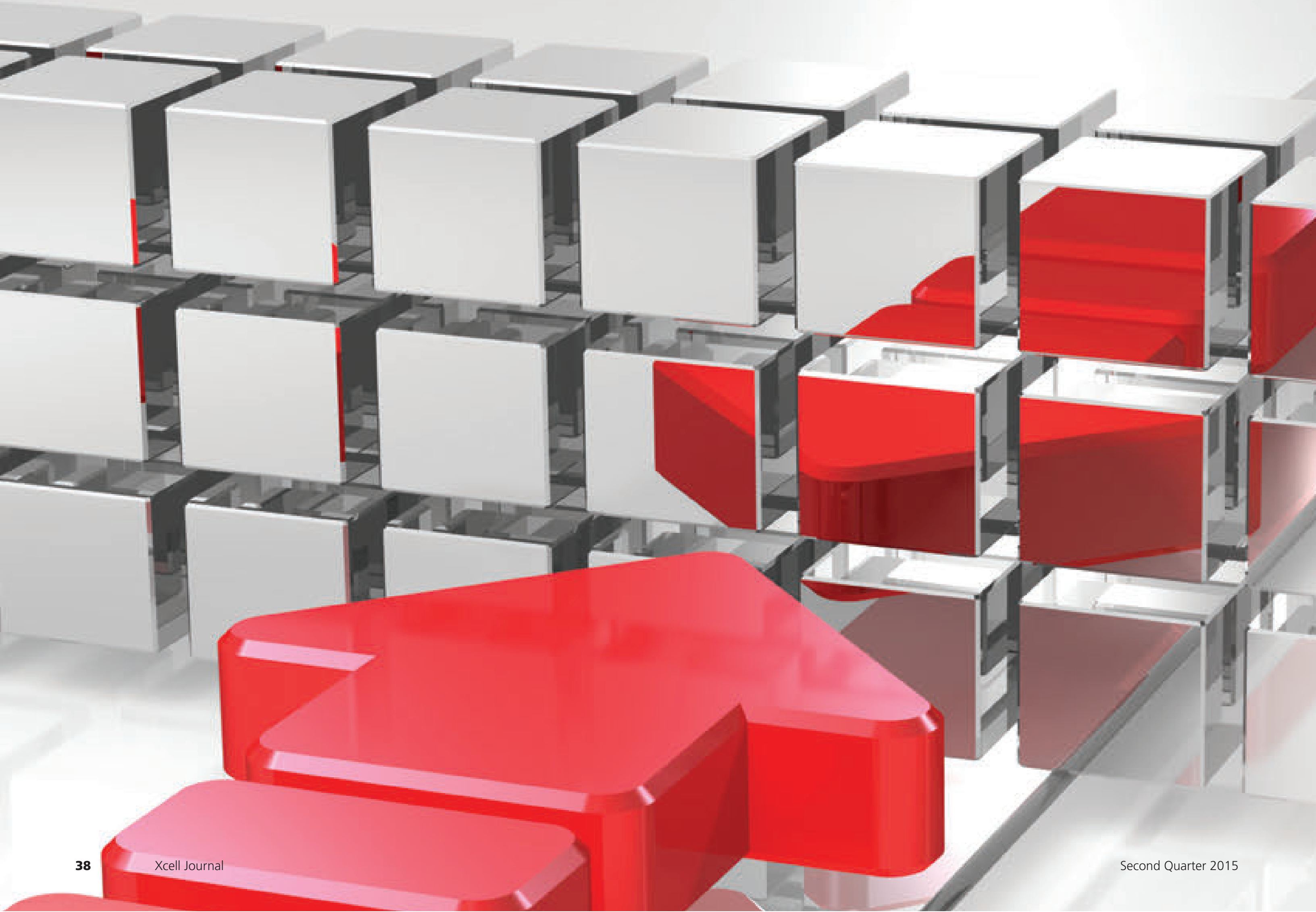
Removing the Barrier for FPGA-Based OpenCL Data Center Servers

Xilinx's SDAccel environment delivers a CPU-like development and run-time experience on FPGAs, easing the data center design burden.



by Devadas Varma
Senior Engineering Director
SDAccel and Vivado High-Level Synthesis
Xilinx, Inc.
dvarma@xilinx.com

Tom Feist
Senior Director
Design Methodology Marketing
Xilinx, Inc.
tfeist@xilinx.com

Data centers today are the backbone of the modern economy, from the server rooms that power small to midsize organizations to the enterprise data centers that support U.S. corporations and provide access to cloud computing services. According to the Natural Resources Defense Council, data centers are one of the largest and fastest-growing consumers of electricity in the United States. In 2013, U.S. data centers consumed an estimated 91 billion kilowatt-hours of electricity—enough to power all the households in New York City twice over—and are on track to reach 140 billion kWh by 2020 [1]. Clearly, lowering power is essential for the scaling of data centers to improve reliability and lower operating costs.

