BumbleBee: A 3 mW Fused-Kernel Flash Attention Edge AI Co-Processor in 18 nm FD-SOI CMOS

Joaquin A. Cornejo*§, Filipe Pouget*, Amelie Poullot*, Raphael Gras*, Dominique Bousquet*, Nicolas Lebouleux*, Tarun Chawla*, Sebastien Marchal*, Sylvain Biard*, Christophe Jégo‡, Camille Leroux‡, Sylvain Clerc *¶, Tifenn Hirtzlin†, Benoit Larras§, Andreia Cathelin* and Antoine Frappé§
* STMicroelectronics, Crolles, France
† Université Grenoble Alpes, CEA, LETI, Grenoble, France
‡ University of Bordeaux, Bordeaux INP, IMS Lab, UMR CNRS 5218, France
§ Univ. Lille, CNRS, Centrale Lille, JUNIA, Univ. Polytechnique Hauts-de-France, UMR 8520 - IEMN, Lille, France
¶ Université Grenoble Alpes, CEA, LIST, Grenoble, France

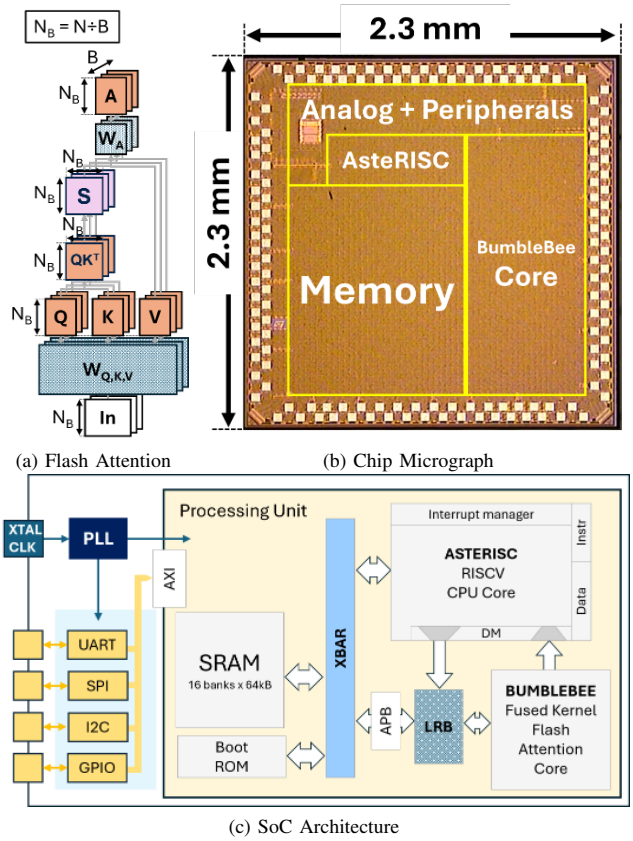
***Abstract*—The attention mechanism is the core operation in Transformer models but remains difficult to deploy at the edge due to high memory access and data movement requirements. This work presents BumbleBee, an 18 nm FD-SOI CMOS co-processor implementing the Flash Attention mechanism for the first time. It achieves 3 mW power consumption and reduces bus transfers by 5 through a fully dataflow-driven architecture. Beyond energy efficiency, this dataflow also reduces latency by minimizing memory stalls, enabling fast token-level processing. The architecture integrates a dedicated attention head combining an addressable MAC array together with systolic arrays, and leverages binarized representations to further reduce computational cost. It achieves 20 and 248 mJ/inference for BERT-Mini and DETR workloads, respectively.**

***Index Terms*—Co-processor, Transformers, RISC-V, Systolic Array, Addressable MAC Array, Flash Attention, Mixed-Precision**

I. Introduction

Transformer-based neural networks, which rely on the scaled dot-product Attention Mechanism (AM) [1], have demonstrated strong performance across various tasks [2]. At scale, these architectures—commonly referred to as Large Language Models (LLMs)—have been deployed on cloud infrastructure due to their substantial computational demands. The growing interest in edge AI has motivated research into efficient on-device inference [3]. However, edge deployment imposes stringent constraints on performance, power, and area, and many works have focused on addressing these constraints, notably through quantization [4], [5].

