



COTS Supercomputing

DRC Harnesses FPGAs

COTS (Commercial Off-The-Shelf) is a term we're accustomed to associating with government or military/aerospace technology. The basic idea is to reduce costs by using readily-available commercial technology wherever possible instead of creating new, purpose-built subsystems. In things like military applications, significant savings can be realized by intelligently substituting COTS technology where it can reasonably do the job.

The field of High Performance Computing (HPC) shares a similar situation. Because high-performance computers are built in very limited quantities, custom-designed subsystems add significantly to the cost of the overall system – even when they're not in the performance-critical path. If we could build supercomputing systems with as many COTS components as possible, we would stand a chance of significantly cutting the cost of all those teraflops.

A couple of years ago, we talked about the potential of FPGAs as reconfigurable computing elements for specialized tasks in high-performance computing. HPC was perhaps the first industry to hit the processor power wall, switching from monolithic processors to large arrays of parallel processing elements. However, for many computing tasks – particularly those with looping constructs, much more power-efficient computation can be accomplished by implementing parts of the algorithm (or in some cases the entire algorithm) directly in hardware such as FPGAs. If you have, for example, an application that requires large numbers of multiply and accumulate operations that could be executed simultaneously, a properly configured high-end FPGA with hundreds of multipliers can outperform a traditional von Neumann architecture by orders of magnitude – both in speed and in computational power efficiency.

Since the limiting factor in many high-end computing systems is total power (either the supply power available or the total power that can be dedicated to operating and cooling the equipment), the gain in power efficiency from adding reconfigurable elements using FPGAs is key. However, life in reconfigurable

computing land isn't as easy as it might appear at first blush. There are significant challenges to overcome to reach the full potential of reconfigurable hardware in HPC systems.

For example, while you can create parallel processing elements with massive computational power using FPGAs, getting the data into and out of those elements efficiently can run into a bottleneck. If your application is running on a traditional processor, it doesn't help much to have the processor passing data to the reconfigurable element or to tie up a primary bus with all the data traffic to and from the FPGAs. The challenges of creating an architecture that allows the massive data access required by a fully-loaded FPGA without stalling the primary processor or wasting FPGA time are significant.

Also, getting your algorithm into an FPGA (as we all know) is quite a bit more complicated than just whipping out some C code (oh, sorry HPC folks – FORTRAN -- HPC people are used to thinking in terms of sequential algorithms, then parallelizing them afterward to match their computing environment.) The traditional approach to FPGA implementation requires an understanding of logic design and a fully parallel mentality from the start – using hardware description languages like VHDL or Verilog that are, by default, parallel.

Today, there are numerous assaults on the parallel programming problem for FPGA-based reconfigurable computing elements. Companies like Celoxica, Impulse Accelerated Technology, SRC Computers, Mitrionics, Mentor Graphics, and many others have developed technology that attempts (in various ways) to simplify the programming paradigm for programming hardware-based parallel processing elements. These tools use a variety of approaches, ranging from automatically extracting parallelism from good 'ol C and C++ to new programming semantics with the notion of parallelism built right in.

In the HPC industry, suppliers have taken a variety of routes to bring in the benefits of reconfigurable computing. A couple of years ago, we wrote about the Cray XD1, which took advantage of technology acquired from Octiga Bay to add reconfigurable FPGA-based boards to a supercomputer architecture. For various reasons, that solution didn't see widespread adoption. Last year, Cray announced that they were adopting FPGA boards from DRC Computer Corporation as reconfigurable coprocessor modules.

Why is DRC's solution attractive, given the problems of the previous generation of reconfigurable computing solutions? DRC has taken something like a plug-and-play, COTS approach to the problem. They have taken commodity FPGAs – in this case, Xilinx Virtex-class devices, and integrated them onto a simple

board that plugs directly into an open 940-processor socket in a multi-way AMD Opteron system. The HyperTransport-based interconnect gives 12.8GBps communication between the FPGA module and any adjacent processor as well as direct access to DDR memory using a patented DDR controller.

With appropriate subroutines loaded into the FPGA, the Opteron can offload those tasks efficiently to the reconfigurable module – typically gaining anywhere from 10X to 1000X performance depending on the application – compared to the same subroutines executed in software on the Opteron. More important than the computational speedup, perhaps, is the processing power efficiency of the resulting system – usually an additional order of magnitude more than the processing speed gains.

Taking advantage of commodity interconnect (like HyperTransport), commodity processing infrastructure (if you consider Opteron “commodity”) and commodity FPGAs put the processing power of FPGAs in the system at a far lower cost, and with considerably less design overhead than with a custom-crafted architecture. From DRC Computing’s point of view, they can evolve their system to take advantage of the latest and best technologies, independent of any pre-conceived notion of the target system. From the programmer’s point of view, subroutines written for the DRC module should be reasonably portable to future incarnations of the technology.

This week, DRC announced the availability of a new RPU110-L200 that beefs up their offering even more. The new module features up to 2 GB of on-board DDR2 memory, any of three HyperTransport bus interfaces, and upgraded FPGAs including Xilinx’s Virtex-4 LX160 or LX200. This increase in available memory and FPGA capacity adds considerably more computing capability to the already-capable modules.

DRC is partnering far beyond their Cray agreement, however. They have also announced partnerships with software companies like Celoxica, DSPlogic, Impulse Accelerated Technology, Mitrion, and Synplicity. Along with Celoxica, for example, they will provide acceleration for applications in performance-hungry vertical markets like oil and gas exploration. Other markets, such as financial analysis, using processing-intensive algorithms could reap enormous benefits from reconfigurable computing technology, and DRC plans to be ready to capitalize on those. “We can bring significant performance gains, along with major reductions in power, for customers that are at the limits of their present systems,” says Larry Laurich, CEO of DRC Computers.

In addition to partnering with software companies that can help in application

development, there is likely to be a number of pre-packaged routines developed by third parties for the DRC module. The portability of the board to various system configurations from multiple vendors, combined with its extremely low cost of acquisition (significantly less than \$5K for a Virtex-4 LX60 module, with prices expected to drop further based on the FPGA market) make it attractive for IP developers targeting their algorithms to specific end markets.

This COTS approach to supercomputing acceleration is likely to be a boon for companies like DRC as well as a lucrative market for the FPGA industry. With more and more supercomputing installations hitting limitations based on available power and cooling, the efficiency of reconfigurable computing elements based on FPGAs is extremely attractive. Expect to see this scenario repeated and expanded dramatically over the next few years.

by Kevin Morris, FPGA and Structured ASIC Journal

July 10, 2007