Embedded FPGA – a New System-Level Programming Paradigm

Achronix® Data Acceleration

WP006 Rev. 1.0 – February 18, 2017 **White Paper**

The current public debate on the future of the semiconductor industry has turned to discussions about a growing selection of technologies that, rather than obsessing on further process geometry shrinks, focuses instead on new system architectures and better use of available silicon through new concepts in circuit, device, and packaging design. Embedded FPGA is the latest offering that promises to be far more than simply a 'better mousetrap.' The emergence of embedded FPGA is, in fact, not only essential at this juncture of the micro-electronics history, but also inevitable. To understand this, a review of the history of FPGA technology is in order.

FPGA Over the Last 30 Years

The pioneers of FPGA programmable logic were Xilinx and Actel in 1984 and 1985, respectively. Over the next decade, products from these companies began pushing TTL, small PLDs and other glue logic off the board. Applications included low-volume industrial designs, prototyping of device and system functions and serving as a highly useful 'band-aids' to patch system designs with a substantial programmable logic resource.

The second FPGA decade began in the 1990's with new market entrants – Altera, Lucent and Agere in 1992, 1995 and 1996 respectively. Networking OEMs such as Cisco and Juniper drove market growth as FPGAs became essential components of routers, switches and other communications equipment. Common applications included traffic management, packet processing offload, NIC and general protocol interface and bridging support. The first FPGA applications also began to emerge in wireless, high-performance computing (HPC) and consumer segments, with FPGA proliferating widely in industrial, scientific and medical segments.

There were two prime instigators in the growth of FPGA usage from 1995–2005: a major shift in the ASIC market and the increasing sophistication of FPGA architectures. The 0.35 µm process node was an ASIC inflection point where mask costs and NREs drove out many smaller customers, leaving low-to-medium volume custom device opportunities up for grabs. Concurrently, FPGA architectures began to embed SRAM blocks, large MACs for lower level DSP functions, hard and soft embedded CPUs, sophisticated configurable I/O and banks of SerDes.

With the inclusion of system-level features in smaller process geometries, FPGAs began to improve markedly in performance, power and cost, which increasingly opened up higher volume sockets that ASIC houses were beginning to leave behind.

Since 2005, FPGAs have proliferated across all electronic sectors. In deep submicron nodes, FPGAs offer performance, density, cost and functionality that provide significant value-add to even high-volume applications. The functional richness of FPGAs has also made segments accessible that were once the exclusive domain of DSPs, GPUs and MCUs.

In today's emerging era of Big Data and Internet of things (IoT), FPGAs are poised to hit another growth inflection point. While IoT designs for sensor boards and access points are emerging that use very low-cost and low-power FPGAs, data center architects have discovered the utility of today's high-end FPGAs as programmable hardware accelerators for applications such as:

- Security
- Packet offload
- Deep packet inspection (DPI)

- Unstructured searches
- Database acceleration
- Machine learning
- Software-defined network (SDN)
- Network acceleration (both wireless and wireline)

The importance of FPGAs to data centers was highlighted by Intel's $16.7B acquisition of Altera, with the strategic goal of developing modules incorporating both Intel CPUs and Altera programmable devices for data center applications. Microsoft announced that their Catapult program, which includes an FPGA in each server to accelerate Bing search, Azure and Microsoft 365, would be expanded to all of their data centers and that adding the FPGA technology increases the compute capabilities of their servers by a factor of two. Beyond Intel-based server systems, the cache-coherent interconnect for accelerators (CCIX) consortium has been formed by ARM, Xilinx, IBM, Huawei, Qualcomm, and AMD to define a standard for a grand heterogeneous computing architecture composed of processors with different ISAs along with FPGAs.

The historical evolution of FPGAs from glue logic and prototyping to a central component of system design has implications for the entire semiconductor industry. Since integration of system-level functions such as IP blocks has been a dominant theme in microelectronics for the last two decades, it is, at this point, a logical supposition to anticipate the FPGA becoming the latest embedded function for SoC and ASIC design.

FPGA and CPU

The claim that FPGA technology is now an appropriate target for embedding in an SoC is a difficult pill for some to swallow, primarily from FPGA's early history of being a poor choice for volume applications because of power, price and performance. Early on, FPGAs were indeed expensive, power-hungry, limited functionality devices by today's standards. This situation is distinctly no longer the case today. As attested by their pervasiveness in system applications across the industry, FPGAs now provide a very compelling value proposition that includes saving power and reducing costs.

