The Erlangen Slot Machine: A Highly Flexible FPGA-Based Reconfigurable Platform

Christophe Bobda, Mateusz Majer, Ali Ahmadinia, Thomas Haller
André Linarth, Jürgen Teich
Department of Computer Science 12, University of Erlangen-Nuremberg,
Am Weichselgarten 3, 91058 Erlangen, Germany
{bobda, majer, ahmadinia, teich}@cs.fau.de

Sándor P. Fekete, Jan van der Veen
Institute of Mathematical Optimization, Braunschweig University of Technology
Pockelsstr. 14, 38106 Braunschweig, Germany
{s.fekete,j. van-der-veen}@tu-bs.de

Abstract

We present a new concept as well as the implementation of an FPGA-based reconfigurable platform, the Erlangen Slot Machine (ESM). The main advantages of this platform are: first, the possibility for each module to access its peripheries independent from its location through a programmable crossbar, and distributed SRAMs among slices. This allows an unrestricted relocation of modules on the device. Second, the intermodule structure allows an unlimited communication among running modules.

1. Introduction

Xilinx FPGAs, one of the few partially reconfigurable devices that offer enough logic to efficiently implement resource-demanding applications arising in video, audio and signal processing as well as in other fields like automotive applications. Partial reconfiguration is useful to increase the flexibility in computation and time-sharing of the device space. It requires run-time loadable modules to be compiled and stored as bitstreams, which will then be used to reconfigure the device. The implementation of on-line placement algorithms [1] on actual platform is limited by two main factors. First, very few FPGA platforms exist on which those algorithms can be implemented. Second, the development process of modules for the Xilinx FPGAs is done with many restrictions that make a systematic development process for partial reconfiguration difficult. As illustrated in Figure 1, a module using resources outside its placement area must use connections running through other
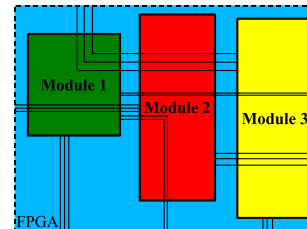

Figure 1. Routing of modules on an FPGA.

modules in order to access its resources. The use of those feed-through lines has two negative consequences: 1) The difficulty of design automation: Each module must be implemented with all possible feed-through lines needed by other modules to reach their resources. 2) Non-relocatable modules: Module accessing pins are no more relocatable, because they are compiled for fixed locations whith a direct access to their pins. Many available systems [3], [4], [5], [2] offer various interfaces, such as audio, video and communication. However, none of them provides solutions to the problems previously mentioned. Each interface is connected to the FPGA using dedicated pins in a fixed location. For example, modules willing to access a given interface like the VGA must be placed in a predefined area of the chip where the FPGA pins are connected to the VGA. Thus, it makes the relocation of this module impossible. The purpose of the Erlangen Slot Machine (ESM) is to overcome the deficiency of existing FPGA platforms by providing high flexiblility of the FPGA platform, in which each component must not be fixed all the time at a given chip location.

2. The ESM Architecture

The ESM is consists of a BabyBoard mounted on a MotherBoard. This separation of the system into two boards allows the BabyBoard that contains the reconfigurable module to be used on a wide variety of systems, without a redesign of the complete system. On the **BabyBoard**, the reconfigurable engine is a Xilinx Virtex II-6000 FPGA, connected to several SRAMs and a configuration circuit. The uniform repartition of resources on the ESM as well as the separation of peripherals from the modules inside the device allow a free relocation of modules. As shown in Figure 2, all the bottom pins of the FPGA are connected to an interface controller (crossbar). The interface devices are also connected to the interface controller. This makes it possible to establish any connection at run-time from one module to its peripheral via the crossbar, no matter where the module resides in the device. Due to the limited capac-
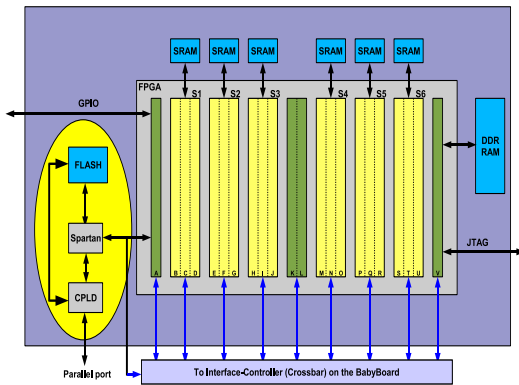

Figure 2. Architecture of the ESM-Baby board.

ity of the available BlockRAMs in FPGA, external SRAM memory is therefore available to allow the storage of large amounts of data by each module as required for applications like video streaming. To allow a module to exclusively access its memory, the SRAM is connected at the top part of the FPGA. In this way, a module has its connections to its periphery through the bottom part. The top part is used to temporally store the computation data. No module needs to feed its lines through the other modules for accessing its resources. The device is divided into a set of exchangeable slots, each of which has access to the SRAM on the top part and to the crossbar at the bottom. The architecture is illustrated in Figure 2. Apart from the main FPGA, the BabyBoard also contains the configuration circuitry. This consists of a CPLD, a configuration FPGA that is a small Spartan II FPGA and a Flash. 6 SRAM banks with 2 MByte each are vertically attached to the board on the top side of the device, thus providing enough memory space to six different slots for temporal data storage.

The **MotherBoard** provides programmable links from the FPGA to all peripheries. The physical connections are established at run-time through a programmable crossbar implemented in an a Spartan chip on the MotherBoard. Besides the run-time programmable crossbar, many peripheries for multimedia and communication are available on the board.

3 Intermodule Communication

In the ESM, we provided three different ways for realizing the communication among different modules: The first one is a direct communication using bus macros between the modules. The second one is the shared memory using the SRAMs or the BlockRAMs. For modules placed in non-adjacent slots, we provide a dynamic signal switching communication architecture called reconfigurable multiple bus (RMB) [6], which allows modules willing to communicate at run-time to set the communication link and exchange data.

4 Conclusion

In this paper, we presented the design and the realization of a highly flexible FPGA-based reconfigurable platform. A main focus has been placed on communication, uniformity of resources, and freedom of placement by means of the crossbar. The running system is available at our institute and can be provided on demand to other researchers working in this area. The ESM can be used in a large variety of fields. In video streaming, a video is streamed picture by picture through a given system and the picture are processed in different parts of the system according to a pipelined structure. In the ESM, each of the modules on the pipeline can be placed on a slot and replaced according to the run-time requirements.

References

[1] A. Ahmadinia, C. Bobda, S. Fekete, J. Teich, and J. van der Veen. Optimal routing-conscious dynamic placement for reconfigurable devices. In *Proceedings of International Conference on Field-Programmable Logic and Applications (FPL)*, volume 3203 of *Lecture Notes in Computer Science (LNCS)*, pages 847–851, Antwerp, Belgium, Aug. 2004. Springer.

[2] Celoxica Ltd. *RC2000 Development Board*, 2004. http://www.celoxica.com/products/boards/rc2000.asp.

[3] X. Corp. http://www.xess.com.

[4] I. Nallatech. http://www.nallatech.com.

[5] M. Platzner and L. Thiele. XFORCES - executives for reconfigurable embedded systems. http://www.ee.ethz.ch/p̄latzner.

[6] R. Vaidyanathan and J. L. Trahan. *Dynamic Reconfiguration: Architectures and Algorithms*. IEEE Computer Society, 2003.