Deploying these models at the edge is particularly challenging. First, it involves $\mathcal{O}(N^2)$ computation and memory access, where N is the sequence length. Second, the SoftMax introduces a dependency on full row completion, creating a bottleneck that limits parallelism. The Flash Attention Mechanism (FAM) proposed in [6] addresses these limitations by introducing tiling, as shown in Fig. 1a, enabled by partial maximum and sum normalization. However, unlike conventional tiling approaches used in convolution or linear layers, this mechanism alternates short matrix multiplications with partial SoftMax computations. This execution pattern reduces the efficiency of conventional Systolic Arrays (SA) matrix multiplication accelerators, which are optimized for long,



(a) Flash Attention (b) Chip Micrograph

(c) SoC Architecture

Fig. 1: (a) Flash Attention Mechanism where In is input matrix, W weight matrices; Q, K and V are Query, Key and Value matrices respectively; S is SoftMax output and A is output Attention matrix [6]. AM is divided through sequence length N in B blocks, processing submatrices of N_B length. (b) 18 nm FD-SOI CMOS BumbleRISC SoC chip micrograph, highlighting AsteRISC RISC-V CPU, 1,024kB SRAM memory and BumbleBee FAM accelerator. (c) SoC architecture diagram.

contiguous operations.

This work presents BumbleRISC SoC, implemented in 18 nm FD-SOI CMOS technology, shown in Fig. 1b. The SoC architecture, depicted in Fig. 1c, is composed of a custom RISC-V CPU (AsteRISC), a 1,024 kB SRAM memory and a FAM Fused-kernel accelerator (BumbleBee), communicating through a crossbar and an APB bus. The SoC peripherals and interfaces are similar to [7].

The SoC architecture includes the following contributions: 1) a Flash Attention mechanism that uses customized number formats for each computation stage to maintain accuracy, while saving computational energy; 2) an Addressable MAC Array (AMA), proposed as an alternative to conventional systolic MAC arrays to mitigate the latency introduced by the Flash Attention mechanism; 3) an optimized dataflow together with a synthesized, shared direct-memory access scheme to reduce the memory access overhead of Transformer models.

II. CPU and Co-processor Co-optimization

A. Dataflow

As illustrated in Fig. 2a, a general purpose vanilla attention dataflow, featuring distinct matrix multiplication and non-linear units, is inherently limited by bus transfers. The FAM transforms the global SoftMax into partial SoftMax operations and enables a fused-kernel scheme to perform the multiple successive computations without intermediate memory access, as shown in Fig. 2b, allowing for a reduction of bus transfers by 5. Beside fused kernels, this work leverages RISC-V/FAM co-integration to tailor specific direct address mapping to

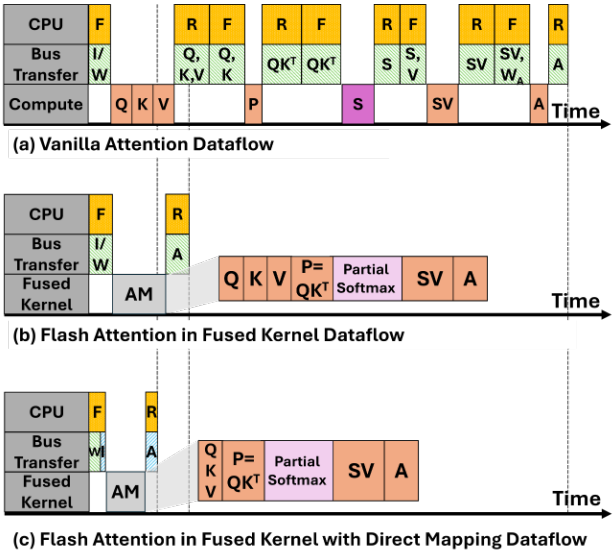


Fig. 2: Attention Mechanism dataflow for three different cases: (a) Vanilla attention dataflow in general purpose hardware with CPU control doing Fetch (F) and Read (R) instructions for bus transfer of different inputs/outputs, distinct compute units for matrix multiplication in red and global SoftMax (S) in purple; (b) fused kernel Flash attention dataflow, with single compute unit doing matrix multiplication and partial SoftMax, reducing bus transfers. (c) proposed implementation with fused kernel computing Q,K,V in parallel and using DM for frequently accessed data I/A.