Yet in some circles, doubts persist. A common argument put forth is that embedded CPU technology, already proven and widely used in a plethora of specialized architectures for the last two decades, offers all of the programmability needed to meet all requirements. Such arguments, however, are misguided. The CPU-versus-FPGA dichotomy is, in fact, not at all a binary choice. A side-by-side comparison of their functionality reveals why this is the case.

The CPU

There are many variations on the basic load-store/modified Harvard architecture of central processing. MCUs and MPUs execute control-plane applications for system-level administration and housekeeping, while media processors such as GPUs, DSPs and other media processors build on the ALU common to CPU architectures - greatly expanding their functionality in order to support data-plane applications that move large volumes of data and execute more complex arithmetic.

An abundant amount of highly creative features have been developed over the years and liberally sprinkled over these architectures to enhance processor throughput. Logic and registers have been added to support multithreading, instantiate multiple pipelines for superscalar execution to supercharge instructions per cycle (IPC), facilitate out-of-order execution and so forth. Innovations such as single instruction, multiple data (SIMD) and very long instruction word (VLIW) spawned from attempts to exploit data and instruction-level parallelism, respectively.

Yet despite these differences in features, all CPUs share certain characteristics:

- They process tasks in an execution pipeline with stages requiring register operations and logical computations of significant complexity.
- An essential portion of these pipeline operations involve reading and writing data and instructions to and from set-associative cache memories of up to four levels of hierarchy (including scratchpad/tightly coupled memory).
- Pipelines are normally designed to support instructions of eight bits or greater. As such, the tasks they perform (the logic and registers needed to support execution of these tasks) tend to become increasingly complex as word size increases.
- The serial nature of pipeline processing places functional efficiency of the CPU at risk from cache misses and fundamental processing conflicts that create open pipeline slots and potentially long latencies.
- CPUs are intended to accommodate frequent context switching – typically within 100 cycles or less.

The biggest change to processor implementations in the last decade has been the development of multicore architectures. Though there certainly are benefits to this brute-force approach in addressing throughput, multiple threads and data/instruction level parallelism, there are penalties incurred as well. Bus hierarchies, programming and memory support become more complicated while power and cost increase. Furthermore, there are diminishing returns with multicore architectures, as total processing capability does not scale in a linear manner with each added CPU core. Stated differently: adding more CPU cores is not necessarily the best approach.

The FPGA

There are some commonalities between the circuitry of CPUs and FPGAs in that they use a mix of memory and logic to hold and process data and instructions. CPUs use logic, registers and memories to collect data and instructions and then execute a continually changing selection of software tasks in both the control and data planes. The essence of FPGAs consists of memory for configuring several elements: lookup tables, multiplexers and partially populated interconnect matrices. Thus, FPGAs also use a mix of logic and memory to process multiple algorithms.

These similarities, however, are somewhat superficial. Recrafting FPGA fabrics requires streaming new patterns to configuration RAM, which is a time-consuming endeavor. There are devices and techniques which permit faster programming times, especially for partial reconfigurations. Even so, FPGAs are not intended for the kind of quick context switching that is normal for most any CPU.

As a consequence, FPGAs are primarily employed in data-plane applications, emulating digital logic functions to perform frequently used algorithms. They are almost exclusively employed for processing loads that repeat for thousands of cycles or more. In fact, FPGA reconfigurations in-system most frequently take the form of software updates separated by days.

Synergy

The work underway by Intel, Microsoft, and the CCIX group exemplify the complementary natures of CPU and FPGA technology. By using these devices in tandem, CPUs can focus on particularly complex algorithms where they switch threads and contexts quickly and relegate repetitive tasks to an FPGA to function as a configurable hardware accelerator/coprocessor/offload engine. Even using FPGAs and CPUs as discrete devices, systems benefit through increased overall efficiency from the fact that the technologies do not clash, but instead fit together like a hand in a glove.

Speedcore Embedded FPGA (eFPGA)

It is a safe conclusion that embedding FPGA fabrics in SoCs is a natural progression of system integration in the ultra-deep submicron era. As an established FPGA company with a broad customer base, Achronix is uniquely positioned to deliver this capability.

Speedcore™ eFPGA technology is derived from the technology and experience of the Achronix Speedster®22i family of FPGAs, which itself employs industry leading core logic, memory and DSP blocks along with a sophisticated routing architecture. The development of the Speedster22i product line led to Achronix absorbing the skills and methodologies for integrating third-party embedded IP such as PHY and MAC blocks for Interlaken, PCI and 10G Ethernet, as well as SerDes banks, programmable I/O, PLLs and DDR3 controllers, all combined in a device with an FPGA fabric. Furthermore, the Achronix design tools, called ACE, is a proven, complete and robust FPGA development suite, offering all the functionality and capability required by experienced FPGA users, along with the feature support for making reconfigurable programmable fabrics work properly with fixed-function hardware.

