

# A Reconfigurable Platform for Academic Purposes

Christof TEUSCHER<sup>1</sup>   Jacques-Olivier HAENNI<sup>1</sup>   Francisco J. GÓMEZ<sup>2</sup>  
Héctor Fabio RESTREPO<sup>1</sup>   Eduardo SANCHEZ<sup>1</sup>

<sup>1</sup>Logic Systems Laboratory, Swiss Federal Institute of Technology  
CH – 1015 Lausanne, Switzerland

E-mail: {name.surname}@di.epfl.ch, Web: <http://lslwww.epfl.ch>

<sup>2</sup>Escuela Técnica Superior de Informática, Universidad Autónoma de Madrid

E – 28049 Madrid, Spain

E-mail: francisco.gomez@ii.uam.es

## Abstract

*Labomat 3 is a reconfigurable platform for teaching and research purposes developed by our laboratory. The main features of the board are: (1) a microprocessor associated with two mid-range FPGAs, (2) a powerful multitasking real-time operating system including a JavaVM, (3) easy-to-use design tools, and (4) a networking interface. In this paper we describe the hardware and software of the board as well as some application domains.*

## 1 Introduction

It is commonly admitted that the reconfigurability and the increasing density of FPGA circuits [6] are bringing about a true revolution in the world of digital circuits. Naturally, teaching has also taken advantage of all these possibilities, even if in the classical university curriculum there is a sharp distinction between software and hardware design. However, with codesign this clear-cut frontier is dissolving, requiring a fundamental change in the engineering curriculum and the hardware used for teaching. The board presented in this paper has been developed for use by students in any kind of hardware design courses.

## 2 Classical and Reconfigurable Hardware

The Labomat 3 board is a simplified and modified version of the RENCO (Reconfigurable Network Computer) [1, 5] board developed in our laboratory.

The board is divided into two main functional blocks (Figure 1): the processor part and the reconfigurable part. The processor – a Motorola MC68EN360 – is connected to 512 KB of boot EPROM, to 32 MB of DRAM (standard SIMM), and to an additional flash memory (512 KB) where FPGA configurations or user

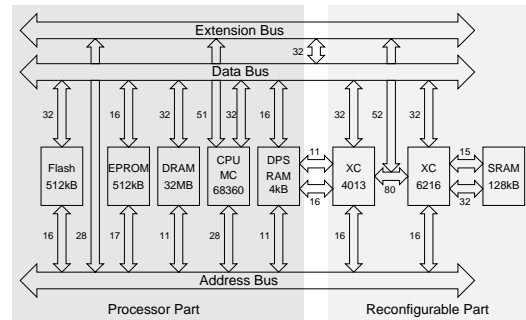


Figure 1: Labomat 3 architecture block diagram

data can be stored. The reconfigurable part consists of a Xilinx XC4013E and a XC6216 [7] FPGA, the latter being connected to a 128-KB SRAM. The processor can access the FPGAs just like peripherals. A large bus of 80 signals interconnects both FPGAs and a subset (52 I/Os) is available on 10-pin connectors. Easy data sharing between the processor and the XC4013 is possible due to a dual-port SRAM whereas Ethernet 10Base-T and RS-232 are used to communicate with the outside world. The bootstrap code is executed from the EPROM whereas the complete operating system and the applications are downloaded from a dedicated server.

## 3 Operating System and Design Tool

The board is running RTEMS [4], a pre-emptive multitasking operating system. It already provides the drivers for Ethernet and a TCP/IP stack, it has been adapted for the 68360 processor, and its source code is available for free. On top of this operating system we set up a Java virtual machine (Kaffe [3]). The entire board can be controlled and programmed via RS-232, Ethernet, or, in the near future, via a web-browser (Figure 2). Thus, students could work at home and communicate with a board installed in our laboratory.

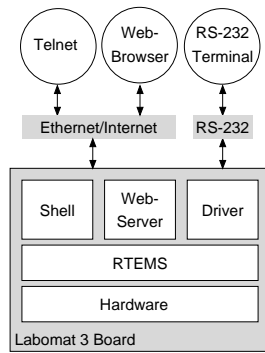


Figure 2: Labomat 3 networking interfaces

The integrated logic design tool RVS6200 we developed takes advantage of the partial and dynamic reconfiguration characteristics of the XC6216 FPGA. The tool is easy to use and directly communicates with the FPGA. The schematic drawn on the screen is in fact an exact representation of the configured hardware, e.g. one can directly configure each cell including the associated connections. The user can impose values and the output values of the design are read back from hardware and displayed in the schematic on the screen.

## 4 Applications for Academic Purposes

### 4.1 A 4x4-bit Multiplier

The design of a 4x4-bit multiplier using the RVS6200 software requires only a short time. A simple 1-bit multiplier must be designed and added to the library. In a second step, this new library element is placed in a new design and assembled to form a 4x4-bit multiplier pipeline. Finally, the software establishes a communication with the Labomat 3 board to configure the FPGA and the user may start testing his design. No simulation is required, as the design is tested on real hardware and the output values are directly read back.

### 4.2 A Simple Pipelined Processor

A VHDL implementation of a simple pipelined processor allows students not only to simulate the design, but to go one step further and test it using real (reconfigurable) hardware. The processor is partially inspired by the DLX architecture [2]: it is an 8-bit load-store architecture with four internal 8-bit general purpose registers and a 5-stage pipeline.

### 4.3 Codesign

Students are asked to design a floating-point unit and to decide which part has to be implemented in hardware and which one in software. The unit can typically be implemented as a coprocessor, i.e. the processor writes the operands into registers in the FPGA

and the latter will compute the result, which in turn is read back by the processor.

## 5 Conclusions

The combination of programmable resources, easy to use software tools, and networking interfaces makes Labomat 3 a system suitable for many applications in teaching and research. Our students' laboratory is currently equipped with fifty Labomat 3 boards which have been successfully used for student exercises. It is possible for students to go beyond simulation, testing their designs with real hardware, allowing them to face problems that do not generally appear in simulation.

For the near future, we are working on tools to assemble and use all the fifty boards in a huge reconfigurable cluster (100 FPGAs) for high-performance experiments, implementing digital neural networks, genetic algorithms, and artificial life experiments.

## Acknowledgments

We are grateful to André Badertscher for the board assembly. This work was supported in part by a grant from the Werner Steiger Foundation.

## References

- [1] J.-O. Haenni, J.-L. Beuchat, and E. Sanchez. Renco: A reconfigurable network computer. In *Proceedings of the IEEE Symposium on Field-Programmable Custom Computing Machines*, pages 288–289, Napa USA, April 1998. IEEE.
- [2] J. L. Hennessy and D. A. Patterson. *Computer Architecture - A Quantitative Approach*. Morgan Kaufmann, 2nd edition, 1990.
- [3] Kaffe. *A free virtual machine to run Java code*. <http://www.kaffe.org>.
- [4] Oarcorp. *RTEMS Real-Time Executive for Multiprocessor Systems*. <http://www.oarcorp.com>.
- [5] E. Sanchez, M. Sipper, J.-O. Haenni, J.-L. Beuchat, A. Stauffer, and A. Perez-Urbe. Static and dynamic configurable systems. *IEEE Transactions on Computers*, Special Issue on Configurable Computing, June 1999.
- [6] J. Villasenor and W. H. Mangione-Smith. Configurable computing. *Scientific American*, 276(6):54–59, June 1997.
- [7] Xilinx. XC6200 Advanced product specification, The Programmable Logic Data Book 1997.