Compiler for Real-Time Multiprocessor Systems with Shared Accelerators

University of Twente, Department of EEMCS, Enschede, The Netherlands
g.kuiper@utwente.nl

ABSTRACT
Accelerators are added in real-time multiprocessor systems for power-efficiency improvement and cost reduction. Sharing of these accelerators improves their utilization but without tool support it also complicates programming.

This demonstration shows a multiprocessor compiler for a real-time multiprocessor system that contains support for the sharing of hardware accelerators. The capabilities of this compiler are demonstrated by mapping a packet based GMSK receiver application onto this multiprocessor system. The multiprocessor system is implemented on a Xilinx Virtex-6 FPGA to which an RF front-end is connected. This multiprocessor system contains 16 Microblaze processors and 5 accelerators. With this system a real-time digital audio stream is received and demodulated.

1. REAL-TIME MULTIPROCESSOR COMPIlER
The input of our multiprocessor compiler Omphale [3, 4] is an application that is specified in the OIL language. The OIL language is designed such that modal multi-rate applications can be described intuitively while deadlock freedom, execution in bounded memory and satisfaction of the throughput constraint still remain decidable. A new element in this OIL language is that it allows the specification of parallel modules where the behavior of the parallel modules is specified as a sequential program. Deadlock remains decidable because sequential programs are deadlock free by definition while the communication of streams between modules is unconditional.

A key feature of our multiprocessor compiler Omphale is that it generates a structured VPDF dataflow model [3] and a CTA model [4] besides a task graph. The CTA model enables incremental design of modules because it enables the characterization of the temporal behavior of a composition of modules. The dataflow model is used for checking whether each module meets its throughput constraint. An other purpose of this dataflow model is the computation of the capacities of the circular buffers given a throughput constraint. The sizing of the buffers is done in such a way that the execution of the tasks is pipelined.

2. HARDWARE PLATFORM FOR SHARING OF ACCELERATORS
The main target of our compiler is a real-time multiprocessor system which is implemented on an FPGA. This multiprocessor system contains a shared SDRAM as well as a low-cost communication rings for point-to-point communication between processors and accelerators [2, 1]. Two rings are employed, one ring for data and another ring for communication of credits required for flow-control. The rings are slotted which enables a minimum bandwidth guarantee. A slot stealing mechanism makes the ring work-conserving such that a higher utilization can be achieved.

So-called gateways have been introduced for the sharing of accelerators. These gateways are responsible for multiplexing blocks of data from streams over hardware accelerators. These gateways check that sufficient data and space is available in the buffers before a block is processed. They also check that a pipeline of accelerators is empty before the next block of data is processed. This prevents head-of-line-blocking and enables the use of a dataflow model for temporal analysis. With this dataflow model we compute the required block sizes such that the throughput constraint is satisfied.

3. DEMONSTRATED APPLICATION
The capabilities of the compiler is demonstrated using a GMSK receiver application which is similar to a Bluetooth application. It includes a convolutional decoder as proposed in the Bluetooth long range standardization proposal. This application is executed on the multiprocessor system using 5 accelerators and 2 Microblaze processors. The application processes a data stream with a sample-rate of up to 2 complex mega samples per second. The real-time requirements of this application originate from the ADC converter in the RF front-end and the DAC converter for audio playback.

4. REFERENCES