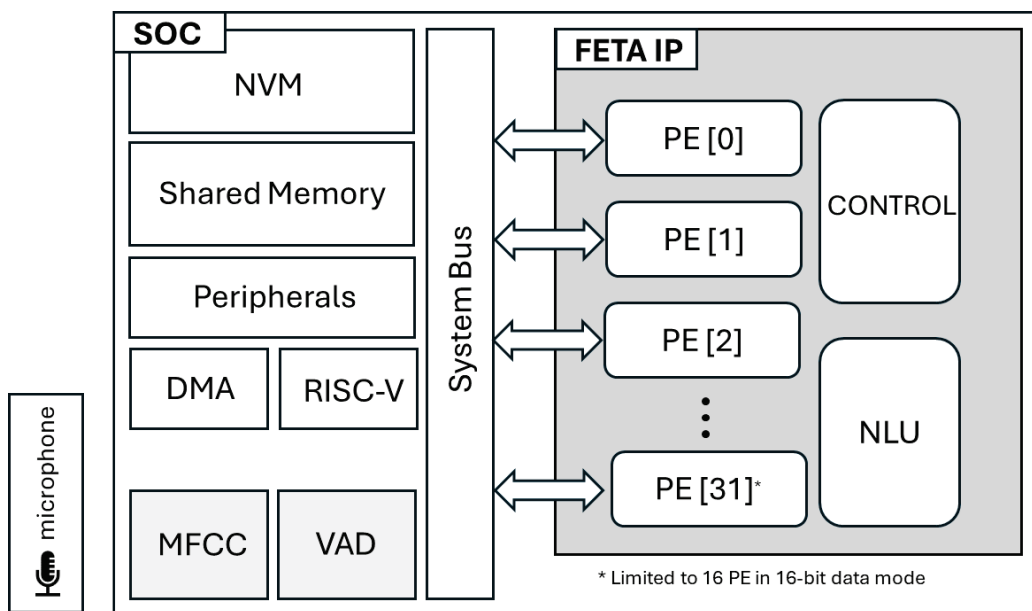


Figure 1: Example of the FETA IP in a SoC

| Baseline Specifications and Assumptions | Value |
|---|---|
| Technology | 22nm ULP Standard Cells – 0.65V Core |
| FETA N _{PE} | 32 |
| FETA Memory Interface width | 256b |
| Max Frequency | 200 MHz |
| Large LSTM (12 keywords) ^[a] | LSTM (344 cells) + FC ⁴ (12 neurons) |
| Small LSTM (10 keywords) ^[a] | LSTM (118 cells) + FC (10 neurons) |

² Long Short-Term Memory neural network

³ Gated Recurrent Unit

⁴ Fully connected layer

The gate-equivalent area of FETA (excluding the memories) is 131kGE⁵.

The Performance-Power comparison for different scenarios is presented in the table below:

| | Peak Efficiency with large LSTM | Lowest Leakage with small LSTM | Lowest power with small LSTM |
|---------------------------------|--|---|------------------------------|
| Standard cell | LVT ⁶ +SLVT ⁷ | ULL ⁸ | LVT |
| Frequency (MHz) | 200 | 1.12 | 1.12 |
| FETA Dynamic power (uW) | 1064 | 11.52 | 6.01 |
| FETA Leakage power (uW) | 246 | 0.14 | 14.99 |
| FETA Total power (uW) | 1310 | 11.66 | 21.00 |
| Performance (GOPS) | 5.9 (16b ⁹) 12.2 (8b ¹⁰) | 0.036 (16b ¹¹) 0.072 (8b ¹²) | 0.036 (16b) 0.072 (8b) |
| FETA Energy efficiency (GOPS/W) | 4496 (16b) 9312 (8b) | 3087 (16b) 6173 (8b) | 1714 (16b) 3428 (8b) |
| Memory size (indicative) | 1 MB ¹³ | 148 kB ¹⁴ | 148 kB |

Easy to use and flexible for integration

CSEM provides seamless integration support for FETA, along with the full software stack and the required toolchain for training (QAT¹⁵ / PQT¹⁶), fine-tuning, bit-true validation, and accelerated inference. A holistic overview of the software framework for FETA is illustrated below:

⁵ Gate-equivalent area is calculated as post-synthesis gate-area divided by the area of the smallest NAND2 gate.

⁶ Low threshold voltage transistors

⁷ Super-low threshold voltage transistors

⁸ Ultra-low leakage transistors

⁹ 16b large LSTM requires more than 1MB of memory.

