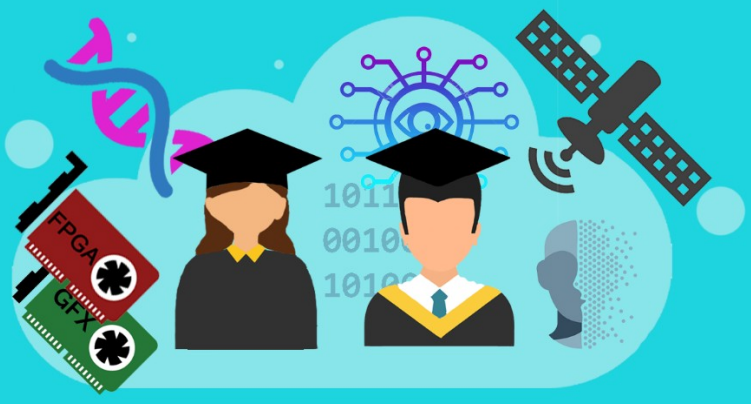
Diploma Thesis

Microproccessors and Digital Systems Laboratory



CXL-Enabled Near-Memory Processing (NMP) Modeling and Design-Space Exploration Using the gem5 Simulator

The growing demand for data-intensive workloads such as AI and scientific computing has exposed the weaknesses of traditional CPU-centric architectures, where large data movement volumes constrains performance and increases power consumption. Near-Memory Processing (NMP) is one of the methods attempting to mitigate it by bringing computation closer to memory, thus reducing data transfer costs and improving bandwidth utilization. Meanwhile, Compute Express Link (CXL) introduces a high-speed, cache-coherent interconnect that allows the usage of memory expanders, and accelerators with memory that the host processor can access in a cache-coherent way, enabling memory pooling and memory sharing between different hosts and accelerators, further optimizing data-movement.

Despite recent progress in NMP and CXL hardware designs, the ability to accurately model, simulate, and analyze such architectures remains limited. Architectural simulators play a major role in early-stage design-exploration before costly hardware implementation. The gem5 simulator provides a modular, extensible, open-source platform for computer architecture research, offering cycle accurate simulation of CPUs, caches, and memory hierarchies, offering an ideal foundation for researching novel design approaches. However, the current gem5 simulator lacks native support for the CXL protocol and NMP modules, but there have been attempts for such support in literature.

The thesis focuses on developing extensions to gem5 to accurately model Near-Memory Processing architectures that leverage the CXL interconnect protocol. This work will implement simulation components that represent CXL-attached memory devices with embedded compute engines, incorporate CXL memory behavior, and support user-defined NMP operations. The resulting framework will enable detailed design-space exploration by evaluating performance metrics such as latency, bandwidth utilization, and energy efficiency under a variety of workload conditions.

PREREQUISITES:

Knowledge of Computer Architecture, Memory Systems, C/C++, Python

RELATED MATERIAL:

[1] Kazi Asifuzzaman, Narasinga Rao Miniskar, Aaron R. Young, Frank Liu, Jeffrey S. Vetter, A survey on processing-in-memory techniques: Advances and challenges Volume 4, 2023, 100022, ISSN 2773-0646, https://doi.org/10.1016/j.memori.2022.100022.

[2] gem5 Project Documentation, *The gem5 Simulator*, [Online]. Available: https://www.gem5.org

[3]Rambus Press, “Compute Express Link (CXL): All you need to know,” Rambus, Jan. 23 2024. [Online]. Available: https://www.rambus.com/blogs/compute-express-link/

[4] Derek Christ, Lukas Steiner, Matthias Jung, and Norbert Wehn. 2024. PIMSys: A Virtual Prototype for Processing in Memory. In Proceedings of the International Symposium on Memory Systems (MEMSYS '24). Association for Computing Machinery, New York, NY, USA, 26–33. https://doi.org/10.1145/3695794.3695797

CONTACT INFORMATION:

Spyros Chalkias, PhD Candidate [schalkias@microlab.ntua.gr]

Prof. Dimitrios Soudris [dsoudris@microlab.ntua.gr]