A 25 GFLOPS/Watt Software Programmable Floating Point Accelerator

Andreas Olofsson, Roman Trogan, Oleg Raikhman
Adapteva Inc.
{andreas, roman, oleg}@adapteva.com

Abstract

This paper presents a hybrid approach to high performance embedded computing that uses FPGAs, general purpose processors and a novel floating point accelerator to push the limits of energy efficiency while keeping with today's well known programming languages and tools. A floating point accelerator has been designed, containing 16 independent ANSI C-programmable processors cores, a high throughput Network-On-Chip and low power FPGA data links. The accelerator chip demonstrates a processing efficiency of 25 GFLOPS/W, and a maximum sustained performance of 32 GFLOPS while operating at 1 GHz.

Technology Trends

Whether we are talking about deeply embedded signal processing systems or stationary high performance computing platforms, the need for performance is virtually limitless and energy efficiency is the number one concern for any advanced application. Battery capacity is not growing quickly enough and electricity delivery costs are becoming prohibitive as we scale HPC systems to Exascale range. To make matters worse, the 40 year CMOS scaling trend of getting more transistors at less power is saturating.[1] General purpose processors, while easy to use, have energy efficiencies that are simply insufficient for leading edge applications. FPGAs have made amazing strides logic density in recent years and today serve as front end processing in the majority of high performance embedded system. However, FPGAs are generally not very efficient at the complex floating point algorithms found in the middle and back end processing and they further require significantly more development effort than an software based signal processing. To address the growing gap between processor energy efficiency and performance needs, a number of specialized architectures have been invented in recent years[2]. However, for the most part, these architectures have either been too difficult to program or have not had sufficiently high energy efficiency and have thus not gained the critical mass needed to survive.

Diverse HPC Proposal

We propose a diverse computing platform, illustrated in Figure 1, wherein FPGAs continue to do the high throughput front end signal processing, but the complex back end floating point signal processing is moved away from the general purpose processors into a specialized floating point accelerator.

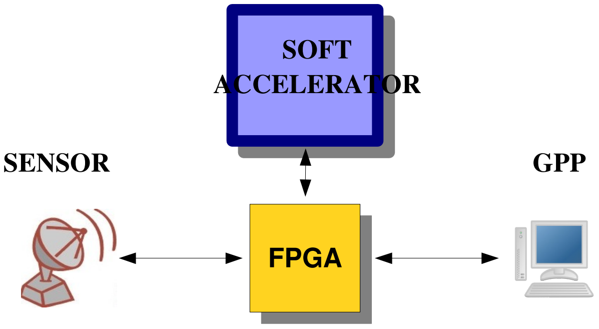

Figure 1. Diverse HPC System

In the hybrid system, the FPGA serves as the data path controller, receiving data from the front-end sensors, moving data around in the system, controlling then floating point coprocessor, and moving data to and from DRAM. The floating point accelerator is responsible for the complex floating point kernels that cannot easily be converted to fixed point implementations, including large FFTs, matrix decomposition, and matrix multiplication. The general purpose processor meanwhile runs tasks such as the highly complex information extraction, data management, and user interfacing. These tasks generally involve a great deal of legacy code and need heavy duty operating systems such as Linux or Windows.

Accelerator Architecture

An overview diagram of the 16 core floating point accelerator chip is shown in Figure 2.