Data center servers vary, depending on the server application. Many servers run for long periods without interruption, making hardware reliability and durability extremely important. Although servers can be built from commodity computer parts, mission-critical enterprise servers often use specialized hardware for application acceleration, including graphics processing units (GPUs) and digital signal processors (DSPs). Now, many companies are looking to add field-programmable gate arrays (FPGAs) for their highly parallel architecture and relatively low power consumption. Xilinx®'s new SDAccel™ development environment removes programming as a gating issue to FPGA utilization in this application by providing developers with a familiar CPU/GPU-like environment.

IMPROVING PERFORMANCE/WATT

Public clouds such as Amazon Web services, Google Compute, Microsoft Azure, Facebook and China's Baidu have huge repositories of pictures and require very fast image recognition. In one implementation, Google scientists created one of the largest neural networks for machine learning by connecting 16,000 computer processors into an entity that they turned loose on the Internet to learn on its own. The research is representative of a new generation of computer science that is exploiting the availability of huge clusters of computers in giant data centers. Potential applications include improvements to image search, speech recognition and machine language translation. However, leveraging CPUs alone is not a power-efficient approach to data center design. For higher speed and lower power, alternative solutions are required.

Baidu, China's largest search-engine specialist, turned to deep-neural-network processing to solve problems in speech recognition, image search and natural-language processing. The company quickly determined that when neural back-propagation algorithms are used in online prediction, FPGA solutions scale far more easily than CPUs and GPUs while also reducing power [2].

The new generation of 28nm and 20nm high-integration FPGA families, such as Xilinx's 7 series and UltraScale™ devices, are changing the dynamics for integration of FPGAs into host cards and line cards in data center servers. Performance per watt can easily exceed 20x that of an equivalent CPU or GPU, while offering up to 50x to 75x latency improvements in some applications over traditional CPUs.

Teams with limited or no FPGA hardware resources, however, have found the transition to FPGAs challenging due to the RTL (VHDL or Verilog) development expertise needed to take full advantage of these devices. To resolve this issue, Xilinx has looked to the Open Computing Language, or OpenCL™, for a way to ease the programming burden.

OPENCL CODE PORTABILITY

OpenCL was developed by Apple Inc. and promoted by Khronos Group [3] precisely to aid the integration of CPUs, GPUs, FPGAs and DSP blocks in heterogeneous designs. To enhance the OpenCL framework for writing programs that execute across heterogeneous platforms, leading CPU, GPU and FPGA vendors, including Xilinx, The SDAccel compiler delivers a 10x performance improvement over CPUs at 1/10 the power consumption of a GPU.

are contributing to development of both the language and its APIs.

The growing acceptance of OpenCL by CPU, GPU and FPGA vendors, server OEMs and data center managers alike is an indication that all parties recognize one overarching fact: C-based compilers for single-processor architectures can only offer small reductions in overall power dissipation within the server rack, even as processors turn to sub-20nm process technologies and add special power-saving states.

OpenCL is a framework for writing programs that execute across heterogeneous platforms consisting of CPUs, GPUs, DSPs, FPGAs and other processors. OpenCL includes a language (based on C99) for programming and an application programming interface (API) to control the platform and execute programs on the target device. OpenCL provides parallel computing using task-based and data-based parallelism.

XILINX SDACCEL DEVELOPMENT ENVIRONMENT FOR OPENCL

Xilinx has worked for nearly a decade on the development of domain-specific specification environments. Concerns about data center performance from both data center managers and server/switch OEMs helped drive one such vertical development toward a unified environment for design optimization in data center applications. The result is SDAccel™, an OpenCL development environment for application acceleration.

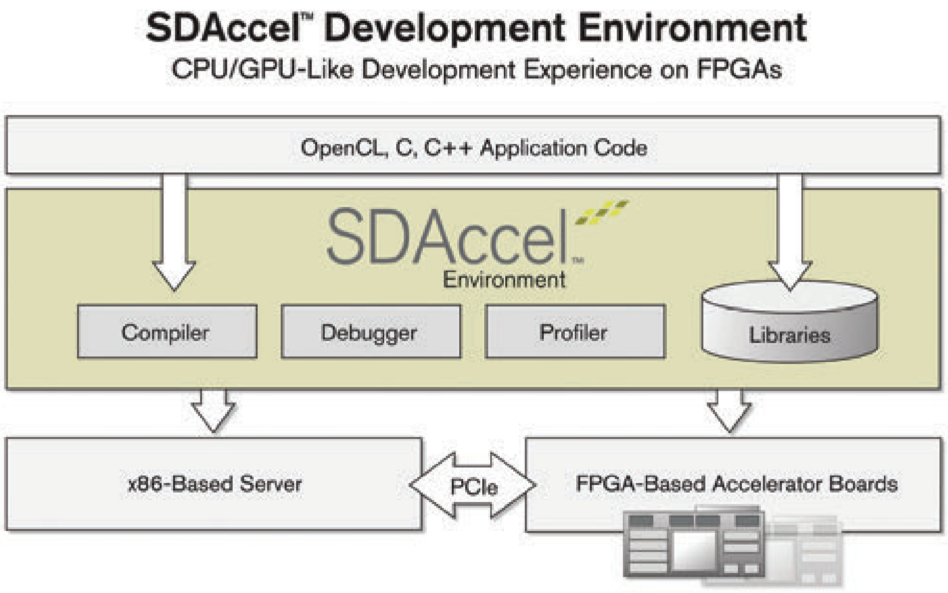

SDAccel™ Development Environment
CPU/GPU-Like Development Experience on FPGAs
OpenCL, C, C++ Application Code
SDAccel™
Environment
Compiler
Debugger
Profiler
Libraries
x86-Based Server
PCIe
FPGA-Based Accelerator Boards

Figure 1 – The SDAccel environment includes an architecturally optimizing compiler, libraries, a debugger and a profiler to provide a CPU/GPU-like programming experience.

