From COTSon to HLS: Translating Timing into an Architecture

Roberto Giorgi[1], Marco Procaccini[1] and Farnam Khalili[1,2]

[1]Department of Information Engineering and Mathematics - University of Siena
[2]Department of Information Engineering - University of Florence
{giorgi,procaccini,khalili}@diism.unisi.it

Nowadays, the increasing core number benefits many workloads, but programming limitations to exploiting full performance still remain. A Data-Flow execution model is capable of taking advantage of the full parallelism offered by multicore systems [1]–[3]. In such model, the execution can be decomposed in fine-grain threads named Data-Flow Threads (DF-Threads) so that each of them can execute only when their inputs are available [4], [5]. The execution overhead and power consumption is lowered thanks to the reduction of the data push-pull, as well as the burden of thread management.

In a preliminary phase, we explored different solutions through to the COTSon simulator environment [6], which provided us with full system simulation and key metrics, such as OS impact. We compared DF-threads against standard parallel programming models such as OpenMPI, Cilk++ and DSM [7]. To further improve the performance and the power consumption, we investigated a hybrid execution model, which relies both on Field Programmable Gate Arrays (FPGAs) and General Purpose Processors (GPPs). GPP cores allow us to support a large set of applications and FPGAs are known for their reconfigurability and power efficiency.

The FPGA takes care of DF-Threads lifetime by creating, updating and providing them to the Processing System (PS), leaving the GPPs free to execute just DF-Threads code. A key point of our design methodology was to map the timing specification that we tested on COTSon onto the programmable logic via the High-Level Synthesis (HLS) language offered by Xilinx.

HLS permits designers to work at a higher level of abstraction in order to define functions to be translated into the hardware. HLS is progressively prominent in the high performance and energy efficient system domains, shortening time-to-market and offering several solutions for a complex design such as DF-Threads hardware.

The DF-Threads management tasks are translated to the corresponding finite state machines (FSMs), by mostly using the HLS toolhain and optimized for efficiency through the HLS directives and libraries. In order to connect the IP to the GPP, a VHDL-based Register Controller with AXI Lite interface has been deployed, which is re-configurable in terms of register quantities and access priorities. In order to optimize the design, appropriate directives through the provided pragmas by Vivado HLS have been used. This can be helpful to reduce the latency, area and resource utilization as well as improve throughput performance of the resulting RTL design. Furthermore, in order to steer data efficiently between the sub-module, HLS stream libraries have been exploited, which additionally suggest extra options to easily enable and use the interface side channels. We realized our model on a Xilinx ZU9EG platform, using 24% of LUTs, 5% of LUTRAM, 15% of FF and 12% of BRAM. The power consumption estimated by the Xilinx XPE tool is 7.1 Watt for the entire chip and 4.049 Watt for PL.

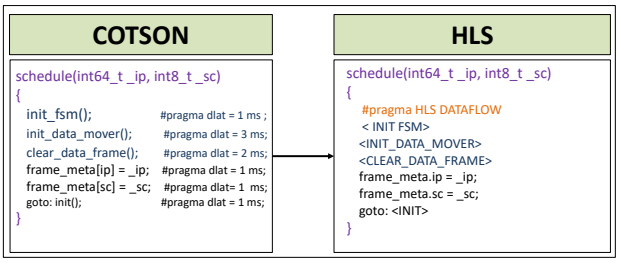

Fig. 1: Translation of the schedule function from COTSon simulator to Xilinx Vivado HLS tool in a nutshell.

References

[1] L. Verdoscia and R. Giorgi, "A data-flow soft-core processor for accelerating scientific calculation on FPGAs," *Mathematical Problems in Engineering*, vol. 2016, pp. 1–21, Apr. 2016. article ID 3190234.

[2] K. Stavrou, D. Pavlou, M. Nikolaides, P. Petrides, P. Evripidou, P. Trancoso, Z. Popovic, and R. Giorgi, "Programming abstractions and toolchain for dataflow multithreading architectures," in *IEEE Proc. Eighth Int.l Symp. on Parallel and Distributed Computing (ISPDC 2009)*, (Lisbon, Portugal), pp. 107–114, IEEE, July 2009.

[3] R. Giorgi, Z. Popovic, and N. Puzovic, "Dta-c: A decoupled multi-threaded architecture for cmp systems," in *Proc. IEEE SBAC-PAD*, (Gramado, Brasil), pp. 263–270, Oct. 2007.

[4] K. M. Kavi, R. Giorgi, and J. Arul, "Scheduled dataflow: Execution paradigm, architecture, and performance evaluation," *IEEE Trans. Computers*, vol. 50, pp. 834–846, Aug. 2001.

[5] R. Giorgi and P. Faraboschi, "An introduction to DF-Threads and their execution model," in *IEEE MPP*, (Paris, France), pp. 60–65, Oct. 2014.

[6] E. Argollo, A. Falcón, P. Faraboschi, M. Monchiero, and D. Ortega, "COTSon: infrastructure for full system simulation," *SIGOPS Oper. Syst. Rev.*, vol. 43, no. 1, pp. 52–61, 2009.

[7] R. Giorgi, "Scalable embedded computing through reconfigurable hardware: comparing df-threads, cilk, OpenMPI and jump," *ELSEVIER Microprocessors and Microsystems*, vol. 63, pp. 66–74, Aug. 2018.