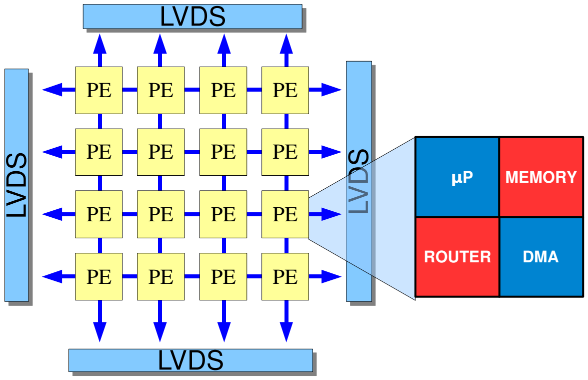

Figure 2. Floating Point Accelerator

The accelerator has 16 software programmable floating point processors that can run any C-code out of the box and has a single unified 32 bit memory map , with each core having read and write access to the complete address map. The simplified memory model was designed to maximize memory bandwidth while minimizing power consumption and simplifying the programming model. The Network-On-Chip connects the 16 cores and the external links together,enabling transparent data movement between resources. Four external LVDS based links are used to interface between the accelerator and an FPGA, allowing for data movement of up to 5GB/sec. The accelerator chip was implemented in a high performance 65nm process, has a maximum performance of 32 GFLOPS at 1 GHZ and an energy efficiency of up to 25 GFLOPS/W.1[1]

Processing Element

It is well documented how little of general purpose processor power is dedicated to doing actual useful work and how much improvement can be had by optimizing a processor for a certain task.[3] By minimizing the waste in the processor and focusing solely on floating point signal processing performance we were able to significantly improve the efficiency of the embedded processors while not sacrificing the C-programmable nature that is a hard requirement for many projects. The chosen processor architecture is fairly straightforward dual issue RISC core designed to run fast with minimal power consumption. Traditional high performance processors and DSPs have multi issue and SIMD techniques to improve the efficiency but at the cost of severely limiting maximum performance in all but the most straightforward parallel applications. The performance boost in tis new accelerator comes not from the parallel issue power from the processing element but from the larger number of cores that can be put down on a die due to the smaller effective size of each processing element, The result is a processing fabric that can be scaled from 4 to 1024 cores in a 65nm technology node with a fixed processing efficiency of 25 GFLOP/W.

Network-On-Chip

The availability of enormous on-chip bandwidth presents an interesting paradigm shift for the programmer. Table 1 shows a comparison between the performance of 16 microprocessors connected through an on-chip mesh vs a board based interconnect grid. As you can see, the speed, performance, and efficiency of the on chip network is orders of magnitude higher, opening up the exciting possibility of medium grained parallelization without suffering communication bottlenecks.

	On Chip Mesh	On Board Mesh
Total BW	4 Tb / sec	512 Gb / sec
Latency	1 ns / hop	10-1000 ns /hop
Power Efficiency	40 Tb / W	60 Gb / W

Table 1. Communication Network Comparison.

External Interfaces

A significant challenge for chip processor manufactures has always been in choosing the right interconnect interfaces to please the most customers. FPGA vendors has done a great job in this area by providing programmable multi standard IO with a completely programmable core fabric, allowing them to connect to every standard imaginable. The proposed floating point accelerator interfaces to the FPGA using LVDS signaling and a custom communication protocol designed to minimize latency and transfer power consumption. The highly configurable FPGA translates between he accelerator transactions and any standard interface, such as DDR, RapidIO, PCIExpress, and Gigabit Ethernet.

Performance Benchmark

Table 2 illustrates the performance of some common signal processing kernels running on the proposed floating point processing architecture, using all 16 cores simultaneously to work on the signal processing kernel described. The benchmarks do not include the IO latency.

	Execution Time
1024 Point FFT	2.4 µs
128x128 MATRIX MULTIPLY	150 µs
128 TAP FIR FILTER (1024 SAMPLES)	9.6 µs

Table 2. DSP Kernel Performance Benchmarks

Summary

We have presented a new floating point accelerator that pushes the envelope in energy efficiency while supporting an easy to use C programming model and standard open source tools.

References

[1] Horowitz M., et al. “Scaling, Power, and the Future of CMOS” IEEE International Electron Devices Meeting, December 2005”

[2] Martinez D, Bond R, Vai M. “High Performance Embedded Computing Handbook: A systems Perspective”, CRC Press,2008

[3] William J. Dally, et al "Efficient Embedded Computing"IEEE Computer, July 2008.

[1] Performance numbers are based on worst case SPICE simulators running FFTapplication code at 110 °C. Silicon measurements to be ready in July 2010