International Symposium on System-on-Chip 2009 was the 11th annual SoC event in Tampere, Finland. This special issue consists of some of the most interesting papers that were selected for inclusion in the journal after some extension and suggested enhancements. At the end there is also a conference report on the SoC 2009 event itself. The special theme is continuing in the next issue of this journal.

Before going to the contents of this issue, I would like to thank editor-in-chief Seppo Virtanen for the opportunity to assemble this special issue. The cooperation could not be any smoother than what it was with Seppo. I express my gratitude also to the reviewers of the extended papers, for their valuable comments and suggestions to further improve the manuscripts.

This issue contains three extensive articles. In the first article, written by David Kammler et al., the challenges of memory interfaces are tackled. In the design of Systems-on-Chip, design and implementation of memory interfaces implied from high-level design decisions may be cumbersome and error-prone. In the worst case the implementations have to be manually adapted to the interfaces of dedicated memories or memory controllers, slowing down the design-space exploration regarding the memory architecture of Multi-Processor System-on-Chip (MP-SoC) devices. By automating the generation of the memory interfaces required, the authors from RWTH Aachen streamlined the design flow and increased the reliability of the resulting interface logic. The work extends an existing architecture description language (ADL). They introduce a new abstract and versatile description format for memory interfaces and their timing protocols in their paper “Automatic Generation of Memory Interfaces” that kicks off the special issue. The feasibility of their approach is proven by real-life case studies including a design space exploration for a banked memory system.

In the second article, by Fabio Garzia et al., the authors evaluate Fast Fourier Transform (FFT) implementations on programmable and reconfigurable architectures. FFT and inverse FFT are intensively applied in modern communication systems based on Orthogonal Frequency Division Multiplexing (OFDM) as their air interface. One challenge is to implement the transforms as flexibly as possible to enable Software-Defined Radio approach to efficiently implement several communication receiver standards on a single device. The researchers of Tampere University of Technology have characterized and compared such flexible implementations on multi-core and coarse-grain reconfigurable architectures implemented on
a programmable logic device. The particular multi-core architecture used in this work is Ninesilica, a nine-core platform consisting of computing clusters based on open-source 32-bit Cappuccino RISC Cores and a hierarchical Network-on-Chip (NoC) interconnecting them. The coarse-grain reconfigurable fabric BUTTER is an 8 x 4 matrix of 32-bit processing elements, local memories, and a Direct Memory Access (DMA) device. The FFT algorithms mapped on these two computing platforms are compared to a single processor, all implemented on the same FPGA device. Both of the parallel platforms show a considerable speed-up in comparison with the single-processor reference architecture. The speed-up is higher in the reconfigurable solution but the MP-SoC provides an easier programming interface that is completely based on C language. The paper “Implementation of FFT on General-Purpose Architectures for FPGA” concludes that their approach of implementing a programmable architecture on FPGA and then programming it using a high-level software language is a viable alternative to designing a dedicated hardware block with a hardware description language (HDL) and mapping it on FPGA.

Di Wu et al. are the authors of the third article. The new 3GPP Long-Term Evolution (LTE) standard is a hot topic among the researchers and engineers in the embedded real-time communication systems domain. It provides an upgrade path for high performance mobile data communications, but by doing that it poses new challenges on the implementation side. In order to provide communication performance for the device, you need computation performance in the advanced (de)modulation, (de)compression, error correction and protocol processing. The authors from Linköping University and Coresonic AB address the LTE modem physical layer implementation by using a parallel processing architecture in their article “System Architecture for 3GPP LTE Modem using a Programmable Baseband Processor.” Their architecture is an application-specific processor adopting the novel Single Instruction Multiple Tasking (SIMT) approach. It is a mixture of ASIC accelerators, an on-chip network, a programmable processor and an efficient memory subsystem. In this work the Multiple-Input Multiple-Output (MIMO) symbol detector and a parallel Turbo decoder have been implemented as configurable ASIC accelerator blocks, and the rest of the modem on the programmable processor. The authors conclude that the chosen architecture provides a feasible solution for the LTE CAT4 modem implementation.

These papers together give a good overview on some of the most interesting aspects dealt with in the Symposium, ranging from design tools and methodology through demanding communications applications down to manufacturing technology related electrical properties. From the SoC 2009 conference report at the end of this issue you will hopefully gain some insight on the special character of the event these papers are originating from. Once again I would like to thank all the contributors of this issue and also all the authors of the Symposium papers that did not make it to the special issue this time. We are planning to continue the fruitful cooperation with IJERTCS in the future as well. Remember to follow up the next issue of this journal which will contain three more papers belonging to this special theme.

Hoping to see you in Tampere – The SoC City – in the coming years!

Jari Nurmi
Guest Editor
IJERTCS