

# Reconfigurable Computing: Applications of Run-time Reconfiguration

Deepti Nipankar-Thakar and Simon Y. Foo

Department of Electrical and Computer Engineering  
FAMU-FSU College of Engineering  
Tallahassee, FL 32310, USA  
foo@eng.fsu.edu

## Abstract

Traditionally, the computations are either implemented in hardware (e.g. ASICs, gate-arrays, custom VLSI) or in software running on the processors (e.g. DSPs, microcontrollers). More recently however FPGAs introduced a new alternative that mixes and matches properties of the available traditional hardware and software solutions. The use of FPGAs for computing has led to a general class of computer organizations that are now commonly known as *re-configurable computing* (RC) architectures. These RC architectures not only allow the computational capacity of the machine to be highly customized to the instantaneous needs of an application but also allow the computational capacity to be reused to meet the changing needs of the application. It can be used to build flexible computing systems that can achieve very high levels of performance. The result changes the traditional ‘hardware’ and ‘software’ boundaries, providing an opportunity for greater computational density and capacity within the programmable media.

Hardware designs offer high performance because they are customized to the problem. They are very efficient and fast due to highly parallel, spatial execution. However, after fabrication, the circuit cannot be altered. This demand forces re-design and re-fabrication of the chip if any part of the circuit requires modification.

Software implementations are far more flexible. The processors execute a set of instructions to perform a computation. By changing the software instructions, the functionality of the system is altered without changing the hardware. However the downside of this flexibility is that they are relatively slow due to the nature of sequential/serial processing and inefficient since operators can be poorly matched to the computational task.

We demonstrate the complete process of run-time reconfiguration (RTR) with three different schemes:

- Run-Time Parameterized Cores
- Virtual hardware
- Partial Reconfiguration

We will describe some example applications developed to demonstrate the different reconfiguration schemes. The tools used are *JBits 2.6* (a Java-based interface for reconfigurable computing) and *SDK2.1* (a java development platform). *JBits*, a software developed by Xilinx, is a set of Java classes that provide an Application Specific Interface (API) to access the Xilinx Virtex FPGA family bitstream. The API has the capability of designing, modifying and dynamically modifying circuits. It can process the

bitstreams generated by Xilinx design tools as well as the bitstreams read back from actual hardware. The tool provides the designer with complete control for individual configuration of all configurable resources like look-up tables, flip-flops and routing.

*Key Words:* Reconfigurable computing, Run-time reconfiguration.