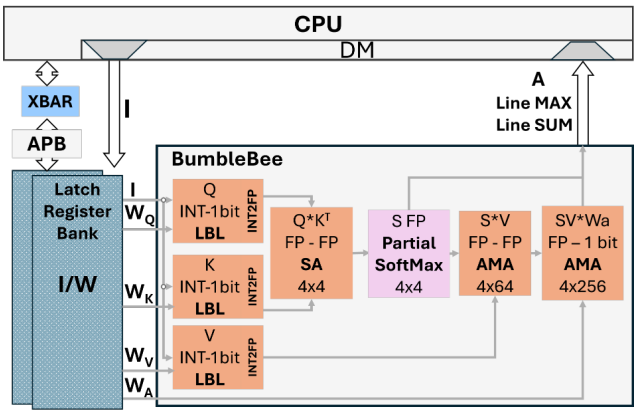


Fig. 3: BumbleBee architecture showing APB bridge accessed through CPU AXI to load weights, inputs written directly through direct mapping. Inputs and weights are accessible through the latch register bank; Q,K,V blocks compute in parallel, $Q{\cdot}K^T$ Systolic Array, SoftMax block, and $S{\cdot}V$ and $SV{\cdot}W_A$ Addressable MAC arrays blocks in FP representation.

further divide by 2 the latency of frequently accessed data, such as the input tokens I and output activations A. Fig. 2c shows the gain in latency using the combinations of flash attention, fused kernels and direct mapping techniques.

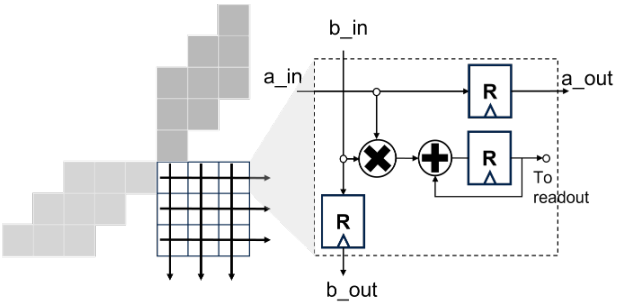
B. Flash Attention Accelerator BumbleBee

BumbleBee is adapted for models with embedding dimension of 256 such as BERT-Mini [8] for natural language processing or DETR [9] for object detection. It is a FAM accelerator, operating on tiles of size 4×256 without limitation on global input sequence length. The data-path diagram is shown in Fig. 3.

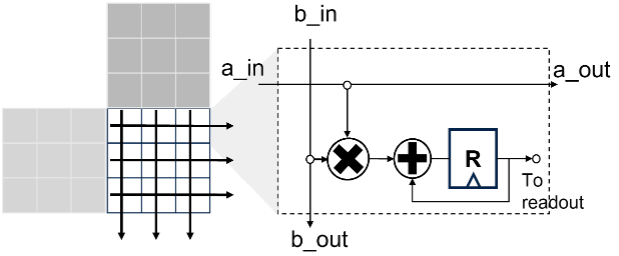
Number formats are customized for each operation to reduce computational cost. Reference [10] proved that the Linear/BitLinear layers (referred as LBL in Fig. 3) can withstand strong quantization. Here, the first linear layer is binarized to allow BIN $\times$ INT8 computations. This allows the parallel computation of Query Q, Key K, and Value V linear layers within a single clock cycle at an acceptable hardware cost.

