LLMMMM: Large Language Models Matrix-Matrix Multiplications Characterization on Open Silicon

Louis Ledoux*†, Marc Casas*†
*Barcelona Supercomputing Center, Barcelona, Spain
†Universitat Politècnica de Catalunya, Barcelona, Spain
E-mail: {louis.ledoux,marc.casas}@bsc.es

Keywords—Large Language Models (LLM), Transformers, Generative Pre-Trained (GPT), Matrix-Matrix Multiplications, Floating-Points, arithmetic, ASIC, Open-Source Silicon (OSS)

I. Extended Abstract

A. Introduction

GPT transformers are useful for various applications, offering significant advancements in natural language processing tasks. However, their operational costs are substantial has shown in prior work which highlights the financial implications of deploying these models [1].

Essentially, matrix-matrix multiplications (MMM), with their intensive data movement and manipulation of arithmetic weights, underscore the computational demands of these architectures. Naturally, these observations are also found in recent efforts within the research community, which have concentrated on devising specialized formats and algorithms aimed at mitigating these costs. These innovations include reducing bit-width exemplified by Machine Learning eXchange (MLX) formats (essentially small floats) [], specialized hardware such as TPUs' systolic arrays [], model pruning of up to 40% [], and more recently, ternary and binary LLMs (see BitNets []).

We introduce a generator of ASIC kernels agnostic to the PDK of MMM units for emerging and small floating-point formats, followed by the evaluation of such units. Concretely, our contributions include the automated generation of circuits for any floating-point format with automated pipelining, a systolic array architecture proposal—these two combined form the foundation of MMM units, a framework to automate the translation from high-level language (Python) to silicon for such matrices (SUF, SuperSet Framework []), the generation of 7 arithmetic formats $\times$ 2 accumulator configurations $\times$ 4 PDKs $=$ 56 chips, and their performance and efficiency evaluation, all provided as open source.

B. Asynchronous and Parallel Compilation

The need for fast generation of specialized circuits is mostly explained by the the known computer science laws being challegend (dennard, moore, the walls, dark silicon era). A recent and emerging solution that allows to follow the pace is OpenSource EDA and its community [**?**]. We adopt this approach and build our own open source tool. Figure 1 depicts this framework which enables the generation of many independant design entries from high level description in python to silicon GDS.

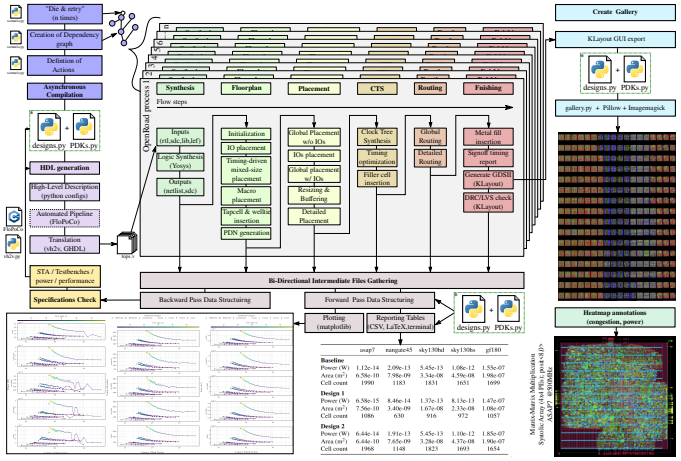

Fig. 1. Schematic Overview SUF: Centralized Management of Asynchronous OpenROAD Forks, Derived from Dependency Task Graphs. This illustration also encapsulates the extended capabilities ranging from Code Generation without manual RTL Writing to Advanced Plotting and Visualization Features.

C. Functional and Performance Specifications

We define and assess four computational formats distinguished by their compactness and mathematical attributes (dynamic range). These include Nvidia's e4m3 and e5m2, and the tapered formats, posit4 (es=0), and posit8(es=2) []. Another significant aspect of our work is the proposal of two variations of internal paths for each of these formats. Internally, we execute the dot product as a fused operation (without rounding) in a fixed accumulator with varying boundaries (bit weights for lsb/msb/ovf). These variations, named α and β, are configured as follows: $(\text{ovf} = 2, \text{msb} = 3, \text{lsb} = -2)$ for an aggregate of 8 bits, and $(\text{ovf} = 5, \text{msb} = 5, \text{lsb} = -5)$ for the 16-bit model. The weights distribution of the embedding layers in the Llama-2-7b model [] dictates these boundaries. All Systolic Arrays of this work are set to $8x8 = 64$ PEs, which ends the defintion