¹⁰ 8b weight and 8b activations

¹¹ 16b weights and 16b activations

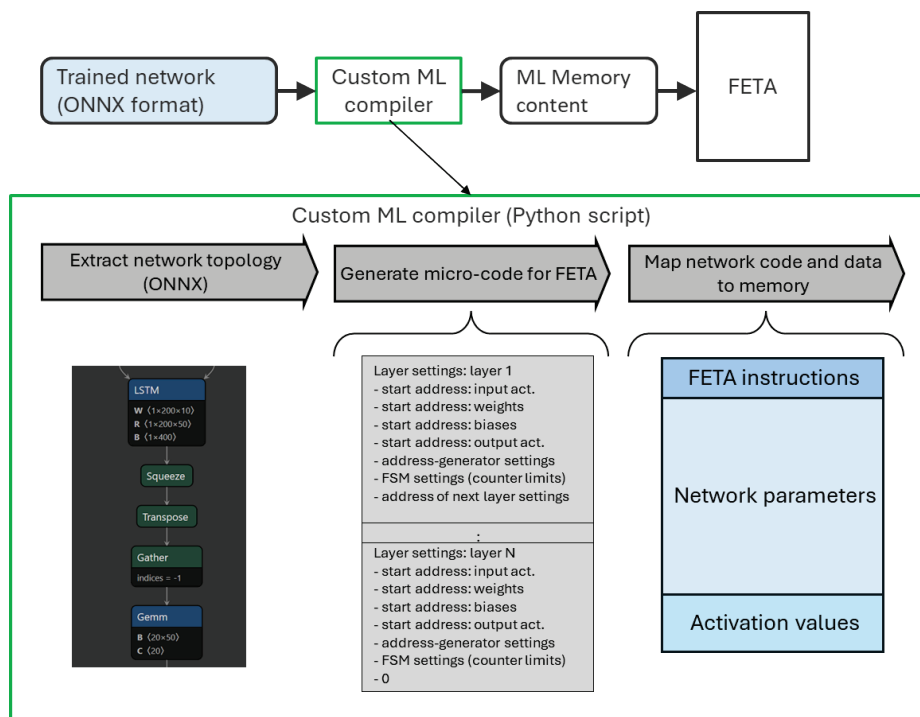
¹² 8b weights and 8b activations

¹³ Sufficient memory to store and run Large LSTM (HelloEdge) on 8-bit activation and parameters

¹⁴ Sufficient memory to store and run Small LSTM (HelloEdge) on 16-bit activation and parameters

¹⁵ Quantization-aware Training

¹⁶ Post-quantization Training



The FETA accelerator is part of an IP library for the acceleration of edge AI/ML. This library 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 processor and 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, and they are based on either SRAM, register files, or NVM. Thus, the best matching storage 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 offered solutions 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.

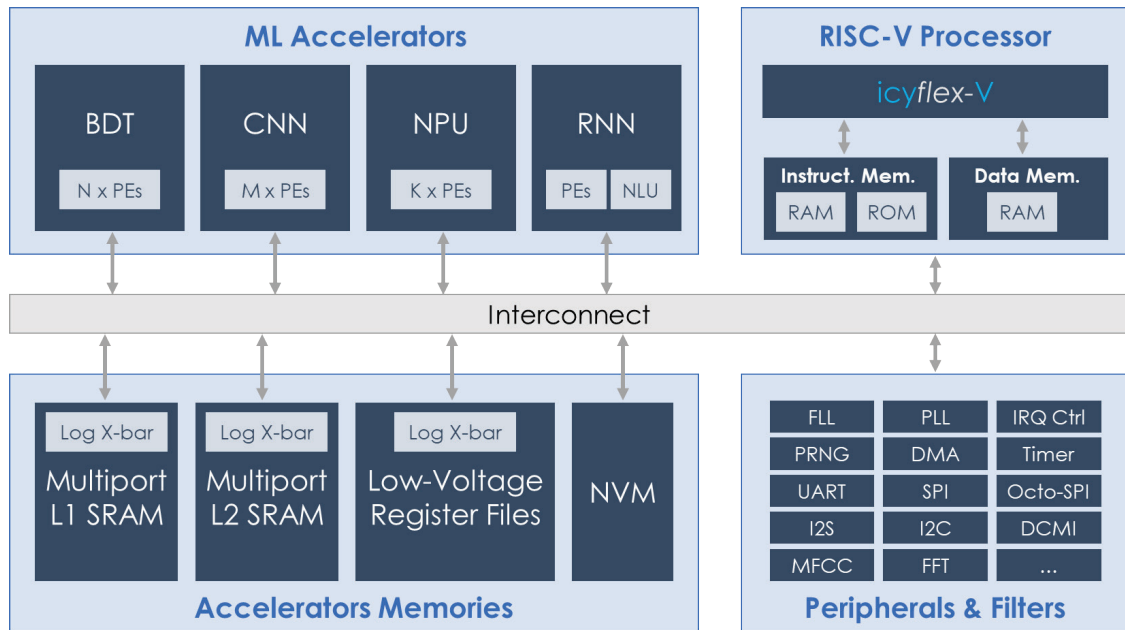


Figure 2: The CSEM IP library for the acceleration of edge AI/ML

References

- [a] Zhang, Yundong, et al. "Hello edge: Keyword spotting on microcontrollers." arXiv preprint arXiv:1711.07128 (2017).