

Bachelor's / Master's / Semester Project

Leveraging and Optimizing Heterogeneous Computing Systems

The end of Moore's law created the need for **turning computers into heterogeneous systems**, i.e., composed by multiple types of processors that can suit better different types of workloads or parts of them. More than a decade ago, Graphics Processing Units (**GPUs**) became general-purpose parallel processors, in order to make their outstanding processing capabilities available to many workloads beyond graphics. GPUs are key in the recent development of **Machine Learning and Artificial Intelligence**, which took unbearable training times before GPUs. Field-Programmable Processing Arrays (**FPGAs**) are another example of computing device that can deliver impressive benefits in terms of performance and energy efficiency. More specific examples are (1) a plethora of specialized accelerators (e.g., Tensor Processing Units for neural networks), and (2) near-data processing architectures (i.e., placing compute capabilities near or inside memory/storage).

Despite the great advances in the adoption of heterogeneous systems in recent years, there are still many challenges to tackle, for example:

- **Heterogeneous implementations** (GPU, FPGA) of modern applications from important fields such as bioinformatics, machine learning, graph processing, medical imaging, etc.
- **Scheduling techniques for heterogeneous systems** with different general-purpose processors and accelerators, e.g., kernel offloading, memory scheduling, etc.
- **Workload characterization and programming tools** that enable easier and more efficient use of heterogeneous systems.

We are looking for enthusiastic students who want to work hands-on on different software, hardware, and architecture projects for heterogeneous systems.

Requirements

- Outstanding programming skills (C/C++)
- Computer architecture background
- Interest in discovering why things do or do not work and solving problems
- Interest in making systems efficient and usable
- Strong work ethic

For **background and example past studies** please see:

- GPU and FPGA implementations:
 - [Chai: heterogeneous benchmarks \(ISPASS 2017\)](#).
 - [GPU for medical imaging \(CMPB 2020\)](#).
 - [GateKeeper: FPGA for bioinformatics \(Bioinformatics 2017\)](#).
- Scheduling techniques:
 - [Thread scheduling \(MICRO 2011\)](#).
 - [DASH: memory scheduling \(TACO 2016\)](#).
- Programming tools and performance portability:
 - [Boyi: execution models for FPGAs \(FPGA 2020\)](#).
 - [Zorua: hardware support for GPU performance portability \(MICRO 2016\)](#).
 - [Locality descriptor: Cross-layer abstraction to express data locality on GPUs \(ISCA 2018\)](#).

If you are interested, please email **Professor Onur Mutlu** and **Dr. Juan Gómez-Luna**:

omutlu@gmail.com

el1goluj@gmail.com

<https://safari.ethz.ch>

<https://people.inf.ethz.ch/omutlu/>