Xilinx SDAccel

A Unified Development Environment
for Tomorrow's Data Center

By Loring Wirbel
Senior Analyst

November 2014

The Linley Group

www.linleygroup.com

This paper examines Xilinx's SDAccel™, a software development environment for OpenCL, C and C++, which is part of the Xilinx SDx™ family. The paper will examine the role of SDAccel in creating more power-optimized compute environments for the data center, and in bringing together CPU/GPU optimized compilation and dynamically reconfigurable accelerators under a common development environment.

The Severe Power Limitations of Tomorrow's Data Center

The most aggressive efforts of microprocessor vendors to shrink feature sizes, move to new FinFET processes, and add special power-saving states to CPUs, have not been enough to solve the power crises of data centers packed with ever denser servers. Even a mega-data center handling standard Internet traffic has struggled to keep power budgets in line with utility company capabilities. When the special demands of high-resolution video, image recognition, deep packet inspection, and parallel algorithm processing are added, it becomes a challenge for a PCI Express board with a standard CPU or GPU to keep within a 25W power budget.

The advent of 28nm and 20nm high-integration FPGA families, such as Xilinx 7 series and UltraScale™, have changed 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 50-75x latency improvements in some applications than traditional processors. FPGAs also offer well-characterized IP cores for high-speed interfaces such as PCI Express, DDR4 SDRAM, 10G Ethernet, and 25/28Gbps serdes, the future building block of 50G and 100G Ethernet.

Baidu, China's largest search-engine specialist, has turned to deep neural-network (DNN) 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 easier than CPUs and GPUs. The 400Gflop Software Defined Accelerator developed by Baidu is based on a Xilinx Kintex®-7 480t-2l PCI Express FPGA board that could be plugged into any type of 1U or 2U server. Under various workloads, Baidu found that the Kintex-7 FPGA boards gave 4x higher performance than GPUs and 9x higher performance than CPUs while consuming 10-20W in real production systems. One limitation of FPGA solutions cited by Baidu, long development times, is a problem which Baidu itself suggests can be solved by just the sort of software tools offered under SDAccel.

Convey Computer, a board-level supercomputer specialist combining x86 CPUs with Xilinx FPGAs, designed a Wolverine accelerator card that could aid content caching for data center servers. Convey has partnered with Dell's data center solutions (DCS) division to offer image resizing solutions (heavily demanded in social media and photo-storage networks) utilizing a dual Virtex®-7 board to speed resizing tasks by 35 to 40 times over an equivalent system using CPUs alone. Again, the key to wider use of the

Convey-Dell solution comes from the ability to customize the FPGA accelerators in higher-level languages.

One owner of large data centers that needs little convincing of FPGA utility is Microsoft, which began an initiative in early 2014 to accelerate Bing search ranking using FPGAs. In a recent keynote speech at the Linley Processor Conference, vice president of Microsoft's server engineering Kushadra Vaid showed results from a production pilot using 1,632 servers with PCIe-based FPGA cards. Relative to unaccelerated servers, Microsoft's implementation resulted in 2x throughput, 29% lower latency, and a 30% cost reduction. Although Vaid showed that ASICs could deliver ultimate efficiency, he stated they simply cannot keep up with rapidly changing requirements. What has been inhibiting broader FPGA use in such data center applications has been the lack of an efficient optimizing compiler and related development environment to match the decades of work on compilers for common CPU and GPU architectures.

Microsoft's focus on Bing search acceleration spotlights the degree to which heterogeneous computing, in which coprocessors are assigned such tasks as document ranking, has risen in importance in the data center to equal that of homogeneous multicore multiprocessing. Developers of compilers originally intended for integer CPUs can add support for coprocessing in a piecemeal fashion, though the heterogeneous elements rarely are as tightly coupled as kernels and CPUs in a unified FPGA architecture. Nvidia's recent move to add OpenCL to its existing parallel CUDA environment indicates that those processor companies who launch compilation tasks with a processor other than a central integer CPU (in Nvidia's case, a GPU) often show more foresight in turning to parallelized languages than those working on either a single-core CPU or multicore CPU design.

Compilers for x86 and other architectures have been tasked with several new parameters for optimization of performance, with power taking precedence over code density in recent years. Intel's C++ compilers (particularly its work with Xeon Phi) and Nvidia's CUDA compilers have attempted to optimize for parallel threads, though neither company has achieved great success in adding coprocessing elements to primary CPUs. Demands on compilers have increased even as processor architectures have grown more complex, with multicore, heavier pipelining, and additional co-processors, both on- and off-chip, among the advanced capabilities compilers are expected to handle.