The new Xilinx SDAccel environment (Figure 1) provides data center application developers with the complete FPGA-based hardware and OpenCL software. SDAccel includes a fast, architecturally optimizing compiler that makes efficient use of on-chip FPGA resources along with a familiar software-development flow based on an Eclipse integrated design environment (IDE) for code development, profiling and debugging. This IDE provides a CPU/GPU-like work environment.

Moreover, SDAccel leverages Xilinx's dynamically reconfigurable technology to enable accelerator kernels optimized for different applications to be swapped in and out on the fly. The applications can have multiple kernels swapped in and out of the FPGA during run-time without disrupting the interface between the server CPU and the FPGA for nonstop application acceleration.

SDAccel's architecturally optimizing compiler allows software developers to optimize and compile streaming, low-latency and custom datapath applications. The SDAccel compiler supports source code using any combination of C, C++ and OpenCL and targets high-performance Xilinx FPGAs. The SDAccel compiler delivers as much as a 10x performance improvement over high-end CPUs and one-tenth the power consumption of a GPU, while maintaining code compatibility and a traditional software-pro-

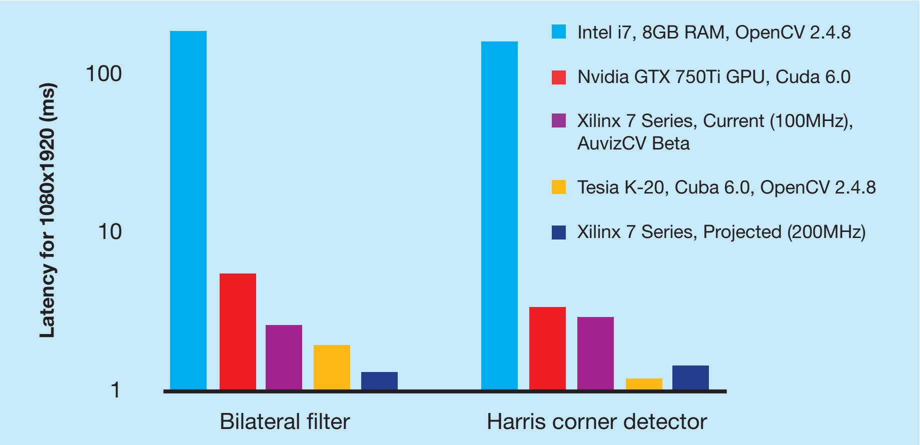


Figure 2 – Video-processing algorithms written in OpenCL for CPU, GPU and FPGA architectures run faster on FPGAs.

(Benchmarks performed by Auviz using the AuvizCV library)

gramming model for easy application migration and cost savings.

SDAccel is the only FPGA-based development environment that includes a wide variety of FPGA-optimized libraries for application acceleration. This library includes OpenCL built-ins, arbitrary-precision data types (fixed-point), floating-point, math.h, video, signal-processing and linear algebra functions.

On real-world computation workloads such as video processing with complex nested datapaths, it is clear that the inherent flexibility of the FPGA fabric has performance and power advantages when compared with the fixed architectures of CPUs and GPUs. As shown by the benchmarks seen in Figure 2, the FPGA solution compiled by SDAccel outperforms the CPU implementation of the same code and offers performance competitive with GPU implementations.

Both the bilateral filter and the Harris corner detector were coded using the standard OpenCL design paradigm in which data between kernels is transferred using device global memory. The FPGA implementation generated by SDAccel optimizes memory access by creating on-chip memory banks that are used for high-bandwidth memory transfers and low-latency computations. The creation and usage of these application-specific memory banks represent some of the architecturally aware capabilities of the SDAccel compiler.

SOFTWARE WORKFLOW

FPGAs have long held the promise of increased algorithm performance at a lower power envelope than CPU and GPU implementations. Until now, that promise has been gated by the programming paradigm required to effectively use FPGAs. SDAccel eliminates this barrier by supporting a software workflow with in-system, on-the-fly reconfigurability that maximizes hardware acceleration ROI in the data center. SDAccel represents a unique and complete FPGA-based solution that far exceeds the capabilities and ease of use of competing point tools. For more information, visit *http://www.xilinx.com/products/design-tools/sdx/sdaccel.html.*

REFERENCES

1. *http://www.nrdc.org/energy/data-center-efficiency-assessment.asp*
2. *http://www.pcworld.com/article/2464260/microsoft-baidu-find-speedier-search-results-through-specialized-chips.html*
3. *https://www.khronos.org/opencl*