On the contrary, subsequent $Q\cdot K^T$, SoftMax (S), $S\cdot V$, and $SV\cdot Wa$ layers require Floating-Point (FP) representation to maintain acceptable model accuracy. A custom 14-bit floating-point (FP14) format is used for these layers. This is derived from a study in [11], with 2 extra bits added to prevent underflow/overflow. For both models, the accelerator achieves an average 2% quantization error relative to the FP32 model in PyTorch.

Systolic Arrays, as specified in Fig. 4a, reach high utilization performing contiguous matrix multiplications. For K square matrix multiplication of $T\times T$, a SA requires approximately $(K+2)\times T$ cycles to complete, as shown in Fig. 5a. However, Flash Attention alternates short matrix multiplications with frequent SoftMax computations, making systolic arrays inefficient, as depicted in Fig. 5b. To reduce SA latency in the context of FAM, we introduce the Addressable MAC Array in Fig. 4b. Instead of relying on systolic propagation, AMA loads



(a) Systolic Array.



(b) Addressable MAC Array.

Fig. 4: Comparison between the Systolic Array (SA) and the Addressable MAC Array (AMA) Matrix Multiplication operators. (a) in SA, systolic propagation relies on registers receiving inputs and passing them to right and down neighboring cells; (b) AMA feeds values directly to each cell, reducing latency of operation.

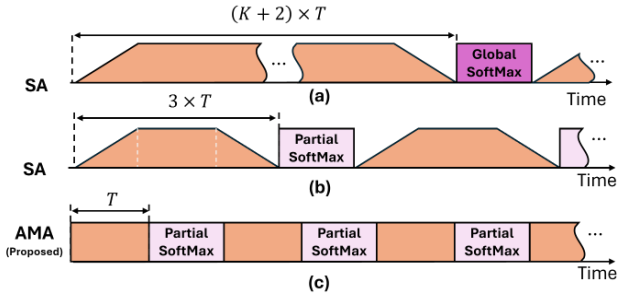


Fig. 5: Comparison between (a) Vanilla Attention dataflow using SA, (b) FAM dataflow using SA, and (c) AMA operators for matrix multiplication.

inputs directly to each individual cell. Thus, AMA structure eliminates the need of propagation registers, but requires the line and column drivers to handle higher loads. This drawback is mitigated during the physical synthesis process, able to optimize the netlist for the increased capacitive loads. As illustrated in Fig. 5c, AMA reduces short matrix multiplication latency by 3, in the case of a single matrix computed between each partial SoftMax.

C. RISC-V CPU AsteRISC

A configurable RISC-V processor core adapted from [12] is used in this work. It handles data transfer, data orchestration and format conversion to INT8 inputs and BIN weights from 32-bit vector words.

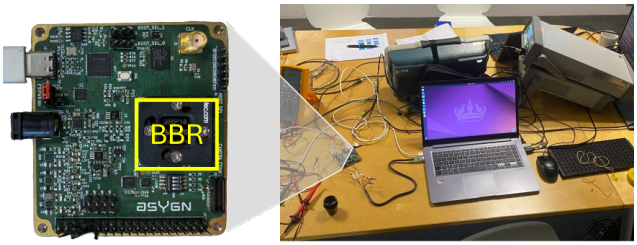


Fig. 6: Test setup and PCB test board with the socket for the BumBleRISC (BBR) chip highlighted in yellow.

The core interfaces with the BumbleBee via a synthesized 2 x 9 kB Latch Register Bank (LRB) composed of level sensitive latches with write hazard controlled via integrated clock gating scheme. Two LRBs are implemented. The first one supports sequential data fill, while the data is transferred to the second instance within a single cycle. Inspired by [13], the second LRB enables parallel weight and input access at any cycle of the computation stage, as opposed to SRAM macro sequential cycle-by-cycle access. The first LRB can be accessed through an APB bridge, or through Direct Memory (DM) mapping, which is implemented by redesigning the address decode block of the CPU. Since it degrades density and enlarges the decoding stage logic, it is limited in capacity and is specifically intended for frequently accessed input I and output A data. DM halves the data movement latency between the processor and the accelerator.