By creating an embedded core out of the Speedster 22i programmable fabric, the inefficiencies of device-to-board communications are eliminated. The happy result is that Speedcore die area is 50% smaller than an equivalently sized Speedster22i standalone FPGA.

Advantages from an embedded FPGA instantiation in an SoC cascade to the system level. Board design is simplified though the removal of the discrete FPGA device so that a less expensive board with fewer layers can be employed. Further power and cost reductions are realized at the system level by the removal of a slew of supporting discrete components such as power regulators, level shifters, clock generators, passive components and FPGA cooling. And all the signal integrity issues between the FPGA and neighboring devices are also eliminated.

Most importantly, system performance improves quite dramatically with Speedcore eFPGA technology. By removing discrete FPGAs from the board and using Speedcore IP instead, the FPGA can be part of the CPU subsystem. This placement ensures that Speedcore operations are inherently memory and cache coherent. Embedding FPGA functionality also provides tremendous performance advantages. Speedcore eFPGAs eliminates the need to transmit and buffer signals over a PCB through SerDes-based chip-to-chip interfaces. Instead, there are direct registered interfaces that reduce latency by a factor of ten and increase bandwidth by an order of magnitude.

Engaging with Speedcore IP

The sizing of a Speedcore IP is customized based on an individual customer's needs. Customers supply the quantities of LUTs, embedded memories and DSP64 blocks for their application. Achronix has a tool called Speedcore Builder that creates a model of the Speedcore instance and reports implemented resource counts, die size, power, available pin connections and configuration and test time details. Customers can make adjustments to lock down the exact functionality to meet the requirements for their application.

Once the Speedcore instance is defined, then Achronix can quickly delivers the IP because the Speedcore architecture is modular. The LUT, LRAM, BRAM and DSP64 blocks can be assembled very much like Lego pieces, as their interfaces are standardized. Speedcore Builder takes care of all design rule checking to ensure that the Speedcore IP will deliver optimal performance and device utilization.

Documentation is a rich resource that is part of the deliverables for each unique Speedcore instantiation. Topics related to integrating a Speedcore instance into the SOC are covered including floorplanning, clock topologies, timing closure, verification and testing.

Though SoC companies have their differences in methodologies for timing closure, integration, verification and so forth, a Speedcore instance can comply with such variations and has done so in previous customer engagements. Static timing analysis (STA) issues with individual custom corners, clocking hierarchies, power distribution, testing coverage, etc., have been addressed in these engagements.

The Speedcore business model is a standard schema, conforming to common market expectations for embedded IP. There is a licensing fee for the core, royalties on unit shipments and maintenance charges for the ACE design tools.

The Opening of the Embedded FPGA Era

It is evident that we are entering an era in technology where driving silicon technology into ever deeper submicron geometries will no longer be sufficient to increase performance, reduce power and shrink cost for devices and systems.

By combining discrete FPGAs and CPUs on the same board, industry leaders are pointing the way to value-adding solutions that partition software-driven innovation between conventional processor architectures associated with reconfigurable hardware accelerators acting as offload engines. Pulling such a combination of programmable capabilities into an SoC is the next logical step.

The menu of required methodology, skills and technology to realistically present an embedded FPGA offering is daunting. A company that can offer SoC design expertise, leading FPGA hardware and software technology and the experience of combining them successfully is an exclusive club indeed. Achronix is the only technology vendor that exhibits all of the necessary qualities. Proof of this is actually available in the company's own silicon products, as well as from multiple Speedcore licensees producing SoCs of their own.

Revision History

The following table lists the revision history of this document.

Version	Revision
1.0	Initial released version.

Data Acceleration

Achronix Semiconductor Corporation

2903 Bunker Hill Lane, Suite 101
Santa Clara, CA 95054
USA

Phone : 408. 286.3645
Fax : 408.286.3645
E-mail : info@achronix.com

NOTICE of DISCLAIMER: The information given in this document is believed to be accurate and reliable. However, Achronix Semiconductor Corporation does not give any representations or warranties as to the completeness or accuracy of such information and shall have no liability for the use of the information contained herein. Achronix Semiconductor Corporation reserves the right to make changes to this document and the information contained herein at any time and without notice. All Achronix trademarks, registered trademarks, and disclaimers are listed at http://www.achronix.com/legal and use of this document and the Information contained therein is subject to such terms.