ACS Quarterly Report

James Lundblad
U.C. Berkeley

1.0 Hardware Task

1.1 Annapolis PCI and Sun Ultra 30

We have received both the Annapolis Systems FPGA PCI bus card and the Sun Ultra 30 workstation which will host the FPGA card. From the Annapolis documentation it is unclear whether the FPGA PCI bus card supports direct memory access (DMA), although PCI bus mastering is indicated. Without PCI card DMA, preferably including multi-word transfers, data transfer between the host and the PCI card my be a bottle-neck for some applications. Program IO by the host Ultra-Sparc CPU should be avoided if possible. Performance analysis of the Annapolis PCI card will help define the architecture of a proposed next generation FPGA AGP bus card with improved performance.

1.2 Xilinx and Synopsys Tools

We have ordered the Xilinx M1 Alliance tools which provide the back-end netlist place and route as well as third-party interfaces to simulation and synthesis tools from Synopsys among others. The Synopsys VHDL simulator will be used for back-end verification and debug in conjunction with the Annapolis PCI board VHDL models. The license for the Synopsys tools including VHDL simulator and RTL synthesis packages have been renewed.

1.3 AGP Card Proposal

The next generation of reconfigurable computing platforms will utilize the latest 0.25 micron FPGA technology (Xilinx Virtex) utilizing system clock frequencies nearing 100MHz. The processing power of these systems will be limited by 33 MHz and even 66 MHz PCI bus bandwidth. The Advanced Graphics Port (AGP) bus which is currently finding it's way into Pentium II PCs is a good match to the next generation reconfigurable computing technologies. The AGP bus provides performance enhancing extensions to the PCI bus specification leading to bandwidths from 500-1000 MBytes/sec. An AGP card containing 4 x 250K gate Xilinx Virtex FPGAs each with local 32 bit wide SDRAM, plus programmable cross-bar interconnect and AGP interface with dedicated 32 data ports for each of the 4 FPGAs should provide adequate performance for the graphics/video processing applications which could provide the “killer app” for reconfigurable computing.

2.0 Software Task

2.1 Ptolemy Core/Corona Implementation

The Core/Corona architecture has been implemented in the new ACS domain and will be released with version 0.7.1 of Ptolemy sometime in June. A single corona provides the interface definition for each star, while multiple cores provide multiple implementations for each star. Cores are organized into categories based on the implementation they provide. The Architecture is extensible by adding new core categories and/or targets. Currently core categories exist for Floating Point SDF, Fixed Point SDF, and Floating Point C-Code Generation.

2.2 Float/Fixed Point Simulation

Floating point and fixed point core implementations have been developed and demonstrated at the Industrial Liaison Program conference in mid-March. Replacement of floating point particles (aka data tokens) by fixed point particles is supported via type lookup in the target. This allows targets to map any type to any other including float to fix.

2.3 C-Code Generation

A base class infrastructure for code generation cores has been implemented along with a c-code generation core category for floating point data. This code generation capability was demonstrated on a limited basis at the Napa FCCM conference in April. A fixed point version of the c-code generation core category will be derived.

3.0 Module Generators

3.1 BRASS Module Generators in Java

Contact with the BRASS project has been ongoing. We have recently received an alpha release of their module generators written in the Java language. Ron Galicia, a Ph.D. student, has demonstrated a Java code generation capability for the BRASS module generators within Ptolemy II (a version of Ptolemy written in Java). We plan to attend the BRASS project retreat in late June at which time we hope to demonstrate Java/BRASS code generation within the ACS domain.

4.0 Algorithm Analysis

4.1 Automatic Float to Fixed Point Translation

This is the area in which the most activity is planned for the next period of the project. Many possibilities exist for statistical analysis within the ACS domain leading to estimation of the precision required for fixed point realizations of an algorithm. As part of this effort, infrastructure in the ACS domain needs to be developed to allow storage and retrieval of parameter states of any given Ptolemy model. Some preliminary discussions have taken place with Marco Re from the University of Rome regarding collaboration on the float to fixed translation problem.

5.0 Interactions

An overview of the project and demonstrations have been given to representatives from University of Cape Town SA, Phillips Research Labs NL, University of Rome IT, and Imperial College UK. Contact has also been made with the SPARCS project of the University of Cincinnati who we plan to visit before the DARPA review at the end of the month.