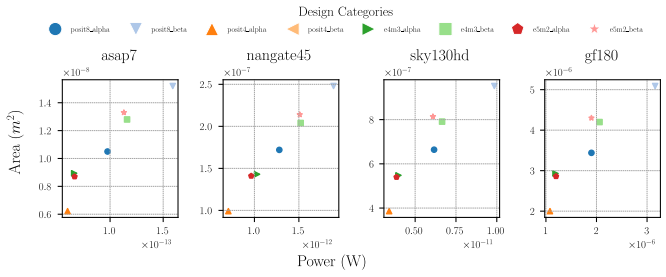

Fig. 2. Area vs. Power

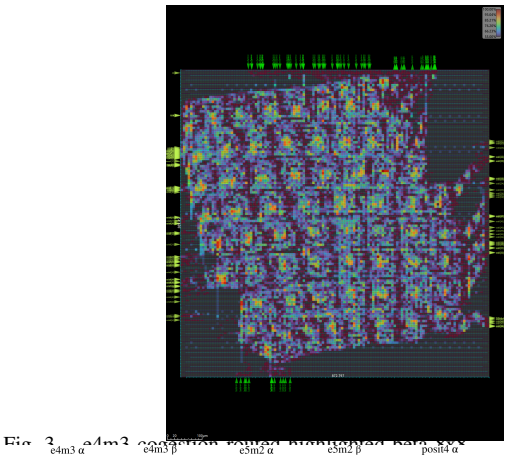

Fig. 3. e4m3 generated arrays

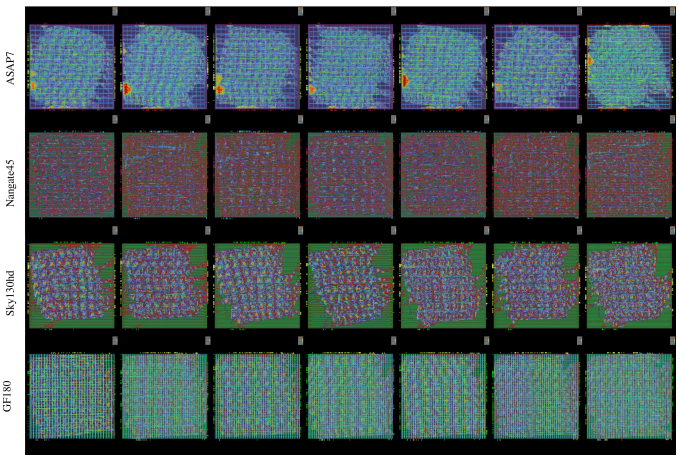

Fig. 4. All generated arrays

of the *Functional* specifications set.

We complement this set with *Performance* specifications that span four accross four PDKs, namely GF180, Sky130hd, nangate45, and ASAP7 which are all open (TBD). Evaluation of several PDKs allow to verify scalability of designs without the use of manual scaling which are often (make it formal and scientific).

D. Results

The evaluation of 14 design entries, with posit 4 beta not working finished in two hours yielding a total of xx chips. Figure 2 shows depicts power and area accross 5 process nodes.

E. Conclusions

F. Biography

Louis Ledoux, originating from a comprehensive computer science background in Rennes (Bretagne, France), has transitioned towards a hardware focus. His journey began with a Bachelor's degree, followed by a Master's in Computer Science, culminating in a one-year internship in 2017, where he explored FPGA virtualization in the cloud. Since 2018, Louis has been engaged in a PhD in computer arithmetic at the Universitat Politècnica de Catalunya and the Barcelona Supercomputing Center, in Barcelona, Spain. His main focus are hardware implementations to address numerical requirements sparsity in HPC workloads. Beyond academia, Louis contributes to the open hardware community, participating in efabless/skywater/Google tapeouts.

II. Acknowledgment

Marc Casas has been partially supported by the Grant RYC-2017-23269 funded by MCIN/AEI/10.13039/501100011033 and by ESF Investing in your future. This research was supported by grant PID2019-107255GB-C21 funded by MCIN/AEI/ 10.13039/501100011033. Els autors agraeixen el suport del Departament de Recerca i Universitats de la Generalitat de Catalunya al Grup de Recerca "Performance understanding, analysis, and simulation/emulation of novel architectures" (Codi: 2021 SGR 00865).

References

[1] A. S. Luccioni, S. Viguier, and A.-L. Ligozat, "Estimating the carbon footprint of bloom, a 176b parameter language model," 2022.