III. Silicon Measurements

The BumbleRISC (BBR) chip was implemented in 18 nm FD-SOI CMOS technology. It supports clock frequencies up to 200 MHz at a 0.6 V supply voltage. The total die area of the chip is 5.3 mm^2, and includes an on-chip SRAM of 1,024 kB.

Fig. 6 shows the test setup and PCB, highlighting the BBR SoC. Data, supply voltages, control signals, and measurements are provided by a PC and interfaced to the PCB.

Twenty-four packaged dies were tested, each characterized to determine its minimum operating voltage. Fig. 7a presents an histogram of the minimum operating voltages, which are centered around 0.6 V. Reported power and energy values are averaged across the full die population and measured at 100 MHz using an on-board ADC. Fig. 7b shows the mean power consumption and corresponding standard deviation for each stage of the FAM dataflow, including External Memory Access.

Measurements results are compared with other embedded Transformer accelerator works in Table I. References [14], [15], and [16] heavily rely on sparsity to demonstrate high energy efficiency. Alternatively, [17] exploits quantization and dynamic number representation for the same objective. The architecture in [16] uses a compute-in-memory scheme to achieve significant processing speed at the cost of power and energy consumption. Reference [14] explores a similar quantization scheme as in our work. However, their solution is specifically tailored for a unique Llama model.

TABLE I: Benchmark Table.

	This work	SlimLLAMA [14]	BROCA [17]	MEGA.mini [15]	D3TA [16]
Technology (nm)	18 FD-SOI CMOS	28	28	28	28
Applications/ Models	NLP/Object Detection	NLP/Llama Model	Text - Speech processing	NLP	NLP
Arch. Features	Flash Attention Core	Sparse-Quant.	Quant. Op.	Sparse-Dense Op.	CIM Sparse-Quant.
Supply voltage (V)	0.6	0.58-1.0	0.7-1.1	0.57-0.9	0.65-1.1
Frequency (MHz)	100-200	25-200	50-200	10-250	25-400
Die Area (mm^2)	5.3	20.25	20.25	12.96	18.63
On-Chip Memory (kB)	1,024	500	834	4,146	340
Precision	INT8/INT1 (I/W) FP14 (A)	INT1-16 (W) INT4-8-16 (A)	INT8 (W) INT8-2 (I)	FXP4-8-12-16(W) FXP8-FP12 (I)	INT16-4
Sparsity Assumption	None	52.2% - 87.5%	None	N/A	70%(W) 90%(A) 50%(O)
Power (mW)	3[a]	4.89-82.05	52.4-559.2	153	399- 3,787.8
Energy Eff.[b] (TOPS/W)	85.2[a]	153.6	2.9 - 31.3	13.1 [c]	4.1-38.9
Energy (mJ/Inf.)	20.5[a] (BERT-Mini) 248.32[a] (DETR)	28.3 (1bit) 42.3 (1.58bit) 232.1(8bit)	63.85 [d] (GPT-2)	-	389.6 (Bert Base) 1316.9 (Bert Large)

a) Averaged values across 24 dies. b) Normalized to INT8. c) Normalization not possible due to dynamic precision FPX representation used. d) Averaged from reported values.

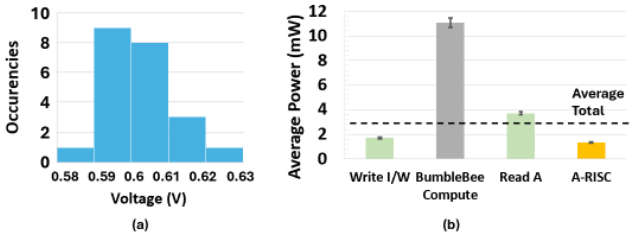


Fig. 7: Silicon measurements: (a) min V_{DD} histogram, and (b) average and standard deviation power consumption for FAM calculation steps and average total on 24 tested die population.

