Numerical Computation in Configurable Hardware

A thesis submitted to the Faculty of Science,
University of Colombo
for the
Degree of Master of Philosophy

W. A. S. Wijesinghe

June 2007

Abstract

In order to achieve higher speed performance compared to microprocessor-based systems, the potential and adoptability of programmable hardware devices (especially Field Programmable Gate Arrays) in experimental and computational work have been investigated.

An implementation of a simple microprocessor on a Complex-Programmable Logic Device (CPLD), with a minimal instruction set, optimized for data acquisition applications is discussed first. Due to the optimization features, it was possible to fit both the CPU and the program memory in the 36 macrocell Xilinx XC9536XL CPLD. Reconfigurability of the CPLD makes it possible to change the features of the CPU, including the instruction set, to suit the user requirements. A prototype data acquisition system implemented using this CPU is also described.

The performance of commonly used processing elements - random number generators and multipliers were examined next by implementing them in FPGA hardware. The Linear Feedback Shift Register (LFSR) was found to be the best random number generator in terms of speed and resource utilization. While the embedded multiplier in the Xilinx Spartan 3 was found to be faster than any of the other tested multipliers, MUX-based Carry Save Adder was found to be faster than the multiplier implemented in the standard VHDL library.

The implementation of a simulation based on Diffusion Limited Aggregation (DLA) on the FPGA hardware is studied as the final work. For simplicity, a 128×128 lattice structure was considered. In order to achieve higher speed performance, mathematical operations, specially multiplication operations, were avoided for transformation of lattice cells to memory. The same algorithm was implemented using Visual C++ in order to compare the performance with the hardware based system. The DLA patterns obtained from both methods have similar features. However, hardware system showed about 200% speed performance compared to the software implementation.