Xilinx has worked for nearly a decade on the development of domain-specific specification environments. Concerns from both data-center managers and server/switch OEMs on data-center performance helped drive one such vertical development toward a unified environment for design optimization in data-center applications. The result is SDAccel, the Software-Defined Development Environment for Acceleration. SDAccel's compiler technology is built on the high-level synthesis (HLS) technology Xilinx acquired in early 2011 when it bought AutoESL. Xilinx has spent the last 3+ years further developing the technology and shipping it as product to over 1,000 FPGA customers. In parallel, it has also been expanding this technology from C and C++ to now optimizing the compilation of native OpenCL code.

This compiler accomplishes the basics a user would expect of any native OpenCL compiler, such as loop merging, flattening, and unrolling, but also does more advanced optimization of memory usage, dataflows, and loop pipelining options. These optimizations allow customers to rely on the compiler to efficiently take C, C++, or OpenCL directly to FPGA hardware. Xilinx benchmarks show that the compiler can reliably get within 5% of optimized RTL with regards to size and Xilinx states that it has seen some instances where it gets over 20% QoR (Quality of Results) improvement vs. RTL. This allows customers to design and optimize their applications on the x86 processor at an architectural level before synthesizing and debugging on the native FPGA hardware with embedded debug capabilities.

Beyond an Effective Compiler: A GPU-Like Development Environment

While many modern C, C++, and OpenCL compilers are elements of broader development suites, many SDKs are comprised of little more than a set of loosely-linked linear tools, some of which remain centered on ancient command-line interfaces. From its inception, the SDAccel environment was designed to be an interactive, dynamic development environment for analyzing applications, parsing them into CPU and kernel elements while using a target x86 card as a platform for emulating the elements for later implementation in FPGAs. Figure 1 shows a typical design flow in the SDAccel environment.

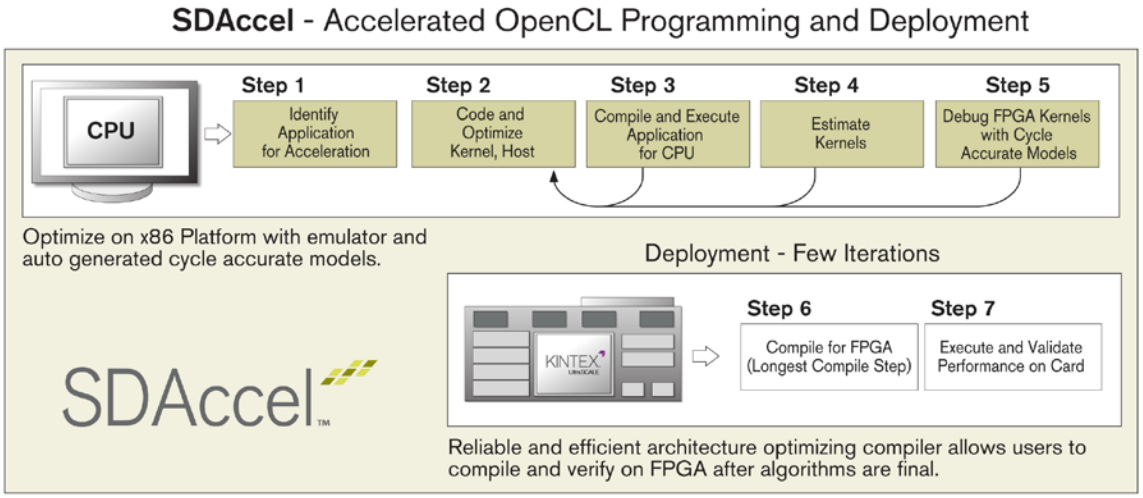

Figure 1: The executable image created through SDAccel compilation can be implemented in Xilinx FPGAs.

The experienced software developer with a minimal experience in FPGA instantiation will find plenty of familiar elements in SDAccel. The goal in allowing for C, C++, and OpenCL inputs was to make the programming environment familiar to those from the standalone CPU world, while insuring that the full unique feature set of FPGAs could be accessed. SDAccel supports full emulation on a CPU, and the programmer can compare parallelized and pipelined elements, and perform such optimization as loop, on-chip

memory, and dataflow operations, all prior to initial compilation for a specific FPGA architecture.

