

DESIGN CONSIDERATIONS OF HETEROGENEOUS MULTIPROCESSOR SYSTEMS

András SZÉLL
Advisor: Béla FEHÉR

I. Introduction

Time-to-market, cost and efficiency drive current embedded system design tool improvements. CAD tools decrease design time from engineer-years to weeks, give guidelines in order to reduce design risk from the very beginning of the design process. Some products also provide automated optimization, validation and simulation even for hardware-software co-design.

SoC (System on a Chip) and SoPC (System on a Programmable Chip) solutions are becoming far more complex than to worth to be designed from scratch, re-usable parts, IP cores and processors are the main building blocks of current designs. Huge functional complexity is often reached by the use of application-specific instruction set processors (ASIP). For different tasks, more processors might be utilized in the same embedded application. These units have their own preferred memory architecture, cache and communication protocols, so integrating them into one system may lead to challenging interfacing issues.

II. Current CAD tools

As functional complexity grows, a higher level of abstraction is necessary in the design process. Current toolsets provide methods for it, and they offer even more. An important engineering task is the hardware-software partitioning: deciding, which functions are better implemented in hardware and which are better in software. For this, hardware budget and speed constraints should also be considered. The intensive research in this field lead to several different optimization algorithms - greedy, hill climbing, binary search, integer linear programming, simulated annealing, tabu search method among others ([1], [2], [3]).

Tools using these profiling and optimization algorithms are now available. With such tools ([5]) it is hard to differentiate between hardware and software: parts of the software can be simply transformed into instruction extensions of the processor executing the application. These extended instructions are calculated only at later stages of the design process, the algorithm description itself does not have to include hardware description. The engineer has the option to modify the optimized and hardware accelerated instructions at a later step. A mixture of system design and algorithm, the SystemC language is another possibility to break the barriers between software code and hardware description.

Processors are good for sequential and control tasks, current complex embedded systems often utilize 2-5 of them instead of a more general, more powerful but less task-specific one. This is due to the strict hardware constraints and power considerations. CAD tools yet do not provide system-wide automated optimization for such designs. Designing communication subsystems, handling shared resources, selecting the best memory architecture for specific heterogeneous multi-processor systems require tools for handling the processors separately as well as handling them together as a system. These demands on CAD tools are yet to be solved ([4]).

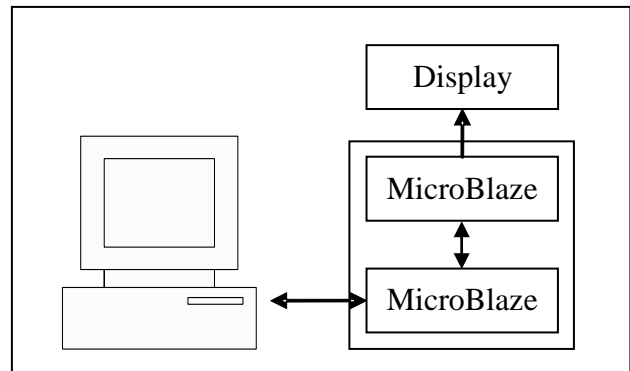
III. Special multiprocessor systems

The task is much easier, if the specialized processors are similar to each other, they use the same architecture and only their special abilities differ. This method is not always acceptable, but in special cases such distribution of resources fits very well.

Small RISC processor cores have several benefits over other solutions: they need less hardware resources, so more of them can be bundled together into a SOPC, their simple architecture makes it easier to integrate them in different projects.

Our tests show that such cores as Xilinx MicroBlaze can be very well utilized for these tasks. It can have up to 8 input and 8 output direct link interfaces (FSL), these are fast dedicated connections to other MicroBlazes or hardware modules. The FSL gives a special possibility to extend MicroBlaze: driving a hardware unit via FSL is just like issuing custom instructions.

The test application consists of two MicroBlazes. One of them (the communication interface) is connected to a PC, it sends and receives signals from the computer. The signals are transformed and passed to the second MicroBlaze over FSL. The second MicroBlaze makes calculations according to data passed to it over FSL, and also handles the display unit. This application was created on a Spartan-III development board. This construction was a simple test of how two (or more) processors can work together, minimizing communication costs and hardware requirements, yet providing much better performance than single-processor SoPC solutions.



For real life examples, high bandwidth (HDTV) MPEG compression was successfully achieved by a farm of MicroBlaze processors ([6]). In this application 1 to 4 PowerPC processors synchronize the MicroBlazes, and the MicroBlazes control the hardware acceleration blocks. The achieved performance of the Virtex-II Pro chip was greater than what could be achieved by conventional DSP solutions.

IV. Conclusion

Embedded applications are becoming more and more complex. To reduce design costs, to improve design efficiency, hardware blocks are re-used, and CAD tools now offer automated design and optimization methods for both software and hardware. Processors designed with these tools can have application specific instruction sets with hardware acceleration. For more complex tasks, often several different processors are used, but communication overhead and design difficulties must be taken into account.

References

- [1] Chao Huang, Srivaths Ravi, Anand Raghunathan, Niraj K. Jha, "Synthesis of Heterogeneous Distributed Architectures for Memory-intensive Applications", *ICCAD'03*, 2003
- [2] Fei Sun, Srivaths Ravi, Anand Raghunathan, Niraj K. Jha, "A Scalable Application-Specific Processor Synthesis Methodology", *ICCAD'03*, 2003
- [3] Petru Eles, Zebo Peng, Krzysztof Kuchcinski, Alexa Doboli, "System Level Hardware/Software Partitioning Based on Annealing and Tabu Search", *Linköping University*, 1997
- [4] Taeweon Suh, Hsien-Hsin S. Lee, Douglas M. Blough, "Integrating Cache Coherence Protocols for Heterogeneous Multiprocessor Systems", *IEEE Micro*, July-August 2004
- [5] CoWare Inc., *LISATek Products*, URL: <http://www.coware.com>
- [6] Xilinx Emerging Standards & Protocols, *MPEG Compression*, URL: http://www.xilinx.com/esp/dvt/prof_brdcst/collateral/mpeg.pdf