Student assignment

Type: Graduation assignment
Contact: Barry de Bruin (e.d.bruin@tue.nl)

Topic: Flexible and energy-efficient code compression/decompression for the Blocks CGRA.

Background:
Modern embedded System-on-Chips (SoCs) have a large portion of their silicon area occupied by memory. This large amount of memory has a major impact on the energy-efficiency of an application-specific processor (ASIP). Instruction compression is a common approach to reduce the program binary size and amount of traffic to the instruction cache, thereby potentially saving energy, provided that the decompression overhead is kept under control. Despite a large amount of literature available on instruction compression for common embedded DSPs (i.e. VLIW or TTA processors), the amount of work done on instruction compression for Coarse-Grained Reconfigurable Arrays (CGRA) is modest.

Task description:
In this MSc thesis project you are tasked to investigate, implement and evaluate potential compression schemes and hardware optimizations to support energy-efficient instruction decompression on the state-of-the-art Blocks Coarse-Grained Reconfigurable Array (CGRA). Instruction decompression in Blocks is not trivial since at design time it is not known which instruction decoders operate in lock-step. Therefore the decompression hardware must be flexible and energy-efficient as well.

The expected outcomes of this MSc thesis are:

1. An extensive literature study on existing approaches for instruction compression in comparable VLIW, TTA and CGRA architectures.
2. An evaluation of the most promising approaches using an analytical model or synthesized RTL-level implementation in a modern ASIC technology.
3. An evaluation of the most promising approach implemented within the Blocks CGRA using a set of (available) benchmarks.
4. A detailed report and/or potential paper.

COVID disclaimer:
This project can be fully carried out remotely. The second half of the project requires a stable connection to a TU/e server (over the VPN) to access the ASIC tooling for simulation and/or logic synthesis.
Recommended courses and experience
- Embedded Computer Architecture (ECA).
- Some experience with the Linux command line and RTL coding and some scripting language like Python is a plus.

Interesting references
- “KAHRISMA: A Novel Hypermorphic Reconfigurable Instruction-Set Multi-grained-Array Architecture” – R. Koenig et al. (DATE 2010)