Three levels of compilation allow for easy design iterations to be undertaken for application programming and deployment for the FPGA. First, C, C++ or OpenCL kernel compilation and emulation on the x86 allows design validation and functional debug. Then, the kernel is compiled and simulated using a System C model on the x86 CPU. The final compilation step is for execution on the FPGA device, though SDAccel allows debugging visibility during all three stages.

One software workflow is used across a CPU emulation target and the FPGA implementation target. Acceleration units can be loaded on demand as heterogeneous FPGA elements, and the compiled accelerators can be profiled across the host and kernels. Co-simulations of multiple kernels in one workflow can be iterated several times before committing the design to a chosen FPGA architecture. The advantage of multiple iterations is discussed below, where accelerators for encryption and for image filtering and scaling are able to achieve efficiencies close to hand-coded designs. The use of the accelerator kernels is simplified by the fact that common interfaces used in the FPGA, such as PCI Express and Gigabit Ethernet, are “always on” for reconfiguring the accelerators.

The ability to work in C, C++ and OpenCL offers a unique benefit to the developers since this allows for the “best of both worlds” with regards to using existing code base and leveraging the new OpenCL environment. Although OpenCL offers some distinct advantages with portability and segmentation of code between hosts and compute units to run kernels, the majority of libraries and existing code base that exists is in either C or C++. SDAccel gives customers the ability to continue to develop in C and C++ when appropriate and concurrently leverage OpenCL portability.

SDAccel includes OpenCL built-in, DSP, video, and linear algebra libraries. SDAccel also integrates third party libraries such as OpenCV and BLAS into its software environment. When designs utilize complex coprocessing blocks such as compression or encryption, the result when using the SDAccel development environment can be a design that is on par or even exceeds hand coding in performance and size.

Achieving the Full Advantages of OpenCL

In assessing the competitive landscape for SDAccel, it is important to recognize that the programmable logic industry still dwells primarily in a world of RTL flows. Significant advances in device design can be made within such an environment, of course. Tabula, for example, augmented its existing Stylus RTL compiler with a tool called DesignInsight. This software environment aids in accelerator verification and debug steps through the insertion of embedded points that allow better visibility of an existing design. But without the front-end application of optimizing compilers, DesignInsight works from a first-pass design that has not tightly linked the CPU and accelerator elements under a common compiler scheme.

Open Computing Language, or OpenCL, was developed by Apple Inc. and promoted by Khronos Group precisely to aid the integration of CPUs, GPUs, and DSP blocks in heterogeneous designs. In order to enhance its use in parallel implementation of designs, leading CPU and GPU vendors including Intel, Nvidia, Qualcomm, AMD, Imagination Technologies, and ARM Holdings contributed to development of both the language and its APIs. OpenCL supports both task-based and data-based parallelism. Xilinx and Altera have committed to its use in new designs, though Xilinx also supports C and C++ kernel inputs for compiled designs. Critical to FPGA library elements is OpenCL's definition of task-specific accelerators as kernels that run on OpenCL devices, a key element enabling the joint compilation environment within SDAccel. These kernels are written in standard C, but annotated with constructs to identify memory hierarchy and parallelism.

While Intel and Nvidia both were contributors to Khronos Group, the two companies are still in early stages of offering OpenCL for their own processor architectures, and have done little to open them to partners with co-processing suites. Nvidia has invested much time and effort in CUDA, and understandably has been approaching compiler efforts for OpenCL slowly. Intel has offered Xeon Phi tools, as well as optimization guides for such architectures as Celeron, Xeon, Ivy Bridge, and Sandy Bridge, but looks to third parties for the most part to develop tools that integrate accelerators. It is interesting to note that two of Intel's FPGA partners, Tabula and Achronix, have yet to indicate interest in OpenCL.

Other CPU and GPU supporters of Khronos, including AMD, ARM, and Imagination (MIPS), joined forces with MediaTek and Texas Instruments in 2012 to form the Heterogeneous System Architecture (HSA) consortium, with an explicit purpose of supporting OpenCL as one means of linking CPUs, GPUs, and accelerators. The HSA group has made significant advances in linking OpenCL to existing compiler environments – for example, with the HSAIL language introduced in 2013, and more recent reference work with AMD on HadoopCL, which offloads Hadoop queries to a GPU through the use of OpenCL.

Similarly, universities such as Imperial College of London, Saarland, UCLA, Ohio State University, and many other institutions with leading engineering schools, have initiated projects to parallelize tasks such as scalar graphics, or implementation of scratchpad memory in FPGAs. In almost all cases, however, published papers cover C and C++ efforts in a pre-OpenCL environment. This reflects the fact that more advanced OpenCL functions such as shared virtual memory and pipes, were only publicized at the end of 2013 with the release of OpenCL 2.0.

