Hardware Accelerators and Accelerated Programming

Special track at
18th International Conference on Advanced Engineering Computing and Applications in Sciences (ADVCOMP 2024)
September 29 – October 03, 2024
Venice, Italy

Biagio Peccerillo
Department of Information Engineering and Mathematics
University of Siena
Via Roma 56, Siena, Italy
e-mail: peccerillo@diism.unisi.it

Abstract—**This paper introduces the role of hardware accelerators in today's computing landscape and summarizes two submissions to the special track "Hardware Accelerators and Accelerated Programming" at the ADVCOMP 2024 conference. These submissions address the growing need for specific hardware modules to accelerate computation in selected domains. The first proposes a Pseudo-Random Number Generator based on a recent cipher and the second proposes a training accelerator for Federated Learning applications on mobile devices, with Differential Privacy functionalities. They both address important topics and propose promising solutions.**

Keywords-hardware accelerators; ciphers; pseudorandom sequences; federated learning; differential privacy.

I. Introduction

With frequency scaling coming to an end and the problems related to "dark silicon" in today's multi-core processors, hardware manufacturers had to shift their focus to hardware accelerators. They were forced to move away from refining general-purpose processors to pursuing specialization in hardware. Thanks to their special-purpose nature, hardware accelerators allow for higher performance and efficiency, as complex and power-hungry structures (e.g., branch predictors, large caches, long pipelines) can be replaced with purely functional units (e.g., replicated processing engines, simple cores). By following the needs of the target family of applications, they allow for exploring original and even *exotic* design solutions.

Today, we are witnessing a flourishing of special-purpose components in our computing systems, being them supercomputers, data-centers, desktop, mobile, embedded, and even wearable devices. To name just the most notable examples, Graphics-Processing Units (GPUs) already manage graphics and data-parallel calculations, not to mention the role of server-class GPUs in training large neural networks. Neural Processing Units (NPUs) are integrated in the Systems-on-a-Chip (SoCs) of our mobile devices, taking care of inference tasks such as speech recognition and photo editing. Smart NICs (Network Interface Cards) accelerate network-packet processing in data-intensive applications in data-centers.

Along with the promised performance and energy efficiency, hardware accelerators carry some risks with them as well. The primary concern is that the associated software layers should allow for a productive exploitation of the accelerator. Otherwise, programmers willing to use them may be forced to adopt abstractions that are far less mature than those available for general-purpose processors. This is a concrete risk: if writing software for a device is too difficult, few programmers will do it, limiting the ecosystem and discouraging adoption. Other challenges include the role of reconfigurability, being it complementary or alternative to programmability; the support of hardware virtualization, which is paramount for a fruitful employment in data-centers; and the relation with the system memory, especially its integration with coherency mechanisms.

However, the promised advantages in performance and efficiency figures surpass the difficulties summarized by the above challenges. There is a tendency to augment virtually every computing system with special-purpose hardware units to reduce the load on the CPU. The two submissions to the special track "Hardware Accelerators and Accelerated Programming" at the ADVCOMP 2024 conference go in this direction.

II. Submissions

In [1], Medaglini et al. present a hardware accelerator for pseudo-random number generation based on the MORUS cipher. Their accelerator, called MORUS-PRNG, is initialized by loading a desired key and the amount of numbers to generate in the accelerator. Then, it generates numbers by repeatedly encrypting the content of an internal counter which is incremented at each generation. Its interface towards the system is achieved with the IXIAM framework, which ensures limited communication latencies in RISC-V systems. They evaluate their proposal in the gem5 architectural simulator, showing that it achieves better performance than the number generators included in the C++ standard library which rely solely on the CPU.

In [2], Mannino et al. propose an idea for a Federated Learning Processing Unit (FLPU), a hardware accelerator for mobile systems aimed at speeding up training in Federated Learning contexts. According to the submission, an FLPU would take care of Differential Privacy at hardware level in a dedicated module. It would constitute an additional layer of security, with the responsibility of adding noise to the trained model and encrypting it before sending it to the server. This way, the noise addition would protect user-data from being deduced by the server, and the encryption would protect the model from potential main-in-the-middle attacks.

Both these submissions address problems of the utmost importance, not only for the research community, but for the general public as well.

Generating pseudo-random numbers is necessary for countless applications in diverse contexts. Adopting an efficient approach as the common way to do it, moving away from CPU to a dedicated hardware accelerator like the one proposed in [1] would save an incalculable amount of hours and energy worldwide.

Moreover, the growing concerns about personal data being used in data-centers with unknown data protection policies necessitates a paradigm shift, for which Federated Learning constitutes a promising direction. However, it necessitates computing power, efficiency, and fulfilling its promise of protecting data privacy. The proposal presented in [2] takes all of them into account.

III. Conclusion

These submissions propose two hardware accelerators for two different purposes: pseudo-random number generation and Differential Privacy-augmented training in Federated Learning contexts. They are two particular examples of a general tendency in hardware manufacturing, which is reducing the responsibilities of the processor in favour of specialized hardware modules, in order to achieve higher performance and energy efficiency. In this regard, these two submissions address important research questions, proposing their original solutions.

Acknowledgement

I would like to thank the conference organizers for their constant support and encouragement in these months. Witnessing this special session steadily come into existence, with all the steps being tackled one after another, has been an enriching experience. A further "thank you" goes to the reviewers: their comments greatly helped the authors of this session to improve the quality of their work. Last but not least, I would like to thank the authors, who sent their original work and, ultimately, made this special session a reality.

References

[1] A. Medaglini, M. Mannino, B. Peccerillo, and S. Bartolini, "MORUS-PRNG: a Hardware Accelerator Based on the MORUS Cipher and the IXIAM Framework," in *Special Track: Hardware Accelerators and Accelerated Programming, along with ADVCOMP 2024*, IARIA XPS Press, 2024.

[2] M. Mannino, A. Medaglini, B. Peccerillo, and S. Bartolini, "Accelerating Differential Privacy Based Federated Learning Systems," in *Special Track: Hardware Accelerators and Accelerated Programming, along with ADVCOMP 2024*, IARIA XPS Press, 2024.