High Performance Computing for heterogeneous Multi-Processor System-on-Chips: A case study in medical imaging applications

Alvin Sashala Naik1, Alberto Garcia Fernandez2, François Duhem2, Philippe Millet2, Eric Lenormand2, Michel Barreteau2, Fabrice Lemonnier2
1) Laboratoire Astroparticule et Cosmologie (APC) UMR 7164, CNRS-IN2P3, Paris, France
2) THALES Research & Technology, STI/LCHP, Campus Polytechnique - 1, avenue Augustin Fresnel, Palaiseau, France

#contact: sashala@apc.in2p3.fr

I. Introduction

The medical imaging use case characteristics:
- The source code is based on a sequential C++ algorithm.
- The software needs to be implemented on a Multi-Processor System-on-Chip (MPSoC) target while optimising all the computing capabilities of the application for real-time image processing.
- Perform extensive architectural exploration for optimal parallel data processing.

II. SLX tool suite

1st approach: SLX Paralleliser
- Analyses C/C++ codes having high level of abstraction and object-oriented dependencies
- Automatic graph call to find parallelism and produces parallelised OpenMP 3.0
- Source-to-source compilation
- Homogeneous targets (ARM-based) or host execution ONLY
- Provides parallelisation hints to the user for possible CPN transcription or OpenMP 4.0 (heterogeneous targets)

2nd approach: Full arch. exploration until target implementation
- Future targets?:
  - FPGA – Virtex, Kintex, Spartan, ZYNQ
  - GPU – AMD, NVIDIA
  - Manycore architectures

SLX tool chain: from raw C/C++ high level of abstraction to SW compilation on heterogeneous elements on an Multi-Processor System-on-Chip (MPSoC) targets:
- CPU – Intel, ARM, PowerPC
- RISC coprocessors
- DSP coprocessors
- C66x processors
- PowerPC processors
- TI Keystone

SLX Paralleliser
1. Analyses C/C++ codes having high level of abstraction and object-oriented dependencies
2. Automatic graph call to find parallelism and produces parallelised OpenMP 3.0
- Source-to-source compilation
- Homogeneous targets (ARM-based) or host execution ONLY
- Provides parallelisation hints to the user for possible CPN transcription or OpenMP 4.0 (heterogeneous targets)

SLX Mapper and SLX Generator
1. Using the hints from the SLX Paralleliser, the user can manually rewrite the C++ code in C and then in CPN.
2. Only then, the user can perform architectural exploration and physical implementation on the SLX catalogue’s targets.
3. The kernel’s parallel execution optimisation can be performed on the SLX mapper’s tool.

E.g. the Parallella:
- Embedded processors:
  1. Zynq-7010 FPGA
  2. Dual-core ARM A9 CPU
  3. Epiphany RISC Manycore Coprocessor
- Features:
  - C/C++ and OpenCL programmable
  - 32-bit IEEE floating point support
  - 512KB on-chip distributed shared memory
  - 32 independent DMA channels
  - Up to 1GHz operating frequency
  - 512 GB/s local memory bandwidth
  - 64 GB/s Network-On-Chip bisection bandwidth
  - 8 GB/s off-chip bandwidth
  - 1.5ns network per-hop latency
  - <2 Watt maximum chip power consumption

The research leading to these results has received funding from the People Programme (Marie Curie Actions) of the European Union’s Seventh Framework Programme FP7/2007-2013/ under REA grant agreement No. [317446] INFIERI “Intelligent Fast Interconnected and Efficient Devices for Frontier Exploitation in Research and Industry”.

Journée Scientifique: Le numérique, parlons-en ! @ Université Paris Diderot, 75013, France [8 décembre 2016]