While Altera has promoted the extension of OpenCL to data plane elements through its use of the OpenCL pipes feature, its own OpenCL compiler continues to treat accelerator code and kernel development as two distinct tasks. Code is developed on an x86 target with an optimization report sent to the designer, while accelerator kernels are designed and prototyped on a virtual FPGA fabric.

The difference in optimizing accelerator kernels can be seen in the comparison of devices for an example compression and streaming encryption benchmarks, shown in Table 1 & 2. SDAccel offers a tripling of throughput in compression benchmark over competitive standard OpenCL compilation for FPGAs, while offering a 4x smaller size for the encryption benchmark at one-third the power. In Table 2, SDAccel shows sevenfold power advantages in HD filtering and image downscaling over an equivalent Nvidia GPU solution. These advantages are achieved by using specific developer optimizations and iterating the design through emulation and simulation steps prior to the final FPGA compilation. This not only minimizes back-end tasks such as place and route, but allows for a coding efficiency matching that of hand-coding.

Application	Metrics	Hand Coded RTL	SDAccel Compilation for FPGAs	Other FPGA Compiler Technology
Compression Benchmark	Throughput	1x	1x	0.3x
	Area	1x	0.9x	3x
Streaming Encryption Benchmark	Throughput	1x	1.2x	1x
	Area	1x	1.25x	5x

Application	Goal	SDAccel Compilation for FPGAs	Other FPGA Compiler Technology	SDAccel Advantage	SDAccel Developer Options
HD Sobel Filter	Highest frames/second	650fps	130fps	5x faster	C-based FPGA optimized libraries
HD Image Downscaling	Highest frames/second	465fps	110fps	4x faster	C-based FPGA optimized libraries

*Data in above table provided by Auviz Systems

Application	Metric	SDAccel Compilation for FPGAs	Nvidia K20	SDAccel Advantage
HD Sobel Filter	Frames/watt	80	11	7x
HD Image Downscaling	Frames/watt	36	5	7x

*Data in above table provided by Auviz Systems

Tables 1, 2 and 3. Comparison of application performance of SDAccel against other FPGA compiler technology for FPGAs (top) and compilation for Nvidia K20 (bottom).

SDAccel's Role in Transcending Data-Center Power Limits

The growing acceptance of OpenCL by CPU/GPU vendors, server OEMs, and data-center managers alike is an indication that all parties recognize that C-based optimizing

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 special power-saving states. This has been evident for some time in applications that make heavy use of DSP and graphics, such as data mining and intelligence, 3D visualization, and parallel algorithm exploitation in microbiology, financial trading, and similar vertical markets.

The broad-based server OEM effort to make heavier use of FPGAs in data centers, however, is an indication that the need for higher integration and lower power in heterogeneous computing is extending to more general-purpose applications such as search-engine queries and Hadoop queries. Efforts to create unified heterogeneous system specifications, such as those of the HSA consortium, are useful, as are such projects as Tabula's RTL compilation and post-design validation/verification, or Altera's use of OpenCL pipes capability in the data plane.

Nothing to date in private industry or academia brings the power of OpenCL to unified FPGA design. The SDAccel Development Environment allows efficient iterations of compilation, emulation with autogenerated cycle accurate kernel models, cosimulation and verification to take place on x86 target boards. The optimized design can then be compiled for the FPGA architecture.

FPGA enabling of the low-power data center will not happen overnight. Promotion of new design methods by Baidu, Dell, Microsoft, and others will help raise awareness of new heterogeneous server design, and the HSA consortium will play a similar role in standardizing the use of OpenCL, perhaps making the parallel language a standard within a few short years.

Getting the most out of an OpenCL FPGA, however, will require more than the point compilation tools offered to date. SDAccel not only brings acceleration kernels closer to traditional CPUs and GPUs than they have been in the past, it also simplifies the task of developing FPGAs for the heterogeneous, low-power data center of the future.

About the Author

Loring Wirbel is a senior analyst at The Linley Group. The Linley Group offers the most comprehensive analysis of the networking-silicon industry. We analyze not only the business strategy but also the technology inside all the announced products. Our weekly publications and in-depth reports cover topics including Ethernet chips, network processors, multicore embedded processors, and wireless base-station processors. For more information, see our web site at www.linleygroup.com.