With only 20 and 248 mJ per inference for BERT-Mini and DETR, respectively, our work demonstrates exceptionally low energy consumption per inference while remaining flexible across mapped model types, requiring no assumption of data sparsity and imposing no input-sequence-length limitation thanks to Flash Attention.

IV. Conclusion

In this work, we introduce an 18 nm FD-SOI CMOS Edge AI co-processor integrating a low-power RISC-V CPU with a novel Fused-Kernel Flash Attention accelerator adapted to compact Transformer models. To our knowledge, this represents the first fully implemented silicon realization of such a Flash Attention core, achieving low power consumption at 3 mW while enabling 20 mJ and 248 mJ per inference for BERT-Mini and DETR workloads, respectively. As a task-agnostic implementation centered on the attention mechanism, it supports Transformer models of arbitrary sequence lengths and supports diverse applications, from natural language processing (BERT-Mini feature extraction) to object detection (DETR). This work paves the ways for fused-kernels architectures in Edge AI ASICs, enabling efficient, low-power deployment of conventionally power-hungry systems.

Acknowledgments

The authors thank ASYGN for the SoC top level infrastructure and their assistance during chip measurements. The authors used an AI-based company-private closed-domain language tool to assist in improving the grammar and clarity of the manuscript. All text was originally written and critically reviewed by the authors, who take full responsibility for the content.

References

[1] A. Vaswani *et al.*, "Attention is All you Need," in *NeurIPS*, 2017.
[2] A. R. Sajun *et al.*, "A Historical Survey of Advances in Transformer Architectures," *Applied Sciences*, 2024.
[3] Y. Tay *et al.*, "Efficient Transformers: A Survey," *ACM Computing Surveys*, 2022.
[4] T. Dettmers *et al.*, "Gpt3.int8(): 8-bit matrix multiplication for transformers at scale," in *NeurIPS*, 2022.
[5] H. Qin *et al.*, "BiBERT: Accurate Fully Binarized BERT," in *International Conference on Learning Representations*, 2022.
[6] T. Dao *et al.*, "FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness," in *NeurIPS*, 2022.
[7] S. Clerc *et al.*, "A 18 nm FD-SOI CMOS 6.38 mW 15 fps 8 -bit features 14.8 μJ/inference QVGA road-traffic monitoring Edge AI SoC demonstrator," in *ESSERC*, 2024.
[8] "BERT Mini," https://hf.co/google/bert_uncased_L-4_H-256_A-4, 2019, accessed: 2026-03-11.
[9] "DETR ResNet-50," https://hf.co/facebook/detr-resnet-50, 2020, accessed: 2026-03-11.
[10] H. Wang *et al.*, "BitNet: 1-bit Pre-training for Large Language Models," in *JMLR*, 2025.
[11] J. Cornejo *et al.*, "Exploration of Low-Energy Floating-Point Flash Attention Mechanism for 18nm FD-SOI CMOS Integration at the Edge," in *MWSCAS*, 2024.
[12] J. Saussereau *et al.*, "Asterisc: A size-optimized risc-v core for design space exploration," in *ISCAS*, 2023.
[13] V. Sze *et al.*, "Designing DNN accelerators," in *Efficient Processing of Deep Neural Networks*, 2020.
[14] S. Kim *et al.*, "Slim-Llama: A 4.69mW Large-Language-Model Processor with Binary/Ternary Weights for Billion-Parameter Llama Model," in *ISSCC*, 2025.
[15] D. Han *et al.*, "MEGA.mini: A Universal Generative AI Processor with a New Big/Little Core Architecture for NPU," in *ISSCC*, 2025.
[16] D. Kim *et al.*, "D3TA: 38.9TOPS/W Transformer Accelerator with Dual-Port 3T-eDRAM Digital Compute-in-Memory Using HyperAttention and Triple-Sparsity-Handling," in *ESSERC*, 2025.
[17] W. Jo *et al.*, "BROCA: A 52.4-to-559.2mW Mobile Social Agent System-on-Chip with Adaptive Bit-Truncate Unit and Acoustic-Cluster Bit Grouping," in *ISSCC*, 2025.