An HdS Meta-Model Case Study: Integrating Orthogonal Computation Models

R. Nane[*], S. van Haastregt[†], T. Stefanov[†], B. Kienhuis[†], K. Bertels [*]

[*] *Delft University of Technology, CE Lab, 2628 CD, Delft, The Netherlands*
[†] *Leiden University, LIACS, 2333 CA, Leiden, The Netherlands*

ABSTRACT

Many advanced consumer products such as mobile phones, set-top-boxes (STB) and digital televisions employ System on Chip (SoC) solutions consisting of integrated circuits and associated software. Typically, SoCs combine specialized IP cores like function specific cores and accelerators, and programmable computing cores like CPUs, DSPs, and ASIPs. The specialized IP cores are controlled by software running on the programmable components, which we refer to as Hardware-dependent Software (HdS). The increasing complexity of today's systems makes integrating the components and associated HdS into a final design more and more time consuming. To cope with the increasing complexity, concepts and tools automating the generation and integration of SoC constituents are being developed by the SoftSoC project consortium. In this context, the CE lab at TU Delft is collaborating with the LIACS institute in Leiden to integrate application specific kernels in a process network based environment. To validate the concepts, we describe a case study with an H.264 decoder application.

DWARV is a C-to-VHDL compiler developed at TU Delft which was designed to target the Molen machine organization that relies on a shared memory model. Daedalus is a system level design toolset developed at LIACS which converts C code to Kahn Process Network (KPN) implementations that rely on a distributed memory model. Daedalus allows a process to be implemented either in software or hardware. For the latter case, an IP core implementing the functional part of a process has to be provided, which can either come from a library or be generated using a C-to-HDL tool. To combine DWARV and Daedalus, which employ different underlying memory models, the designer needs to manually adapt or wrap the IP cores generated by DWARV such that they adhere to the Kahn Process Network semantics. We have developed a method to inform DWARV about the special needs of the required core such that the generation and integration process can be fully automated. For this purpose we use IP-XACT metadata descriptions, which provide a standardized way to specify requirements about software. At the integration side, the cores generated by DWARV are again complemented with IP-XACT metadata describing the generated VHDL. Additionally, an HdS package is generated such that the generated DWARV core can be controlled from software. Since the HdS contains many platform, operating system and application specific parts, we use a 3-layered HdS structure.

As a case study, we have performed experiments with an H.264 decoder application. The top level application is synthesized into a 5-processor software implementation using Daedalus. By implementing the IDCT function that accounts for about 22% of the execution time of the whole

---

[1]E-mail: r.nane@tudelft.nl,k.l.m.bertels@tudelft.nl
[2]E-mail: svhaastr@liacs.nl, stefanov@liacs.nl, kienhuis@liacs.nl

H.264 application using DWARV, we show the robustness of the approach and that even though the two underlying models are orthogonal, they can be automatically integrated using IP-XACT meta models complemented with HdS. Furthermore, by generating a custom hardware IP for the IDCT C-function and integrating this into the H.264 process network generated by Daedalus, we achieved an overall 12.9% application speedup.